JAIN DEEMED-TO-BE UNIVERSITY | CENTRE FOR DISTANCE AND ONLINE EDUCATION

JAIN DEEMED-TO-BE UNIVERSITY | ONLINE

Master of Computer Applications

[Online Mode]

Student Handbook

Jan 2026 Batch

** The University reserves its right to update / change any part of these regulations as approved by the competent authority*

Table of Content

1. About the Program

This is a Master's Level program, leading to the grant of Master of Computer Applications (MCA) Degree from Jain (Deemed-to-be University). This two-year program offers foundation courses, Elective courses and application-oriented comprehensive projects apart from an option to pursue a cross-functional and open elective course. The core courses incorporate the model curriculum as prescribed for a full-time MCA program. The courses have been designed keeping in mind present- day industry requirements. The program has been structured to provide in-depth knowledge of tools and technologies and their applications in designing and developing Information Technology (IT) centric applications. The program curriculum is intended to enhance a learner’s technical skills and abilities, thus providing a way to accelerate one’s career across key corporate roles in the IT domain.

Programme Educational Objectives

Graduates of the Online Master of Computer Applications programme will:

PEO1: Have a comprehensive understanding of core computer science concepts, enabling them to analyze, design, and develop innovative solutions to complex problems in the field.

PEO2: Demonstrate proficiency in applying data analytics techniques and tools to extract valuable insights from large datasets, facilitating informed decision-making in various domains.

PEO3: Possess a deep understanding of cybersecurity principles, practices, and technologies, enabling them to safeguard digital assets and mitigate cybersecurity risks in dynamic environments.

PEO4: Be equipped with effective communication, collaboration, and leadership skills, enabling them to work in interdisciplinary teams and contribute meaningfully to the advancement of technology and society.

Programme Outcomes:

The Programme Outcomes (POs) for Online MCA Programme:

By the end of the Online MCA Programme, students will be able to:

1. **Computational Knowledge:** Apply a strong foundation in mathematics, computing, and domain-specific knowledge to conceptualize and model solutions for well-defined and real-world computing problems.
2. **Problem Analysis:** Analyze and interpret complex computing problems by integrating core computer science principles and application domain knowledge for data-driven decision-making.
3. **Design and Development of Solutions:** Design, develop, and deploy integrated computing solutions using contemporary tools and emerging technologies.
4. **Investigation of Complex Computing Problems:** Conduct systematic investigations and virtual experiments, analyze digital data sets, and derive evidence-based conclusions to address advanced computing problems.
5. **Innovation and Entrepreneurship:** Foster entrepreneurial thinking and ideation through digital ecosystems and startup culture to develop scalable, sustainable solutions that drive personal growth and societal advancement.
6. **Societal & Environmental Concern:** Demonstrate awareness of and address legal, Environmental, Social, and Governance (ESG) issues in IT responsibly in a connected digital world.
7. **Individual & Team Work:** Demonstrate the ability to work effectively in virtual, cross-functional/multi-cultural team environments.
8. **Communication Efficacy:** Demonstrate effective communication and project management skills by utilizing digital tools.
9. **Life-long Learning & Digital Fluency:** Engage in continuous professional development and self-directed learning using online resources.

Graduate Attributes:

Strong Technical Proficiency	Graduates will demonstrate a high level of technical competence in computer science, software development, and other relevant areas, comparable to international standards.
Problem-Solving Aptitude	Graduates will exhibit exceptional problem-solving skills, with the ability to analyze complex issues and apply innovative solutions in diverse computing contexts.
Critical Thinking	Graduates will have a well-developed capacity for critical thinking, enabling them to evaluate, synthesize, and integrate information from various sources to make informed decisions.
Ethical and Professional Behavior	Graduates will adhere to the highest ethical standards in their professional practice, understanding and respecting the legal and societal responsibilities associated with computing and technology.
Effective Communication	Graduates will possess strong communication skills, both written and oral, allowing them to convey complex technical concepts effectively to both technical and non- technical audiences.
Collaborative Teamwork	Graduates will excel in collaborative team environments, contributing positively to multidisciplinary teams and demonstrating leadership when required.

Adaptability and Lifelong Learning	Graduates will recognize the importance of continuous learning and possess the skills to adapt to rapidly evolving technologies and methodologies throughout their careers.
Innovation and Creativity	Graduates will demonstrate creativethinking and innovative problem-solving, contributing to the development of novel solutions and technologies.
Global and Cultural Awareness	Graduates will be culturally sensitive and aware of global technology trends, fostering an inclusive and diverse perspective in their work.
Entrepreneurial Mindset	Graduates will exhibit an entrepreneurial spirit, identifying opportunities for technological innovation and translating these ideas into practical applications and solutions.
Quality Focused	Graduates will strive for excellence in their work, adhering to international quality standards and best practices in software development, data analysis, and related areas.
Continuous Improvement	Graduates will engage in self-assessment and seek opportunities for personal and professional growth, aiming to consistently enhance their skills and knowledge.

2. Electives Offered

The following Electives are offered:

Table: Choice of Elective

Sl. No.	Electives
1	Computer Science and IT
2	Data Analytics (Accredited by Institute of Analytics (IoA), UK)
3	Cyber Security (Accredited by British Computer Society (BCS), UK)
4	DevOps
5	NLP & LLM Development
6	Full Stack Development
7	Cloud Computing
8	Data Science
9	Artificial Intelligence

Indication of Elective: Candidate is required to indicate his/her Elective at the time of admission. Elective cannot be changed after the commencement of the Program.

3. Admission

Eligibility: Pass in an Undergraduate (Bachelor) Programme of a minimum duration of Three (3) years in any stream from a UGC recognized University, with a minimum aggregate of 50% or an equivalent letter/numerical grade. A relaxation of 5% shall be given to SC/ST candidates.

Candidates who are in the final semester of the Bachelor Programme are also eligible to apply.

Academic Documents: Colour scan of the below mentioned original document is required to be uploaded on the admission portal at the time of seeking admission –

i. Grade sheet of Class 10
ii. Grade sheet of Class 12
iii. Grade sheet of all the Semesters of Bachelor's Degree / Consolidated mark sheet
iv. UG Degree Certificate and UG Provisional Degree Certificate
v. Aadhar Card for Indian Nationals and Passport for Foreign Nationals
vi. ABC and DEB/APAAR ID.

Scans from a photocopy or a faxed copy are not accepted. The University reserves the right to demand hard-copy of the original document as part of the process of verifying the authenticity and may revoke the admission at any time for non-fulfillment of any eligibility requirements.

Admission Intake: There will be two intakes in a year – January Cycle and July Cycle.

4. Duration and Credits

The duration of the MCA Program is 2 years divided into 4 Semesters. The concept of credit is used to define the weightage of a course in the curriculum. Each course earns 4 credits totaling 90 program credits. Courses which include Practical (Lab Programs and Exercise) will have 3 credits of Theory and 1 credit towards Practical.

www.jainuniversity.ac.in
Corporate Office
91/2, Dr. A N Krishna Rao Road V V Puram,
Bengaluru - 560 004
P: +91 80 4343 1000
Centre for Distance & Online Education
319, 25th Main, 17th Cross,
J P Nagar 6th Phase
Bengaluru – 560 078
P: +91 80 4343 0100

4A. Multiple Entry and Multiple Exit Option

In accordance with the provisions of the National Education Policy (NEP) 2020, University Grants Commission (UGC) Regulations and the National Credit Framework (NCrF), learners enrolled in the **MCA (Online)** programme may avail the Multiple Entry and Multiple Exit (MEME) option, subject to fulfillment of the prescribed credit requirements and University regulations.

Table: Distribution of Credits (CS & IT)

Semester	**No. of Core Courses**	**No. of Elective Courses**	**Total Credits**
1	4	0	22
2	4	0	24
3	2	1	20
4	0	4	24

* Includes Specialization Course # Project / Open Elective Course

Table: Distribution of Credits (Data Analytics)

Semester	No. of Core Courses	No. of Elective Courses	Total Credits
1	4	0	22
2	4	0	24
3	2	1	20
4	0	4	24

* Includes Specialization Course

Project / Cross-Functional Elective Course

www.jainuniversity.ac.in

Corporate Office
91/2, Dr. A N Krishna Rao Road V V Puram,
Bengaluru - 560 004
P: +91 80 4343 1000

Centre for Distance & Online Education
319, 25th Main, 17th Cross,
J P Nagar 6th Phase
Bengaluru – 560 078
P: +91 80 4343 0100

Table: Distribution of Credits (Cyber Security)

Semester	No. of Core Courses	No. of Elective Courses	Total Credits
1	4	0	22
2	4	0	24
3	2	1	20
4	0	4	24

* Includes Specialization Course

Project / Cross-Functional Elective Course

Table: Distribution of Credits (DevOps)

Semester	No. of Core Courses	No. of Elective Courses	Total Credits
1	4	0	22
2	4	0	24
3	2	1	20
4	0	4	24

* Includes Specialization Course

Project / Cross-Functional Elective Course

Table: Distribution of Credits (NLP & LLM Development)

Semester	No. of Core Courses	No. of Elective Courses	Total Credits
1	4	0	22
2	4	0	24
3	2	1	20
4	0	4	24

* Includes Specialization Course

Project / Cross-Functional Elective Course

Table: Distribution of Credits (Artificial Intelligence)

Semester	No. of Core Courses	No. of Elective Courses	Total Credits
1	4	0	22
2	4	0	24
3	2	1	20
4	0	4	24

* Includes Specialization Course

Project / Cross-Functional Elective Course

Table: Distribution of Credits (Full Stack Development)

Semester	No. of Core Courses	No. of Elective Courses	Total Credits
1	4	0	22
2	4	0	24
3	2	1	20
4	0	4	24

* Includes Specialization Course

Project / Cross-Functional Elective Course

Table: Distribution of Credits (Data Science)

Semester	No. of Core Courses	No. of Elective Courses	Total Credits
1	4	0	22
2	4	0	24
3	2	1	20
4	0	4	24

* Includes Specialization Course

Project / Cross-Functional Elective Course

Table: Distribution of Credits (Cloud Computing)

Semester	No. of Core Courses	No. of Elective Courses	Total Credits
1	4	0	22
2	4	0	24
3	2	1	20
4	0	4	24

* Includes Specialization Course

Project / Cross-Functional Elective Course

A total of 2700 learning hours ensures that a student has acquired knowledge at par with the face-to-face classroom mode of delivery and learning over the two years.

5. Program Delivery

Mode of Program delivery: Online

Pedagogy: Program delivery follows the prescribed four Quadrants approach resulting in maximum learner engagement. Each course involves 120 hours of learning (1 credit equals 30 learning hours). A combination of the following formats will be used: two-way live online audio-video lectures, pre- recorded audio-video lecture available on the Learning Management System um(LMS), multimedia content, interactions through the discussion forum on the LMS, exhaustive e-content/printed material for in-depth reference, self-study activities that Includes Assignments, Quiz & Multiple Choice Questions (MCQ), Essay-type questions, Case Study etc., individual and group projects, programming exercise, dissertation, Massive Open Online Courses (MOOC) and various experiential learning methods.

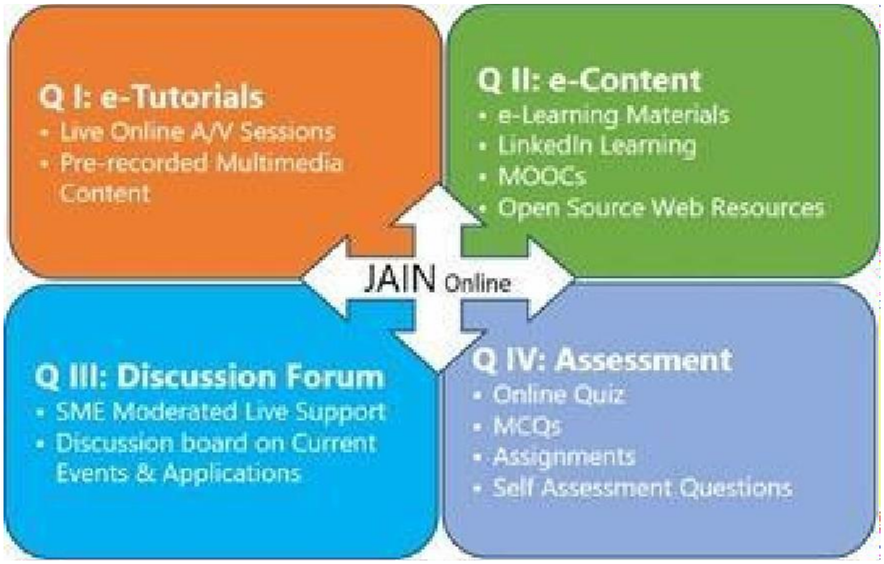

Fig: Four Quadrant Approach Table: Distribution of Learning Hours

Quadrant	Weightage
I	27 %
II	20 %
III	16 %
IV	37 %

University Management System: Our LMS platform has been designed to engage and inspire a learner by providing access to all the learning resources including texts, videos, screencasts and lecture recordings; participate in discussion forums; communicate with the faculty and the program team. A learner has the option to take- up the several quizzes and the MCQs that follow at the end of every unit of the course, track learning progress, submit assignments and much more.

e-Learning Material: A learner is provided access to an exhaustive and customized Self Learning Material (SLM) on the LMS. The SLM will also indicate sources of additional readings and resources available on the internet.

Medium of instruction: The medium of instruction and examination is English.

e-Tutorials: "Two-way live interactive sessions will be scheduled on Saturday for 5-7 hours."

6. Assessment Scheme

The performance of students will be based on Continuous Assessment (CA) and End Semester Examination (ESE) as per the weightage given in following table:

Table: Distribution of Weightage (CA: ESE)

Assessment Type	Weightage (%)
Continuous Assessment	30
End Semester Examination	70
Total	100

The courses with Practical's /Lab Exercises are assessed based on the performance in both theory and Lab exercises in CA and ESE

Assessment of performance in Project is based on – Project Synopsis (Weightage 15%) Interim Report (Weightage 15%) Five-minute Video Presentation on the project (Weightage 20%) and Project Report (Weightage 50%).

Note: The Components of the Project should be submitted in sequential manner only viz., Project Synopsis, Interim Report followed by Project Report after which the Video Presentation is can be submitted.

Continuous Assessment Structure for Lab Courses

CA Component	Type	Marks
CA 1	Program Execution	25
CA 2	Program Execution	25
CA 3	Program Execution	25
Practice Questions	Practice Exercises	25
Total IA marks		100

I. Question Structure and Marks Allocation for Each CA

Part A: 1 out of 2 Question = 10 Marks Part B: 1 out of 2 Question = 15 Marks **Total Marks per CA: 25 Marks**

II. Practice Questions

Number of Practice Questions: 10

Learners must complete the execution of practice questions on the Virtual Lab platform within a duration of 12 weeks.

Question Type	Number of Questions	Marks per Questions	Total Marks
Section 1	5	2	10
Section 2	5	3	15
Total	-	-	25

www.jainuniversity.ac.in
Corporate Office
91/2, Dr. A N Krishna Rao Road V V Puram,
Bengaluru - 560 004
P: +91 80 4343 1000
Centre for Distance & Online Education
319, 25th Main, 17th Cross,
J P Nagar 6th Phase
Bengaluru – 560 078
P: +91 80 4343 0100

Note: All four Lab Continuous Assessment (CA) components are **mandatory**. Learners must score a minimum of **40%** in aggregate to pass the lab course.

Criteria for Continuous Assessment

The assessment scheme is designed not only to assess the attainment of course outcomes by the learner but also to help and guide them to undertake systematic studies. The Continuous Assessment (CA) marks will be awarded based on Three Assignments.

There are three Continuous Assessments. Each continuous assessment will be for 30 marks.

- The first assessment will be scheduled after the completion of Module 2 of the course (Post Week 5). (Questions will be from Module 1 and Module 2).
- The second assessment will be scheduled after the completion of Module 4 of the course (Post Week 10). (Questions will be from Module 3 and Module 4).
- The third assessment will be scheduled after the completion of Module 5 of the course (Post Week 12). (Questions will be from Module 1, 2, 3, 4 and 5 (weightage approx. 50%).
- The average of the best two scores out of the three assessments will be considered as the Continuous Assessment marks.

Assessment Details:

Window Period: According to Calendar of Events

Maximum Duration: 60 minutes in a single sitting (After Login)

Number of attempts: One

Portion: 2 Modules for CA1 and CA2; All 5 Modules for CA3

NOTE: Once you start attempting the assessment, you do not have the option of pausing or completing it partially. You have to complete the entire assessment in a single attempt.

These MCQs will be based on the application of concepts learned (real-life or hypothetical situations) and case studies and will require substantial preparation by the learners.

Each question in the CA and ESE will be carefully mapped to the attainment of Course Outcomes considering the levels as per Bloom's Taxonomy.

Additional Information on Assessment:

1. Each MCQ will have four options of which only one of them will be correct.
2. There will be no negative marking for selecting a wrong response, hence students are advised to attempt all the questions.
3. The assessment will have to be attempted online as per the schedule notified.
4. A learner will be required to take an assignment in a single sitting of a maximum 60 minutes (based on login).
5. The assessment can be attempted only once during the schedule announced using any device.
6. Courses may have a different pattern of CA.

1. End-Semester Examination

Examination mode: Online Proctored Examination

System requirement*: A desktop or laptop computer with a working webcam and microphone facility connected to a stable and non-shared internet connection for the entire duration of the examination.*

Exam Date: The University will conduct end-semester examination for both odd and even Semesters twice in a year typically in the months of June/July and November/December. The month of examination may change as per guidelines issued by University Grants Commission (UGC) and/or relevant authority.

Eligibility: There is no requirement of minimum marks to be scored in Continuous Assessment in a course in order to be eligible to appear for End Semester Examination.

Examination fee and registration: In the first attempt, a learner has to register for all the courses of the Semester by paying the prescribed examination fee. For subsequent attempts, a learner can pay the examination fee on a per course basis. The prescribed examination fee will have to be paid as per due date and is non-refundable nor will be adjusted towards subsequent examinations in case a student does not appear in examinations of any courses.

Exam Duration: 180 Minutes, Single sitting (based on login)

Maximum Marks: 70

Eligibility for Pass: A learner shall be declared to have passed in a course if he/she secures minimum C Grade in that course.

In addition,

i) A minimum of 40% marks in aggregate (Total of scores in Continuous Assessment and End Semester Examination) is required to secure a C Grade; and

ii) A minimum of 25 marks should be scored separately out of 70 marks in End Semester Examination.

Similarly, a learner shall be declared to have passed in Project if he/she secures;

i) A minimum of 40% marks in aggregate.

7. Grading Scheme

The University awards the grades and grade points for each course as per the table below:

Table: Grade and Grade Point

Grade	**A +**	**A**	**B +**	**B**	**C +**	**C**	**F**
Corresponding Grade Point	10	9	8	7	6	5	0
Percentage of Marks	≥90-≤100	≥80-90	≥70-<80	≥60-<70	≥50-<60	≥40 -<50	< 35
CLASS	Outstanding	Excellent	Very Good	Good	Above Average	Pass	Fail

The Semester performance of a student will be indicated as "Semester Grade Point Average (SGPA). The SGPA will be weighted average of Grade Points of all letter grades received by a student for all the Course units in the semester.

The final Grade Card will indicate Cumulative Grade Point Average (CGPA) and shall be based only on Grade Points obtained in courses for which units have been earned

8. Question Paper Pattern

The End-Semester University examination for 70 marks will have the following pattern*-

Section – A: 50 Marks

(A learner is advised to assign 90-120 minutes for this section)

This section will have 50 MCQs of 1 mark each with varying difficulty level. Each MCQ will have four options of which only one of them will be correct. There will be no negative marking for selecting a wrong response, hence learners are advised to attempt all the questions.

Section – B: 20 Marks

(A learner is advised to assign 60-90 minutes for this section)

This section will have 6 descriptive questions requiring descriptive answers of 5 marks each and the students will be required to answer any 4 of them. Generally, a word count of 200 to 300 words is sufficient to provide a satisfactory answer to a 5-mark question.

** Certain Courses may have a different pattern of ESE.*

9. Re-examination Policy

i. A learner has to register by paying a prescribed fee to reappear for End Semester Examination as per notification issued by the University subject to completion of a programme within the maximum period prescribed.

ii. The Continuous Assessment marks originally secured by the learner, in the first appearance in the course(s) if any, will be carried forward.

10. Promotion Policy

Learners will be promoted from one year to another provided they have paid all the fee dues and do not have any discipline case pending against them. Appearing in the CA may be considered while deciding the promotion. They should however note that they have to pass in each course as per the passing requirements and earn minimum credit units required for the award of a degree/ qualification.

11. Award of Degree

The Learner will be awarded the Master of Computer Applications degree upon fulfillment of the following criteria:

i. Must have passed all the courses of the four semesters.

ii. Must have complied with all other assessment guidelines and criteria notified during the conduct of the Program.

iii. Must have submitted the UG Convocation degree certificate

The Degree Certificate will indicate the Elective opted by the learner.

12. Break and Maximum Period for Completion

With prior approval, a learner may be allowed to take a break (temporary withdrawal) from the Program for a Semester or more for valid reasons of health/career. The learner shall be allowed to continue the program after re-registration as per the university norms.

13. Project Work

Students undertake a project after the end of third semester. It provides an opportunity for the students to apply classroom learning and practice in a real-world and applied environment. The duration of the project is a minimum of 8 weeks. A learner can work with a company as an intern, undertake projects, perform activities identified by the company, and assist the organization by executing the project objectives; or alternatively can carry out an independent research based on contemporary problem areas. The project will help enhance the learner's knowledge and experience through practical implementation and development of skillsets in the processes of analyzing, assessing, designing and developing computing solutions using relevant software tools and platforms. The learner is required to submit a project report in the prescribed format. Learners are encouraged to convert their research into a paper/case and publish in association with a mentor.

The assessment will be according to pre-defined Rubrics based on performance Indicators like Similarity Check (Plagiarisms), Quality of References, Continuity of Work, Attainment of Learning Outcomes and Overall Quality in terms of potential of publishing/ patenting. While the students will give a certificate of it being his/ her original work, they will also give a No Objection Certificate of it being published or patented under the name of JAIN (Deemed-to-be University).

14. Academic Integrity and Ethics

i. A learner who has committed an act of academic dishonesty will be deemed to have failed to meet a basic requirement of satisfactory academic performance. Thus, academic dishonesty is not only a basis for disciplinary action but also is relevant to the evaluation of a student's level of performance and progress.

ii. Where there has been violation of the basic ethos and principles of academic integrity and ethics, the Dean/Board of Examiners/Course Coordinator may use their discretion during the Semester on the disciplinary action to be taken.

iii. Academic dishonesty includes, but is not necessarily limited, to the following:

a) Using more than one gadget/device during the conduct of the online examination;

b) Switching off the webcam during the conduct of the online examination;

c) Cheating or knowingly assisting another learner in committing an act of cheating;

d) Unauthorized possession of learning material, examination materials, destruction or hiding of relevant materials;

e) Act of plagiarism;

f) Unauthorized changing of marks or marking on examination records.

Master of Computer Applications

[Online Mode]

www.jainuniversity.ac.in

Corporate Office
91/2, Dr. A N Krishna Rao Road V V Puram,
Bengaluru - 560 004
P: +91 80 4343 1000

Centre for Distance & Online Education
319, 25th Main, 17th Cross,
J P Nagar 6th Phase
Bengaluru – 560 078
P: +91 80 4343 0100

Program Specific Outcome

By the end of the Online MCA Programme, students of all the electives will be able to:

Computer Science & IT:

PSO1: Design, implement, and manage scalable and securely engineered IT solutions catering to a variety of industry demands, leveraging the virtual environment.

PSO2: Proficiently assess and enhance the performance of software applications, databases, and network systems in online settings, ensuring

optimal digital functionality.

PSO3: Proficiently assess and enhancethe performance of software applications, databases, and network systems in online settings, ensuring optimal digital functionality.

Data Analytics

PSO1: Apply cutting-edge statistical and machine learning methods to analyze complex datasets in virtual environments, revealing valuable insights and patterns.

PSO2: Develop and execute data preprocessing, transformation, and visualization strategies tailored for online data analytics, ensuring accuracy and meaningful representation.

PSO3: Design and deploy predictive and prescriptive models suitable for virtualized data-driven decision- making, enhancing remote and distributed problem-solving.

Cyber Security

PSO1: Identify, assess, and proactively mitigate cybersecurity threats and vulnerabilities across diverse digital landscapes in virtual environments.

PSO2: Execute penetration testing and vulnerability assessments in virtual systems, fortifying the security posture of online platforms and networks.

PSO3: Formulate and execute strategies for digital forensics, incident response, and security risk management in online settings, ensuring rapid and effective virtual threat resolution.

Devops

PSO1: Apply DevOps methodologies—including Continuous Integration/Continuous Deployment (CI/CD), Infrastructure as Code (IaC), and container orchestration—to automate and optimize software development and operational workflows.

PSO2: Design and implement secure, scalable, and resilient cloud-native applications using microservices, DevSecOps practices, and Site Reliability Engineering (SRE) principles.

PSO3: Integrate AI and Big Data workloads into DevOps pipelines, demonstrating proficiency in data-driven automation, performance monitoring, and intelligent system adaptation for real-world business scenarios.

NLP & LLM Development

PSO1: Apply mathematical and algorithmic foundations to build intelligent systems using machine learning and statistical models.

PSO2: Design and deploy scalable AI solutions using modern tools for tasks like prediction, recommendation, and natural language understanding.

PSO3: Practice ethical AI development with strong communication and teamwork, addressing societal concerns through responsible innovation.

Artificial Intelligence

PSO1: Apply mathematical and algorithmic foundations to build intelligent systems using machine learning and statistical models.

PSO2: Design and deploy scalable AI solutions using modern tools for tasks like prediction, recommendation, and natural language understanding.

PSO3: Practice ethical AI development with strong communication and teamwork, addressing societal concerns through responsible innovation.

Full stack Development

PSO1: Full Stack Application Proficiency: Apply front-end and back-end development skills using HTML, CSS, JavaScript, React, Python, and Java to build responsive, scalable, and secure web applications.

PSO2: Database and DevOps Integration: Design and manage relational and advanced database systems, and implement DevOps practices for continuous integration, deployment, and monitoring of full stack solutions.

PSO3: Security, Testing & API Development: Demonstrate proficiency in developing secure web APIs, applying software testing methodologies, and implementing cryptographic techniques to ensure robust and reliable application performance

Data Science

PSO1: Apply mathematical, statistical, and algorithmic principles to design and implement data-driven solutions using tools such as Python, SQL, and Java across structured and unstructured data environments.

PSO2: Demonstrate proficiency in data visualization, predictive modeling, and machine learning techniques to extract insights and support decision-making in diverse domains including marketing, social media, and web analytics.

PSO3: Integrate advanced data science methodologies—including time series analysis, natural language processing, and GenAI—with ethical and entrepreneurial perspectives to develop scalable, socially responsible digital solutions.

Cloud Computing

PSO1: Develop expertise in designing, deploying, and managing scalable cloud-based solutions using contemporary tools and platforms such as AWS, Azure, and Google Cloud

PSO2: Apply advanced knowledge of cloud foundations, microservices, and security practices to build efficient, secure, and reliable cloud infrastructures.

PSO3: Demonstrate the ability to innovate and manage cloud solutions responsibly while addressing environmental, social, and governance (ESG) concerns and working effectively in diverse teams.

Course Matrix

ELECTIVE: Computer Science and IT

Corporate Office
91/2, Dr. A N Krishna Rao Road V V Puram,
Bengaluru - 560 004
P: +91 80 4343 1000

Centre for Distance & Online Education
319, 25th Main, 17th Cross,
J P Nagar 6th Phase
Bengaluru – 560 078
P: +91 80 4343 0100

ELECTIVE: Computer Science and IT

Semester I:

Course Type	Course Code	Course Title	Credits
CC	25VMT0C101	Mathematical Foundations for Computer Science	4
CC	25VMT0C102	Computing Concepts and Problem Solving using C	4
CC	25VMT0C103	Operating Systems	4
CC	25VMT0C104	Data Structures	4
SEC	25VMT0L101	Problem Solving using C Lab	2
SEC	25VMT0L102	Data Structures Lab	2
SEC	26VMX0S105	AI Skills for Future	2
Total Credits			22

Semester II:

Course Type	Course Code	Course Title	Credits
CC	25VMT0C201	Advanced Database Systems	4
CC	25VMT0C202	Java Programming	4
CC	25VMT0C203	Web Technologies	4
CC	25VMT0C204	Cloud Computing Foundations	4
SEC	25VMT0L201	Advanced Database Systems Lab	2
SEC	25VMT0L202	Java Programming Lab	2
AECC	25VMT0C205	Environmental, Social, and Governance (ESG) in IT	4
Total Credits			24

Semester III:

Course Type	Course Code	Course Title	Credits
CC	251VMT0C301	Artificial Intelligence and Machine Learning	4
CC	251VMT0C302	Python Programming	4
DSE	251VMT1S303	Software Testing	4
SEC	251VMT0L301	AI & Machine Learning Lab	2
SEC	251VMT0L302	Python Programming Lab	2
OE		Open Elective	4
Total Credits			20

Semester IV:

Course Type	Course Code	Course Title	Credits
DSE	251VMT1S401	IT Project Management	4
DSE	251VMT1S402	Big Data Analytics	4
DSE	251VMT1S403	Low-Code/No-Code Application Development	4
DSE	251VMT1S404	Data Visualization	4
SEC	251VMT0C405	Strategic Leadership in IT	2
Thesis	251VMT0P406	Project	6
Total Credits			24

Course Description:

This course builds a strong mathematical base essential for various domains in computer science such as algorithms, cryptography, artificial intelligence, and data science. It covers discrete structures, logic, combinatory, graph theory, and number theory. Emphasis is laid on both theoretical understanding and application to computational problems.

Course Outcomes (COs)

Upon successful completion of this course, learners will be equipped to:

CO	**Course Outcome**	**BTL**
CO1	Comprehend the fundamentals of logic, sets, and relations used in computer science.	L3
CO2	Apply principles of combinatory and counting for solving algorithmic and real-world problems.	L3
CO3	Analyze and construct graphs, trees, and networks relevant to data structures and networking.	L4
CO4	Solve recurrence relations and use mathematical induction in algorithm design.	L3
CO5	Evaluate number theoretic and algebraic methods in cryptographic and coding applications.	L4

Course Module

No. of Hours:24

Module 1: Logic, Sets, and Relations
Scope of Module: Provides foundational concepts in mathematical logic, set theory, and relations essential for modeling and designing algorithms.
Topics: Propositions and logical operators, Truth tables, Tautology and contradiction, Predicates and quantifiers, Rules of inference computation, Logical equivalence, Methods of proof, Set theory basics, Venn diagrams, Set operations, Cartesian product of sets, Types of relations, Properties of relations, Equivalence and partial order relations, Closure of relations.
Learning Outcome: *Students will be able to construct logical arguments and model relationships using sets and relations.*

No. of Hours:24

Module 2: Combinatorics and Counting
Scope of Module: Focuses on combinatorial problem-solving techniques relevant to algorithm complexity and probability theory.
Topics: Basic counting principles, Addition and multiplication rules, Permutations and combinations, Pigeonhole principle, Principle of inclusion and exclusion,

Binomial theorem, Pascal's triangle, Combinatorial identities, Multisets and combinations with repetition, Generating functions, Recurrence relations, Solving linear recurrence, Introduction to partitions, Derangements, Applications in probability.
Learning Outcome: *Students will be able to apply combinatorial reasoning to solve complex counting and probability problems.*

No. of Hours:24

Module 3: Graph Theory and Trees
Scope of Module: Introduces graph-based models to represent and solve problems in data structures, networks, and algorithms.
Topics: Basic concepts of graphs, Types of graphs, Graph representations, Degree of vertices, Paths and cycles, Connected components, Eulerian and Hamiltonian graphs, Planar graphs, Graph coloring, Matrix representation of graphs, Trees and their properties, Binary and n-ary trees, Tree traversal algorithms, Spanning trees, Minimum spanning tree algorithms (Kruskal, Prim)
Learning Outcome: *Students will be able to model and solve problems using graphs and trees in computer science.*

No. of Hours:24

Module 4: Recurrence Relations and Mathematical Induction
Scope of Module: Covers recurrence relations and inductive reasoning techniques used in algorithm analysis and design.
Topics: Sequences and recurrence relations, Solving homogeneous linear recurrence, Non-homogeneous recurrence, Generating functions for recurrence, Divide-and-conquer relations, Recursion trees and Master's theorem, Mathematical induction, Strong induction, Well-ordering principle, Applications in algorithm correctness, Tower of Hanoi, Binary search, Fibonacci sequence, Ackermann's function.
Learning Outcome: *Students will be able to formulate and solve recurrence relations and validate algorithms using induction.*

No. of Hours:24

Module 5: Number Theory and Algebraic Structures
Scope of Module: Explores concepts in number theory and abstract algebra crucial for cryptography, hashing, and finite state systems.
Topics: Divisibility and Euclidean algorithm, Prime numbers and factorization, Congruence relations, Modular arithmetic, Fermat's little theorem, Euler's theorem, Chinese Remainder Theorem, Greatest common divisor (GCD), Least

common multiple (LCM), Rings and fields basics, Group theory introduction, Subgroups and cyclic groups, Applications in cryptography, Hashing functions, Finite automata fundamentals
Learning Outcome: *Students will be able to apply number theoretic and algebraic techniques in areas like security and finite state systems.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites (If Any):

- Basic mathematics at undergraduate level, understanding of algebra and logic.

Pedagogy / Teaching Methodology:

- ✓ Conceptual and problem-based learning
- ✓ Interactive class exercises and proofs
- ✓ Real-world applications and modeling
- ✓ Assignments and quizzes using math software.

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. Discrete Mathematics by NPTEL: https://nptel.ac.in/courses/111106086

2. Mathematics for Computer Science by MIT Open Courseware: https://ocw.mit.edu

Suggested Readings:

- Kenneth H. Rosen, Discrete Mathematics and Its Applications, McGraw-Hill, 2017
- Tremblay and Manohar, Discrete Mathematical Structures, Tata McGraw-Hill, 2001
- C.L. Liu and D.P. Mohapatra, Elements of Discrete Mathematics, McGraw-Hill, 2010
- Online resources from IEEE Xplore, Springer Link for mathematical modeling and computation

Recommended Software/Tools (If applicable):

1. MATLAB/Octave
2. SageMath
3. Python (SymPy, NetworkX)
4. GeoGebra for visualization

Curriculum Development:

Sl. No	**Identify and highlight in the module**
1	Local, Regional, National and Global
2	Employability, Entrepreneurship and Skill Development
3	Value Added Courses (Module)
4	Integration of Digital Learning Tools
5	Crossing Cutting Edges (Ethics, Gender Equality, Sustainability and Environmental concerns)
6	Indian Knowledge System

CO/PO Mapping Table

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	-	2	1	-	1	1
CO2	2	3	2	1	1	-	-	2	1
CO3	3	2	2	1	1	1	1	1	2
CO4	2	3	3	1	2	2	-	1	2
CO5	3	2	1	2	2	1	1	2	2
Articulation	2.6	2.4	1.8	0.8	1.6	1.25	1	1.4	1.6

Note: The Articulation is indicative of how a particular course maps to a PO.

Articulation is calculated as:

$$= \frac{Sum\ of\ CO\ Mappiung\ Levels\ for\ a\ perticular\ PO}{No.of\ COs\ that\ mapped\ to\ that\ PO}$$

Example:1

$$Articulation\ of\ PO1 = \frac{(3+2+3+2+3)}{5} = 2.6$$

Example: 2

$$Articulation\ of\ PO7 = \frac{(1+1)}{2} = 1.0$$

Course Title:	**Computing concepts and problem solving in C**
Course Code:	**25VMT0C102**
Semester:	**I**
Credits:	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course covers Advanced concepts of computing concepts and problem solving in C and their applications in computer science. Students will develop problem-solving abilities and practical programming skills essential for core areas such as algorithms, systems programming, software development, and emerging technologies.

Course Outcomes (COs)

CO	Course Outcome	BTL
CO1	Comprehend the advanced Programming constructs	L3
CO2	Apply File Operations and handle errors effortlessly	L3
CO3	Integrate Linux Programming with C Scripting	L4
CO4	Demonstrate C System Programming	L3
CO5	Evaluate real-world scenarios and Case Studies	L5

Upon successful completion of this course, learners will be equipped to:

Course Modules

No. of Hours:24*

Module 1: Fundamentals of Computing and C Programming
Scope of Module: Introduces basic computing concepts, algorithm design, and foundational elements of the C programming language
Topics: Basics of Computers: Architecture, memory, input/output units; Introduction to Algorithms and Flowcharts; C Programming Basics: Structure, syntax, variables, constants, data types; Operators: Arithmetic, relational, logical, bitwise, assignment, conditional; Compilation and Execution process.
Learning Outcome:*Students will be able to describe computing concepts, write simple C programs, and understand compilation processes.*

No. of Hours:24

Module 2: Control Structures, Functions, Arrays, Strings, and Modular Programming
Scope of Module: Explores problem-solving strategies using control flow, modular programming, and essential data structures.
Topics: Control Structures: if, if-else, nested if, switch-case; Loops: for, while, do-while, nested loops; Functions: Declaration, definition, call, recursion; Scope, Storage Classes (auto, static, extern, register); Modular Programming: Header files,

preprocessor directives (#define, #include); Arrays: 1D, 2D arrays, array manipulation; Strings: String functions (strlen, strcpy, strcat, strcmp), string manipulation.
Learning Outcome:*Students will be able to apply control structures, modularize code, and manipulate arrays and strings effectively.*

No. of Hours:24

Module 3: Pointers, Structures, Dynamic Memory Allocation, and File Handling
Scope of Module: Provides in-depth understanding of memory management, structured data, and file operations in C.
Topics: Introduction to Pointers: Pointer arithmetic, pointers and arrays; Pointers to Functions and Pointers to Structures; Structures and Unions: Declaration, definition, accessing members; Dynamic Memory Allocation: malloc, calloc, realloc, free; File Handling: File operations (read, write, append), file pointers; Case Study: Creating a file-based student management system.
Learning Outcome:*Students will be able to manipulate memory using pointers, manage structured data, and perform file operations.*

No. of Hours:24

Module 4: ***System-Level Programming and Real-Time Applications in C***
Scope of Module: *Focuses on industry-relevant applications, system programming concepts, and real-time project development.*
Topics: *Command-line Arguments and Environment Variables; Error Handling (perror, strerror) and Debugging Techniques; Mini Projects: Text-based Games (Tic-Tac-Toe), Simple Shell Implementation, Real-Time Data Logger for IoT (file-based); Case Study: Simulating a Banking System.*
Learning Outcome: *Students will be able to design and implement system-level programs and real-time applications using C.*

No. of Hours:24

Module 5: Advanced Industry Applications and Case Studies
Scope of Module: Encourages application of C programming skills in solving industry-specific problems and case studies.
Topics: Writing basic command-line tools; Simple network programs (client-server); Debugging and optimizing small C programs; Working with file compression and encryption tools; Implementing simple database file storage; Case Study:

Building a basic calculator or file management tool; Mini Project: Simple automation program, data logger, or system monitor.
Learning Outcome: *Students will be able to apply C programming skills to solve advanced, industry-relevant problems through case studies and projects.*

Module	No. of Pre-Recordings
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10.

Prerequisites (If Any):

- Basic knowledge of computer fundamentals.

Pedagogy / Teaching Methodology:

- ✓ Lectures, Hands-on Lab Sessions, Problem-solving Tasks, Case-based Assignments, Group Projects

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. NPTEL, Coursera, edX, MIT OCW — based on relevance to topics.

Suggested Readings:

- Standard textbooks, research articles, online tutorials and eBooks related to the subject.

Recommended Software/Tools (If applicable):

1. GCC Compiler, Visual Studio Code, Linux Terminal, C IDEs

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	3	2	2	2	1	-	1	2
CO2	3	2	2	3	2	1	-	1	1
CO3	2	3	2	2	-	1	1	1	1
CO4	3	2	3	2	2	1	-	1	2
CO5	2	2	2	3	2	2	-	2	2
Articulation	2.6	2.4	2.1	2.4	2	1.1	1	1.1	1.6

Course Title:	**Operating Systems**
Course Code:	**25VMT0C103**
Semester:	**I**
Credits:	**4**
Course Type	Core Course
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides an in-depth exploration of Advanced operating system concepts, focusing on the design, implementation, and management of modern operating systems. Topics include process synchronization, memory management, file systems, distributed systems, and security mechanisms. Practical assignments and case studies will reinforce theoretical knowledge, preparing students for real-world system-level programming and research.

Course Outcomes(COs)

CO	Course Outcome	BTL
CO1	Apply process management and synchronization techniques.	L3
CO2	Implement memory Management Techniques	L3
CO3	Implement File Management Techniques	L3
CO4	Design solutions using modern OS tools and case studies in distributed systems.	L3
CO5	Evaluate modern tools and techniques in real-world OS environments.	L5

Upon successful completion of this course, learners will be equipped to:

Course Modules:

No. of Hours:24

Module 1: Introduction to Operating Systems and Process Management
Scope of Module: Introduces core OS concepts, structures, and the fundamentals of process management.
Topics: Overview of Operating Systems: Definitions, functions, types (batch, time-sharing, distributed, real-time); OS structure and components; System calls and APIs; Microkernel vs Monolithic Kernel architectures; System Boot Process and Initialization (GRUB, systemd); Process concepts, process states, process control block (PCB); Process scheduling algorithms (FCFS, SJF, Round Robin, Priority); Context switching; Process scheduling in multi-core and NUMA systems; Case studies on modern OS architectures (Linux, Windows).
Learning Outcome: *Students will be able to describe OS structures, functions, and manage basic process scheduling scenarios.*

No. of Hours:24

Module 2: Process Synchronization, Deadlock, and CPU Scheduling
Scope of Module: Covers critical section problems, synchronization mechanisms, deadlock handling, and CPU scheduling strategies.
Topics:

Critical section problem, Peterson's solution, semaphores, monitors; Spinlocks, Mutexes, and Condition Variables; Hardware support for synchronization (Test-and-Set, Compare-and-Swap); Classic synchronization problems (producer-consumer, readers-writers, dining philosophers); Deadlock characterization, prevention, avoidance (Banker's algorithm), detection, recovery, and resource preemption; CPU scheduling strategies (multilevel queue, multilevel feedback queue); Real-world scheduling policies (CFS in Linux, Windows scheduler comparison); Case studies on synchronization in Linux kernel.
Learning Outcome: *Students will be able to apply synchronization techniques and deadlock handling strategies in process management.*

No. of Hours:24

Module 3: Memory Management, Virtual Memory, and Storage Systems
Scope of Module: Focuses on memory organization, virtual memory, and file systems management.
Topics: Memory management strategies: contiguous allocation, paging, segmentation; Virtual memory: demand paging, page replacement algorithms (LRU, FIFO, Optimal); Thrashing and performance tuning; Memory-mapped I/O and Direct Memory Access (DMA); Swap space management and tuning; File system interface and implementation, allocation methods (contiguous, linked, indexed); Directory structures, file system mounting, journaling file systems; Advanced file system topics: RAID levels, ZFS/Btrfs file systems; Security implications in storage systems (encryption at rest); Case study: EXT4 file system analysis.

***Learning Outcome:**Students will be able to analyze memory and storage management strategies in OS environment*

No. of Hours:24

Module 4: Virtualization, Containerization, and Real-Time Systems
Scope of Module: Explores virtualization technologies, container-based systems, and introduces real-time OS concepts.
Topics: Virtualization concepts, hypervisors (Type 1 and Type 2), virtual machine monitors (VMWare, KVM); Nested Virtualization (running VMs inside VMs); Containerization with Docker, Kubernetes orchestration basics; Container Security (namespaces, cgroups, seccomp); Comparison of VMs vs Containers; Performance tuning of VMs and Containers; Introduction to real-time operating systems (RTOS), real-time scheduling algorithms; RTOS performance metrics and real-world RTOS examples (FreeRTOS, VxWorks); Case study: Docker-based microservices in cloud environments.
***Learning Outcome:**Students will be able to evaluate virtualization and containerization technologies and apply real-time scheduling concepts.*

No. of Hours:24

Module 5: Security, Protection Mechanisms, and Advanced OS Case Studies
Scope of Module:

Focuses on OS security, protection models, and advanced industrial applications of operating systems.
Topics: OS security principles, access control models (DAC, MAC, RBAC); Kernel hardening techniques (grsecurity, seccomp); Network security in OS: Firewall configurations (iptables, nftables); Secure Boot and Trusted Platform Module (TPM); Security vulnerabilities and mitigation strategies; Encryption, authentication, and auditing in OS; Introduction to SELinux, AppArmor; Forensics in OS environments (log analysis, intrusion detection); Case studies on OS security breaches and their prevention; Mini Project: Design a simple OS-level security module, process monitoring utility, or Intrusion Detection System (IDS) script.
Learning Outcome:*Students will be able to assess and implement security models and protection mechanisms in modern OS environments.*

Module	No. of Pre-Recordings
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10.

Prerequisites (If Any):

- Basic knowledge of computer fundamentals.

Pedagogy / Teaching Methodology:

✓ Lectures, Hands-on Lab Sessions, Problem-solving Tasks, Case-based Assignments, Group Projects

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. NPTEL, Coursera, edX, MIT OCW — based on relevance to topics.

Suggested Readings:

- Standard textbooks, research articles, online tutorials and eBooks related to the subject.

Recommended Software/Tools (If applicable):

1. GCC Compiler, Visual Studio Code, Linux Terminal, C IDEs

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	2	2	2	1	-	1	2
CO2	2	3	3	2	2	1	1	2	1
CO3	3	2	2	2	2	1	1	2	2
CO4	2	3	3	2	2	2	1	1	2
CO5	3	2	2	3	3	2	1	2	2
Articulation	2.6	2.4	2.4	2.2	2.2	1.4	1	1.4	1.6

Course Title:	**Data Structures**
Course Code:	**25VMT0C104**
Semester:	**I**
Credits:	**4**
Course Type	Core Course
Learning Hours	120
Live Sessions	**12 hours**

Course Description

This course introduces Advanced Concepts of data structures, algorithms, and their practical use in computing applications. Students will learn how to implement, analyze, and apply various linear and non-linear data structures to build efficient and scalable software systems

Course Outcomes(Cos)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Understand the fundamentals of advanced linear and data structures.	L1
CO2	Apply appropriate data structures for solving complex computational problems.	L3
CO3	Apply tree graph techniques in software solutions.	L3
CO4	Implement hashing techniques in software solutions.	L3
CO5	Evaluate efficient programs using modern tools and real-world case studies.	L5

Note: Implementation can be done in C/C++

Course Modules

No. of Hours:24

Module 1: : Linear Data Structures
Scope of Module Covers implementation and usage of linear storage structures.
Topics: Arrays, Stacks (array/linked), Queues (normal, circular), Dequeues, Linked list operations (SLL, DLL, CLL), Stack/Queue with linked lists, Applications in expression evaluation, Bracket matching, Recursion stack, Queue in scheduling, List-based simulation, Sparse matrices, Time/space complexity basics, Array vs list.
Learning Outcome*: Apply linear structures in solving algorithmic problems.*

No. of Hours:24

Module 2: Trees and Binary Trees
Scope of Module Focuses on hierarchical data structures with operations and applications of binary trees and advanced trees.
Topics: Tree definitions and terminologies, Binary tree representation, Binary tree traversals (inorder, preorder, postorder), Binary search tree (BST), Insertion in BST, Deletion in BST, Balanced trees (AVL trees), Height-balanced trees, Expression trees, Threaded binary trees, Heap trees, Min-heap and max-heap, Priority queue using heap, B-Trees, Applications of trees.

***Learning Outcome**: Students will be able to implement tree structures and use them for hierarchical data processing.*

No. of Hours:24

Module 3: Graphs and Applications
Scope of Module: Introduction to graphs and path-finding algorithms.
Topics: Graph representation (adjacency list/matrix), BFS, DFS, Topological sorting, Dijkstra's algorithm, Bellman-Ford, Floyd-Warshall, Kruskal's algorithm, Prim's algorithm, Network flow, Graph coloring, Euler and Hamiltonian paths, Graph traversals for cycles, Applications in networks, Graph problem modeling.
Learning Outcome: *Analyze and solve graph-based problems using suitable algorithms*

No. of Hours:24

Module 4: Hashing and Searching Structures
Scope of Module: Efficient search, hash, and map-based data access.
Topics: Hash tables, Hash functions, Collision resolution (chaining, open addressing), Load factor, Linear probing, Double hashing, Bloom filters, Dictionaries, Sets,

Unordered maps, Binary search trees vs hashing, Search time analysis, Skip lists, LRU Cache design, Memory-optimized hashing, Applications in OS/DB.
Learning Outcome: *Design and implement efficient search and mapping structures.*

No. of Hours:24

Module 5: Case Studies and Tools
Scope of Module: Tools and case implementations using advanced DS.
Topics: STL containers in C++ (vector, list, map, set), Data structure libraries in C++, Visualization tools (VisuAlgo, Graphviz), IDEs (CLion, Eclipse), Real-world DS projects, Competitive programming platforms, Time-space profiler tools, LeetCode/Codeforces case studies, DS in AI/ML preprocessing, Game development DS, Cache designs, Real DBMS indexing, Distributed DS structures.
Learning Outcome: *Evaluate and apply real-world DS solutions using modern tools and case studies.*

Module	**No. of Pre-Recordings**
1	**10**
2	**12**

3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10.

Prerequisites (If Any):

- Basic knowledge of programming and algorithms.

Pedagogy / Teaching Methodology:

- ✓ Interactive Lectures
- ✓ Hands-on Programming Labs
- ✓ Assignments and Case Studies
- ✓ Mini Projects

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. NPTEL: Data Structures and Algorithms
2. Coursera: Data Structures by UC San Diego

Suggested Readings:

- Seymour Lipschutz, *Data Structures*, McGraw-Hill
- Ellis Horowitz, Sartaj Sahni, *Fundamentals of Data Structures*, Universities Press
- Mark Allen Weiss, *Data Structures and Algorithm Analysis in C*, Pearson Education
- Online resources and research papers related to data structures

Recommended Software/Tools (If applicable):

1. C/C++ Compiler
2. Visual Studio Code / Code::Blocks

3. Online IDEs (Repl.it, GeeksforGeeks IDE)
4. Visualgo.net for algorithm visualization

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	3	2	2	3	-	2	1
CO2	3	3	3	2	2	2	1	2	2
CO3	2	2	3	3	2	2	3	1	2
CO4	3	2	3	2	2	-	2	1	2
CO5	2	3	2	3	2	2	-	2	1
Articulation	2.6	2.4	2.8	2.4	2	2.2	2	1.6	1.6

Course Title:	**AI Skills for Future**
Course Code:	**26VMX0S105**
Semester:	**I**
Credits:	**2**
Course Type	**Skill Enhancement Course**
Learning Hours	60
Live Sessions	**12 hours**

Course Outcomes:

CO1: Understand Generative AI and LLMs in depth
CO2: Create text, image, and voice content using AI
CO3: Automate daily tasks and business workflows
CO4: Build AI agents that act independently
CO5: Design and deploy simple AI apps without coding

Course Modules

Module 1: Generative AI Prompt Engineering for Real Workflows

What is Generative AI and how it's transforming industries, Structure of an effective prompt: *Role → Task → Context → Output,* Common prompting mistakes and how to fix them, How large language models (LLMs) like ChatGPT, Claude, and Google AI Studio work, advanced prompting (few-shot, iterative, style-based), Using your own files in ChatGPT and Claude (RAG basics), Research and summarization using AI

Learning Outcome: Students will be able to understand how Generative AI works and master the art of prompting.

Module 2: Creative AI and Voice & Audio Creation

Understanding AI image generation, creating posters, campaign assets, and product visuals, building brand identity using Krea.ai, Generating realistic voiceovers using ElevenLabs, Script writing and tone control with ChatGPT & Claude, Integrating voice into short videos or podcasts ***Learning Outcome:*** Students will be able to use AI to generate visuals, voiceovers, and creative storytelling content.

Module 3: AI Automation & Productivity
Logic of triggers, actions, and workflows, setting up automation between apps (Gmail, Sheets, Drive), Connecting OpenAI to Make.com for intelligent automation, creating social media Management Workflow and research automation flows, automating daily reports and reminders, building personal productivity dashboards, Integrating Claude and ChatGPT into existing workflows
Learning Outcome: Automate daily workflows and tasks using AI and no-code tools.

Module 4: Advanced AI Agents & Decision Flows
Introduction to n8n as a visual AI workflow builder, understanding nodes, triggers, and data passing, Using OpenAI node for smart responses and actions, designing multi-step reasoning workflows, Connecting Google Sheets, Slack, and other APIs, building autonomous agents that perform specific tasks
Learning Outcome: Students will be able to build AI agents that can think, decide, and act automatically.

Module 5: AI App Building & Deployment
Fundamentals of UI/UX for AI apps, structuring front-end and AI back-end flows, designing app logic using n8n and Make.com, integrating OpenAI and Claude APIs to your apps, handling user inputs and dynamic responses, using Google AI Studio to experiment with models, hosting and testing your AI app online, adding analytics and monitoring usage, final demo preparation and peer feedback
Learning Outcome: Students will be able to learn to design, connect, and deploy simple AI apps using no-code tools

Semester-II

Course Title:	**Advanced Database Systems**
Course Code:	**25VMT0C201**
Semester:	**II**
Credits:	**4**
Course Type	Core Course
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course delves into advanced concepts and techniques in database systems, focusing on design, implementation, optimization, and emerging trends. Topics include distributed databases, NoSQL systems, data warehousing, transaction management, and big data technologies. The course emphasizes practical applications through case studies and hands-on projects.

Course Outcomes (COs)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Understand advanced database models and architectures.	L1
CO2	Design and optimize relational and NoSQL databases.	L3
CO3	Implement transaction management and concurrency control.	L4
CO4	Evaluate distributed database systems and their challenges.	L5
CO5	Apply data warehousing and big data technologies.	L3

Course Modules

No. of Hours:24

Module 1: Advanced Database Models and Architectures
Scope of Module: Covers advanced database models, including object-oriented, object-relational, and NoSQL databases, and their architectures.
Topics: Review of relational database systems; Object-oriented databases; Object-relational databases; NoSQL database overview; Document databases (MongoDB); Key-value databases (Redis); Column-family databases (Cassandra); Graph databases (Neo4j); NewSQL systems; Polyglot persistence; CAP theorem; BASE properties; Cloud-native database systems; Multi-model databases; Case study: MongoDB vs. PostgreSQL.
Learning Outcome: *Students will be able to compare and contrast different database models and select appropriate architectures for specific applications*

No. of Hours:24

Module 2: Database Design and Query Optimization
Scope of Module: Focuses on advanced techniques for database design, indexing, query optimization, and performance tuning
Topics: Advanced normalization (1NF–5NF); Denormalization strategies; Physical vs. logical design Partitioning strategies; Indexing: B-tree, hash, bitmap; Covering

indexes and index-only scans Query execution plans; Cost-based query optimization; Heuristic-based optimization; Materialized views; Query rewriting techniques; Adaptive query optimization; Statistics collection and use; SQL performance tuning tools; Hands-on: SQL tuning with EXPLAIN and ANALYZE.
Learning Outcome: *Students will be able to design efficient database schemas and optimize query performance.*

No. of Hours:24

Module 3: Transaction Management and Concurrency Control
Scope of Module: Explores transaction properties, concurrency control mechanisms, and recovery techniques in database systems
Topics: ACID properties; Transaction lifecycle; Serializability and schedules; Lock-based concurrency control; Deadlock detection and prevention; Timestamp ordering; Optimistic concurrency control; Two-phase locking; Multiversion Concurrency Control (MVCC); Isolation levels and anomalies; Serializable snapshot isolation; Write-ahead logging; Checkpointing; Recovery methods; Case study: PostgreSQL concurrency and recovery.
Learning Outcome: *Students will be able to implement and analyze transaction management and concurrency control mechanisms.*

No. of Hours:24

Module 4: Distributed Database Systems
Scope of Module: Examines distributed database architectures, data fragmentation, replication, and consistency models.
Topics: Distributed database architecture; Client-server vs. peer-to-peer; Horizontal and vertical fragmentation; Data allocation and replication; Distributed concurrency control; Distributed transactions; Two-phase commit protocol; Distributed query processing. Query decomposition and localization; CAP theorem in distributed databases; Eventual consistency models; Conflict resolution strategies; Fault tolerance and availability; Case study: Google Spanner; Hands-on: Setting up a distributed Cassandra cluster.
Learning Outcome: *Students will be able to evaluate distributed database systems and address their challenges.*

No. of Hours:24

Module 5: Data Warehousing and Big Data Technologies
Scope of Module: Introduces data warehousing concepts, ETL processes, OLAP, and big data technologies
Topics: Data warehousing concepts; Star schema vs. snowflake schema; Fact and dimension tables ETL process design; Data cleaning and transformation; Data staging and loading; OLAP operations and cubes; Hadoop ecosystem (HDFS, MapReduce); Apache Hive and Pig Apache Spark fundamentals; NoSQL for Big

Data: HBase and Cassandra; Data lakes and lakehouse architecture; Real-time data pipelines (Kafka + Spark Streaming); Columnar storage formats (Parquet, ORC); Mini-project: Building a Spark-based data pipeline
Learning Outcome: *Students will be able to apply data warehousing and big data technologies for analytical processing*

Module	No. of Pre-Recordings
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10.

Prerequisites (If Any):

- Basic knowledge of database management systems (DBMS) and SQL.

Pedagogy / Teaching Methodology:

- ✓ Conceptual and problem-based learning
- ✓ Hands-on labs and case studies
- ✓ Group projects and presentations
- ✓ Quizzes and assignments using database tools

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. Advanced Database Systems by NPTEL: https://nptel.ac.in/courses/106105171
2. Database Systems by Stanford Online: https://online.stanford.edu/courses/soe-ydatabases-database-systems

Suggested Readings:

- Raghu Ramakrishnan and Johannes Gehrke, *Database Management Systems*, McGraw-Hill, 2003.
- Hector Garcia-Molina, Jeffrey D. Ullman, and Jennifer Widom, *Database Systems: The Complete Book*, Pearson, 2008.
- Martin Kleppmann, *Designing Data-Intensive Applications*, O'Reilly, 2017.

Recommended Software/Tools:

1. PostgreSQL, MongoDB, Cassandra
2. Hadoop, Spark
3. SQL Developer, Tableau

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools

5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO/PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	-	2	1	-	1	-
CO2	2	3	2	1	1	-	-	2	-
CO3	3	2	2	1	1	1	1	1	2
CO4	2	3	3	1	2	2	-	2	2
CO5	3	2	1	2	2	1	1	2	2
Articulation	2.6	2.6	1.8	1.25	1.6	1.25	1	1.4	2

Course Title:	**Java Programming**
Course Code:	**25VMT0C202**
Semester:	**II**
Credits:	**4**
Course Type	Core Course
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides an in-depth understanding of Java programming, covering object-oriented principles, core APIs, multithreading, networking, and modern frameworks. Emphasis is placed on practical implementation through projects, case studies, and hands-on labs to build robust applications.

Course Outcomes (Cos)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Understand object-oriented principles in Java.	L1
CO2	Develop applications using Java core APIs and collections.	L3
CO3	Implement multithreading and exception handling.	L4
CO4	Design GUI applications and network-based solutions.	L5
CO5	Apply Java frameworks (Spring Boot) for enterprise applications.	L3

Course Modules

No. of Hours:24

Module 1: Object-Oriented Programming in Java
Scope of Module: Covers fundamental OOP concepts, Java syntax, and class design
Topics: Introduction to Java and JVM; Data types, operators, and control structures; Class structure and object creation; Encapsulation and constructors; Inheritance and method overriding; Polymorphism and dynamic binding; Interfaces and abstract classes; Package creation and import; Access modifiers and scope; Static vs. instance members; Exception handling: try-catch-finally; Custom exceptions; Assertions and error handling; Object class and method overriding; Case study: Modular banking system design.
Learning Outcome: *Students will be able to apply OOP principles such as encapsulation, inheritance, and polymorphism to design structured and reusable Java applications.*

No. of Hours:24

Module 2: Java Core APIs and Collections
Scope of Module: Focuses on Java standard libraries and data structures.
Topics: String manipulation and immutability; Regular expressions; Wrapper classes and autoboxing Generics in Java; ArrayList, LinkedList implementations; HashSet, TreeSet, LinkedHashSet usage; HashMap, TreeMap, LinkedHashMap differences; Iterators, ListIterators, and enhanced for-loop; Comparable vs.

Comparator; Collections utility class; File handling with java.io and java.nio; Object serialization and deserialization; Properties and configuration files; Practical: Inventory management system; Debugging collections and memory leaks.
Learning Outcome: *Students will be able to leverage core Java libraries and the Collections Framework to build efficient, maintainable applications with complex data structures.*

No. of Hours:24

Module 3: Multithreading and Concurrency
Scope of Module: Explores concurrent programming in Java.
Topics: Thread lifecycle and states; Thread creation via Thread class; Runnable interface and Lambda expressions; Thread prioritization and yielding; Synchronized methods and blocks; Intrinsic locks and synchronized collections; Volatile keyword and atomic variables; Deadlock scenarios and prevention strategies; wait(), notify(), and notifyAll(); Thread pooling and ExecutorService; Callable and Future interfaces; ConcurrentHashMap and thread-safe queues; ForkJoinPool basics; Case study: Multithreaded chat application; Concurrency debugging tools and techniques.
Learning Outcome: *Students will be able to implement multithreaded programs and apply Java concurrency tools to ensure safe, scalable execution of parallel tasks.*

No. of Hours:24

Module 4: GUI and Networking
Scope of Module: Introduces Java FX and network programming.
Topics: Introduction to JavaFX and GUI architecture; SceneBuilder and JavaFX controls; Layout managers: BorderPane, VBox, HBox; Event-driven programming in Java; Creating forms and validation; Swing basics and component lifecycle; Action listeners and adapters; Socket programming: TCP; Datagram programming: UDP; HTTP communication using HttpClient Using REST APIs in Java; JDBC for database connectivity; CRUD operations with JDBC Mini-project: GUI Weather App with REST API; Threading in GUI and networking applications
Learning Outcome: *Students will be able to design and build user-friendly GUI applications integrated with network communication and backend services using Java.*

No. of Hours:24

Module 5: Java Frameworks and Modern Tools
Scope of Module: Covers Spring Boot and testing frameworks.
Topics: Introduction to Spring Boot and its architecture; Dependency Injection (DI) in

Spring; Creating RESTful APIs with Spring Boot; Spring annotations and application.properties; CRUD operations with Spring Data JPA; Connecting to MySQL/PostgreSQL; Validation with Spring MVC; Unit testing with JUnit 5; Mocking with Mockito; Postman for API testing; Building with Maven and Gradle; Logging with Logback and SLF4J; Creating JAR/WAR packages; Dockerizing a Spring Boot app; Capstone project: E-commerce backend.
Learning Outcome: *Students will be able to develop full-stack Java applications using Spring Boot and related modern tools, and deploy them in enterprise-ready environments.*

Module	**No. of Pre-Recordings**
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10

Prerequisites (If Any):

- Basic programming knowledge (C/C++/Python).

Pedagogy / Teaching Methodology:

- ✓ Interactive lectures with live coding
- ✓ Hands-on labs and debugging sessions
- ✓ Agile-style group projects
- ✓ Code reviews and peer assessments

Certificate / Value Added Courses Recommended:

- Java Programming by NPTEL: https://nptel.ac.in/courses/106105168
- Java Fundamentals by Oracle Academy: https://academy.oracle.com

Suggested Readings:

- Herbert Schildt, *Java: The Complete Reference*, McGraw-Hill, 2021.
- Joshua Bloch, *Effective Java*, Addison-Wesley, 2017.
- Craig Walls, *Spring Boot in Action*, Manning, 2020.

Recommended Software/Tools:

1. JDK 17+, IntelliJ IDEA/Eclipse
2. Spring Tool Suite, Postman
3. Git, Maven

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)

6.	Indian Knowledge System

CO PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	–	2	1	–	1	2
CO2	2	3	2	1	-	–	–	2	-
CO3	3	2	2	-	-	1	2	-	2
CO4	2	3	3	1	2	2	–	1	2
CO5	3	2	-	2	2	1	1	2	2
Articulation	2.6	2.4	2	1.33	2	1.25	1.5	1.5	2

Course Title:	**Web Technologies**
Course Code:	**25VMT0C203**
Semester:	**II**
Credits:	**4**
Course Type	Core Course
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides comprehensive knowledge of modern web development technologies, covering front-end design, back-end programming, databases, and web security. Students will learn to build responsive, interactive web applications using HTML5, CSS3, JavaScript, PHP/Node.js, and frameworks like React and Express. Practical implementation through projects and case studies is emphasized.

Course Outcomes (COs)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Understand client-side web technologies (HTML5, CSS3, JavaScript).	L1
CO2	Develop interactive front-end applications using frameworks.	L3
CO3	Implement server-side programming and database integration.	L4
CO4	Evaluate web security threats and countermeasures.	L5
CO5	Build full-stack web applications using modern tools.	L3

Course Modules

No. of Hours:24

Module 1: Client-Side Web Technologies
Scope of Module: Covers fundamental technologies for building web interfaces.
Topics: HTML5 document structure and semantic tags; Forms and input validation; CSS3 selectors and specificity; Box model, flexbox, and grid layouts; Responsive design principles (media queries); Typography and UI consistency; JavaScript syntax and data types; Control structures and functions; ES6+ features: let/const, arrow functions, template literals; DOM tree and traversal; Event handling and delegation; Form validation using JavaScript; Browser console and developer tools; Animation with CSS and JavaScript; Project: Portfolio website with interactivity.
Learning Outcome: *Students will be able to create accessible, responsive, and well-structured web pages using HTML5, CSS3, and modern JavaScript.*

No. of Hours:24

Module 2: Front-End Frameworks
Scope of Module: Focuses on modern JavaScript frameworks for UI development
Topics: Introduction to React.js and component lifecycle; JSX syntax and virtual DOM; Functional components and props; State and event management; Conditional rendering and lists; Forms and controlled components; React Hooks (useState, useEffect); Component communication; React Router for navigation; Context API for global state; AJAX requests with Axios and Fetch API; Using UI libraries (Bootstrap, Material UI); Creating reusable components; Code splitting and lazy loading; Project: Product listing page for e-commerce.
Learning Outcome: *Students will be able to build scalable and maintainable front-end applications using modern JavaScript frameworks like React.*

No. of Hours:24

Module 3: Server-Side Programming
Scope of Module: Explores back-end development and database integration.
Topics: Node.js basics and NPM; Creating a local web server; Express.js routing and middleware; Handling POST and GET requests; Template engines (EJS/Handlebars); Form data parsing and validation; MySQL and MongoDB CRUD operations; Connection pooling and error handling; RESTful API design principles; JWT and session-based authentication; Role-based access control;

File uploads and file system module; Environment variables and configuration; Project: Blog system with authentication and database; Debugging Node.js applications.
Learning Outcome: *Students will be able to build robust server-side applications and RESTful APIs integrated with databases using Node.js and Express.*

No. of Hours:24

Module 4: Web Security
Scope of Module: Addresses security vulnerabilities and protection mechanisms.
Topics: Understanding HTTP protocol and HTTPS; SSL certificates and secure headers; OWASP Top 10 vulnerabilities; SQL injection and input sanitization; Cross-Site Scripting (XSS) and mitigation; Cross-Site Request Forgery (CSRF); Password hashing with bcrypt; Authentication vs. Authorization; Content Security Policy (CSP); Secure cookie practices; Same-Origin Policy and CORS; Rate limiting and API throttling; Penetration testing tools (Burp Suite, OWASP ZAP); Security audit and logging; Project: Secure login and audit trail system
Learning Outcome: *Students will be able to identify, prevent, and mitigate common web application security threats using industry best practices.*

No. of Hours:24

Module 5: Full-Stack Development
Scope of Module: Integrates all layers of web application development.
Topics: Overview of MERN/MEAN stacks; Connecting frontend with backend via APIs; Redux basics and state management; User authentication and protected routes; Role-based content rendering; API testing with Postman; Code versioning with Git and GitHub; Continuous deployment workflows; Hosting frontend (Netlify, Vercel); Hosting backend (Render, Heroku) Performance optimization tips; Unit testing with Jest and Mocha; Linting and code formatting (Prettier, ESLint); DevOps basics for deployment; Capstone Project: Social media dashboard with analytics.
Learning Outcome: *Students will be able to develop, test, and deploy full-stack web applications using modern development tools and deployment platforms.*

Module	**No. of Pre-Recordings**
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10

Prerequisites (If Any):

- Basic programming knowledge (HTML/CSS recommended).

Pedagogy / Teaching Methodology:

- Interactive coding demonstrations
- Pair programming exercises
- Project-based learning
- Security hackathons

Certificate / Value Added Courses Recommended:

1. Web Development by MDN: https://developer.mozilla.org
2. Full-Stack Open: https://fullstackopen.com

Suggested Readings:

- Jon Duckett, *HTML & CSS: Design and Build Websites*, Wiley, 2011
- Eric Freeman, *Head First JavaScript Programming*, O'Reilly, 2014
- David Flanagan, *JavaScript: The Definitive Guide*, O'Reilly, 2020

Recommended Software/Tools:

1. VS Code, Chrome DevTools
2. Postman, MongoDB Compass
3. React DevTools, Git

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	–	2	1	–	1	-
CO2	2	3	2	1	1	–	–	2	1
CO3	3	2	2	1	1	1	1	1	2
CO4	2	3	3	1	2	2	–	1	2
CO5	3	2	1	2	2	1	1	2	2
Articulation	2.6	2.4	1.8	1.25	1.6	1.25	1	1.4	1.75

Course Title:	**Cloud Computing Foundations**
Course Code:	**25VMT0C204**
Semester:	**II**
Credits:	**4**
Course Type	Core Course
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course introduces fundamental concepts of cloud computing, including service models, deployment models, virtualization, and cloud architecture. Students will gain hands-on experience with major cloud platforms (AWS/Azure/GCP) and learn to deploy, manage, and secure cloud-based solutions. The course emphasizes practical applications through labs and real-world case studies.

Course Outcomes (COs)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Understand cloud computing concepts and service models.	L1
CO2	Deploy and manage virtualized infrastructure.	L3
CO3	Analyze cloud storage and networking solutions.	L4
CO4	Evaluate cloud security and compliance frameworks.	L5
CO5	Develop cloud-native applications using PaaS.	L3

Course Modules

No. of Hours:24

Module 1: Cloud Computing Fundamentals
Scope of Module: Covers basic concepts, characteristics, and service models of cloud computing.
Topics: Definition and evolution of cloud computing; Essential characteristics of cloud (NIST model); Service models: IaaS, PaaS, SaaS; Deployment models: Public, Private, Hybrid, Community; Benefits and challenges of cloud adoption; Cloud computing use cases; Cloud vs. traditional IT infrastructure; Cloud service providers overview (AWS, Azure, GCP); Cloud service level agreements (SLAs); Multitenancy and resource pooling; Elasticity and scalability; Cloud computing cost models and pricing; Case study: Cloud migration strategy; Green computing and cloud efficiency; Cloud-first strategies in digital transformation.
Learning Outcome: *Students will be able to differentiate between cloud service and deployment models, and evaluate when and how to migrate to cloud environments.*

No. of Hours:24

Module 2: Virtualization and Cloud Infrastructure
Scope of Module: Focuses on virtualization technologies and cloud infrastructure management.
Topics: Virtualization concepts and benefits; Types of virtualization: Server, Storage, Network; Hypervisors: Type 1 (bare-metal) and Type 2 (hosted); Virtual machine creation and lifecycle; Introduction to containers and Docker;

Container orchestration basics (Kubernetes overview); Infrastructure as a Service (IaaS); AWS EC2, Azure VMs, GCP Compute Engine; Cloud autoscaling and elasticity; Elastic Load Balancing (ELB); Monitoring and cloud resource usage; Infrastructure as Code: Terraform and CloudFormation; Hands-on: Launching a VM on AWS/GCP; Cloud cost optimization techniques; Best practices for resource provisioning.
Learning Outcome: *Students will be able to configure and manage virtualized infrastructure on cloud platforms using IaaS and container technologies.*

No. of Hours:24

Module 3: Cloud Storage and Networking
Scope of Module: Explores storage solutions and network architectures in cloud environments.
Topics: Cloud storage types: Object, Block, File; Storage services: Amazon S3, Azure Blob, Google Cloud Storage; Data redundancy and availability zones; Content Delivery Networks (CDNs) Edge computing and latency reduction; Data lifecycle and storage classes; Virtual Private Cloud (VPC) architecture; Subnets, gateways, and NAT; Cloud firewalls and security groups Cloud DNS and networking services; VPNs and hybrid cloud connectivity ;Database as a Service (DBaaS): RDS, Cosmos DB, Cloud SQL; Data backup and disaster recovery; Case study: Designing a high availability architecture; Storage and network performance benchmarking.
Learning Outcome: *Students will be able to design and configure secure, high-performance storage and networking solutions within cloud environments.*

No. of Hours:24

Module 4: Cloud Security and Compliance
Scope of Module: Addresses security challenges and compliance requirements in cloud computing.
Topics: Cloud security principles and models; Shared responsibility model; Identity and Access Management (IAM); Role-based access control (RBAC); Data encryption in transit and at rest; Key management services; Security groups vs. network ACLs; Firewall policies and VPN security; Security incident detection and response; Cloud compliance frameworks (GDPR, HIPAA, ISO 27001); Security posture assessments; Penetration testing and vulnerability scanning; Logging and auditing best practices; Case study: Securing a multi-tenant cloud app. Mini-project: Implementing IAM and encryption in AWS.
Learning Outcome: *Students will be able to implement cloud security measures and ensure compliance with data protection regulations and security frameworks.*

No. of Hours:24

Module 5: Cloud-Native Application Development
Scope of Module: Covers development of applications optimized for cloud environments
Topics: Introduction to cloud-native principles; Twelve-factor app methodology; Serverless architecture overview; AWS Lambda and Azure Functions; Event-

driven programming in the cloud; Microservices architecture and benefits; REST API design and testing (Postman); Containers vs. serverless vs. VMs; CI/CD pipelines: GitHub Actions, AWS CodePipeline; Monitoring and logging (CloudWatch, Stackdriver); Application scaling and performance tuning; API Gateway integration and throttling; DevOps tools for cloud (Jenkins, Terraform) Case study: Microservices deployment; Capstone Project: Build and deploy a serverless application.
Learning Outcome: *Students will be able to develop, deploy, and monitor scalable, cloud-native applications using modern cloud services and DevOps tools.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10

Prerequisites (If Any):

- Basic knowledge of operating systems and networking concepts.

Pedagogy / Teaching Methodology:

- ✓ Hands-on labs with cloud platforms
- ✓ Case studies of enterprise cloud migrations

- ✓ Security audit simulations
- ✓ Group projects with real-world scenarios

Certificate / Value Added Courses Recommended:

1. AWS Cloud Practitioner: https://aws.amazon.com/certification/
2. Microsoft Azure Fundamentals: https://learn.microsoft.com/en-us/certifications/

Suggested Readings:

- ✓ Dan Marinescu, *Cloud Computing: Theory and Practice*, 3rd Ed., Morgan Kaufmann, 2022
- ✓ Thomas Erl, *Cloud Computing: Concepts, Technology & Architecture*, Prentice Hall, 2021
- ✓ Official documentation from AWS, Azure, and GCP

Recommended Software/Tools:

1. AWS Free Tier, Azure Portal
2. Docker, Kubernetes (basic)
3. Terraform, CloudFormation

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)

6.	Indian Knowledge System

CO PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	–	2	1	–	1	1
CO2	2	3	2	1	-	–	–	2	1
CO3	3	2	2	1	-	1	2	1	2
CO4	2	3	3	1	2	2	–	1	2
CO5	3	2	1	2	2	-	2	2	2
Articulation	2.6	2.4	1.8	1.25	2	1.33	2	1.5	1.6

Course Title:	Environmental, Social, and Governance (ESG) in IT
Course Code:	25VMT0C205
Semester:	II
Credits:	4
Course Type	AECC
Learning Hours	120
Live Sessions	12 hours

Course Description:

This course explores the integration of Environmental, Social, and Governance (ESG) principles into information technology. It covers sustainable IT practices, ethical AI, data privacy, and corporate governance in tech organizations. Students will learn to assess ESG risks, implement green computing solutions, and align IT strategies with global sustainability goals through case studies and projects.

Course Outcomes (Cos)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Explain ESG frameworks relevant to IT	L1
CO2	Apply green computing techniques	L3
CO3	Analyze social impacts of AI and big data	L4
CO4	Evaluate governance models for IT ethics	L5
CO5	Evaluate sustainable IT policies for organizations	L5

Course Modules

No. of Hours:24

Module 1: ESG Fundamentals in IT
Scope of Module: Introduces core ESG concepts and their application in technology sectors.
Topics: Understanding ESG in the context of IT; Global ESG frameworks: UN SDGs, GRI, SASB; Environmental impacts of digital infrastructure; Carbon footprint of data centers and cloud computing; E-waste management and recycling strategies; IT lifecycle sustainability assessments; Role of ICT in sustainable development goals; Green certifications for IT (Energy Star, EPEAT); Sustainability KPIs in IT projects; ESG compliance for tech startups AI and sustainability intersections; Digital ethics and sustainable design; Sustainable supply chain management in IT; Case study: Google's carbon-neutral cloud initiatives; Future of ESG in emerging technologies
Learning Outcome: *Students will articulate how ESG principles apply to IT infrastructure, service models, and sustainability reporting.*

No. of Hours:24

Module 2: Green Computing
Scope of Module: Focuses on sustainable IT practices and resource optimization.
Topics: Green hardware lifecycle management; Energy-efficient processor design;

Power-aware scheduling algorithms; Green software design and refactoring; Virtualization for sustainability; Resource optimization in cloud environments; Carbon-aware computing; Renewable energy integration in data centers; Measuring IT carbon footprints; Green networking and IoT devices; Cooling technologies and data center PUE; Server consolidation strategies; Green code practices in DevOps; Life Cycle Assessment (LCA) tools for IT systems; Lab: Calculate emissions from virtual machines.
Learning Outcome: *Students will be able to apply green computing principles to minimize the environmental impacts of IT operations and software systems.*

No. of Hours:24

Module 3: Social Responsibility in Tech
Scope of Module: Examines ethical implications of AI, data, and digital inclusion.
Topics: Digital equity and inclusion challenges; Algorithmic bias and mitigation techniques; AI transparency and explainability; Ethical data collection and consent; GDPR, CCPA, and global data privacy laws; Bias in training data for machine learning; Accessible design principles in software; Digital literacy and education programs; Tech for underserved communities. Social media and societal impact; AI in healthcare and social justice; Tech activism and digital rights; Workplace inclusivity in tech companies; Case study: Microsoft’s AI for Accessibility Lab: Bias detection in data classification models.
Learning Outcome: *Students will analyze social implications of emerging technologies and advocate for inclusive, ethical digital innovation.*

No. of Hours:24

Module 4: IT Governance and Ethics
Scope of Module: Covers governance structures for ethical IT decision-making.
Topics: Foundations of IT corporate governance; ESG reporting frameworks and tools; Whistleblower policies and ethics channels; IT ethics boards and oversight; Compliance versus ethical leadership; IT audit trails and data governance; Transparency and accountability practices; Ethical AI governance and regulation; Cybersecurity ethics in surveillance systems; Boardroom simulation: resolving tech dilemmas; Balancing profit and public interest in IT firms; Decision-making under ethical uncertainty; ISO standards for IT governance; ESG ratings and public disclosures; Case study: Governance failure in a tech giant.
Learning Outcome: *Students will evaluate and develop governance frameworks that support ethical decision-making and stakeholder alignment in IT environments.*

No. of Hours:24

Module 5: Sustainable IT Strategy
Scope of Module: Integrates ESG principles into organizational IT planning.

Topics:
Integrating ESG into IT strategy roadmaps; Stakeholder mapping and engagement models; Circular economy in IT hardware; Supply chain resilience and transparency; Cloud sustainability strategy; Responsible tech innovation lifecycle; Designing ESG-aligned KPIs for IT; Tech sustainability scorecards and benchmarking; Policy modeling for net-zero IT operations; Carbon offsetting through tech initiatives; Funding and investing in green IT Organizational change for ESG transformation; Capstone: ESG audit of a mock IT enterprise Scenario planning for climate-resilient IT; Report writing: ESG strategy white paper.

Learning Outcome: *Students will design sustainable IT policies and long-term strategic plans incorporating ESG values for enterprise impact.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Basic knowledge of IT systems and corporate social responsibility.

Pedagogy:

- ✓ Guest lectures from ESG tech leaders

- ✓ Carbon accounting workshops
- ✓ Ethics hackathons
- ✓ ESG report analysis

Certifications Recommended:

1. Certified Green IT Professional (GITP)
2. SASB's ESG Fundamentals

Readings:

- "Green IT for Sustainable Business Practice" by Mark O'Neill
- "AI Ethics" by Mark Coeckelbergh

Tools:

- AWS Customer Carbon Footprint Tool
- ESG reporting software (Sustainalytics)

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)

6	Indian Knowledge System

CO PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	2	1	-	3	1	-	2	1	2
CO2	3	2	1	2	3	1	-	2	1
CO3	1	3	2	-	2	3	1	1	3
CO4	2	1	3	2	1	2	3	-	2
CO5	3	2	2	3	2	1	2	2	3
Articulation	2.6	1.8	2	2.5	1.8	1.75	2	1.5	2.2

CENTRE FOR DISTANCE AND ONLINE EDUCATION

ONLINE

Semester III

Corporate Office
91/2, Dr. A N Krishna Rao Road V V Puram,
Bengaluru - 560 004
P: +91 80 4343 1000

Centre for Distance & Online Education
319, 25th Main, 17th Cross,
J P Nagar 6th Phase
Bengaluru – 560 078
P: +91 80 4343 0100

Course Title:	**Artificial Intelligence and Machine Learning**
Course Code:	**251VMT0C301**
Semester:	**III**
Credits:	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides MCA students with an industry-relevant and application-driven exploration of Artificial Intelligence (AI) and Machine Learning (ML). The syllabus is designed to offer a progressive and logical learning path from AI fundamentals to advanced ML techniques and deployment strategies. The course also emphasizes ethical, explainable, and production-ready AI systems through a mix of practical labs, case studies, and modern tools.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO No.	Course Outcome	BTL
CO1	Understand the evolution and fundamentals of Artificial Intelligence and Machine Learning.	L1
CO2	Apply advanced ML algorithms for supervised, unsupervised, and semi-supervised learning tasks.	L3
CO3	Analyze deep learning models and reinforcement learning strategies.	L4

CO4	Evaluate models for bias, interpretability, and effectiveness across varied datasets.	L5
CO5	Design and deploy scalable, explainable, and secure AI/ML applications.	L3

Course Modules

No. of Hours:24

Module 1: Foundations of AI and ML
Scope of Module: Establishes core concepts of AI and ML, setting the stage for in-depth exploration of models, learning paradigms, and industry practices.
Topics: AI and ML definitions and key differences, history and evolution of AI, AI vs ML vs DL, symbolic vs sub-symbolic AI, types of ML (supervised, unsupervised, reinforcement), ML lifecycle and pipeline stages, CRISP-DM methodology, introduction to datasets and data types, common data challenges, bias and fairness in ML, ML vs traditional programming, use cases in various industries, recent trends in AI (Generative AI, LLMs), AI ethics overview.
Learning Outcome: *Students will be able to articulate core AI/ML definitions, classify learning types, and describe modern trends and challenges.*

No. of Hours:24

Module 2: Advanced Supervised Learning and Ensemble Models
Scope of Module:

Covers core and advanced supervised learning techniques including ensemble strategies and hyperparameter tuning.
Topics: Linear and logistic regression with regularization (Ridge, Lasso), decision trees, random forest, support vector machines (SVM), KNN classification, Naive Bayes, model evaluation metrics (accuracy, precision, recall, F1, ROC-AUC), model selection and validation strategies (train/test split, cross-validation), bias-variance tradeoff, grid search and random search, ensemble learning (bagging, boosting), introduction to XGBoost, LightGBM, and CatBoost, multi-output and multiclass classification, managing imbalanced datasets (SMOTE, undersampling), SHAP for tree models.
Learning Outcome: *Students will be able to apply and evaluate supervised learning and ensemble methods to solve classification and regression tasks.*

No. of Hours:24

Module 3: Unsupervised Learning, Dimensionality Reduction, and Anomaly Detection
Scope of Module: Explores a wide range of unsupervised techniques and introduces preprocessing and anomaly detection as critical skills.
Topics: Clustering algorithms: k-means, DBSCAN, hierarchical, Gaussian Mixture Models (GMM), elbow method and silhouette score, association rule mining (Apriori, FP-Growth), dimensionality reduction techniques: PCA, t-SNE, UMAP, feature selection vs extraction, feature importance, anomaly detection using Isolation Forest, LOF, one-class SVM, preprocessing: encoding, scaling,

normalization, dealing with missing values, pipeline creation using scikit-learn, unsupervised model evaluation strategies.
Learning Outcome:* -Students will be able to perform unsupervised learning, reduce dimensionality for visualization, and detect data anomalies effectively.*

No. of Hours:24

Module 4: Deep Learning and Reinforcement Learning
Scope of Module: Introduces deep neural networks and reinforcement learning paradigms for intelligent decision making.
Topics: Artificial Neural Networks (ANN), forward and backward propagation, activation functions (ReLU, sigmoid, tanh), loss functions and optimizers, CNNs for image processing, RNNs, LSTMs for sequential data, autoencoders and applications, GANs and generative models, overview of BERT and Transformer models, introduction to reinforcement learning, Q-learning and policy gradients, environment-reward-agent loop, OpenAI Gym, applications in recommendation, robotics, gaming.
Learning Outcome:* -Students will be able to build and analyze neural networks and apply RL for decision-making scenarios.*

No. of Hours:24

Module 5: AI/ML Model Deployment, MLOps, and Explainability
Scope of Module: Focuses on end-to-end AI/ML model deployment, production readiness, monitoring, and responsible AI practices.
Topics: Deployment using Flask, FastAPI, Streamlit, Docker containerization, introduction to Kubernetes for ML, CI/CD pipelines in ML, MLflow and model versioning, explainable AI tools (SHAP, LIME, ELI5), fairness audits and bias detection tools, logging and monitoring models in production, introduction to cloud ML platforms (AWS SageMaker, GCP Vertex AI), securing ML models and data, case studies from industry, best practices in production ML systems. ***Learning Outcome: -****Students will be able to deploy, explain, and monitor AI/ML systems using modern DevOps practices and frameworks*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Strong understanding of Python programming, statistics, and data structures.

Pedagogy / Teaching Methodology:

- ✓ Project-based learning with datasets from real-world domains
- ✓ Lab-based exercises using Jupyter and Google Colab
- ✓ Mini case studies and research paper reviews
- ✓ Live demonstrations and group activities
- ✓ Tool walkthroughs and guided deployment

Certificate / Value Added Courses Recommended:

1. Machine Learning Engineering for Production (MLOps) – Coursera
2. Deep Learning Specialization – Coursera
3. AI for Everyone – Coursera
4. Google ML Crash Course – https://developers.google.com/machine-learning/crash-course

Suggested Readings:

- Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, O'Reilly, 2022
- Ian Goodfellow, Deep Learning, MIT Press, 2016
- Sutton & Barto, Reinforcement Learning: An Introduction, MIT Press, 2018
- Kelleher, D. J., Data Science Ethics, MIT Press
- Relevant research from IEEE, ACM, and arXiv.org

Recommended Software/Tools:

1. Python, Scikit-learn, TensorFlow, Keras, PyTorch
2. Google Colab / JupyterLab, MLflow, Optuna
3. Docker, FastAPI, Streamlit, AWS SageMaker, GCP Vertex AI
4. SHAP, LIME, OpenAI Gym

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO-PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	1	2	-	1	1	2
CO2	3	2	3	2	2	1	-	2	2
CO3	2	3	2	3	2	1	2	-	2
CO4	2	2	2	-	2	3	2	2	2
CO5	3	3	3	2	3	2	2	3	3
Articulation	2.6	2.4	2.2	2	2.2	1.75	1.75	2	2.2

Course Title:	**Python Programming**
Course Code:	**251VMT0C302**
Semester:	**III**
Credits:	**4**
Course Type	Core Course
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides MCA students with foundational to advanced skills in Python programming, blending basic concepts with real-world applications. It encompasses control structures, functions, OOP, file operations, exception handling, unit testing, web development, API integration, and data analysis. Special emphasis is placed on writing efficient, modular, and scalable Python code with exposure to industry-grade tools and frameworks.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO No.	Course Outcome	BTL
CO1	Understand Python syntax, control flow, data structures, and functions.	L1
CO2	Apply OOP principles and modular coding in real-world applications.	L3
CO3	Analyze problems and implement file, error handling, and unit testing.	L4

CO4	Evaluate web-based interactions, APIs, and database connectivity using Python.	L5
CO5	Develop interactive data-driven applications using Pandas, NumPy, and visualization libraries.	L3

Course Modules

No. of Hours:24

Module 1: Python Basics, Syntax, and Control Flow
Scope of Module: Introduces core elements of Python such as syntax, variables, data types, operators, and control flow.
Topics: Installing Python and IDE setup, writing and executing Python scripts, identifiers, keywords, variables, constants, data types, type conversion, operators and precedence, conditional statements (if, elif, else), iterative control (for, while), break and continue, pass statement, nested loops, input/output handling, formatted string literals (f-strings), Python coding style (PEP8).
Learning Outcome: *Students will be able to write structured Python scripts using control flow constructs and input/output operations.*

No. of Hours:24

Module 2: Functions, Collections, Iterators, and Modules
Scope of Module:

Covers function design, advanced collection operations, and module creation for reusable code.
Topics: Defining and invoking functions, argument types (*args, **kwargs), scope and lifetime, recursion, anonymous functions (lambda), Python built-in functions, comprehension (list, dict, set), advanced list methods, tuples and unpacking, sets and operations, dictionaries and operations, iterators and generators, creating custom modules, importing built-in modules (math, sys, os, datetime, calendar, random), packaging with pip and venv.
Learning Outcome: *Students will be able to modularize programs using custom and built-in Python functions and libraries.*

No. of Hours:24

Module 3: Object-Oriented Programming, File I/O, and Exception Handling
Scope of Module: Explores OOP principles, file operations, and robust exception handling techniques.
Topics: Classes and objects, class and instance variables, methods, constructor and destructor, inheritance and types, method overriding, polymorphism, encapsulation, use of __str__, __repr__, __len__, working with text/binary/CSV files, reading and writing files, context manager and file closing, exception types and hierarchy, try-except-finally, raising and handling custom exceptions, logging module basics, assertions for debugging.

Learning Outcome: *Students will be able to design OOP-based applications with efficient file handling and error management*

No. of Hours:24

Module 4: Testing, Automation, and Database Connectivity
Scope of Module: Provides practices in automated testing, scripting for task automation, and integrating Python with databases.
Topics: Writing test cases using unittest and pytest, test discovery and test suites, test-driven development, automation with shutil and subprocess, working with file system using os and pathlib, scheduling tasks, introduction to SQLite and PostgreSQL, executing queries from Python using sqlite3 and psycopg2, performing CRUD operations, using context managers with databases, parameterized queries to prevent SQL injection.
Learning Outcome: *Students will be able to automate tasks and integrate databases securely in Python programs.*

No. of Hours:24

Module 5: Web Development, APIs, and Data Science Libraries
Scope of Module: Bridges Python to web and data science fields using APIs, Flask, and data analysis tools

Topics:
Flask framework structure and routing, GET and POST methods, HTML rendering with Jinja2 templates, form data processing, REST API consumption using requests, OpenWeatherMap API integration example, Pandas: dataframes, merging, filtering, groupby, NumPy: array creation and operations, Matplotlib/Seaborn for plotting, exporting data to Excel/CSV, introduction to Streamlit for interactive dashboards, mini project combining Flask and Pandas.
Learning Outcome: *Students will be able to develop and deploy data-centric and interactive Python web applications*

Module	**No. of Pre-Recordings**
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Basic logic building, familiarity with any programming language is helpful.

Pedagogy / Teaching Methodology:

- ✓ Concept-based demonstrations using Jupyter and IDEs
- ✓ Mini assignments and use-case-driven labs

✓ Group activities and debugging challenges

✓ Integration of GitHub for version control practices

✓ End-of-module review quizzes and projects

Certificate / Value Added Courses Recommended:

1. Python for Everybody – Coursera
2. Python OOP and Flask Projects – Udemy
3. Automate the Boring Stuff with Python – Online Book
4. Introduction to Data Science in Python – Coursera

Suggested Readings:

- Mark Lutz, Learning Python, O'Reilly
- Eric Matthes, Python Crash Course, No Starch Press
- Luciano Ramalho, Fluent Python, O'Reilly
- Python Documentation – https://docs.python.org

Recommended Software/Tools:

1. Python 3.x, PyCharm, VS Code
2. JupyterLab, Google Colab
3. Flask, Pandas, NumPy, Matplotlib, Streamlit
4. GitHub, SQLite, PostgreSQL

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global

2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO-PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	2	1	2	1	-	1	2
CO2	3	2	3	2	3	1	2	2	3
CO3	3	3	-	3	2	-	2	2	2
CO4	2	3	3	3	3	2	2	2	-
CO5	3	3	2	2	3	2	2	3	3
Articulation	2.8	2.6	2.5	2.2	2.6	1.7	2	2	2.5

Course Title:	**Software Testing**
Course Code:	**251VMT1S303**
Semester:	**III**
Credits:	**4**
Course Type	Core Course
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course equips MCA students with essential and advanced software testing methodologies. It combines theoretical foundations with hands-on practices, including test case design, automation tools, agile testing strategies, and defect tracking. The course also emphasizes modern trends in software quality assurance and continuous testing within CI/CD pipelines.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO No.	Course Outcome	BTL
CO1	Understand the fundamentals of software testing, life cycles, and testing strategies	L1
CO2	Apply functional and non-functional testing techniques to different types of software systems	L3

CO3	Analyze testing requirements and create test plans, test cases, and traceability matrices	L4
CO4	Evaluate defect tracking systems, quality metrics, and static and dynamic testing outcomes	L5
CO5	Design and implement automation test scripts using industry tools and frameworks	L3

Course Modules

No. of Hours:24

Module 1: Introduction to Software Testing and SDLC/STLC
Scope of Module: Introduces software testing, the software development life cycle (SDLC), and testing life cycle models to establish foundational knowledge.
Topics: Basics of software testing, Importance of testing in SDLC, Software Testing Life Cycle (STLC), Testing principles, Verification and Validation, Quality assurance vs Quality control, Software development models (Waterfall, Agile, V-Model), Levels of testing (Unit, Integration, System, Acceptance), Types of testing (Functional, Non-functional), Static vs Dynamic testing, Cost of quality, Defect leakage and defect clustering, Test deliverables and exit criteria
Learning Outcome: *Students will understand testing concepts, life cycles, and differentiate between various testing strategies and models.*

No. of Hours:24

Module 2: Test Design, Planning, and Techniques

Scope of Module:

Focuses on systematic test case design, planning and estimation, traceability, and functional testing strategies.

Topics:
Test planning and strategy, Requirement analysis and traceability matrix, Test case creation and prioritization, Test scenario vs test case, Test estimation techniques (Work Breakdown Structure, Function Point), Black-box techniques (Equivalence partitioning, Boundary value analysis, Decision tables, State transition testing), White-box techniques (Statement, Branch, Path, Condition testing), Grey-box testing, Risk-based testing, Exploratory testing, Test data preparation, Entry and exit criteria.

Learning Outcome: *Students will develop the ability to prepare efficient test plans and test cases using formal techniques and testing principles.*

No. of Hours:24

Module 3: Defect Lifecycle, Reviews, and Quality Metrics

Scope of Module:
Introduces defect tracking and quality review processes and evaluates testing effectiveness using metrics.

Topics:
Defect life cycle and defect states, Severity and priority classification, Defect tracking tools (JIRA, Bugzilla), Root cause analysis techniques, Static testing techniques (Reviews, Walkthroughs, Inspections), Peer reviews and audit process, Software quality attributes (Reliability, Usability, Maintainability), Software quality models (ISO 9126, McCall's Model), Metrics for test effectiveness (Defect density, Test coverage, Mean time to failure), Reporting and dashboarding tools, Quality audits and test maturity models

Learning Outcome: *Students will be able to monitor, analyze, and improve software quality using defects and process metrics.*

No. of Hours:24

Module 4: Test Automation and Frameworks

Scope of Module:

Explores automation frameworks, scripting basics, and integration of automation within development pipelines.

Topics:
Need and scope of test automation, Selenium WebDriver fundamentals, TestNG and JUnit frameworks, Page Object Model (POM), Keyword-driven and Data-driven frameworks, Automation in Agile and DevOps, CI/CD integration with Jenkins, BDD with Cucumber and Gherkin, Python/Java scripting for automation, Locators and XPath, Logging and reporting in automation, Debugging automation scripts, Introduction to mobile automation (Appium), Cross-browser testing.

Learning Outcome: *Students will be able to write automation scripts and use tools like Selenium and Jenkins for continuous testing*

No. of Hours:24

Module 5: Advanced Testing and Emerging Trends

Scope of Module:

Covers specialized testing types and introduces modern testing practices including AI in testing.

Topics:
Performance testing using JMeter and LoadRunner, Security testing basics (OWASP Top 10, SQL Injection, Cross-site scripting), API testing using Postman, Agile testing concepts (Scrum, Sprint, TDD, BDD), Regression testing strategies, Mobile testing tools and cloud device labs, Cloud-based testing platforms (Sauce Labs, BrowserStack), AI/ML in testing (test generation, defect prediction), Test environment setup and virtualization, Ethical considerations in automated testing, Shift-left and shift-right testing approaches

Learning Outcome: *Students will gain insights into specialized domains, emerging tools, and trends in software testing.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites (If Any):

- Programming knowledge (Python/Java preferred)
- Understanding of SDLC and basic software engineering principles

Pedagogy / Teaching Methodology:

- ✓ Hands-on labs with Selenium, JMeter, and Jenkins
- ✓ Mini-projects involving test planning and automation scripting
- ✓ Real-world case studies and defect simulation exercises
- ✓ Peer reviews and group presentations
- ✓ Weekly quizzes and scenario-based problem solving

Certificate / Value Added Courses Recommended:

1. Selenium WebDriver with Java – Udemy
2. Software Testing and Automation – Coursera
3. Test Automation University – Free Online Platform
4. JMeter Load Testing – Pluralsight
5. ISTQB Foundation Level – Official Training Resources

Suggested Readings:

- Ron Patton, *Software Testing*, Pearson
- Glenford Myers, *The Art of Software Testing*, Wiley
- Rex Black, *Foundations of Software Testing*
- Lisa Crispin & Janet Gregory, *Agile Testing*
- Online Resources: Ministry of Testing, Test Automation University, OWASP

Recommended Software/Tools:

1. Selenium WebDriver, JUnit/TestNG
2. JIRA, Bugzilla, Jenkins
3. JMeter, Postman
4. Cucumber, PyTest
5. Visual Studio Code / IntelliJ IDEA

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO-PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	2	1	1	2	-	1	2
CO2	2	2	2	2	2	2	2	2	3
CO3	3	3	3	3	-	2	2	2	-
CO4	2	2	2	-	2	3	2	-	3
CO5	3	3	3	2	3	2	2	3	2
Articulation	2.6	2.4	1.8	2	2	2.2	2	2	2.5

Semester IV

Corporate Office
91/2, Dr. A N Krishna Rao Road V V Puram,
Bengaluru - 560 004
P: +91 80 4343 1000

Centre for Distance & Online Education
319, 25th Main, 17th Cross,
J P Nagar 6th Phase
Bengaluru – 560 078
P: +91 80 4343 0100

Course Title:	**IT Project Management**
Course Code:	**251VMT1S401**
Semester:	**IV**
Credits:	**4**
Course Type	DSE
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course introduces MCA students to the principles, tools, and techniques of Information Technology (IT) Project Management. It focuses on the complete project lifecycle—from initiation and planning to execution, monitoring, and closure. Students gain hands-on experience with project scheduling, cost estimation, risk assessment, stakeholder engagement, and agile methodologies using real-world case studies and tools.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Assess the fundamental principles of IT project management and project lifecycle stages	L3
CO2	Apply project planning techniques including scope, time, and cost estimation for IT-based projects	L3
CO3	Analyze project risks, resource allocation, and stakeholder roles in IT project success	L4

CO4	Evaluate and monitor IT project performance using metrics, tools, and quality standards	L5
CO5	Design IT project frameworks using traditional and agile methodologies, tools, and documentation	L3

Course Modules

No. of Hours:24

Module 1: Introduction to Project Management and Life Cycl

Scope of Module:
Covers foundational concepts of project management and introduces the phases in IT project lifecycles.

Topics:
Definition and characteristics of a project, Importance of project management in IT, Project lifecycle phases (Initiation, Planning, Execution, Monitoring, Closure), Triple constraint (Scope, Time, Cost), Project manager roles and responsibilities, Organizational structures and their influence on projects, Project selection methods (NPV, IRR, Payback period), Overview of PMBOK and PRINCE2 frameworks, Project charter creation, Business case and stakeholder identification.

Learning Outcome: *Students will understand the lifecycle, structure, and strategic role of IT projects in business environments.*

No. of Hours:24

Module 2: Project Planning – Scope, Time, and Cost Management

Scope of Module:
Focuses on detailed project planning involving deliverables, timelines, budgets, and work breakdown structures.

Topics:
Project scope definition, Work Breakdown Structure (WBS), Activity sequencing and dependency determination, Critical Path Method (CPM), Gantt charts and PERT diagrams, Time

estimation techniques (Analogous, Parametric, Three-point), Cost estimation methods, Budgeting and cost baseline, Resource leveling and allocation, Tools: MS Project, OpenProject, Project scope statement, Scope creep and change control.

Learning Outcome: Students will be able to prepare complete project plans, schedules, and cost estimates using project management tools.

No. of Hours:24

Module 3: Risk, Quality, and Communication Management

Scope of Module:
Explores risk analysis, quality assurance practices, and effective stakeholder ommunication strategies.

Topics:
Risk identification and analysis (qualitative and quantitative), Risk register and mitigation planning, Risk response strategies, Quality planning, assurance, and control, Cost of quality, Benchmarking and continuous improvement, Communication plan and stakeholder engagement, Communication channels and models, Conflict management strategies, Tools for collaboration (Slack, Microsoft Teams), Project reporting and meeting management

Learning Outcome: *Students will be able to manage risks and quality while establishing clear communication plans in IT projects.*

No. of Hours:24

odule 4: Project Execution, Monitoring, and Control

Scope of Module:
Deals with project tracking, performance measurement, and managing changes effectively.

Topics:
Project execution workflows, Resource management, Project performance metrics (EV, PV, AC, CPI, SPI), Earned Value Management (EVM), Change control systems and processes, Issue logs and escalation mechanisms, Scope verification and control, Schedule and cost variance analysis, Corrective and preventive actions, Project dashboards and automated reporting, Agile project boards (Trello, Jira)

Learning Outcome: *Students will gain the ability to monitor and control project performance using industry-standard techniques***.**

No. of Hours:24

Module 5: Agile Project Management and Emerging Trends

Scope of Module:
Introduces Agile, Scrum, hybrid approaches, and emerging trends in IT project execution.

Topics:
Agile principles and manifesto, Scrum framework: roles, ceremonies, artifacts, Product backlog and sprint planning, Burndown charts, Kanban methodology, Hybrid project management models, Project governance, DevOps culture in project delivery, Use of AI and analytics in project prediction, Sustainability and green IT project considerations, Project closure and lessons learned documentation.

Learning Outcome: *Students will be able to implement Agile and hybrid project models and stay informed about current trends in IT project management.*

Module	**No of Pre-Recordings**
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10.

Prerequisites (If Any):

- Basic knowledge of software engineering and systems analysis
- Familiarity with business scenarios and IT development processes

Pedagogy / Teaching Methodology:

- ✓ Use of case studies and role-play simulations
- ✓ Real-world project planning using MS Project, Trello, or Jira
- ✓ Team-based projects and SCRUM sprints
- ✓ Guest sessions from project managers and industry consultants
- ✓ Continuous assessments via presentations, reviews, and progress reports

Certificate / Value Added Courses Recommended:

1. Project Management Professional (PMP) – PMI
2. Google Project Management: Coursera
3. Scrum Master Certification – Scrum.org
4. Agile Project Management – LinkedIn Learning
5. Microsoft Project Training – edX

Suggested Readings:

- Kathy Schwalbe, *Information Technology Project Management*, Cengage
- Harold Kerzner, *Project Management: A Systems Approach to Planning, Scheduling, and Controlling*, Wiley

- Bob Hughes & Mike Cotterell, *Software Project Management*, McGraw-Hill
- Ken Schwaber, *Agile Project Management with Scrum*, Microsoft Press
- PMI's *PMBOK® Guide* – Latest Edition

Recommended Software/Tools:

1. Microsoft Project, Trello, Jira
2. GanttProject, OpenProject, Wrike
3. Google Workspace, MS Teams, Slack
4. Lucidchart or Draw.io for diagrams
5. GitHub Projects for Agile workflows

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6	Indian Knowledge System

CO-PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	2	1	2	2	2	2	2
CO2	3	2	-	2	2	2	2	1	-
CO3	3	3	2	3	2	1	-	2	3
CO4	2	2	3	3	2	3	2	-	1
CO5	3	3	3	-	3	2	2	3	3
Articulation	2.6	2.4	2.4	2.25	2.2	2	2	2	2.25

Course Title:	Big Data Analytics
Course Code:	251VMT1S402
Semester:	IV
Credits:	4
Course Type	DSE
Learning Hours	120
Live Sessions	12 hours

Course Description:

This course empowers MCA students to understand, process, and analyze massive datasets using modern big data technologies. It combines foundational concepts with practical tools like Hadoop, Spark, and NoSQL databases, and focuses on solving real-world problems through data engineering and analytics pipelines. Emphasis is also placed on data governance, scalability, and ethical handling of big data.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Understand the characteristics of big data, the data analytics lifecycle, and big data ecosystem	L1
CO2	Apply Hadoop and MapReduce techniques for distributed data processing	L3
CO3	Analyze real-time and batch data processing using Spark and NoSQL systems	L4

CO4	Evaluate big data architectures, tools, and their applications across various domains	L5
CO5	Design scalable data pipelines and perform analytics using big data frameworks and case studies	L3

Course Modules:

No. of Hours:24

Module 1: Introduction to Big Data and Analytics Lifecycle

Scope of Module:
Provides foundational knowledge of big data, its properties, and the end-to-end analytics process.

Topics:
Definition and characteristics of big data (3Vs, 5Vs), Structured vs semi-structured vs unstructured data, Data sources and types, Data analytics lifecycle (discovery, preparation, modeling, evaluation, deployment), Challenges in big data processing, Traditional vs big data analytics, Big data architecture overview, Big data use cases in business, healthcare, and IoT, Introduction to the Hadoop ecosystem.***Learning Outcome:*** *Students will understand the concepts and significance of big data and analytics lifecycle.*

No. of Hours:24

Module 2: Hadoop Ecosystem and Distributed Data Processing

Scope of Module:
Explores the Hadoop ecosystem for distributed storage and batch processing using MapReduce.

Topics:
HDFS architecture and file system operations, Hadoop cluster setup (single and multi-node), MapReduce programming model, Input/output formats in MapReduce, Combiners and partitioners, Job scheduling and optimization, Hadoop streaming, YARN: Yet Another

Resource Negotiator, Introduction to Hive and Pig, Data ingestion tools (Sqoop, Flume), HBase basics, Use cases of batch processing

Learning Outcome: *Students will apply Hadoop and MapReduce to perform distributed processing over large datasets.*

No. of Hours:24

Module 3: Apache Spark and In-Memory Processing

Scope of Module:
Focuses on real-time and in-memory data processing using Spark and its libraries.

Topics:
Introduction to Apache Spark, RDDs and transformations/actions, Spark vs Hadoop MapReduce, Spark architecture and execution flow, DataFrames and Spark SQL, Caching and persistence, Spark MLlib for machine learning, Structured streaming and windowed operations, Integrating Spark with Hive, Spark on YARN/Mesos, Hands-on with PySpark or Scala, Case studies using real-time data.

Learning Outcome:

Students will analyze batch and real-time data using Spark's high-performance framework.

No. of Hours:24

Module 4: NoSQL Databases and Data Management

Scope of Module:
Introduces scalable data storage techniques using NoSQL systems and explores data governance.

Topics:
RDBMS vs NoSQL databases, Types of NoSQL: Key-Value (Redis), Document (MongoDB), Column (HBase), Graph (Neo4j), CAP theorem and BASE properties, Data modeling in NoSQL, CRUD operations and indexing in MongoDB, ACID vs eventual consistency, Introduction to

Cassandra, Sharding and replication strategies, Data governance, metadata management, data lineage, Ethical handling of big data

Learning Outcome: *Students will be able to model and query large datasets using NoSQL databases effectively.*

No. of Hours:24

Module 5: Big Data Analytics Tools and Emerging Trends

Scope of Module:
Covers big data visualization, analytics pipelines, and current trends in AI, cloud, and edge analytics.

Topics:
Big data visualization with Tableau and Power BI, Integration with cloud platforms (AWS EMR, GCP BigQuery, Azure Data Lake), Data pipeline orchestration (Apache NiFi, Airflow), Real-world analytics pipelines (ELT/ETL), Role of big data in AI/ML, Streaming analytics with Kafka and Spark, Data security and privacy issues, Data lakes vs data warehouses, Edge computing for big data, Big data trends: Data mesh, data fabric, federated learning

Learning Outcome: *Students will design scalable analytics workflows and explore modern applications and tools in big data.*

Module	**No. of Pre-Recordings**
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites (If Any):

- Good understanding of Python/Java
- Basics of databases and data structures
- Exposure to cloud and distributed systems is helpful but not mandatory

Pedagogy / Teaching Methodology:

- ✓ Tool-driven demonstrations using Hadoop, Spark, MongoDB
- ✓ Lab sessions with real-world datasets (healthcare, finance, e-commerce)
- ✓ Mini-projects and pipeline simulations
- ✓ Group case study analysis and presentation
- ✓ Cloud-based labs and deployment exercises

Certificate / Value Added Courses Recommended:

1. Big Data Specialization – Coursera (UC San Diego)
2. Hadoop Platform and Application Framework – Coursera
3. Big Data Analysis with Spark – edX
4. MongoDB University Free Courses
5. Data Engineering with Google Cloud – Coursera

Suggested Readings:

- Tom White, *Hadoop: The Definitive Guide*, O'Reilly
- Holden Karau, *Learning Spark*, O'Reilly
- Kristina Chodorow, *MongoDB: The Definitive Guide*
- Jules S. Damji, *Spark: The Definitive Guide*, O'Reilly

- Latest research articles and tutorials from Databricks, Cloudera, AWS

Recommended Software/Tools:

1. Hadoop, Hive, Pig, HBase
2. Apache Spark (Scala/PySpark)
3. MongoDB, Cassandra
4. Tableau, Power BI
5. Kafka, Apache NiFi, Apache Airflow
6. AWS EMR, GCP BigQuery, Azure Synapse

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO-PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	2	2	2	2	1	1	-
CO2	3	-	3	2	2	2	2	2	3
CO3	3	3	2	3	2	2	2	2	2
CO4	2	2	3	-	3	3	2	3	3
CO5	3	3	2	3	3	3	2	3	3
Articulation	2.6	2.4	2.2	2.2	2.4	2.4	1.8	2.2	2.2

Course Title:	**Low-Code/No-Code Application Development**
Course Code:	**251VMT1S403**
Semester:	**IV**
Credits:	**4**
Course Type	**DSE**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course empowers MCA students to rapidly prototype, build, and deploy applications using modern low-code and no-code platforms. It focuses on building user interfaces, automating workflows, integrating data, and deploying scalable applications. Through hands-on labs and mini-projects, students gain real-world experience using platforms like Microsoft Power Apps, OutSystems, Mendix, AppSheet, and Airtable, while understanding their limitations and governance best practices.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Understand the principles and architecture of low-code/no-code application development platforms	L1
CO2	Apply platform features to design user interfaces, workflows, and data-driven applications	L3
CO3	Analyze application requirements and choose suitable tools, data models, and logic components	L4

CO4	Evaluate the scalability, performance, and security considerations of low-code/no-code applications	L5
CO5	Design and deploy functional apps with third-party service integration and user access management	L3

Course Modules

No. of Hours:24

Module 1: Introduction to Low-Code/No-Code Development

Scope of Module:
Covers the fundamentals, scope, and value of low-code/no-code approaches in software development.

Topics:
Definition and evolution of low-code/no-code (LCNC), Traditional coding vs LCNC development, Advantages and limitations, Key components: UI builder, workflow engine, data connectors, Use cases in business, citizen development, and IT, Architecture of LCNC platforms, Types of LCNC platforms: general-purpose, BPM, mobile-first, Examples: Power Apps, Mendix, OutSystems, Appgyver, Bubble..

Learning Outcome: *Students will understand the ecosystem and use cases of low-code/no-code tools in modern IT environments.*

No. of Hours:24

Module 2: Application Design and Interface Development

Scope of Module:
Focuses on designing user interfaces and application structure using visual programming elements.

Topics:
Form and screen design, Drag-and-drop components, Theme customization and branding, UI logic and expressions, Responsive design principles, Navigation flows and page transitions,

Mobile-first vs desktop-first design, Best practices in UX/UI with LCNC tools, Creating reusable components, Prototyping and wireframing tools, Role-based interface views

Learning Outcome: Students will be able to design user-friendly interfaces for applications using visual components.

No. of Hours:24

Module 3: Data Modeling, Workflow Automation, and Logic

Scope of Module:
Introduces data modeling, visual logic building, and automation with triggers and flows**.**

Topics:
Entity and relationship modeling, Creating data tables and collections, CRUD operations with forms, Triggers and events, Conditional logic and rules, Visual workflow builders (Power Automate, Zapier), Email/SMS notifications, Scheduling and recurrence, Data validation and formatting, Integrating APIs using connectors, Basic scripting or expressions (formula logic, JSON, Excel-style expressions)

Learning Outcome: *Students will automate tasks and build intelligent workflows to support business logic without traditional coding.*

No. of Hours:24

Module 4: Integration, Deployment, and Governance

Scope of Module:
Focuses on connecting applications with external services, publishing apps, and handling access control.

Topics:
Connecting with external data (Google Sheets, SharePoint, SQL), REST API integration using connectors, Webhooks and external triggers, Authentication and user management, Role-based access control (RBAC), Deployment options: cloud, mobile, internal portals, Version control and rollback, Governance: permissions, audit logs, change management, Application lifecycle management (ALM) in LCNC tools

Learning Outcome: *Students will integrate third-party services, control access, and deploy applications effectively.*

No. of Hours:24

Module 5: Advanced Features, Use Cases, and Trends

Scope of Module:
Explores advanced LCNC features, emerging trends, and capstone project building.

Topics:
AI integration in LCNC platforms (Power Automate AI Builder, AppSheet AI), Embedding chatbots and smart assistants, Progressive web apps (PWAs), Offline capabilities, Performance tuning in large apps, Cross-platform development strategies, Use cases: employee onboarding app, ticketing system, CRM dashboard, Emerging trends: LCNC + DevOps, fusion teams, citizen developers, Capstone project (individual/group): end-to-end app

Learning Outcome: *Students will build end-to-end applications and explore trends shaping the future of software development.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum no. of pre-recordings:10

Prerequisites (If Any):

- Basic understanding of databases and software development concepts
- Familiarity with Excel formulas and workflows is helpful

Pedagogy / Teaching Methodology:

- ✓ Hands-on labs using Power Apps, AppSheet, or OutSystems
- ✓ Mini-projects in small teams simulating business problems
- ✓ Use of templates and walkthroughs from real platforms
- ✓ Peer demos, app reviews, and iterative feedback
- ✓ Access to trial versions or sandbox environments

Certificate / Value Added Courses Recommended:

1. Microsoft Power Platform Fundamentals – Microsoft Learn
2. Google AppSheet Certification – Qwiklabs
3. OutSystems Developer Certification
4. No-Code Foundations – Udemy
5. Citizen Developer Program – PMI Institute

Suggested Readings:

- Phil Simon, *Low-Code/No-Code: Citizen Developers and the Surprising Future of Business Applications*
- Microsoft Learn: Power Apps and Power Automate documentation

- AppSheet and Mendix official blogs
- Tutorials from OutSystems, Bubble, and Glide
- Gartner & Forrester reports on LCNC market trends

Recommended Software/Tools:

1. Microsoft Power Apps, Power Automate
2. Google AppSheet, Zapier, Glide
3. OutSystems, Mendix, Airtable, Bubble
4. Trello or Jira for project management
5. Postman for API testing (where applicable)

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6	Indian Knowledge System

CO-PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	2	2	2	2	1	1	2
CO2	3	3	3	2	-	2	2	-	-
CO3	3	2	3	1	2	3	-	2	1
CO4	2	3	2	3	1	3	2	1	3
CO5	3	3	3	3	3	3	2	3	-
Articulation	2.6	2.6	2.6	2.2	2	2.6	1.4	1.4	2

Course Title:	**Data Visualization**
Course Code:	**251VMT1S404**
Semester:	**IV**
Credits:	**4**
Course Type	**DSE**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course equips MCA students with practical and theoretical knowledge in data visualization. It covers principles of visual perception, storytelling with data, and hands-on experience using tools like Tableau, Power BI, Python (Matplotlib, Seaborn, Plotly), and Excel. The course emphasizes effective communication of complex data, dashboard creation, interactivity, and ethical aspects of visual analytics in business and research contexts.

Course Outcomes (COs):

CO No.	Course Outcome	BTL
CO1	Asses the foundations and importance of data visualization and visual perception principles	L3
CO2	Apply data visualization tools and libraries to create static and interactive charts	L3
CO3	Analyze datasets to derive insights and select appropriate visual encodings and storytelling techniques	L4
CO4	Evaluate visualizations based on accuracy, aesthetics, interactivity, and data integrity	L5

CO5	Design dashboards and visual reports for decision-making across domains	L3

No of Hours:24

Module 1: Introduction to Data Visualization and Visual Perception

Scope of Module:
Introduces the fundamentals of visual analytics, human perception, and the role of data visualization in decision-making.

Topics:
Definition and purpose of data visualization, Importance in data analysis and communication, Evolution of visual analytics, Elements of visualization: marks and channels, Human visual perception (pre-attentive attributes, Gestalt principles), Principles of good visual design, Common chart types (bar, line, pie, scatter, histogram), Data types and visualization mappings, Misleading visualizations and ethics

Learning Outcome: *Students will understand foundational concepts and identify the key principles for effective visualization.*

No of Hours:24

Module 2: Data Preparation and Visualization with Python Libraries

Scope of Module:
Covers data wrangling and visualizing datasets using Python tools.

Topics:
Pandas for data handling, Data cleaning and transformation, Matplotlib: line plots, bar charts, subplots, Seaborn: categorical plots, heatmaps, pairplots, Plotly for interactivity, Customizing plots (labels, colors, themes), Annotating and exporting charts, Visualizing time series and distributions, Combining plots with grids, Introduction to Dash for web-based visuals.

Learning Outcome: *Students will use Python to process data and build compelling visualizations for analytics purposes.*

No of Hours:24

Module 3: Interactive Dashboards with Tableau and Power BI

Scope of Module:
Focuses on using industry-standard BI tools to create dashboards and reports.

Topics:
Overview of Tableau and Power BI interfaces, Connecting to various data sources (Excel, SQL, CSV, APIs), Calculated fields and expressions, Filters, hierarchies, and parameters, Interactive visualizations (tooltips, actions), Dashboards and stories, Data blending and joins, Drill-down and drill-through reports, Publishing and sharing reports, Performance optimization and best practices

Learning Outcome: *Students will be able to create dynamic, interactive dashboards and reports for decision-makers using BI tools.*

No of Hours:24

Module 4: Advanced Visualization Techniques and Storytelling

Scope of Module:
Teaches how to build narratives and apply advanced visual design techniques for data storytelling.

Topics:
Data storytelling frameworks, Choosing the right chart for the message, Using annotations and color meaningfully, Funnel, bullet, donut, treemap, and sunburst charts, Geospatial visualization with maps, Highlighting trends and outliers, Visualizing multivariate data, Time-based storytelling, Case studies: Business, healthcare, education, Avoiding cognitive overload and clutter.

Learning Outcome: *Students will craft visually engaging and meaningful stories using advanced visualization techniques.*

No of Hours:24

Module 5: Real-Time Dashboards, Web Embedding, and Visualization Ethics

Scope of Module:
Explores real-time dashboards, embedding visualizations in web apps, and responsible data visualization.

Topics:
Real-time data streaming (Power BI + Azure, Plotly + Dash), API-based visual updates, Embedding dashboards into websites/apps, Exporting dashboards and presentations, Visualization performance considerations, Accessibility in visualizations, Ethical data representation, Avoiding bias in charts, Data privacy and consent, Final project: End-to-end dashboard with real dataset.

Learning Outcome: *Students will deploy interactive, real-time dashboards and apply ethical principles in visualization.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Min no of pre recordings 10.

Prerequisites (If Any):

- Basic understanding of Python and data analysis
- Familiarity with databases or spreadsheets is helpful

Pedagogy / Teaching Methodology:

1. Tool-based tutorials using Python, Tableau, and Power BI
2. Lab assignments using open datasets (Kaggle, World Bank, etc.)
3. Mini projects focused on domain-specific dashboards
4. Visualization critiques and peer feedback
5. Real-time dashboard walkthroughs and team presentations

Certificate / Value Added Courses Recommended:

1. Tableau Desktop Specialist Certification
2. Power BI Data Analyst Associate – Microsoft
3. Data Visualization with Python – Coursera
4. Data Storytelling for Business – LinkedIn Learning
5. Introduction to Dash and Plotly – Plotly Academy

Suggested Readings:

- Alberto Cairo, *The Truthful Art*
- Cole Nussbaumer Knaflic, *Storytelling with Data*
- Nathan Yau, *Visualize This*
- Edward Tufte, *The Visual Display of Quantitative Information*
- Tableau, Power BI, and Plotly official documentation

Recommended Software/Tools:

1. Python (Pandas, Matplotlib, Seaborn, Plotly, Dash)

2. Tableau Desktop / Public
3. Microsoft Power BI Desktop
4. Excel (for data prep)
5. Google Data Studio (optional)

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO-PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	2	2	2	2	1	-	2
CO2	3	3	3	1	2	2	-	2	3
CO3	3	3	1	3	1	2	2	2	-
CO4	2	1	3	3	2	-	2	-	3
CO5	3	3	3	3	3	3	2	3	3
Articulation	2.6	2.4	2.4	2.4	2	2.25	1.75	2.33	2.75

Course Title:	**Strategic Leadership in IT**
Course Code:	**251VMT0C405**
Semester:	**IV**
Credits:	**4**
Course Type	**Skill Enhancement Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course explores the critical role of leadership in driving strategic IT initiatives within organizations. It focuses on leadership theories, IT governance, digital transformation strategies, risk management, and innovation leadership. Practical case studies and simulations are integrated to build decision-making skills required for aligning IT with business strategies.

Course Outcomes (Cos)

Upon successful completion of this course, learners will be equipped to:

CO	**Course Outcome**	**BTL**
CO 1	Understand leadership theories and their application in IT strategy formulation.	L1
CO 2	Evaluate and apply IT governance frameworks and practices.	L5
CO 3	Design digital transformation strategies aligned with business goals.	L6

CO 4	Identify and mitigate IT-related risks through strategic leadership.	L4
CO 5	Demonstrate innovation and change management skills in IT leadership contexts.	L3

Course Modules

No. of Hours:24

Module 1: Foundations of Strategic Leadership and IT Alignment

Scope of Module:
Covers the core leadership theories and IT-business alignment models.

Topics:
Defining leadership and management in the context of IT; Strategic leadership competencies; Leadership styles: transformational, transactional, situational; Vision building and strategic thinking in IT; Models of IT-business alignment (Henderson & Venkatraman Model); Organizational structures and their impact on IT leadership; Emotional intelligence in IT leadership; Ethical leadership in technology environments; Leading virtual and remote IT teams; Leadership maturity models; Case Study: IT leadership challenges in global enterprises.

Learning Outcome: *Students will be able to understand leadership theories and how to align IT strategies with broader business objectives.*

No. of Hours:24

Module 2: IT Governance and Strategic Control

Scope of Module:
Focuses on frameworks and practices for effective IT governance.

Topics:
Introduction to IT governance and compliance; COBIT, ITIL, and ISO/IEC 38500 frameworks; Governance of Enterprise IT (GEIT) principles; Strategic decision rights and accountability in IT; Metrics and KPIs for IT governance; Managing outsourcing and vendor relationships; Sarbanes-Oxley Act and IT implications; Risk appetite and tolerance in IT governance;

Blockchain's role in IT governance; Shadow IT and governance challenges; Audit management and IT policy development; Practical Exercise: Designing an IT governance framework for an SME.

Learning Outcome: *Students will be able to evaluate, select, and apply IT governance frameworks to ensure strategic control and value delivery.*

No. of Hours:24

Module 3: Digital Transformation Leadership

Scope of Module:
Develops skills to lead digital transformation initiatives.

Topics:
Drivers of digital transformation in organizations; Role of CIOs and CTOs in transformation efforts; Technology adoption frameworks (e.g., Diffusion of Innovations); Agile and DevOps in digital leadership; Managing cultural change during transformation; Leadership in cloud, AI, and emerging technologies adoption; Case Study: Successful vs. failed digital transformation initiatives.

Learning Outcome: *Students will be able to design strategic digital transformation initiatives aligned with business innovation needs.*

No. of Hours:24

Module 4: IT Risk Management and Strategic Decision-Making

Scope of Module:
Focuses on managing strategic risks in IT leadership roles.

Topics:
Understanding IT-related risks: security, operational, reputational; Risk management lifecycle in IT projects; Enterprise risk management frameworks (COSO ERM); Crisis leadership and contingency planning; Ethical considerations and cybersecurity leadership; Legal and regulatory compliance (e.g., GDPR, HIPAA); Threat modeling and strategic defense planning;

Business continuity planning and IT disaster recovery leadership; Strategic handling of data breaches and incident response, Practical: IT risk assessment simulation

Learning Outcome: *Students will be able to analyze, manage, and mitigate IT risks strategically to ensure organizational resilience.*

***N*o. of Hours:24**

Module 5: Innovation, Change Management, and Future of IT Leadership

Scope of Module:

Covers fostering innovation and managing technological change.

Topics:

Leading organizational innovation in IT; Change management models (Kotter's 8-Step, ADKAR); Talent development and IT leadership pipelines; Diversity, inclusion, and ethics in IT leadership; Future trends: Quantum Computing, Web3, and Leadership implications;

Building a sustainable IT innovation culture; Intrapreneurship and fostering innovation within IT teams; Managing resistance to technological change; IT leadership in the age of Automation and RPA (Robotic Process Automation); Ethical considerations in AI and emerging tech leadership; Capstone: Strategic Leadership Plan for a tech startup.

Learning Outcome: *Students will be able to demonstrate strategic leadership skills in driving innovation, change management, and future-proofing IT organizations*

Module	No. of Pre-Recordings
1	10
2	12
3	10

4	**12**
5	**10**

Note: Min no of pre recordings 10.

Prerequisites (If Any):

- Basic knowledge of IT management concepts and organizational behavior.

Pedagogy / Teaching Methodology:

- ✓ Interactive case studies and leadership simulations
- ✓ Group strategy projects and role plays
- ✓ Workshops on IT governance frameworks
- ✓ Industry expert talks and guest lectures

Certificate / Value Added Courses Recommended:

1. IT Leadership Certificate by edX (MIT Sloan)
2. COBIT 2019 Foundation Certification
3. Digital Transformation Strategy by Coursera (Boston University)

Suggested Readings:

- John P. Kotter, Leading Change, Harvard Business Review Press, 2012.
- Peter Weill & Jeanne W. Ross, IT Governance: How Top Performers Manage IT Decision Rights for Superior Results, Harvard Business Review Press, 2004.
- George Westerman, Leading Digital: Turning Technology into Business Transformation, Harvard Business Review Press, 2014.

Recommended Software/Tools:

1. MS Project, Trello, Jira (for project leadership)

2. ServiceNow, SolarWinds (for IT governance)
3. Risk assessment tools (RSA Archer, RiskLens)

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	-	2	1	-	2	1
CO2	3	3	2	1	2	1	-	2	-
CO3	2	3	3	2	3	2	1	3	2
CO4	3	2	2	2	2	1	1	2	2
CO5	2	2	2	1	3	2	1	2	2
Articulation	2.6	2.4	2	1.5	2.4	1.4	1	2.2	1.4

Course Matrix

ELECTIVE: Data Analytics

Elective: Data Analytics

Semester-I

Course Code	Course Title	Course Category	CA	ESE	Credits
25VMT0C101	Mathematical Foundations for Computer Science	CC	30	70	4
25VMT0C102	Computing Concepts and Problem Solving using C	CC	30	70	4
25VMT0C103	Operating Systems	CC	30	70	4
25VMT0C104	Data Structures	CC	30	70	4
25VMT0L101	Problem Solving using C Lab	SEC	100	-	2
25VMT0L102	Data Structures Lab	SEC	100	-	2
26VMX0S105	AI Skills for Future	SEC	30	70	2
Total No of Credits					**22**

Semester-II

Course Code	Course Title	Course Category	CA	ESE	Credits
25VMT0C201	Advanced Database Systems	CC	30	70	4
25VMT0C202	Java Programming	CC	30	70	4
25VMT0C203	Web Technologies	CC	30	70	4
25VMT0C204	Cloud Computing Foundations	CC	30	70	4
25VMT0L201	Advanced Database Systems Lab	SEC	100	-	2
25VMT0L202	Java Programming Lab	SEC	100	-	2
25VMT0C205	Environmental, Social, and Governance (ESG) in IT	AECC	30	70	4
Total No of Credits					**24**

Semester-III

Course Code	Course Title	Course Category	CA	ESE	Credits
251VMT0C301	Artificial Intelligence and Machine Learning	CC	30	70	4
251VMT0C302	Python Programming	CC	30	70	4
251VMT4S303	Computer Vision	DSE	30	70	4
251VMT0L301	AI & Machine Learning Lab	SEC	100	-	2
251VMT0L302	Python Programming Lab	SEC	100	-	2
251VMT0C301	Open Elective	OE	30	70	4
Total No of Credits					**20**

Semester-IV

Course Code	Course Title	Course Category	CA	ESE	Credits
251VMT4S401	Natural Language Processing	DSE	30	70	4
251VMT4S402	Predictive Analytics using Machine Learning	DSE	30	70	4
251VMT4S403	Deep Learning	DSE	30	70	4
251VMT4S404	Big Data Hadoop	DSE	30	70	4
251VMT0C405	Strategic Leadership in IT	SEC	30	70	2
251VMT0P406	Project	Thesis	30	70	6
Total No of Credits					**24**

Legend

CC	Core Course
DSE	Discipline Specific Elective
SEC	Skill Enhancement Courses
AECC	Ability Enhancement Compulsory Course
OE	Open Elective

Semester I

Courses

Corporate Office
91/2, Dr. A N Krishna Rao Road V V Puram,
Bengaluru - 560 004
P: +91 80 4343 1000

Centre for Distance & Online Education
319, 25th Main, 17th Cross,
J P Nagar 6th Phase
Bengaluru – 560 078
P: +91 80 4343 0100

Course Title:	**Mathematical Foundations for Computer Science**
Course Code:	**25VMT0C101**
Semester:	**I**
Credits:	**4**
Course Type:	**Core Course**
Learning Hours:	120
Live Sessions:	**12**

Course Description:

This course builds a strong mathematical base essential for various domains in computer science such as algorithms, cryptography, artificial intelligence, and data science. It covers discrete structures, logic, combinatorics, graph theory, and number theory. Emphasis is laid on both theoretical understanding and application to computational problems.

Course Outcomes (Cos)

Upon successful completion of this course, learners will be equipped to:

CO	**Course Outcome**	**BTL**
CO1	Understand the fundamentals of logic, sets, and relations used in computer science.	L1
CO2	Apply principles of combinatorics and counting for solving algorithmic and real-world problems.	L3
CO3	Analyze and construct graphs, trees, and networks relevant to data structures and networking.	L4
CO4	Solve recurrence relations and use mathematical induction in algorithm design.	L3

CO5	Evaluate number theoretic and algebraic methods in cryptographic and coding applications.	L5

Course Modules

No. of Hours: 24

Module 1: Logic, Sets and Relations

Scope of Module:
Provides foundational concepts in mathematical logic, set theory, and relations essential for modeling computation and designing algorithms.

Topics:
Propositions and logical operators, Truth tables, Tautology and contradiction, Predicates and quantifiers, Rules of inference, Logical equivalence, Methods of proof, Set theory basics, Venn diagrams, Set operations, Cartesian product of sets, Types of relations, Properties of relations, Equivalence and partial order relations, Closure of relations

Learning Outcome: *Students will be able to construct logical arguments and model relationships using sets and relations.*

No. of Hours: 24

Module 2: Combinatorics and Counting

Scope of Module:
Focuses on combinatorial problem-solving techniques relevant to algorithm complexity and probability theory.

Topics:
Basic counting principles, Addition and multiplication rules, Permutations and combinations, Pigeonhole principle, Principle of inclusion and exclusion, Binomial theorem, Pascal's triangle, Combinatorial identities, Multisets and combinations with repetition, Generating functions, Recurrence relations, Solving linear recurrence, Introduction to partitions, Derangements, Applications in probability

Learning Outcome: *Students will be able to apply combinatorial reasoning to solve complex counting and probability problems.*

No. of Hours: 24

Module 3: Graph Theory and Trees

Scope of Module:
Introduces graph-based models to represent and solve problems in data structures, networks, and algorithms.

Topics:
Basic concepts of graphs, Types of graphs, Graph representations, Degree of vertices, Paths and cycles, Connected components, Eulerian and Hamiltonian graphs, Planar graphs, Graph coloring, Matrix representation of graphs, Trees and their properties, Binary and n-ary trees, Tree traversal algorithms, Spanning trees, Minimum spanning tree algorithms (Kruskal, Prim)

Learning Outcome:*Students will be able to model and solve problems using graphs and trees in computer science.*

No. of Hours: 24

Module 4: Recurrence Relations and Mathematical Induction

Scope of Module:
Covers recurrence relations and inductive reasoning techniques used in algorithm analysis and design.

Topics:
Sequences and recurrence relations, Solving homogeneous linear recurrence, Non-homogeneous recurrence, Generating functions for recurrence, Divide-and-conquer relations, Recursion trees and Master's theorem, Mathematical induction, Strong induction, Well-ordering principle, Applications in algorithm correctness, Tower of Hanoi, Binary search, Fibonacci sequence, Ackermann's function

Learning Outcome: *Students will be able to formulate and solve recurrence relations and validate algorithms using induction.*

No. of Hours: 24

Module 5: Number Theory and Algebraic Structures

Scope of Module:
Explores concepts in number theory and abstract algebra crucial for cryptography, hashing, and finite state systems.

Topics:
Divisibility and Euclidean algorithm, Prime numbers and factorization, Congruence relations, Modular arithmetic, Fermat's little theorem, Euler's theorem, Chinese Remainder Theorem, Greatest common divisor (GCD), Least common multiple (LCM), Rings and fields basics, Group theory introduction, Subgroups and cyclic groups, Applications in cryptography, Hashing functions, Finite automata fundamentals

Learning Outcome: *Students will be able to apply number theoretic and algebraic techniques in areas like security and finite state systems.*

Module	**No of Pre-Recordings**
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Basic mathematics at undergraduate level, understanding of algebra and logic.

Pedagogy / Teaching Methodology:

- ✓ Conceptual and problem-based learning
- ✓ Interactive class exercises and proofs
- ✓ Real-world applications and modeling
- ✓ Assignments and quizzes using math software

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. Discrete Mathematics by NPTEL: https://nptel.ac.in/courses/111106086
2. Mathematics for Computer Science by MIT OpenCourseWare: https://ocw.mit.edu

Suggested Readings:

- Kenneth H. Rosen, Discrete Mathematics and Its Applications, McGraw-Hill, 2017
- Tremblay and Manohar, Discrete Mathematical Structures, Tata McGraw-Hill, 2001
- C.L. Liu and D.P. Mohapatra, Elements of Discrete Mathematics, McGraw-Hill, 2010
- Online resources from IEEE Xplore, SpringerLink for mathematical modeling and computation

Recommended Software/Tools (If applicable):

1. MATLAB/Octave
2. SageMath
3. Python (SymPy, NetworkX)
4. GeoGebra for visualization

Curriculum Development:

Sl.No	Identify and highlight in the module
1	Local, Regional, National and Global
2	Employability, Entrepreneurship and Skill Development
3	Value Added Courses (Module)
4	Integration of Digital Learning Tools
5	Crossing Cutting Edges (Ethics, Gender Equality, Sustainability and Environmental concerns)
6	Indian Knowledge System

CO PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	–	2	1	–	1	1
CO2	2	3	2	1	1	–	–	2	1
CO3	3	2	2	1	1	1	1	1	2
CO4	2	3	3	1	2	2	–	1	2
CO5	3	2	1	2	2	1	1	2	2
Articulation	2.6	2.4	1.8	1.25	1.6	1.25	1	1.4	1.6

Note: The Articulation is indicative of how a particular course maps to a PO.

Articulation is calculated as:

$$=\frac{Sum\ of\ CO\ Mapping\ Levels\ for\ a\ particular\ PO}{No.of\ COs\ that\ mapped\ to\ that\ PO}$$

Example:1

$$Articulation\ of\ PO1 = \frac{(3+2+3+2+3)}{5} = 2.6$$

Example: 2

$$Articulation\ of\ PO7 = \frac{(1+1)}{2} = 1.0$$

Course Title	**Computing Concepts and Problem Solving using C**
Course Code	**25VMT0C102**
Semester	**I**
Credits	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course introduces students to core computing concepts and problem-solving strategies using the C programming language. It provides a blend of theoretical foundations and hands-on programming practices. Students will develop algorithmic thinking, explore C's system-level capabilities, and apply structured programming techniques. The course advances from basic programming constructs to memory management, file handling, and real-time applications, preparing students for industry-level coding challenges.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO No.	**Course Outcome**	**BTL**
CO1	Understand the fundamentals of computing, algorithms, and structured programming in C.	L1
CO2	Apply control structures, functions, arrays, and strings to solve basic computational tasks.	L3

CO3	Analyze memory management and pointer operations for efficient program execution.	L 4
CO4	Develop and evaluate system-level programs including file handling and dynamic memory usage.	L 5
CO5	Design and implement real-time applications and case studies using C programming constructs.	L 5

Course Modules

No. of Hours: 24

Module 1: Fundamentals of Computing and C Programming

Scope of Module:
Introduces basic computing concepts, algorithm design, and foundational elements of the C programming language.

Topics:
Basics of Computers: Architecture, memory, input/output units; Introduction to Algorithms and Flowcharts; C Programming Basics: Structure, syntax, variables, constants, data types; Operators: Arithmetic, relational, logical, bitwise, assignment, conditional; Compilation and Execution Process.

Learning Outcome: *Students will be able to describe computing concepts, write simple C programs, and understand compilation processes.*

No. of Hours: 24

Module 2: Control Structures, Functions, Arrays, Strings, and Modular Programming

Scope of Module:
Explores problem-solving strategies using control flow, modular programming, and essential data structures.

Topics:
Control Structures: if, if-else, nested if, switch-case; Loops: for, while, do-while, nested loops; Functions: Declaration, definition, call, recursion; Scope, Storage Classes (auto, static, extern, register); Modular Programming: Header files, preprocessor directives (#define, #include); Arrays: 1D, 2D arrays, array manipulation; Strings: String functions (strlen, strcpy, strcat, strcmp), string manipulation.

Learning Outcome:*Students will be able to apply control structures, modularize code, and manipulate arrays and strings effectively.*

No. of Hours: 24

Module 3: Pointers, Structures, Dynamic Memory Allocation, and File Handling

Scope of Module:
Provides in-depth understanding of memory management, structured data, and file operations in C.

Topics:
Introduction to Pointers: Pointer arithmetic, pointers and arrays; Pointers to Functions and Pointers to Structures; Structures and Unions: Declaration, definition, accessing members; Dynamic Memory Allocation: malloc, calloc, realloc, free; File Handling: File operations (read, write, append), file pointers; Case Study: Creating a file-based student management system.

Learning Outcome:*Students will be able to manipulate memory using pointers, manage structured data, and perform file operations.*

No. of Hours: 24

Module 4: System-Level Programming and Real-Time Applications in C

Scope of Module:
Focuses on industry-relevant applications, system programming concepts, and real-time project development.

Topics:
Command-line Arguments and Environment Variables; Error Handling (perror, strerror) and Debugging Techniques; Mini Projects: Text-based Games (Tic-Tac-Toe), Simple Shell Implementation, Real-Time Data Logger for IoT (file-based); Case Study: Simulating a Banking System.

Learning Outcome: *Students will be able to design and implement system-level programs and real-time applications using C.*

No. of Hours: 24

Module 5: Advanced Industry Applications and Case Studies

Scope of Module:
Encourages application of C programming skills in solving industry-specific problems and case studies.

Topics:
Writing basic command-line tools; Simple network programs (client-server); Debugging and optimizing small C programs; Working with file compression and encryption tools; Implementing simple database file storage; Case Study: Building a basic calculator or file management tool; Mini Project: Simple automation program, data logger, or system monitor.

Learning Outcome*: Students will be able to apply C programming skills to solve advanced, industry-relevant problems through case studies and projects.*

Module	No of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Basic understanding of computer systems and logic design.

Pedagogy / Teaching Methodology:

- ✓ Hands-on programming in lab environments (using GCC, Code::Blocks)
- ✓ Algorithm design and implementation exercises
- ✓ Mini projects and group coding activities
- ✓ Debugging and system-level programming demonstrations
- ✓ Peer code reviews and optimization tasks

Certificate / Value Added Courses Recommended:

1. Introduction to C Programming – Coursera
2. Advanced C Programming – Udemy
3. System Programming with C and Linux – edX

Suggested Readings:

- E. Balagurusamy, Programming in ANSI C, McGraw-Hill Education

- Kernighan & Ritchie, The C Programming Language, Pearson
- Yashavant Kanetkar, Let Us C, BPB Publications
- Recent industry articles and whitepapers on system-level programming

Recommended Software/Tools:

1. GCC (GNU Compiler Collection)
2. Code::Blocks / Visual Studio Code
3. GDB (GNU Debugger)
4. Valgrind (Memory management tool)
5. Linux-based terminal for system programming

Curriculum Development:

Sl.No	**Identify and highlight in the module**
1	Local, Regional, National and Global
2	Employability, Entrepreneurship and Skill Development
3	Value Added Courses (Module)
4	Integration of Digital Learning Tools
5	Crossing Cutting Edges (Ethics, Gender Equality, Sustainability and Environmental concerns)
6	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO-PO Mapping Table:

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	3	2	2	1	1	–	–	2
CO2	3	3	3	2	2	1	–	–	2
CO3	3	3	3	3	2	2	1	–	3
CO4	3	3	3	3	3	2	1	1	3
CO5	3	3	3	3	3	3	2	2	3
Articulation	3	3	2.8	2.6	2.2	1.8	1.25	1.5	2.6

Course Title	Operating Systems
Course Code	25VMT0C103
Semester	I
Credits	4
Course Type	Core Course
Learning Hours	120
Live Sessions	12 hours

Course Description:

This course provides MCA students with a comprehensive understanding of operating system concepts, focusing on process management, memory management, storage systems, and security mechanisms. The syllabus integrates theoretical knowledge with hands-on lab experiences, progressing from fundamental OS principles to advanced topics like virtualization, containerization, and real-time systems. The course prepares students for system-level programming, cloud infrastructure management, and security in modern computing environments.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO No.	Course Outcome	BTL
CO1	Understand the fundamental concepts and architecture of operating systems.	L1
CO2	Apply process management, scheduling, and synchronization mechanisms in modern OS environments.	L3

CO3	Analyze memory management, virtual memory, and storage management techniques.	L4
CO4	Evaluate advanced OS topics like virtualization, containerization, and security models.	L5
CO5	Design and implement small-scale system utilities or simulators focusing on OS principles.	L5

Course Modules

No. of Hours: 24

Module 1: Introduction to Operating Systems and Process Management

Scope of Module:
Introduces core OS concepts, structures, and the fundamentals of process management.

Topics:
Overview of Operating Systems: Definitions, functions, types (batch, time-sharing, distributed, real-time); OS structure and components; System calls and APIs; Microkernel vs Monolithic Kernel architectures; System Boot Process and Initialization (GRUB, systemd); Process concepts, process states, process control block (PCB); Process scheduling algorithms (FCFS, SJF, Round Robin, Priority); Context switching; Process scheduling in multi-core and NUMA systems; Case studies on modern OS architectures (Linux, Windows).

Learning Outcome: *Students will be able to describe OS structures, functions, and manage basic process scheduling scenarios.*

No. of Hours: 24

Module 2: Process Synchronization, Deadlock, and CPU Scheduling

Scope of Module:
Covers critical section problems, synchronization mechanisms, deadlock handling, and CPU scheduling strategies.

Topics:
Critical section problem, Peterson's solution, semaphores, monitors; Spinlocks, Mutexes, and Condition Variables; Hardware support for synchronization (Test-and-Set, Compare-and-Swap); Classic synchronization problems (producer-consumer, readers-writers, dining philosophers); Deadlock characterization, prevention, avoidance (Banker's algorithm), detection, recovery, and resource preemption; CPU scheduling strategies (multilevel queue, multilevel feedback queue); Real-world scheduling policies (CFS in Linux, Windows scheduler comparison); Case studies on synchronization in Linux kernel.

Learning Outcome: *Students will be able to apply synchronization techniques and deadlock handling strategies in process management.*

No. of Hours: 24

Module 3: Memory Management, Virtual Memory, and Storage Systems

Scope of Module:
Focuses on memory organization, virtual memory, and file systems management.

Topics:
Memory management strategies: contiguous allocation, paging, segmentation; Virtual memory: demand paging, page replacement algorithms (LRU, FIFO, Optimal); Thrashing and performance tuning; Memory-mapped I/O and Direct Memory Access (DMA); Swap space management and tuning; File system interface and implementation, allocation methods (contiguous, linked, indexed); Directory structures, file system mounting, journaling file systems; Advanced file system topics: RAID levels, ZFS/Btrfs file systems; Security implications in storage systems (encryption at rest); Case study: EXT4 file system analysis.

Learning Outcome:*Students will be able to analyze memory and storage management strategies in OS environments.*

No. of Hours: 24

Module 4: Virtualization, Containerization, and Real-Time Systems

Scope of Module:
Explores virtualization technologies, container-based systems, and introduces real-time OS concepts.

Topics:
Virtualization concepts, hypervisors (Type 1 and Type 2), virtual machine monitors (VMWare, KVM); Nested Virtualization (running VMs inside VMs); Containerization with Docker, Kubernetes orchestration basics; Container Security (namespaces, cgroups, seccomp); Comparison of VMs vs Containers; Performance tuning of VMs and Containers; Introduction to real-time operating systems (RTOS), real-time scheduling algorithms; RTOS performance metrics and real-world RTOS examples (FreeRTOS, VxWorks); Case study: Docker-based microservices in cloud environments.

Learning Outcome:*Students will be able to evaluate virtualization and containerization technologies and apply real-time scheduling concepts.*

No. of Hours: 24

Module 5: Security, Protection Mechanisms, and Advanced OS Case Studies

Scope of Module:
Focuses on OS security, protection models, and advanced industrial applications of operating systems.

Topics:
OS security principles, access control models (DAC, MAC, RBAC); Kernel hardening techniques (grsecurity, seccomp); Network security in OS: Firewall configurations (iptables, nftables); Secure Boot and Trusted Platform Module (TPM); Security vulnerabilities and mitigation strategies; Encryption, authentication, and auditing in OS; Introduction to SELinux, AppArmor; Forensics in OS environments (log analysis, intrusion detection); Case studies on OS security breaches and their prevention; Mini Project: Design a simple OS-level security module, process monitoring utility, or Intrusion Detection System (IDS) script.

Learning Outcome:*Students will be able to assess and implement security models and protection mechanisms in modern OS environments.*

Module	No of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Basic understanding of computer architecture and programming in C.

Pedagogy / Teaching Methodology:

- ✓ Hands-on lab exercises (Linux command-line, virtualization tools)
- ✓ Case studies and simulation exercises
- ✓ Mini-projects on OS utilities (schedulers, memory managers)
- ✓ Interactive discussions on real-world OS scenarios
- ✓ Tool walkthroughs (Docker, Kubernetes, VM monitors)

Certificate / Value Added Courses Recommended:

1. Operating Systems and You: Becoming a Power User – Coursera
2. Linux for Developers – edX

3. Introduction to Containers w/ Docker, Kubernetes & OpenShift – Coursera
4. System Programming and Linux – Udemy

Suggested Readings:

- Abraham Silberschatz, Operating System Concepts, Wiley
- Andrew S. Tanenbaum, Modern Operating Systems, Pearson
- William Stallings, Operating Systems: Internals and Design Principles, Pearson
- Industry whitepapers on virtualization, containerization, and OS security

Recommended Software/Tools:

1. Linux (Ubuntu/CentOS)
2. Docker, Kubernetes (Minikube)
3. VirtualBox / VMWare
4. Wireshark (for security analysis)
5. GDB for kernel-level debugging

Curriculum Development:

Sl.No	Identify and highlight in the module
1	Local, Regional, National and Global
2	Employability, Entrepreneurship and Skill Development
3	Value Added Courses (Module)
4	Integration of Digital Learning Tools
5	Crossing Cutting Edges (Ethics, Gender Equality, Sustainability and Environmental concerns)
6	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO-PO Mapping Table:

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	3	2	1	1	–	–	–	2
CO2	3	3	3	2	2	–	–	–	2
CO3	3	3	3	3	2	1	–	–	3
CO4	3	3	3	3	3	2	1	–	3
CO5	3	3	3	3	3	2	1	1	3
Articulation	3	3	2.8	2.4	2.2	1.75	1	1	2.6

Course Title	**Data Structures**
Course Code	**25VMT0C104**
Semester	**I**
Credits	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description

This course introduces Advanced Concepts of data structures, algorithms, and their practical use in computing applications. Students will learn how to implement, analyze, and apply various linear and non-linear data structures to build efficient and scalable software systems

Course Outcomes (Cos)

Upon successful completion of this course, learners will be equipped to:

CO	**Course Outcome**	**BTL**
CO1	Understand the fundamentals of advanced linear and data structures.	L1
CO2	Apply appropriate data structures for solving complex computational problems.	L3
CO3	Apply tree, graph techniques in software solutions.	L4

CO4	Implement hashing techniques in software solutions.	L3
CO5	Create efficient programs using modern tools and real-world case studies.	L5

Note: Implementation can be done in C/C++

Course Modules

No. of Hours: 24

Module 1: Linear Data Structures

Scope of Module
Covers implementation and usage of linear storage structures.

Topics:
Arrays, Stacks (array/linked), Queues (normal, circular), Dequeues, Linked list operations (SLL, DLL, CLL), Stack/Queue with linked lists, Applications in expression evaluation, Bracket matching, Recursion stack, Queue in scheduling, List-based simulation, Sparse matrices, Time/space complexity basics, Array vs list.

Learning Outcome*: Apply linear structures in solving algorithmic problems.*

No. of Hours: 24

Module 2: Trees and Binary Trees

Scope of Module
Focuses on hierarchical data structures with operations and applications of binary trees and advanced trees.

Topics:
Tree definitions and terminologies, Binary tree representation, Binary tree traversals (inorder, preorder, postorder), Binary search tree (BST), Insertion in BST, Deletion in BST, Balanced trees

(AVL trees), Height-balanced trees, Expression trees, Threaded binary trees, Heap trees, Min-heap and max-heap, Priority queue using heap, B-Trees, Applications of trees

Learning Outcome: *Students will be able to implement tree structures and use them for hierarchical data processing.*

No. of Hours: 24

Module 3: Graphs and Applications

Scope of Module:
Introduction to graphs and path-finding algorithms.

Topics:
Graph representation (adjacency list/matrix), BFS, DFS, Topological sorting, Dijkstra's algorithm, Bellman-Ford, Floyd-Warshall, Kruskal's algorithm, Prim's algorithm, Network flow, Graph coloring, Euler and Hamiltonian paths, Graph traversals for cycles, Applications in networks, Graph problem modeling.

Learning Outcome: *Analyze and solve graph-based problems using suitable algorithms.*

No. of Hours: 24

Module 4: Hashing and Searching Structures

Scope of Module:
Efficient search, hash, and map-based data access.

Topics:
Hash tables, Hash functions, Collision resolution (chaining, open addressing), Load factor, Linear probing, Double hashing, Bloom filters, Dictionaries, Sets, Unordered maps, Binary search trees vs hashing, Search time analysis, Skip lists, LRU Cache design, Memory-optimized hashing, Applications in OS/DB.

Learning Outcome: *Design and implement efficient search and mapping structures.*

No. of Hours: 24

Module 5: Case Studies and Tools

Scope of Module:
Tools and case implementations using advanced DS.

Topics:
STL containers in C++ (vector, list, map, set), Data structure libraries in C++, Visualization tools (VisuAlgo, Graphviz), IDEs (CLion, Eclipse), Real-world DS projects, Competitive programming platforms, Time-space profiler tools, LeetCode/Codeforces case studies, DS in AI/ML preprocessing, Game development DS, Cache designs, Real DBMS indexing, Distributed DS structures.

Learning Outcome: *Evaluate and apply real-world DS solutions using modern tools and case studies.*

Module	No of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Basic knowledge of programming and algorithms.

Pedagogy / Teaching Methodology:

- ✓ Interactive Lectures
- ✓ Algorithm Visualization
- ✓ Hands-on Programming Labs
- ✓ Assignments and Case Studies
- ✓ Mini Projects

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. NPTEL: Data Structures and Algorithms
2. Coursera: Data Structures by UC San Diego

Suggested Readings:

- Seymour Lipschutz, *Data Structures*, McGraw-Hill
- Ellis Horowitz, Sartaj Sahni, *Fundamentals of Data Structures*, Universities Press
- Mark Allen Weiss, *Data Structures and Algorithm Analysis in C*, Pearson Education
- Online resources and research papers related to data structures

Recommended Software/Tools (If applicable):

1. C/C++ Compiler
2. Visual Studio Code / Code::Blocks
3. Online IDEs (Repl.it, GeeksforGeeks IDE)
4. Visualgo.net for algorithm visualization

Curriculum Development:

Sl.No	Identify and highlight in the module
1	Local, Regional, National and Global
2	Employability, Entrepreneurship and Skill Development
3	Value Added Courses (Module)
4	Integration of Digital Learning Tools
5	Crossing Cutting Edges (Ethics, Gender Equality, Sustainability and Environmental concerns)
6	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	3	2	2	1	–	2	1
CO2	3	3	3	2	2	1	1	2	2
CO3	2	2	3	3	2	1	1	1	2
CO4	3	2	3	2	2	1	1	1	2
CO5	2	3	2	3	2	2	–	2	1
Articulation	2.6	2.4	2.8	2.4	2	1.2	1	1.6	1.6

CENTRE FOR
DISTANCE AND
ONLINE EDUCATION

ONLINE

Corporate Office
91/2, Dr. A N Krishna Rao Road V V Puram,
Bengaluru - 560 004
P: +91 80 4343 1000

Centre for Distance & Online Education
319, 25th Main, 17th Cross,
J P Nagar 6th Phase
Bengaluru – 560 078
P: +91 80 4343 0100

Course Title:	**AI Skills for Future**
Course Code:	**26VMX0S105**
Semester:	**I**
Credits:	**2**
Course Type	**Skill Enhancement Course**
Learning Hours	60
Live Sessions	**12 hours**

Course Outcomes:

CO1: Understand Generative AI and LLMs in depth
CO2: Create text, image, and voice content using AI
CO3: Automate daily tasks and business workflows
CO4: Build AI agents that act independently
CO5: Design and deploy simple AI apps without coding

Course Modules

Module 1: Generative AI Prompt Engineering for Real Workflows

What is Generative AI and how it's transforming industries, Structure of an effective prompt: *Role → Task → Context → Output,* Common prompting mistakes and how to fix them, How large language models (LLMs) like ChatGPT, Claude, and Google AI Studio work, advanced prompting (few-shot, iterative, style-based), Using your own files in ChatGPT and Claude (RAG basics), Research and summarization using AI

Learning Outcome: Students will be able to understand how Generative AI works and master the art of prompting.

Module 2: Creative AI and Voice & Audio Creation

Understanding AI image generation, creating posters, campaign assets, and product visuals, building brand identity using Krea.ai, Generating realistic voiceovers using ElevenLabs, Script writing and tone control with ChatGPT & Claude, Integrating voice into short videos or podcasts ***Learning Outcome:*** Students will be able to use AI to generate visuals, voiceovers, and creative storytelling content.

Module 3: AI Automation & Productivity

Logic of triggers, actions, and workflows, setting up automation between apps (Gmail, Sheets, Drive), Connecting OpenAI to Make.com for intelligent automation, creating social media Management Workflow and research automation flows, automating daily reports and reminders, building personal productivity dashboards, Integrating Claude and ChatGPT into existing workflows

Learning Outcome: Automate daily workflows and tasks using AI and no-code tools.

Module 4: Advanced AI Agents & Decision Flows

Introduction to n8n as a visual AI workflow builder, understanding nodes, triggers, and data passing, Using OpenAI node for smart responses and actions, designing multi-step reasoning workflows, Connecting Google Sheets, Slack, and other APIs, building autonomous agents that perform specific tasks

Learning Outcome: Students will be able to build AI agents that can think, decide, and act automatically.

Module 5: AI App Building & Deployment

Fundamentals of UI/UX for AI apps, structuring front-end and AI back-end flows, designing app logic using n8n and Make.com, integrating OpenAI and Claude APIs to your apps, handling user inputs and dynamic responses, using Google AI Studio to experiment with models, hosting and testing your AI app online, adding analytics and monitoring usage, final demo preparation and peer feedback

Learning Outcome: Students will be able to learn to design, connect, and deploy simple AI apps using no-code tools

Semester II

Common Courses

Course Title	Advanced Database Systems
Course Code	25VMT0C201
Semester	II
Credits	4
Course Type	Core Course
Learning Hours	120
Live Sessions	12 hours

Course Description:

This course delves into advanced concepts and techniques in database systems, focusing on design, implementation, optimization, and emerging trends. Topics include distributed databases, NoSQL systems, data warehousing, transaction management, and big data technologies. The course emphasizes practical applications through case studies and hands-on projects.

Course Outcomes (Cos)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Understand advanced database models and architectures.	L1
CO2	Design and optimize relational and NoSQL databases.	L3
CO3	Implement transaction management and concurrency control.	L4
CO4	Evaluate distributed database systems and their challenges.	L5
CO5	Apply data warehousing and big data technologies.	L3

Course Modules

No. of Hours: 24

Module 1: Advanced Database Models and Architectures

Scope of Module:
Covers advanced database models, including object-oriented, object-relational, and NoSQL databases, and their architectures.

Topics:
Review of relational database systems; Object-oriented databases; Object-relational databases; NoSQL database overview; Document databases (MongoDB); Key-value databases (Redis); Column-family databases (Cassandra); Graph databases (Neo4j); NewSQL systems; Polyglot persistence; CAP theorem; BASE properties; Cloud-native database systems; Multi-model databases; Case study: MongoDB vs. PostgreSQL.

Learning Outcome: *Students will be able to compare and contrast different database models and select appropriate architectures for specific applications.*

No. of Hours: 24

Module 2: Database Design and Query Optimization

Scope of Module:
Focuses on advanced techniques for database design, indexing, query optimization, and performance tuning.

Topics:
Advanced normalization (1NF–5NF); Denormalization strategies; Physical vs. logical design Partitioning strategies; Indexing: B-tree, hash, bitmap; Covering indexes and index-only scans Query execution plans; Cost-based query optimization; Heuristic-based optimization; Materialized views; Query rewriting techniques; Adaptive query optimization; Statistics collection and use; SQL performance tuning tools; Hands-on: SQL tuning with EXPLAIN and ANALYZE.

Learning Outcome: *Students will be able to design efficient database schemas and optimize query performance.*

No. of Hours: 24

Module 3: Transaction Management and Concurrency Control

Scope of Module:
Explores transaction properties, concurrency control mechanisms, and recovery techniques in database systems.

Topics:
ACID properties; Transaction lifecycle; Serializability and schedules; Lock-based concurrency control; Deadlock detection and prevention; Timestamp ordering; Optimistic concurrency control; Two-phase locking; Multiversion Concurrency Control (MVCC); Isolation levels and anomalies; Serializable snapshot isolation; Write-ahead logging; Checkpointing; Recovery methods; Case study: PostgreSQL concurrency and recovery.

Learning Outcome: *Students will be able to implement and analyze transaction management and concurrency control mechanisms.*

No. of Hours: 24

Module 4: Distributed Database Systems

Scope of Module:
Examines distributed database architectures, data fragmentation, replication, and consistency models.

Topics:
Distributed database architecture; Client-server vs. peer-to-peer; Horizontal and vertical fragmentation; Data allocation and replication; Distributed concurrency control; Distributed transactions; Two-phase commit protocol; Distributed query processing Query decomposition and localization; CAP theorem in distributed databases; Eventual consistency models; Conflict resolution strategies; Fault tolerance and availability; Case study: Google Spanner; Hands-on: Setting up a distributed Cassandra cluster.

Learning Outcome: *Students will be able to evaluate distributed database systems and address their challenges.*

No. of Hours: 24

Module 5: Data Warehousing and Big Data Technologies

Scope of Module:
Introduces data warehousing concepts, ETL processes, OLAP, and big data technologies.

Topics:

Data warehousing concepts; Star schema vs. snowflake schema; Fact and dimension tables ETL process design; Data cleaning and transformation; Data staging and loading; OLAP operations and cubes; Hadoop ecosystem (HDFS, MapReduce); Apache Hive and Pig Apache Spark fundamentals; NoSQL for Big Data: HBase and Cassandra; Data lakes and lakehouse architecture; Real-time data pipelines (Kafka + Spark Streaming); Columnar storage formats (Parquet, ORC); Mini-project: Building a Spark-based data pipeline.

Learning Outcome: *Students will be able to apply data warehousing and big data technologies for analytical processing.*

Module	**No of Pre-Recordings**
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Basic knowledge of database management systems (DBMS) and SQL.

Pedagogy / Teaching Methodology:

- ✓ Conceptual and problem-based learning
- ✓ Hands-on labs and case studies
- ✓ Group projects and presentations
- ✓ Quizzes and assignments using database tools

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. Advanced Database Systems by NPTEL: https://nptel.ac.in/courses/106105171
2. Database Systems by Stanford Online: https://online.stanford.edu/courses/soe-ydatabases-database-systems

Suggested Readings:

- Raghu Ramakrishnan and Johannes Gehrke, *Database Management Systems*, McGraw-Hill, 2003.
- Hector Garcia-Molina, Jeffrey D. Ullman, and Jennifer Widom, *Database Systems: The Complete Book*, Pearson, 2008.
- Martin Kleppmann, *Designing Data-Intensive Applications*, O'Reilly, 2017.

Recommended Software/Tools:

1. PostgreSQL, MongoDB, Cassandra
2. Hadoop, Spark
3. SQL Developer, Tableau

Curriculum Development:

Sl.No	Identify and highlight in the module
1	Local, Regional, National and Global
2	Employability, Entrepreneurship and Skill Development
3	Value Added Courses (Module)
4	Integration of Digital Learning Tools
5	Crossing Cutting Edges (Ethics, Gender Equality, Sustainability and Environmental concerns)
6	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	–	2	1	–	1	1
CO2	2	3	2	1	1	–	–	2	1
CO3	3	2	2	1	1	1	1	1	2
CO4	2	3	3	1	2	2	–	1	2
CO5	3	2	1	2	2	1	1	2	2
Articulation	2.6	2.4	1.8	1.25	1.6	1.25	1	1.4	1.6

Course Title	**Java Programming**
Course Code	**25VMT0C202**
Semester	**II**
Credits	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides an in-depth understanding of Java programming, covering object-oriented principles, core APIs, multithreading, networking, and modern frameworks. Emphasis is placed on practical implementation through projects, case studies, and hands-on labs to build robust applications.

Course Outcomes (Cos)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Understand object-oriented principles in Java.	L1
CO2	Develop applications using Java core APIs and collections.	L3
CO3	Implement multithreading and exception handling.	L4
CO4	Design GUI applications and network-based solutions.	L5
CO5	Apply Java frameworks (Spring Boot) for enterprise applications.	L3

Course Modules

No. of Hours: 24

Module 1: Object-Oriented Programming in Java

Scope of Module:
Covers fundamental OOP concepts, Java syntax, and class design.

Topics:
Introduction to Java and JVM; Data types, operators, and control structures; Class structure and object creation; Encapsulation and constructors; Inheritance and method overriding; Polymorphism and dynamic binding; Interfaces and abstract classes; Package creation and import; Access modifiers and scope; Static vs. instance members; Exception handling: try-catch-finally; Custom exceptions; Assertions and error handling; Object class and method overriding; Case study: Modular banking system design.

Learning Outcome: *Students will be able to apply OOP principles such as encapsulation, inheritance, and polymorphism to design structured and reusable Java applications.*

No. of Hours: 24

Module 2: Java Core APIs and Collections

Scope of Module:
Focuses on Java standard libraries and data structures.

Topics:
String manipulation and immutability; Regular expressions; Wrapper classes and autoboxing Generics in Java; ArrayList, LinkedList implementations; HashSet, TreeSet, LinkedHashSet usage; HashMap, TreeMap, LinkedHashMap differences; Iterators, ListIterators, and enhanced for-loop; Comparable vs. Comparator; Collections utility class; File handling with java.io and java.nio; Object serialization and deserialization; Properties and configuration files; Practical: Inventory management system; Debugging collections and memory leaks.

Learning Outcome: *Students will be able to leverage core Java libraries and the Collections Framework to build efficient, maintainable applications with complex data structures.*

No. of Hours: 24

Module 3: Multithreading and Concurrency

Scope of Module:
Explores concurrent programming in Java.

Topics:
Thread lifecycle and states; Thread creation via Thread class; Runnable interface and Lambda expressions; Thread prioritization and yielding; Synchronized methods and blocks; Intrinsic locks and synchronized collections; Volatile keyword and atomic variables; Deadlock scenarios and prevention strategies; wait(), notify(), and notifyAll(); Thread pooling and ExecutorService; Callable and Future interfaces; ConcurrentHashMap and thread-safe queues; ForkJoinPool basics; Case study: Multithreaded chat application; Concurrency debugging tools and techniques.

Learning Outcome: *Students will be able to implement multithreaded programs and apply Java concurrency tools to ensure safe, scalable execution of parallel tasks.*

No. of Hours: 24

Module 4: GUI and Networking

Scope of Module:
Introduces Java FX and network programming.

Topics:
Introduction to JavaFX and GUI architecture; SceneBuilder and JavaFX controls; Layout managers: BorderPane, VBox, HBox; Event-driven programming in Java; Creating forms and validation; Swing basics and component lifecycle; Action listeners and adapters; Socket programming: TCP; Datagram programming: UDP; HTTP communication using HttpClient Using REST APIs in Java; JDBC for database connectivity; CRUD operations with JDBC Mini-project: GUI Weather App with REST API; Threading in GUI and networking applications

Learning Outcome: *Students will be able to design and build user-friendly GUI applications integrated with network communication and backend services using Java.*

No. of Hours: 24

Module 5: Java Frameworks and Modern Tools

Scope of Module:
Covers Spring Boot and testing frameworks.

Topics:
Introduction to Spring Boot and its architecture; Dependency Injection (DI) in Spring; Creating RESTful APIs with Spring Boot; Spring annotations and application.properties; CRUD operations with Spring Data JPA; Connecting to MySQL/PostgreSQL; Validation with Spring MVC; Unit testing with JUnit 5; Mocking with Mockito; Postman for API testing; Building with Maven and Gradle; Logging with Logback and SLF4J; Creating JAR/WAR packages; Dockerizing a Spring Boot app; Capstone project: E-commerce backend.

Learning Outcome: *Students will be able to develop full-stack Java applications using Spring Boot and related modern tools, and deploy them in enterprise-ready environments.*

Module	No of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Basic programming knowledge (C/C++/Python).

Pedagogy / Teaching Methodology:

- ✓ Interactive lectures with live coding
- ✓ Hands-on labs and debugging sessions
- ✓ Agile-style group projects
- ✓ Code reviews and peer assessments

Certificate / Value Added Courses Recommended:

1. Java Programming by NPTEL: https://nptel.ac.in/courses/106105168
2. Java Fundamentals by Oracle Academy: https://academy.oracle.com

Suggested Readings:

- Herbert Schildt, *Java: The Complete Reference*, McGraw-Hill, 2021.
- Joshua Bloch, *Effective Java*, Addison-Wesley, 2017.
- Craig Walls, *Spring Boot in Action*, Manning, 2020.

Recommended Software/Tools:

1. JDK 17+, IntelliJ IDEA/Eclipse
2. Spring Tool Suite, Postman
3. Git, Maven

Curriculum Development:

Sl.No	Identify and highlight in the module
1	Local, Regional, National and Global
2	Employability, Entrepreneurship and Skill Development
3	Value Added Courses (Module)

4	Integration of Digital Learning Tools
5	Crossing Cutting Edges (Ethics, Gender Equality, Sustainability and Environmental concerns)
6	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	–	2	1	–	1	1
CO2	2	3	2	1	1	–	–	2	1
CO3	3	2	2	1	1	1	1	1	2
CO4	2	3	3	1	2	2	–	1	2
CO5	3	2	1	2	2	1	1	2	2
Articulation	2.6	2.4	1.8	1.25	1.6	1.25	1	1.4	1.6

Course Title	**Web Technologies**
Course Code	**25VMT0C203**
Semester	**II**
Credits	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides comprehensive knowledge of modern web development technologies, covering front-end design, back-end programming, databases, and web security. Students will learn to build responsive, interactive web applications using HTML5, CSS3, JavaScript, PHP/Node.js, and frameworks like React and Express. Practical implementation through projects and case studies is emphasized.

Course Outcomes (Cos)

Upon successful completion of this course, learners will be equipped to:

CO	**Course Outcome**	**BTL**
CO1	Understand client-side web technologies (HTML5, CSS3, JavaScript).	L1
CO2	Develop interactive front-end applications using frameworks.	L3
CO3	Implement server-side programming and database integration.	L4
CO4	Evaluate web security threats and countermeasures.	L5
CO5	Build full-stack web applications using modern tools.	L3

Course Modules

No. of Hours: 24

Module 1: Client-Side Web Technologies

Scope of Module:
Covers fundamental technologies for building web interfaces.

Topics:
HTML5 document structure and semantic tags; Forms and input validation; CSS3 selectors and specificity; Box model, flexbox, and grid layouts; Responsive design principles (media queries); Typography and UI consistency; JavaScript syntax and data types; Control structures and functions; ES6+ features: let/const, arrow functions, template literals; DOM tree and traversal; Event handling and delegation; Form validation using JavaScript; Browser console and developer tools; Animation with CSS and JavaScript; Project: Portfolio website with interactivity.

Learning Outcome: *Students will be able to create accessible, responsive, and well-structured web pages using HTML5, CSS3, and modern JavaScript.*

No. of Hours: 24

Module 2: Front-End Frameworks

Scope of Module:
Focuses on modern JavaScript frameworks for UI development**.**

Topics:
Introduction to React.js and component lifecycle; JSX syntax and virtual DOM; Functional components and props; State and event management; Conditional rendering and lists; Forms and controlled components; React Hooks (useState, useEffect); Component communication; React Router for navigation; Context API for global state; AJAX requests with Axios and Fetch API; Using UI libraries (Bootstrap, Material UI); Creating reusable components; Code splitting and lazy loading; Project: Product listing page for e-commerce.

Learning Outcome: *Students will be able to build scalable and maintainable front-end applications using modern JavaScript frameworks like React.*

No. of Hours: 24

Module 3: Server-Side Programming

Scope of Module:
Explores back-end development and database integration.

Topics:
Node.js basics and NPM; Creating a local web server; Express.js routing and middleware; Handling POST and GET requests; Template engines (EJS/Handlebars); Form data parsing and validation; MySQL and MongoDB CRUD operations; Connection pooling and error handling; RESTful API design principles; JWT and session-based authentication; Role-based access control; File uploads and file system module; Environment variables and configuration; Project: Blog system with authentication and database; Debugging Node.js applications.

Learning Outcome: *Students will be able to build robust server-side applications and RESTful APIs integrated with databases using Node.js and Express.*

No. of Hours: 24

Module 4: Web Security

Scope of Module:
Addresses security vulnerabilities and protection mechanisms.

Topics:
Understanding HTTP protocol and HTTPS; SSL certificates and secure headers; OWASP Top 10 vulnerabilities; SQL injection and input sanitization; Cross-Site Scripting (XSS) and mitigation; Cross-Site Request Forgery (CSRF); Password hashing with bcrypt; Authentication vs. Authorization; Content Security Policy (CSP); Secure cookie practices; Same-Origin Policy and CORS; Rate limiting and API throttling; Penetration testing tools (Burp Suite, OWASP ZAP); Security audit and logging; Project: Secure login and audit trail system

Learning Outcome: *Students will be able to identify, prevent, and mitigate common web application security threats using industry best practices.*

No. of Hours: 24

Module 5: Full-Stack Development

Scope of Module:
Integrates all layers of web application development.

Topics:
Overview of MERN/MEAN stacks; Connecting frontend with backend via APIs; Redux basics and state management; User authentication and protected routes; Role-based content rendering; API testing with Postman; Code versioning with Git and GitHub; Continuous deployment workflows; Hosting frontend (Netlify, Vercel); Hosting backend (Render, Heroku) Performance optimization tips; Unit testing with Jest and Mocha; Linting and code formatting (Prettier, ESLint); DevOps basics for deployment; Capstone Project: Social media dashboard with analytics.

Learning Outcome: *Students will be able to develop, test, and deploy full-stack web applications using modern development tools and deployment platforms.*

Module	No of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Basic programming knowledge (HTML/CSS recommended).

Pedagogy / Teaching Methodology:

✓ Interactive coding demonstrations

✓ Pair programming exercises

✓ Project-based learning

✓ Security hackathons

Certificate / Value Added Courses Recommended:

1. Web Development by MDN: https://developer.mozilla.org
2. Full-Stack Open: https://fullstackopen.com

Suggested Readings:

- Jon Duckett, *HTML & CSS: Design and Build Websites*, Wiley, 2011
- Eric Freeman, *Head First JavaScript Programming*, O'Reilly, 2014
- David Flanagan, *JavaScript: The Definitive Guide*, O'Reilly, 2020

Recommended Software/Tools:

1. VS Code, Chrome DevTools
2. Postman, MongoDB Compass
3. React DevTools, Git

Curriculum Development:

Sl.No	**Identify and highlight in the module**
1	Local, Regional, National and Global
2	Employability, Entrepreneurship and Skill Development
3	Value Added Courses (Module)
4	Integration of Digital Learning Tools

5	Crossing Cutting Edges (Ethics, Gender Equality, Sustainability and Environmental concerns)
6	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	–	2	1	–	1	1
CO2	2	3	2	1	1	–	–	2	1
CO3	3	2	2	1	1	1	1	1	2
CO4	2	3	3	1	2	2	–	1	2
CO5	3	2	1	2	2	1	1	2	2
Articulation	2.6	2.4	1.8	1.25	1.6	1.25	1	1.4	1.6

Course Title	**Cloud Computing Foundations**

Course Code	**25VMT0C204**
Semester	**II**
Credits	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course introduces fundamental concepts of cloud computing, including service models, deployment models, virtualization, and cloud architecture. Students will gain hands-on experience with major cloud platforms (AWS/Azure/GCP) and learn to deploy, manage, and secure cloud-based solutions. The course emphasizes practical applications through labs and real-world case studies.

Course Outcomes (Cos)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Understand cloud computing concepts and service models.	L1
CO2	Deploy and manage virtualized infrastructure.	L3
CO3	Analyze cloud storage and networking solutions.	L4
CO4	Evaluate cloud security and compliance frameworks.	L5
CO5	Develop cloud-native applications using PaaS.	L3

Course Modules

No. of Hours: 24

Module 1: Cloud Computing Fundamentals

Scope of Module:
Covers basic concepts, characteristics, and service models of cloud computing.

Topics:
Definition and evolution of cloud computing; Essential characteristics of cloud (NIST model); Service models: IaaS, PaaS, SaaS; Deployment models: Public, Private, Hybrid, Community; Benefits and challenges of cloud adoption; Cloud computing use cases; Cloud vs. traditional IT infrastructure; Cloud service providers overview (AWS, Azure, GCP); Cloud service level agreements (SLAs); Multitenancy and resource pooling; Elasticity and scalability; Cloud computing cost models and pricing; Case study: Cloud migration strategy; Green computing and cloud efficiency; Cloud-first strategies in digital transformation.

Learning Outcome: *Students will be able to differentiate between cloud service and deployment models, and evaluate when and how to migrate to cloud environments.*

No. of Hours: 24

Module 2: Virtualization and Cloud Infrastructure

Scope of Module:
Focuses on virtualization technologies and cloud infrastructure management.

Topics:
Virtualization concepts and benefits; Types of virtualization: Server, Storage, Network; Hypervisors: Type 1 (bare-metal) and Type 2 (hosted); Virtual machine creation and lifecycle; Introduction to containers and Docker; Container orchestration basics (Kubernetes overview); Infrastructure as a Service (IaaS); AWS EC2, Azure VMs, GCP Compute Engine; Cloud autoscaling and elasticity; Elastic Load Balancing (ELB); Monitoring and cloud resource usage; Infrastructure as Code: Terraform and CloudFormation; Hands-on: Launching a VM on AWS/GCP; Cloud cost optimization techniques; Best practices for resource provisioning.

Learning Outcome: *Students will be able to configure and manage virtualized infrastructure on cloud platforms using IaaS and container technologies.*

No. of Hours: 24

Module 3: Cloud Storage and Networking

Scope of Module:
Explores storage solutions and network architectures in cloud environments.

Topics:
Cloud storage types: Object, Block, File; Storage services: Amazon S3, Azure Blob, Google Cloud Storage; Data redundancy and availability zones; Content Delivery Networks (CDNs) Edge computing and latency reduction; Data lifecycle and storage classes; Virtual Private Cloud (VPC) architecture; Subnets, gateways, and NAT; Cloud firewalls and security groups Cloud DNS and networking services; VPNs and hybrid cloud connectivity ;Database as a Service (DBaaS): RDS, Cosmos DB, Cloud SQL; Data backup and disaster recovery; Case study: Designing a high availability architecture; Storage and network performance benchmarking.

Learning Outcome: *Students will be able to design and configure secure, high-performance storage and networking solutions within cloud environments.*

No. of Hours: 24

Module 4: Cloud Security and Compliance

Scope of Module:
Addresses security challenges and compliance requirements in cloud computing.

Topics:
Cloud security principles and models; Shared responsibility model; Identity and Access Management (IAM); Role-based access control (RBAC); Data encryption in transit and at rest; Key management services; Security groups vs. network ACLs; Firewall policies and VPN security; Security incident detection and response; Cloud compliance frameworks (GDPR, HIPAA, ISO 27001); Security posture assessments; Penetration testing and vulnerability scanning; Logging and auditing best practices; Case study: Securing a multi-tenant cloud app. Mini-project: Implementing IAM and encryption in AWS.

Learning Outcome: *Students will be able to implement cloud security measures and ensure compliance with data protection regulations and security frameworks.*

No. of Hours: 24

Module 5: Cloud-Native Application Development

Scope of Module:
Covers development of applications optimized for cloud environments.

Topics:
Introduction to cloud-native principles; Twelve-factor app methodology; Serverless architecture overview; AWS Lambda and Azure Functions; Event-driven programming in the cloud; Microservices architecture and benefits; REST API design and testing (Postman); Containers vs. serverless vs. VMs; CI/CD pipelines: GitHub Actions, AWS CodePipeline; Monitoring and logging (CloudWatch, Stackdriver); Application scaling and performance tuning; API Gateway integration and throttling; DevOps tools for cloud (Jenkins, Terraform) Case study: Microservices deployment; Capstone Project: Build and deploy a serverless application.

Learning Outcome: *Students will be able to develop, deploy, and monitor scalable, cloud-native applications using modern cloud services and DevOps tools. pplications using modern cloud services and DevOps tools.*

Module	No of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Basic knowledge of operating systems and networking concepts.

Pedagogy / Teaching Methodology:

- ✓ Hands-on labs with cloud platforms
- ✓ Case studies of enterprise cloud migrations
- ✓ Security audit simulations
- ✓ Group projects with real-world scenarios

Certificate / Value Added Courses Recommended:

1. AWS Cloud Practitioner: https://aws.amazon.com/certification/
2. Microsoft Azure Fundamentals: https://learn.microsoft.com/en-us/certifications/

Suggested Readings:

- Dan Marinescu, *Cloud Computing: Theory and Practice*, 3rd Ed., Morgan Kaufmann, 2022
- Thomas Erl, *Cloud Computing: Concepts, Technology & Architecture*, Prentice Hall, 2021
- Official documentation from AWS, Azure, and GCP

Recommended Software/Tools:

1. AWS Free Tier, Azure Portal
2. Docker, Kubernetes (basic)
3. Terraform, CloudFormation

Curriculum Development:

Sl.No	Identify and highlight in the module

1	Local, Regional, National and Global
2	Employability, Entrepreneurship and Skill Development
3	Value Added Courses (Module)
4	Integration of Digital Learning Tools
5	Crossing Cutting Edges (Ethics, Gender Equality, Sustainability and Environmental concerns)
6	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	–	2	1	–	1	1
CO2	2	3	2	1	1	–	–	2	1
CO3	3	2	2	1	1	1	1	1	2
CO4	2	3	3	1	2	2	–	1	2
CO5	3	2	1	2	2	1	1	2	2
Articulation	2.6	2.4	1.8	1.25	1.6	1.25	1	1.4	1.6

Course Title	**Environmental, Social, and Governance (ESG) in IT**
Course Code	**25VMT0C205**

Semester	**II**
Credits	**4**
Course Type	**Ability Enhancement Compulsory Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course explores the integration of Environmental, Social, and Governance (ESG) principles into information technology. It covers sustainable IT practices, ethical AI, data privacy, and corporate governance in tech organizations. Students will learn to assess ESG risks, implement green computing solutions, and align IT strategies with global sustainability goals through case studies and projects.

Course Outcomes (Cos)

CO	**Course Outcome**	**BT L**
CO1	Explain ESG frameworks relevant to IT	L1
CO2	Apply green computing techniques	L3
CO3	Analyze social impacts of AI and big data	L4
CO4	Evaluate governance models for IT ethics	L5
CO5	Design sustainable IT policies for organizations	L6

Upon successful completion of this course, learners will be equipped to:

Course Modules

No. of Hours: 24

Module 1: ESG Fundamentals in IT

Scope of Module: Introduces core ESG concepts and their application in technology sectors.

Topics:
Understanding ESG in the context of IT; Global ESG frameworks: UN SDGs, GRI, SASB; Environmental impacts of digital infrastructure; Carbon footprint of data centers and cloud computing; E-waste management and recycling strategies; IT lifecycle sustainability assessments; Role of ICT in sustainable development goals; Green certifications for IT (Energy Star, EPEAT); Sustainability KPIs in IT projects; ESG compliance for tech startups AI and sustainability intersections; Digital ethics and sustainable design; Sustainable supply chain management in IT; Case study: Google's carbon-neutral cloud initiatives; Future of ESG in emerging technologies.

Learning Outcome: *Students will articulate how ESG principles apply to IT infrastructure, service models, and sustainability reporting.*

No. of Hours: 24

Module 2: Green Computing

Scope of Module:
Focuses on sustainable IT practices and resource optimization.

Topics:
Green hardware lifecycle management; Energy-efficient processor design; Power-aware scheduling algorithms; Green software design and refactoring; Virtualization for sustainability; Resource optimization in cloud environments; Carbon-aware computing; Renewable energy integration in data centers; Measuring IT carbon footprints; Green networking and IoT devices; Cooling technologies and data center PUE; Server consolidation strategies; Green code practices in DevOps; Life Cycle Assessment (LCA) tools for IT systems; Lab: Calculate emissions from virtual machines.

Learning Outcome: *Students will be able to apply green computing principles to minimize environmental impacts of IT operations and software systems.*

No. of Hours: 24

Module 3: Social Responsibility in Tech

Scope of Module:
Examines ethical implications of AI, data, and digital inclusion.

Topics:
Digital equity and inclusion challenges; Algorithmic bias and mitigation techniques; AI transparency and explainability; Ethical data collection and consent; GDPR, CCPA, and global data privacy laws; Bias in training data for machine learning; Accessible design principles in software; Digital literacy and education programs; Tech for underserved communities. Social media and societal impact; AI in healthcare and social justice; Tech activism and digital rights; Workplace inclusivity in tech companies; Case study: Microsoft's AI for Accessibility Lab: Bias detection in data classification models.

Learning Outcome: *Students will analyze social implications of emerging technologies and advocate for inclusive, ethical digital innovation.*

No. of Hours: 24

Module 4: IT Governance and Ethics

Scope of Module:
Covers governance structures for ethical IT decision-making.

Topics:
Foundations of IT corporate governance; ESG reporting frameworks and tools; Whistleblower policies and ethics channels; IT ethics boards and oversight; Compliance versus ethical leadership; IT audit trails and data governance; Transparency and accountability practices; Ethical AI governance and regulation; Cybersecurity ethics in surveillance systems; Boardroom simulation: resolving tech dilemmas; Balancing profit and public interest in IT firms; Decision-making under ethical uncertainty; ISO standards for IT governance; ESG ratings and public disclosures; Case study: Governance failure in a tech giant.

Learning Outcome: *Students will evaluate and develop governance frameworks that support ethical decision-making and stakeholder alignment in IT environments.*

No. of Hours: 24

Module 5: Sustainable IT Strategy

Scope of Module:
Integrates ESG principles into organizational IT planning.

Topics:
Integrating ESG into IT strategy roadmaps; Stakeholder mapping and engagement models; Circular economy in IT hardware; Supply chain resilience and transparency; Cloud sustainability strategy; Responsible tech innovation lifecycle; Designing ESG-aligned KPIs for IT; Tech sustainability scorecards and benchmarking; Policy modeling for net-zero IT operations; Carbon offsetting through tech initiatives; Funding and investing in green IT Organizational change for ESG transformation; Capstone: ESG audit of a mock IT enterprise Scenario planning for climate-resilient IT; Report writing: ESG strategy white paper.

Learning Outcome: *Students will design sustainable IT policies and long-term strategic plans incorporating ESG values for enterprise impact.*

Module	**No of Pre-Recordings**
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Basic knowledge of IT systems and corporate social responsibility.

Pedagogy:

- ✓ Guest lectures from ESG tech leaders
- ✓ Carbon accounting workshops
- ✓ Ethics hackathons
- ✓ ESG report analysis

Certifications Recommended:

1. Certified Green IT Professional (GITP)
2. SASB's ESG Fundamentals

Readings:

- "Green IT for Sustainable Business Practice" by Mark O'Neill
- "AI Ethics" by Mark Coeckelbergh

Tools:

1. AWS Customer Carbon Footprint Tool
2. ESG reporting software (Sustainalytics)

Curriculum Development:

Sl.No	Identify and highlight in the module
1	Local, Regional, National and Global
2	Employability, Entrepreneurship and Skill Development
3	Value Added Courses (Module)
4	Integration of Digital Learning Tools
5	Crossing Cutting Edges (Ethics, Gender Equality, Sustainability and Environmental concerns)

6	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	2	1	–	3	1	–	2	1	2
CO2	3	2	1	2	3	1	–	2	1
CO3	1	3	2	1	2	3	1	1	3
CO4	2	1	3	2	1	2	3	–	2
CO5	3	2	2	3	2	1	2	2	3
Articulation	2.2	1.8	2	2.2	1.8	1.75	2	1.5	2.2

Semester III

Corporate Office
91/2, Dr. A N Krishna Rao Road V V Puram,
Bengaluru - 560 004
P: +91 80 4343 1000

Centre for Distance & Online Education
319, 25th Main, 17th Cross,
J P Nagar 6th Phase
Bengaluru – 560 078
P: +91 80 4343 0100

Course Title	Artificial Intelligence and Machine Learning
Course Code	251VMT0C301
Semester	III
Credits	4
Course Type	Core Course
Learning Hours	120
Live Sessions	12 hours

Course Description:

This course provides MCA students with an industry-relevant and application-driven exploration of Artificial Intelligence (AI) and Machine Learning (ML). The syllabus is designed to offer a progressive and logical learning path from AI fundamentals to advanced ML techniques and deployment strategies. The course also emphasizes ethical, explainable, and production-ready AI systems through a mix of practical labs, case studies, and modern tools.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO No.	Course Outcome	BTL
CO1	Understand the evolution and fundamentals of Artificial Intelligence and Machine Learning.	L1
CO2	Apply advanced ML algorithms for supervised, unsupervised, and semi-supervised learning tasks.	L3
CO3	Analyze deep learning models and reinforcement learning strategies.	L4

CO4	Evaluate models for bias, interpretability, and effectiveness across varied datasets.	L5
CO5	Design and deploy scalable, explainable, and secure AI/ML applications.	L3

Course Modules

No. of Hours: 24

Module 1: Foundations of AI and ML

Scope of Module:
Establishes core concepts of AI and ML, setting the stage for in-depth exploration of models, learning paradigms, and industry practices.

Topics:
AI and ML definitions and key differences, history and evolution of AI, AI vs ML vs DL, symbolic vs sub-symbolic AI, types of ML (supervised, unsupervised, reinforcement), ML lifecycle and pipeline stages, CRISP-DM methodology, introduction to datasets and data types, common data challenges, bias and fairness in ML, ML vs traditional programming, use cases in various industries, recent trends in AI (Generative AI, LLMs), AI ethics overview.

Learning Outcome: *Students will be able to articulate core AI/ML definitions, classify learning types, and describe modern trends and challenges.*

No. of Hours: 24

Module 2: Advanced Supervised Learning and Ensemble Models

Scope of Module:
Covers core and advanced supervised learning techniques including ensemble strategies and hyperparameter tuning.

Topics:
Linear and logistic regression with regularization (Ridge, Lasso), decision trees, random forest, support vector machines (SVM), KNN classification, Naive Bayes, model evaluation metrics (accuracy, precision, recall, F1, ROC-AUC), model selection and validation strategies (train/test split, cross-validation), bias-variance tradeoff, grid search and random search, ensemble learning (bagging, boosting), introduction to XGBoost, LightGBM, and CatBoost, multi-output and multiclass classification, managing imbalanced datasets (SMOTE, undersampling), SHAP for tree models.

Learning Outcome: *Students will be able to apply and evaluate supervised learning and ensemble methods to solve classification and regression tasks.*

No. of Hours: 24

Module 3: Unsupervised Learning, Dimensionality Reduction, and Anomaly Detection

Scope of Module:
Explores a wide range of unsupervised techniques and introduces preprocessing and anomaly detection as critical skills.

Topics:
Clustering algorithms: k-means, DBSCAN, hierarchical, Gaussian Mixture Models (GMM), elbow method and silhouette score, association rule mining (Apriori, FP-Growth), dimensionality reduction techniques: PCA, t-SNE, UMAP, feature selection vs extraction, feature importance, anomaly detection using Isolation Forest, LOF, one-class SVM, preprocessing: encoding, scaling, normalization, dealing with missing values, pipeline creation using scikit-learn, unsupervised model evaluation strategies.

Learning Outcome: -*Students will be able to perform unsupervised learning, reduce dimensionality for visualization, and detect data anomalies effectively.*

No. of Hours: 24

Module 4: Deep Learning and Reinforcement Learning

Scope of Module:
Introduces deep neural networks and reinforcement learning paradigms for intelligent decision making.

Topics:
Artificial Neural Networks (ANN), forward and backward propagation, activation functions (ReLU, sigmoid, tanh), loss functions and optimizers, CNNs for image processing, RNNs, LSTMs for sequential data, autoencoders and applications, GANs and generative models, overview of BERT and Transformer models, introduction to reinforcement learning, Q-learning and policy gradients, environment-reward-agent loop, OpenAI Gym, applications in recommendation, robotics, gaming.

Learning Outcome: -*Students will be able to build and analyze neural networks and apply RL for decision-making scenarios.*

No. of Hours: 24

Module 5: AI/ML Model Deployment, MLOps, and Explainability

Scope of Module:
Focuses on end-to-end AI/ML model deployment, production readiness, monitoring, and responsible AI practices.

Topics:
Deployment using Flask, FastAPI, Streamlit, Docker containerization, introduction to Kubernetes for ML, CI/CD pipelines in ML, MLflow and model versioning, explainable AI tools (SHAP, LIME, ELI5), fairness audits and bias detection tools, logging and monitoring models in production, introduction to cloud ML platforms (AWS SageMaker, GCP Vertex AI), securing ML models and data, case studies from industry, best practices in production ML systems.

Learning Outcome: -*Students will be able to deploy, explain, and monitor AI/ML systems using modern DevOps practices and frameworks.*

Module	**No of Pre-Recordings**
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Strong understanding of Python programming, statistics, and data structures.

Pedagogy / Teaching Methodology:

- Project-based learning with datasets from real-world domains
- Lab-based exercises using Jupyter and Google Colab
- Mini case studies and research paper reviews
- Live demonstrations and group activities
- Tool walkthroughs and guided deployment

Certificate / Value Added Courses Recommended:

1. Machine Learning Engineering for Production (MLOps) – Coursera
2. Deep Learning Specialization – Coursera
3. AI for Everyone – Coursera

4. Google ML Crash Course – https://developers.google.com/machine-learning/crash-course

Suggested Readings:

- Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, O'Reilly, 2022
- Ian Goodfellow, Deep Learning, MIT Press, 2016
- Sutton & Barto, Reinforcement Learning: An Introduction, MIT Press, 2018
- Kelleher, D. J., Data Science Ethics, MIT Press
- Relevant research from IEEE, ACM, and arXiv.org

Recommended Software/Tools:

1. Python, Scikit-learn, TensorFlow, Keras, PyTorch
2. Google Colab / JupyterLab, MLflow, Optuna
3. Docker, FastAPI, Streamlit, AWS SageMaker, GCP Vertex AI
4. SHAP, LIME, OpenAI Gym

Curriculum Development:

Sl.No	Identify and highlight in the module
1	Local, Regional, National and Global
2	Employability, Entrepreneurship and Skill Development
3	Value Added Courses (Module)
4	Integration of Digital Learning Tools
5	Crossing Cutting Edges (Ethics, Gender Equality, Sustainability and Environmental concerns)
6	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO-PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	1	2	1	1	1	2
CO2	3	3	3	2	2	1	1	2	2
CO3	2	3	3	3	2	1	2	1	2
CO4	2	3	2	3	2	3	2	2	2
CO5	3	3	3	2	3	2	2	3	3
Articulation	2.6	2.8	2.4	2.2	2.2	1.6	1.6	1.8	2.2

Course Title	Python Programming
Course Code	251VMT0C302
Semester	III
Credits	4
Course Type	Core Course
Learning Hours	120
Live Sessions	12 hours

Course Description:

This course provides MCA students with foundational to advanced skills in Python programming, blending basic concepts with real-world applications. It encompasses control structures, functions, OOP, file operations, exception handling, unit testing, web development, API integration, and data analysis. Special emphasis is placed on writing efficient, modular, and scalable Python code with exposure to industry-grade tools and frameworks.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO No.	Course Outcome	BTL
CO1	Understand Python syntax, control flow, data structures, and functions.	L1
CO2	Apply OOP principles and modular coding in real-world applications.	L3
CO3	Analyze problems and implement file, error handling, and unit testing.	L4

CO4	Evaluate web-based interactions, APIs, and database connectivity using Python.	L5
CO5	Develop interactive data-driven applications using Pandas, NumPy, and visualization libraries.	L3

No. of Hours: 24

Module 1: Python Basics, Syntax, and Control Flow

Scope of Module:
Introduces core elements of Python such as syntax, variables, data types, operators, and control flow.

Topics:
Installing Python and IDE setup, writing and executing Python scripts, identifiers, keywords, variables, constants, data types, type conversion, operators and precedence, conditional statements (if, elif, else), iterative control (for, while), break and continue, pass statement, nested loops, input/output handling, formatted string literals (f-strings), Python coding style *(PEP8).*

Learning Outcome: *Students will be able to write structured Python scripts using control flow constructs and input/output operations.*

No. of Hours: 24

Module 2: Functions, Collections, Iterators, and Modules

Scope of Module:
Covers function design, advanced collection operations, and module creation for reusable code.

Topics:
Defining and invoking functions, argument types (*args, **kwargs), scope and lifetime, recursion, anonymous functions (lambda), Python built-in functions, comprehension (list, dict,

set), advanced list methods, tuples and unpacking, sets and operations, dictionaries and operations, iterators and generators, creating custom modules, importing built-in modules (math, sys, os, datetime, calendar, random), packaging with pip and venv.

Learning Outcome: *Students will be able to modularize programs using custom and built-in Python functions and libraries.*

No. of Hours: 24

Module 3: Object-Oriented Programming, File I/O, and Exception Handling

Scope of Module:
Explores OOP principles, file operations, and robust exception handling techniques.

Topics:
Classes and objects, class and instance variables, methods, constructor and destructor, inheritance and types, method overriding, polymorphism, encapsulation, use of __str__, _repr_, _len_, working with text/binary/CSV files, reading and writing files, context manager and file closing, exception types and hierarchy, try-except-finally, raising and handling custom exceptions, logging module basics, assertions for debugging.

Learning Outcome: *Students will be able to design OOP-based applications with efficient file handling and error management.*

No. of Hours: 24

Module 4: Testing, Automation, and Database Connectivity

Scope of Module:
Provides practices in automated testing, scripting for task automation, and integrating Python with databases.

Topics:

Writing test cases using unittest and pytest, test discovery and test suites, test-driven development, automation with shutil and subprocess, working with file system using os and pathlib, scheduling tasks, introduction to SQLite and PostgreSQL, executing queries from Python using sqlite3 and psycopg2, performing CRUD operations, using context managers with databases, parameterized queries to prevent SQL injection.

Learning Outcome: *Students will be able to automate tasks and integrate databases securely in Python programs.*

No. of Hours: 24

Module 5: Web Development, APIs, and Data Science Libraries

Scope of Module:
Bridges Python to web and data science fields using APIs, Flask, and data analysis tools.

Topics:
Flask framework structure and routing, GET and POST methods, HTML rendering with Jinja2 templates, form data processing, REST API consumption using requests, OpenWeatherMap API integration example, Pandas: dataframes, merging, filtering, groupby, NumPy: array creation and operations, Matplotlib/Seaborn for plotting, exporting data to Excel/CSV, introduction to Streamlit for interactive dashboards, mini project combining Flask and Pandas.

Learning Outcome: *Students will be able to develop and deploy data-centric and interactive Python web applications.*

Module	No of Pre-Recordings
1	10
2	12
3	10
4	12

5	**10**

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Basic logic building, familiarity with any programming language is helpful.

Pedagogy / Teaching Methodology:

- Concept-based demonstrations using Jupyter and IDEs
- Mini assignments and use-case-driven labs
- Group activities and debugging challenges
- Integration of GitHub for version control practices
- End-of-module review quizzes and projects

Certificate / Value Added Courses Recommended:

1. Python for Everybody – Coursera
2. Python OOP and Flask Projects – Udemy
3. Automate the Boring Stuff with Python – Online Book
4. Introduction to Data Science in Python – Coursera

Suggested Readings:

- Mark Lutz, Learning Python, O'Reilly
- Eric Matthes, Python Crash Course, No Starch Press
- Luciano Ramalho, Fluent Python, O'Reilly
- Python Documentation – https://docs.python.org

Recommended Software/Tools:

1. Python 3.x, PyCharm, VS Code
2. JupyterLab, Google Colab
3. Flask, Pandas, NumPy, Matplotlib, Streamlit
4. GitHub, SQLite, PostgreSQL

Curriculum Development:

Sl.No	Identify and highlight in the module
1	Local, Regional, National and Global
2	Employability, Entrepreneurship and Skill Development
3	Value Added Courses (Module)
4	Integration of Digital Learning Tools
5	Crossing Cutting Edges (Ethics, Gender Equality, Sustainability and Environmental concerns)
6	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO-PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	2	1	2	1	1	1	2
CO2	3	3	3	2	3	1	2	2	3
CO3	3	3	3	3	2	1	2	2	3
CO4	2	3	3	3	3	2	2	2	3
CO5	3	3	3	2	3	2	2	3	3
Articulation	2.8	2.8	2.8	2.2	2.6	1.4	1.8	2	2.8

Course Title	**Computer Vision**
Course Code	**251VMT4S303**
Semester	**III**
Credits	**4**
Course Type	**Discipline Specific Elective**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course equips MCA students with foundational to advanced knowledge in computer vision. Students will explore concepts from image formation and processing to deep learning-based vision tasks. Emphasis is placed on practical applications using OpenCV, CNNs, object detection models, and deployment strategies. Through hands-on projects, students will build skills to solve real-world problems in healthcare, automotive, retail, and surveillance sectors.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO No.	Course Outcome	BTL
CO 1	Understand the fundamental concepts of image formation and image processing.	L1
CO 2	Apply computer vision techniques for image enhancement and feature extraction.	L3
CO 3	Analyze classical and deep learning methods for object detection and recognition.	L4

CO 4	Evaluate the performance of vision models across various datasets and tasks.	L5
CO 5	Design and deploy computer vision systems using OpenCV and deep learning frameworks.	L3

No. of Hours: 24

Module 1: Introduction to Computer Vision and Image Formation

Scope of Module:
Provides a foundation in image generation, representations, and the visual perception pipeline.

Topics:
Human vision vs computer vision, applications and challenges, image formation and sensing, camera models and calibration, color models (RGB, HSV, LAB), image formats and storage, pixel operations, intensity transformation, geometric transformations, histogram analysis, image thresholding, contrast enhancement, perspective transformation., Gamma correction, Bitwise operations on images, Alpha blending and image overlays

Learning Outcome: *Students will be able to explain the basic principles of image formation and perform preprocessing operations.*

No. of Hours: 24

Module 2: Image Processing and Feature Extraction Techniques

Scope of Module:
Focuses on classical techniques used to enhance and extract key features from images.

Topics:
Noise filtering techniques (Gaussian, Median), morphological operations (erosion, dilation, opening, closing), edge detection (Sobel, Prewitt, Canny), corner detection (Harris, Shi-Tomasi), blob detection (DoG, LoG), image pyramids, contour detection and hierarchy, image segmentation techniques, thresholding (Otsu), region growing, watershed algorithm., Hough

transform for line and circle detection, Connected components analysis, Texture analysis using GLCM

Learning Outcome: *Students will be able to process raw images and extract useful features for analysis.*

No. of Hours: 24

Module 3: Object Detection and Recognition using Classical Approaches

Scope of Module:
Explores traditional methods for object detection, matching, and tracking.

Topics:
Template matching, HOG descriptors, Haar cascades, object tracking with Kalman filters and MeanShift, feature descriptors (SIFT, SURF, ORB), keypoint matching with FLANN and BFMatcher, RANSAC algorithm for homography, motion detection and background subtraction, blob tracking, multi-object tracking basics., Color-based object tracking using HSV, Foreground segmentation using GrabCut, Kalman vs Particle filters

Learning Outcome: *Students will be able to implement classical vision pipelines for detection, matching, and tracking tasks.*

No. of Hours: 24

Module 4: Deep Learning for Computer Vision

Scope of Module:
Introduces neural networks and CNN-based solutions for object detection and classification.

Topics:
Overview of neural networks, convolutional layers, pooling and activation, CNN architectures (LeNet, AlexNet, VGG, ResNet), image classification using Keras and TensorFlow, transfer learning and fine-tuning, object detection using YOLO and SSD, face recognition models, semantic and instance segmentation (U-Net, Mask R-CNN), visualizing filters and activations., EfficientNet overview, Data augmentation techniques for training, Model evaluation: confusion matrix, IoU, mAP

Learning Outcome: *Students will be able to build and fine-tune CNN-based models for image classification and detection tasks.*

No. of Hours: 24

Module 5: Advanced Applications and Vision System Deployment

Scope of Module:
Covers real-time vision systems, deployment, and applied case studies.

Topics:
Working with OpenCV and MediaPipe, pose estimation and hand tracking, OCR using Tesseract and EasyOCR, barcode and QR code detection, scene text recognition, real-time face detection with OpenCV, integrating vision models in web and mobile apps, deploying models with TensorFlow Lite and ONNX, ethical issues and bias in computer vision, mini project using a custom dataset., Edge TPU and Jetson Nano deployment, Real-time video analytics using OpenCV, Streamlit-based vision dashboards

Learning Outcome: *Students will be able to develop and deploy end-to-end computer vision applications across platforms.*

Module	No of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Basic knowledge of Python programming and linear algebra.

Pedagogy / Teaching Methodology:

✓ Hands-on OpenCV and TensorFlow coding labs

✓ Live coding demonstrations and visual explanations

✓ Mini projects with datasets (e.g., face, traffic, license plate)

✓ Research paper reviews and use-case discussions

✓ Assessment via weekly lab-based challenges

Certificate / Value Added Courses Recommended:

1. Computer Vision Fundamentals – Coursera
2. Deep Learning with OpenCV and TensorFlow – Udemy
3. Real-Time Computer Vision with Python – YouTube/Blogs
4. Intro to TensorFlow for Computer Vision – Udacity

Suggested Readings:

- Richard Szeliski, Computer Vision: Algorithms and Applications
- Adrian Rosebrock, Practical Python and OpenCV
- Simon J.D. Prince, Computer Vision: Models, Learning, and Inference
- OpenCV Documentation – https://docs.opencv.org

Recommended Software/Tools:

1. Python 3.x, OpenCV, MediaPipe
2. TensorFlow, Keras, PyTorch
3. LabelImg, Roboflow, EasyOCR
4. JupyterLab, Google Colab, GitHub

Curriculum Development:

Sl.No	Identify and highlight in the module
1	Local, Regional, National and Global
2	Employability, Entrepreneurship and Skill Development
3	Value Added Courses (Module)
4	Integration of Digital Learning Tools
5	Crossing Cutting Edges (Ethics, Gender Equality, Sustainability and Environmental concerns)
6	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO-PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	2	1	1	1	1	1	2
CO2	3	3	3	2	2	1	1	2	3
CO3	3	3	3	3	2	1	2	2	3
CO4	2	3	3	3	3	2	2	2	3
CO5	3	3	3	2	3	2	2	3	3
Articulation	2.8	2.8	2.8	2.2	2.2	1.4	1.6	2	2.8

Semester IV

Course Title	**Natural Language Processing**
Course Code	**251VMT4S401**
Semester	**IV**
Credits	**4**
Course Type	**Discipline Specific Elective**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides MCA students with an in-depth understanding of Natural Language Processing (NLP), covering classical and modern deep learning techniques for working with text data. Students will learn about text preprocessing, syntactic and semantic analysis, information extraction, and advanced transformer-based models like BERT. Emphasis is placed on practical skills using Python NLP libraries to build applications such as sentiment analysis, question answering, and text summarization.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO No.	**Course Outcome**	**BTL**
CO 1	Understand the fundamentals of natural language structure and linguistic processing.	L1
CO 2	Apply classical and statistical NLP techniques for information retrieval and extraction.	L3

CO 3	Analyze machine learning and deep learning models for NLP tasks.	L4
CO 4	Evaluate language models and transformer-based architectures for performance.	L5
CO 5	Design and deploy NLP applications for real-world scenarios using Python libraries.	L3

No. of Hours: 24

Module 1: Introduction to NLP and Text Processing

Scope of Module:
Lays the foundation of language structure and text preprocessing required for NLP tasks.

Topics:
Components of NLP, linguistic concepts: morphology, syntax, semantics, and pragmatics, text encoding (ASCII, Unicode), text preprocessing (tokenization, stemming, lemmatization, stop word removal), regular expressions, part-of-speech tagging, n-grams and language models, vocabulary building, frequency distributions, introduction to NLTK and spaCy., Understanding corpus and lexicons, Differences between stemming and lemmatization, POS tagging using NLTK and spaCy, Language modeling: count-based and probabilistic models, Creating a custom tokenizer and stop word list

Learning Outcome: *Students will be able to preprocess text and apply linguistic analysis for downstream NLP tasks.*

No. of Hours: 24

Module 2: Text Classification and Information Retrieval

Scope of Module:
Explores document classification, sentiment analysis, and classical retrieval systems.

Topics:
Text classification techniques, bag-of-words and TF-IDF vectorization, cosine similarity, Naive Bayes and logistic regression for text classification, sentiment analysis pipeline, spam detection, introduction to information retrieval, inverted index, precision, recall, F1-score, confusion matrix, ranking and retrieval models (BM25), using Scikit-learn and Scikit-multilearn., TF-IDF vectorizer implementation, Comparing Naive Bayes and logistic regression classifiers, Using confusion matrix to assess model performance, Building a simple search engine using inverted index, Evaluation metrics for retrieval systems (MAP, MRR)

Learning Outcome: *Students will be able to classify and retrieve text documents using statistical NLP approaches.*

No. of Hours: 24

Module 3: Named Entity Recognition and Syntactic Parsing

Scope of Module:
Covers information extraction techniques and parsing of sentence structures.

Topics:
Named Entity Recognition (NER) using rule-based and statistical methods, spaCy and Hugging Face for NER, dependency parsing, constituency parsing, shallow vs deep parsing, chunking, co-reference resolution, semantic role labeling, introduction to knowledge graphs and relationship extraction, case study on news/medical corpora., NER pipelines using spaCy and Hugging Face, Treebanks and parse trees visualization, Coreference tools like NeuralCoref, Case study: Relationship extraction from news data, Dependency graphs and parsing trees interpretation

Learning Outcome: *Students will be able to extract structured information from unstructured text and analyze sentence syntax.*

No. of Hours: 24

Module 4: Word Embeddings and Deep Learning for NLP

Scope of Module:
Focuses on neural network-based embeddings and architectures used for NLP tasks.

Topics:
Word2Vec (CBOW and Skip-gram), GloVe embeddings, FastText, evaluating embeddings (cosine similarity, analogy tasks), RNNs and LSTMs for sequence modeling, attention mechanism, transformer architecture overview, using pretrained models for text classification and generation, BERT architecture and fine-tuning, comparison of GPT, BERT, RoBERTa, DistilBERT., Visualizing embeddings with PCA and t-SNE, Implementing a sentiment classifier using RNN, Fine-tuning BERT with Hugging Face Transformers, Use cases: text classification vs generation, Training and evaluating LSTM-based sequence models

Learning Outcome: *Students will be able to implement and fine-tune deep learning models using embeddings and transformers.*

No. of Hours: 24

Module 5: NLP Applications and Deployment

Scope of Module:
Applies NLP in real-world applications and demonstrates how to build, evaluate, and deploy NLP systems.

Topics:
Text summarization (extractive and abstractive), topic modeling (LDA), question answering systems, chatbot design with Rasa or Dialogflow, speech-to-text (overview), translation systems, introduction to Hugging Face Transformers, deploying NLP models using FastAPI/Streamlit, model evaluation and error analysis, ethical considerations in NLP, mini project development and demonstration., Comparing extractive vs abstractive summarization, Topic modeling implementation using Gensim LDA, Creating a simple chatbot using Rasa

framework, Building a REST API with FastAPI for text classification, Packaging and sharing NLP models via Hugging Face Hub

Learning Outcome: *Students will be able to build and deploy real-world NLP applications using open-source tools and libraries.*

Module	No of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites:

➢ Basic understanding of Python programming, linear algebra, and probability.

Pedagogy / Teaching Methodology:

✓ Hands-on programming using Python NLP libraries

✓ Lab sessions with NLTK, spaCy, and Hugging Face

✓ Project-based learning for NLP applications

✓ Mini case studies using domain-specific corpora

✓ Interactive demos and cloud-based model deployment

Certificate / Value Added Courses Recommended:

1. Natural Language Processing with Classification and Vector Spaces – Coursera
2. Practical Deep Learning for NLP – FastAI
3. Transformers for NLP – Hugging Face Course
4. Speech and Language Processing – edX

Suggested Readings:

- Daniel Jurafsky & James H. Martin, Speech and Language Processing
- Steven Bird, Ewan Klein, and Edward Loper, Natural Language Processing with Python
- Yoav Goldberg, Neural Network Methods in NLP
- Online tutorials from Hugging Face, spaCy, and NLTK docs

Recommended Software/Tools:

1. Python 3.x, NLTK, spaCy, Scikit-learn
2. TensorFlow, Keras, PyTorch, Hugging Face Transformers
3. Streamlit, FastAPI, Google Colab, GitHub

Curriculum Development:

Sl.No	Identify and highlight in the module
1	Local, Regional, National and Global
2	Employability, Entrepreneurship and Skill Development
3	Value Added Courses (Module)
4	Integration of Digital Learning Tools

5	Crossing Cutting Edges (Ethics, Gender Equality, Sustainability and Environmental concerns)
6	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO-PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	2	1	1	1	1	1	2
CO2	3	3	3	2	2	1	1	2	3
CO3	3	3	3	3	2	1	2	2	3
CO4	2	3	3	3	3	2	2	2	3
CO5	3	3	3	2	3	2	2	3	3
Articulation	2.8	2.8	2.8	2.2	2.2	1.4	1.6	2	2.8

Course Title	**Predictive Analytics using Machine Learning**
Course Code	**251VMT4S402**
Semester	**IV**
Credits	**4**
Course Type	**Discipline Specific Elective**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course introduces MCA students to the core principles and practical techniques of predictive analytics using machine learning. Students will gain hands-on experience building predictive models for classification and regression problems using real-world datasets. The course covers the complete analytics pipeline including data preprocessing, feature selection, model selection, evaluation, and deployment with an emphasis on modern tools and explainability.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO No.	**Course Outcome**	**BTL**
CO1	Understand the fundamentals of predictive analytics and machine learning concepts.	L1
CO2	Apply preprocessing and feature engineering techniques to prepare data for modeling.	L3
CO3	Analyze and compare predictive models using appropriate performance metrics.	L4

CO4	Evaluate model generalizability and prevent overfitting using validation techniques.	L5
CO5	Design and deploy machine learning pipelines for real-world predictive tasks.	L3

No. of Hours: 24

Module 1: Introduction to Predictive Analytics and ML Concepts

Scope of Module:
Establishes the foundation of predictive modeling and supervised machine learning.

Topics:
Definition of predictive analytics, types of analytics (descriptive, diagnostic, predictive, prescriptive), business use cases, machine learning overview, types of ML (supervised, unsupervised, reinforcement), overview of classification and regression problems, CRISP-DM methodology, ML project lifecycle, problem framing and success metrics., Case studies in predictive analytics (loan default, churn prediction), Overview of supervised vs unsupervised models, ML lifecycle stages: goal setting to deployment, Ethical considerations in predictive modeling

Learning Outcome: *Students will be able to define predictive analytics concepts and identify suitable ML techniques for different problem types.*

No. of Hours: 24

Module 2: Data Preprocessing and Feature Engineering

Scope of Module:
Covers data cleaning, transformation, and feature engineering for robust predictive modeling.

Topics:
Handling missing values, outlier detection and imputation, data normalization and standardization, encoding categorical variables (one-hot, label), binning and discretization,

feature scaling, data splitting strategies (train/test/validation), feature extraction and transformation, correlation analysis, feature selection techniques (filter, wrapper, embedded), using Scikit-learn pipeline., Skewed data handling and SMOTE, Feature importance ranking using tree-based models, Polynomial feature creation, Target encoding and hashing for high cardinality features

Learning Outcome: *Students will be able to preprocess and transform datasets for input into machine learning models.*

No. of Hours: 24

Module 3: Predictive Modeling Techniques

Scope of Module:
Explores core classification and regression algorithms used in predictive analytics.

Topics:
Linear regression, logistic regression, decision trees, k-nearest neighbors, Naive Bayes, support vector machines, ensemble methods (bagging, random forest, boosting with XGBoost), model assumptions and interpretability, hyperparameter tuning using GridSearchCV and RandomizedSearchCV, model complexity and bias-variance tradeoff., Model stacking and blending, ElasticNet regularization for regression, ROC and precision-recall curves for classifier comparison, Case study: predicting customer churn

Learning Outcome: Students will be able to build and evaluate machine learning models for regression and classification tasks.

No. of Hours: 24

Module 4: Model Evaluation and Validation Strategies

Scope of Module:
Focuses on evaluating and validating predictive models using standard metrics and techniques.

Topics:
Performance metrics for classification: accuracy, precision, recall, F1-score, ROC-AUC, confusion matrix; for regression: MAE, MSE, RMSE, R^2; k-fold cross-validation, stratified sampling, nested cross-validation, overfitting and underfitting, learning curves, model selection and ensembling strategies, SHAP and LIME for explainability., Learning curves and validation curves, Hyperparameter search space design, Permutation feature importance, SHAP summary plots and decision plots

Learning Outcome: *Students will be able to evaluate models using appropriate metrics and enhance generalizability through validation.*

No. of Hours: 24

Module 5: Deployment and Applications of Predictive Analytics

Scope of Module:
Demonstrates how to build end-to-end predictive systems using modern tools and techniques.

Topics:
Creating machine learning pipelines with Scikit-learn, model persistence using joblib and pickle, building interactive apps using Streamlit, deploying models via Flask/FastAPI, integrating predictions into business workflows, introduction to MLOps tools (MLflow), case studies in finance, healthcare, and retail, capstone project implementation., Streamlit components for visualizing predictions, Model deployment on Hugging Face Spaces, Introduction to BentoML for packaging models, MLflow experiment tracking and model registry

Learning Outcome: *Students will be able to design, build, and deploy real-time predictive analytics solutions.*

Module	No of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Basic knowledge of Python, statistics, and data handling with Pandas.

Pedagogy / Teaching Methodology:

✓ Lab-driven sessions using Python and Scikit-learn

✓ Mini projects and real-world case studies

✓ Model comparison exercises and error analysis

✓ Live demonstrations using Jupyter and Streamlit

✓ Concept reinforcement with short quizzes and coding challenges

Certificate / Value Added Courses Recommended:

1. Applied Machine Learning in Python – Coursera
2. Machine Learning with Python – IBM/Coursera
3. Practical Predictive Analytics – Udemy
4. Introduction to MLOps – DataCamp

Suggested Readings:

- Tariq Rashid, Make Your Own Neural Network
- Andriy Burkov, The Hundred-Page Machine Learning Book
- Aurélien Géron, Hands-On Machine Learning with Scikit-Learn and TensorFlow
- Articles from Towards Data Science, KDnuggets, and arXiv

Recommended Software/Tools:

1. Python, Scikit-learn, Pandas, NumPy, Matplotlib, Seaborn
2. JupyterLab, Streamlit, Flask, FastAPI
3. joblib, pickle, MLflow
4. Google Colab, GitHub

Curriculum Development:

Sl.No	Identify and highlight in the module
1	Local, Regional, National and Global
2	Employability, Entrepreneurship and Skill Development
3	Value Added Courses (Module)
4	Integration of Digital Learning Tools
5	Crossing Cutting Edges (Ethics, Gender Equality, Sustainability and Environmental concerns)
6	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO-PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	2	1	2	1	1	1	2
CO2	3	3	3	2	2	1	1	2	3
CO3	3	3	3	3	2	1	2	2	3
CO4	2	3	2	3	2	2	2	2	3
CO5	3	3	3	2	3	2	2	3	3
Articulation	2.8	2.8	2.6	2.2	2.2	1.4	1.6	2	2.8

Course Title	**Deep Learning**
Course Code	**251VMT4S403**
Semester	**IV**
Credits	**4**
Course Type	**Discipline Specific Elective**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course introduces MCA students to the foundational and advanced principles of deep learning. It covers neural network architectures, training techniques, regularization, optimization, and modern advances such as CNNs, RNNs, GANs, and transformer-based models. Through hands-on implementation using TensorFlow and PyTorch, students will learn to solve real-world problems in computer vision, natural language processing, and sequence modeling.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO No.	**Course Outcome**	**BTL**
CO1	Understand deep learning fundamentals and neural network architectures.	L1
CO2	Apply training, optimization, and regularization techniques to neural networks.	L3

CO3	Analyze CNN and RNN models for image and sequence-based data.	L4
CO4	Evaluate advanced architectures such as GANs and Transformers.	L5
CO5	Design, implement, and deploy deep learning models using open-source tools.	L3

No. of Hours: 24

Module 1: Foundations of Neural Networks

Scope of Module:
Introduces basic components of artificial neurons, forward and backward propagation, and activation functions.

Topics:
Biological neuron vs artificial neuron, perceptron model, activation functions (sigmoid, tanh, ReLU, Leaky ReLU), loss functions (MSE, cross-entropy), forward and backward propagation, weight initialization, gradient descent variants, vanishing and exploding gradient problems, batch normalization, dropout., Hebbian learning, Learning rate scheduling strategies, Use of activation functions in different layers

Learning Outcome: *Students will be able to explain the structure of neural networks and implement basic training workflows.*

No. of Hours: 24

Module 2: Deep Feedforward and Optimization Techniques

Scope of Module:
Covers multilayer neural networks, optimizers, and hyperparameter tuning strategies.

Topics:
Multilayer perceptron, universal approximation theorem, learning rate strategies, optimization algorithms (SGD, Momentum, RMSProp, Adam), early stopping, weight decay, learning rate schedules, hyperparameter search (Grid, Random, Bayesian), visualization with TensorBoard, Keras tuning., Glorot and He initialization, Overfitting in deep nets, Weight clipping and gradient clipping

Learning Outcome: *Students will be able to train deep neural networks using proper optimizers and tuning methods.*

No. of Hours: 24

Module 3: Convolutional and Recurrent Neural Networks

Scope of Module:
Explores CNNs for spatial data and RNNs for sequence data.

Topics:
Convolution operation and filters, padding, stride, pooling layers, CNN architectures (LeNet, AlexNet, VGG, ResNet), RNNs, LSTM, GRU, challenges with RNN training (vanishing gradients), applications in text and time series, sequence-to-sequence models, encoder-decoder architecture., Depthwise separable convolutions, 1D and 3D convolutions, Bidirectional RNNs and Attention with RNNs

Learning Outcome: *Students will be able to implement CNNs and RNNs to handle image and sequence data.*

No. of Hours: 24

Module 4: Advanced Architectures and Techniques

Scope of Module:
Focuses on state-of-the-art models and deep generative methods.

Topics:
Introduction to generative modeling, autoencoders (vanilla, denoising, variational), GANs and

variants (DCGAN, Conditional GAN), attention mechanism, transformer architecture basics, BERT overview, positional encoding, pretraining and fine-tuning transformers, deep reinforcement learning overview., Self-attention and Multi-head attention, GPT model family overview, Energy-based models in deep learning

Learning Outcome: *Students will be able to explore advanced models and generative approaches for modern AI applications.*

No. of Hours: 24

Module 5: Applications, Toolkits, and Deployment of Deep Learning Models

Scope of Module:
Applies deep learning across various domains and introduces model deployment techniques.

Topics:
Transfer learning and pretrained models (VGG, Inception, EfficientNet), fine-tuning strategies, use cases in healthcare, finance, autonomous systems, deploying models with TensorFlow Serving and TorchServe, ONNX for interoperability, deploying with Streamlit, Flask, or FastAPI, performance monitoring, introduction to model compression and quantization., End-to-end deployment with Docker, Model interpretability in production, Benchmarking deep learning models for speed and accuracy

Learning Outcome: *Students will be able to build, optimize, and deploy deep learning models to solve real-world problems.*

Module	No of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Corporate Office
91/2, Dr. A N Krishna Rao Road V V Puram,
Bengaluru - 560 004
P: +91 80 4343 1000

Centre for Distance & Online Education
319, 25th Main, 17th Cross,
J P Nagar 6th Phase
Bengaluru – 560 078
P: +91 80 4343 0100

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Basic understanding of Python, linear algebra, and machine learning fundamentals.

Pedagogy / Teaching Methodology:

- ✓ Hands-on experiments using TensorFlow and PyTorch
- ✓ Concept-driven code walkthroughs and visualizations
- ✓ Assignments involving real datasets (MNIST, IMDB, etc.)
- ✓ Capstone mini-project using deep learning APIs
- ✓ Model debugging and performance tuning workshops

Certificate / Value Added Courses Recommended:

1. Deep Learning Specialization – Andrew Ng (Coursera)
2. Practical Deep Learning for Coders – FastAI
3. Deep Learning with PyTorch – Udacity
4. AI for Medicine – Coursera

Suggested Readings:

- Ian Goodfellow, Deep Learning, MIT Press
- Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow
- Francois Chollet, Deep Learning with Python
- Official TensorFlow and PyTorch documentation

Recommended Software/Tools:

1. Python 3.x, TensorFlow, Keras, PyTorch
2. JupyterLab, Google Colab, Streamlit

3. TensorBoard, ONNX, TorchServe, FastAPI

Curriculum Development:

Sl.No	Identify and highlight in the module
1	Local, Regional, National and Global
2	Employability, Entrepreneurship and Skill Development
3	Value Added Courses (Module)
4	Integration of Digital Learning Tools
5	Crossing Cutting Edges (Ethics, Gender Equality, Sustainability and Environmental concerns)
6	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO-PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	2	1	1	1	1	1	2
CO2	3	3	3	2	2	1	1	2	3
CO3	3	3	3	3	2	1	2	2	3
CO4	2	3	3	3	3	2	2	2	3
CO5	3	3	3	2	3	2	2	3	3
Articulation	2.8	2.8	2.8	2.2	2.2	1.4	1.6	2	2.8

Course Title	Big Data Hadoop
Course Code	251VMT4S404
Semester	IV
Credits	4
Course Type	Discipline Specific Elective
Learning Hours	120
Live Sessions	12 hours

Course Description:

This course introduces MCA students to the architecture, tools, and techniques associated with Big Data and the Hadoop ecosystem. It covers data ingestion, distributed storage and processing using HDFS and MapReduce, and dives deep into advanced tools like Hive, Pig, HBase, and Spark. Students will learn to build scalable big data processing applications using real-world datasets and industry tools.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO No.	Course Outcome	BTL
CO1	Understand the characteristics of Big Data and the architecture of Hadoop ecosystem.	L1
CO2	Apply HDFS and MapReduce programming for distributed data storage and processing.	L3
CO3	Analyze structured and unstructured data using Hadoop-based tools like Hive and Pig.	L4

CO4	Evaluate NoSQL data handling with HBase and in-memory processing with Spark.	L5
CO5	Design scalable big data workflows for practical business applications.	L3

No. of Hours: 24

Module 1: Introduction to Big Data and Hadoop Architecture

Scope of Module:
Covers the fundamentals of Big Data and explores the Hadoop framework and ecosystem.

Topics:
Definition and types of big data (structured, unstructured, semi-structured), characteristics of Big Data (5Vs), use cases across industries, overview of traditional vs Hadoop architecture, Hadoop core components, Hadoop Distributed File System (HDFS), Hadoop YARN, MapReduce framework overview, Hadoop ecosystem components (Hive, Pig, HBase, Spark)., Hadoop vs Spark comparison, Real-time data vs batch data processing, Data lake vs data warehouse

Learning Outcome: *Students will be able to explain big data concepts and describe the components and architecture of Hadoop.*

No. of Hours: 24

Module 2: HDFS and MapReduce Programming

Scope of Module:
Focuses on distributed storage using HDFS and computation using MapReduce.

Topics:
HDFS architecture: blocks, namenode, datanode, replication, read/write mechanism, MapReduce programming model, Mapper and Reducer classes, writing MapReduce programs in Java/Python, input/output formats, combiner and partitioner, counters and joins in

MapReduce, job configuration and execution flow, Hadoop streaming., Chaining MapReduce jobs, Optimization tips for MapReduce jobs, Running MapReduce using Python with Hadoop Streaming

Learning Outcome: *Students will be able to implement distributed data storage and batch processing using HDFS and MapReduce.*

No. of Hours: 24

Module 3: Data Analysis Using Hive and Pig

Scope of Module:
Covers high-level tools for structured and semi-structured data analysis in Hadoop.

Topics:
Hive architecture and components, HiveQL basics: databases, tables, partitions, DDL, DML, queries, user-defined functions (UDF), joins and subqueries, performance tuning in Hive, Pig architecture and execution model, Pig Latin scripting, data types, control structures, operators, joins and grouping, writing UDFs in Pig., Hive optimization techniques and indexing, Partitioning vs bucketing in Hive, Skew join handling in Pig

Learning Outcome: *Students will be able to analyze large-scale datasets using Hive and Pig in Hadoop ecosystem.*

No. of Hours: 24

Module 4: NoSQL with HBase and Introduction to Apache Spark

Scope of Module:
Introduces NoSQL data models with HBase and real-time processing using Apache Spark.

Topics:
Need for NoSQL databases, HBase architecture: regions, region server, HMaster, data model and CRUD operations, integrating HBase with Hive, Spark overview and architecture, RDDs vs DataFrames, Spark transformations and actions, Spark SQL, caching and persistence, running

Spark in standalone and cluster mode., Schema design in HBase, Bulk loading data into HBase, Spark MLlib introduction

Learning Outcome: *Students will be able to work with HBase for NoSQL storage and perform in-memory analytics with Spark.*

No. of Hours: 24

Module 5: Big Data Applications and Workflow Management

Scope of Module:
Applies the learned tools in solving industry use cases and managing data pipelines.

Topics:
Building big data pipelines with Sqoop and Flume, data ingestion from RDBMS and logs, introduction to Oozie workflow scheduler, deploying big data solutions on cloud (AWS EMR, GCP Dataproc), basics of performance tuning and fault tolerance, monitoring with Ambari, mini project with Hadoop, Hive, and Spark using real dataset., Airflow vs Oozie comparison, Building real-time dashboards using Spark Structured Streaming, Data lineage and governance with Apache Atlas

Learning Outcome: *Students will be able to build end-to-end data processing workflows using Hadoop ecosystem tools.*

Module	No of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Basic knowledge of Java/Python, databases, and Linux commands.

Pedagogy / Teaching Methodology:

- ✓ Hands-on labs using Hadoop on local/cluster setup
- ✓ Real-world dataset analysis with Hive and Pig
- ✓ Mini project with full Hadoop workflow
- ✓ Concept visualizations and execution tracing
- ✓ Industry examples and cloud-based demo sessions

Certificate / Value Added Courses Recommended:

1. Big Data Specialization – Coursera
2. Hadoop Platform and Application Framework – edX
3. Big Data Analysis with Hive and Pig – Udemy
4. Introduction to Apache Spark – DataCamp

Suggested Readings:

- Tom White, Hadoop: The Definitive Guide
- Vignesh Prajapati, Big Data Analytics with R and Hadoop
- Chuck Lam, Hadoop in Action
- Official Apache Documentation: https://hadoop.apache.org/

Recommended Software/Tools:

1. Hadoop, Hive, Pig, HBase, Apache Spark
2. Sqoop, Flume, Oozie, Ambari
3. Cloudera/Hortonworks Sandbox, AWS EMR, Google Dataproc

4. Linux Terminal, Eclipse or IntelliJ, Jupyter Notebook

Curriculum Development:

Sl.No	**Identify and highlight in the module**
1	Local, Regional, National and Global
2	Employability, Entrepreneurship and Skill Development
3	Value Added Courses (Module)
4	Integration of Digital Learning Tools
5	Crossing Cutting Edges (Ethics, Gender Equality, Sustainability and Environmental concerns)
6	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO-PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	2	1	2	1	1	1	2
CO2	3	3	3	2	2	1	1	2	2
CO3	3	3	3	3	2	1	2	2	3
CO4	2	3	3	3	3	2	2	2	3
CO5	3	3	3	2	3	2	2	3	3
Articulation	2.8	2.8	2.8	2.2	2.4	1.4	1.6	2	2.6

Course Title	**Strategic Leadership in IT**
Course Code	**251VMT0C405**
Semester	**IV**
Credits	**2**
Course Type	**Skill Enhancement Course**
Learning Hours	60
Live Sessions	**12 hours**

Course Description:

This course explores the critical role of leadership in driving strategic IT initiatives within organizations. It focuses on leadership theories, IT governance, digital transformation strategies, risk management, and innovation leadership. Practical case studies and simulations are integrated to build decision-making skills required for aligning IT with business strategies.

Course Outcomes (Cos)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Understand leadership theories and their application in IT strategy formulation.	L1
CO2	Evaluate and apply IT governance frameworks and practices.	L5
CO3	Design digital transformation strategies aligned with business goals.	L6
CO4	Identify and mitigate IT-related risks through strategic leadership.	L4
CO5	Demonstrate innovation and change management skills in IT leadership contexts.	L3

Course Modules

No. of Hours: 12

Module 1: Foundations of Strategic Leadership and IT Alignment

Scope of Module:
Covers the core leadership theories and IT-business alignment models.

Topics:
Defining leadership and management in the context of IT; Strategic leadership competencies; Leadership styles: transformational, transactional, situational; Vision building and strategic thinking in IT; Models of IT-business alignment (Henderson & Venkatraman Model); Organizational structures and their impact on IT leadership; Emotional intelligence in IT leadership; Ethical leadership in technology environments; Leading virtual and remote IT teams; Leadership maturity models; Case Study: IT leadership challenges in global enterprises.

Learning Outcome: *Students will be able to understand leadership theories and how to align IT strategies with broader business objectives.*

No. of Hours: 12

Module 2: IT Governance and Strategic Control

Scope of Module:
Focuses on frameworks and practices for effective IT governance.

Topics:
Introduction to IT governance and compliance; COBIT, ITIL, and ISO/IEC 38500 frameworks; Governance of Enterprise IT (GEIT) principles; Strategic decision rights and accountability in IT; Metrics and KPIs for IT governance; Managing outsourcing and vendor relationships; Sarbanes-Oxley Act and IT implications; Risk appetite and tolerance in IT governance; Blockchain's role in IT governance; Shadow IT and governance challenges; Audit management and IT policy development; Practical Exercise: Designing an IT governance framework for an SME.

Learning Outcome: *Students will be able to evaluate, select, and apply IT governance frameworks to ensure strategic control and value delivery.*

No. of Hours: 12

Module 3: Digital Transformation Leadership

Scope of Module:
Develops skills to lead digital transformation initiatives.

Topics:
Drivers of digital transformation in organizations; Role of CIOs and CTOs in transformation efforts; Technology adoption frameworks (e.g., Diffusion of Innovations); Agile and DevOps in digital leadership; Managing cultural change during transformation; Leadership in cloud, AI, and emerging technologies adoption; Case Study: Successful vs. failed digital transformation initiatives.

Learning Outcome: *Students will be able to design strategic digital transformation initiatives aligned with business innovation needs.*

No. of Hours: 12

Module 4: IT Risk Management and Strategic Decision-Making

Scope of Module:
Focuses on managing strategic risks in IT leadership roles.

Topics:
Understanding IT-related risks: security, operational, reputational; Risk management lifecycle in IT projects; Enterprise risk management frameworks (COSO ERM); Crisis leadership and contingency planning; Ethical considerations and cybersecurity leadership; Legal and regulatory compliance (e.g., GDPR, HIPAA); Threat modeling and strategic defense planning; Business continuity planning and IT disaster recovery leadership; Strategic handling of data breaches and incident response, Practical: IT risk assessment simulation

Learning Outcome: *Students will be able to analyze, manage, and mitigate IT risks strategically to ensure organizational resilience.*

No. of Hours: 12

Module 5: Innovation, Change Management, and Future of IT Leadership

Scope of Module:
Covers fostering innovation and managing technological change.

Topics:
Leading organizational innovation in IT; Change management models (Kotter's 8-Step, ADKAR); Talent development and IT leadership pipelines; Diversity, inclusion, and ethics in IT leadership; Future trends: Quantum Computing, Web3, and Leadership implications; Building a sustainable IT innovation culture; Intrapreneurship and fostering innovation within IT teams; Managing resistance to technological change; IT leadership in the age of Automation and RPA (Robotic Process Automation); Ethical considerations in AI and emerging tech leadership; Capstone: Strategic Leadership Plan for a tech startup.

Learning Outcome: *Students will be able to demonstrate strategic leadership skills in driving innovation, change management, and future-proofing IT organizations.*

Module	No of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Min no of pre recordings 10.

Prerequisites (If Any):

- Basic knowledge of IT management concepts and organizational behavior.

Pedagogy / Teaching Methodology:

- ✓ Interactive case studies and leadership simulations
- ✓ Group strategy projects and role plays
- ✓ Workshops on IT governance frameworks

✓ Industry expert talks and guest lectures

Certificate / Value Added Courses Recommended:

1. IT Leadership Certificate by edX (MIT Sloan)
2. COBIT 2019 Foundation Certification
3. Digital Transformation Strategy by Coursera (Boston University)

Suggested Readings:

- John P. Kotter, Leading Change, Harvard Business Review Press, 2012.
- Peter Weill & Jeanne W. Ross, IT Governance: How Top Performers Manage IT Decision Rights for Superior Results, Harvard Business Review Press, 2004.
- George Westerman, Leading Digital: Turning Technology into Business Transformation, Harvard Business Review Press, 2014.

Recommended Software/Tools:

1. MS Project, Trello, Jira (for project leadership)
2. ServiceNow, SolarWinds (for IT governance)
3. Risk assessment tools (RSA Archer, RiskLens)

Curriculum Development:

Sl.No	**Identify and highlight in the module**
1	Local, Regional, National and Global
2	Employability, Entrepreneurship and Skill Development
3	Value Added Courses (Module)
4	Integration of Digital Learning Tools

5	Crossing Cutting Edges (Ethics, Gender Equality, Sustainability and Environmental concerns)
6	Indian Knowledge System

CO PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	–	2	1	–	2	1
CO2	3	3	2	1	2	1	–	2	1
CO3	2	3	3	2	3	2	1	3	2
CO4	3	2	2	2	2	1	1	2	2
CO5	2	2	2	1	3	2	1	2	2
Articulation	2.6	2.4	2	1.5	2.4	1.4	1	2.2	1.6

Course Matrix

ELECTIVE: Cyber Security

Elective: Cyber Security

Semester-I

Course Code	Course Name	Course Category	CA	ESE	Credits
25VMT0C101	Mathematical Foundations for Computer Science	CC	30	70	4
25VMT0C102	Computing Concepts and Problem Solving using C	CC	30	70	4
25VMT0C103	Operating Systems	CC	30	70	4
25VMT0C104	Data Structures	CC	30	70	4
25VMT0L101	Problem Solving using C Lab	SEC	100	-	2
25VMT0L102	Data Structures Lab	SEC	100	-	2
26VMX0S105	AI Skills for Future	SEC	30	70	2
	Total No of Credits				**24**

Semester-II

Course code	Course Name	Course category	CA	ESE	Credits
25VMT0C201	Advanced Database Systems	CC	30	70	4
25VMT0C202	Java Programming	CC	30	70	4
25VMT0C203	Web Technologies	CC	30	70	4
25VMT0C204	Cloud Computing Foundations	CC	30	70	4
25VMT0L201	Advanced Database Systems Lab	SEC	100	-	2
25VMT0L202	Java Programming Lab	SEC	100	-	2
25VMT0C205	Environmental, Social, and Governance (ESG) in IT	AECC	30	70	4
	Total No of Credits				**24**

Semester-III

Course code	Course Name	Course category	CA	ESE	Credits
251VMT0C301	Artificial Intelligence and Machine Learning	CC	30	70	4
251VMT0C302	Python Programming	CC	30	70	4
251VMT1S303	Cyber Threat Intelligence	DSE	30	70	4
251VMT0L301	AI & Machine Learning Lab	SEC	100	-	2
251VMT0L302	Python Programming Lab	SEC	100	-	2
	Open Elective	OE	30	70	4
Total No of Credits					**20**

Semester-IV

Course code	Course Name	Course category	CA	ESE	Credits
251VMT1S401	Ethical Hacking	DSE	30	70	4
251VMT1S402	Defensive Cyber Security Technologies	DSE	30	70	4
251VMT1S403	Vulnerability Analysis	DSE	30	70	4
251VMT1S404	Penetration Testing	DSE	30	70	4
251VMT0C405	Strategic Leadership in IT	SEC	30	70	2
251VMT0P406	Project	Thesis			6
Total No of Credits					**24**

Legend	
CC	Core Course
DSE	Discipline Specific Elective

SEC	Skill Enhancement Courses
AECC	Ability Enhancement Compulsory Course
GE	Generic Elective

Semester I

Courses

Corporate Office
91/2, Dr. A N Krishna Rao Road V V Puram,
Bengaluru - 560 004
P: +91 80 4343 1000

Centre for Distance & Online Education
319, 25th Main, 17th Cross,
J P Nagar 6th Phase
Bengaluru – 560 078
P: +91 80 4343 0100

Course Title:	**Mathematical Foundations for Computer Science**
Course Code:	**25VMT0C101**
Semester:	**I**
Credits:	**4**
Course Type	Core Course
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course builds a strong mathematical base essential for various domains in computer science such as algorithms, cryptography, artificial intelligence, and data science. It covers discrete structures, logic, combinatory, graph theory, and number theory. Emphasis is laid on both theoretical understanding and application to computational problems.

Course Outcomes (COs)

Upon successful completion of this course, learners will be equipped to:

CO	**Course Outcome**	**BTL**
CO1	Comprehend the fundamentals of logic, sets, and relations used in computer science.	L3
CO2	Apply principles of combinatory and counting for solving algorithmic and real-world problems.	L3
CO3	Analyze and construct graphs, trees, and networks relevant to data structures and networking.	L4

CO4	Solve recurrence relations and use mathematical induction in algorithm design.	L3
CO5	Evaluate number theoretic and algebraic methods in cryptographic and coding applications.	L4

Course Module

No. of Hours:24

Module 1: Logic, Sets, and Relations
Scope of Module: Provides foundational concepts in mathematical logic, set theory, and relations essential for modeling and designing algorithms.
Topics: Propositions and logical operators, Truth tables, Tautology and contradiction, Predicates and quantifiers, Rules of inference computation, Logical equivalence, Methods of proof, Set theory basics, Venn diagrams, Set operations, Cartesian product of sets, Types of relations, Properties of relations, Equivalence and partial order relations, Closure of relations.
Learning Outcome: *Students will be able to construct logical arguments and model relationships using sets and relations.*

No. of Hours:24

Module 2: Combinatorics and Counting
Scope of Module: Focuses on combinatorial problem-solving techniques relevant to algorithm complexity and probability theory.

Topics: Basic counting principles, Addition and multiplication rules, Permutations and combinations, Pigeonhole principle, Principle of inclusion and exclusion, Binomial theorem, Pascal's triangle, Combinatorial identities, Multisets and combinations with repetition, Generating functions, Recurrence relations, Solving linear recurrence, Introduction to partitions, Derangements, Applications in probability.
Learning Outcome: *Students will be able to apply combinatorial reasoning to solve complex counting and probability problems.*

No. of Hours:24

Module 3: Graph Theory and Trees
Scope of Module: Introduces graph-based models to represent and solve problems in data structures, networks, and algorithms.
Topics: Basic concepts of graphs, Types of graphs, Graph representations, Degree of vertices, Paths and cycles, Connected components, Eulerian and Hamiltonian graphs, Planar graphs, Graph coloring, Matrix representation of graphs, Trees and their properties, Binary and n-ary trees, Tree traversal algorithms, Spanning trees, Minimum spanning tree algorithms (Kruskal, Prim)
Learning Outcome: *Students will be able to model and solve problems using graphs and trees in computer science.*

No. of Hours:24

Module 4: Recurrence Relations and Mathematical Induction
Scope of Module:

Covers recurrence relations and inductive reasoning techniques used in algorithm analysis and design.
Topics: Sequences and recurrence relations, Solving homogeneous linear recurrence, Non-homogeneous recurrence, Generating functions for recurrence, Divide-and-conquer relations, Recursion trees and Master's theorem, Mathematical induction, Strong induction, Well-ordering principle, Applications in algorithm correctness, Tower of Hanoi, Binary search, Fibonacci sequence, Ackermann's function.
Learning Outcome: *Students will be able to formulate and solve recurrence relations and validate algorithms using induction.*

No. of Hours:24

Module 5: Number Theory and Algebraic Structures
Scope of Module: Explores concepts in number theory and abstract algebra crucial for cryptography, hashing, and finite state systems.
Topics: Divisibility and Euclidean algorithm, Prime numbers and factorization, Congruence relations, Modular arithmetic, Fermat's little theorem, Euler's theorem, Chinese Remainder Theorem, Greatest common divisor (GCD), Least common multiple (LCM), Rings and fields basics, Group theory introduction, Subgroups and cyclic groups, Applications in cryptography, Hashing functions, Finite automata fundamentals
Learning Outcome: *Students will be able to apply number theoretic and algebraic techniques in areas like security and finite state systems.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites :

- Basic mathematics at undergraduate level, understanding of algebra and logic.

Pedagogy / Teaching Methodology:

- ✓ Conceptual and problem-based learning
- ✓ Interactive class exercises and proofs
- ✓ Real-world applications and modeling
- ✓ Assignments and quizzes using math software.

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. Discrete Mathematics by NPTEL: https://nptel.ac.in/courses/111106086
2. Mathematics for Computer Science by MIT Open Courseware: https://ocw.mit.edu

Suggested Readings:

- Kenneth H. Rosen, Discrete Mathematics and Its Applications, McGraw-Hill, 2017
- Tremblay and Manohar, Discrete Mathematical Structures, Tata McGraw-Hill, 2001
- C.L. Liu and D.P. Mohapatra, Elements of Discrete Mathematics, McGraw-Hill, 2010

- Online resources from IEEE Xplore, Springer Link for mathematical modeling and computation

Recommended Software/Tools (If applicable):

1. MATLAB/Octave
2. SageMath
3. Python (SymPy, NetworkX)
4. GeoGebra for visualization

Curriculum Development:

Sl.No	Identify and highlight in the module
1	Local, Regional, National and Global
2	Employability, Entrepreneurship and Skill Development
3	Value Added Courses (Module)
4	Integration of Digital Learning Tools
5	Crossing Cutting Edges (Ethics, Gender Equality, Sustainability and Environmental concerns)
6	Indian Knowledge System

CO/PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	-	2	1	-	1	1
CO2	2	3	2	1	1	-	-	2	1
CO3	3	2	2	1	1	1	1	1	2
CO4	2	3	3	1	2	2	-	1	2
CO5	3	2	1	2	2	1	1	2	2
Articulation	2.6	2.4	1.8	0.8	1.6	1.25	1	1.4	1.6

Note: The Articulation is indicative of how a particular course maps to a PO.

Articulation is calculated as:

$$= \frac{Sum\ of\ CO\ Mappiung\ Levels\ for\ a\ perticular\ PO}{No.of\ COs\ that\ mapped\ to\ that\ PO}$$

Example:1

$Articulation\ of\ PO1 = \frac{(3+2+3+2+3)}{5}$ = 2.6

Example: 2

$Articulation\ of\ PO7 = \frac{(1+1)}{2}$ = 1.0

Course Title:	**Computing concepts and problem solving in C**
Course Code:	**25VMT0C102**
Semester:	**I**
Credits:	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course covers Advanced concepts of computing concepts and problem solving in C and their applications in computer science. Students will develop problem-solving abilities and practical programming skills essential for core areas such as algorithms, systems programming, software development, and emerging technologies.

Course Outcomes (COs)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Comprehend the advanced Programming constructs	L3
CO2	Apply File Operations and handle errors effortlessly	L3
CO3	Integrate linux Programming with C Scripting	L4
CO4	Demonstrate C System Programming	L3
CO5	Evaluate real-world scenarios and Case Studies	L5

Course Modules

No. of Hours:24

Module 1: Fundamentals of Computing and C Programming
Scope of Module: Introduces basic computing concepts, algorithm design, and foundational elements of the C programming language
Topics: Basics of Computers: Architecture, memory, input/output units; Introduction to Algorithms and Flowcharts; C Programming Basics: Structure, syntax, variables, constants, data types; Operators: Arithmetic, relational, logical, bitwise, assignment, conditional; Compilation and Execution process.
Learning Outcome:*Students will be able to describe computing concepts, write simple C programs, and understand compilation processes.*

No. of Hours:24

Module 2: Control Structures, Functions, Arrays, Strings, and Modular Programming
Scope of Module: Explores problem-solving strategies using control flow, modular programming, and essential data structures.
Topics: Control Structures: if, if-else, nested if, switch-case; Loops: for, while, do-while, nested loops; Functions: Declaration, definition, call, recursion; Scope, Storage Classes (auto, static, extern, register); Modular Programming: Header files,

preprocessor directives (#define, #include); Arrays: 1D, 2D arrays, array manipulation; Strings: String functions (strlen, strcpy, strcat, strcmp), string manipulation.
Learning Outcome:*Students will be able to apply control structures, modularize code, and manipulate arrays and strings effectively.*

No. of Hours:24

Module 3: Pointers, Structures, Dynamic Memory Allocation, and File Handling
Scope of Module: Provides in-depth understanding of memory management, structured data, and file operations in C.
Topics: Introduction to Pointers: Pointer arithmetic, pointers and arrays; Pointers to Functions and Pointers to Structures; Structures and Unions: Declaration, definition, accessing members; Dynamic Memory Allocation: malloc, calloc, realloc, free; File Handling: File operations (read, write, append), file pointers; Case Study: Creating a file-based student management system.
Learning Outcome:*Students will be able to manipulate memory using pointers, manage structured data, and perform file operations.*

No. of Hours:24

Module 4: ***System-Level Programming and Real-Time Applications in C***

Scope of Module: *Focuses on industry-relevant applications, system programming concepts, and real-time project development.*
Topics: Command-line Arguments and Environment Variables; Error Handling (perror, strerror) and Debugging Techniques; Mini Projects: Text-based Games (Tic-Tac-Toe), Simple Shell Implementation, Real-Time Data Logger for IoT (file-based); Case Study: Simulating a Banking System.
Learning Outcome: *Students will be able to design and implement system-level programs and real-time applications using C.*

No. of Hours:24

Module 5: Advanced Industry Applications and Case Studies
Scope of Module: Encourages application of C programming skills in solving industry-specific problems and case studies.
Topics: Writing basic command-line tools; Simple network programs (client-server); Debugging and optimizing small C programs; Working with file compression and encryption tools; Implementing simple database file storage; Case Study: Building a basic calculator or file management tool; Mini Project: Simple automation program, data logger, or system monitor.
Learning Outcome*: Students will be able to apply C programming skills to solve advanced, industry-relevant problems through case studies and projects.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Basic knowledge of computer fundamentals.

Pedagogy / Teaching Methodology:

- ✓ Lectures, Hands-on Lab Sessions, Problem-solving Tasks, Case-based Assignments, Group Projects

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. NPTEL, Coursera, edX, MIT OCW — based on relevance to topics.

Suggested Readings:

- Standard textbooks, research articles, online tutorials and eBooks related to the subject.

Recommended Software/Tools (If applicable):

1. GCC Compiler, Visual Studio Code, Linux Terminal, C IDEs

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO/PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	3	2	2	2	1	-	1	2
CO2	3	2	2	3	2	1	-	1	1
CO3	2	3	2	2	-	1	1	1	1
CO4	3	2	3	2	2	1	-	1	2
CO5	2	2	2	3	2	2	-	2	2
Articulation	2.6	2.4	2.1	2.4	2	1.1	1	1.1	1.6

Course Title:	**Operating Systems**
Course Code:	**25VMT0C103**
Semester:	**I**
Credits:	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides an in-depth exploration of Advanced operating system concepts, focusing on the design, implementation, and management of modern operating systems. Topics include process synchronization, memory management, file systems, distributed systems, and security mechanisms. Practical assignments and case studies will reinforce theoretical knowledge, preparing students for real-world system-level programming and research.

Course Outcomes(COs)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Apply process management and synchronization techniques.	L3
CO2	Implement memory Management Techniques	L3
CO3	Implement File Management Techniques	L3
CO4	Design solutions using modern OS tools and case studies in distributed systems.	L3

CO5	Evaluate modern tools and techniques in real-world OS environments.	L5

Course Modules:

No. of Hours:24

Module 1: Introduction to Operating Systems and Process Management
Scope of Module: Introduces core OS concepts, structures, and the fundamentals of process management.
Topics: Overview of Operating Systems: Definitions, functions, types (batch, time-sharing, distributed, real-time); OS structure and components; System calls and APIs; Microkernel vs Monolithic Kernel architectures; System Boot Process and Initialization (GRUB, systemd); Process concepts, process states, process control block (PCB); Process scheduling algorithms (FCFS, SJF, Round Robin, Priority); Context switching; Process scheduling in multi-core and NUMA systems; Case studies on modern OS architectures (Linux, Windows).
Learning Outcome: *Students will be able to describe OS structures, functions, and manage basic process scheduling scenarios.*

No. of Hours:24

Module 2: Process Synchronization, Deadlock, and CPU Scheduling
Scope of Module:

Covers critical section problems, synchronization mechanisms, deadlock handling, and CPU scheduling strategies.
Topics: Critical section problem, Peterson's solution, semaphores, monitors; Spinlocks, Mutexes, and Condition Variables; Hardware support for synchronization (Test-and-Set, Compare-and-Swap); Classic synchronization problems (producer-consumer, readers-writers, dining philosophers); Deadlock characterization, prevention, avoidance (Banker's algorithm), detection, recovery, and resource preemption; CPU scheduling strategies (multilevel queue, multilevel feedback queue); Real-world scheduling policies (CFS in Linux, Windows scheduler comparison); Case studies on synchronization in Linux kernel.
Learning Outcome: *Students will be able to apply synchronization techniques and deadlock handling strategies in process management.*

No. of Hours:24

Module 3: Memory Management, Virtual Memory, and Storage Systems
Scope of Module: Focuses on memory organization, virtual memory, and file systems management.
Topics: Memory management strategies: contiguous allocation, paging, segmentation; Virtual memory: demand paging, page replacement algorithms (LRU, FIFO, Optimal); Thrashing and performance tuning; Memory-mapped I/O and Direct Memory Access (DMA); Swap space management and tuning; File system interface and implementation, allocation methods (contiguous, linked, indexed); Directory structures, file system mounting, journaling file systems;

Advanced file system topics: RAID levels, ZFS/Btrfs file systems; Security implications in storage systems (encryption at rest); Case study: EXT4 file system analysis.
Learning Outcome:*Students will be able to analyze memory and storage management strategies in OS environment*

No. of Hours:24

Module 4: Virtualization, Containerization, and Real-Time Systems
Scope of Module: Explores virtualization technologies, container-based systems, and introduces real-time OS concepts.
Topics: Virtualization concepts, hypervisors (Type 1 and Type 2), virtual machine monitors (VMWare, KVM); Nested Virtualization (running VMs inside VMs); Containerization with Docker, Kubernetes orchestration basics; Container Security (namespaces, cgroups, seccomp); Comparison of VMs vs Containers; Performance tuning of VMs and Containers; Introduction to real-time operating systems (RTOS), real-time scheduling algorithms; RTOS performance metrics and real-world RTOS examples (FreeRTOS, VxWorks); Case study: Docker-based microservices in cloud environments.
Learning Outcome:*Students will be able to evaluate virtualization and containerization technologies and apply real-time scheduling concepts.*

No. of Hours:24

Module 5: Security, Protection Mechanisms, and Advanced OS Case Studies
Scope of Module: Focuses on OS security, protection models, and advanced industrial applications of operating systems.
Topics: OS security principles, access control models (DAC, MAC, RBAC); Kernel hardening techniques (grsecurity, seccomp); Network security in OS: Firewall configurations (iptables, nftables); Secure Boot and Trusted Platform Module (TPM); Security vulnerabilities and mitigation strategies; Encryption, authentication, and auditing in OS; Introduction to SELinux, AppArmor; Forensics in OS environments (log analysis, intrusion detection); Case studies on OS security breaches and their prevention; Mini Project: Design a simple OS-level security module, process monitoring utility, or Intrusion Detection System (IDS) script.
Learning Outcome:*Students will be able to assess and implement security models and protection mechanisms in modern OS environments.*

Module	**No. of Pre-Recordings**
1	**10**
2	**12**
3	**10**
4	**12**

5	**10**

Note: Minimum number of pre-recordings is 10.

Prerequisites :

- Basic knowledge of computer fundamentals.

Pedagogy / Teaching Methodology:

- ✓ Lectures, Hands-on Lab Sessions, Problem-solving Tasks, Case-based Assignments, Group Projects

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. NPTEL, Coursera, edX, MIT OCW — based on relevance to topics.

Suggested Readings:

- Standard textbooks, research articles, online tutorials and eBooks related to the subject.

Recommended Software/Tools (If applicable):

1. GCC Compiler, Visual Studio Code, Linux Terminal, C IDEs

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools

5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO/PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	2	2	2	1	-	1	2
CO2	2	3	3	2	2	1	1	2	1
CO3	3	2	2	2	2	1	1	2	2
CO4	2	3	3	2	2	2	1	1	2
CO5	3	2	2	3	3	2	1	2	2
Articulation	2.6	2.4	2.4	2.2	2.2	1.4	1	1.4	1.6

Course Title:	**Data Structures**
Course Code:	**25VMT0C104**
Semester:	**I**
Credits:	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description

This course introduces Advanced Concepts of data structures, algorithms, and their practical use in computing applications. Students will learn how to implement, analyze, and apply various linear and non-linear data structures to build efficient and scalable software systems

Course Outcomes(Cos)

Upon successful completion of this course, learners will be equipped to:

CO	**Course Outcome**	**BTL**
CO1	Understand the fundamentals of advanced linear and data structures.	L1
CO2	Apply appropriate data structures for solving complex computational problems.	L3
CO3	Apply tree graph techniques in software solutions.	L3
CO4	Implement hashing techniques in software solutions.	L3
CO5	Evaluate efficient programs using modern tools and real-world case studies.	L5

Note: Implementation can be done in C/C++

Course Modules

No. of Hours:24

Module 1: : Linear Data Structures
Scope of Module Covers implementation and usage of linear storage structures.
Topics: Arrays, Stacks (array/linked), Queues (normal, circular), Dequeues, Linked list operations (SLL, DLL, CLL), Stack/Queue with linked lists, Applications in expression evaluation, Bracket matching, Recursion stack, Queue in scheduling, List-based simulation, Sparse matrices, Time/space complexity basics, Array vs list.
Learning Outcome: *Apply linear structures in solving algorithmic problems.*

No. of Hours:24

Module 2: Trees and Binary Trees
Scope of Module Focuses on hierarchical data structures with operations and applications of binary trees and advanced trees.
Topics: Tree definitions and terminologies, Binary tree representation, Binary tree traversals (inorder, preorder, postorder), Binary search tree (BST), Insertion in BST, Deletion in BST, Balanced trees (AVL trees), Height-balanced trees,

Expression trees, Threaded binary trees, Heap trees, Min-heap and max-heap, Priority queue using heap, B-Trees, Applications of trees.
Learning Outcome: *Students will be able to implement tree structures and use them for hierarchical data processing.*

No. of Hours:24

Module 3: Graphs and Applications
Scope of Module: Introduction to graphs and path-finding algorithms.
Topics: Graph representation (adjacency list/matrix), BFS, DFS, Topological sorting, Dijkstra's algorithm, Bellman-Ford, Floyd-Warshall, Kruskal's algorithm, Prim's algorithm, Network flow, Graph coloring, Euler and Hamiltonian paths, Graph traversals for cycles, Applications in networks, Graph problem modeling.
Learning Outcome: *Analyze and solve graph-based problems using suitable algorithms*

No. of Hours:24

Module 4: Hashing and Searching Structures
Scope of Module: Efficient search, hash, and map-based data access.
Topics:

Hash tables, Hash functions, Collision resolution (chaining, open addressing), Load factor, Linear probing, Double hashing, Bloom filters, Dictionaries, Sets, Unordered maps, Binary search trees vs hashing, Search time analysis, Skip lists, LRU Cache design, Memory-optimized hashing, Applications in OS/DB.
Learning Outcome: *Design and implement efficient search and mapping structures.*

No. of Hours:24

Module 5: Case Studies and Tools
Scope of Module: Tools and case implementations using advanced DS.
Topics: STL containers in C++ (vector, list, map, set), Data structure libraries in C++, Visualization tools (VisuAlgo, Graphviz), IDEs (CLion, Eclipse), Real-world DS projects, Competitive programming platforms, Time-space profiler tools, LeetCode/Codeforces case studies, DS in AI/ML preprocessing, Game development DS, Cache designs, Real DBMS indexing, Distributed DS structures.
Learning Outcome: *Evaluate and apply real-world DS solutions using modern tools and case studies.*

Module	**No. of Pre-Recordings**
1	**10**
2	**12**

3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10.

Prerequisites :

- Basic knowledge of programming and algorithms.

Pedagogy / Teaching Methodology:

- ✓ Interactive Lectures
- ✓ Hands-on Programming Labs
- ✓ Assignments and Case Studies
- ✓ Mini Projects

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. NPTEL: Data Structures and Algorithms
2. Coursera: Data Structures by UC San Diego

Suggested Readings:

- Seymour Lipschutz, *Data Structures*, McGraw-Hill
- Ellis Horowitz, Sartaj Sahni, *Fundamentals of Data Structures*, Universities Press
- Mark Allen Weiss, *Data Structures and Algorithm Analysis in C*, Pearson Education
- Online resources and research papers related to data structures

Recommended Software/Tools (If applicable):

1. C/C++ Compiler

2. Visual Studio Code / Code::Blocks
3. Online IDEs (Repl.it, GeeksforGeeks IDE)
4. Visualgo.net for algorithm visualization

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO/PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	3	2	2	3	-	2	1
CO2	3	3	3	2	2	2	1	2	2
CO3	2	2	3	3	2	2	3	1	2
CO4	3	2	3	2	2	-	2	1	2
CO5	2	3	2	3	2	2	-	2	1
Articulation	2.6	2.4	2.8	2.4	2	2.2	2	1.6	1.6

Corporate Office
91/2, Dr. A N Krishna Rao Road V V Puram,
Bengaluru - 560 004
P: +91 80 4343 1000

Centre for Distance & Online Education
319, 25th Main, 17th Cross,
J P Nagar 6th Phase
Bengaluru – 560 078
P: +91 80 4343 0100

Course Title:	**AI Skills for Future**
Course Code:	**26VMX0S105**
Semester:	**I**
Credits:	**2**
Course Type	**Skill Enhancement Course**
Learning Hours	60
Live Sessions	**12 hours**

Course Outcomes:

CO1: Understand Generative AI and LLMs in depth
CO2: Create text, image, and voice content using AI
CO3: Automate daily tasks and business workflows
CO4: Build AI agents that act independently
CO5: Design and deploy simple AI apps without coding

Course Modules
Module 1: Generative AI Prompt Engineering for Real Workflows
What is Generative AI and how it's transforming industries, Structure of an effective prompt: *Role → Task → Context → Output,* Common prompting mistakes and how to fix them, How large language models (LLMs) like ChatGPT, Claude, and Google AI Studio work, advanced prompting (few-shot, iterative, style-based), Using your own files in ChatGPT and Claude (RAG basics), Research and summarization using AI
Learning Outcome: Students will be able to understand how Generative AI works and master the art of prompting.

Module 2: Creative AI and Voice & Audio Creation
Understanding AI image generation, creating posters, campaign assets, and product visuals, building brand identity using Krea.ai, Generating realistic voiceovers using ElevenLabs, Script

writing and tone control with ChatGPT & Claude, Integrating voice into short videos or podcasts ***Learning Outcome:*** Students will be able to use AI to generate visuals, voiceovers, and creative storytelling content.

Module 3: AI Automation & Productivity

Logic of triggers, actions, and workflows, setting up automation between apps (Gmail, Sheets, Drive), Connecting OpenAI to Make.com for intelligent automation, creating social media Management Workflow and research automation flows, automating daily reports and reminders, building personal productivity dashboards, Integrating Claude and ChatGPT into existing workflows

Learning Outcome: Automate daily workflows and tasks using AI and no-code tools.

Module 4: Advanced AI Agents & Decision Flows

Introduction to n8n as a visual AI workflow builder, understanding nodes, triggers, and data passing, Using OpenAI node for smart responses and actions, designing multi-step reasoning workflows, Connecting Google Sheets, Slack, and other APIs, building autonomous agents that perform specific tasks

Learning Outcome: Students will be able to build AI agents that can think, decide, and act automatically.

Module 5: AI App Building & Deployment

Fundamentals of UI/UX for AI apps, structuring front-end and AI back-end flows, designing app logic using n8n and Make.com, integrating OpenAI and Claude APIs to your apps, handling user inputs and dynamic responses, using Google AI Studio to experiment with models, hosting and testing your AI app online, adding analytics and monitoring usage, final demo preparation and peer feedback

Learning Outcome: Students will be able to learn to design, connect, and deploy simple AI apps using no-code tools

CENTRE FOR DISTANCE AND ONLINE EDUCATION

ONLINE

Semester-II

Corporate Office
91/2, Dr. A N Krishna Rao Road V V Puram,
Bengaluru - 560 004
P: +91 80 4343 1000

Centre for Distance & Online Education
319, 25th Main, 17th Cross,
J P Nagar 6th Phase
Bengaluru – 560 078
P: +91 80 4343 0100

Course Title:	**Advanced Database Systems**
Course Code:	**25VMT0C201**
Semester:	**II**
Credits:	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course delves into advanced concepts and techniques in database systems, focusing on design, implementation, optimization, and emerging trends. Topics include distributed databases, NoSQL systems, data warehousing, transaction management, and big data technologies. The course emphasizes practical applications through case studies and hands-on projects.

Course Outcomes (COs)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Understand advanced database models and architectures.	L1
CO2	Design and optimize relational and NoSQL databases.	L3
CO3	Implement transaction management and concurrency control.	L4
CO4	Evaluate distributed database systems and their challenges.	L5
CO5	Apply data warehousing and big data technologies.	L3

Course Modules

No. of Hours:24

Module 1: Advanced Database Models and Architectures
Scope of Module: Covers advanced database models, including object-oriented, object-relational, and NoSQL databases, and their architectures.
Topics: Review of relational database systems; Object-oriented databases; Object-relational databases; NoSQL database overview; Document databases (MongoDB); Key-value databases (Redis); Column-family databases (Cassandra); Graph databases (Neo4j); NewSQL systems; Polyglot persistence; CAP theorem; BASE properties; Cloud-native database systems; Multi-model databases; Case study: MongoDB vs. PostgreSQL.
***Learning Outcome**: Students will be able to compare and contrast different database models and select appropriate architectures for specific applications.*

No. of Hours:24

Module 2: Database Design and Query Optimization
Scope of Module: Focuses on advanced techniques for database design, indexing, query optimization, and performance tuning
Topics: Advanced normalization (1NF–5NF); Denormalization strategies; Physical vs. logical design Partitioning strategies; Indexing: B-tree, hash, bitmap; Covering indexes and index-only scans Query execution plans; Cost-based query optimization; Heuristic-based optimization; Materialized views; Query rewriting

techniques; Adaptive query optimization; Statistics collection and use; SQL performance tuning tools; Hands-on: SQL tuning with EXPLAIN and ANALYZE.
Learning Outcome: *Students will be able to design efficient database schemas and optimize query performance.*

No. of Hours:24

Module 3: Transaction Management and Concurrency Control
Scope of Module: Explores transaction properties, concurrency control mechanisms, and recovery techniques in database systems
Topics: ACID properties; Transaction lifecycle; Serializability and schedules; Lock-based concurrency control; Deadlock detection and prevention; Timestamp ordering; Optimistic concurrency control; Two-phase locking; Multiversion Concurrency Control (MVCC); Isolation levels and anomalies; Serializable snapshot isolation; Write-ahead logging; Checkpointing; Recovery methods; Case study: PostgreSQL concurrency and recovery.
Learning Outcome: *Students will be able to implement and analyze transaction management and concurrency control mechanisms.*

No. of Hours:24

Module 4: Distributed Database Systems
Scope of Module: Examines distributed database architectures, data fragmentation, replication, and consistency models.

Topics: Distributed database architecture; Client-server vs. peer-to-peer; Horizontal and vertical fragmentation; Data allocation and replication; Distributed concurrency control; Distributed transactions; Two-phase commit protocol; Distributed query processing. Query decomposition and localization; CAP theorem in distributed databases; Eventual consistency models; Conflict resolution strategies; Fault tolerance and availability; Case study: Google Spanner; Hands-on: Setting up a distributed Cassandra cluster.
Learning Outcome: *Students will be able to evaluate distributed database systems and address their challenges.*

No. of Hours:24

Module 5: Data Warehousing and Big Data Technologies
Scope of Module: Introduces data warehousing concepts, ETL processes, OLAP, and big data technologies
Topics: Data warehousing concepts; Star schema vs. snowflake schema; Fact and dimension tables ETL process design; Data cleaning and transformation; Data staging and loading; OLAP operations and cubes; Hadoop ecosystem (HDFS, MapReduce); Apache Hive and Pig Apache Spark fundamentals; NoSQL for Big Data: HBase and Cassandra; Data lakes and lakehouse architecture; Real-time data pipelines (Kafka + Spark Streaming); Columnar storage formats (Parquet, ORC); Mini-project: Building a Spark-based data pipeline
Learning Outcome: *Students will be able to apply data warehousing and big data technologies for analytical processing*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites :

- Basic knowledge of database management systems (DBMS) and SQL.

Pedagogy / Teaching Methodology:

- ✓ Conceptual and problem-based learning
- ✓ Hands-on labs and case studies
- ✓ Group projects and presentations
- ✓ Quizzes and assignments using database tools

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. Advanced Database Systems by NPTEL: https://nptel.ac.in/courses/106105171
2. Database Systems by Stanford Online: https://online.stanford.edu/courses/soe-ydatabases-database-systems

Suggested Readings:

- Raghu Ramakrishnan and Johannes Gehrke, *Database Management Systems*, McGraw-Hill, 2003.

- Hector Garcia-Molina, Jeffrey D. Ullman, and Jennifer Widom, *Database Systems: The Complete Book*, Pearson, 2008.
- Martin Kleppmann, *Designing Data-Intensive Applications*, O'Reilly, 2017.

Recommended Software/Tools:

1. PostgreSQL, MongoDB, Cassandra
2. Hadoop, Spark
3. SQL Developer, Tableau

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO/PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	-	2	1	-	1	-
CO2	2	3	2	1	1	-	-	2	-
CO3	3	2	2	1	1	1	1	1	2
CO4	2	3	3	1	2	2	-	2	2
CO5	3	2	1	2	2	1	1	2	2
Articulation	2.6	2.6	1.8	1.25	1.6	1.25	1	1.4	2

Course Title:	**Java Programming**
Course Code:	**25VMT0C202**
Semester:	**II**
Credits:	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides an in-depth understanding of Java programming, covering object-oriented principles, core APIs, multithreading, networking, and modern frameworks. Emphasis is placed on practical implementation through projects, case studies, and hands-on labs to build robust applications.

Course Outcomes (Cos)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Understand object-oriented principles in Java.	L1
CO2	Develop applications using Java core APIs and collections.	L3
CO3	Implement multithreading and exception handling.	L4
CO4	Design GUI applications and network-based solutions.	L5
CO5	Apply Java frameworks (Spring Boot) for enterprise applications.	L3

Course Modules

No. of Hours:24

Module 1: Object-Oriented Programming in Java
Scope of Module: Covers fundamental OOP concepts, Java syntax, and class design
Topics: Introduction to Java and JVM; Data types, operators, and control structures; Class structure and object creation; Encapsulation and constructors; Inheritance and method overriding; Polymorphism and dynamic binding; Interfaces and abstract classes; Package creation and import; Access modifiers and scope; Static vs. instance members; Exception handling: try-catch-finally; Custom exceptions; Assertions and error handling; Object class and method overriding; Case study: Modular banking system design.
Learning Outcome: *Students will be able to apply OOP principles such as encapsulation, inheritance, and polymorphism to design structured and reusable Java applications.*

No. of Hours:24

Module 2: Java Core APIs and Collections
Scope of Module: Focuses on Java standard libraries and data structures.

Topics: String manipulation and immutability; Regular expressions; Wrapper classes and autoboxing Generics in Java; ArrayList, LinkedList implementations; HashSet, TreeSet, LinkedHashSet usage; HashMap, TreeMap, LinkedHashMap differences; Iterators, ListIterators, and enhanced for-loop; Comparable vs. Comparator; Collections utility class; File handling with java.io and java.nio; Object serialization and deserialization; Properties and configuration files; Practical: Inventory management system; Debugging collections and memory leaks.
Learning Outcome: *Students will be able to leverage core Java libraries and the Collections Framework to build efficient, maintainable applications with complex data structures.*

No. of Hours:24

Module 3: Multithreading and Concurrency
Scope of Module: Explores concurrent programming in Java.
Topics: Thread lifecycle and states; Thread creation via Thread class; Runnable interface and Lambda expressions; Thread prioritization and yielding; Synchronized methods and blocks; Intrinsic locks and synchronized collections; Volatile keyword and atomic variables; Deadlock scenarios and prevention strategies; wait(), notify(), and notifyAll(); Thread pooling and ExecutorService; Callable and Future interfaces; ConcurrentHashMap and thread-safe queues; ForkJoinPool basics; Case study: Multithreaded chat application; Concurrency debugging tools and techniques.

Learning Outcome: *Students will be able to implement multithreaded programs and apply Java concurrency tools to ensure safe, scalable execution of parallel tasks.*

No. of Hours:24

Module 4: GUI and Networking
Scope of Module: Introduces Java FX and network programming.
Topics: Introduction to JavaFX and GUI architecture; SceneBuilder and JavaFX controls; Layout managers: BorderPane, VBox, HBox; Event-driven programming in Java; Creating forms and validation; Swing basics and component lifecycle; Action listeners and adapters; Socket programming: TCP; Datagram programming: UDP; HTTP communication using HttpClient Using REST APIs in Java; JDBC for database connectivity; CRUD operations with JDBC Mini-project: GUI Weather App with REST API; Threading in GUI and networking applications
Learning Outcome: *Students will be able to design and build user-friendly GUI applications integrated with network communication and backend services using Java.*

No. of Hours:24

Module 5: Java Frameworks and Modern Tools
Scope of Module: Covers Spring Boot and testing frameworks.

Topics: Introduction to Spring Boot and its architecture; Dependency Injection (DI) in Spring; Creating RESTful APIs with Spring Boot; Spring annotations and application.properties; CRUD operations with Spring Data JPA; Connecting to MySQL/PostgreSQL; Validation with Spring MVC; Unit testing with JUnit 5; Mocking with Mockito; Postman for API testing; Building with Maven and Gradle; Logging with Logback and SLF4J; Creating JAR/WAR packages; Dockerizing a Spring Boot app; Capstone project: E-commerce backend.
Learning Outcome: *Students will be able to develop full-stack Java applications using Spring Boot and related modern tools, and deploy them in enterprise-ready environments.*

Module	**No. of Pre-Recordings**
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10

Prerequisites :

- Basic programming knowledge (C/C++/Python).

Pedagogy / Teaching Methodology:

- ✓ Interactive lectures with live coding
- ✓ Hands-on labs and debugging sessions

- ✓ Agile-style group projects
- ✓ Code reviews and peer assessments

Certificate / Value Added Courses Recommended:

1. Java Programming by NPTEL: https://nptel.ac.in/courses/106105168
2. Java Fundamentals by Oracle Academy: https://academy.oracle.com

Suggested Readings:

- Herbert Schildt, *Java: The Complete Reference*, McGraw-Hill, 2021.
- Joshua Bloch, *Effective Java*, Addison-Wesley, 2017.
- Craig Walls, *Spring Boot in Action*, Manning, 2020.

Recommended Software/Tools:

1. JDK 17+, IntelliJ IDEA/Eclipse
2. Spring Tool Suite, Postman
3. Git, Maven

Curriculum Development:

Sl.No	**Identify and Highlight in the Modules**
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools

5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO/PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	–	2	1	–	1	2
CO2	2	3	2	1	-	–	–	2	-
CO3	3	2	2	-	-	1	2	-	2
CO4	2	3	3	1	2	2	–	1	2
CO5	3	2	-	2	2	1	1	2	2
Articulation	2.6	2.4	2	1.33	2	1.25	1.5	1.5	2

Course Title:	**Web Technologies**
Course Code:	**25VMT0C203**
Semester:	**II**
Credits:	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides comprehensive knowledge of modern web development technologies, covering front-end design, back-end programming, databases, and web security. Students will learn to build responsive, interactive web applications using HTML5, CSS3, JavaScript, PHP/Node.js, and frameworks like React and Express. Practical implementation through projects and case studies is emphasized.

Course Outcomes (COs)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Understand client-side web technologies (HTML5, CSS3, JavaScript).	L1
CO2	Develop interactive front-end applications using frameworks.	L3
CO3	Implement server-side programming and database integration.	L4
CO4	Evaluate web security threats and countermeasures.	L5

CO5	Build full-stack web applications using modern tools.	L3

Course Modules

No. of Hours:24

Module 1: Client-Side Web Technologies
Scope of Module: Covers fundamental technologies for building web interfaces.
Topics: HTML5 document structure and semantic tags; Forms and input validation; CSS3 selectors and specificity; Box model, flexbox, and grid layouts; Responsive design principles (media queries); Typography and UI consistency; JavaScript syntax and data types; Control structures and functions; ES6+ features: let/const, arrow functions, template literals; DOM tree and traversal; Event handling and delegation; Form validation using JavaScript; Browser console and developer tools; Animation with CSS and JavaScript; Project: Portfolio website with interactivity.
Learning Outcome: *Students will be able to create accessible, responsive, and well-structured web pages using HTML5, CSS3, and modern JavaScript.*

No. of Hours:24

Module 2: Front-End Frameworks
Scope of Module: Focuses on modern JavaScript frameworks for UI development

Topics: Introduction to React.js and component lifecycle; JSX syntax and virtual DOM; Functional components and props; State and event management; Conditional rendering and lists; Forms and controlled components; React Hooks (useState, useEffect); Component communication; React Router for navigation; Context API for global state; AJAX requests with Axios and Fetch API; Using UI libraries (Bootstrap, Material UI); Creating reusable components; Code splitting and lazy loading; Project: Product listing page for e-commerce.
Learning Outcome: *Students will be able to build scalable and maintainable front-end applications using modern JavaScript frameworks like React.*

No. of Hours:24

Module 3: Server-Side Programming
Scope of Module: Explores back-end development and database integration.
Topics: Node.js basics and NPM; Creating a local web server; Express.js routing and middleware; Handling POST and GET requests; Template engines (EJS/Handlebars); Form data parsing and validation; MySQL and MongoDB CRUD operations; Connection pooling and error handling; RESTful API design principles; JWT and session-based authentication; Role-based access control; File uploads and file system module; Environment variables and configuration; Project: Blog system with authentication and database; Debugging Node.js applications**.**
Learning Outcome: *Students will be able to build robust server-side applications and RESTful APIs integrated with databases using Node.js and Express.*

No. of Hours:24

Module 4: Web Security
Scope of Module: Addresses security vulnerabilities and protection mechanisms.
Topics: Understanding HTTP protocol and HTTPS; SSL certificates and secure headers; OWASP Top 10 vulnerabilities; SQL injection and input sanitization; Cross-Site Scripting (XSS) and mitigation; Cross-Site Request Forgery (CSRF); Password hashing with bcrypt; Authentication vs. Authorization; Content Security Policy (CSP); Secure cookie practices; Same-Origin Policy and CORS; Rate limiting and API throttling; Penetration testing tools (Burp Suite, OWASP ZAP); Security audit and logging; Project: Secure login and audit trail system
Learning Outcome: *Students will be able to identify, prevent, and mitigate common web application security threats using industry best practices.*

No. of Hours:24

Module 5: Full-Stack Development
Scope of Module: Integrates all layers of web application development.
Topics: Overview of MERN/MEAN stacks; Connecting frontend with backend via APIs; Redux basics and state management; User authentication and protected routes; Role-based content rendering; API testing with Postman; Code

versioning with Git and GitHub; Continuous deployment workflows; Hosting frontend (Netlify, Vercel); Hosting backend (Render, Heroku) Performance optimization tips; Unit testing with Jest and Mocha; Linting and code formatting (Prettier, ESLint); DevOps basics for deployment; Capstone Project: Social media dashboard with analytics.
Learning Outcome: *Students will be able to develop, test, and deploy full-stack web applications using modern development tools and deployment platforms.*

Module	**No. of Pre-Recordings**
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10

Prerequisites :

- Basic programming knowledge (HTML/CSS recommended).

Pedagogy / Teaching Methodology:

- Interactive coding demonstrations
- Pair programming exercises
- Project-based learning
- Security hackathons

Certificate / Value Added Courses Recommended:

1. Web Development by MDN: https://developer.mozilla.org
2. Full-Stack Open: https://fullstackopen.com

Suggested Readings:

- Jon Duckett, *HTML & CSS: Design and Build Websites*, Wiley, 2011
- Eric Freeman, *Head First JavaScript Programming*, O'Reilly, 2014
- David Flanagan, *JavaScript: The Definitive Guide*, O'Reilly, 2020

Recommended Software/Tools:

1. VS Code, Chrome DevTools
2. Postman, MongoDB Compass
3. React DevTools, Git

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO/PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	–	2	1	–	1	-
CO2	2	3	2	1	1	–	–	2	1
CO3	3	2	2	1	1	1	1	1	2
CO4	2	3	3	1	2	2	–	1	2
CO5	3	2	1	2	2	1	1	2	2
Articulation	2.6	2.4	1.8	1.25	1.6	1.25	1	1.4	1.75

Course Title:	Cloud Computing Foundations
Course Code:	**25VMT0C204**
Semester:	**II**
Credits:	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course introduces fundamental concepts of cloud computing, including service models, deployment models, virtualization, and cloud architecture. Students will gain hands-on experience with major cloud platforms (AWS/Azure/GCP) and learn to deploy, manage, and secure cloud-based solutions. The course emphasizes practical applications through labs and real-world case studies.

Course Outcomes (COs)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Understand cloud computing concepts and service models.	L1
CO2	Deploy and manage virtualized infrastructure.	L3
CO3	Analyze cloud storage and networking solutions.	L4

CO4	Evaluate cloud security and compliance frameworks.	L5
CO5	Develop cloud-native applications using PaaS.	L3

Course Modules

No. of Hours:24

Module 1: Cloud Computing Fundamentals
Scope of Module: Covers basic concepts, characteristics, and service models of cloud computing.
Topics: Definition and evolution of cloud computing; Essential characteristics of cloud (NIST model); Service models: IaaS, PaaS, SaaS; Deployment models: Public, Private, Hybrid, Community; Benefits and challenges of cloud adoption; Cloud computing use cases; Cloud vs. traditional IT infrastructure; Cloud service providers overview (AWS, Azure, GCP); Cloud service level agreements (SLAs); Multitenancy and resource pooling; Elasticity and scalability; Cloud computing cost models and pricing; Case study: Cloud migration strategy; Green computing and cloud efficiency; Cloud-first strategies in digital transformation.
Learning Outcome: Students will be able to differentiate between cloud service and deployment models, and evaluate when and how to migrate to cloud environments.

No. of Hours:24

Module 2: Virtualization and Cloud Infrastructure
Scope of Module: Focuses on virtualization technologies and cloud infrastructure management.

Topics: Virtualization concepts and benefits; Types of virtualization: Server, Storage, Network; Hypervisors: Type 1 (bare-metal) and Type 2 (hosted); Virtual machine creation and lifecycle; Introduction to containers and Docker; Container orchestration basics (Kubernetes overview); Infrastructure as a Service (IaaS); AWS EC2, Azure VMs, GCP Compute Engine; Cloud autoscaling and elasticity; Elastic Load Balancing (ELB); Monitoring and cloud resource usage; Infrastructure as Code: Terraform and CloudFormation; Hands-on: Launching a VM on AWS/GCP; Cloud cost optimization techniques; Best practices for resource provisioning**.**
Learning Outcome**:** *Students will be able to configure and manage virtualized infrastructure on cloud platforms using IaaS and container technologies.*

No. of Hours:24

Module 3: Cloud Storage and Networking
Scope of Module: Explores storage solutions and network architectures in cloud environments.
Topics: Cloud storage types: Object, Block, File; Storage services: Amazon S3, Azure Blob, Google Cloud Storage; Data redundancy and availability zones; Content Delivery Networks (CDNs) Edge computing and latency reduction; Data lifecycle and storage classes; Virtual Private Cloud (VPC) architecture; Subnets, gateways, and NAT; Cloud firewalls and security groups Cloud DNS and networking services; VPNs and hybrid cloud connectivity ;Database as a Service (DBaaS): RDS, Cosmos DB, Cloud SQL; Data backup and disaster recovery; Case study: Designing a high availability architecture; Storage and network performance benchmarking.

Learning Outcome: *Students will be able to design and configure secure, high-performance storage and networking solutions within cloud environments.*

No. of Hours:24

Module 4: Cloud Security and Compliance
Scope of Module: Addresses security challenges and compliance requirements in cloud computing.
Topics: Cloud security principles and models; Shared responsibility model; Identity and Access Management (IAM); Role-based access control (RBAC); Data encryption in transit and at rest; Key management services; Security groups vs. network ACLs; Firewall policies and VPN security; Security incident detection and response; Cloud compliance frameworks (GDPR, HIPAA, ISO 27001); Security posture assessments; Penetration testing and vulnerability scanning; Logging and auditing best practices; Case study: Securing a multi-tenant cloud app. Mini-project: Implementing IAM and encryption in AWS.
Learning Outcome: *Students will be able to implement cloud security measures and ensure compliance with data protection regulations and security frameworks.*

No. of Hours:24

Module 5: Cloud-Native Application Development

Scope of Module: Covers development of applications optimized for cloud environments
Topics: Introduction to cloud-native principles; Twelve-factor app methodology; Serverless architecture overview; AWS Lambda and Azure Functions; Event-driven programming in the cloud; Microservices architecture and benefits; REST API design and testing (Postman); Containers vs. serverless vs. VMs; CI/CD pipelines: GitHub Actions, AWS CodePipeline; Monitoring and logging (CloudWatch, Stackdriver); Application scaling and performance tuning; API Gateway integration and throttling; DevOps tools for cloud (Jenkins, Terraform) Case study: Microservices deployment; Capstone Project: Build and deploy a serverless application.
Learning Outcome: *Students will be able to develop, deploy, and monitor scalable, cloud-native applications using modern cloud services and DevOps tools.*

Module	**No. of Pre-Recordings**
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10

Prerequisites :

- Basic knowledge of operating systems and networking concepts.

Pedagogy / Teaching Methodology:

- ✓ Hands-on labs with cloud platforms
- ✓ Case studies of enterprise cloud migrations
- ✓ Security audit simulations
- ✓ Group projects with real-world scenarios

Certificate / Value Added Courses Recommended:

1. AWS Cloud Practitioner: https://aws.amazon.com/certification/
2. Microsoft Azure Fundamentals: https://learn.microsoft.com/en-us/certifications/

Suggested Readings:

- ✓ Dan Marinescu, *Cloud Computing: Theory and Practice*, 3rd Ed., Morgan Kaufmann, 2022
- ✓ Thomas Erl, *Cloud Computing: Concepts, Technology & Architecture*, Prentice Hall, 2021
- ✓ Official documentation from AWS, Azure, and GCP

Recommended Software/Tools:

1. AWS Free Tier, Azure Portal
2. Docker, Kubernetes (basic)
3. Terraform, CloudFormation

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO/PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	–	2	1	–	1	1
CO2	2	3	2	1	-	–	–	2	1
CO3	3	2	2	1	-	1	2	1	2
CO4	2	3	3	1	2	2	–	1	2
CO5	3	2	1	2	2	-	2	2	2
Articulation	2.6	2.4	1.8	1.25	2	1.33	2	1.5	1.6

Course Title:	**Environmental, Social, and Governance (ESG) in IT**
Course Code:	**25VMT0C205**
Semester:	**I**
Credits:	**4**
Course Type	**AECC**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course explores the integration of Environmental, Social, and Governance (ESG) principles into information technology. It covers sustainable IT practices, ethical AI, data privacy, and corporate governance in tech organizations. Students will learn to assess ESG risks, implement green computing solutions, and align IT strategies with global sustainability goals through case studies and projects.

Course Outcomes (Cos)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Explain ESG frameworks relevant to IT	L1
CO2	Apply green computing techniques	L3
CO3	Analyze social impacts of AI and big data	L4
CO4	Evaluate governance models for IT ethics	L5
CO5	Evaluate sustainable IT policies for organizations	L5

Course Modules

No. of Hours:24

Module 1: ESG Fundamentals in IT
Scope of Module: Introduces core ESG concepts and their application in technology sectors.
Topics: Understanding ESG in the context of IT; Global ESG frameworks: UN SDGs, GRI, SASB; Environmental impacts of digital infrastructure; Carbon footprint of data centers and cloud computing; E-waste management and recycling strategies; IT lifecycle sustainability assessments; Role of ICT in sustainable development goals; Green certifications for IT (Energy Star, EPEAT); Sustainability KPIs in IT projects; ESG compliance for tech startups AI and sustainability intersections; Digital ethics and sustainable design; Sustainable supply chain management in IT; Case study: Google's carbon-neutral cloud initiatives; Future of ESG in emerging technologies
Learning Outcome: *Students will articulate how ESG principles apply to IT infrastructure, service models, and sustainability reporting.*

No. of Hours:24

Module 2: Green Computing
Scope of Module: Focuses on sustainable IT practices and resource optimization.
Topics: Green hardware lifecycle management; Energy-efficient processor design; Power-aware scheduling algorithms; Green software design and refactoring;

Virtualization for sustainability; Resource optimization in cloud environments; Carbon-aware computing; Renewable energy integration in data centers; Measuring IT carbon footprints; Green networking and IoT devices; Cooling technologies and data center PUE; Server consolidation strategies; Green code practices in DevOps; Life Cycle Assessment (LCA) tools for IT systems; Lab: Calculate emissions from virtual machines.
Learning Outcome: *Students will be able to apply green computing principles to minimize the environmental impacts of IT operations and software systems.*

No. of Hours:24

Module 3: Social Responsibility in Tech
Scope of Module: Examines ethical implications of AI, data, and digital inclusion.
Topics: Digital equity and inclusion challenges; Algorithmic bias and mitigation techniques; AI transparency and explainability; Ethical data collection and consent; GDPR, CCPA, and global data privacy laws; Bias in training data for machine learning; Accessible design principles in software; Digital literacy and education programs; Tech for underserved communities. Social media and societal impact; AI in healthcare and social justice; Tech activism and digital rights; Workplace inclusivity in tech companies; Case study: Microsoft's AI for Accessibility Lab: Bias detection in data classification models.
Learning Outcome: *Students will analyze social implications of emerging technologies and advocate for inclusive, ethical digital innovation.*

No. of Hours:24

Module 4: IT Governance and Ethics
Scope of Module: Covers governance structures for ethical IT decision-making.
Topics: Foundations of IT corporate governance; ESG reporting frameworks and tools; Whistleblower policies and ethics channels; IT ethics boards and oversight; Compliance versus ethical leadership; IT audit trails and data governance; Transparency and accountability practices; Ethical AI governance and regulation; Cybersecurity ethics in surveillance systems; Boardroom simulation: resolving tech dilemmas; Balancing profit and public interest in IT firms; Decision-making under ethical uncertainty; ISO standards for IT governance; ESG ratings and public disclosures; Case study: Governance failure in a tech giant.
Learning Outcome: *Students will evaluate and develop governance frameworks that support ethical decision-making and stakeholder alignment in IT environments.*

No. of Hours:24

Module 5: Sustainable IT Strategy
Scope of Module: Integrates ESG principles into organizational IT planning.
Topics: Integrating ESG into IT strategy roadmaps; Stakeholder mapping and engagement models; Circular economy in IT hardware; Supply chain resilience

and transparency; Cloud sustainability strategy; Responsible tech innovation lifecycle; Designing ESG-aligned KPIs for IT; Tech sustainability scorecards and benchmarking; Policy modeling for net-zero IT operations; Carbon offsetting through tech initiatives; Funding and investing in green IT Organizational change for ESG transformation; Capstone: ESG audit of a mock IT enterprise Scenario planning for climate-resilient IT; Report writing: ESG strategy white paper.
Learning Outcome: *Students will design sustainable IT policies and long-term strategic plans incorporating ESG values for enterprise impact.*

Module	**No. of Pre-Recordings**
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Basic knowledge of IT systems and corporate social responsibility.

Pedagogy:

- ✓ Guest lectures from ESG tech leaders
- ✓ Carbon accounting workshops
- ✓ Ethics hackathons
- ✓ ESG report analysis

Certifications Recommended:

1. Certified Green IT Professional (GITP)
2. SASB's ESG Fundamentals

Readings:

- "Green IT for Sustainable Business Practice" by Mark O'Neill
- "AI Ethics" by Mark Coeckelbergh

Tools:

1. AWS Customer Carbon Footprint Tool
2. ESG reporting software (Sustainalytics)

Curriculum Development:

Sl.No	**Identify and Highlight in the Modules**
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO/PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	2	1	-	3	1	-	2	1	2
CO2	3	2	1	2	3	1	-	2	1
CO3	1	3	2	-	2	3	1	1	3
CO4	2	1	3	2	1	2	3	-	2
CO5	3	2	2	3	2	1	2	2	3
Articulation	2.6	1.8	2	2.5	1.8	1.75	2	1.5	2.2

JAIN DEEMED-TO-BE UNIVERSITY | CENTRE FOR DISTANCE AND ONLINE EDUCATION

JAIN DEEMED-TO-BE UNIVERSITY | ONLINE

Semester III

Corporate Office
91/2, Dr. A N Krishna Rao Road V V Puram,
Bengaluru - 560 004
P: +91 80 4343 1000

Centre for Distance & Online Education
319, 25th Main, 17th Cross,
J P Nagar 6th Phase
Bengaluru – 560 078
P: +91 80 4343 0100

Course Title:	**Artificial Intelligence and Machine Learning**
Course Code:	**251VMT0C301**
Semester:	**III**
Credits:	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides MCA students with an industry-relevant and application-driven exploration of Artificial Intelligence (AI) and Machine Learning (ML). The syllabus is designed to offer a progressive and logical learning path from AI fundamentals to advanced ML techniques and deployment strategies. The course also emphasizes ethical, explainable, and production-ready AI systems through a mix of practical labs, case studies, and modern tools.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO 1	Understand the evolution and fundamentals of Artificial Intelligence and Machine Learning.	L1
CO 2	Apply advanced ML algorithms for supervised, unsupervised, and semi-supervised learning tasks.	L3
CO 3	Analyze deep learning models and reinforcement learning strategies.	L4

CO 4	Evaluate models for bias, interpretability, and effectiveness across varied datasets.	L5
CO 5	Design and deploy scalable, explainable, and secure AI/ML applications.	L3

Course Modules

No. of Hours:24

Module 1: Foundations of AI and ML
Scope of Module: Establishes core concepts of AI and ML, setting the stage for in-depth exploration of models, learning paradigms, and industry practices.
Topics: AI and ML definitions and key differences, history and evolution of AI, AI vs ML vs DL, symbolic vs sub-symbolic AI, types of ML (supervised, unsupervised, reinforcement), ML lifecycle and pipeline stages, CRISP-DM methodology, introduction to datasets and data types, common data challenges, bias and fairness in ML, ML vs traditional programming, use cases in various industries, recent trends in AI (Generative AI, LLMs), AI ethics overview.
Learning Outcome: *Students will be able to articulate core AI/ML definitions, classify learning types, and describe modern trends and challenges.*

No. of Hours:24

Module 2: Advanced Supervised Learning and Ensemble Models
Scope of Module:

Covers core and advanced supervised learning techniques including ensemble strategies and hyperparameter tuning.
Topics: Linear and logistic regression with regularization (Ridge, Lasso), decision trees, random forest, support vector machines (SVM), KNN classification, Naive Bayes, model evaluation metrics (accuracy, precision, recall, F1, ROC-AUC), model selection and validation strategies (train/test split, cross-validation), bias-variance tradeoff, grid search and random search, ensemble learning (bagging, boosting), introduction to XGBoost, LightGBM, and CatBoost, multi-output and multiclass classification, managing imbalanced datasets (SMOTE, undersampling), SHAP for tree models.
Learning Outcome: *Students will be able to apply and evaluate supervised learning and ensemble methods to solve classification and regression tasks.*

No. of Hours:24

Module 3: Unsupervised Learning, Dimensionality Reduction, and Anomaly Detection
Scope of Module: Explores a wide range of unsupervised techniques and introduces preprocessing and anomaly detection as critical skills.
Topics: Clustering algorithms: k-means, DBSCAN, hierarchical, Gaussian Mixture Models (GMM), elbow method and silhouette score, association rule mining (Apriori, FP-Growth), dimensionality reduction techniques: PCA, t-SNE, UMAP, feature selection vs extraction, feature importance, anomaly detection using Isolation Forest, LOF, one-class SVM, preprocessing: encoding, scaling,

normalization, dealing with missing values, pipeline creation using scikit-learn, unsupervised model evaluation strategies.
Learning Outcome: -*Students will be able to perform unsupervised learning, reduce dimensionality for visualization, and detect data anomalies effectively.*

No. of Hours:24

Module 4: Deep Learning and Reinforcement Learning
Scope of Module: Introduces deep neural networks and reinforcement learning paradigms for intelligent decision making.
Topics: Artificial Neural Networks (ANN), forward and backward propagation, activation functions (ReLU, sigmoid, tanh), loss functions and optimizers, CNNs for image processing, RNNs, LSTMs for sequential data, autoencoders and applications, GANs and generative models, overview of BERT and Transformer models, introduction to reinforcement learning, Q-learning and policy gradients, environment-reward-agent loop, OpenAI Gym, applications in recommendation, robotics, gaming.
Learning Outcome: -*Students will be able to build and analyze neural networks and apply RL for decision-making scenarios.*

No. of Hours:24

Module 5: AI/ML Model Deployment, MLOps, and Explainability

Scope of Module: Focuses on end-to-end AI/ML model deployment, production readiness, monitoring, and responsible AI practices.
Topics: Deployment using Flask, FastAPI, Streamlit, Docker containerization, introduction to Kubernetes for ML, CI/CD pipelines in ML, MLflow and model versioning, explainable AI tools (SHAP, LIME, ELI5), fairness audits and bias detection tools, logging and monitoring models in production, introduction to cloud ML platforms (AWS SageMaker, GCP Vertex AI), securing ML models and data, case studies from industry, best practices in production ML systems. ***Learning Outcome: -****Students will be able to deploy, explain, and monitor AI/ML systems using modern DevOps practices and frameworks*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Corporate Office
91/2, Dr. A N Krishna Rao Road V V Puram,
Bengaluru - 560 004
P: +91 80 4343 1000

Centre for Distance & Online Education
319, 25th Main, 17th Cross,
J P Nagar 6th Phase
Bengaluru – 560 078
P: +91 80 4343 0100

Prerequisites:

- Strong understanding of Python programming, statistics, and data structures.

Pedagogy / Teaching Methodology:

- ✓ Project-based learning with datasets from real-world domains
- ✓ Lab-based exercises using Jupyter and Google Colab
- ✓ Mini case studies and research paper reviews
- ✓ Live demonstrations and group activities
- ✓ Tool walkthroughs and guided deployment

Certificate / Value Added Courses Recommended:

1. Machine Learning Engineering for Production (MLOps) – Coursera
2. Deep Learning Specialization – Coursera
3. AI for Everyone – Coursera
4. Google ML Crash Course – https://developers.google.com/machine-learning/crash-course

Suggested Readings:

- Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, O'Reilly, 2022
- Ian Goodfellow, Deep Learning, MIT Press, 2016
- Sutton & Barto, Reinforcement Learning: An Introduction, MIT Press, 2018
- Kelleher, D. J., Data Science Ethics, MIT Press
- Relevant research from IEEE, ACM, and arXiv.org

Recommended Software/Tools:

1. Python, Scikit-learn, TensorFlow, Keras, PyTorch

2. Google Colab / JupyterLab, MLflow, Optuna
3. Docker, FastAPI, Streamlit, AWS SageMaker, GCP Vertex AI
4. SHAP, LIME, OpenAI Gym

Curriculum Development:

Sl.No	**Identify and Highlight in the Modules**
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO/PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	1	2	-	1	1	2
CO2	3	2	3	2	2	1	-	2	2
CO3	2	3	2	3	2	1	2	-	2
CO4	2	2	2	-	2	3	2	2	2
CO5	3	3	3	2	3	2	2	3	3
Articulation	2.6	2.4	2.2	2	2.2	1.75	1.75	2	2.2

Course Title:	Python Programming
Course Code:	251VMT0C302
Semester:	**III**
Credits:	**4**
Course Type	Core Course
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides MCA students with foundational to advanced skills in Python programming, blending basic concepts with real-world applications. It encompasses control structures, functions, OOP, file operations, exception handling, unit testing, web development, API integration, and data analysis. Special emphasis is placed on writing efficient, modular, and scalable Python code with exposure to industry-grade tools and frameworks.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO	**Course Outcome**	**BTL**
CO1	Understand Python syntax, control flow, data structures, and functions.	L1
CO2	Apply OOP principles and modular coding in real-world applications.	L3
CO3	Analyze problems and implement file, error handling, and unit testing.	L4

CO4	Evaluate web-based interactions, APIs, and database connectivity using Python.	L5
CO5	Develop interactive data-driven applications using Pandas, NumPy, and visualization libraries.	L3

Course Modules

No. of Hours:24

Module 1: Python Basics, Syntax, and Control Flow
Scope of Module: Introduces core elements of Python such as syntax, variables, data types, operators, and control flow.
Topics: Installing Python and IDE setup, writing and executing Python scripts, identifiers, keywords, variables, constants, data types, type conversion, operators and precedence, conditional statements (if, elif, else), iterative control (for, while), break and continue, pass statement, nested loops, input/output handling, formatted string literals (f-strings), Python coding style (PEP8).
Learning Outcome: *Students will be able to write structured Python scripts using control flow constructs and input/output operations.*

No. of Hours:24

Module 2: Functions, Collections, Iterators, and Modules
Scope of Module:

Covers function design, advanced collection operations, and module creation for reusable code.
Topics: Defining and invoking functions, argument types (*args, **kwargs), scope and lifetime, recursion, anonymous functions (lambda), Python built-in functions, comprehension (list, dict, set), advanced list methods, tuples and unpacking, sets and operations, dictionaries and operations, iterators and generators, creating custom modules, importing built-in modules (math, sys, os, datetime, calendar, random), packaging with pip and venv.
Learning Outcome: *Students will be able to modularize programs using custom and built-in Python functions and libraries.*

No. of Hours:24

Module 3: Object-Oriented Programming, File I/O, and Exception Handling
Scope of Module: Explores OOP principles, file operations, and robust exception handling techniques.
Topics: Classes and objects, class and instance variables, methods, constructor and destructor, inheritance and types, method overriding, polymorphism, encapsulation, use of __str__, __repr__, __len__, working with text/binary/CSV files, reading and writing files, context manager and file closing, exception types and hierarchy, try-except-finally, raising and handling custom exceptions, logging module basics, assertions for debugging.

Corporate Office
91/2, Dr. A N Krishna Rao Road V V Puram,
Bengaluru - 560 004
P: +91 80 4343 1000

Centre for Distance & Online Education
319, 25th Main, 17th Cross,
J P Nagar 6th Phase
Bengaluru – 560 078
P: +91 80 4343 0100

Learning Outcome: *Students will be able to design OOP-based applications with efficient file handling and error management*

No. of Hours:24

Module 4: Testing, Automation, and Database Connectivity
Scope of Module: Provides practices in automated testing, scripting for task automation, and integrating Python with databases.
Topics: Writing test cases using unittest and pytest, test discovery and test suites, test-driven development, automation with shutil and subprocess, working with file system using os and pathlib, scheduling tasks, introduction to SQLite and PostgreSQL, executing queries from Python using sqlite3 and psycopg2, performing CRUD operations, using context managers with databases, parameterized queries to prevent SQL injection.
Learning Outcome: *Students will be able to automate tasks and integrate databases securely in Python programs.*

No. of Hours:24

Module 5: Web Development, APIs, and Data Science Libraries
Scope of Module: Bridges Python to web and data science fields using APIs, Flask, and data analysis tools

Topics:
Flask framework structure and routing, GET and POST methods, HTML rendering with Jinja2 templates, form data processing, REST API consumption using requests, OpenWeatherMap API integration example, Pandas: dataframes, merging, filtering, groupby, NumPy: array creation and operations, Matplotlib/Seaborn for plotting, exporting data to Excel/CSV, introduction to Streamlit for interactive dashboards, mini project combining Flask and Pandas.
Learning Outcome: *Students will be able to develop and deploy data-centric and interactive Python web applications*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Basic logic building, familiarity with any programming language is helpful.

Pedagogy / Teaching Methodology:

- ✓ Concept-based demonstrations using Jupyter and IDEs
- ✓ Mini assignments and use-case-driven labs

- ✓ Group activities and debugging challenges
- ✓ Integration of GitHub for version control practices
- ✓ End-of-module review quizzes and projects

Certificate / Value Added Courses Recommended:

1. Python for Everybody – Coursera
2. Python OOP and Flask Projects – Udemy
3. Automate the Boring Stuff with Python – Online Book
4. Introduction to Data Science in Python – Coursera

Suggested Readings:

- Mark Lutz, Learning Python, O'Reilly
- Eric Matthes, Python Crash Course, No Starch Press
- Luciano Ramalho, Fluent Python, O'Reilly
- Python Documentation – https://docs.python.org

Recommended Software/Tools:

1. Python 3.x, PyCharm, VS Code
2. JupyterLab, Google Colab
3. Flask, Pandas, NumPy, Matplotlib, Streamlit
4. GitHub, SQLite, PostgreSQL

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO/PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	2	1	2	1	-	1	2
CO2	3	2	3	2	3	1	2	2	3
CO3	3	3	-	3	2	-	2	2	2
CO4	2	3	3	3	3	2	2	2	-
CO5	3	3	2	2	3	2	2	3	3
Articulation	2.8	2.6	2.5	2.2	2.6	1.7	2	2	2.5

Course Title:	**Cyber Threat Intelligence**
Course Code:	**251VMT3S303**
Semester:	**III**
Credits:	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides insights into threat intelligence gathering, analysis, and dissemination to support proactive cybersecurity decisions. Students will explore tactical, operational, and strategic intelligence disciplines.

Course Outcomes (COs):

CO	Course Outcome	BTL
CO1	Understand the intelligence cycle and its application in cybersecurity	L1
CO2	Collect and analyze indicators of compromise (IOCs) and threat data	L3
CO3	Correlate threat intelligence with current cybersecurity incidents	L4
CO4	Evaluate different sources and formats of threat intelligence	L5
CO5	Design actionable threat intelligence strategies and reports	L5

Course Modules

No. of Hours:24

Module 1: Introduction to Threat Intelligence
Scope of Module: Foundations of cyber threat intelligence including concepts and the intelligence lifecycle.
Topics: Threat Intelligence Overview, Tactical vs Strategic Intelligence, Intelligence Lifecycle, Collection Management, Requirements Definition, Data Sources, Threat Actor Profiling, Open Source Intelligence, Indicators of Compromise, Attribution Challenges, Analysis Methods, Report Writing, Operational Intelligence, Dissemination Techniques, Use Case Mapping
Learning Outcome: *Students will be able to define and apply the intelligence lifecycle to cybersecurity operations.*

No. of Hours:24

Module 2: Data Collection and Aggregation
Scope of Module: Focuses on automated and manual collection methods of threat intelligence.
Topics: OSINT Tools, Social Media Intelligence, Pastebin Monitoring, Dark Web Analysis, Malware Sandbox Feeds, Threat Intelligence Platforms, Feed Normalization, STIX/TAXII Protocols, IOC Enrichment, YARA Rules, Threat Intelligence Sharing, ISAC Participation, Data Validation, Indicator Confidence Levels, Metadata Collection

Learning Outcome: *Students will be able to collect and correlate threat data from diverse sources*

No. of Hours: 24

Module 3: Threat Analysis and Adversary Tactics
Scope of Module: Analyzing threat actor behavior using frameworks and reverse engineering data.
Topics: Adversary TTPs, MITRE ATT&CK Matrix, Diamond Model, Kill Chain Analysis, Campaign Tracking, APT Profiles, TTP Mapping, Behavioral Patterns, Malware Classification, Exploit Kits, Exploit Trends, Threat Emulation, Phishing Kit Analysis, Fileless Malware, Cyber Crime Groups
Learning Outcome: *Students will be able to assess adversary capabilities using analytical frameworks*

No. of Hours: 24

Module 4: Operationalizing Threat Intelligence
Scope of Module: Using threat intelligence to support security controls and decision-making.
Topics: SIEM Integration, Alert Enrichment, Threat Hunting, IOC Matching, Dashboard Visualization, TI Automation, Response Prioritization, Security Control Tuning, Anomaly Correlation, Honeypot Intelligence, Case Study Usage, IOC Lifespan, Automation Tools, Threat Scenario Mapping, Playbook Updates

Learning Outcome: *Students will be able to implement threat intelligence in cybersecurity operations.*

No. of Hours: 24

Module 5: Strategic Threat Intelligence and Case Studies

Scope of Module:

Developing long-term threat forecasts and reviewing global threat reports.

Topics:

Threat Trend Forecasting, Sector-specific Threat Analysis, Cyber Warfare Trends, Nation-state Threats, Threat Intelligence KPIs, Maturity Models, Threat Landscape Reports, Cybercrime Economics, Emerging Threats, AI in Threat Intelligence, Case Study: WannaCry, Case Study: NotPetya, Case Study: Colonial Pipeline, Strategic Reporting, Executive Briefings

***Learning Outcome*:** *Students will be able to create strategic threat intelligence assessments with long-term value.*

Module	**No of Pre Recordings**
1	**10**
2	**11**
3	**12**
4	**12**
5	**11**

Note: Min no of pre recordings 10.

Prerequisites:

- Basic knowledge of networking, cybersecurity concepts, and incident response.

Pedagogy / Teaching Methodology:

- ✓ Interactive Lectures
- ✓ Case-Based Learning
- ✓ Tool-Based Labs
- ✓ Group Projects on Threat Reports
- ✓ Industry Whitepaper Discussions

Certificate / Value Added Courses Recommended:

1. Certified Threat Intelligence Analyst (CTIA) by EC-Council
2. IBM Cybersecurity Analyst Path (https://www.coursera.org/professional-certificates/ibm-cybersecurity-analyst)
3. Threat Intelligence Fundamentals by Mandiant (https://www.mandiant.com/resources)

Suggested Readings:

Books:

- McMillan, J., *Cyber Threat Intelligence*, Packt Publishing, 2022
- Ratnayake, R., *Practical Threat Intelligence and Data-Driven Threat Hunting*, Apress, 2021
- Stevens, R., *Cybersecurity and Threat Intelligence*, Wiley, 2020

Online Articles / Journals:

- Journal of Cybersecurity & Information Integrity
- Threat Intelligence Reports from FireEye, Cisco Talos, and MITRE

Web-based Resources:

- https://attack.mitre.org/
- https://unit42.paloaltonetworks.com/
- https://www.recordedfuture.com/blog

Recommended Software/Tools:

1. MISP (Malware Information Sharing Platform)
2. OpenCTI

3. IBM X-Force Exchange
4. ThreatConnect
5. Splunk with CTI Add-ons
6. VirusTotal
7. YARA
8. MITRE ATT&CK Navigator

Curriculum Development:

Sl.No	**Identify and Highlight in the Modules**
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO/PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	2	1	2	1	1	1	2
CO2	2	3	3	2	3	1	2	2	2
CO3	2	3	3	3	2	1	2	2	3
CO4	2	3	3	3	3	2	2	2	3
CO5	1	3	3	2	3	2	2	3	3
Articulation	2	2.8	2.8	2.2	2.6	1.4	1.8	2	2.6

Course Title:	**Open Elective**
Course Code:	
Semester:	**III**
Credits:	**4**
Course Type	**Open Elective**
Learning Hours	120
Live Sessions	**12 hours**

Semester IV

Corporate Office
91/2, Dr. A N Krishna Rao Road V V Puram,
Bengaluru - 560 004
P: +91 80 4343 1000

Centre for Distance & Online Education
319, 25th Main, 17th Cross,
J P Nagar 6th Phase
Bengaluru – 560 078
P: +91 80 4343 0100

Course Title:	**Ethical Hacking**
Course Code:	**251VMT3S401**
Semester:	**IV**
Credits:	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course explores advanced methodologies and tools used by ethical hackers to assess and secure systems against cyber threats. Students will delve into modern attack strategies, countermeasures, and industry-grade ethical hacking tools for protecting networks, applications, and data systems.

Course Outcomes (COs):

CO	**Course Outcome**	**BTL**
CO1	Understand the concepts and need for ethical hacking.	L1
CO2	Identify and analyze different types of security threats and vulnerabilities.	L4
CO3	Apply various tools and techniques for penetration testing.	L3
CO4	Evaluate network and system security to detect and fix vulnerabilities.	L5
CO5	Design ethical hacking strategies and countermeasures for real-world scenarios.	L5

Course Module

No. of Hours:24

Module 1: Ethical Hacking Frameworks and Methodologies
Scope of Module: Covers deep dive into frameworks like OWASP, PTES, and NIST. Emphasizes understanding the structure and scope of ethical hacking assessments.
Topics: Advanced Reconnaissance, OSINT Frameworks, Vulnerability Identification Workflows, Threat Modelling, OWASP Testing Guide v4, PTES Technical Guidelines, NIST SP 800-115, Target Profiling, Legal Compliance Considerations, Ethics in Red Teaming, Pre-engagement Interactions, Risk Assessment Matrix, Reporting Protocols, Rules of Engagement (RoE), NDA & Authorization Forms
Learning Outcome:*Students will be able to implement standard frameworks for ethical hacking assessments.*

No. of Hours:24

Topics:
Advanced Network Scanning, Deep Packet Inspection, Protocol Fuzzing, Network Sniffers, MITM Attacks, VLAN Hopping, ARP Poisoning, DNS Spoofing, Metasploit Scripting, Pivoting Techniques, Custom Payload Creation, Exploit Development Basics, Reverse Shells, Lateral Movement, Evasion Techniques
Learning Outcome:*Students will be able to exploit network vulnerabilities and apply mitigation strategies.*

No. of Hours:24

Module 3: Web Application Hacking
Scope of Module: Hands-on focus on exploiting vulnerabilities in modern web applications using tools like Burp Suite, SQLMap, and custom scripts.
Topics: Advanced SQL Injection, DOM-based XSS, CSRF Exploitation, Burp Suite Automation, Authentication Bypass, Session Hijacking, IDOR Attacks, Business Logic Bypass, API Hacking, Web Shell Deployment, SSRF Attacks, RCE in Web Apps, HTML5 Attacks, Broken Access Control, WAF Evasion
Learning Outcome: *Students will be able to identify and exploit web application vulnerabilities in a controlled environment.*

No. of Hours:24

Module 4: System and Wireless Exploitation
Scope of Module: Advanced exploitation techniques on operating systems and wireless technologies. Includes custom malware deployment and rootkit analysis.
Topics:
Linux Privilege Escalation, Windows Exploitation, Kernel Exploits, Fileless Malware, Advanced Ransomware Techniques, Post-Exploitation Enumeration, Active Directory Attacks, PowerShell Abuse, Wireless Packet Injection, WPA3 Attacks, Evil Twin Attacks, Bluetooth Exploitation, Rogue AP Setup, Mac Spoofing, Credential Dumping.

Learning Outcome: *Students will be able to carry out ethical attacks on system and wireless components for vulnerability analysis.*

No. of Hours:24

Module 5: Red Team Simulations and Case Studies
Scope of Module: Applies all knowledge in a red team simulation environment; reviews real-world attack case studies and uses professional reporting formats.
Topics: Red Teaming vs Blue Teaming, Purple Team Operations, Chain of Exploits, Report Writing Standards, Mitre ATT&CK Framework, Incident Simulation, Zero-day Case Studies, Nation-state Attacks, APTs and TTPs, Post-exploitation Cleanup, Custom Rules in SIEM, Forensics in Red Team, Real-world Breach Analysis, Continuous Monitoring Techniques, Ethical Guidelines in Simulations
Learning Outcome: *Students will be able to simulate and analyze real-world cyberattacks with ethical intent.*

Module	**No of Pre-Recordings**
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10

Prerequisites :

- Basic knowledge of computer networks, operating systems, and programming.

Pedagogy / Teaching Methodology:

- ✓ Lectures, hands-on labs, case studies, group discussions, online demonstrations.

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. Certified Ethical Hacker (CEH) - https://www.eccouncil.org/ceh/
2. TryHackMe Ethical Hacking Course - https://tryhackme.com/

Suggested Readings:

Book

- Michael T. Simpson, Kent Backman, James Corley, 'Hands-On Ethical Hacking and Network Defense', Cengage Learning, 2020

Online Articles, Journals, and Whitepapers –

- OWASP Top 10 Web Application Security Risks - https://owasp.org/www-project-top-ten/
- NIST Cybersecurity Framework - https://www.nist.gov/cyberframework

Web-based Resources –

- Cybrary - https://www.cybrary.it/
- Hacker101 - https://www.hacker101.com/

Recommended Software/Tools: (If applicable)

1. Nmap, Wireshark, Metasploit, Burp Suite, John the Ripper, Nikto, Aircrack-ng, Kali Linux

Curriculum Development:

Sl.No	**Identify and Highlight in the Modules**
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO/PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	2	2	2	3	2	1	1	2	2
CO2	2	3	2	3	3	2	2	1	3
CO3	2	3	3	2	2	2	1	3	2
CO4	2	2	2	3	2	2	2	2	3
CO5	2	3	2	3	3	3	2	3	3
Articulation	2	2.6	2.2	2.8	2.4	2	1.6	2.2	2.6

Course Title:	**Defensive Cyber Security Technologies**
Course Code:	**251VMT3S402**
Semester:	**IV**
Credits:	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course explores advanced methods and technologies used in cyber defense, including threat detection systems, endpoint protection, intrusion prevention, and incident response. It aims to prepare learners for real-time defense implementation in modern enterprise environments.

Course Outcomes (COs):

CO	Course Outcome	BTL
CO1	Understand the principles and technologies of cyber defense systems	L1
CO2	Analyze different detection and response mechanisms for cyber threats	L3
CO3	Implement endpoint and network-based defensive strategies	L4
CO4	Evaluate and enhance system defenses against modern attack vectors	L5
CO5	Design comprehensive defensive architecture and response plans	L5

Course Modules

No. of Hours:24

Module 1: Cyber Defense Overview and Architecture
Scope of Module: Introduction to defense-in-depth strategies, architectures, and operational models.
Topics: Defense-in-Depth, Layered Security, Zero Trust Architecture, Cyber Kill Chain, Defense Frameworks (MITRE ATT&CK), Enterprise Architecture Design, Security Domains, Network Segmentation, Access Control Models, Security Zoning, Deception Technology, Breach Containment, Asset Protection Levels, Network Trust Levels, Incident Preparedness
Learning Outcome: *Students will be able to describe and model advanced cyber defense architectures*

No. of Hours:24

Module 2: Network and Endpoint Security
Scope of Module: Covers advanced configuration and deployment of network and host protection tools.
Topics: Next-gen Firewalls, Network Access Control, NAC Policies, IDS/IPS Deployment, Endpoint Detection and Response, SIEM Integration, Host-based Firewalls, Antivirus and Anti-malware, Endpoint Hardening, Registry Monitoring, Kernel-mode Monitoring, USB Control, Log Aggregation, Secure Baselines, File Integrity Monitoring

Learning Outcome: *Students will be able to deploy and tune security mechanisms across hosts and networks.*

No. of Hours:24

Module 3: Intrusion Detection and Anomaly Detection Systems
Scope of Module: Techniques and tools for monitoring and identifying anomalies in system behavior.
Topics: Signature-based Detection, Anomaly-based Detection, Heuristic Analysis, Hybrid IDS, Snort, Suricata, Zeek, NetFlow Analysis, Protocol Anomalies, Behavioral Analytics, DNS Monitoring, Custom Rule Writing, False Positive Reduction, Alert Tuning, Machine Learning IDS
Learning Outcome**:** *Students will be able to configure and analyze output from modern IDS tools.*

No. of Hours:24

Module 4: Incident Detection, Response, and Automation
Scope of Module: Focuses on operational response to threats using tools and playbooks.
Topics: SOC Operations, SIEM Use Cases, Incident Playbooks, Automation with SOAR, MITRE ATT&CK Mapping, Phishing Response, Malware Containment, Credential Theft Response, Triage Process, Chain of Custody, Escalation Matrix, Threat Categorization, Forensics Collection, Incident Report Writing, Post-Incident Review

Learning Outcome: *Students will be able to respond to incidents using automated and manual techniques.*

No. of Hours:24

Module 5: Cyber Defense Strategies and Case Studies
Scope of Module: Real-world defensive implementations and evaluations of major incidents.
Topics: Nation-state Threat Defenses, Insider Threat Defense, DDoS Mitigation, Defense Metrics, Security Benchmarking, Cloud Defense Models, Endpoint Isolation, Microsegmentation, Deception-based Defenses, SOC Maturity Models, Case Study: SolarWinds Response, Case Study: Log4j Containment, Defense Test Planning, Simulation Exercises, Lessons Learned
Learning Outcome: *Students will be able to formulate and evaluate defense strategies against real-world threats.*

Module	**No of Pre-Recordings**
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10

Prerequisites:

- Basic knowledge of computer networks, operating systems, and programming.

Pedagogy / Teaching Methodology:

- ✓ Lectures, hands-on labs, case studies, group discussions, online demonstrations.

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. Offensive Security Beginner Path - https://www.offensive-security.com/

Suggested Readings:

Book

- William Stallings, 'Computer Security: Principles and Practice', Pearson, 2018

Online Articles, Journals, and Whitepapers –

- OWASP Top 10 Web Application Security Risks - https://owasp.org/www-project-top-ten/
- NIST Cybersecurity Framework - https://www.nist.gov/cyberframework

Web-based Resources –

- Cybrary - https://www.cybrary.it/
- Hacker101 - https://www.hacker101.com/

Recommended Software/Tools: (If applicable)

1. Nmap, Wireshark, Metasploit, Burp Suite, John the Ripper, Nikto, Aircrack-ng, Kali Linux

Curriculum Development:

Sl.No	**Identify and Highlight in the Modules**

1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO/PO Mapping Table

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**
CO1	3	3	2	2	3	2	1	2	2
CO2	2	3	3	3	2	2	2	2	2
CO3	3	3	2	2	3	2	1	3	2
CO4	2	2	3	3	2	3	1	2	3
CO5	3	3	3	2	3	3	2	2	3
Articulation	2.6	2.8	2.6	2.4	2.6	2.4	1.4	2.2	2.4

Course Title:	**Vulnerability Analysis**
Course Code:	**251VMT3S403**
Semester:	**IV**
Credits:	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

The course emphasizes identifying and analyzing security weaknesses in systems and applications. It includes scanning tools, risk evaluation, and report generation, preparing students for securing systems proactively.

Course Outcomes (COs):

CO	**Course Outcome**	**BTL**
CO1	Understand types and causes of system vulnerabilities.	L1
CO2	Use vulnerability scanning tools effectively.	L3
CO3	Analyze vulnerability scan results and assess risk levels.	L4
CO4	Recommend remediation strategies for identified vulnerabilities.	L5
CO5	Design a vulnerability management process.	L5

Course Modules

No. of Hours:24

Module 1: Introduction to Vulnerability Assessment
Scope of Module: Covers basic concepts, types, and need for vulnerability analysis.
Topics: Vulnerability vs. threats, Common vulnerabilities, CVE, CVSS, OWASP Top 10, NVD database, Vulnerability lifecycle, Risk management basics, Security misconfigurations, Buffer overflows, Broken authentication, Vulnerability assessment methodology, Discovery process, Exploit vs. exposure, Patch management process
***Learning Outcome**: Students will identify types of vulnerabilities and understand their risks*

No. of Hours:24

Module 2: Vulnerability Scanning Tools and Techniques
Scope of Module: Hands-on exposure to various scanning tools.
Topics: Nessus, OpenVAS, Nikto, Burp Suite Scanner, Qualys, Acunetix, Nmap scripts, Metasploit scanners, Credentialed vs. non-credentialed scans, Authenticated scanning, Web vulnerability scanning, Host discovery, Service enumeration, Vulnerability detection, Tool comparison
***Learning Outcome:** Learners will operate scanning tools and analyze output.*

No. of Hours:24

Module 3: System and Network Vulnerabilities
Scope of Module: Analyzes OS, applications, and network-level flaws.
Topics: OS vulnerabilities, Privilege escalation, Kernel flaws, Patch delays, Third-party applications, Service vulnerabilities, Web server misconfiguration, Default settings, Credential leaks, Port security issues, Router vulnerabilities, Switch security, SNMP weaknesses, Unsecured APIs, Wireless protocol flaws
Learning Outcome:*Students will recognize platform-specific vulnerabilities and implications*

No. of Hours:24

Module 4: Vulnerability Reporting and Risk Assessment
Scope of Module: Methods to evaluate impact and communicate findings.
Topics: Risk matrix, Asset classification, Severity rating, Exploitability analysis, Business impact, Remediation timeline, Executive summaries, Technical reporting, Security recommendations, Sample report formats, Vulnerability disclosure, Responsible reporting, Dashboard creation, Data visualization, Compliance requirements
Learning Outcome:L*earners will create actionable vulnerability reports with prioritization*

No. of Hours:24

Module 5: Vulnerability Management Programs
Scope of Module: Design and implement an enterprise-wide vulnerability management process.
Topics: Policy creation, Periodic scanning, Patch management integration, Change management, Scan automation, Asset inventory, Continuous monitoring, Tool orchestration, Metrics and KPIs, Regulatory needs, Compliance checks, VA in DevOps, Cloud vulnerability scanning, Reporting frequency, Case studies
Learning Outcome:*Students will design a comprehensive and compliant vulnerability management system.*

Module	**No of Pre-Recordings**
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10

Prerequisites:

- Basic knowledge of computer networks, operating systems, and programming.

Pedagogy / Teaching Methodology:

- Lectures, hands-on labs, case studies, group discussions, online demonstrations.

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. TryHackMe Ethical Hacking Course - https://tryhackme.com
2. Offensive Security Beginner Path - https://www.offensive-security.com/

Suggested Readings:

Books –

- Michael T. Simpson, Kent Backman, James Corley, 'Hands-On Ethical Hacking and Network Defense', Cengage Learning, 2020
- William Stallings, 'Computer Security: Principles and Practice', Pearson, 2018

Online Articles, Journals, and Whitepapers –

- OWASP Top 10 Web Application Security Risks - https://owasp.org/www-project-top-ten/
- NIST Cybersecurity Framework - https://www.nist.gov/cyberframework

Web-based Resources –

- Cybrary-https://www.cybrary.i
- Hacker101 - https://www.hacker101.com/

Recommended Software/Tools: (If applicable)

1. **Nmap, Wireshark, Metasploit, Burp Suite, John the Ripper, Nikto, Aircrack-ng, Kali Linux**

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO/PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	2	3	2	1	1	2	2
CO2	3	3	2	3	3	2	1	3	3
CO3	2	3	3	2	2	2	2	2	2
CO4	2	2	3	3	2	3	2	2	3
CO5	3	3	3	2	3	3	2	3	3
Articulation	2.6	2.6	2.6	2.6	2.4	2.4	1.6	2.2	2.6

Course Title:	**Penetration Testing**
Course Code:	**251VMT3S404**
Semester:	**IV**
Credits:	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course emphasizes in-depth penetration testing techniques for modern IT systems, including network, web, wireless, and cloud infrastructure. Learners will explore red team operations, tool development, and test execution strategies.

Course Outcomes (COs):

CO	**Course Outcome**	**BTL**
CO1	Understand advanced penetration testing strategies and tools	L1
CO2	Apply various techniques to conduct real-world penetration tests	L3
CO3	Analyze systems and networks for deep-seated vulnerabilities	L4
CO4	Evaluate the effectiveness of security controls through exploits	L5
CO5	Develop and customize penetration testing toolkits and scripts	L5

Course Modules

No. of Hours:24

Module 1: Penetration Testing Frameworks
Scope of Module: Overview of methodologies and penetration testing lifecycle for networks and applications.
Topics: PTES Lifecycle, NIST SP 800-115, OWASP WSTG, Information Gathering, Attack Modeling, Exploitation Strategy, Pen Test Scoping, Target Validation, Engagement Management, Documentation Standards, Tools Selection, Pre-test Communication, Defining Boundaries, Client Expectations, Post-test Review
Learning Outcome: *Students will be able to execute structured penetration tests using accepted frameworks.*

No. of Hours:24

Module 2: Internal Network Penetration Testing
Scope of Module: Covers lateral movement, privilege escalation, and service exploitation in enterprise environments.
Topics: Internal Reconnaissance, Port Redirection, Active Directory Exploits, Kerberoasting, NTLM Relay, Token Impersonation, Admin Shares Exploitation, DNS Zone Transfers, NetBIOS Enumeration, Pass-the-Hash, Command & Control, Credential Extraction, AV Evasion, Privilege Escalation Scripting, Service Misconfiguration

Learning Outcome: *Students will be able to identify and exploit internal system weaknesses*

No. of Hours:24

Module 3: External and Perimeter Testing
Scope of Module: Testing public-facing infrastructure including firewalls, VPNs, and DMZ systems.
Topics: External Recon Tools, Subdomain Enumeration, Web and Email Servers Testing, Cloud Perimeter Assessments, SSL Testing, Firewall Rule Evasion, Open Ports Abuse, Email Spoofing, VPN Brute Force, Reverse Proxy Attacks, Web Cache Poisoning, API Endpoint Fuzzing, SMB Exploitation, Remote Exploits, Perimeter Defense Bypass
Learning Outcome: S*tudents will be able to exploit external and perimeter systems using advanced tools.*

No. of Hours:24

Module 4: Web, Mobile, and Cloud Pen Testing
Scope of Module: Advanced penetration techniques applied to web, mobile, and cloud platforms.
Topics: Mobile App Decompilation, Cloud Enumeration, IAM Exploits, Bucket Misconfigurations, Web App Authentication, Mobile SSL Pinning Bypass, Token Replay Attacks, Advanced XXE, Deserialization Exploits, Firebase Insecurity,

OAuth Vulnerabilities, JWT Cracking, Cloud SQL Injection, Role Escalation, Cloud Storage Abuse
Learning Outcome:*Students will be able to perform thorough testing on modern application and cloud platforms.*

No. of Hours:24

Module 5: Red Team Operations and Real-world Cases
Scope of Module: Comprehensive red teaming engagements and case analysis of real-world attacks.
Topics: Initial Access Techniques, Social Engineering Payloads, Droppers and Stagers, Red Team Infrastructure, Beacon Management, Evasion Techniques, Post-exploitation Planning, Threat Simulation, Insider Threat Modeling, Red Team Tools (Cobalt Strike, Empire), Report Crafting, Lessons Learned, Case Study: SolarWinds, Case Study: Capital One, Lessons from APT Groups
Learning Outcome*:Students will be able to simulate real-world attack scenarios using red team methodology.*

Module	**No of Pre-Recordings**
1	**10**
2	**12**
3	**10**

4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10

Prerequisites (If Any):

- Basic knowledge of computer networks, operating systems, and programming.

Pedagogy / Teaching Methodology:

- ✓ Lectures, hands-on labs, case studies, group discussions, online demonstrations.

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. Certified Ethical Hacker (CEH) - https://www.eccouncil.org/ceh/
2. TryHackMe Ethical Hacking Course - https://tryhackme.com/
3. Offensive Security Beginner Path - https://www.offensive-security.com/

Suggested Readings:

Books

- Michael T. Simpson, Kent Backman, James Corley, 'Hands-On Ethical Hacking and Network Defense', Cengage Learning, 2020
- William Stallings, 'Computer Security: Principles and Practice', Pearson, 2018

Online Articles, Journals, and Whitepapers

- OWASP Top 10 Web Application Security Risks - https://owasp.org/www-project-top-ten/
- NIST Cybersecurity Framework - https://www.nist.gov/cyberframework

Web-based Resources

- Cybrary-https://www.cybrary.it/
- Hacker101 - https://www.hacker101.com/

Recommended Software/Tools: (If applicable)

1. Nmap, Wireshark, Metasploit, Burp Suite, John the Ripper, Nikto, Aircrack-ng, Kali Linux

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO/PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	3	2	3	3	2	-	3	3
CO2	2	2	3	2	2	2	2	2	3
CO3	3	2	3	3	2	2	1	2	2
CO4	2	3	3	2	2	3	2	3	3
CO5	3	3	2	3	3	3	2	3	3
Articulation	2.6	2.6	2.6	2.6	2.4	2.4	1.75	2.6	2.8

Course Title:	**Strategic Leadership in IT**
Course Code:	**251VMT0C405**
Semester:	**IV**
Credits:	**2**
Course Type	**Core Course**
Learning Hours	60
Live Sessions	**12 hours**

Course Description:

This course explores the critical role of leadership in driving strategic IT initiatives within organizations. It focuses on leadership theories, IT governance, digital transformation strategies, risk management, and innovation leadership. Practical case studies and simulations are integrated to build decision-making skills required for aligning IT with business strategies.

Course Outcomes (Cos)

CO	**Course Outcome**	**BTL**
CO1	Understand leadership theories and their application in IT strategy formulation.	L1
CO2	Evaluate and apply IT governance frameworks and practices.	L5
CO3	Design digital transformation strategies aligned with business goals.	L6
CO4	Identify and mitigate IT-related risks through strategic leadership.	L4

CO5	Demonstrate innovation and change management skills in IT leadership contexts.	L3

Course Modules

No. of Hours: 12

Module 1: Foundations of Strategic Leadership and IT Alignment
Scope of Module: Covers the core leadership theories and IT-business alignment models.
Topics: Defining leadership and management in the context of IT; Strategic leadership competencies; Leadership styles: transformational, transactional, situational; Vision building and strategic thinking in IT; Models of IT-business alignment (Henderson & Venkatraman Model); Organizational structures and their impact on IT leadership; Emotional intelligence in IT leadership; Ethical leadership in technology environments; Leading virtual and remote IT teams; Leadership maturity models; Case Study: IT leadership challenges in global enterprises.
Learning Outcome*: *Students will be able to understand leadership theories and how to align IT strategies with broader business objectives.

No. of Hours: 12

Module 2: IT Governance and Strategic Control
Scope of Module: Focuses on frameworks and practices for effective IT governance.
Topics: Introduction to IT governance and compliance; COBIT, ITIL, and ISO/IEC 38500 frameworks; Governance of Enterprise IT (GEIT) principles; Strategic decision

rights and accountability in IT; Metrics and KPIs for IT governance; Managing outsourcing and vendor relationships; Sarbanes-Oxley Act and IT implications; Risk appetite and tolerance in IT governance; Blockchain's role in IT governance; Shadow IT and governance challenges; Audit management and IT policy development; Practical Exercise: Designing an IT governance framework for an SME.
Learning Outcome*: *Students will be able to evaluate, select, and apply IT governance frameworks to ensure strategic control and value delivery.

No. of Hours: 12

Module 3: Digital Transformation Leadership
Scope of Module: Develops skills to lead digital transformation initiatives.
Topics: Drivers of digital transformation in organizations; Role of CIOs and CTOs in transformation efforts; Technology adoption frameworks (e.g., Diffusion of Innovations); Agile and DevOps in digital leadership; Managing cultural change during transformation; Leadership in cloud, AI, and emerging technologies adoption; Case Study: Successful vs. failed digital transformation initiatives.
Learning Outcome: *Students will be able to design strategic digital transformation initiatives aligned with business innovation needs.*

No. of Hours: 12

Module 4: IT Risk Management and Strategic Decision-Making
Scope of Module: Focuses on managing strategic risks in IT leadership roles.

Topics: Understanding IT-related risks: security, operational, reputational; Risk management lifecycle in IT projects; Enterprise risk management frameworks (COSO ERM); Crisis leadership and contingency planning; Ethical considerations and cybersecurity leadership; Legal and regulatory compliance (e.g., GDPR, HIPAA); Threat modeling and strategic defense planning; Business continuity planning and IT disaster recovery leadership; Strategic handling of data breaches and incident response, Practical: IT risk assessment simulation
Learning Outcome**:** *Students will be able to analyze, manage, and mitigate IT risks strategically to ensure organizational resilience.*

No. of Hours: 12

Module 5: Innovation, Change Management, and Future of IT Leadership
Scope of Module: Covers fostering innovation and managing technological change.
Topics: Leading organizational innovation in IT; Change management models (Kotter's 8-Step, ADKAR); Talent development and IT leadership pipelines; Diversity, inclusion, and ethics in IT leadership; Future trends: Quantum Computing, Web3, and Leadership implications; Building a sustainable IT innovation culture; Intrapreneurship and fostering innovation within IT teams; Managing resistance to technological change; IT leadership in the age of Automation and RPA (Robotic Process Automation); Ethical considerations in AI and emerging tech leadership; Capstone: Strategic Leadership Plan for a tech startup.

Learning Outcome: *Students will be able to demonstrate strategic leadership skills in driving innovation, change management, and future-proofing IT organizations*

.

Module	No of Pre Recordings
1	**10**
2	**11**
3	**12**
4	**12**
5	**11**

Note: Min no of pre recordings 10.

Prerequisites :

- Basic knowledge of IT management concepts and organizational behavior.

Pedagogy / Teaching Methodology:

- ✓ Interactive case studies and leadership simulations
- ✓ Group strategy projects and role plays
- ✓ Workshops on IT governance frameworks
- ✓ Industry expert talks and guest lectures

Certificate / Value Added Courses Recommended:

1. IT Leadership Certificate by edX (MIT Sloan)
2. COBIT 2019 Foundation Certification
3. Digital Transformation Strategy by Coursera (Boston University)

Suggested Readings:

- John P. Kotter, Leading Change, Harvard Business Review Press, 2012.
- Peter Weill & Jeanne W. Ross, IT Governance: How Top Performers Manage IT Decision Rights for Superior Results, Harvard Business Review Press, 2004.
- George Westerman, Leading Digital: Turning Technology into Business Transformation, Harvard Business Review Press, 2014.

Recommended Software/Tools:

1. MS Project, Trello, Jira (for project leadership)
2. ServiceNow, SolarWinds (for IT governance)
3. Risk assessment tools (RSA Archer, RiskLens)

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools

5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO/PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	-	2	1	-	2	1
CO2	3	3	2	1	2	1	-	2	1
CO3	2	3	3	2	3	2	1	3	2
CO4	3	2	2	2	2	1	1	2	2
CO5	2	2	2	1	3	2	1	2	2
Articulation	2.6	2.4	2	1.5	2.4	1.4	1	2.2	1.6

Course Matrix

ELECTIVE: Devops

Corporate Office
91/2, Dr. A N Krishna Rao Road V V Puram,
Bengaluru - 560 004
P: +91 80 4343 1000

Centre for Distance & Online Education
319, 25th Main, 17th Cross,
J P Nagar 6th Phase
Bengaluru – 560 078
P: +91 80 4343 0100

Elective: Devops

Semester-I

Course Code	Course Name	Course Category	CA	ESE	Credits
25VMT0C101	Mathematical Foundations for Computer Science	CC	30	70	4
25VMT0C102	Computing Concepts and Problem Solving using C	CC	30	70	4
25VMT0C103	Operating Systems	CC	30	70	4
25VMT0C104	Data Structures	CC	30	70	4
25VMT0L101	Problem Solving using C Lab	SEC	100	-	2
25VMT0L102	Data Structures Lab	SEC	100	-	2
26VMX0S105	AI Skills for Future	SEC	30	70	2
	Total No of Credits				**22**

Semester-II

Course code	Course Name	Course category	CA	ESE	Credits
25VMT0C201	Advanced Database Systems	CC	30	70	4
25VMT0C202	Java Programming	CC	30	70	4
25VMT0C203	Web Technologies	CC	30	70	4
25VMT0C204	Cloud Computing Foundations	CC	30	70	4
25VMT0L201	Advanced Database Systems Lab	SEC	100	-	2
25VMT0L202	Java Programming Lab	SEC	100	-	2
25VMT0C205	Environmental, Social, and Governance (ESG) in IT	AECC	30	70	4
	Total No of Credits				**24**

Semester-III

Course code	Course Name	Course category	CA	ESE	Credits
251VMT0C301	Artificial Intelligence and Machine Learning	CC	30	70	4
251VMT0C302	Python Programming	CC	30	70	4
251VMT2S303	Infrastructure as Code & Automation	DSE	30	70	4
251VMT0L301	AI & Machine Learning Lab	SEC	100	-	2
251VMT0L302	Python Programming Lab	SEC	100	-	2
	Open Elective	OE	30	70	4
Total No of Credits					**20**

Semester-IV

Course code	Course Name	Course category	CA	ESE	Credits
251VMT2S401	Site Reliability Engineering	DSE	30	70	4
251VMT2S402	DevSecOps	DSE	30	70	4
251VMT2S403	Microservices & Containerization	DSE	30	70	4
251VMT2S404	DevOps in AI & Big Data	DSE	30	70	4
251VMT0C405	Strategic Leadership in IT	SEC	30	70	2
251VMT0P406	Project	Thesis			6
Total No of Credits					**24**

Legend

CC	Core Course
DSE	Discipline Specific Elective
SEC	Skill Enhancement Courses
AECC	Ability Enhancement Compulsory Course
GE	Generic Elective

Semester I

Common Courses

Course Title:	**Mathematical Foundations for Computer Science**
Course Code:	**25VMT0C101**
Semester:	**I**
Credits:	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course builds a strong mathematical base essential for various domains in computer science such as algorithms, cryptography, artificial intelligence, and data science. It covers discrete structures, logic, combinatory, graph theory, and number theory. Emphasis is laid on both theoretical understanding and application to computational problems.

Course Outcomes (COs)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Comprehend the fundamentals of logic, sets, and relations used in computer science.	L3
CO2	Apply principles of combinatory and counting for solving algorithmic and real-world problems.	L3

CO3	Analyze and construct graphs, trees, and networks relevant to data structures and networking.	L4
CO4	Solve recurrence relations and use mathematical induction in algorithm design.	L3
CO5	Evaluate number theoretic and algebraic methods in cryptographic and coding applications.	L4

Course Module

No. of Hours:24

Module 1: Logic, Sets, and Relations
Scope of Module: *Provides foundational concepts in mathematical logic, set theory, and relations essential for modeling and designing algorithms.*
Topics: Propositions and logical operators, Truth tables, Tautology and contradiction, Predicates and quantifiers, Rules of inference computation, Logical equivalence, Methods of proof, Set theory basics, Venn diagrams, Set operations, Cartesian product of sets, Types of relations, Properties of relations, Equivalence and partial order relations, Closure of relations.
Learning Outcome: *Students will be able to construct logical arguments and model relationships using sets and relations.*

No. of Hours:24

Module 2: Combinatorics and Counting
Scope of Module: Focuses on combinatorial problem-solving techniques relevant to algorithm complexity and probability theory.
Topics: Basic counting principles, Addition and multiplication rules, Permutations and combinations, Pigeonhole principle, Principle of inclusion and exclusion, Binomial theorem, Pascal's triangle, Combinatorial identities, Multisets and combinations with repetition, Generating functions, Recurrence relations, Solving linear recurrence, Introduction to partitions, Derangements, Applications in probability.
Learning Outcome: *Students will be able to apply combinatorial reasoning to solve complex counting and probability problems.*

No. of Hours:24

Module 3: Graph Theory and Trees
Scope of Module: Introduces graph-based models to represent and solve problems in data structures, networks, and algorithms.
Topics: Basic concepts of graphs, Types of graphs, Graph representations, Degree of vertices, Paths and cycles, Connected components, Eulerian and Hamiltonian graphs, Planar graphs, Graph coloring, Matrix representation of graphs, Trees and their properties, Binary and n-ary trees, Tree traversal algorithms, Spanning trees, Minimum spanning tree algorithms (Kruskal, Prim)
Learning Outcome: *Students will be able to model and solve problems using graphs and trees in computer science.*

No. of Hours:24

Module 4: Recurrence Relations and Mathematical Induction
Scope of Module: Covers recurrence relations and inductive reasoning techniques used in algorithm analysis and design.
Topics: Sequences and recurrence relations, Solving homogeneous linear recurrence, Non-homogeneous recurrence, Generating functions for recurrence, Divide-and-conquer relations, Recursion trees and Master's theorem, Mathematical induction, Strong induction, Well-ordering principle, Applications in algorithm correctness, Tower of Hanoi, Binary search, Fibonacci sequence, Ackermann's function.
Learning Outcome: *Students will be able to formulate and solve recurrence relations and validate algorithms using induction.*

No. of Hours:24

Module 5: Number Theory and Algebraic Structures
Scope of Module: Explores concepts in number theory and abstract algebra crucial for cryptography, hashing, and finite state systems.
Topics: Divisibility and Euclidean algorithm, Prime numbers and factorization, Congruence relations, Modular arithmetic, Fermat's little theorem, Euler's theorem, Chinese Remainder Theorem, Greatest common divisor (GCD), Least common multiple (LCM), Rings and fields basics, Group theory introduction, Subgroups and cyclic groups, Applications in cryptography, Hashing functions, Finite automata fundamentals
Learning Outcome: *Students will be able to apply number theoretic and algebraic techniques in areas like security and finite state systems.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites (If Any):

- Basic mathematics at undergraduate level, understanding of algebra and logic.

Pedagogy / Teaching Methodology:

- ✓ Conceptual and problem-based learning
- ✓ Interactive class exercises and proofs
- ✓ Real-world applications and modeling
- ✓ Assignments and quizzes using math software.

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. Discrete Mathematics by NPTEL: https://nptel.ac.in/courses/111106086
2. Mathematics for Computer Science by MIT Open Courseware: https://ocw.mit.edu

Suggested Readings:

- Kenneth H. Rosen, Discrete Mathematics and Its Applications, McGraw-Hill, 2017
- Tremblay and Manohar, Discrete Mathematical Structures, Tata McGraw-Hill, 2001

- C.L. Liu and D.P. Mohapatra, Elements of Discrete Mathematics, McGraw-Hill, 2010
- Online resources from IEEE Xplore, Springer Link for mathematical modeling and computation

Recommended Software/Tools (If applicable):

1. MATLAB/Octave
2. SageMath
3. Python (SymPy, NetworkX)
4. GeoGebra for visualization

Curriculum Development:

Sl.No	Identify and highlight in the module
1	Local, Regional, National and Global
2	Employability, Entrepreneurship and Skill Development
3	Value Added Courses (Module)
4	Integration of Digital Learning Tools
5	Crossing Cutting Edges (Ethics, Gender Equality, Sustainability and Environmental concerns)
6	Indian Knowledge System

CO/PO Mapping Table

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**
CO1	3	2	1	-	2	1	-	1	1
CO2	2	3	2	1	1	-	-	2	1
CO3	3	2	2	1	1	1	1	1	2
CO4	2	3	3	1	2	2	-	1	2
CO5	3	2	1	2	2	1	1	2	2
Articulation	2.6	2.4	1.8	0.8	1.6	1.25	1.0	1.4	1.6

Note: The Articulation is indicative of how a particular course maps to a PO.

Articulation is calculated as:

$$= \frac{Sum\ of\ CO\ Mappiung\ Levels\ for\ a\ perticular\ PO}{No.of\ COs\ that\ mapped\ to\ that\ PO}$$

Example:1

$$Articulation\ of\ PO1 = \frac{(3+2+3+2+3)}{5} = 2.6$$

Example: 2

$$Articulation\ of\ PO7 = \frac{(1+1)}{2} = 1.0$$

Course Title:	**Computing concepts and problem solving in C**
Course Code:	**25VMT0C102**
Semester:	**I**
Credits:	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course covers Advanced concepts of computing concepts and problem solving in C and their applications in computer science. Students will develop problem-solving abilities and practical programming skills essential for core areas such as algorithms, systems programming, software development, and emerging technologies.

Course Outcomes (COs)

Upon successful completion of this course, learners will be equipped to:

CO	**Course Outcome**	**BTL**
CO1	Comprehend the advanced Programming constructs	L3
CO2	Apply File Operations and handle errors effortlessly	L3
CO3	Integrate linux Programming with C Scripting	L4
CO4	Demonstrate C System Programming	L3
CO5	Evaluate real-world scenarios and Case Studies	L5

Course Modules

No. of Hours:24

Module 1: Fundamentals of Computing and C Programming
Scope of Module: Introduces basic computing concepts, algorithm design, and foundational elements of the C programming language
Topics: Basics of Computers: Architecture, memory, input/output units; Introduction to Algorithms and Flowcharts; C Programming Basics: Structure, syntax, variables, constants, data types; Operators: Arithmetic, relational, logical, bitwise, assignment, conditional; Compilation and Execution process.
Learning Outcome*:Students will be able to describe computing concepts, write simple C programs, and understand compilation processes.*

No. of Hours:24

Module 2: Control Structures, Functions, Arrays, Strings, and Modular Programming
Scope of Module: Explores problem-solving strategies using control flow, modular programming, and essential data structures.
Topics: Control Structures: if, if-else, nested if, switch-case; Loops: for, while, do-while, nested loops; Functions: Declaration, definition, call, recursion; Scope, Storage Classes (auto, static, extern, register); Modular Programming: Header files, preprocessor directives

(#define, #include); Arrays: 1D, 2D arrays, array manipulation; Strings: String functions (strlen, strcpy, strcat, strcmp), string manipulation.
Learning Outcome:_Students will be able to apply control structures, modularize code, and manipulate arrays and strings effectively._

No. of Hours:24

Module 3: Pointers, Structures, Dynamic Memory Allocation, and File Handling
Scope of Module: Provides in-depth understanding of memory management, structured data, and file operations in C.
Topics: Introduction to Pointers: Pointer arithmetic, pointers and arrays; Pointers to Functions and Pointers to Structures; Structures and Unions: Declaration, definition, accessing members; Dynamic Memory Allocation: malloc, calloc, realloc, free; File Handling: File operations (read, write, append), file pointers; Case Study: Creating a file-based student management system.
Learning Outcome:_Students will be able to manipulate memory using pointers, manage structured data, and perform file operations._

No. of Hours:24

Module 4: ***System-Level Programming and Real-Time Applications in C***
Scope of Module:

Focuses on industry-relevant applications, system programming concepts, and real-time project development.
Topics: *Command-line Arguments and Environment Variables; Error Handling (perror, strerror) and Debugging Techniques; Mini Projects: Text-based Games (Tic-Tac-Toe), Simple Shell Implementation, Real-Time Data Logger for IoT (file-based); Case Study: Simulating a Banking System.*
Learning Outcome: *Students will be able to design and implement system-level programs and real-time applications using C.*

No. of Hours:24

Module 5: Advanced Industry Applications and Case Studies
Scope of Module: Encourages application of C programming skills in solving industry-specific problems and case studies.
Topics: Writing basic command-line tools; Simple network programs (client-server); Debugging and optimizing small C programs; Working with file compression and encryption tools; Implementing simple database file storage; Case Study: Building a basic calculator or file management tool; Mini Project: Simple automation program, data logger, or system monitor.
Learning Outcome*: Students will be able to apply C programming skills to solve advanced, industry-relevant problems through case studies and projects.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites (If Any):

- Basic knowledge of computer fundamentals.

Pedagogy / Teaching Methodology:

- ✓ Lectures, Hands-on Lab Sessions, Problem-solving Tasks, Case-based Assignments, Group Projects

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. NPTEL, Coursera, edX, MIT OCW — based on relevance to topics.

Suggested Readings:

- Standard textbooks, research articles, online tutorials and eBooks related to the subject.

Recommended Software/Tools (If applicable):

1. GCC Compiler, Visual Studio Code, Linux Terminal, C IDEs

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO PO Mapping Table

COs	**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**
CO1	3	3	2	2	2	1	-	1	2
CO2	3	2	2	3	2	1	-	1	1
CO3	2	3	2	2	-	1	1	1	1
CO4	3	2	3	2	2	1	-	1	2
CO5	2	2	2	3	2	2	-	2	2

Articulation	2.6	2. 4	2. 1	2. 4	2	1. 1	1	1. 1	1. 6

Course Title:	**Operating Systems**
Course Code:	**25VMT0C103**
Semester:	**I**
Credits:	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides an in-depth exploration of Advanced operating system concepts, focusing on the design, implementation, and management of modern operating systems. Topics include process synchronization, memory management, file systems, distributed systems, and security mechanisms. Practical assignments and case studies will reinforce theoretical knowledge, preparing students for real-world system-level programming and research.

Course Outcomes(COs)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Apply process management and synchronization techniques.	L3
CO2	Implement memory Management Techniques	L3
CO3	Implement File Management Techniques	L3
CO4	Design solutions using modern OS tools and case studies in distributed systems.	L3

CO5	Evaluate modern tools and techniques in real-world OS environments.	L5

Course Modules:

No. of Hours:24

Module 1: Introduction to Operating Systems and Process Management
Scope of Module: Introduces core OS concepts, structures, and the fundamentals of process management.
Topics: Overview of Operating Systems: Definitions, functions, types (batch, time-sharing, distributed, real-time); OS structure and components; System calls and APIs; Microkernel vs Monolithic Kernel architectures; System Boot Process and Initialization (GRUB, systemd); Process concepts, process states, process control block (PCB); Process scheduling algorithms (FCFS, SJF, Round Robin, Priority); Context switching; Process scheduling in multi-core and NUMA systems; Case studies on modern OS architectures (Linux, Windows).
Learning Outcome: *Students will be able to describe OS structures, functions, and manage basic process scheduling scenarios.*

No. of Hours:24

Module 2: Process Synchronization, Deadlock, and CPU Scheduling
Scope of Module: Covers critical section problems, synchronization mechanisms, deadlock handling, and CPU scheduling strategies.

Topics: Critical section problem, Peterson's solution, semaphores, monitors; Spinlocks, Mutexes, and Condition Variables; Hardware support for synchronization (Test-and-Set, Compare-and-Swap); Classic synchronization problems (producer-consumer, readers-writers, dining philosophers); Deadlock characterization, prevention, avoidance (Banker's algorithm), detection, recovery, and resource preemption; CPU scheduling strategies (multilevel queue, multilevel feedback queue); Real-world scheduling policies (CFS in Linux, Windows scheduler comparison); Case studies on synchronization in Linux kernel.
Learning Outcome: *Students will be able to apply synchronization techniques and deadlock handling strategies in process management.*

No. of Hours:24

Module 3: Memory Management, Virtual Memory, and Storage Systems
Scope of Module: Focuses on memory organization, virtual memory, and file systems management.
Topics: Memory management strategies: contiguous allocation, paging, segmentation; Virtual memory: demand paging, page replacement algorithms (LRU, FIFO, Optimal); Thrashing and performance tuning; Memory-mapped I/O and Direct Memory Access (DMA); Swap space management and tuning; File system interface and implementation, allocation methods (contiguous, linked, indexed); Directory structures, file system mounting, journaling file systems; Advanced file system topics: RAID levels, ZFS/Btrfs file systems; Security implications in storage systems (encryption at rest); Case study: EXT4 file system analysis.

Learning Outcome:*Students will be able to analyze memory and storage management strategies in OS environment*

No. of Hours:24

Module 4: Virtualization, Containerization, and Real-Time Systems
Scope of Module: Explores virtualization technologies, container-based systems, and introduces real-time OS concepts.
Topics: Virtualization concepts, hypervisors (Type 1 and Type 2), virtual machine monitors (VMWare, KVM); Nested Virtualization (running VMs inside VMs); Containerization with Docker, Kubernetes orchestration basics; Container Security (namespaces, cgroups, seccomp); Comparison of VMs vs Containers; Performance tuning of VMs and Containers; Introduction to real-time operating systems (RTOS), real-time scheduling algorithms; RTOS performance metrics and real-world RTOS examples (FreeRTOS, VxWorks); Case study: Docker-based microservices in cloud environments.
Learning Outcome:*Students will be able to evaluate virtualization and containerization technologies and apply real-time scheduling concepts.*

No. of Hours:24

Module 5: Security, Protection Mechanisms, and Advanced OS Case Studies
Scope of Module: Focuses on OS security, protection models, and advanced industrial applications of operating systems.
Topics: OS security principles, access control models (DAC, MAC, RBAC); Kernel hardening techniques (grsecurity, seccomp); Network security in OS: Firewall configurations (iptables, nftables); Secure Boot and Trusted Platform Module (TPM); Security vulnerabilities and mitigation strategies; Encryption, authentication, and auditing in OS; Introduction to SELinux, AppArmor; Forensics in OS environments (log analysis, intrusion detection); Case studies on OS security breaches and their prevention; Mini Project: Design a simple OS-level security module, process monitoring utility, or Intrusion Detection System (IDS) script.
Learning Outcome:*Students will be able to assess and implement security models and protection mechanisms in modern OS environments.*

Module	**No. of Pre-Recordings**
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10.

Prerequisites (If Any):

- Basic knowledge of computer fundamentals.

Pedagogy / Teaching Methodology:

- Lectures, Hands-on Lab Sessions, Problem-solving Tasks, Case-based Assignments, Group Projects

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. NPTEL, Coursera, edX, MIT OCW — based on relevance to topics.

Suggested Readings:

- Standard textbooks, research articles, online tutorials and eBooks related to the subject.

Recommended Software/Tools (If applicable):

1. GCC Compiler, Visual Studio Code, Linux Terminal, C IDEs

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools

5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO PO Mapping Table

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**
CO1	3	2	2	2	2	1	-	1	2
CO2	2	3	3	2	2	1	1	2	1
CO3	3	2	2	2	2	1	1	2	2
CO4	2	3	3	2	2	2	1	1	2
CO5	3	2	2	3	3	2	1	2	2
Articulation	2.6	2.4	2.4	2.2	2.2	1.4	1	1.4	1.6

Course Title:	**Data Structures**
Course Code:	**25VMT0C104**
Semester:	**I**
Credits:	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description

This course introduces Advanced Concepts of data structures, algorithms, and their practical use in computing applications. Students will learn how to implement, analyze, and apply various linear and non-linear data structures to build efficient and scalable software systems

Course Outcomes(Cos)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Understand the fundamentals of advanced linear and data structures.	L1
CO2	Apply appropriate data structures for solving complex computational problems.	L3
CO3	Apply tree graph techniques in software solutions.	L3
CO4	Implement hashing techniques in software solutions.	L3
CO5	Evaluate efficient programs using modern tools and real-world case studies.	L5

Note: Implementation can be done in C/C++

Course Modules

No. of Hours:24

Module 1: : Linear Data Structures
Scope of Module Covers implementation and usage of linear storage structures.
Topics: Arrays, Stacks (array/linked), Queues (normal, circular), Dequeues, Linked list operations (SLL, DLL, CLL), Stack/Queue with linked lists, Applications in expression evaluation, Bracket matching, Recursion stack, Queue in scheduling, List-based simulation, Sparse matrices, Time/space complexity basics, Array vs list.
***Learning Outcome**: Apply linear structures in solving algorithmic problems.*

No. of Hours:24

Module 2: Trees and Binary Trees
Scope of Module Focuses on hierarchical data structures with operations and applications of binary trees and advanced trees.
Topics: Tree definitions and terminologies, Binary tree representation, Binary tree traversals (inorder, preorder, postorder), Binary search tree (BST), Insertion in BST, Deletion in BST, Balanced trees (AVL trees), Height-balanced trees, Expression trees, Threaded

binary trees, Heap trees, Min-heap and max-heap, Priority queue using heap, B-Trees, Applications of trees.
***Learning Outcome**: Students will be able to implement tree structures and use them for hierarchical data processing.*

No. of Hours:24

Module 3: Graphs and Applications
Scope of Module: Introduction to graphs and path-finding algorithms.
Topics: Graph representation (adjacency list/matrix), BFS, DFS, Topological sorting, Dijkstra's algorithm, Bellman-Ford, Floyd-Warshall, Kruskal's algorithm, Prim's algorithm, Network flow, Graph coloring, Euler and Hamiltonian paths, Graph traversals for cycles, Applications in networks, Graph problem modeling.
***Learning Outcome:** Analyze and solve graph-based problems using suitable algorithms*

No. of Hours:24

Module 4: Hashing and Searching Structures
Scope of Module: Efficient search, hash, and map-based data access.
Topics:

Hash tables, Hash functions, Collision resolution (chaining, open addressing), Load factor, Linear probing, Double hashing, Bloom filters, Dictionaries, Sets, Unordered maps, Binary search trees vs hashing, Search time analysis, Skip lists, LRU Cache design, Memory-optimized hashing, Applications in OS/DB.
Learning Outcome: *Design and implement efficient search and mapping structures.*

No. of Hours:24

Module 5: Case Studies and Tools
Scope of Module: Tools and case implementations using advanced DS.
Topics: STL containers in C++ (vector, list, map, set), Data structure libraries in C++, Visualization tools (VisuAlgo, Graphviz), IDEs (CLion, Eclipse), Real-world DS projects, Competitive programming platforms, Time-space profiler tools, LeetCode/Codeforces case studies, DS in AI/ML preprocessing, Game development DS, Cache designs, Real DBMS indexing, Distributed DS structures.
Learning Outcome: *Evaluate and apply real-world DS solutions using modern tools and case studies.*

Module	**No. of Pre-Recordings**
1	**10**
2	**12**
3	**10**

4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10.

Prerequisites (If Any):

- Basic knowledge of programming and algorithms.

Pedagogy / Teaching Methodology:

- ✓ Interactive Lectures
- ✓ Hands-on Programming Labs
- ✓ Assignments and Case Studies
- ✓ Mini Projects

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. NPTEL: Data Structures and Algorithms
2. Coursera: Data Structures by UC San Diego

Suggested Readings:

- Seymour Lipschutz, *Data Structures*, McGraw-Hill
- Ellis Horowitz, Sartaj Sahni, *Fundamentals of Data Structures*, Universities Press
- Mark Allen Weiss, *Data Structures and Algorithm Analysis in C*, Pearson Education
- Online resources and research papers related to data structures

Recommended Software/Tools (If applicable):

1. C/C++ Compiler
2. Visual Studio Code / Code::Blocks
3. Online IDEs (Repl.it, GeeksforGeeks IDE)

4. Visualgo.net for algorithm visualization

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO PO Mapping Table

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**
CO1	3	2	3	2	2	3	-	2	1
CO2	3	3	3	2	2	2	1	2	2
CO3	2	2	3	3	2	2	3	1	2

CO4	3	2	3	2	2	-	2	1	2
CO5	2	3	2	3	2	2	-	2	1
Articulation	2.6	2.4	2.8	2.4	2	2.2	2	1.6	1.6

Course Title:	**AI Skills for Future**
Course Code:	**26VMX0S105**
Semester:	**I**
Credits:	**2**
Course Type	**Skill Enhancement Course**
Learning Hours	60
Live Sessions	**12 hours**

Course Outcomes:

CO1: Understand Generative AI and LLMs in depth
CO2: Create text, image, and voice content using AI
CO3: Automate daily tasks and business workflows
CO4: Build AI agents that act independently
CO5: Design and deploy simple AI apps without coding

Course Modules

Module 1: Generative AI Prompt Engineering for Real Workflows

What is Generative AI and how it's transforming industries, Structure of an effective prompt: *Role* → *Task* → *Context* → *Output,* Common prompting mistakes and how to fix them, How large language models (LLMs) like ChatGPT, Claude, and Google AI Studio work, advanced prompting (few-shot, iterative, style-based), Using your own files in ChatGPT and Claude (RAG basics), Research and summarization using AI

Learning Outcome: Students will be able to understand how Generative AI works and master the art of prompting.

Module 2: Creative AI and Voice & Audio Creation

Understanding AI image generation, creating posters, campaign assets, and product visuals, building brand identity using Krea.ai, Generating realistic voiceovers using ElevenLabs, Script

writing and tone control with ChatGPT & Claude, Integrating voice into short videos or podcasts ***Learning Outcome:*** Students will be able to use AI to generate visuals, voiceovers, and creative storytelling content.

Module 3: AI Automation & Productivity

Logic of triggers, actions, and workflows, setting up automation between apps (Gmail, Sheets, Drive), Connecting OpenAI to Make.com for intelligent automation, creating social media Management Workflow and research automation flows, automating daily reports and reminders, building personal productivity dashboards, Integrating Claude and ChatGPT into existing workflows

Learning Outcome: Automate daily workflows and tasks using AI and no-code tools.

Module 4: Advanced AI Agents & Decision Flows

Introduction to n8n as a visual AI workflow builder, understanding nodes, triggers, and data passing, Using OpenAI node for smart responses and actions, designing multi-step reasoning workflows, Connecting Google Sheets, Slack, and other APIs, building autonomous agents that perform specific tasks

Learning Outcome: Students will be able to build AI agents that can think, decide, and act automatically.

Module 5: AI App Building & Deployment

Fundamentals of UI/UX for AI apps, structuring front-end and AI back-end flows, designing app logic using n8n and Make.com, integrating OpenAI and Claude APIs to your apps, handling user inputs and dynamic responses, using Google AI Studio to experiment with models, hosting and testing your AI app online, adding analytics and monitoring usage, final demo preparation and peer feedback

Learning Outcome: Students will be able to learn to design, connect, and deploy simple AI apps using no-code tools

Corporate Office
91/2, Dr. A N Krishna Rao Road V V Puram,
Bengaluru - 560 004
P: +91 80 4343 1000

Centre for Distance & Online Education
319, 25th Main, 17th Cross,
J P Nagar 6th Phase
Bengaluru – 560 078
P: +91 80 4343 0100

CENTRE FOR DISTANCE AND ONLINE EDUCATION

ONLINE

Semester-II

Course Title:	Advanced Database Systems
Course Code:	25VMT0C201
Semester:	II
Credits:	4
Course Type	Core Course
Learning Hours	120
Live Sessions	12 hours

Course Description:

This course delves into advanced concepts and techniques in database systems, focusing on design, implementation, optimization, and emerging trends. Topics include distributed databases, NoSQL systems, data warehousing, transaction management, and big data technologies. The course emphasizes practical applications through case studies and hands-on projects.

Course Outcomes (COs)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Understand advanced database models and architectures.	L1
CO2	Design and optimize relational and NoSQL databases.	L3
CO3	Implement transaction management and concurrency control.	L4
CO4	Evaluate distributed database systems and their challenges.	L5
CO5	Apply data warehousing and big data technologies.	L3

Course Modules

No. of Hours:24

Module 1: Advanced Database Models and Architectures
Scope of Module: Covers advanced database models, including object-oriented, object-relational, and NoSQL databases, and their architectures.
Topics: Review of relational database systems; Object-oriented databases; Object-relational databases; NoSQL database overview; Document databases (MongoDB); Key-value databases (Redis); Column-family databases (Cassandra); Graph databases (Neo4j); NewSQL systems; Polyglot persistence; CAP theorem; BASE properties; Cloud-native database systems; Multi-model databases; Case study: MongoDB vs. PostgreSQL.
Learning Outcome: *Students will be able to compare and contrast different database models and select appropriate architectures for specific applications*

No. of Hours:24

Module 2: Database Design and Query Optimization
Scope of Module: Focuses on advanced techniques for database design, indexing, query optimization, and performance tuning
Topics: Advanced normalization (1NF–5NF); Denormalization strategies; Physical vs. logical design Partitioning strategies; Indexing: B-tree, hash, bitmap; Covering indexes and index-only scans Query execution plans; Cost-based query optimization; Heuristic-based optimization; Materialized views; Query rewriting techniques; Adaptive query

optimization; Statistics collection and use; SQL performance tuning tools; Hands-on: SQL tuning with EXPLAIN and ANALYZE.
Learning Outcome: *Students will be able to design efficient database schemas and optimize query performance.*

No. of Hours:24

Module 3: Transaction Management and Concurrency Control
Scope of Module: Explores transaction properties, concurrency control mechanisms, and recovery techniques in database systems
Topics: ACID properties; Transaction lifecycle; Serializability and schedules; Lock-based concurrency control; Deadlock detection and prevention; Timestamp ordering; Optimistic concurrency control; Two-phase locking; Multiversion Concurrency Control (MVCC); Isolation levels and anomalies; Serializable snapshot isolation; Write-ahead logging; Checkpointing; Recovery methods; Case study: PostgreSQL concurrency and recovery.
Learning Outcome: *Students will be able to implement and analyze transaction management and concurrency control mechanisms.*

No. of Hours:24

Module 4: Distributed Database Systems
Scope of Module:

Examines distributed database architectures, data fragmentation, replication, and consistency models.
Topics: Distributed database architecture; Client-server vs. peer-to-peer; Horizontal and vertical fragmentation; Data allocation and replication; Distributed concurrency control; Distributed transactions; Two-phase commit protocol; Distributed query processing. Query decomposition and localization; CAP theorem in distributed databases; Eventual consistency models; Conflict resolution strategies; Fault tolerance and availability; Case study: Google Spanner; Hands-on: Setting up a distributed Cassandra cluster.
Learning Outcome: *Students will be able to evaluate distributed database systems and address their challenges.*

No. of Hours:24

Module 5: Data Warehousing and Big Data Technologies
Scope of Module: Introduces data warehousing concepts, ETL processes, OLAP, and big data technologies
Topics: Data warehousing concepts; Star schema vs. snowflake schema; Fact and dimension tables ETL process design; Data cleaning and transformation; Data staging and loading; OLAP operations and cubes; Hadoop ecosystem (HDFS, MapReduce); Apache Hive and Pig Apache Spark fundamentals; NoSQL for Big Data: HBase and Cassandra; Data lakes and lakehouse architecture; Real-time data pipelines (Kafka + Spark Streaming); Columnar storage formats (Parquet, ORC); Mini-project: Building a Spark-based data pipeline

Learning Outcome: *Students will be able to apply data warehousing and big data technologies for analytical processing*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites (If Any):

- Basic knowledge of database management systems (DBMS) and SQL.

Pedagogy / Teaching Methodology:

- ✓ Conceptual and problem-based learning
- ✓ Hands-on labs and case studies
- ✓ Group projects and presentations
- ✓ Quizzes and assignments using database tools

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. Advanced Database Systems by NPTEL: https://nptel.ac.in/courses/106105171
2. Database Systems by Stanford Online: https://online.stanford.edu/courses/soe-ydatabases-database-systems

Suggested Readings:

- Raghu Ramakrishnan and Johannes Gehrke, *Database Management Systems*, McGraw-Hill, 2003.
- Hector Garcia-Molina, Jeffrey D. Ullman, and Jennifer Widom, *Database Systems: The Complete Book*, Pearson, 2008.
- Martin Kleppmann, *Designing Data-Intensive Applications*, O'Reilly, 2017.

Recommended Software/Tools:

1. PostgreSQL, MongoDB, Cassandra
2. Hadoop, Spark
3. SQL Developer, Tableau

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO/PO Mapping Table

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**
CO1	3	2	1	-	2	1	-	1	-
CO2	2	3	2	1	1	-	-	2	-
CO3	3	2	2	1	1	1	1	1	2
CO4	2	3	3	1	2	2	-	2	2
CO5	3	2	1	2	2	1	1	2	2
Articulation	2.6	2.6	1.8	1.25	1.6	1.25	1	1.4	2

Course Title:	Java Programming
Course Code:	25VMT0C202
Semester:	II
Credits:	4
Course Type	Core Course
Learning Hours	120
Live Sessions	12 hours

Course Description:

This course provides an in-depth understanding of Java programming, covering object-oriented principles, core APIs, multithreading, networking, and modern frameworks. Emphasis is placed on practical implementation through projects, case studies, and hands-on labs to build robust applications.

Course Outcomes (Cos)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Understand object-oriented principles in Java.	L1
CO2	Develop applications using Java core APIs and collections.	L3
CO3	Implement multithreading and exception handling.	L4
CO4	Design GUI applications and network-based solutions.	L5
CO5	Apply Java frameworks (Spring Boot) for enterprise applications.	L3

Course Modules

No. of Hours:24

Module 1: Object-Oriented Programming in Java
Scope of Module: Covers fundamental OOP concepts, Java syntax, and class design
Topics: Introduction to Java and JVM; Data types, operators, and control structures; Class structure and object creation; Encapsulation and constructors; Inheritance and method overriding; Polymorphism and dynamic binding; Interfaces and abstract classes; Package creation and import; Access modifiers and scope; Static vs. instance members; Exception handling: try-catch-finally; Custom exceptions; Assertions and error handling; Object class and method overriding; Case study: Modular banking system design.
Learning Outcome: *Students will be able to apply OOP principles such as encapsulation, inheritance, and polymorphism to design structured and reusable Java applications.*

No. of Hours:24

Module 2: Java Core APIs and Collections
Scope of Module: Focuses on Java standard libraries and data structures.
Topics: String manipulation and immutability; Regular expressions; Wrapper classes and autoboxing Generics in Java; ArrayList, LinkedList implementations; HashSet, TreeSet, LinkedHashSet usage; HashMap, TreeMap, LinkedHashMap differences; Iterators, ListIterators, and enhanced for-loop; Comparable vs. Comparator; Collections utility

class; File handling with java.io and java.nio; Object serialization and deserialization; Properties and configuration files; Practical: Inventory management system; Debugging collections and memory leaks.
Learning Outcome: *Students will be able to leverage core Java libraries and the Collections Framework to build efficient, maintainable applications with complex data structures.*

No. of Hours:24

Module 3: Multithreading and Concurrency
Scope of Module: Explores concurrent programming in Java.
Topics: Thread lifecycle and states; Thread creation via Thread class; Runnable interface and Lambda expressions; Thread prioritization and yielding; Synchronized methods and blocks; Intrinsic locks and synchronized collections; Volatile keyword and atomic variables; Deadlock scenarios and prevention strategies; wait(), notify(), and notifyAll(); Thread pooling and ExecutorService; Callable and Future interfaces; ConcurrentHashMap and thread-safe queues; ForkJoinPool basics; Case study: Multithreaded chat application; Concurrency debugging tools and techniques.
Learning Outcome: *Students will be able to implement multithreaded programs and apply Java concurrency tools to ensure safe, scalable execution of parallel tasks.*

No. of Hours:24

Module 4: GUI and Networking
Scope of Module: Introduces Java FX and network programming.
Topics: Introduction to JavaFX and GUI architecture; SceneBuilder and JavaFX controls; Layout managers: BorderPane, VBox, HBox; Event-driven programming in Java; Creating forms and validation; Swing basics and component lifecycle; Action listeners and adapters; Socket programming: TCP; Datagram programming: UDP; HTTP communication using HttpClient Using REST APIs in Java; JDBC for database connectivity; CRUD operations with JDBC Mini-project: GUI Weather App with REST API; Threading in GUI and networking applications
Learning Outcome: *Students will be able to design and build user-friendly GUI applications integrated with network communication and backend services using Java.*

No. of Hours:24

Module 5: Java Frameworks and Modern Tools
Scope of Module: Covers Spring Boot and testing frameworks.
Topics: Introduction to Spring Boot and its architecture; Dependency Injection (DI) in Spring; Creating RESTful APIs with Spring Boot; Spring annotations and application.properties; CRUD operations with Spring Data JPA; Connecting to MySQL/PostgreSQL; Validation with Spring MVC; Unit testing with JUnit 5; Mocking with Mockito; Postman for API testing; Building with Maven and Gradle; Logging with

Logback and SLF4J; Creating JAR/WAR packages; Dockerizing a Spring Boot app; Capstone project: E-commerce backend.
Learning Outcome: *Students will be able to develop full-stack Java applications using Spring Boot and related modern tools, and deploy them in enterprise-ready environments.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10

Prerequisites (If Any):

- Basic programming knowledge (C/C++/Python).

Pedagogy / Teaching Methodology:

- ✓ Interactive lectures with live coding
- ✓ Hands-on labs and debugging sessions
- ✓ Agile-style group projects
- ✓ Code reviews and peer assessments

Certificate / Value Added Courses Recommended:

- Java Programming by NPTEL: https://nptel.ac.in/courses/106105168

- Java Fundamentals by Oracle Academy: https://academy.oracle.com

Suggested Readings:

- Herbert Schildt, *Java: The Complete Reference*, McGraw-Hill, 2021.
- Joshua Bloch, *Effective Java*, Addison-Wesley, 2017.
- Craig Walls, *Spring Boot in Action*, Manning, 2020.

Recommended Software/Tools:

1. JDK 17+, IntelliJ IDEA/Eclipse
2. Spring Tool Suite, Postman
3. Git, Maven

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	–	2	1	–	1	2
CO2	2	3	2	1	-	–	–	2	-
CO3	3	2	2	-	-	1	2	-	2
CO4	2	3	3	1	2	2	–	1	2
CO5	3	2	-	2	2	1	1	2	2
Articulation	2.6	2.4	2	1.33	2	1.25	1.5	1.5	2

Course Title:	**Web Technologies**
Course Code:	**25VMT0C203**
Semester:	**II**
Credits:	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides comprehensive knowledge of modern web development technologies, covering front-end design, back-end programming, databases, and web security. Students will learn to build responsive, interactive web applications using HTML5, CSS3, JavaScript, PHP/Node.js, and frameworks like React and Express. Practical implementation through projects and case studies is emphasized.

Course Outcomes (COs)

Upon successful completion of this course, learners will be equipped to:

CO	**Course Outcome**	**BTL**
CO1	Understand client-side web technologies (HTML5, CSS3, JavaScript).	L1
CO2	Develop interactive front-end applications using frameworks.	L3
CO3	Implement server-side programming and database integration.	L4
CO4	Evaluate web security threats and countermeasures.	L5

CO5	Build full-stack web applications using modern tools.	L3

Course Modules

No. of Hours:24

Module 1: Client-Side Web Technologies
Scope of Module: Covers fundamental technologies for building web interfaces.
Topics: HTML5 document structure and semantic tags; Forms and input validation; CSS3 selectors and specificity; Box model, flexbox, and grid layouts; Responsive design principles (media queries); Typography and UI consistency; JavaScript syntax and data types; Control structures and functions; ES6+ features: let/const, arrow functions, template literals; DOM tree and traversal; Event handling and delegation; Form validation using JavaScript; Browser console and developer tools; Animation with CSS and JavaScript; Project: Portfolio website with interactivity.
Learning Outcome: *Students will be able to create accessible, responsive, and well-structured web pages using HTML5, CSS3, and modern JavaScript.*

No. of Hours:24

Module 2: Front-End Frameworks
Scope of Module: Focuses on modern JavaScript frameworks for UI development
Topics: Introduction to React.js and component lifecycle; JSX syntax and virtual DOM; Functional components and props; State and event management; Conditional

rendering and lists; Forms and controlled components; React Hooks (useState, useEffect); Component communication; React Router for navigation; Context API for global state; AJAX requests with Axios and Fetch API; Using UI libraries (Bootstrap, Material UI); Creating reusable components; Code splitting and lazy loading; Project: Product listing page for e-commerce.
Learning Outcome: *Students will be able to build scalable and maintainable front-end applications using modern JavaScript frameworks like React.*

No. of Hours:24

Module 3: Server-Side Programming
Scope of Module: Explores back-end development and database integration.
Topics: Node.js basics and NPM; Creating a local web server; Express.js routing and middleware; Handling POST and GET requests; Template engines (EJS/Handlebars); Form data parsing and validation; MySQL and MongoDB CRUD operations; Connection pooling and error handling; RESTful API design principles; JWT and session-based authentication; Role-based access control; File uploads and file system module; Environment variables and configuration; Project: Blog system with authentication and database; Debugging Node.js applications.
Learning Outcome: *Students will be able to build robust server-side applications and RESTful APIs integrated with databases using Node.js and Express.*

No. of Hours:24

Module 4: Web Security

Scope of Module: Addresses security vulnerabilities and protection mechanisms.
Topics: Understanding HTTP protocol and HTTPS; SSL certificates and secure headers; OWASP Top 10 vulnerabilities; SQL injection and input sanitization; Cross-Site Scripting (XSS) and mitigation; Cross-Site Request Forgery (CSRF); Password hashing with bcrypt; Authentication vs. Authorization; Content Security Policy (CSP); Secure cookie practices; Same-Origin Policy and CORS; Rate limiting and API throttling; Penetration testing tools (Burp Suite, OWASP ZAP); Security audit and logging; Project: Secure login and audit trail system
Learning Outcome: *Students will be able to identify, prevent, and mitigate common web application security threats using industry best practices.*

No. of Hours:24

Module 5: Full-Stack Development
Scope of Module: Integrates all layers of web application development.
Topics: Overview of MERN/MEAN stacks; Connecting frontend with backend via APIs; Redux basics and state management; User authentication and protected routes; Role-based content rendering; API testing with Postman; Code versioning with Git and GitHub; Continuous deployment workflows; Hosting frontend (Netlify, Vercel); Hosting backend (Render, Heroku) Performance optimization tips; Unit testing with Jest and Mocha; Linting and code formatting (Prettier, ESLint); DevOps basics for deployment; Capstone Project: Social media dashboard with analytics.

Learning Outcome: *Students will be able to develop, test, and deploy full-stack web applications using modern development tools and deployment platforms.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10

Prerequisites (If Any):

- Basic programming knowledge (HTML/CSS recommended).

Pedagogy / Teaching Methodology:

- Interactive coding demonstrations
- Pair programming exercises
- Project-based learning
- Security hackathons

Certificate / Value Added Courses Recommended:

1. Web Development by MDN: https://developer.mozilla.org
2. Full-Stack Open: https://fullstackopen.com

Suggested Readings:

- Jon Duckett, *HTML & CSS: Design and Build Websites*, Wiley, 2011

- Eric Freeman, *Head First JavaScript Programming*, O'Reilly, 2014
- David Flanagan, *JavaScript: The Definitive Guide*, O'Reilly, 2020

Recommended Software/Tools:

1. VS Code, Chrome DevTools
2. Postman, MongoDB Compass
3. React DevTools, Git

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	–	2	1	–	1	-
CO2	2	3	2	1	1	–	–	2	1
CO3	3	2	2	1	1	1	1	1	2
CO4	2	3	3	1	2	2	–	1	2
CO5	3	2	1	2	2	1	1	2	2
Articulation	2.6	2.4	1.8	1.25	1.6	1.25	1	1.4	1.75

Course Title:	**Cloud Computing Foundations**
Course Code:	**25VMT0C204**
Semester:	**II**
Credits:	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course introduces fundamental concepts of cloud computing, including service models, deployment models, virtualization, and cloud architecture. Students will gain hands-on experience with major cloud platforms (AWS/Azure/GCP) and learn to deploy, manage, and secure cloud-based solutions. The course emphasizes practical applications through labs and real-world case studies.

Course Outcomes (COs)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Understand cloud computing concepts and service models.	L1
CO2	Deploy and manage virtualized infrastructure.	L3
CO3	Analyze cloud storage and networking solutions.	L4
CO4	Evaluate cloud security and compliance frameworks.	L5
CO5	Develop cloud-native applications using PaaS.	L3

Course Modules

No. of Hours:24

Module 1: Cloud Computing Fundamentals
Scope of Module: Covers basic concepts, characteristics, and service models of cloud computing.
Topics: Definition and evolution of cloud computing; Essential characteristics of cloud (NIST model); Service models: IaaS, PaaS, SaaS; Deployment models: Public, Private, Hybrid, Community; Benefits and challenges of cloud adoption; Cloud computing use cases; Cloud vs. traditional IT infrastructure; Cloud service providers overview (AWS, Azure, GCP); Cloud service level agreements (SLAs); Multitenancy and resource pooling; Elasticity and scalability; Cloud computing cost models and pricing; Case study: Cloud migration strategy; Green computing and cloud efficiency; Cloud-first strategies in digital transformation**.**
Learning Outcome: *Students will be able to differentiate between cloud service and deployment models, and evaluate when and how to migrate to cloud environments.*

No. of Hours:24

Module 2: Virtualization and Cloud Infrastructure
Scope of Module: Focuses on virtualization technologies and cloud infrastructure management.
Topics: Virtualization concepts and benefits; Types of virtualization: Server, Storage, Network; Hypervisors: Type 1 (bare-metal) and Type 2 (hosted); Virtual machine creation and lifecycle; Introduction to containers and Docker; Container orchestration basics

(Kubernetes overview); Infrastructure as a Service (IaaS); AWS EC2, Azure VMs, GCP Compute Engine; Cloud autoscaling and elasticity; Elastic Load Balancing (ELB); Monitoring and cloud resource usage; Infrastructure as Code: Terraform and CloudFormation; Hands-on: Launching a VM on AWS/GCP; Cloud cost optimization techniques; Best practices for resource provisioning.
Learning Outcome: *Students will be able to configure and manage virtualized infrastructure on cloud platforms using IaaS and container technologies.*

No. of Hours:24

Module 3: Cloud Storage and Networking
Scope of Module: Explores storage solutions and network architectures in cloud environments.
Topics: Cloud storage types: Object, Block, File; Storage services: Amazon S3, Azure Blob, Google Cloud Storage; Data redundancy and availability zones; Content Delivery Networks (CDNs) Edge computing and latency reduction; Data lifecycle and storage classes; Virtual Private Cloud (VPC) architecture; Subnets, gateways, and NAT; Cloud firewalls and security groups Cloud DNS and networking services; VPNs and hybrid cloud connectivity ;Database as a Service (DBaaS): RDS, Cosmos DB, Cloud SQL; Data backup and disaster recovery; Case study: Designing a high availability architecture; Storage and network performance benchmarking.
Learning Outcome: *Students will be able to design and configure secure, high-performance storage and networking solutions within cloud environments.*

No. of Hours:24

Module 4: Cloud Security and Compliance
Scope of Module: Addresses security challenges and compliance requirements in cloud computing.
Topics: Cloud security principles and models; Shared responsibility model; Identity and Access Management (IAM); Role-based access control (RBAC); Data encryption in transit and at rest; Key management services; Security groups vs. network ACLs; Firewall policies and VPN security; Security incident detection and response; Cloud compliance frameworks (GDPR, HIPAA, ISO 27001); Security posture assessments; Penetration testing and vulnerability scanning; Logging and auditing best practices; Case study: Securing a multi-tenant cloud app. Mini-project: Implementing IAM and encryption in AWS.
Learning Outcome: *Students will be able to implement cloud security measures and ensure compliance with data protection regulations and security frameworks.*

No. of Hours:24

Module 5: Cloud-Native Application Development
Scope of Module: Covers development of applications optimized for cloud environments
Topics: Introduction to cloud-native principles; Twelve-factor app methodology; Serverless architecture overview; AWS Lambda and Azure Functions; Event-driven programming in the cloud; Microservices architecture and benefits; REST API design and testing (Postman); Containers vs. serverless vs. VMs; CI/CD pipelines: GitHub Actions, AWS

CodePipeline; Monitoring and logging (CloudWatch, Stackdriver); Application scaling and performance tuning; API Gateway integration and throttling; DevOps tools for cloud (Jenkins, Terraform) Case study: Microservices deployment; Capstone Project: Build and deploy a serverless application.
Learning Outcome: *Students will be able to develop, deploy, and monitor scalable, cloud-native applications using modern cloud services and DevOps tools.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10

Prerequisites (If Any):

- Basic knowledge of operating systems and networking concepts.

Pedagogy / Teaching Methodology:

- ✓ Hands-on labs with cloud platforms
- ✓ Case studies of enterprise cloud migrations
- ✓ Security audit simulations
- ✓ Group projects with real-world scenarios

Certificate / Value Added Courses Recommended:

1. AWS Cloud Practitioner: https://aws.amazon.com/certification/
2. Microsoft Azure Fundamentals: https://learn.microsoft.com/en-us/certifications/

Suggested Readings:

- ✓ Dan Marinescu, *Cloud Computing: Theory and Practice*, 3rd Ed., Morgan Kaufmann, 2022
- ✓ Thomas Erl, *Cloud Computing: Concepts, Technology & Architecture*, Prentice Hall, 2021
- ✓ Official documentation from AWS, Azure, and GCP

Recommended Software/Tools:

1. AWS Free Tier, Azure Portal
2. Docker, Kubernetes (basic)
3. Terraform, CloudFormation

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)

6.	Indian Knowledge System

CO PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	–	2	1	–	1	1
CO2	2	3	2	1	-	–	–	2	1
CO3	3	2	2	1	-	1	2	1	2
CO4	2	3	3	1	2	2	–	1	2
CO5	3	2	1	2	2	-	2	2	2
Articulation	2.6	2.4	1.8	1.25	2	1.33	2	1.5	1.6

Course Title:	**Environmental, Social, and Governance (ESG) in IT**
Course Code:	**25VMT0C205**
Semester:	**II**
Credits:	**4**
Course Type	**Ability Enhancement Compulsory Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course explores the integration of Environmental, Social, and Governance (ESG) principles into information technology. It covers sustainable IT practices, ethical AI, data privacy, and corporate governance in tech organizations. Students will learn to assess ESG risks, implement green computing solutions, and align IT strategies with global sustainability goals through case studies and projects.

Course Outcomes (Cos)

Upon successful completion of this course, learners will be equipped to:

CO	**Course Outcome**	**BTL**
CO1	Explain ESG frameworks relevant to IT	L1
CO2	Apply green computing techniques	L3
CO3	Analyze social impacts of AI and big data	L4
CO4	Evaluate governance models for IT ethics	L5
CO5	Evaluate sustainable IT policies for organizations	L5

Course Modules

No. of Hours:24

Module 1: ESG Fundamentals in IT
Scope of Module: Introduces core ESG concepts and their application in technology sectors.
Topics: Understanding ESG in the context of IT; Global ESG frameworks: UN SDGs, GRI, SASB; Environmental impacts of digital infrastructure; Carbon footprint of data centers and cloud computing; E-waste management and recycling strategies; IT lifecycle sustainability assessments; Role of ICT in sustainable development goals; Green certifications for IT (Energy Star, EPEAT); Sustainability KPIs in IT projects; ESG compliance for tech startups AI and sustainability intersections; Digital ethics and sustainable design; Sustainable supply chain management in IT; Case study: Google's carbon-neutral cloud initiatives; Future of ESG in emerging technologies
Learning Outcome: *Students will articulate how ESG principles apply to IT infrastructure, service models, and sustainability reporting.*

No. of Hours:24

Module 2: Green Computing
Scope of Module: Focuses on sustainable IT practices and resource optimization.
Topics: Green hardware lifecycle management; Energy-efficient processor design; Power-aware scheduling algorithms; Green software design and refactoring; Virtualization for sustainability; Resource optimization in cloud environments; Carbon-aware

computing; Renewable energy integration in data centers; Measuring IT carbon footprints; Green networking and IoT devices; Cooling technologies and data center PUE; Server consolidation strategies; Green code practices in DevOps; Life Cycle Assessment (LCA) tools for IT systems; Lab: Calculate emissions from virtual machines.
Learning Outcome: *Students will be able to apply green computing principles to minimize the environmental impacts of IT operations and software systems.*

No. of Hours:24

Module 3: Social Responsibility in Tech
Scope of Module: Examines ethical implications of AI, data, and digital inclusion.
Topics: Digital equity and inclusion challenges; Algorithmic bias and mitigation techniques; AI transparency and explainability; Ethical data collection and consent; GDPR, CCPA, and global data privacy laws; Bias in training data for machine learning; Accessible design principles in software; Digital literacy and education programs; Tech for underserved communities. Social media and societal impact; AI in healthcare and social justice; Tech activism and digital rights; Workplace inclusivity in tech companies; Case study: Microsoft's AI for Accessibility Lab: Bias detection in data classification models.
Learning Outcome: *Students will analyze social implications of emerging technologies and advocate for inclusive, ethical digital innovation.*

No. of Hours:24

Module 4: IT Governance and Ethics
Scope of Module: Covers governance structures for ethical IT decision-making.
Topics: Foundations of IT corporate governance; ESG reporting frameworks and tools; Whistleblower policies and ethics channels; IT ethics boards and oversight; Compliance versus ethical leadership; IT audit trails and data governance; Transparency and accountability practices; Ethical AI governance and regulation; Cybersecurity ethics in surveillance systems; Boardroom simulation: resolving tech dilemmas; Balancing profit and public interest in IT firms; Decision-making under ethical uncertainty; ISO standards for IT governance; ESG ratings and public disclosures; Case study: Governance failure in a tech giant.
Learning Outcome: *Students will evaluate and develop governance frameworks that support ethical decision-making and stakeholder alignment in IT environments.*

No. of Hours:24

Module 5: Sustainable IT Strategy
Scope of Module: Integrates ESG principles into organizational IT planning.
Topics: Integrating ESG into IT strategy roadmaps; Stakeholder mapping and engagement models; Circular economy in IT hardware; Supply chain resilience and transparency; Cloud sustainability strategy; Responsible tech innovation lifecycle; Designing ESG-aligned KPIs for IT; Tech sustainability scorecards and benchmarking; Policy modeling

for net-zero IT operations; Carbon offsetting through tech initiatives; Funding and investing in green IT Organizational change for ESG transformation; Capstone: ESG audit of a mock IT enterprise Scenario planning for climate-resilient IT; Report writing: ESG strategy white paper.
Learning Outcome: *Students will design sustainable IT policies and long-term strategic plans incorporating ESG values for enterprise impact.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Basic knowledge of IT systems and corporate social responsibility.

Pedagogy:

- ✓ Guest lectures from ESG tech leaders
- ✓ Carbon accounting workshops
- ✓ Ethics hackathons
- ✓ ESG report analysis

Certifications Recommended:

1. Certified Green IT Professional (GITP)

2. SASB's ESG Fundamentals

Readings:

- "Green IT for Sustainable Business Practice" by Mark O'Neill
- "AI Ethics" by Mark Coeckelbergh

Tools:

- AWS Customer Carbon Footprint Tool
- ESG reporting software (Sustainalytics)

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO PO Mapping Table

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**
CO1	2	1	-	3	1	-	2	1	2
CO2	3	2	1	2	3	1	-	2	1
CO3	1	3	2	-	2	3	1	1	3
CO4	2	1	3	2	1	2	3	-	2
CO5	3	2	2	3	2	1	2	2	3
Articulation	2.6	1.8	2	2.5	1.8	1.75	2	1.5	2.2

CENTRE FOR DISTANCE AND ONLINE EDUCATION

ONLINE

Semester III

Course Title:	**Artificial Intelligence and Machine Learning**
Course Code:	**251VMT0C301**
Semester:	**III**
Credits:	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides MCA students with an industry-relevant and application-driven exploration of Artificial Intelligence (AI) and Machine Learning (ML). The syllabus is designed to offer a progressive and logical learning path from AI fundamentals to advanced ML techniques and deployment strategies. The course also emphasizes ethical, explainable, and production-ready AI systems through a mix of practical labs, case studies, and modern tools.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO No.	Course Outcome	BTL
CO1	Understand the evolution and fundamentals of Artificial Intelligence and Machine Learning.	L1
CO2	Apply advanced ML algorithms for supervised, unsupervised, and semi-supervised learning tasks.	L3
CO3	Analyze deep learning models and reinforcement learning strategies.	L4

CO4	Evaluate models for bias, interpretability, and effectiveness across varied datasets.	L5
CO5	Design and deploy scalable, explainable, and secure AI/ML applications.	L3

Course Modules

No. of Hours:24

Module 1: Foundations of AI and ML
Scope of Module: Establishes core concepts of AI and ML, setting the stage for in-depth exploration of models, learning paradigms, and industry practices.
Topics: AI and ML definitions and key differences, history and evolution of AI, AI vs ML vs DL, symbolic vs sub-symbolic AI, types of ML (supervised, unsupervised, reinforcement), ML lifecycle and pipeline stages, CRISP-DM methodology, introduction to datasets and data types, common data challenges, bias and fairness in ML, ML vs traditional programming, use cases in various industries, recent trends in AI (Generative AI, LLMs), AI ethics overview.
Learning Outcome: *Students will be able to articulate core AI/ML definitions, classify learning types, and describe modern trends and challenges.*

No. of Hours:24

Module 2: Advanced Supervised Learning and Ensemble Models
Scope of Module:

Covers core and advanced supervised learning techniques including ensemble strategies and hyperparameter tuning.
Topics: Linear and logistic regression with regularization (Ridge, Lasso), decision trees, random forest, support vector machines (SVM), KNN classification, Naive Bayes, model evaluation metrics (accuracy, precision, recall, F1, ROC-AUC), model selection and validation strategies (train/test split, cross-validation), bias-variance tradeoff, grid search and random search, ensemble learning (bagging, boosting), introduction to XGBoost, LightGBM, and CatBoost, multi-output and multiclass classification, managing imbalanced datasets (SMOTE, undersampling), SHAP for tree models.
Learning Outcome: *Students will be able to apply and evaluate supervised learning and ensemble methods to solve classification and regression tasks.*

No. of Hours:24

Module 3: Unsupervised Learning, Dimensionality Reduction, and Anomaly Detection
Scope of Module: Explores a wide range of unsupervised techniques and introduces preprocessing and anomaly detection as critical skills.
Topics: Clustering algorithms: k-means, DBSCAN, hierarchical, Gaussian Mixture Models (GMM), elbow method and silhouette score, association rule mining (Apriori, FP-Growth), dimensionality reduction techniques: PCA, t-SNE, UMAP, feature selection vs extraction, feature importance, anomaly detection using Isolation Forest, LOF, one-class SVM, preprocessing: encoding, scaling, normalization, dealing with missing values, pipeline creation using scikit-learn, unsupervised model evaluation strategies.

Learning Outcome: -*Students will be able to perform unsupervised learning, reduce dimensionality for visualization, and detect data anomalies effectively.*

No. of Hours:24

Module 4: Deep Learning and Reinforcement Learning
Scope of Module: Introduces deep neural networks and reinforcement learning paradigms for intelligent decision making.
Topics: Artificial Neural Networks (ANN), forward and backward propagation, activation functions (ReLU, sigmoid, tanh), loss functions and optimizers, CNNs for image processing, RNNs, LSTMs for sequential data, autoencoders and applications, GANs and generative models, overview of BERT and Transformer models, introduction to reinforcement learning, Q-learning and policy gradients, environment-reward-agent loop, OpenAI Gym, applications in recommendation, robotics, gaming.
Learning Outcome: -*Students will be able to build and analyze neural networks and apply RL for decision-making scenarios.*

No. of Hours:24

Module 5: AI/ML Model Deployment, MLOps, and Explainability
Scope of Module: Focuses on end-to-end AI/ML model deployment, production readiness, monitoring, and responsible AI practices.

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Topics:

Deployment using Flask, FastAPI, Streamlit, Docker containerization, introduction to Kubernetes for ML, CI/CD pipelines in ML, MLflow and model versioning, explainable AI tools (SHAP, LIME, ELI5), fairness audits and bias detection tools, logging and monitoring models in production, introduction to cloud ML platforms (AWS SageMaker, GCP Vertex AI), securing ML models and data, case studies from industry, best practices in production ML systems.

Learning Outcome: -*Students will be able to deploy, explain, and monitor AI/ML systems using modern DevOps practices and frameworks*

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Strong understanding of Python programming, statistics, and data structures.

Pedagogy / Teaching Methodology:

- ✓ Project-based learning with datasets from real-world domains
- ✓ Lab-based exercises using Jupyter and Google Colab
- ✓ Mini case studies and research paper reviews
- ✓ Live demonstrations and group activities
- ✓ Tool walkthroughs and guided deployment

Certificate / Value Added Courses Recommended:

1. Machine Learning Engineering for Production (MLOps) – Coursera
2. Deep Learning Specialization – Coursera
3. AI for Everyone – Coursera
4. Google ML Crash Course – https://developers.google.com/machine-learning/crash-course

Suggested Readings:

- Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, O'Reilly, 2022
- Ian Goodfellow, Deep Learning, MIT Press, 2016
- Sutton & Barto, Reinforcement Learning: An Introduction, MIT Press, 2018
- Kelleher, D. J., Data Science Ethics, MIT Press
- Relevant research from IEEE, ACM, and arXiv.org

Recommended Software/Tools:

1. Python, Scikit-learn, TensorFlow, Keras, PyTorch

2. Google Colab / JupyterLab, MLflow, Optuna
3. Docker, FastAPI, Streamlit, AWS SageMaker, GCP Vertex AI
4. SHAP, LIME, OpenAI Gym

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO-PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	1	2	-	1	1	2
CO2	3	2	3	2	2	1	-	2	2
CO3	2	3	2	3	2	1	2	-	2
CO4	2	2	2	-	2	3	2	2	2
CO5	3	3	3	2	3	2	2	3	3
Articulation	2.6	2.4	2.2	2	2.2	1.75	1.75	2	2.2

Course Title:	**Python Programming**
Course Code:	**251VMT0C302**
Semester:	**III**
Credits:	**4**
Course Type	**Core Course**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides MCA students with foundational to advanced skills in Python programming, blending basic concepts with real-world applications. It encompasses control structures, functions, OOP, file operations, exception handling, unit testing, web development, API integration, and data analysis. Special emphasis is placed on writing efficient, modular, and scalable Python code with exposure to industry-grade tools and frameworks.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO No.	Course Outcome	BTL
CO1	Understand Python syntax, control flow, data structures, and functions.	L1
CO2	Apply OOP principles and modular coding in real-world applications.	L3
CO3	Analyze problems and implement file, error handling, and unit testing.	L4

CO4	Evaluate web-based interactions, APIs, and database connectivity using Python.	L5
CO5	Develop interactive data-driven applications using Pandas, NumPy, and visualization libraries.	L3

Course Modules

No. of Hours:24

Module 1: Python Basics, Syntax, and Control Flow
Scope of Module: Introduces core elements of Python such as syntax, variables, data types, operators, and control flow.
Topics: Installing Python and IDE setup, writing and executing Python scripts, identifiers, keywords, variables, constants, data types, type conversion, operators and precedence, conditional statements (if, elif, else), iterative control (for, while), break and continue, pass statement, nested loops, input/output handling, formatted string literals (f-strings), Python coding style (PEP8).
Learning Outcome: *Students will be able to write structured Python scripts using control flow constructs and input/output operations.*

No. of Hours:24

Module 2: Functions, Collections, Iterators, and Modules
Scope of Module: Covers function design, advanced collection operations, and module creation for reusable code.

Topics: Defining and invoking functions, argument types (*args, **kwargs), scope and lifetime, recursion, anonymous functions (lambda), Python built-in functions, comprehension (list, dict, set), advanced list methods, tuples and unpacking, sets and operations, dictionaries and operations, iterators and generators, creating custom modules, importing built-in modules (math, sys, os, datetime, calendar, random), packaging with pip and venv.
Learning Outcome: *Students will be able to modularize programs using custom and built-in Python functions and libraries.*

No. of Hours:24

Module 3: Object-Oriented Programming, File I/O, and Exception Handling
Scope of Module: Explores OOP principles, file operations, and robust exception handling techniques.
Topics: Classes and objects, class and instance variables, methods, constructor and destructor, inheritance and types, method overriding, polymorphism, encapsulation, use of __str__, __repr__, __len__, working with text/binary/CSV files, reading and writing files, context manager and file closing, exception types and hierarchy, try-except-finally, raising and handling custom exceptions, logging module basics, assertions for debugging.
Learning Outcome: *Students will be able to design OOP-based applications with efficient file handling and error management*

No. of Hours:24

Module 4: Testing, Automation, and Database Connectivity
Scope of Module: Provides practices in automated testing, scripting for task automation, and integrating Python with databases.
Topics: Writing test cases using unittest and pytest, test discovery and test suites, test-driven development, automation with shutil and subprocess, working with file system using os and pathlib, scheduling tasks, introduction to SQLite and PostgreSQL, executing queries from Python using sqlite3 and psycopg2, performing CRUD operations, using context managers with databases, parameterized queries to prevent SQL injection.
Learning Outcome: *Students will be able to automate tasks and integrate databases securely in Python programs.*

No. of Hours:24

Module 5: Web Development, APIs, and Data Science Libraries
Scope of Module: Bridges Python to web and data science fields using APIs, Flask, and data analysis tools
Topics: Flask framework structure and routing, GET and POST methods, HTML rendering with Jinja2 templates, form data processing, REST API consumption using requests, OpenWeatherMap API integration example, Pandas: dataframes, merging, filtering, groupby, NumPy: array creation and operations, Matplotlib/Seaborn for plotting,

exporting data to Excel/CSV, introduction to Streamlit for interactive dashboards, mini project combining Flask and Pandas.
Learning Outcome: *Students will be able to develop and deploy data-centric and interactive Python web applications*

Module	No. of Pre-Recordings
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Basic logic building, familiarity with any programming language is helpful.

Pedagogy / Teaching Methodology:

- ✓ Concept-based demonstrations using Jupyter and IDEs
- ✓ Mini assignments and use-case-driven labs
- ✓ Group activities and debugging challenges
- ✓ Integration of GitHub for version control practices
- ✓ End-of-module review quizzes and projects

Certificate / Value Added Courses Recommended:

1. Python for Everybody – Coursera

2. Python OOP and Flask Projects – Udemy
3. Automate the Boring Stuff with Python – Online Book
4. Introduction to Data Science in Python – Coursera

Suggested Readings:

- Mark Lutz, Learning Python, O'Reilly
- Eric Matthes, Python Crash Course, No Starch Press
- Luciano Ramalho, Fluent Python, O'Reilly
- Python Documentation – https://docs.python.org

Recommended Software/Tools:

1. Python 3.x, PyCharm, VS Code
2. JupyterLab, Google Colab
3. Flask, Pandas, NumPy, Matplotlib, Streamlit
4. GitHub, SQLite, PostgreSQL

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools

5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO-PO Mapping Table

CO/PO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**
CO1	3	2	2	1	2	1	-	1	2
CO2	3	2	3	2	3	1	2	2	3
CO3	3	3	-	3	2	-	2	2	2
CO4	2	3	3	3	3	2	2	2	-
CO5	3	3	2	2	3	2	2	3	3
Articulation	2.8	2.6	2.5	2.2	2.6	1.7	2	2	2.5

Course Title:	**Infrastructure as Code & Automation**
Course Code:	**251VMT2S303**
Semester:	**III**
Credits:	**4**
Course Type	**Discipline Specific Elective**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course equips the students with the knowledge and hands-on experience to design, deploy, and manage cloud infrastructure using modern Infrastructure as Code (IaC) tools and automation practices. Students will explore version-controlled, repeatable infrastructure provisioning using tools like Terraform, Ansible, and scripting. Emphasis is placed on DevOps integration, scalability, configuration management, and CI/CD pipelines.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO No.	Course Outcome	BTL
CO1	Understand the principles and benefits of Infrastructure as Code and automation.	L1
CO2	Apply scripting and automation tools to provision and configure infrastructure.	L3
CO3	Analyze cloud environments and select appropriate IaC tools and architectures.	L4

CO4	Evaluate the performance, security, and compliance of automated infrastructure.	L5
CO5	Design end-to-end automated workflows integrating IaC with CI/CD pipelines.	L3

Course Modules

No. of Hours : 24

Module 1: Introduction to Infrastructure as Code (IaC) and DevOps Principles
Scope of Module: Covers the fundamental concepts of Infrastructure as Code and how it fits into modern DevOps practices.
Topics: Need for automation, benefits of IaC, declarative vs imperative approaches, pets vs cattle analogy, version control for infrastructure, introduction to DevOps lifecycle, IaC in cloud-native environments, provisioning vs configuration management, introduction to popular IaC tools (Terraform, Ansible, CloudFormation, Pulumi), key IaC design patterns, YAML and JSON fundamentals, infrastructure testing strategies, overview of CI/CD pipeline stages, IaC best practices, infrastructure drift.
Learning Outcome: *Students will be able to explain IaC principles and identify suitable use cases for automation in DevOps.*

No. of Hours : 24

Module 2: Infrastructure Provisioning with Terraform

Scope of Module:
Focuses on automating cloud infrastructure using Terraform for provisioning.

Topics:
Terraform installation and CLI, providers and backends, writing and structuring HCL code, defining resources and variables, modules and reusability, state management and remote backends, data sources and outputs, interpolation and expressions, lifecycle rules, input validation and type constraints, Terraform workspaces, secrets management, error handling, connecting with AWS, Azure, and GCP, infrastructure testing using Terratest.

Learning Outcome: *Students will be able to automate and manage infrastructure provisioning using Terraform.*

No. of Hours : 24

Module 3: Configuration Management with Ansible

Scope of Module:
Covers infrastructure configuration and orchestration using Ansible.

Topics:
Ansible architecture and inventory, ad-hoc commands, YAML and playbooks, roles and tasks, variables and facts, conditionals and loops, handlers and templates (Jinja2), dynamic inventory, configuration idempotency, Ansible Vault for secrets management, Ansible Galaxy and collections, working with Linux and Windows hosts, integrating with cloud modules (AWS/GCP), writing custom modules, use case-driven playbook development, error handling and debugging.

Learning Outcome: *Students will be able to write and execute Ansible playbooks for scalable infrastructure configuration.*

No. of Hours : 24

Module 4: Scripting and Automation using Shell and Python

Scope of Module:
Covers essential scripting skills for task automation in infrastructure operations.

Topics:
Bash scripting basics: variables, conditionals, loops, functions, cron jobs, text processing (grep, awk, sed), error handling and logging, Python for automation: subprocess, os, pathlib, shutil, working with APIs using requests, automation libraries (fabric, paramiko), file and service monitoring, integrating scripts with CI/CD tools, Git hooks and webhooks, automation of backups, patching, and logging.

Learning Outcome: *Students will be able to automate infrastructure tasks using shell and Python scripting techniques.*

No. of Hours : 24

Module 5: CI/CD Integration and Infrastructure Automation in Production
Scope of Module: Focuses on deploying infrastructure automation within CI/CD pipelines and managing real-world production scenarios.
Topics: CI/CD concepts: build, test, deploy stages, integrating Terraform and Ansible in pipelines, GitLab CI/CD, GitHub Actions, Jenkins pipelines, secrets and credentials management, environment-based deployments, policy-as-code using Sentinel/OPA, automated compliance and security checks, rollback and recovery strategies, monitoring and alerting setup, real-world DevOps case studies, troubleshooting automated infrastructure, scaling IaC projects in production, documentation and collaboration strategies.

Learning Outcome: *Students will be able to design and implement CI/CD-enabled automated infrastructure workflows for production systems.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Min no of pre recordings 10.

Prerequisites:

- Basic knowledge of Linux, cloud computing, and programming (Python or Bash).

Pedagogy / Teaching Methodology:

- ✓ Hands-on labs and live demonstrations
- ✓ Use-case based scripting and provisioning exercises
- ✓ GitHub-integrated project submissions
- ✓ Group activities and automation challenges

Certificate / Value Added Courses Recommended:

1. Terraform on AWS (Udemy)

2. Ansible for the Absolute Beginner (Udemy)
3. GitOps and IaC with ArgoCD (Pluralsight)
4. Infrastructure Automation with Python (LinkedIn Learning)

Suggested Readings:

- Yevgeniy Brikman, *Terraform: Up & Running*, O'Reilly
- Lorin Hochstein, *Ansible: Up and Running*, O'Reilly
- Kelsey Hightower, *The DevOps Handbook*
- Scott Chacon, *Pro Git*, Apress

Recommended Software/Tools:

1. Terraform CLI, Ansible
2. Visual Studio Code
3. Git, GitHub
4. Jenkins, GitHub Actions

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development

3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO - PO Mapping:

CO/PO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**
CO1	3	2	2	2	2	2	-	-	2
CO2	2	3	3	2	2	2	2	2	3
CO3	2	3	3	3	2	2	2	2	2
CO4	2	2	3	3	3	3	2	2	3
CO5	2	2	3	2	3	2	3	3	3
Articulation	2.2	2.4	2.8	2.4	2.4	2.2	2.25	2.25	2.6

Semester-IV

Course Title:	Site Reliability Engineering
Course Code:	251VMT2S401
Semester:	IV
Credits:	4
Course Type	Discipline Specific Elective
Learning Hours	120
Live Sessions	12 hours

Course Description:

This course introduces MCA students to Site Reliability Engineering (SRE), a specialized discipline within DevOps that focuses on maintaining high availability, scalability, and reliability of software systems. It equips students with skills in automation, observability, incident management, and infrastructure management. Learners will gain hands-on experience with modern SRE tools, explore real-world practices adopted by top tech companies, and design robust systems using industry-relevant strategies.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO No.	Course Outcome	BTL
CO1	Understand the principles of Site Reliability Engineering and its role in DevOps.	L1
CO2	Apply monitoring, alerting, and incident management practices.	L3

CO3	Analyze service-level indicators and implement error budgets.	L4
CO4	Evaluate reliability risks and design for scalability and fault-tolerance.	L5
CO5	Build automation pipelines and manage production-grade infrastructure using SRE tools.	L3

Course Modules

No. of Hours : 24

Module 1: Introduction to Site Reliability Engineering
Scope of Module: Introduces the core philosophy and evolution of Site Reliability Engineering, its distinction from traditional system administration, and how it fits within the DevOps lifecycle.
Topics: Introduction to SRE, history and evolution of SRE at Google, key tenets of SRE, comparison between DevOps and SRE, SRE vs. traditional operations, eliminating toil and manual tasks, SRE mindset and engineering principles, organizational alignment of SRE teams, responsibilities of an SRE, cultural shift from reactive to proactive operations, principles of automation and reliability, impact of SRE on software delivery, case studies from leading companies, overview of the Google SRE book, introduction to service lifecycle.
Learning Outcome: *Students will be able to explain the core principles and organizational role of SRE in delivering reliable software systems.*

No. of Hours : 24

Module 2: Monitoring, Alerting, and Incident Management
Scope of Module: Focuses on developing observability into systems through modern monitoring, structured alerting mechanisms, and effective incident response strategies.
Topics: Definition of monitoring and observability, understanding telemetry and instrumentation, working with metrics logs and traces, introduction to Prometheus and Grafana, setting up alert rules and thresholds, using OpenTelemetry for distributed tracing, defining and measuring SLIs, creating meaningful SLOs and SLAs, configuring alerting with Alertmanager, on-call rotations and escalation policies, playbooks and runbooks for incident response, root cause analysis (RCA) techniques, writing effective postmortems, incident triaging and communication, managing service health dashboards.
Learning Outcome: *Students will be able to configure observability tools, define alerts, and effectively manage incidents using structured processes.*

No. of Hours : 24

Module 3: Error Budgets, Reliability Metrics, and System Design
Scope of Module: Covers strategies for designing reliable systems using quantitative measures such as error budgets and service-level metrics while incorporating resilient architecture principles.
Topics:

Defining SLIs and their characteristics, calculating SLOs and error budgets, establishing error budget policies, balancing innovation and reliability, availability vs. reliability vs. durability, designing with failure in mind, chaos engineering fundamentals, circuit breakers and fallback mechanisms, retry and timeout strategies, load balancing concepts and tools, distributed system patterns and pitfalls, application of CAP theorem, failover design and redundancy, capacity planning for reliability, trade-offs in cost and reliability, implementing graceful degradation.
Learning Outcome: *Students will be able to evaluate system reliability through metrics and design resilient, fault-tolerant system architectures.*

No. of Hours : 24

Module 4: Infrastructure Automation and Continuous Delivery
Scope of Module: Delivers hands-on skills in automating infrastructure provisioning and software deployment using tools and methodologies central to DevOps and SRE.
Topics: Infrastructure as Code concepts, provisioning with Terraform, configuration management using Ansible, introduction to Kubernetes architecture, deploying apps with Helm charts, GitOps workflow and tools, CI/CD pipeline design principles, Jenkins and GitHub Actions integration, automated testing and code quality checks, blue-green and canary deployments, rollback strategies and feature flags, containerization using Docker, Docker Compose basics, using cloud-native CI/CD tools (e.g., AWS CodePipeline), managing secrets and environment variables, securing infrastructure-as-code pipelines.
Learning Outcome: *Students will be able to automate and deploy applications using industry-grade CI/CD pipelines and infrastructure tools.*

No. of Hours : 24

Module 5: SRE in Production and Scalability Practices
Scope of Module: Explores real-world production management, tuning performance, and scalability approaches critical to maintaining long-term reliability in complex systems.
Topics: Performance profiling and benchmarking strategies, capacity forecasting methods, scaling vertical vs. horizontal approaches, caching strategies and cache invalidation, using content delivery networks (CDNs), production logging and analysis with ELK stack, tracing request flows using Jaeger, load testing using Apache JMeter or k6, disaster recovery planning, building high availability architectures, runbook automation using Python scripts, cloud-native observability with AWS/GCP tools, cost monitoring and optimization techniques, handling production outages and mitigation, case studies on global-scale incidents.
Learning Outcome: *Students will be able to manage and optimize large-scale production systems using performance, cost, and availability strategies.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Min no of pre recordings 10.

Prerequisites:

- Basic knowledge of Linux, shell scripting, cloud fundamentals, and networking concepts.

Pedagogy / Teaching Methodology:

- ✓ Hands-on labs with tools like Prometheus, Grafana, and Kubernetes
- ✓ Simulation-based incident management exercises
- ✓ Infrastructure design projects
- ✓ Tool walkthroughs on Terraform, GitHub Actions, and Docker
- ✓ Analysis of real SRE case studies and industry war stories

Certificate / Value Added Courses Recommended:

1. Google Site Reliability Engineering Course – Coursera
2. Introduction to Site Reliability Engineering – edX (Linux Foundation)

3. Monitoring & Observability Fundamentals – New Relic University

Suggested Readings:

- Beyer, B., Jones, C., Petoff, J., & Murphy, N., *Site Reliability Engineering: How Google Runs Production Systems*, O'Reilly, 2016
- Beyer, B. et al., *The Site Reliability Workbook*, O'Reilly, 2018
- Burns, B., *Designing Distributed Systems*, O'Reilly, 2018

Recommended Software/Tools:

1. Prometheus
2. Grafana
3. Alertmanager
4. OpenTelemetry

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO - PO Mapping

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	2	1	2	1	1	1	2
CO2	2	2	3	3	2	-	3	3	2
CO3	3	3	2	3	2	-	2	2	2
CO4	2	3	3	3	3	2	2	2	2
CO5	3	3	3	2	3	2	3	3	3
Articulation	2.6	2.6	2.6	2.4	2.4	1.6	2.2	2.2	2.2

Course Title:	**DevSecOps**
Course Code:	**251VMT2S402**
Semester:	**IV**
Credits:	4
Course Type	Discipline Specific Elective
Learning Hours	120
Live Sessions	12 hours

Course Description:

This course introduces the students to the fundamentals and applications of DevSecOps—an integrated approach to embedding security throughout the software development lifecycle. Students will gain practical experience in implementing secure DevOps practices using automation tools, CI/CD pipelines, container orchestration, and compliance frameworks. The course emphasizes security automation, real-world case studies, and modern tools to help students deliver secure, scalable, and resilient software solutions.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO No.	Course Outcome	BTL
CO1	Understand the principles of DevOps and DevSecOps and their significance in modern software systems.	L1
CO2	Apply secure development practices in CI/CD pipelines and use automated security tools.	L3
CO3	Analyze vulnerabilities in applications and infrastructure using threat modeling and security tools.	L4
CO4	Evaluate security compliance and governance within cloud-native DevSecOps environments.	L5
CO5	Develop secure and automated DevSecOps pipelines using modern tools and platforms.	L3

Course Modules

No. of Hours : 24

Module 1: Introduction to DevOps and DevSecOps
Scope of Module: Provides foundational knowledge on DevOps principles, the transition to DevSecOps, and its growing relevance in secure software development lifecycles.
Topics: DevOps culture and principles, DevOps vs Agile vs Waterfall, CI/CD fundamentals, Introduction to DevSecOps, DevSecOps pipeline overview, Security as Code, Shift-left security, DevSecOps maturity models, Benefits of integrating security early, Common DevSecOps myths, Toolchain overview, Industry use cases of DevSecOps, Key stakeholders in DevSecOps, Challenges in DevSecOps adoption, Security testing types
Learning Outcome: *Students will be able to define DevSecOps and distinguish it from traditional DevOps and security models.*

No. of Hours : 24

Module 2: Secure SDLC and Threat Modeling
Scope of Module: Covers secure software development lifecycle concepts and how to identify potential threats early using modeling techniques and secure design patterns.
Topics: Software security fundamentals, Secure Software Development Life Cycle (SSDLC), STRIDE threat modeling, DREAD risk assessment model, OWASP Top 10

vulnerabilities, Secure coding guidelines, Input validation and sanitization, Common coding pitfalls, Authentication and authorization practices, Session management, Secure configuration, Error handling and logging, Security design patterns, Attack surface analysis, Code review strategies
Learning Outcome: *Students will be able to perform threat modeling and apply secure development strategies.*

No. of Hours : 24

Module 3: CI/CD Security and Automation Tools
Scope of Module: Focuses on automating security across the CI/CD pipeline using tools and scripting for continuous security integration and testing.
Topics: Overview of CI/CD pipelines, Jenkins for pipeline orchestration, GitLab CI/CD for automation, SAST (Static Application Security Testing), DAST (Dynamic Application Security Testing), IAST (Interactive Application Security Testing), Dependency scanning (Snyk, OWASP Dependency-Check), Secret detection and management, Git secret scanning, Container scanning with Trivy and Clair, Secure build practices, Secure artifact storage (Nexus, Artifactory), Configuration management security, Pipeline as Code (Jenkinsfile), Automated compliance checks
Learning Outcome: *Students will be able to implement security automation in CI/CD pipelines.*

No. of Hours : 24

Module 4: Container and Cloud Security
Scope of Module:

Introduces best practices and tools to secure cloud-native environments including containers, orchestrators, and Infrastructure as Code (IaC).
Topics: Introduction to containers and Docker security, Container image vulnerabilities, Dockerfile hardening practices, Kubernetes architecture overview, Kubernetes RBAC policies, Network policies in Kubernetes, Pod security policies (PSP), Admission controllers, Secrets management in Kubernetes, Infrastructure as Code (IaC) with Terraform, Secure IaC scanning tools (Checkov, TFSec), Cloud security benchmarks (CIS), Cloud IAM best practices, Cloud workload protection platforms (CWPP), Logging and monitoring in cloud
Learning Outcome: *Students will be able to analyze and enhance security in containerized and cloud environments*

.

No. of Hours : 24

Module 5: Compliance, Monitoring, and Case Studies
Scope of Module: Covers compliance frameworks, security governance, observability, and how organizations implement DevSecOps through real-world examples.
Topics: Overview of security compliance frameworks (ISO 27001, NIST, SOC2), GDPR and data protection, DevSecOps governance models, SIEM (Security Information and Event Management) tools, Vulnerability management lifecycle, Patch management strategies, Logging best practices, Centralized monitoring and alerting, Incident response workflows, Post-incident reviews and root cause analysis, DevSecOps KPIs

and metrics, Compliance-as-Code, Real-world DevSecOps failures, Success case studies from industry, DevSecOps strategy implementation roadmap
Learning Outcome: *Students will be able to evaluate compliance requirements and respond to security incidents in DevOps environments.*

Module	**No. of Pre-Recordings**
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Min no of pre recordings 10.

Prerequisites:

- Basic understanding of software development lifecycle, version control systems, and cloud computing fundamentals.

Pedagogy / Teaching Methodology:

- ✓ Hands-on lab exercises on CI/CD security
- ✓ Case studies on real-world security breaches
- ✓ Mini-projects involving container security and IaC

✓ Tool walkthroughs and demos

✓ Peer evaluations of pipeline security configurations

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. DevSecOps Essentials – edX
2. GitLab DevSecOps Course
3. Kubernetes Security Fundamentals – Udemy
4. OWASP DevSecOps Guideline

Suggested Readings:

- Jim Bird, DevSecOps: A leader's guide to producing secure software without compromising flow, feedback and continuous improvement, IT Revolution Press, 2020
- Tony Hsiang-Chih Hsu, Hands-On DevSecOps: A Guide to Implementing DevSecOps Pipelines, Packt Publishing, 2023

Recommended Software/Tools:

1. Jenkins
2. GitLab CI/CD
3. SonarQube
4. Trivy

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global

2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO - PO Mapping:

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	2	-	2	-	1	1	2
CO2	2	3	3	2	2	2	2	2	2
CO3	2	3	3	3	2	2	2	2	2
CO4	2	2	2	3	2	3	2	2	2
CO5	3	3	3	2	3	2	3	3	3
Articulation	2.4	2.6	2.6	2.5	2.2	2.25	2	2	2.2

Course Title:	**Microservices & Containerization**

Course Code:	**251VMT2S403**
Semester:	**IV**
Credits:	4
Course Type	Discipline Specific Elective
Learning Hours	120
Live Sessions	12 hours

Course Description:

This course provides the students with comprehensive understanding of microservices architecture and containerization technologies. Students will learn to design, develop, and deploy scalable, resilient microservices applications using industry-standard tools and practices. The course emphasizes practical implementation using Docker, Kubernetes, and modern CI/CD pipelines while addressing challenges in distributed systems, service orchestration, and DevOps integration.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO No.	Course Outcome	BTL
CO1	Understand the fundamentals of microservices architecture, containerization, and their evolution from monolithic systems.	L1
CO2	Apply containerization techniques using Docker and container orchestration with Kubernetes.	L3
CO3	Analyze performance, scalability, and resilience patterns in microservices-based applications.	L4

CO4	Evaluate API design, security approaches, and infrastructure requirements for microservices ecosystems.	L5
CO5	Design and implement end-to-end CI/CD pipelines for microservices deployment with monitoring and observability.	L3

Course Modules

No. of Hours :24

Module 1: Microservices Architecture Fundamentals
Scope of Module: Provides a foundation in microservices architecture principles, comparing them with traditional monolithic applications and exploring their benefits and challenges in modern software development.
Topics: Evolution from monolithic to microservices architecture, core principles and characteristics of microservices, bounded contexts and domain-driven design in microservices, single responsibility principle and service isolation, advantages and disadvantages of microservices, decomposition strategies for breaking down monoliths, communication patterns: synchronous vs asynchronous, service discovery and registry mechanisms, API gateway patterns and implementations, event-driven architecture in microservices, Backend for Frontend (BFF) pattern, database choices for microservices (SQL vs NoSQL), polyglot persistence and data isolation, infrastructure requirements for microservices, industry case studies of successful microservices implementations.

Learning Outcome: *Students will be able to articulate microservices architectural principles and design considerations for transitioning from monolithic applications.*

No. of Hours : 24

Module 2: Containerization with Docker
Scope of Module: Explores container technology with Docker, covering image creation, storage, networking, and best practices for containerizing applications.
Topics: Introduction to containerization and virtualization differences, Docker architecture and components (client, daemon, registry), writing effective Dockerfiles and build context, container lifecycle management (create, start, stop, remove), Docker Hub and container registries, Docker networking modes and inter-container communication, Docker volumes and persistent storage strategies, multi-stage builds for optimizing container images, Docker Compose for multi-container applications, Docker security best practices and vulnerability scanning, container resource limits and constraints, Docker logging and debugging techniques, environment variables and configuration management, container health checks and self-healing mechanisms, Dockerizing different application types (Node.js, Java, Python).
Learning Outcome: *Students will be able to build, deploy, and manage containerized applications using Docker and implement multi-container applications with Docker Compose.*

No. of Hours : 24

Module 3: Container Orchestration with Kubernetes
Scope of Module: Covers container orchestration with Kubernetes, focusing on deployment strategies, scaling, and managing complex microservices ecosystems.
Topics: Kubernetes architecture and components (control plane, nodes), Kubernetes objects: Pods, Services, Deployments, ReplicaSets and scaling applications in Kubernetes, namespaces and resource isolation, ConfigMaps and Secrets for configuration management, volumes and persistent storage in Kubernetes, Helm charts for application packaging and deployment, StatefulSets for stateful applications, DaemonSets and specialized workloads, Jobs and CronJobs for batch processing, resource requests and limits in Kubernetes, probes: liveness, readiness, and startup, Horizontal Pod Autoscaling (HPA) and Vertical Pod Autoscaling (VPA), deployment strategies: Blue/Green, Canary, Rolling updates, Kubernetes operators and custom resource definitions (CRDs).
Learning Outcome: *Students will be able to deploy, scale, and manage containerized microservices using Kubernetes orchestration platform.*

No. of Hours : 24

Module 4: API Gateway, Service Mesh, and Microservices Security
Scope of Module: Addresses advanced topics in microservices communication, networking, and security patterns essential for production-ready deployments.
Topics: API design principles and REST API best practices, GraphQL for microservices integration, API gateway implementations (Kong, Amazon API Gateway), OpenAPI

specifications and documentation, service mesh architecture and benefits, Istio service mesh capabilities and configuration, circuit breakers and bulkhead patterns, API rate limiting and throttling strategies, authentication and authorization in microservices, JWT, OAuth 2.0, and OpenID Connect implementation, mTLS and service-to-service encryption, API versioning strategies, CORS and API security considerations, resilience patterns: retry, timeout, fallback, distributed tracing with Jaeger and Zipkin.
Learning Outcome: *Students will be able to implement and secure communication between microservices using API gateways and service mesh technologies.*

No. of Hours : 24

Module 5: CI/CD, Monitoring, and Observability for Microservices
Scope of Module: Focuses on implementing continuous integration and delivery pipelines for microservices with comprehensive monitoring and observability solutions.
Topics: CI/CD pipeline components for microservices, Jenkins pipelines for microservices deployment, GitOps workflow with ArgoCD, Infrastructure as Code with Terraform for microservices, implementing blue/green and canary deployments, container image scanning and security in CI/CD, monitoring frameworks for microservices (Prometheus), Grafana dashboards for visualization, distributed logging architecture (ELK/EFK stack), log aggregation and analysis for microservices, alerting and notification strategies, SLIs, SLOs, and SLAs for microservices, chaos engineering principles for microservices, cost optimization for containerized applications, disaster recovery and high availability strategies.
Learning Outcome: *Students will be able to design and implement CI/CD pipelines with comprehensive monitoring and observability for microservices architectures.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Min no of pre recordings 10.

Prerequisites:

- Basic knowledge of web development and REST APIs
- Familiarity with Linux commands and shell scripting
- Understanding of networking fundamentals

Pedagogy / Teaching Methodology:

- ✓ Hands-on labs with containerization and orchestration tools
- ✓ Microservices case study analysis and implementation
- ✓ Team-based projects mimicking real-world scenarios
- ✓ Interactive demonstrations of DevOps workflows
- ✓ Industry expert guest lectures

Certificate / Value Added Courses Recommended:

1. Docker Certified Associate (DCA) preparation
2. Certified Kubernetes Administrator (CKA)
3. AWS Certified Developer - Associate (focus on container services)
4. Introduction to Microservices - https://www.edx.org/course/introduction-to-microservices
5. Docker and Kubernetes: The Complete Guide - https://www.udemy.com/course/docker-and-kubernetes-the-complete-guide/

Suggested Readings:

- Sam Newman, "Building Microservices: Designing Fine-Grained Systems", O'Reilly Media, 2021
- Brendan Burns, "Kubernetes Up & Running: Dive into the Future of Infrastructure", O'Reilly Media, 2019
- Chris Richardson, "Microservices Patterns", Manning Publications, 2018
- Nigel Poulton, "Docker Deep Dive", Packt Publishing, 2020

Recommended Software/Tools:

1. Docker Desktop/Engine
2. Kubernetes (Minikube, kind, K3s)
3. Jenkins, GitLab CI
4. Prometheus, Grafana

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO - PO Mapping:

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	2	-	2	1	1	1	2
CO2	3	3	3	2	2	1	2	2	2
CO3	2	3	3	3	2	-	2	2	2
CO4	2	3	3	3	3	2	2	2	2
CO5	3	3	3	2	3	2	3	3	3

Articulation	2.6	2.8	2.8	2.5	2.4	1.5	2	2	2.2

Course Title:	DevOps in AI & Big Data

Course Code:	251VMT2S404
Semester:	IV
Credits:	4
Course Type	Discipline Specific Elective
Learning Hours	120
Live Sessions	12 hours

Course Description:

This course provides the students with comprehensive knowledge of DevOps principles and practices specifically tailored for AI and Big Data environments. Students will learn to streamline the development, deployment, and operation of data-intensive and AI applications through automation, containerization, and continuous integration/delivery pipelines. The course emphasizes practical implementation of MLOps and DataOps methodologies using industry-standard tools while addressing unique challenges in managing large-scale data and AI model lifecycles.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO No.	Course Outcome	BTL
CO1	Understand DevOps principles and their application to AI and Big Data workflows.	L1
CO2	Apply containerization, orchestration, and CI/CD techniques to data-intensive applications.	L3
CO3	Analyze performance, monitoring, and optimization strategies for AI/ML pipelines.	L4

CO4	Evaluate security, compliance, and governance requirements for AI and Big Data systems.	L5
CO5	Design and implement end-to-end MLOps and DataOps workflows with version control and reproducibility.	L3

Course Modules

No. of Hours : 24

Module 1: Fundamentals of DevOps for AI and Big Data
Scope of Module: Establishes the core DevOps principles and their specific applications in AI and Big Data environments, highlighting the differences from traditional software development.
Topics: Introduction to DevOps for data-intensive applications, evolution from traditional DevOps to MLOps and DataOps, CI/CD/CT (Continuous Training) pipeline concepts, key challenges in AI and Big Data operations, infrastructure as code foundations, configuration management principles, version control for code, data, and models, DevOps metrics and KPIs for data projects, team roles and responsibilities in AI/Big Data DevOps, microservices architecture for data applications, data workflow orchestration tools, environment parity and reproducibility, automated testing strategies for data pipelines, DevOps culture and collaboration in data science teams, industry case studies of successful AI/Big Data DevOps implementations.
Learning Outcome: *Students will be able to articulate the core DevOps principles and strategies specific to AI and Big Data applications and explain how they differ from traditional DevOps approaches.*

No. of Hours :24

Module 2: Infrastructure and Resource Management for Data-Intensive Applications
Scope of Module: Focuses on provisioning, managing, and optimizing infrastructure resources for data processing at scale, using automation and cloud-native technologies.
Topics: Cloud platforms for AI/ML workloads (AWS, Azure, GCP), infrastructure as code using Terraform and CloudFormation, configuration management with Ansible and Chef, resource provisioning for GPU/TPU compute clusters, autoscaling strategies for variable workloads, spot instances and cost optimization techniques, virtual machine management for data processing, serverless computing for AI inference, storage solutions for large-scale datasets, network configuration for distributed training, high-performance computing (HPC) cluster setup, data lake and data warehouse infrastructure, disaster recovery strategies for data assets, infrastructure monitoring and logging, capacity planning for AI and Big Data systems.
Learning Outcome: *Students will be able to design, provision, and manage scalable infrastructure for AI and Big Data applications using infrastructure-as-code and cloud platforms.*

No. of Hours : 24

Module 3: Containerization, Orchestration, and CI/CD for Data Pipelines

Scope of Module: Covers container technologies and orchestration platforms specialized for data processing and ML workflows, with CI/CD integration for automated deployment.
Topics: Containerizing data science workloads with Docker, building optimized containers for ML frameworks, Docker Compose for multi-container data applications, container registries and versioning strategies, Kubernetes fundamentals for data orchestration, deploying data processing jobs on Kubernetes, Kubeflow components and workflows, continuous integration for data pipelines with Jenkins, GitLab CI/CD for ML projects, artifact management and model storage, feature stores and data versioning tools, deployment strategies (blue/green, canary) for ML models, CI/CD pipeline security for data applications, distributed training with container orchestration, automated testing for data pipelines and models.
Learning Outcome: *Students will be able to implement containerization, orchestration, and CI/CD pipelines for efficient development and deployment of AI and Big Data applications.*

No. of Hours : 24

Module 4: MLOps and DataOps: Tools and Practices
Scope of Module: Explores specialized tools and methodologies for managing machine learning model lifecycles and data processing workflows in production environments.
Topics: MLOps principles and frameworks, model versioning and lineage tracking, feature engineering automation, model registry implementation, experiment tracking with MLflow and Weights & Biases, hyperparameter optimization and AutoML, continuous training and model updates, A/B testing strategies for models, model monitoring and drift detection, data validation and quality checks, Apache Airflow for workflow orchestration, Databricks and Spark workflow management, data pipelines

with Apache Beam and Flink, metadata management and documentation, batch vs. real-time processing considerations.
Learning Outcome: *Students will be able to implement MLOps and DataOps practices to streamline development, deployment, and maintenance of AI models and data pipelines.*

No. of Hours :24

Module 5: Security, Observability, and Governance for AI and Big Data
Scope of Module: Addresses critical aspects of security, monitoring, governance, and compliance specifically for AI and data-intensive systems in production..
Topics: Security considerations for AI and Big Data systems, role-based access control implementation, encryption strategies for data at rest and in transit, secrets management in CI/CD pipelines, vulnerability scanning for containers and dependencies, compliance requirements for AI systems (GDPR, CCPA), monitoring and observability architecture, metrics collection with Prometheus and OpenTelemetry, visualization with Grafana and Kibana, distributed tracing for data pipelines, logging strategies for AI applications, alerting and incident response, governance frameworks for responsible AI, cost monitoring and optimization, managing technical debt in data systems.
Learning Outcome: *Students will be able to implement comprehensive security, monitoring, and governance practices for AI and Big Data systems in compliance with regulatory requirements..*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Min no of pre recordings 10.

Prerequisites:

- Basic understanding of cloud computing concepts
- Familiarity with Python programming and Linux command line
- Understanding of fundamental data processing and machine learning concepts

Pedagogy / Teaching Methodology:

- ✓ Hands-on labs with cloud platforms and DevOps tools
- ✓ Case study analysis of real-world AI/Big Data DevOps implementations
- ✓ Team-based projects simulating industry scenarios
- ✓ Interactive demonstrations of MLOps and DataOps workflows
- ✓ Guest lectures from industry practitioners

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. AWS Certified DevOps Engineer Professional
2. Azure DevOps Engineer Expert
3. Certified Kubernetes Administrator (CKA)
4. Google Professional Machine Learning Engineer
5. MLOps on Azure - https://docs.microsoft.com/en-us/learn/paths/mlops-fundamentals/
6. Kubeflow for Machine Learning - https://www.kubeflow.org/docs/started/

Suggested Readings:

- Danilo Sato, "Data Science in Production: Building Scalable Model Pipelines with Python", Apress, 2021
- Noah Gift, Alfredo Deza, "Practical MLOps: Operationalizing Machine Learning Models", O'Reilly Media, 2021
- Chris Fregly, Antje Barth, "Data Science on AWS", O'Reilly Media, 2021
- Mark Treveil, et al., "Introducing MLOps", O'Reilly Media, 2020
- Chip Huyen, "Designing Machine Learning Systems", O'Reilly Media, 2022

Recommended Software/Tools:

1. Docker,
2. Kubernetes,
3. Kubeflow,
4. MLflow,
5. DVC,

6. Jenkins,
7. GitLab CI

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO - PO Mapping:

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	2	1	2	1	1	-	2
CO2	3	3	3	2	2	-	2	2	3
CO3	3	3	3	3	2	1	2	2	2
CO4	2	3	2	3	2	3	2	2	2
CO5	3	3	3	3	3	2	3	3	3
Articulation	2.8	2.8	2.6	2.4	2.2	1.75	2	2.25	2.4

Course Title:	Strategic Leadership in IT

Course Code:	251VMT0C405
Semester:	IV
Credits:	2
Course Type	Skill Enhancement Course
Learning Hours	60
Live Sessions	12 hours

Course Description:

This course explores the critical role of leadership in driving strategic IT initiatives within organizations. It focuses on leadership theories, IT governance, digital transformation strategies, risk management, and innovation leadership. Practical case studies and simulations are integrated to build decision-making skills required for aligning IT with business strategies.

Course Outcomes (Cos)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Understand leadership theories and their application in IT strategy formulation.	L1
CO2	Evaluate and apply IT governance frameworks and practices.	L5
CO3	Design digital transformation strategies aligned with business goals.	L6
CO4	Identify and mitigate IT-related risks through strategic leadership.	L4
CO5	Demonstrate innovation and change management skills in IT leadership contexts.	L3

Course Modules

No. of Hours: 12

Module 1: Foundations of Strategic Leadership and IT Alignment

Scope of Module:
Covers the core leadership theories and IT-business alignment models.

Topics:
Defining leadership and management in the context of IT; Strategic leadership competencies; Leadership styles: transformational, transactional, situational; Vision building and strategic thinking in IT; Models of IT-business alignment (Henderson & Venkatraman Model); Organizational structures and their impact on IT leadership; Emotional intelligence in IT leadership; Ethical leadership in technology environments; Leading virtual and remote IT teams; Leadership maturity models; Case Study: IT leadership challenges in global enterprises.

Learning Outcome: *Students will be able to understand leadership theories and how to align IT strategies with broader business objectives.*

No. of Hours: 12

Module 2: IT Governance and Strategic Control

Scope of Module:
Focuses on frameworks and practices for effective IT governance.

Topics:
Introduction to IT governance and compliance; COBIT, ITIL, and ISO/IEC 38500 frameworks; Governance of Enterprise IT (GEIT) principles; Strategic decision rights and accountability in IT; Metrics and KPIs for IT governance; Managing outsourcing and vendor relationships; Sarbanes-Oxley Act and IT implications; Risk appetite and tolerance in IT governance; Blockchain's role in IT governance; Shadow IT and governance challenges; Audit management

and IT policy development; Practical Exercise: Designing an IT governance framework for an SME.

Learning Outcome: *Students will be able to evaluate, select, and apply IT governance frameworks to ensure strategic control and value delivery.*

No. of Hours: 12

Module 3: Digital Transformation Leadership

Scope of Module:
Develops skills to lead digital transformation initiatives.

Topics:
Drivers of digital transformation in organizations; Role of CIOs and CTOs in transformation efforts; Technology adoption frameworks (e.g., Diffusion of Innovations); Agile and DevOps in digital leadership; Managing cultural change during transformation; Leadership in cloud, AI, and emerging technologies adoption; Case Study: Successful vs. failed digital transformation initiatives.

Learning Outcome: *Students will be able to design strategic digital transformation initiatives aligned with business innovation needs.*

No. of Hours: 12

Module 4: IT Risk Management and Strategic Decision-Making

Scope of Module:
Focuses on managing strategic risks in IT leadership roles.

Topics:
Understanding IT-related risks: security, operational, reputational; Risk management lifecycle in IT projects; Enterprise risk management frameworks (COSO ERM); Crisis leadership and contingency planning; Ethical considerations and cybersecurity leadership; Legal and regulatory compliance (e.g., GDPR, HIPAA); Threat modeling and strategic defense planning;

Business continuity planning and IT disaster recovery leadership; Strategic handling of data breaches and incident response, Practical: IT risk assessment simulation

Learning Outcome: *Students will be able to analyze, manage, and mitigate IT risks strategically to ensure organizational resilience.*

No. of Hours: 12

Module 5: Innovation, Change Management, and Future of IT Leadership

Scope of Module:
Covers fostering innovation and managing technological change.

Topics:
Leading organizational innovation in IT; Change management models (Kotter's 8-Step, ADKAR); Talent development and IT leadership pipelines; Diversity, inclusion, and ethics in IT leadership; Future trends: Quantum Computing, Web3, and Leadership implications; Building a sustainable IT innovation culture; Intrapreneurship and fostering innovation within IT teams; Managing resistance to technological change; IT leadership in the age of Automation and RPA (Robotic Process Automation); Ethical considerations in AI and emerging tech leadership; Capstone: Strategic Leadership Plan for a tech startup.

Learning Outcome: *Students will be able to demonstrate strategic leadership skills in driving innovation, change management, and future-proofing IT organizations.*

Mo dule	No of Pre Recordings
1	11
2	12
3	10
4	12
5	11

Note: Min no of pre recordings 10.

Prerequisites (If Any):

➢ Basic knowledge of IT management concepts and organizational behavior.

Pedagogy / Teaching Methodology:

✓ Interactive case studies and leadership simulations

✓ Group strategy projects and role plays

✓ Workshops on IT governance frameworks

✓ Industry expert talks and guest lectures

Certificate / Value Added Courses Recommended:

1. IT Leadership Certificate by edX (MIT Sloan)
2. COBIT 2019 Foundation Certification
3. Digital Transformation Strategy by Coursera (Boston University)

Suggested Readings:

- John P. Kotter, Leading Change, Harvard Business Review Press, 2012.
- Peter Weill & Jeanne W. Ross, IT Governance: How Top Performers Manage IT Decision Rights for Superior Results, Harvard Business Review Press, 2004.
- George Westerman, Leading Digital: Turning Technology into Business Transformation, Harvard Business Review Press, 2014.

Recommended Software/Tools:

1. MS Project, Trello, Jira (for project leadership)
2. ServiceNow, SolarWinds (for IT governance)
3. Risk assessment tools (RSA Archer, RiskLens)

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO PO Mapping Table

CO/PO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**
CO1	3	2	1	-	2	1	-	2	1
CO2	3	3	2	1	2	1	-	2	1
CO3	2	3	3	2	3	2	1	3	2
CO4	3	2	2	2	2	1	1	2	2
CO5	2	2	2	1	3	2	1	2	2
Articulation	2.6	2.4	2	1.5	2.4	1.4	1	2.2	1.6

Course Matrix

ELECTIVE: NLP and LLM Development

Corporate Office
91/2, Dr. A N Krishna Rao Road V V Puram,
Bengaluru - 560 004
P: +91 80 4343 1000

Centre for Distance & Online Education
319, 25th Main, 17th Cross,
J P Nagar 6th Phase
Bengaluru – 560 078
P: +91 80 4343 0100

Elective: NLP and LLM Development

Semester-I

Course Code	Course Name	Course Category	CA	ESE	Credits
25VMT0C101	Mathematical Foundations for Computer Science	CC	30	70	4
25VMT0C102	Computing Concepts and Problem Solving using C	CC	30	70	4
25VMT0C103	Operating Systems	CC	30	70	4
25VMT0C104	Data Structures	CC	30	70	4
25VMT0L101	Problem Solving using C Lab	SEC	100	-	2
25VMT0L102	Data Structures Lab	SEC	100	-	2
26VMX0S105	AI Skills for Future	SEC	30	70	2
Total No of Credits					**22**

Semester-II

Course code	Course Name	Course category	CA	ESE	Credits
25VMT0C201	Advanced Database Systems	CC	30	70	4
25VMT0C202	Java Programming	CC	30	70	4
25VMT0C203	Web Technologies	CC	30	70	4
25VMT0C204	Cloud Computing Foundations	CC	30	70	4
25VMT0L201	Advanced Database Systems Lab	SEC	100	-	2
25VMT0L202	Java Programming Lab	SEC	100	-	2
25VMT0C205	Environmental, Social, and Governance (ESG) in IT	AECC	30	70	4
Total No of Credits					**24**

Semester-III

Course code	Course Name	Course category	CA	ESE	Credit
251VMT0C301	Artificial Intelligence and Machine Learning	CC	30	70	4
251VMT0C302	Python Programming	CC	30	70	4
251VMT5S303	Transformer Models & Attention Mechanisms	DSE	30	70	4
251VMT0L301	AI & Machine Learning Lab	SEC	100	-	2
251VMT0L302	Python Programming Lab	SEC	100	-	2
	Open Elective	OE	30	70	4
Total No of Credits					**2**

Semester-IV

Course code	Course Name	Course category	CA	ESE	Credit
251VMT5S401	Large-Scale AI Model Deployment	DSE	30	70	4
251VMT5S402	Multimodal AI	DSE	30	70	4
251VMT5S403	Reinforcement Learning	DSE	30	70	4
251VMT5S404	Fine-Tuning & Optimization	DSE	30	70	4
251VMT0C405	Strategic Leadership in IT	SEC	30	70	2
251VMT0P406	Project	Thesis			6
Total No of Credits					**2**

Semester I

Courses

Corporate Office
91/2, Dr. A N Krishna Rao Road V V Puram,
Bengaluru - 560 004
P: +91 80 4343 1000

Centre for Distance & Online Education
319, 25th Main, 17th Cross,
J P Nagar 6th Phase
Bengaluru – 560 078
P: +91 80 4343 0100

Course Title:	**Mathematical Foundations for Computer Science**
Course Code:	**25VMT0C101**
Semester:	**I**
Credits:	**4**
Course Type	Core Course
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course builds a strong mathematical base essential for various domains in computer science such as algorithms, cryptography, artificial intelligence, and data science. It covers discrete structures, logic, combinatory, graph theory, and number theory. Emphasis is laid on both theoretical understanding and application to computational problems.

Course Outcomes (COs)

Upon successful completion of this course, learners will be equipped to:

CO	**Course Outcome**	**BTL**
CO1	Comprehend the fundamentals of logic, sets, and relations used in computer science.	L3
CO2	Apply principles of combinatory and counting for solving algorithmic and real-world problems.	L3
CO3	Analyze and construct graphs, trees, and networks relevant to data structures and networking.	L4

CO4	Solve recurrence relations and use mathematical induction in algorithm design.	L3
CO5	Evaluate number theoretic and algebraic methods in cryptographic and coding applications.	L4

Course Module

No. of Hours:24

Module 1: Logic, Sets, and Relations
Scope of Module: *Provides foundational concepts in mathematical logic, set theory, and relations essential for modeling and designing algorithms.*
Topics: Propositions and logical operators, Truth tables, Tautology and contradiction, Predicates and quantifiers, Rules of inference computation, Logical equivalence, Methods of proof, Set theory basics, Venn diagrams, Set operations, Cartesian product of sets, Types of relations, Properties of relations, Equivalence and partial order relations, Closure of relations.
Learning Outcome: *Students will be able to construct logical arguments and model relationships using sets and relations.*

No. of Hours:24

Module 2: Combinatorics and Counting
Scope of Module: Focuses on combinatorial problem-solving techniques relevant to algorithm complexity and probability theory.

Topics: Basic counting principles, Addition and multiplication rules, Permutations and combinations, Pigeonhole principle, Principle of inclusion and exclusion, Binomial theorem, Pascal's triangle, Combinatorial identities, Multisets and combinations with repetition, Generating functions, Recurrence relations, Solving linear recurrence, Introduction to partitions, Derangements, Applications in probability.
Learning Outcome: *Students will be able to apply combinatorial reasoning to solve complex counting and probability problems.*

No. of Hours:24

Module 3: Graph Theory and Trees
Scope of Module: Introduces graph-based models to represent and solve problems in data structures, networks, and algorithms.
Topics: Basic concepts of graphs, Types of graphs, Graph representations, Degree of vertices, Paths and cycles, Connected components, Eulerian and Hamiltonian graphs, Planar graphs, Graph coloring, Matrix representation of graphs, Trees and their properties, Binary and n-ary trees, Tree traversal algorithms, Spanning trees, Minimum spanning tree algorithms (Kruskal, Prim)
Learning Outcome: *Students will be able to model and solve problems using graphs and trees in computer science.*

No. of Hours:24

Module 4: Recurrence Relations and Mathematical Induction
Scope of Module:

Covers recurrence relations and inductive reasoning techniques used in algorithm analysis and design.
Topics: Sequences and recurrence relations, Solving homogeneous linear recurrence, Non-homogeneous recurrence, Generating functions for recurrence, Divide-and-conquer relations, Recursion trees and Master's theorem, Mathematical induction, Strong induction, Well-ordering principle, Applications in algorithm correctness, Tower of Hanoi, Binary search, Fibonacci sequence, Ackermann's function.
Learning Outcome: *Students will be able to formulate and solve recurrence relations and validate algorithms using induction.*

No. of Hours:24

Module 5: Number Theory and Algebraic Structures
Scope of Module: Explores concepts in number theory and abstract algebra crucial for cryptography, hashing, and finite state systems.
Topics: Divisibility and Euclidean algorithm, Prime numbers and factorization, Congruence relations, Modular arithmetic, Fermat's little theorem, Euler's theorem, Chinese Remainder Theorem, Greatest common divisor (GCD), Least common multiple (LCM), Rings and fields basics, Group theory introduction, Subgroups and cyclic groups, Applications in cryptography, Hashing functions, Finite automata fundamentals
Learning Outcome: *Students will be able to apply number theoretic and algebraic techniques in areas like security and finite state systems.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites (If Any):

- Basic mathematics at undergraduate level, understanding of algebra and logic.

Pedagogy / Teaching Methodology:

- ✓ Conceptual and problem-based learning
- ✓ Interactive class exercises and proofs
- ✓ Real-world applications and modeling
- ✓ Assignments and quizzes using math software.

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. Discrete Mathematics by NPTEL: https://nptel.ac.in/courses/111106086
2. Mathematics for Computer Science by MIT Open Courseware: https://ocw.mit.edu

Suggested Readings:

- Kenneth H. Rosen, Discrete Mathematics and Its Applications, McGraw-Hill, 2017
- Tremblay and Manohar, Discrete Mathematical Structures, Tata McGraw-Hill, 2001
- C.L. Liu and D.P. Mohapatra, Elements of Discrete Mathematics, McGraw-Hill, 2010

- Online resources from IEEE Xplore, Springer Link for mathematical modeling and computation

Recommended Software/Tools (If applicable):

1. MATLAB/Octave
2. SageMath
3. Python (SymPy, NetworkX)
4. GeoGebra for visualization

Curriculum Development:

Sl.No	Identify and highlight in the module
1	Local, Regional, National and Global
2	Employability, Entrepreneurship and Skill Development
3	Value Added Courses (Module)
4	Integration of Digital Learning Tools
5	Crossing Cutting Edges (Ethics, Gender Equality, Sustainability and Environmental concerns)
6	Indian Knowledge System

CO/PO Mapping Table

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**
CO1	3	2	1	-	2	1	-	1	1
CO2	2	3	2	1	1	-	-	2	1
CO3	3	2	2	1	1	1	1	1	2
CO4	2	3	3	1	2	2	-	1	2
CO5	3	2	1	2	2	1	1	2	2
Articulation	2.6	2.4	1.8	0.8	1.6	1.25	1.0	1.4	1.6

Note: The Articulation is indicative of how a particular course maps to a PO.

Articulation is calculated as:

$$=\frac{Sum\ of\ CO\ Mappiung\ Levels\ for\ a\ perticular\ PO}{No.of\ COs\ that\ mapped\ to\ that\ PO}$$

Example:1

$$Articulation\ of\ PO1 = \frac{(3+2+3+2+3)}{5} = 2.6$$

Example: 2

$$Articulation\ of\ PO7 = \frac{(1+1)}{2} = 1.0$$

JAIN CENTRE FOR DISTANCE AND ONLINE EDUCATION — JAIN ONLINE

Course Title:	**Computing concepts and problem solving in C**
Course Code:	**25VMT0C102**
Semester:	**I**
Credits:	**4**
Course Type	Core Course
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course covers Advanced concepts of computing concepts and problem solving in C and their applications in computer science. Students will develop problem-solving abilities and practical programming skills essential for core areas such as algorithms, systems programming, software development, and emerging technologies.

Course Outcomes (COs)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Comprehend the advanced Programming constructs	L3
CO2	Apply File Operations and handle errors effortlessly	L3
CO3	Integrate linux Programming with C Scripting	L4
CO4	Demonstrate C System Programming	L3
CO5	Evaluate real-world scenarios and Case Studies	L5

Course Modules

No. of Hours:24*

Module 1: Fundamentals of Computing and C Programming
Scope of Module: Introduces basic computing concepts, algorithm design, and foundational elements of the C programming language
Topics: Basics of Computers: Architecture, memory, input/output units; Introduction to Algorithms and Flowcharts; C Programming Basics: Structure, syntax, variables, constants, data types; Operators: Arithmetic, relational, logical, bitwise, assignment, conditional; Compilation and Execution process.
Learning Outcome:*Students will be able to describe computing concepts, write simple C programs, and understand compilation processes.*

No. of Hours:24

Module 2: Control Structures, Functions, Arrays, Strings, and Modular Programming
Scope of Module: Explores problem-solving strategies using control flow, modular programming, and essential data structures.
Topics: Control Structures: if, if-else, nested if, switch-case; Loops: for, while, do-while, nested loops; Functions: Declaration, definition, call, recursion; Scope, Storage Classes (auto, static, extern, register); Modular Programming: Header files,

preprocessor directives (#define, #include); Arrays: 1D, 2D arrays, array manipulation; Strings: String functions (strlen, strcpy, strcat, strcmp), string manipulation.
Learning Outcome:*Students will be able to apply control structures, modularize code, and manipulate arrays and strings effectively.*

No. of Hours:24

Module 3: Pointers, Structures, Dynamic Memory Allocation, and File Handling
Scope of Module: Provides in-depth understanding of memory management, structured data, and file operations in C.
Topics: Introduction to Pointers: Pointer arithmetic, pointers and arrays; Pointers to Functions and Pointers to Structures; Structures and Unions: Declaration, definition, accessing members; Dynamic Memory Allocation: malloc, calloc, realloc, free; File Handling: File operations (read, write, append), file pointers; Case Study: Creating a file-based student management system.
Learning Outcome:*Students will be able to manipulate memory using pointers, manage structured data, and perform file operations.*

No. of Hours:24

Module 4: ***System-Level Programming and Real-Time Applications in C***

Scope of Module: *Focuses on industry-relevant applications, system programming concepts, and real-time project development.*
Topics: *Command-line Arguments and Environment Variables; Error Handling (perror, strerror) and Debugging Techniques; Mini Projects: Text-based Games (Tic-Tac-Toe), Simple Shell Implementation, Real-Time Data Logger for IoT (file-based); Case Study: Simulating a Banking System.*
Learning Outcome: *Students will be able to design and implement system-level programs and real-time applications using C.*

No. of Hours:24

Module 5: Advanced Industry Applications and Case Studies
Scope of Module: Encourages application of C programming skills in solving industry-specific problems and case studies.
Topics: Writing basic command-line tools; Simple network programs (client-server); Debugging and optimizing small C programs; Working with file compression and encryption tools; Implementing simple database file storage; Case Study: Building a basic calculator or file management tool; Mini Project: Simple automation program, data logger, or system monitor.
Learning Outcome*: Students will be able to apply C programming skills to solve advanced, industry-relevant problems through case studies and projects.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites (If Any):

- Basic knowledge of computer fundamentals.

Pedagogy / Teaching Methodology:

- ✓ Lectures, Hands-on Lab Sessions, Problem-solving Tasks, Case-based Assignments, Group Projects

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. NPTEL, Coursera, edX, MIT OCW — based on relevance to topics.

Suggested Readings:

- Standard textbooks, research articles, online tutorials and eBooks related to the subject.

Recommended Software/Tools (If applicable):

1. GCC Compiler, Visual Studio Code, Linux Terminal, C IDEs

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO PO Mapping Table

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**
CO1	3	3	2	2	2	1	-	1	2
CO2	3	2	2	3	2	1	-	1	1
CO3	2	3	2	2	-	1	1	1	1
CO4	3	2	3	2	2	1	-	1	2
CO5	2	2	2	3	2	2	-	2	2

Articulation	2.6	2.4	2.1	2.4	2	1.1	1	1.1	1.6

Course Title:	**Operating Systems**
Course Code:	**25VMT0C103**
Semester:	**I**
Credits:	**4**
Course Type	Core Course
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides an in-depth exploration of Advanced operating system concepts, focusing on the design, implementation, and management of modern operating systems. Topics include process synchronization, memory management, file systems, distributed systems, and security mechanisms. Practical assignments and case studies will reinforce theoretical knowledge, preparing students for real-world system-level programming and research.

Course Outcomes(COs)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Apply process management and synchronization techniques.	L3
CO2	Implement memory Management Techniques	L3
CO3	Implement File Management Techniques	L3
CO4	Design solutions using modern OS tools and case studies in distributed systems.	L3

CO5	Evaluate modern tools and techniques in real-world OS environments.	L5

Course Modules:

No. of Hours:24

Module 1: Introduction to Operating Systems and Process Management
Scope of Module: Introduces core OS concepts, structures, and the fundamentals of process management.
Topics: Overview of Operating Systems: Definitions, functions, types (batch, time-sharing, distributed, real-time); OS structure and components; System calls and APIs; Microkernel vs Monolithic Kernel architectures; System Boot Process and Initialization (GRUB, systemd); Process concepts, process states, process control block (PCB); Process scheduling algorithms (FCFS, SJF, Round Robin, Priority); Context switching; Process scheduling in multi-core and NUMA systems; Case studies on modern OS architectures (Linux, Windows).
Learning Outcome: *Students will be able to describe OS structures, functions, and manage basic process scheduling scenarios.*

No. of Hours:24

Module 2: Process Synchronization, Deadlock, and CPU Scheduling
Scope of Module: Covers critical section problems, synchronization mechanisms, deadlock handling, and CPU scheduling strategies.

Topics: Critical section problem, Peterson's solution, semaphores, monitors; Spinlocks, Mutexes, and Condition Variables; Hardware support for synchronization (Test-and-Set, Compare-and-Swap); Classic synchronization problems (producer-consumer, readers-writers, dining philosophers); Deadlock characterization, prevention, avoidance (Banker's algorithm), detection, recovery, and resource preemption; CPU scheduling strategies (multilevel queue, multilevel feedback queue); Real-world scheduling policies (CFS in Linux, Windows scheduler comparison); Case studies on synchronization in Linux kernel.
Learning Outcome: *Students will be able to apply synchronization techniques and deadlock handling strategies in process management.*

No. of Hours:24

Module 3: Memory Management, Virtual Memory, and Storage Systems
Scope of Module: Focuses on memory organization, virtual memory, and file systems management.
Topics: Memory management strategies: contiguous allocation, paging, segmentation; Virtual memory: demand paging, page replacement algorithms (LRU, FIFO, Optimal); Thrashing and performance tuning; Memory-mapped I/O and Direct Memory Access (DMA); Swap space management and tuning; File system interface and implementation, allocation methods (contiguous, linked, indexed); Directory structures, file system mounting, journaling file systems; Advanced file system topics: RAID levels, ZFS/Btrfs file systems; Security

implications in storage systems (encryption at rest); Case study: EXT4 file system analysis.
Learning Outcome:*Students will be able to analyze memory and storage management strategies in OS environment*

No. of Hours:24

Module 4: Virtualization, Containerization, and Real-Time Systems
Scope of Module: Explores virtualization technologies, container-based systems, and introduces real-time OS concepts.
Topics: Virtualization concepts, hypervisors (Type 1 and Type 2), virtual machine monitors (VMWare, KVM); Nested Virtualization (running VMs inside VMs); Containerization with Docker, Kubernetes orchestration basics; Container Security (namespaces, cgroups, seccomp); Comparison of VMs vs Containers; Performance tuning of VMs and Containers; Introduction to real-time operating systems (RTOS), real-time scheduling algorithms; RTOS performance metrics and real-world RTOS examples (FreeRTOS, VxWorks); Case study: Docker-based microservices in cloud environments.
Learning Outcome:*Students will be able to evaluate virtualization and containerization technologies and apply real-time scheduling concepts.*

No. of Hours:24

Module 5: Security, Protection Mechanisms, and Advanced OS Case Studies
Scope of Module: Focuses on OS security, protection models, and advanced industrial applications of operating systems.
Topics: OS security principles, access control models (DAC, MAC, RBAC); Kernel hardening techniques (grsecurity, seccomp); Network security in OS: Firewall configurations (iptables, nftables); Secure Boot and Trusted Platform Module (TPM); Security vulnerabilities and mitigation strategies; Encryption, authentication, and auditing in OS; Introduction to SELinux, AppArmor; Forensics in OS environments (log analysis, intrusion detection); Case studies on OS security breaches and their prevention; Mini Project: Design a simple OS-level security module, process monitoring utility, or Intrusion Detection System (IDS) script.
Learning Outcome:*Students will be able to assess and implement security models and protection mechanisms in modern OS environments.*

Module	**No. of Pre-Recordings**
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10.

Prerequisites (If Any):

➢ Basic knowledge of computer fundamentals.

Pedagogy / Teaching Methodology:

✓ Lectures, Hands-on Lab Sessions, Problem-solving Tasks, Case-based Assignments, Group Projects

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. NPTEL, Coursera, edX, MIT OCW — based on relevance to topics.

Suggested Readings:

- Standard textbooks, research articles, online tutorials and eBooks related to the subject.

Recommended Software/Tools (If applicable):

1. GCC Compiler, Visual Studio Code, Linux Terminal, C IDEs

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools

5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO PO Mapping Table

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**
CO1	3	2	2	2	2	1	-	1	2
CO2	2	3	3	2	2	1	1	2	1
CO3	3	2	2	2	2	1	1	2	2
CO4	2	3	3	2	2	2	1	1	2
CO5	3	2	2	3	3	2	1	2	2
Articulation	2.6	2.4	2.4	2.2	2.2	1.4	1	1.4	1.6

Course Title:	**Data Structures**
Course Code:	**25VMT0C104**
Semester:	**I**
Credits:	**4**
Course Type	Core Course
Learning Hours	120
Live Sessions	**12 hours**

Course Description

This course introduces Advanced Concepts of data structures, algorithms, and their practical use in computing applications. Students will learn how to implement, analyze, and apply various linear and non-linear data structures to build efficient and scalable software systems

Course Outcomes(Cos)

Upon successful completion of this course, learners will be equipped to:

CO	**Course Outcome**	**BTL**
CO1	Understand the fundamentals of advanced linear and data structures.	L1
CO2	Apply appropriate data structures for solving complex computational problems.	L3
CO3	Apply tree graph techniques in software solutions.	L3
CO4	Implement hashing techniques in software solutions.	L3
CO5	Evaluate efficient programs using modern tools and real-world case studies.	L5

Note: Implementation can be done in C/C++

Course Modules

No. of Hours:24

Module 1: : Linear Data Structures
Scope of Module Covers implementation and usage of linear storage structures.
Topics: Arrays, Stacks (array/linked), Queues (normal, circular), Dequeues, Linked list operations (SLL, DLL, CLL), Stack/Queue with linked lists, Applications in expression evaluation, Bracket matching, Recursion stack, Queue in scheduling, List-based simulation, Sparse matrices, Time/space complexity basics, Array vs list.
Learning Outcome: *Apply linear structures in solving algorithmic problems.*

No. of Hours:24

Module 2: Trees and Binary Trees
Scope of Module Focuses on hierarchical data structures with operations and applications of binary trees and advanced trees.
Topics: Tree definitions and terminologies, Binary tree representation, Binary tree traversals (inorder, preorder, postorder), Binary search tree (BST), Insertion in

BST, Deletion in BST, Balanced trees (AVL trees), Height-balanced trees, Expression trees, Threaded binary trees, Heap trees, Min-heap and max-heap, Priority queue using heap, B-Trees, Applications of trees.
Learning Outcome: *Students will be able to implement tree structures and use them for hierarchical data processing.*

No. of Hours:24

Module 3: Graphs and Applications
Scope of Module: Introduction to graphs and path-finding algorithms.
Topics: Graph representation (adjacency list/matrix), BFS, DFS, Topological sorting, Dijkstra's algorithm, Bellman-Ford, Floyd-Warshall, Kruskal's algorithm, Prim's algorithm, Network flow, Graph coloring, Euler and Hamiltonian paths, Graph traversals for cycles, Applications in networks, Graph problem modeling.
Learning Outcome: *Analyze and solve graph-based problems using suitable algorithms*

No. of Hours:24

Module 4: Hashing and Searching Structures
Scope of Module: Efficient search, hash, and map-based data access.

Topics: Hash tables, Hash functions, Collision resolution (chaining, open addressing), Load factor, Linear probing, Double hashing, Bloom filters, Dictionaries, Sets, Unordered maps, Binary search trees vs hashing, Search time analysis, Skip lists, LRU Cache design, Memory-optimized hashing, Applications in OS/DB.
Learning Outcome: *Design and implement efficient search and mapping structures.*

No. of Hours:24

Module 5: Case Studies and Tools
Scope of Module: Tools and case implementations using advanced DS.
Topics: STL containers in C++ (vector, list, map, set), Data structure libraries in C++, Visualization tools (VisuAlgo, Graphviz), IDEs (CLion, Eclipse), Real-world DS projects, Competitive programming platforms, Time-space profiler tools, LeetCode/Codeforces case studies, DS in AI/ML preprocessing, Game development DS, Cache designs, Real DBMS indexing, Distributed DS structures.
Learning Outcome: *Evaluate and apply real-world DS solutions using modern tools and case studies.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites (If Any):

- Basic knowledge of programming and algorithms.

Pedagogy / Teaching Methodology:

- ✓ Interactive Lectures
- ✓ Hands-on Programming Labs
- ✓ Assignments and Case Studies
- ✓ Mini Projects

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. NPTEL: Data Structures and Algorithms
2. Coursera: Data Structures by UC San Diego

Suggested Readings:

- Seymour Lipschutz, *Data Structures*, McGraw-Hill
- Ellis Horowitz, Sartaj Sahni, *Fundamentals of Data Structures*, Universities Press
- Mark Allen Weiss, *Data Structures and Algorithm Analysis in C*, Pearson Education

- Online resources and research papers related to data structures

Recommended Software/Tools (If applicable):

1. C/C++ Compiler
2. Visual Studio Code / Code::Blocks
3. Online IDEs (Repl.it, GeeksforGeeks IDE)
4. Visualgo.net for algorithm visualization

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO PO Mapping Table

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**
CO1	3	2	3	2	2	3	-	2	1
CO2	3	3	3	2	2	2	1	2	2
CO3	2	2	3	3	2	2	3	1	2
CO4	3	2	3	2	2	-	2	1	2
CO5	2	3	2	3	2	2	-	2	1
Articulation	2.6	2.4	2.8	2.4	2	2.2	2	1.6	1.6

Course Title:	AI Skills for Future
Course Code:	26VMX0S105
Semester:	I
Credits:	2
Course Type	Skill Enhancement Course
Learning Hours	60
Live Sessions	12 hours

Course Outcomes:

CO1: Understand Generative AI and LLMs in depth
CO2: Create text, image, and voice content using AI
CO3: Automate daily tasks and business workflows
CO4: Build AI agents that act independently
CO5: Design and deploy simple AI apps without coding

Course Modules
Module 1: Generative AI Prompt Engineering for Real Workflows
What is Generative AI and how it's transforming industries, Structure of an effective prompt: *Rol e*
→ *Task* → *Context* → *Output,* Common prompting mistakes and how to fix them, How large language models (LLMs) like ChatGPT, Claude, and Google AI Studio work, advanced prompti ng (few-shot, iterative, style-based), Using your own files in ChatGPT and Claude (RAG basics), Research and summarization using AI
Learning Outcome: Students will be able to understand how Generative AI works and master the art of prompting.

Module 2: Creative AI and Voice & Audio Creation
Understanding AI image generation, creating posters, campaign assets, and product visuals, building brand identity using Krea.ai, Generating realistic voiceovers using ElevenLabs, Script

writing and tone control with ChatGPT & Claude, Integrating voice into short videos or podcasts ***Learning Outcome:*** Students will be able to use AI to generate visuals, voiceovers, and creative storytelling content.

Module 3: AI Automation & Productivity

Logic of triggers, actions, and workflows, setting up automation between apps (Gmail, Sheets, Drive), Connecting OpenAI to Make.com for intelligent automation, creating social media Management Workflow and research automation flows, automating daily reports and reminders, building personal productivity dashboards, Integrating Claude and ChatGPT into existing workflows

Learning Outcome: Automate daily workflows and tasks using AI and no-code tools.

Module 4: Advanced AI Agents & Decision Flows

Introduction to n8n as a visual AI workflow builder, understanding nodes, triggers, and data passing, Using OpenAI node for smart responses and actions, designing multi-step reasoning workflows, Connecting Google Sheets, Slack, and other APIs, building autonomous agents that perform specific tasks

Learning Outcome: Students will be able to build AI agents that can think, decide, and act automatically.

Module 5: AI App Building & Deployment

Fundamentals of UI/UX for AI apps, structuring front-end and AI back-end flows, designing app logic using n8n and Make.com, integrating OpenAI and Claude APIs to your apps, handling user inputs and dynamic responses, using Google AI Studio to experiment with models, hosting and testing your AI app online, adding analytics and monitoring usage, final demo preparation and peer feedback

Learning Outcome: Students will be able to learn to design, connect, and deploy simple AI apps using no-code tools

Semester-II

Corporate Office
91/2, Dr. A N Krishna Rao Road V V Puram,
Bengaluru - 560 004
P: +91 80 4343 1000

Centre for Distance & Online Education
319, 25th Main, 17th Cross,
J P Nagar 6th Phase
Bengaluru – 560 078
P: +91 80 4343 0100

Course Title:	**Advanced Database Systems**
Course Code:	**25VMT0C201**
Semester:	**II**
Credits:	**4**
Course Type	Core Course
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course delves into advanced concepts and techniques in database systems, focusing on design, implementation, optimization, and emerging trends. Topics include distributed databases, NoSQL systems, data warehousing, transaction management, and big data technologies. The course emphasizes practical applications through case studies and hands-on projects.

Course Outcomes (COs)

Upon successful completion of this course, learners will be equipped to:

CO	**Course Outcome**	**BTL**
CO1	Understand advanced database models and architectures.	L1
CO2	Design and optimize relational and NoSQL databases.	L3
CO3	Implement transaction management and concurrency control.	L4
CO4	Evaluate distributed database systems and their challenges.	L5
CO5	Apply data warehousing and big data technologies.	L3

Course Modules

No. of Hours:24

Module 1: Advanced Database Models and Architectures
Scope of Module: Covers advanced database models, including object-oriented, object-relational, and NoSQL databases, and their architectures.
Topics: Review of relational database systems; Object-oriented databases; Object-relational databases; NoSQL database overview; Document databases (MongoDB); Key-value databases (Redis); Column-family databases (Cassandra); Graph databases (Neo4j); NewSQL systems; Polyglot persistence; CAP theorem; BASE properties; Cloud-native database systems; Multi-model databases; Case study: MongoDB vs. PostgreSQL.
Learning Outcome: *Students will be able to compare and contrast different database models and select appropriate architectures for specific applications*

No. of Hours:24

Module 2: Database Design and Query Optimization
Scope of Module: Focuses on advanced techniques for database design, indexing, query optimization, and performance tuning
Topics: Advanced normalization (1NF–5NF); Denormalization strategies; Physical vs. logical design Partitioning strategies; Indexing: B-tree, hash, bitmap; Covering indexes and index-only scans Query execution plans; Cost-based query optimization; Heuristic-based optimization; Materialized views; Query rewriting

techniques; Adaptive query optimization; Statistics collection and use; SQL performance tuning tools; Hands-on: SQL tuning with EXPLAIN and ANALYZE.
Learning Outcome: *Students will be able to design efficient database schemas and optimize query performance.*

No. of Hours:24

Module 3: Transaction Management and Concurrency Control
Scope of Module: Explores transaction properties, concurrency control mechanisms, and recovery techniques in database systems
Topics: ACID properties; Transaction lifecycle; Serializability and schedules; Lock-based concurrency control; Deadlock detection and prevention; Timestamp ordering; Optimistic concurrency control; Two-phase locking; Multiversion Concurrency Control (MVCC); Isolation levels and anomalies; Serializable snapshot isolation; Write-ahead logging; Checkpointing; Recovery methods; Case study: PostgreSQL concurrency and recovery.
Learning Outcome: *Students will be able to implement and analyze transaction management and concurrency control mechanisms.*

No. of Hours:24

Module 4: Distributed Database Systems
Scope of Module:

Examines distributed database architectures, data fragmentation, replication, and consistency models.
Topics: Distributed database architecture; Client-server vs. peer-to-peer; Horizontal and vertical fragmentation; Data allocation and replication; Distributed concurrency control; Distributed transactions; Two-phase commit protocol; Distributed query processing. Query decomposition and localization; CAP theorem in distributed databases; Eventual consistency models; Conflict resolution strategies; Fault tolerance and availability; Case study: Google Spanner; Hands-on: Setting up a distributed Cassandra cluster.
Learning Outcome: *Students will be able to evaluate distributed database systems and address their challenges.*

No. of Hours:24

Module 5: Data Warehousing and Big Data Technologies
Scope of Module: Introduces data warehousing concepts, ETL processes, OLAP, and big data technologies
Topics: Data warehousing concepts; Star schema vs. snowflake schema; Fact and dimension tables ETL process design; Data cleaning and transformation; Data staging and loading; OLAP operations and cubes; Hadoop ecosystem (HDFS, MapReduce); Apache Hive and Pig Apache Spark fundamentals; NoSQL for Big Data: HBase and Cassandra; Data lakes and lakehouse architecture; Real-time data pipelines (Kafka + Spark Streaming); Columnar storage formats (Parquet, ORC); Mini-project: Building a Spark-based data pipeline

Learning Outcome: *Students will be able to apply data warehousing and big data technologies for analytical processing*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites (If Any):

- Basic knowledge of database management systems (DBMS) and SQL.

Pedagogy / Teaching Methodology:

- ✓ Conceptual and problem-based learning
- ✓ Hands-on labs and case studies
- ✓ Group projects and presentations
- ✓ Quizzes and assignments using database tools

Certificate / Value Added Courses Recommended (with Free Resource Links):

1. Advanced Database Systems by NPTEL: https://nptel.ac.in/courses/106105171
2. Database Systems by Stanford Online: https://online.stanford.edu/courses/soe-ydatabases-database-systems

Suggested Readings:

- Raghu Ramakrishnan and Johannes Gehrke, *Database Management Systems*, McGraw-Hill, 2003.
- Hector Garcia-Molina, Jeffrey D. Ullman, and Jennifer Widom, *Database Systems: The Complete Book*, Pearson, 2008.
- Martin Kleppmann, *Designing Data-Intensive Applications*, O'Reilly, 2017.

Recommended Software/Tools:

1. PostgreSQL, MongoDB, Cassandra
2. Hadoop, Spark
3. SQL Developer, Tableau

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO/PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	-	2	1	-	1	-
CO2	2	3	2	1	1	-	-	2	-
CO3	3	2	2	1	1	1	1	1	2
CO4	2	3	3	1	2	2	-	2	2
CO5	3	2	1	2	2	1	1	2	2
Articulation	2.6	2.6	1.8	1.25	1.6	1.25	1	1.4	2

Course Title:	**Java Programming**
Course Code:	**25VMT0C202**
Semester:	**II**
Credits:	**4**
Course Type	Core Course
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides an in-depth understanding of Java programming, covering object-oriented principles, core APIs, multithreading, networking, and modern frameworks. Emphasis is placed on practical implementation through projects, case studies, and hands-on labs to build robust applications.

Course Outcomes (Cos)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Understand object-oriented principles in Java.	L1
CO2	Develop applications using Java core APIs and collections.	L3
CO3	Implement multithreading and exception handling.	L4
CO4	Design GUI applications and network-based solutions.	L5
CO5	Apply Java frameworks (Spring Boot) for enterprise applications.	L3

Course Modules

No. of Hours:24

Module 1: Object-Oriented Programming in Java
Scope of Module: Covers fundamental OOP concepts, Java syntax, and class design
Topics: Introduction to Java and JVM; Data types, operators, and control structures; Class structure and object creation; Encapsulation and constructors; Inheritance and method overriding; Polymorphism and dynamic binding; Interfaces and abstract classes; Package creation and import; Access modifiers and scope; Static vs. instance members; Exception handling: try-catch-finally; Custom exceptions; Assertions and error handling; Object class and method overriding; Case study: Modular banking system design.
Learning Outcome: *Students will be able to apply OOP principles such as encapsulation, inheritance, and polymorphism to design structured and reusable Java applications.*

No. of Hours:24

Module 2: Java Core APIs and Collections
Scope of Module: Focuses on Java standard libraries and data structures.
Topics: String manipulation and immutability; Regular expressions; Wrapper classes and autoboxing Generics in Java; ArrayList, LinkedList implementations; HashSet, TreeSet, LinkedHashSet usage; HashMap, TreeMap, LinkedHashMap differences; Iterators, ListIterators, and enhanced for-loop; Comparable vs.

Comparator; Collections utility class; File handling with java.io and java.nio; Object serialization and deserialization; Properties and configuration files; Practical: Inventory management system; Debugging collections and memory leaks.
Learning Outcome: *Students will be able to leverage core Java libraries and the Collections Framework to build efficient, maintainable applications with complex data structures.*

No. of Hours:24

Module 3: Multithreading and Concurrency
Scope of Module: Explores concurrent programming in Java.
Topics: Thread lifecycle and states; Thread creation via Thread class; Runnable interface and Lambda expressions; Thread prioritization and yielding; Synchronized methods and blocks; Intrinsic locks and synchronized collections; Volatile keyword and atomic variables; Deadlock scenarios and prevention strategies; wait(), notify(), and notifyAll(); Thread pooling and ExecutorService; Callable and Future interfaces; ConcurrentHashMap and thread-safe queues; ForkJoinPool basics; Case study: Multithreaded chat application; Concurrency debugging tools and techniques.
Learning Outcome: *Students will be able to implement multithreaded programs and apply Java concurrency tools to ensure safe, scalable execution of parallel tasks.*

No. of Hours:24

Module 4: GUI and Networking
Scope of Module: Introduces Java FX and network programming.
Topics: Introduction to JavaFX and GUI architecture; SceneBuilder and JavaFX controls; Layout managers: BorderPane, VBox, HBox; Event-driven programming in Java; Creating forms and validation; Swing basics and component lifecycle; Action listeners and adapters; Socket programming: TCP; Datagram programming: UDP; HTTP communication using HttpClient Using REST APIs in Java; JDBC for database connectivity; CRUD operations with JDBC Mini-project: GUI Weather App with REST API; Threading in GUI and networking applications
Learning Outcome: *Students will be able to design and build user-friendly GUI applications integrated with network communication and backend services using Java.*

No. of Hours:24

Module 5: Java Frameworks and Modern Tools
Scope of Module: Covers Spring Boot and testing frameworks.
Topics: Introduction to Spring Boot and its architecture; Dependency Injection (DI) in Spring; Creating RESTful APIs with Spring Boot; Spring annotations and application.properties; CRUD operations with Spring Data JPA; Connecting to

MySQL/PostgreSQL; Validation with Spring MVC; Unit testing with JUnit 5; Mocking with Mockito; Postman for API testing; Building with Maven and Gradle; Logging with Logback and SLF4J; Creating JAR/WAR packages; Dockerizing a Spring Boot app; Capstone project: E-commerce backend.
Learning Outcome: *Students will be able to develop full-stack Java applications using Spring Boot and related modern tools, and deploy them in enterprise-ready environments.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10

Prerequisites (If Any):

- Basic programming knowledge (C/C++/Python).

Pedagogy / Teaching Methodology:

- ✓ Interactive lectures with live coding
- ✓ Hands-on labs and debugging sessions
- ✓ Agile-style group projects
- ✓ Code reviews and peer assessments

Certificate / Value Added Courses Recommended:

- Java Programming by NPTEL: https://nptel.ac.in/courses/106105168
- Java Fundamentals by Oracle Academy: https://academy.oracle.com

Suggested Readings:

- Herbert Schildt, *Java: The Complete Reference*, McGraw-Hill, 2021.
- Joshua Bloch, *Effective Java*, Addison-Wesley, 2017.
- Craig Walls, *Spring Boot in Action*, Manning, 2020.

Recommended Software/Tools:

1. JDK 17+, IntelliJ IDEA/Eclipse
2. Spring Tool Suite, Postman
3. Git, Maven

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO PO Mapping Table

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**
CO1	3	2	1	–	2	1	–	1	2
CO2	2	3	2	1	-	–	–	2	-
CO3	3	2	2	-	-	1	2	-	2
CO4	2	3	3	1	2	2	–	1	2
CO5	3	2	-	2	2	1	1	2	2
Articulation	2.6	2.4	2	1.33	2	1.25	1.5	1.5	2

Course Title:	**Web Technologies**
Course Code:	**25VMT0C203**
Semester:	**II**
Credits:	**4**
Course Type	Core Course
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides comprehensive knowledge of modern web development technologies, covering front-end design, back-end programming, databases, and web security. Students will learn to build responsive, interactive web applications using HTML5, CSS3, JavaScript, PHP/Node.js, and frameworks like React and Express. Practical implementation through projects and case studies is emphasized.

Course Outcomes (COs)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Understand client-side web technologies (HTML5, CSS3, JavaScript).	L1
CO2	Develop interactive front-end applications using frameworks.	L3
CO3	Implement server-side programming and database integration.	L4
CO4	Evaluate web security threats and countermeasures.	L5

CO5	Build full-stack web applications using modern tools.	L3

Course Modules

No. of Hours:24

Module 1: Client-Side Web Technologies
Scope of Module: Covers fundamental technologies for building web interfaces.
Topics: HTML5 document structure and semantic tags; Forms and input validation; CSS3 selectors and specificity; Box model, flexbox, and grid layouts; Responsive design principles (media queries); Typography and UI consistency; JavaScript syntax and data types; Control structures and functions; ES6+ features: let/const, arrow functions, template literals; DOM tree and traversal; Event handling and delegation; Form validation using JavaScript; Browser console and developer tools; Animation with CSS and JavaScript; Project: Portfolio website with interactivity.
Learning Outcome: *Students will be able to create accessible, responsive, and well-structured web pages using HTML5, CSS3, and modern JavaScript.*

No. of Hours:24

Module 2: Front-End Frameworks
Scope of Module: Focuses on modern JavaScript frameworks for UI development
Topics: Introduction to React.js and component lifecycle; JSX syntax and virtual DOM;

Functional components and props; State and event management; Conditional rendering and lists; Forms and controlled components; React Hooks (useState, useEffect); Component communication; React Router for navigation; Context API for global state; AJAX requests with Axios and Fetch API; Using UI libraries (Bootstrap, Material UI); Creating reusable components; Code splitting and lazy loading; Project: Product listing page for e-commerce.
Learning Outcome: *Students will be able to build scalable and maintainable front-end applications using modern JavaScript frameworks like React.*

No. of Hours:24

Module 3: Server-Side Programming
Scope of Module: Explores back-end development and database integration.
Topics: Node.js basics and NPM; Creating a local web server; Express.js routing and middleware; Handling POST and GET requests; Template engines (EJS/Handlebars); Form data parsing and validation; MySQL and MongoDB CRUD operations; Connection pooling and error handling; RESTful API design principles; JWT and session-based authentication; Role-based access control; File uploads and file system module; Environment variables and configuration; Project: Blog system with authentication and database; Debugging Node.js applications.
Learning Outcome: *Students will be able to build robust server-side applications and RESTful APIs integrated with databases using Node.js and Express.*

No. of Hours:24

Module 4: Web Security
Scope of Module: Addresses security vulnerabilities and protection mechanisms.
Topics: Understanding HTTP protocol and HTTPS; SSL certificates and secure headers; OWASP Top 10 vulnerabilities; SQL injection and input sanitization; Cross-Site Scripting (XSS) and mitigation; Cross-Site Request Forgery (CSRF); Password hashing with bcrypt; Authentication vs. Authorization; Content Security Policy (CSP); Secure cookie practices; Same-Origin Policy and CORS; Rate limiting and API throttling; Penetration testing tools (Burp Suite, OWASP ZAP); Security audit and logging; Project: Secure login and audit trail system
Learning Outcome: *Students will be able to identify, prevent, and mitigate common web application security threats using industry best practices.*

No. of Hours:24

Module 5: Full-Stack Development
Scope of Module: Integrates all layers of web application development.
Topics: Overview of MERN/MEAN stacks; Connecting frontend with backend via APIs; Redux basics and state management; User authentication and protected routes; Role-based content rendering; API testing with Postman; Code versioning with Git and GitHub; Continuous deployment workflows; Hosting frontend (Netlify, Vercel); Hosting backend (Render, Heroku) Performance

optimization tips; Unit testing with Jest and Mocha; Linting and code formatting (Prettier, ESLint); DevOps basics for deployment; Capstone Project: Social media dashboard with analytics.
Learning Outcome: *Students will be able to develop, test, and deploy full-stack web applications using modern development tools and deployment platforms.*

Module	**No. of Pre-Recordings**
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10

Prerequisites (If Any):

- Basic programming knowledge (HTML/CSS recommended).

Pedagogy / Teaching Methodology:

- Interactive coding demonstrations
- Pair programming exercises
- Project-based learning
- Security hackathons

Certificate / Value Added Courses Recommended:

1. Web Development by MDN: https://developer.mozilla.org

2. Full-Stack Open: https://fullstackopen.com

Suggested Readings:

- Jon Duckett, *HTML & CSS: Design and Build Websites*, Wiley, 2011
- Eric Freeman, *Head First JavaScript Programming*, O'Reilly, 2014
- David Flanagan, *JavaScript: The Definitive Guide*, O'Reilly, 2020

Recommended Software/Tools:

1. VS Code, Chrome DevTools
2. Postman, MongoDB Compass
3. React DevTools, Git

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Consderns)
6.	Indian Knowledge System

CO PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	–	2	1	–	1	-
CO2	2	3	2	1	1	–	–	2	1
CO3	3	2	2	1	1	1	1	1	2
CO4	2	3	3	1	2	2	–	1	2
CO5	3	2	1	2	2	1	1	2	2
Articulation	2.6	2.4	1.8	1.25	1.6	1.25	1	1.4	1.75

Course Title:	**Cloud Computing Foundations**
Course Code:	**25VMT0C204**
Semester:	**II**
Credits:	**4**
Course Type	Core Course
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course introduces fundamental concepts of cloud computing, including service models, deployment models, virtualization, and cloud architecture. Students will gain hands-on experience with major cloud platforms (AWS/Azure/GCP) and learn to deploy, manage, and secure cloud-based solutions. The course emphasizes practical applications through labs and real-world case studies.

Course Outcomes (COs)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Understand cloud computing concepts and service models.	L1
CO2	Deploy and manage virtualized infrastructure.	L3
CO3	Analyze cloud storage and networking solutions.	L4
CO4	Evaluate cloud security and compliance frameworks.	L5

CO5	Develop cloud-native applications using PaaS.	L3

Course Modules

No. of Hours:24

Module 1: Cloud Computing Fundamentals
Scope of Module: Covers basic concepts, characteristics, and service models of cloud computing.
Topics: Definition and evolution of cloud computing; Essential characteristics of cloud (NIST model); Service models: IaaS, PaaS, SaaS; Deployment models: Public, Private, Hybrid, Community; Benefits and challenges of cloud adoption; Cloud computing use cases; Cloud vs. traditional IT infrastructure; Cloud service providers overview (AWS, Azure, GCP); Cloud service level agreements (SLAs); Multitenancy and resource pooling; Elasticity and scalability; Cloud computing cost models and pricing; Case study: Cloud migration strategy; Green computing and cloud efficiency; Cloud-first strategies in digital transformation**.**
Learning Outcome: *Students will be able to differentiate between cloud service and deployment models, and evaluate when and how to migrate to cloud environments.*

No. of Hours:24

Module 2: Virtualization and Cloud Infrastructure
Scope of Module: Focuses on virtualization technologies and cloud infrastructure management.

Topics: Virtualization concepts and benefits; Types of virtualization: Server, Storage, Network; Hypervisors: Type 1 (bare-metal) and Type 2 (hosted); Virtual machine creation and lifecycle; Introduction to containers and Docker; Container orchestration basics (Kubernetes overview); Infrastructure as a Service (IaaS); AWS EC2, Azure VMs, GCP Compute Engine; Cloud autoscaling and elasticity; Elastic Load Balancing (ELB); Monitoring and cloud resource usage; Infrastructure as Code: Terraform and CloudFormation; Hands-on: Launching a VM on AWS/GCP; Cloud cost optimization techniques; Best practices for resource provisioning**.**
Learning Outcome: *Students will be able to configure and manage virtualized infrastructure on cloud platforms using IaaS and container technologies.*

No. of Hours:24

Module 3: Cloud Storage and Networking
Scope of Module: Explores storage solutions and network architectures in cloud environments.
Topics: Cloud storage types: Object, Block, File; Storage services: Amazon S3, Azure Blob, Google Cloud Storage; Data redundancy and availability zones; Content Delivery Networks (CDNs) Edge computing and latency reduction; Data lifecycle and storage classes; Virtual Private Cloud (VPC) architecture; Subnets, gateways, and NAT; Cloud firewalls and security groups Cloud DNS and networking services; VPNs and hybrid cloud connectivity ;Database as a Service (DBaaS): RDS, Cosmos DB, Cloud SQL; Data backup and disaster recovery; Case study: Designing a high availability architecture; Storage and network performance benchmarking.

Learning Outcome: *Students will be able to design and configure secure, high-performance storage and networking solutions within cloud environments.*

No. of Hours:24

Module 4: Cloud Security and Compliance
Scope of Module: Addresses security challenges and compliance requirements in cloud computing.
Topics: Cloud security principles and models; Shared responsibility model; Identity and Access Management (IAM); Role-based access control (RBAC); Data encryption in transit and at rest; Key management services; Security groups vs. network ACLs; Firewall policies and VPN security; Security incident detection and response; Cloud compliance frameworks (GDPR, HIPAA, ISO 27001); Security posture assessments; Penetration testing and vulnerability scanning; Logging and auditing best practices; Case study: Securing a multi-tenant cloud app. Mini-project: Implementing IAM and encryption in AWS.
Learning Outcome: *Students will be able to implement cloud security measures and ensure compliance with data protection regulations and security frameworks.*

No. of Hours:24

Module 5: Cloud-Native Application Development
Scope of Module:

Covers development of applications optimized for cloud environments
Topics: Introduction to cloud-native principles; Twelve-factor app methodology; Serverless architecture overview; AWS Lambda and Azure Functions; Event-driven programming in the cloud; Microservices architecture and benefits; REST API design and testing (Postman); Containers vs. serverless vs. VMs; CI/CD pipelines: GitHub Actions, AWS CodePipeline; Monitoring and logging (CloudWatch, Stackdriver); Application scaling and performance tuning; API Gateway integration and throttling; DevOps tools for cloud (Jenkins, Terraform) Case study: Microservices deployment; Capstone Project: Build and deploy a serverless application.
Learning Outcome: *Students will be able to develop, deploy, and monitor scalable, cloud-native applications using modern cloud services and DevOps tools.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10

Prerequisites (If Any):

- Basic knowledge of operating systems and networking concepts.

Pedagogy / Teaching Methodology:

- ✓ Hands-on labs with cloud platforms
- ✓ Case studies of enterprise cloud migrations
- ✓ Security audit simulations
- ✓ Group projects with real-world scenarios

Certificate / Value Added Courses Recommended:

1. AWS Cloud Practitioner: https://aws.amazon.com/certification/
2. Microsoft Azure Fundamentals: https://learn.microsoft.com/en-us/certifications/

Suggested Readings:

- ✓ Dan Marinescu, *Cloud Computing: Theory and Practice*, 3rd Ed., Morgan Kaufmann, 2022
- ✓ Thomas Erl, *Cloud Computing: Concepts, Technology & Architecture*, Prentice Hall, 2021
- ✓ Official documentation from AWS, Azure, and GCP

Recommended Software/Tools:

1. AWS Free Tier, Azure Portal
2. Docker, Kubernetes (basic)
3. Terraform, CloudFormation

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development

3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	–	2	1	–	1	1
CO2	2	3	2	1	-	–	–	2	1
CO3	3	2	2	1	-	1	2	1	2
CO4	2	3	3	1	2	2	–	1	2
CO5	3	2	1	2	2	-	2	2	2
Articulation	2.6	2.4	1.8	1.25	2	1.33	2	1.5	1.6

Course Title:	Environmental, Social, and Governance (ESG) in IT
Course Code:	**25VMT0C205**
Semester:	**II**
Credits:	**4**
Course Type	AECC
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course explores the integration of Environmental, Social, and Governance (ESG) principles into information technology. It covers sustainable IT practices, ethical AI, data privacy, and corporate governance in tech organizations. Students will learn to assess ESG risks, implement green computing solutions, and align IT strategies with global sustainability goals through case studies and projects.

Course Outcomes (Cos)

Upon successful completion of this course, learners will be equipped to:

CO	**Course Outcome**	**BTL**
CO1	Explain ESG frameworks relevant to IT	L1
CO2	Apply green computing techniques	L3
CO3	Analyze social impacts of AI and big data	L4
CO4	Evaluate governance models for IT ethics	L5
CO5	Evaluate sustainable IT policies for organizations	L5

Course Modules

No. of Hours:24

Module 1: ESG Fundamentals in IT
Scope of Module: Introduces core ESG concepts and their application in technology sectors.
Topics: Understanding ESG in the context of IT; Global ESG frameworks: UN SDGs, GRI, SASB; Environmental impacts of digital infrastructure; Carbon footprint of data centers and cloud computing; E-waste management and recycling strategies; IT lifecycle sustainability assessments; Role of ICT in sustainable development goals; Green certifications for IT (Energy Star, EPEAT); Sustainability KPIs in IT projects; ESG compliance for tech startups AI and sustainability intersections; Digital ethics and sustainable design; Sustainable supply chain management in IT; Case study: Google's carbon-neutral cloud initiatives; Future of ESG in emerging technologies
Learning Outcome: *Students will articulate how ESG principles apply to IT infrastructure, service models, and sustainability reporting.*

No. of Hours:24

Module 2: Green Computing
Scope of Module: Focuses on sustainable IT practices and resource optimization.
Topics: Green hardware lifecycle management; Energy-efficient processor design; Power-aware scheduling algorithms; Green software design and refactoring;

Virtualization for sustainability; Resource optimization in cloud environments; Carbon-aware computing; Renewable energy integration in data centers; Measuring IT carbon footprints; Green networking and IoT devices; Cooling technologies and data center PUE; Server consolidation strategies; Green code practices in DevOps; Life Cycle Assessment (LCA) tools for IT systems; Lab: Calculate emissions from virtual machines.
Learning Outcome: *Students will be able to apply green computing principles to minimize the environmental impacts of IT operations and software systems.*

No. of Hours:24

Module 3: Social Responsibility in Tech
Scope of Module: Examines ethical implications of AI, data, and digital inclusion.
Topics: Digital equity and inclusion challenges; Algorithmic bias and mitigation techniques; AI transparency and explainability; Ethical data collection and consent; GDPR, CCPA, and global data privacy laws; Bias in training data for machine learning; Accessible design principles in software; Digital literacy and education programs; Tech for underserved communities. Social media and societal impact; AI in healthcare and social justice; Tech activism and digital rights; Workplace inclusivity in tech companies; Case study: Microsoft's AI for Accessibility Lab: Bias detection in data classification models.
Learning Outcome: *Students will analyze social implications of emerging technologies and advocate for inclusive, ethical digital innovation.*

No. of Hours:24

Module 4: IT Governance and Ethics
Scope of Module: Covers governance structures for ethical IT decision-making.
Topics: Foundations of IT corporate governance; ESG reporting frameworks and tools; Whistleblower policies and ethics channels; IT ethics boards and oversight; Compliance versus ethical leadership; IT audit trails and data governance; Transparency and accountability practices; Ethical AI governance and regulation; Cybersecurity ethics in surveillance systems; Boardroom simulation: resolving tech dilemmas; Balancing profit and public interest in IT firms; Decision-making under ethical uncertainty; ISO standards for IT governance; ESG ratings and public disclosures; Case study: Governance failure in a tech giant.
Learning Outcome: *Students will evaluate and develop governance frameworks that support ethical decision-making and stakeholder alignment in IT environments.*

No. of Hours:24

Module 5: Sustainable IT Strategy
Scope of Module: Integrates ESG principles into organizational IT planning.
Topics: Integrating ESG into IT strategy roadmaps; Stakeholder mapping and engagement models; Circular economy in IT hardware; Supply chain resilience

and transparency; Cloud sustainability strategy; Responsible tech innovation lifecycle; Designing ESG-aligned KPIs for IT; Tech sustainability scorecards and benchmarking; Policy modeling for net-zero IT operations; Carbon offsetting through tech initiatives; Funding and investing in green IT Organizational change for ESG transformation; Capstone: ESG audit of a mock IT enterprise Scenario planning for climate-resilient IT; Report writing: ESG strategy white paper.
Learning Outcome: *Students will design sustainable IT policies and long-term strategic plans incorporating ESG values for enterprise impact.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Basic knowledge of IT systems and corporate social responsibility.

Pedagogy:

- ✓ Guest lectures from ESG tech leaders
- ✓ Carbon accounting workshops
- ✓ Ethics hackathons
- ✓ ESG report analysis

Certifications Recommended:

1. Certified Green IT Professional (GITP)
2. SASB's ESG Fundamentals

Readings:

- "Green IT for Sustainable Business Practice" by Mark O'Neill
- "AI Ethics" by Mark Coeckelbergh

Tools:

- AWS Customer Carbon Footprint Tool
- ESG reporting software (Sustainalytics)

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO PO Mapping Table

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	2	1	-	3	1	-	2	1	2
CO2	3	2	1	2	3	1	-	2	1
CO3	1	3	2	-	2	3	1	1	3
CO4	2	1	3	2	1	2	3	-	2
CO5	3	2	2	3	2	1	2	2	3
Articulation	2.6	1.8	2	2.5	1.8	1.75	2	1.5	2.2

CENTRE FOR DISTANCE AND ONLINE EDUCATION

ONLINE

Semester III

Course Title:	Artificial Intelligence and Machine Learning

Course Code:	**251VMT0C301**
Semester:	**III**
Credits:	**4**
Course Type	Core Course
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides MCA students with an industry-relevant and application-driven exploration of Artificial Intelligence (AI) and Machine Learning (ML). The syllabus is designed to offer a progressive and logical learning path from AI fundamentals to advanced ML techniques and deployment strategies. The course also emphasizes ethical, explainable, and production-ready AI systems through a mix of practical labs, case studies, and modern tools.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO No.	Course Outcome	BTL
CO1	Understand the evolution and fundamentals of Artificial Intelligence and Machine Learning.	L1
CO2	Apply advanced ML algorithms for supervised, unsupervised, and semi-supervised learning tasks.	L3
CO3	Analyze deep learning models and reinforcement learning strategies.	L4
CO4	Evaluate models for bias, interpretability, and effectiveness across varied datasets.	L5

CO5	Design and deploy scalable, explainable, and secure AI/ML applications.	L3

Course Modules

No. of Hours:24

Module 1: Foundations of AI and ML
Scope of Module: Establishes core concepts of AI and ML, setting the stage for in-depth exploration of models, learning paradigms, and industry practices.
Topics: AI and ML definitions and key differences, history and evolution of AI, AI vs ML vs DL, symbolic vs sub-symbolic AI, types of ML (supervised, unsupervised, reinforcement), ML lifecycle and pipeline stages, CRISP-DM methodology, introduction to datasets and data types, common data challenges, bias and fairness in ML, ML vs traditional programming, use cases in various industries, recent trends in AI (Generative AI, LLMs), AI ethics overview.
Learning Outcome: *Students will be able to articulate core AI/ML definitions, classify learning types, and describe modern trends and challenges.*

No. of Hours:24

Module 2: Advanced Supervised Learning and Ensemble Models
Scope of Module: Covers core and advanced supervised learning techniques including ensemble strategies and hyperparameter tuning.
Topics:

Linear and logistic regression with regularization (Ridge, Lasso), decision trees, random forest, support vector machines (SVM), KNN classification, Naive Bayes, model evaluation metrics (accuracy, precision, recall, F1, ROC-AUC), model selection and validation strategies (train/test split, cross-validation), bias-variance tradeoff, grid search and random search, ensemble learning (bagging, boosting), introduction to XGBoost, LightGBM, and CatBoost, multi-output and multiclass classification, managing imbalanced datasets (SMOTE, undersampling), SHAP for tree models.
Learning Outcome: *Students will be able to apply and evaluate supervised learning and ensemble methods to solve classification and regression tasks.*

No. of Hours:24

Module 3: Unsupervised Learning, Dimensionality Reduction, and Anomaly Detection
Scope of Module: Explores a wide range of unsupervised techniques and introduces preprocessing and anomaly detection as critical skills.
Topics: Clustering algorithms: k-means, DBSCAN, hierarchical, Gaussian Mixture Models (GMM), elbow method and silhouette score, association rule mining (Apriori, FP-Growth), dimensionality reduction techniques: PCA, t-SNE, UMAP, feature selection vs extraction, feature importance, anomaly detection using Isolation Forest, LOF, one-class SVM, preprocessing: encoding, scaling, normalization, dealing with missing values, pipeline creation using scikit-learn, unsupervised model evaluation strategies.

Learning Outcome: *-Students will be able to perform unsupervised learning, reduce dimensionality for visualization, and detect data anomalies effectively.*

No. of Hours:24

Module 4: Deep Learning and Reinforcement Learning
Scope of Module: Introduces deep neural networks and reinforcement learning paradigms for intelligent decision making.
Topics: Artificial Neural Networks (ANN), forward and backward propagation, activation functions (ReLU, sigmoid, tanh), loss functions and optimizers, CNNs for image processing, RNNs, LSTMs for sequential data, autoencoders and applications, GANs and generative models, overview of BERT and Transformer models, introduction to reinforcement learning, Q-learning and policy gradients, environment-reward-agent loop, OpenAI Gym, applications in recommendation, robotics, gaming.
Learning Outcome: *-Students will be able to build and analyze neural networks and apply RL for decision-making scenarios.*

No. of Hours:24

Module 5: AI/ML Model Deployment, MLOps, and Explainability
Scope of Module:

Focuses on end-to-end AI/ML model deployment, production readiness, monitoring, and responsible AI practices.
Topics: Deployment using Flask, FastAPI, Streamlit, Docker containerization, introduction to Kubernetes for ML, CI/CD pipelines in ML, MLflow and model versioning, explainable AI tools (SHAP, LIME, ELI5), fairness audits and bias detection tools, logging and monitoring models in production, introduction to cloud ML platforms (AWS SageMaker, GCP Vertex AI), securing ML models and data, case studies from industry, best practices in production ML systems. ***Learning Outcome: -****Students will be able to deploy, explain, and monitor AI/ML systems using modern DevOps practices and frameworks*

Module	No. of Pre-Recordings
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Strong understanding of Python programming, statistics, and data structures.

Pedagogy / Teaching Methodology:

- ✓ Project-based learning with datasets from real-world domains
- ✓ Lab-based exercises using Jupyter and Google Colab
- ✓ Mini case studies and research paper reviews
- ✓ Live demonstrations and group activities
- ✓ Tool walkthroughs and guided deployment

Certificate / Value Added Courses Recommended:

1. Machine Learning Engineering for Production (MLOps) – Coursera
2. Deep Learning Specialization – Coursera
3. AI for Everyone – Coursera
4. Google ML Crash Course – https://developers.google.com/machine-learning/crash-course

Suggested Readings:

- Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, O'Reilly, 2022
- Ian Goodfellow, Deep Learning, MIT Press, 2016
- Sutton & Barto, Reinforcement Learning: An Introduction, MIT Press, 2018
- Kelleher, D. J., Data Science Ethics, MIT Press
- Relevant research from IEEE, ACM, and arXiv.org

Recommended Software/Tools:

1. Python, Scikit-learn, TensorFlow, Keras, PyTorch
2. Google Colab / JupyterLab, MLflow, Optuna
3. Docker, FastAPI, Streamlit, AWS SageMaker, GCP Vertex AI

4. SHAP, LIME, OpenAI Gym

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO-PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	1	2	-	1	1	2
CO2	3	2	3	2	2	1	-	2	2
CO3	2	3	2	3	2	1	2	-	2
CO4	2	2	2	-	2	3	2	2	2
CO5	3	3	3	2	3	2	2	3	3
Articulation	2.6	2.4	2.2	2	2.2	1.75	1.75	2	2.2

Course Title:	**Python Programming**
Course Code:	**251VMT0C302**
Semester:	**III**
Credits:	**4**
Course Type	Core Course
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides MCA students with foundational to advanced skills in Python programming, blending basic concepts with real-world applications. It encompasses control structures, functions, OOP, file operations, exception handling, unit testing, web development, API integration, and data analysis. Special emphasis is placed on writing efficient, modular, and scalable Python code with exposure to industry-grade tools and frameworks.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO No.	Course Outcome	BTL
CO1	Understand Python syntax, control flow, data structures, and functions.	L1
CO2	Apply OOP principles and modular coding in real-world applications.	L3
CO3	Analyze problems and implement file, error handling, and unit testing.	L4

CO4	Evaluate web-based interactions, APIs, and database connectivity using Python.	L5
CO5	Develop interactive data-driven applications using Pandas, NumPy, and visualization libraries.	L3

Course Modules

No. of Hours:24

Module 1: Python Basics, Syntax, and Control Flow
Scope of Module: Introduces core elements of Python such as syntax, variables, data types, operators, and control flow.
Topics: Installing Python and IDE setup, writing and executing Python scripts, identifiers, keywords, variables, constants, data types, type conversion, operators and precedence, conditional statements (if, elif, else), iterative control (for, while), break and continue, pass statement, nested loops, input/output handling, formatted string literals (f-strings), Python coding style (PEP8).
Learning Outcome: *Students will be able to write structured Python scripts using control flow constructs and input/output operations.*

No. of Hours:24

Module 2: Functions, Collections, Iterators, and Modules
Scope of Module:

Covers function design, advanced collection operations, and module creation for reusable code.
Topics: Defining and invoking functions, argument types (*args, **kwargs), scope and lifetime, recursion, anonymous functions (lambda), Python built-in functions, comprehension (list, dict, set), advanced list methods, tuples and unpacking, sets and operations, dictionaries and operations, iterators and generators, creating custom modules, importing built-in modules (math, sys, os, datetime, calendar, random), packaging with pip and venv.
Learning Outcome: *Students will be able to modularize programs using custom and built-in Python functions and libraries.*

No. of Hours:24

Module 3: Object-Oriented Programming, File I/O, and Exception Handling
Scope of Module: Explores OOP principles, file operations, and robust exception handling techniques.
Topics: Classes and objects, class and instance variables, methods, constructor and destructor, inheritance and types, method overriding, polymorphism, encapsulation, use of __str__, __repr__, __len__, working with text/binary/CSV files, reading and writing files, context manager and file closing, exception types and hierarchy, try-except-finally, raising and handling custom exceptions, logging module basics, assertions for debugging.

Learning Outcome: *Students will be able to design OOP-based applications with efficient file handling and error management*

No. of Hours:24

Module 4: Testing, Automation, and Database Connectivity
Scope of Module: Provides practices in automated testing, scripting for task automation, and integrating Python with databases.
Topics: Writing test cases using unittest and pytest, test discovery and test suites, test-driven development, automation with shutil and subprocess, working with file system using os and pathlib, scheduling tasks, introduction to SQLite and PostgreSQL, executing queries from Python using sqlite3 and psycopg2, performing CRUD operations, using context managers with databases, parameterized queries to prevent SQL injection.
Learning Outcome: *Students will be able to automate tasks and integrate databases securely in Python programs.*

No. of Hours:24

Module 5: Web Development, APIs, and Data Science Libraries
Scope of Module: Bridges Python to web and data science fields using APIs, Flask, and data analysis tools

Topics:
Flask framework structure and routing, GET and POST methods, HTML rendering with Jinja2 templates, form data processing, REST API consumption using requests, OpenWeatherMap API integration example, Pandas: dataframes, merging, filtering, groupby, NumPy: array creation and operations, Matplotlib/Seaborn for plotting, exporting data to Excel/CSV, introduction to Streamlit for interactive dashboards, mini project combining Flask and Pandas.
Learning Outcome: *Students will be able to develop and deploy data-centric and interactive Python web applications*

Module	**No. of Pre-Recordings**
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10.

Prerequisites:

- Basic logic building, familiarity with any programming language is helpful.

Pedagogy / Teaching Methodology:

- ✓ Concept-based demonstrations using Jupyter and IDEs
- ✓ Mini assignments and use-case-driven labs

- ✓ Group activities and debugging challenges
- ✓ Integration of GitHub for version control practices
- ✓ End-of-module review quizzes and projects

Certificate / Value Added Courses Recommended:

1. Python for Everybody – Coursera
2. Python OOP and Flask Projects – Udemy
3. Automate the Boring Stuff with Python – Online Book
4. Introduction to Data Science in Python – Coursera

Suggested Readings:

- Mark Lutz, Learning Python, O'Reilly
- Eric Matthes, Python Crash Course, No Starch Press
- Luciano Ramalho, Fluent Python, O'Reilly
- Python Documentation – https://docs.python.org

Recommended Software/Tools:

1. Python 3.x, PyCharm, VS Code
2. JupyterLab, Google Colab
3. Flask, Pandas, NumPy, Matplotlib, Streamlit
4. GitHub, SQLite, PostgreSQL

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global

2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above said points.

CO-PO Mapping Table

CO/PO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**
CO1	3	2	2	1	2	1	-	1	2
CO2	3	2	3	2	3	1	2	2	3
CO3	3	3	-	3	2	-	2	2	2
CO4	2	3	3	3	3	2	2	2	-
CO5	3	3	2	2	3	2	2	3	3
Articulation	2.8	2.6	2.5	2.2	2.6	1.7	2	2	2.5

Course Title:	**Transformer Models & Attention Mechanisms**
Course Code:	**251VMT5S303**
Semester:	**III**
Credits:	**4**
Course Type	DSE
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides a comprehensive study of transformer architectures and attention mechanisms that revolutionized natural language processing and computer vision. It covers theoretical foundations, implementation details, and advanced applications including large language models (LLMs). Hands-on labs with PyTorch/TensorFlow and real-world projects are emphasized.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO No.	Course Outcome	BTL
CO1	Explain attention mechanisms and transformer components	L2
CO2	Implement transformer models using deep learning frameworks	L3
CO3	Analyze different attention variants (self/cross/multi-head)	L4
CO4	Evaluate transformer applications in NLP/CV tasks	L5

CO5	Design modified transformer architectures for domain-specific problems	L6

Course Modules

No. of Hours:24

Module 1: Foundations of Attention Mechanisms
Scope of Module: Introduces core concepts of attention and its evolution in deep learning
Topics: Sequence modeling challenges in NLP; Vanishing gradients in RNNs and LSTMs; Bottlenecks of sequential processing; Introduction to the attention paradigm; Key-query-value concept Dot-product attention; Scaled dot-product attention and stability; Softmax and attention weights; Additive attention vs. multiplicative attention; Self-attention: Concept and intuition; Cross-attention overview; Positional encodings (sinusoidal and learned); Visualizing attention in neural networks; Implementation: Basic attention layer in PyTorch/TensorFlow; Case study: Seq2Seq with attention for translation.
Learning Outcome: *Students will be able to implement and visualize fundamental attention mechanisms and explain their impact on sequence modeling.*

No. of Hours:24

Module 2: Transformer Architecture

Scope of Module:

Detailed study of original transformer architecture..

Topics:
Motivation for transformer models; Overview of encoder-decoder structure; Multi-head attention mechanism; Scaled dot-product within multi-head attention; Position-wise feed-forward networks; Residual connections and layer normalization; Masking strategies: Padding mask and look-ahead mask; Tokenization and input embeddings; Encoder architecture in detail; Decoder architecture in detail; Transformer parameterization. Positional encoding in transformers; Hands-on: Building transformer blocks from scratch. Comparison with RNN/LSTM-based architectures; Understanding model depth and layer stacking.

Learning Outcome: *Students will construct and analyze functional transformer modules and understand their architectural components in detail.*

No. of Hours:24

Module 3: Advanced Attention Variants

Scope of Module:
Explores modern attention improvements.

Topics:
Sparse attention for long sequences; Linear attention and kernelized dot-product; Performer, Linformer, and Longformer techniques; Memory-compressed attention Global-local attention mechanisms; Cross-attention in multi-modal inputs; Relative positional embeddings; Efficient transformers: Reformer and BigBird; Attention mechanisms in audio and time-series; Dynamic attention routing; Deformable attention in vision tasks Benchmarking attention models on NLP tasks; Implementation: Custom attention types Profiling computational complexity of attention; Lab: Text classification using various attention variants.

Learning Outcome: *Students will design and benchmark advanced attention mechanisms optimized for scalability and task-specific performance.*

No. of Hours:24

Module 4: Training & Optimization Techniques

Scope of Module:

Covers transformer-specific training techniques.

Topics:
Review of transformer training challenges; Optimizer: Adam vs. AdamW; Learning rate warm-up and scheduling; Weight decay and gradient clipping; Gradient checkpointing for memory efficiency; Mixed-precision training with FP16; Label smoothing and dropout Batch size strategies for large models; Pretraining vs. fine-tuning approaches; Masked language modeling and next sentence prediction; Transfer learning with transformers Model parallelism and distributed training; Debugging and error analysis in large models; Hands-on: Fine-tuning BERT on custom datasets; Evaluation metrics for transformer tasks.

Learning Outcome: *Students will apply transformer-specific training strategies to optimize model performance across different hardware settings and data scales.*

No. of Hours:24

Module 5: Applications & Frontiers

Scope of Module:

Examines cutting-edge applications and research.

Topics:
Vision transformers (ViT) and hybrid models; Transformers in audio, speech, and music modeling; Large language models: BERT, GPT, T5, and their variants; Multimodal transformers (text + image/video/audio); Prompt tuning and instruction-based learning; Model compression: Quantization and pruning; Deployment strategies for transformer models; Latency optimization for real-time inference; Federated learning with transformers. Transformer safety and ethical considerations; Bias detection and fairness in language models; Interpretable transformer outputs; Case study: Transformers in healthcare or finance; Emerging trends: MoE (Mixture of Experts), RETRO; Capstone: Domain-specific transformer implementation.

Learning Outcome: *Students will adapt and extend transformer models for real-world applications across domains, while addressing ethical, computational, and deployment concerns.*

Module	**No. of Pre-Recordings**
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10.

Prerequisites (If Any):

- Python programming, Deep Learning fundamentals (CNNs/RNNs), Linear Algebra

Pedagogy / Teaching Methodology:

- ✓ Theory lectures with mathematical derivations
- ✓ Weekly coding assignments
- ✓ Model interpretability exercises
- ✓ Research paper discussions

Certificate / Value Added Courses Recommended:

1. Hugging Face Transformer Certification
2. NVIDIA DLI Accelerated AI with Transformers

Suggested Readings:

- "Attention Is All You Need", by Vaswani et al.
- "The Illustrated Transformer", by Jay Alammar
- "Transformers for Natural Language Processing", by Denis Rothman

Recommended Software/Tools:

1. PyTorch Lightning, Hugging Face Transformers
2. Weights & Biases for experiment tracking
3. ONNX Runtime for deployment

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO-PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	-	1	-	2	1	1
CO2	2	3	3	1	2	1	-	2	1
CO3	3	3	2	2	1	2	1	1	2
CO4	2	2	3	3	2	2	-	2	3
CO5	3	3	3	2	3	1	2	3	2
Articulation	2.6	2.6	2.4	2.0	1.8	1.5	1.8	1.8	1.8

Course Title:	**Open Elective**
Course Code:	
Semester:	**III**
Credits:	**4**
Course Type	Open Elective
Learning Hours	120
Live Sessions	**12 hours**

CENTRE FOR DISTANCE AND ONLINE EDUCATION

ONLINE

Semester IV

Corporate Office
91/2, Dr. A N Krishna Rao Road V V Puram,
Bengaluru - 560 004
P: +91 80 4343 1000

Centre for Distance & Online Education
319, 25th Main, 17th Cross,
J P Nagar 6th Phase
Bengaluru – 560 078
P: +91 80 4343 0100

Course Title:	**Large-Scale AI Model Deployment**
Course Code:	**251VMT5S401**
Semester:	**IV**
Credits:	**4**
Course Type	DSE
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides comprehensive training in deploying AI models at scale, covering containerization, distributed computing, model optimization, and MLOps practices. Students will learn to deploy models across cloud/edge environments while addressing latency, cost, and ethical constraints through hands-on projects with industry-standard tools.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO	**Course Outcome**	**BTL**
CO1	Explain deployment challenges for large AI models	L2
CO2	Containerize models using Docker/Kubernetes	L3
CO3	Optimize models for edge/cloud environments	L4
CO4	Evaluate deployment architectures for scalability	L5
CO5	Design CI/CD pipelines for AI systems	L6

Course Modules

No. of Hours:24

Module 1: Foundations of AI Deployment

Scope of Module:
Covers deployment lifecycle and infrastructure requirements.

Topics:
AI deployment lifecycle: overview and challenges; Deployment environments: cloud, edge, hybrid; Batch vs. real-time inference systems; Model packaging formats: ONNX, PMML, TorchScript; Model metadata and versioning; Latency vs. throughput: trade-offs and metrics. Hardware considerations: CPU, GPU, TPU; Infrastructure-as-Code basics for deployment; Environment reproducibility with Conda/VirtualEnv; Model validation before deployment. Serialization techniques: Pickle, Joblib, SavedModel; Security concerns in AI deployment; Inference pipelines and post-processing; Scaling AI across multiple endpoints; Case study: OpenAI/ChatGPT serving stack.

Learning Outcome: *Students will articulate infrastructure and deployment trade-offs for AI models across varied environments and use cases.*

No. of Hours:24

Module 2: Containerization & Orchestration

Scope of Module:
Focuses on container-based deployment strategies.

Topics:
Basics of containerization with Docker; Creating Dockerfiles for AI model deployment; Docker Compose for multi-container setups; Introduction to Kubernetes for ML; Deploying containers with Kubernetes YAML; Kubernetes Operators for ML workflows; Horizontal and vertical auto-scaling; Resource limits and request management in K8s; Model versioning with Kubernetes rollout strategies; GPU scheduling in Kubernetes; Istio and service mesh for model routing. Deploying model inference servers (Triton, TFServing); Traffic shifting with K8s + Istio

Secure access and secrets management; Lab: Deploying a transformer model on Kubernetes cluster.

Learning Outcome: *Students will containerize AI models and orchestrate scalable deployments using Kubernetes and modern DevOps practices*.

No. of Hours:24

Module 3: Model Optimization

Scope of Module:
Techniques for efficient model serving.

Topics:
Model quantization techniques: FP16, INT8; Pruning strategies: structured and unstructured; Knowledge distillation: student-teacher networks; Low-rank factorization methods; Graph-level optimization using ONNX Runtime; Compiler-based optimizations: TVM, XLA; Operator fusion for better inference latency; Batching strategies and throughput control Quantization-aware training; Post-training quantization pipelines; Hardware-specific optimization: TensorRT, OpenVINO; Model simplification with dynamic/static shapes; Latency benchmarking tools; Hands-on: Optimize a ResNet model for Jetson Nano; Project: Model compression and acceleration toolkit

Learning Outcome: *Students will apply various model optimization techniques to improve inference speed and reduce resource usage for deployment at scale.*

No. of Hours:24

Module 4: Scalable Serving Architectures

Scope of Module:
Designs for high-availability systems.

Topics:
Introduction to serving architectures; Monolith vs. microservices-based deployment; Model parallelism for large models; Model sharding and checkpoint distribution; Inference API design best practices; Caching strategies (Redis, CDN, in-memory); Load balancing across inference nodes; Request queuing and priority mechanisms; A/B testing and canary deployments; Auto-

scaling using cloud-native tools; Batch inference frameworks (NVIDIA Triton, SageMaker); Cold start mitigation strategies; HAProxy and NGINX for traffic routing Distributed tracing and observability; Case study: Netflix recommender model serving

Learning Outcome: Students will design high-availability, scalable, and fault-tolerant serving architectures for large-scale AI systems

No. of Hours:24

Module 5: MLOps & Monitoring

Scope of Module:
End-to-end pipeline management.

Topics:
CI/CD for ML using MLflow and GitHub Actions; Pipeline orchestration with Kubeflow and Airflow; Model registry and reproducibility; Model drift and data drift detection; Logging inference outputs and monitoring bias; Performance monitoring with Prometheus and Grafana. Realtime anomaly detection in AI services; Ethical logging and audit trails; Security scanning in model pipelines; Rollback strategies for model deployment; Monitoring resource usage across pipelines; Alerting systems for service degradation; Metadata management and tracking. Capstone: Full deployment pipeline for computer vision system; End-to-end monitoring dashboard implementation

Learning Outcome: *Students will implement a fully automated, monitored, and ethical ML deployment pipeline using modern MLOps frameworks.*

Module	No of Pre-Recordings
1	10
2	12

3	**10**
4	**12**
5	**10**

Note: Minimum number of pre-recordings is 10.

Prerequisites (If Any):

- Machine Learning, Cloud Computing, Python

Pedagogy / Teaching Methodology:

- ✓ Cloud lab environments (AWS/GCP)
- ✓ Performance benchmarking contests
- ✓ Incident response simulations
- ✓ Guest lectures from AI platform engineers

Certificate / Value Added Courses Recommended:

1. AWS Certified Machine Learning Specialty
2. Google Professional ML Engineer

Suggested Readings:

- Jurafsky & Martin, Speech and Language Processing, Pearson, 2020.
- Goodfellow, Bengio & Courville, Deep Learning, MIT Press, 2016.
- Chip Huyen, Designing Machine Learning Systems, O'Reilly Media, 2022.
- Andriy Burkov, Machine Learning Engineering, True Positive Inc.,

Recommended Software/Tools:

1. Docker, Kubernetes, Terraform
2. Prometheus/Grafana for monitoring
3. Seldon Core for model serving

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO-PO Mapping Table

CO/PO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**
CO1	3	2	2	1	1	1	1	1	2

CO2	3	3	3	2	2	1	1	2	3
CO3	3	3	3	3	2	1	2	2	3
CO4	2	3	3	3	3	2	2	2	3
CO5	3	3	3	2	3	2	2	3	3
Articulation	2.8	2.8	2.8	2.2	2.2	1.4	1.6	2.0	2.8

Course Title:	**Multimodal AI**
Course Code:	**251VMT5S402**
Semester:	**IV**
Credits:	**4**
Course Type	**DSE**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides a comprehensive understanding of Multimodal Artificial Intelligence, focusing on the integration and processing of multiple data modalities such as text, images, audio, and video. Students will learn foundational concepts, models, and frameworks for building multimodal systems, including transformer architectures, cross-modal learning, and real-world applications. Practical implementation through projects and case studies is emphasized.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO	**Course Outcome**	**BTL**
CO1	Understand the fundamentals of multimodal AI and its applications.	L1
CO2	Implement models for processing and fusing multiple data modalities.	L3
CO3	Analyze and evaluate multimodal architectures.	L4
CO4	Design and optimize multimodal systems for real-world problems.	L5
CO5	Build end-to-end multimodal AI applications using modern tools.	L3

Course Modules:

No. of Hours:24

Module 1: Introduction to Multimodal AI

Scope of Module:
Covers foundational concepts, challenges, and applications of multimodal AI.

Topics:
Definition and significance of multimodal AI; Examples of modalities: text, image, audio, video, sensor data; Historical context and evolution of multimodal learning; Multimodal representation learning: motivation and benefits; Challenges in multimodal AI: alignment, heterogeneity, missing data; Fusion types: early, late, and hybrid; Alignment techniques: temporal, semantic, and spatial; Cross-modal retrieval and translation tasks; Case study: Multimodal AI in healthcare (text + image); Case study: AI in entertainment (music + video + emotion); Tools and datasets: COCO, VGGSound, MM-IMDB; Preprocessing challenges across modalities; Multimodal data annotation and labeling issues; Applications across industries, Future trends in multimodal research

Learning Outcome: *Students will be able to explain the foundations of multimodal AI and discuss its challenges and key application domains.*

No. of Hours:24

Module 2: Multimodal Data Processing

Scope of Module:
Focuses on techniques for processing and representing multimodal data.

Topics:
Preprocessing pipelines for multimodal inputs; Text feature extraction: tokenization, word2vec, BERT; Image feature extraction: CNNs and pre-trained models (ResNet, EfficientNet); Audio feature extraction: MFCCs, spectrograms, wavelets; Temporal alignment of video and speech Multimodal data synchronization techniques; Normalization and standardization across modalities; Dimensionality alignment: padding and truncation strategies; Embedding alignment: joint vs. modality-specific embeddings; Handling missing or corrupted modality

data; Dataset construction for multimodal tasks; Metadata and context encoding in preprocessing; Multi-task learning input strategies; Data augmentation for multimodal inputs; Practical lab: Building a multimodal preprocessing pipeline

Learning Outcome: *Students will be able to preprocess, align, and represent multimodal data using industry-standard techniques and tools.*

No. of Hours:24

Module 3: Multimodal Architectures

Scope of Module:
Explores advanced models and frameworks for multimodal learning.

Topics:
Transformer-based architectures for multimodal AI; Introduction to ViLBERT, LXMERT, and CLIP
Cross-modal attention mechanisms; Vision-language pretraining (VLP) models; Multimodal fusion strategies: early, late, and hybrid fusion; Joint embedding spaces for cross-modal retrieval; Contrastive learning in multimodal contexts; Hierarchical fusion and attention mechanisms; Multimodal encoder-decoder structures; Graph neural networks for structured multimodal data; Memory networks for long-context fusion; Transfer learning across modalities; Fine-tuning pre-trained multimodal models; Evaluation of fusion techniques
Project: Implementing a multimodal sentiment analysis system.

Learning Outcome: *Students will be able to design and implement multimodal deep learning architectures and evaluate fusion and alignment strategies*

No. of Hours:24

Module 4: Applications and Optimization

Scope of Module:
Addresses real-world applications and performance optimization.

Topics:
Multimodal NLP tasks: VQA, image captioning, document understanding; Multimodal computer vision: scene understanding, emotion recognition; Speech and audio processing with visual grounding; Robotic control using vision-language commands; Real-time inference and latency trade-offs; Scalability and cloud-based deployment; Lightweight architectures for edge devices
Model distillation for multimodal settings; Bias and fairness challenges in multimodal data. Explainability and interpretability in multimodal models; Multimodal system generalization across domains; Robustness testing under noisy/missing modality scenarios; Ethical and legal concerns in real-world applications; Mini-project: Building a multimodal customer support chatbot; Case study: Multimodal emotion recognition in mental health tools

***Learning Outcome*:** *Students will be able to design, optimize, and ethically evaluate multimodal AI systems for various real-world applications.*

No. of Hours:24

Module 5: End-to-End Multimodal Systems

Scope of Module:
Integrates all aspects of multimodal AI into complete systems.

Topics:
System architecture for production-grade multimodal pipelines; End-to-end training and inference workflows; Multimodal model versioning and lifecycle management; Containerization (Docker) and deployment; Using APIs for multimodal data exchange; Cloud services for multimodal AI (AWS, GCP, Azure); Model serving frameworks: TensorFlow Serving, TorchServe
Performance monitoring and logging; Real-time vs. batch processing of multimodal pipelines.Evaluation metrics: accuracy, BLEU, METEOR, recall@k; A/B testing and user feedback loops. Security, privacy, and compliance; Scalability: handling large-scale datasets; Capstone: Build and deploy a multimodal recommendation system; Team presentations and peer evaluation.

Learning Outcome: *Students will be able to build, deploy, and evaluate robust end-to-end multimodal AI systems integrating multiple data streams and services.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum number of pre-recordings is 10.

Prerequisites (If Any):

- Basic knowledge of machine learning, Python programming, and familiarity with NLP or computer vision concepts

Pedagogy / Teaching Methodology:

- ✓ Interactive lectures and coding demonstrations.
- ✓ Hands-on labs with real-world datasets.
- ✓ Group projects and hackathons.
- ✓ Guest lectures from industry experts

Certificate / Value Added Courses Recommended:

1. Deep Learning Specialization by Andrew Ng (Coursera).
2. Multimodal Machine Learning Course (CMU).

Suggested Readings:

- Jurafsky & Martin, *Speech and Language Processing*, Pearson, 2020.
- Goodfellow et al., *Deep Learning*, MIT Press, 2016.
- Research papers on CLIP, ViLBERT, and other multimodal models

Recommended Software/Tools:

1. Python, PyTorch, TensorFlow.
2. Hugging Face Transformers, OpenCV, Librosa.
3. Jupyter Notebooks, Google Colab

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO-PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	--	1	--	--	1	1
CO2	2	3	2	1	1	--	--	2	1
CO3	3	2	2	1	1	1	1	1	2
CO4	2	3	3	1	2	2	--	1	2
CO5	3	2	1	2	2	1	1	2	2
Articulation	2.6	2.4	1.8	1.2	1.4	1.3	1.0	1.4	1.6

Course Title:	**Reinforcement Learning**
Course Code:	**251VMT5S403**
Semester:	**IV**
Credits:	**4**
Course Type	**DSE**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides a comprehensive introduction to Reinforcement Learning (RL), covering fundamental algorithms, theoretical foundations, and practical applications. Students will learn to model sequential decision-making problems, implement RL algorithms, and apply them to real-world scenarios such as robotics, gaming, and autonomous systems. Hands-on projects and case studies are emphasized to bridge theory and practice.

Course Outcomes (COs):

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Understand the fundamentals of RL, including Markov Decision Processes (MDPs) and key concepts like rewards, policies, and value functions.	L1
CO2	Implement basic RL algorithms and analyze their performance.	L3
CO3	Evaluate and compare advanced RL methods.	L4
CO4	Design RL solutions for real-world problems, considering challenges like exploration vs. exploitation and scalability.	L5
CO5	Build and deploy RL agents in simulated or physical environments.	L3

Course Modules

No. of Hours:24

Module 1: Foundations of Reinforcement Learning

Scope of Module:
Introduces core concepts and mathematical frameworks for RL.

Topics:
Introduction to agents and environments; Reward signals and return formulation; Policy definitions: deterministic and stochastic; Markov Decision Processes (MDPs); States, actions, transitions, and rewards; Value functions: V(s), Q(s, a); Bellman equations for value functions; Policy evaluation and improvement; Exploration vs. exploitation dilemma; Mathematical assumptions in RL; Finite horizon vs. infinite horizon settings; Discount factors and convergence; Taxonomy of RL algorithms; Practical: Visualizing grid-world MDPs;
Case study: Solving Tic-Tac-Toe using policy iteration

Learning Outcome: *Students will be able to formalize real-world problems using MDPs and understand the fundamental concepts of reinforcement learning*.

No. of Hours:24

Module 2: Tabular RL Methods

Scope of Module:
Covers classical RL algorithms for finite state-action spaces.

Topics:
Model-based dynamic programming: Policy Iteration; Model-based dynamic programming: Value Iteration; Monte Carlo prediction methods; Monte Carlo control: Exploring starts and ε-greedy policies; Temporal Difference (TD) learning: TD(0); TD control: SARSA (on-policy learning); Q-learning (off-policy learning); Eligibility traces and TD(λ); Comparing Monte Carlo, TD, and dynamic programming; Exploration strategies: ε-greedy, softmax, UCB; Convergence properties of tabular methods; Performance metrics for policy evaluation Hyperparameter

tuning for tabular RL; Implementation lab: Q-learning in grid-world Case study: Learning path navigation for a virtual agent

Learning Outcome: *Students will be able to implement, evaluate, and compare classical tabular reinforcement learning algorithms*.

No. of Hours:24

Module 3: Function Approximation and Deep RL

Scope of Module:
Function Approximation and Deep RL.

Topics:
Challenges with large/continuous state spaces; Linear function approximation techniques; Neural network approximators for value functions; Deep Q-Network (DQN) architecture and loss functions; Stabilization: Experience replay and target networks; Double DQN for overestimation bias; Dueling DQN architecture; Prioritized Experience Replay; Policy gradient theorem and REINFORCE algorithm; Actor-Critic methods: A2C, A3C; Proximal Policy Optimization (PPO) overview; Continuous control with Deep Deterministic Policy; Gradient (DDPG); Exploration strategies in deep RL; Experimentation with OpenAI Gym environments; Case study: CartPole and LunarLander using DQN and PPO.

Learning Outcome: *Students will be able to implement deep reinforcement learning algorithms and apply function approximation techniques effectively.*

No. of Hours:24

Module 4: Advanced RL Techniques

Scope of Module:
Explores cutting-edge RL methods and challenges

Topics:
Multi-agent reinforcement learning concepts; Competitive and cooperative multi-agent

setups; Hierarchical RL: options framework; Intrinsic motivation and curiosity-driven learning; Meta-learning in RL; Inverse reinforcement learning (IRL) principles; Behavior cloning and imitation learning; Offline RL and batch-constrained methods; Model-based RL with learned dynamics; Safe RL and constrained MDPs; Exploration via uncertainty estimation (e.g., Bootstrapped DQN); Population-based training; Transfer learning in reinforcement learning; Custom environment design using Gym Mini-project: Hierarchical agent in a multi-goal task.

Learning Outcome: *Students will be able to develop custom solutions using advanced RL paradigms and analyze complex learning behaviors.*

No. of Hours:24

Module 5: Applications and Deployment

Scope of Module:
Integrates RL into real-world systems and addresses practical constraints.

Topics:
Reinforcement learning in robotics (e.g., navigation, manipulation); RL for recommendation systems and ad placement; Game AI: policy learning for adversarial settings; RL in finance: portfolio optimization; Sim-to-real transfer learning for physical agents; Ethical considerations in autonomous decision-making; Safe exploration and risk-aware deployment; Scalability of RL in production environments; Deployment platforms: TensorFlow Serving, TorchServe; Integration with ROS for robotics applications; Real-time inference optimization; Logging and telemetry for deployed agents; Monitoring reward drift and agent behavior; Capstone: Build and deploy an RL-based control or game agent;
Group presentations and feedback

Learning Outcome*: Students will be able to design, implement, and deploy reinforcement learning agents for real-world environments with performance and safety considerations.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Minimum no. of pre-recordings:10

Prerequisites (If Any):

- Basic probability and linear algebra.
- Programming proficiency in Python.
- Familiarity with machine learning concepts (e.g., supervised learning).

Pedagogy / Teaching Methodology:

- ✓ Interactive lectures with algorithmic walkthroughs.
- ✓ Coding labs using OpenAI Gym, PyTorch, or TensorFlow.
- ✓ Group projects with real-world datasets.
- ✓ Competitions (e.g., solving RL benchmarks).

Certificate / Value Added Courses Recommended:

1. Deep Reinforcement Learning Nanodegree (Udacity).
2. David Silver's RL Course (University College London).

Suggested Readings:

- Richard S. Sutton & Andrew G. Barto, *Reinforcement Learning: An Introduction*, 2nd Ed.
- Maxim Lapan, *Deep Reinforcement Learning Hands-On*, Packt.
- Research papers on AlphaGo, DQN, and PPO.

Recommended Software/Tools:

1. Python, PyTorch/TensorFlow.
2. OpenAI Gym, Unity ML-Agents.
3. Google Colab, Jupyter Notebooks

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO-PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	--	1	--	--	1	1
CO2	2	3	2	1	1	--	--	2	1
CO3	3	2	2	1	1	1	1	1	2
CO4	2	3	3	1	2	2	--	1	2
CO5	3	2	1	2	2	1	1	2	2
Articulation	2.6	2.4	1.8	1.2	1.4	1.3	1.0	1.4	1.6

Course Title:	**Fine-Tuning & Optimization**
Course Code:	**251VMT5S404**
Semester:	**IV**
Credits:	**4**
Course Type	**DSE**
Learning Hours	120
Live Sessions	**12 hours**

Course Description:

This course provides an in-depth exploration of fine-tuning and optimization techniques for machine learning and deep learning models. Students will learn to enhance model performance through hyperparameter tuning, regularization, and advanced optimization algorithms. The course covers both theoretical foundations and practical applications, with hands-on labs using frameworks like TensorFlow and PyTorch. Emphasis is placed on real-world case studies and performance benchmarking.

Course Outcomes (COs):

CO No.	Course Outcome	BTL
CO1	Understand the principles of model fine-tuning and optimization in machine learning.	L3
CO2	Apply hyperparameter tuning techniques to improve model performance.	L3
CO3	Analyze and implement advanced optimization algorithms.	L4
CO4	Evaluate trade-offs between model complexity, performance, and computational efficiency.	L5
CO5	Develop optimized models for real-world applications, ensuring scalability and robustness.	L3

No of Hours:24

Module 1: Introduction to Model Optimization

Scope of Module:
Introduces foundational concepts and challenges in model optimization.

Topics:
Understanding the model optimization pipeline; Bias-variance tradeoff and generalization; Underfitting vs. overfitting: causes and effects; Model capacity and expressiveness; Types of machine learning errors; Performance metrics: accuracy, precision, recall; F1-score, ROC-AUC, and average precision; Confusion matrix and error types (FP/FN); Visual diagnostics: learning curves and validation curves; Impact of data imbalance on evaluation; Cross-validation for reliable performance estimation; Model optimization vs. hyperparameter tuning; Importance of context in choosing metrics; Tooling: scikit-learn metrics and visualizations. Case study: Optimization trade-offs in healthcare ML.

Learning Outcome: *Students will be able to explain the role of optimization in ML pipelines and evaluate models using the appropriate performance metrics.*

No of Hours:24

Module 2: Hyperparameter Tuning

Scope of Module:
Covers techniques for selecting optimal hyperparameters.

Topics:
Distinction between parameters and hyperparameters; Common hyperparameters across ML models; Manual tuning: intuition and experience-driven search; Grid search: exhaustive evaluation strategy; Random search: efficient alternative for large spaces; Bayesian optimization and Gaussian processes; Tree-structured Parzen Estimator (TPE); Advanced

frameworks: Optuna and Ray Tune; Hyperparameter tuning in deep learning (batch size, dropout); Early stopping as an implicit tuning mechanism; Search space design and narrowing techniques; Use of validation sets for tuning; Hyperparameter transfer across tasks; Automation with Keras Tuner and Sklearn GridSearchCV; Practical lab: Hyperparameter tuning for a CNN.

Learning Outcome: *Students will be able to apply manual and automated hyperparameter tuning strategies to improve model performance.*

No of Hours:24

Module 3: Optimization Algorithms

Scope of Module:
Focuses on advanced optimization algorithms for training models.

Topics:
Loss functions and their role in optimization; Gradient-based optimization: fundamentals; Stochastic Gradient Descent (SGD) and its variants; Momentum and Nesterov Accelerated Gradient; Adaptive methods: AdaGrad, RMSprop, Adam, AdamW; Learning rate schedules: step decay, cosine annealing; Warm-up strategies and dynamic adjustments; Loss surface visualization and saddle points; Dealing with exploding and vanishing gradients; Optimizer selection for various tasks (CV, NLP, RL); Hands-on: Comparing optimizers on an NLP task; Learning rate finder method for tuning; Impact of batch size on convergence; Regularization synergy with optimization; Case study: Fine-tuning BERT for sentiment classification.

Learning Outcome: *Students will be able to implement and compare optimization algorithms and fine-tune learning configurations for diverse ML tasks.*

No of Hours:24

Module 4: Regularization and Efficiency

Scope of Module:
Explores techniques to prevent overfitting and improve model efficiency.

Topics:
Overfitting control through regularization; L1 and L2 regularization: Ridge and Lasso; Dropout: theory, implementation, and impact; Early stopping: theory and best practices; Batch normalization for stable training; Model pruning for size reduction; Quantization: dynamic, static, and QAT (quantization-aware training); Knowledge distillation: student-teacher frameworks; Tensor decomposition for lightweight models; Mixed-precision training using NVIDIA Apex; Distributed training fundamentals (DDP, Horovod);
Edge deployment constraints and model tuning; Practical lab: Deploying a quantized model with TensorFlow Lite; Efficiency-vs-accuracy trade-offs; Mini-project: Build and deploy a mobile-ready classification model.

Learning Outcome: *Students will be able to design and train models that are computationally efficient without sacrificing performance.*

No of Hours:24

Module 5: Real-World Applications and Deployment

Scope of Module:
Integrates optimization techniques into end-to-end workflows.

Topics:
Domain-specific optimization: CV, NLP, RL use cases; Model compression pipelines for deployment; Mobile and embedded deployment tools: TFLite, ONNX, Core ML; Model packaging with Docker for reproducibility; Serving models with Flask, FastAPI, TensorFlow Serving; API-based model interactions and RESTful services; CI/CD for ML workflows; ML observability and performance monitoring tools; Error feedback loops and model retraining triggers; Metrics for deployment: latency, throughput, load; Handling data drift in production Security and privacy in model serving; Trade-offs: explainability vs. complexity vs. speed; Capstone: Optimize and deploy a vision/NLP model; Final presentations: Results, trade-offs, and business impact.

Learning Outcome: *Students will be able to integrate optimization techniques into real-world deployment pipelines and deliver performant ML solutions.*

Module	No. of Pre-Recordings
1	10
2	12
3	10
4	12
5	10

Note: Min no of pre recordings 10.

Prerequisites (If Any):

- Basic knowledge of machine learning and deep learning.
- Proficiency in Python and familiarity with TensorFlow/PyTorch

Pedagogy / Teaching Methodology:

- ✓ Interactive lectures with mathematical derivations and visualizations.
- ✓ Hands-on coding labs using Jupyter Notebooks and cloud-based GPUs.
- ✓ Group projects focusing on optimization challenges.
- ✓ Performance benchmarking competitions

Certificate / Value Added Courses Recommended:

1. Deep Learning Specialization (Coursera, Andrew Ng).

2. Practical Deep Learning for Coders (fast.ai)

Suggested Readings:

- Ian Goodfellow et al., *Deep Learning*, MIT Press, 2016.
- Aurélien Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, 2nd Ed
- Research papers on optimization algorithms (e.g., Adam, NAS)

Recommended Software/Tools:

1. Python, TensorFlow, PyTorch.
2. Weights & Biases, MLflow for experiment tracking.
3. Google Colab, AWS SageMaker

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)

6.	Indian Knowledge System

CO-PO Mapping Table

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9
CO1	3	2	1	--	1	--	--	1	1
CO2	2	3	2	1	1	--	--	2	1
CO3	3	2	2	1	1	1	1	1	2
CO4	2	3	3	1	2	2	--	1	2
CO5	3	2	1	2	2	1	1	2	2
Articulation	2.6	2.4	1.8	1.25	1.4	1.25	1.0	1.4	1.6

Course Title:	**Strategic Leadership in IT**
Course Code:	**251VMT0C405**
Semester:	**IV**
Credits:	**2**
Course Type	**Skill Enhancement Course**
Learning Hours	60
Live Sessions	**12 hours**

Course Description:

This course explores the critical role of leadership in driving strategic IT initiatives within organizations. It focuses on leadership theories, IT governance, digital transformation strategies, risk management, and innovation leadership. Practical case studies and simulations are integrated to build decision-making skills required for aligning IT with business strategies.

Course Outcomes (Cos)

Upon successful completion of this course, learners will be equipped to:

CO	Course Outcome	BTL
CO1	Understand leadership theories and their application in IT strategy formulation.	L1
CO2	Evaluate and apply IT governance frameworks and practices.	L5
CO3	Design digital transformation strategies aligned with business goals.	L6
CO4	Identify and mitigate IT-related risks through strategic leadership.	L4
CO5	Demonstrate innovation and change management skills in IT leadership contexts.	L3

Course Modules

No. of Hours:12

Module 1: Foundations of Strategic Leadership and IT Alignment

Scope of Module:
Covers the core leadership theories and IT-business alignment models.

Topics:
Defining leadership and management in the context of IT; Strategic leadership competencies; Leadership styles: transformational, transactional, situational; Vision building and strategic thinking in IT; Models of IT-business alignment (Henderson & Venkatraman Model); Organizational structures and their impact on IT leadership; Emotional intelligence in IT leadership; Ethical leadership in technology environments; Leading virtual and remote IT teams; Leadership maturity models; Case Study: IT leadership challenges in global enterprises.

Learning Outcome: *Students will be able to understand leadership theories and how to align IT strategies with broader business objectives.*

No. of Hours:12

Module 2: IT Governance and Strategic Control

Scope of Module:
Focuses on frameworks and practices for effective IT governance.

Topics:
Introduction to IT governance and compliance; COBIT, ITIL, and ISO/IEC 38500 frameworks; Governance of Enterprise IT (GEIT) principles; Strategic decision rights and accountability in IT; Metrics and KPIs for IT governance; Managing outsourcing and vendor relationships; Sarbanes-Oxley Act and IT implications; Risk appetite and tolerance in IT governance; Blockchain's role in IT governance; Shadow IT and governance challenges; Audit management and IT policy development; Practical Exercise: Designing an IT governance framework for an SME.

Learning Outcome: *Students will be able to evaluate, select, and apply IT governance frameworks to ensure strategic control and value delivery.*

No. of Hours:12

Module 3: Digital Transformation Leadership

Scope of Module:
Develops skills to lead digital transformation initiatives.

Topics:
Drivers of digital transformation in organizations; Role of CIOs and CTOs in transformation efforts; Technology adoption frameworks (e.g., Diffusion of Innovations); Agile and DevOps in digital leadership; Managing cultural change during transformation; Leadership in cloud, AI, and emerging technologies adoption; Case Study: Successful vs. failed digital transformation initiatives.

Learning Outcome: *Students will be able to design strategic digital transformation initiatives aligned with business innovation needs.*

No. of Hours:12

Module 4: IT Risk Management and Strategic Decision-Making

Scope of Module:
Focuses on managing strategic risks in IT leadership roles.

Topics:
Understanding IT-related risks: security, operational, reputational; Risk management lifecycle in IT projects; Enterprise risk management frameworks (COSO ERM); Crisis leadership and contingency planning; Ethical considerations and cybersecurity leadership; Legal and regulatory compliance (e.g., GDPR, HIPAA); Threat modeling and strategic defense planning;

Business continuity planning and IT disaster recovery leadership; Strategic handling of data breaches and incident response, Practical: IT risk assessment simulation

Learning Outcome: *Students will be able to analyze, manage, and mitigate IT risks strategically to ensure organizational resilience.*

*N*o. of Hours:12

Module 5: Innovation, Change Management, and Future of IT Leadership

Scope of Module:
Covers fostering innovation and managing technological change.

Topics:
Leading organizational innovation in IT; Change management models (Kotter's 8-Step, ADKAR); Talent development and IT leadership pipelines; Diversity, inclusion, and ethics in IT leadership; Future trends: Quantum Computing, Web3, and Leadership implications;

Building a sustainable IT innovation culture; Intrapreneurship and fostering innovation within IT teams; Managing resistance to technological change; IT leadership in the age of Automation and RPA (Robotic Process Automation); Ethical considerations in AI and emerging tech leadership; Capstone: Strategic Leadership Plan for a tech startup.

Learning Outcome: *Students will be able to demonstrate strategic leadership skills in driving innovation, change management, and future-proofing IT organizations*

Module	**No. of Pre-Recordings**
1	**10**
2	**12**
3	**10**
4	**12**
5	**10**

Note: Min no of pre recordings 10.

Prerequisites (If Any):

- Basic knowledge of IT management concepts and organizational behavior.

Pedagogy / Teaching Methodology:

- ✓ Interactive case studies and leadership simulations
- ✓ Group strategy projects and role plays
- ✓ Workshops on IT governance frameworks
- ✓ Industry expert talks and guest lectures

Certificate / Value Added Courses Recommended:

1. IT Leadership Certificate by edX (MIT Sloan)
2. COBIT 2019 Foundation Certification
3. Digital Transformation Strategy by Coursera (Boston University)

Suggested Readings:

- John P. Kotter, Leading Change, Harvard Business Review Press, 2012.
- Peter Weill & Jeanne W. Ross, IT Governance: How Top Performers Manage IT Decision Rights for Superior Results, Harvard Business Review Press, 2004.
- George Westerman, Leading Digital: Turning Technology into Business Transformation, Harvard Business Review Press, 2014.

Recommended Software/Tools:

1. MS Project, Trello, Jira (for project leadership)
2. ServiceNow, SolarWinds (for IT governance)
3. Risk assessment tools (RSA Archer, RiskLens)

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global
2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

CO PO Mapping Table

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**
CO1	3	2	1	-	2	1	-	2	1
CO2	3	3	2	1	2	1	-	2	-
CO3	2	3	3	2	3	2	1	3	2
CO4	3	2	2	2	2	1	1	2	2
CO5	2	2	2	1	3	2	1	2	2
Articulation	2.6	2.4	2	1.5	2.4	1.4	1	2.2	1.4

Module	No of Pre-Recordings
1	29
2	18
3	22
4	21
5	9

Prerequisites (If Any):

- Basic understanding of programming concepts and computer architecture.

Pedagogy / Teaching Methodology:

- Lectures & Discussions: Instructor-led sessions to explain key concepts.
- Hands-on Exercises: Practical tasks to apply learned skills.
- Case Studies: In-depth analysis of real or simulated situations to reinforce learning.

- Assessments: Quizzes or Assignments to evaluate understanding and progress.
- Tools & Platforms: Software and technologies used to support learning and practice.

Certificate / Value Added Courses Recommended (with Free Resource Links):

- Course Name: Shell Scripting Crash Course

Platform: YouTube

Link: https://www.youtube.com/watch?v=Oxoj4hEHb5A

Suggested Readings:

- *Operating System Concepts – Abraham Silberschatz, Peter B. Galvin, Greg Gagne, Wiley, 2022*
- *Cloud Security and Privacy – Tim Mather, O'Reilly, 2009*

Recommended Software/Tools:

- *Ubuntu/Linux OS*
- *Bash Shell*
- *vi/vim Editor*
- *awk, sed*
- *VirtualBox*

Curriculum Development:

Sl.No	Identify and Highlight in the Modules
1	Local/Regional/National/Global

2	Employability/Entrep/ Skill Development
3	Value Added Courses(Module)
4	Integration of Digital Learning Tools
5	Cross Cutting Edges(Ethics, Gender Equality, Sustainability, Envt Conserns)
6.	Indian Knowledge System

Note: It is not necessary that all the courses must have all the above-mentioned points

CO-PO Mapping Table

	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**
CO1	3	3	2	2	1	1	1	2	3
CO2	2	3	2	3	1	1	1	3	2
CO3	1	2	3	3	3	2	2	3	2
CO4	1	2	2	3	2	3	2	2	2
CO5	1	1	1	2	3	3	3	1	3
Articulation	1.6	2.2	2	2.6	2	2	1.8	2.2	2.4

CENTRE FOR
DISTANCE AND
ONLINE EDUCATION

ONLINE