Dr. Ambedkar Institute of Technology

(An Autonomous Institute affiliated to VTU, Accredited by NAAC with 'A+' grade)



School of Computer Science and Engineering
Computer Science and Engineering Programme

Scheme of Teaching and Examinations
for
M.Tech. in Computer Science and Engineering

Scheme of 1st & 2nd Semester
Academic Year 2026-2027 to 2027-2028
Choice Based Credit System (CBCS) and Outcome Based Education (OBE)

Dr. AMBEDKAR INSTITUTE OF TECHNOLOGY

School of Computer Science and Engineering

Computer Science and Engineering Programme

Scheme of Teaching and Examinations – 2026-27, 2027-28

M.Tech in Computer Science and Engineering

I SEMESTER (Core Courses related to main Engineering Stream)

	Course Type	Course Code	Course Title	Teaching Hours per Week			TW &SL	Total Number of Hours /Semester	Examination				
				Theory	Practical/ Seminar	Tutorial/SDA			Duration in hours	CIE Marks	SEE Marks	Total Marks	
				L	P	T/SDA							
1	PCC	1MSCST101	Probability Statistics and Queueing Theory	03	00	00	48	90	03	50	50	100	3
2	PCC	1MSCST102	Advanced Algorithms and Data Structures	03	00	00	48	90	03	50	50	100	3
3	PCC	1MSCST103	Data Science and Machine Learning Techniques	03	00	00	48	90	03	50	50	100	3
4	IPCC	1MSCSU104X	Integrated Professional Core Course - I	03	00	28	50	120	03	50	50	100	4
5	PEC	1MSCS105X	Professional Elective Course - I	03	00	00	48	90	04	50	50	100	3
6	PEC	1MSCS106X	Professional Elective Course - II	00	03	00	48	90	03	50	50	100	3
7	PCCL	1MSCSL107X	Professional Core Course Laboratory	00	28	02	30	30	03	50	50	100	1
9	NCMC	1MRMI107	Research Methodology and IPR	**Online courses (online.vtu.ac.in)**									PP/NP
10	NCMC	1MCDN108	Career Development Skills	**offline**									PP/NP

	TOTAL	**350**	**350**	**700**	**20**

Integrated Professional Core Courses (IPCC)	
Code	**Title of the Course**
1MSCSU104A	Advanced Blockchain Technologies
1MSCSU104B	Advanced Computer Networks
1MSCSU104C	Advanced Bigdata Analytics
1MSCSU104D	Deep Learning

Professional Elective Courses-I	
Code	**Title of the Course**
1MSCS105A	Data Analytics for Fraud Detection
1MSCS105B	Data Engineering
1MSCS105C	Artificial Intelligence for Cyber Security
1MSCS105D	Artificial Intelligence and IoT

Professional Elective Courses-II (PEC)	
Code	**Title of the Course**
1MSCS106A	Generative Models
1MSCS106B	Multi Modal Data Analytics
1MSCS106C	Explainable AI
1MSCS106D	Software Engineering and Web Applications

Professional Core Courses Laboratory-I (PCCL)	
Code	**Title of the Course**
1MSCSL107A	Software Automation Testing Laboratory
1MSCSL107B	Data Science and Machine Learning Laboratory
1MSCSL107C	Advanced Algorithm and Data Structure Laboratory
1MSCSL107D	Microservices Development Laboratory

HEAD of School (HOS)
School of Comp... and Engineering
Dr. Ambedk... Technology
Be...

Dr. AMBEDKAR INSTITUTE OF TECHNOLOGY

School of Computer Science and Engineering

Computer Science and Engineering Programme

Scheme of Teaching and Examinations – 2026-27, 2027-28

M.Tech in Computer Science and Engineering

II SEMESTER (Core Courses related to main Engineering Stream)

	Course Type	Course Code	Course Title	Teaching Hours per Week			TW &SL	Total number of Hours /semesters	Examination				
				Theory	Practical/Seminar	Tutorial/SDA			Duration in hours	CIE Marks	SEE Marks	Total Marks	
				L	P	T/SDA							
1	IPCC	1MSCSU201	Digital Forensics	03	00	28	50	120	04	50	50	100	4
2	PCC	1MSCST202	Cryptography and Network Security	03	00	00	48	90	03	50	50	100	3
3	PCC	1MSCST203	Soft Computing	03	00	00	48	90	03	50	50	100	3
4	PCC	1MSCST204	Data Analytics	03	00	28	48	90	03	50	50	100	3
5	PEC	1MSCS205X	Professional Elective Course - I	02	02	00	48	90	04	50	50	100	3
6	PEC	1MSCS206X	Professional Elective Course - II	00	03	00	48	90	03	50	50	100	3
7	PCCL	1MSCSL207	Data Analytics Lab	00	28	02	30	30	03	50	50	100	1
8	PCC	1MSCSS208	Minor Project (Project Phase-I)				60	60	03	50	50	100	2

9	NCMC	1MCDN209	Career Development Skills	offline				PP/NP
TOTAL					**400**	**400**	**800**	**22**

Professional Elective Courses-I	
Code	**Title of the Course**
1MSCSU205A	Quantum Computing
1MSCSU205B	Fog and Edge Computing
1MSCSU205C	High Performance Computing
1MSCSU205D	Intelligent Computing

Professional Elective Courses-II (PEC)	
Code	**Title of the Course**
1MSCS206A	Software Defined Networks
1MSCS206B	Prompt Engineering
1MSCS206C	Genetic Algorithms
1MSCS206D	Mobile and Web Application Security

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

Course Title	**Probability Statistics and Queueing Theory**				
Course Code	**1MSCST101**				
Course Category	**PCC (Professional Core Course)**				
Teaching - Learning Scheme & Credit	**L**	**T**	**P**	**Total Credits**	**Total Teaching Hours**
	3	**0**	**0**	**3**	**Lecture: 42 + Tutorials: 0 + Practical: 0 + Self Study: 48 = 90**
CIE Marks: 50	**SEE Marks: 50**		**Total Max. Marks: 100**		**Duration of SEE: 03 Hours**

Course Learning Objectives:	
CLO1	To develop analytical capability and to impart knowledge of Probability, Statistics and Queuing.
CLO2	Master standard theoretical models used to map random phenomena.
CLO3	Formulate relationships between multiple random traits or variables.
CLO4	Make reliable decisions and extract meaningful conclusions from sample data.
CLO5	Students acquire knowledge of Hypothesis testing and Queueing methods and their applications so as to enable them to apply them for solving real world problems.

SYLLABUS	**No. of Hours**	**BTLs**
UNIT 1	**8**	**L1-L4**
Probability: Axioms of Probability, Conditional probability, Total probability, Bayes' theorem, Discrete Random variable, Probability mass function, Continuous Random variable. Probability density function, Cumulative Distribution Function, and its properties, Engineering Application:		
SELF-STUDY COMPONENT	**10 Hours**	
Moments of a random variable, Optical communication system		
UNIT 2	**8**	**L1-L4**
Standard Probability Distributions: Discrete distributions: Binomial, Poisson, Geometric and their properties. Continuous distributions: Normal, exponential Weibull distributions and their properties. Two-dimensional Random variables, Joint pdf / cdf and their properties.		
SELF-STUDY COMPONENT	**10 Hours**	
Gamma distribution, Entropy and Source coding		
UNIT 3	**9**	**L1-L4**

Semester: I **Scheme: 2026**

Testing Hypothesis: Testing of Hypothesis: Formulation of Null hypothesis, critical region, level of significance, errors in testing, Test2s of significance for Large and Small Samples, t-distribution, its properties and uses.	10 Hours	
SELF-STUDY COMPONENT	**10 Hours**	
F-distribution, its properties and uses, Chi-square distribution		
UNIT 4	**9**	**L1-L4**
Random Processes: Classification, Methods of description, Special classes, Average values of Random Processes, Analytical representation of Random Process, Auto correlation Function, Cross-correlation function and their properties, Markov Process, Markov chain.		
SELF-STUDY COMPONENT	**9**	**Hours**
Ergodicity and Poisson process, Chapman-Kolmogorov theorem		
UNIT 5	**8**	**L1-L4**
Queueing Theory: Symbolic Representation of a Queuing Model, Poisson Queue system, Little Law, Types of Stochastic Processes, Birth-Death Process, The M/M/1 Queuing System, The M/M/s Queuing System.		
SELF-STUDY COMPONENT	**9**	**Hours**
The M/M/s Queuing with Finite buffer.		

TEXT BOOKS	
1.	Probability, Statistics and Queuing Theory, V. Sundarapandian, Eastern Economy Edition, PHI Learning Pvt. Ltd, 2009. Published by PHI Learning, New Delhi (2009)

REFERENCE BOOKS	
1.	Probability and Random Processes, Scott L.Miller and Donald Childers, Sixth indian reprint,2018.
2.	Probability & Statistics with Reliability, Queuing and Computer Applications, 2nd Edition by Kishor. S. Trivedi , Prentice Hall of India, 2004. ISBN: 978-0-471-33341-8.
3.	Probability ,Queing theory and Engineering Reliability, Haribhaskaran, First Edition ,Laxmi Publication 2005.

e-LEARNING RESOURCES
https://www.probabilitycourse.com/ https://www.mheducation.co.in/probability-and-statistics-random-variables-and-queuing-theory-9789364449038-india

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

	Assessment Details for both CIE and SEE
	The weightage assigned to the Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) is 50% each. To pass the course, students must secure a minimum of 40% marks separately in both CIE and SEE. **Continuous Internal Evaluation (CIE)** The CIE consists of four components: 1. CIE – Theory Component 2. Assignment 3. Group Activity 4. Seminar Students must secure a minimum of 50% marks in each component CIE Theory, Assignment, and Group Activity components to qualify for appearing in the SEE. **Semester End Examination (SEE)** The Theory SEE is conducted for 100 marks with a duration of 3 hours. The marks obtained are subsequently scaled down to 50 marks for calculating the final course grade. The student must secure a minimum of 40% marks in the SEE to pass the course. ****Self-Study Component should be included for CIE.**

CONTINUOUS INTERNAL EVALUATION (CIE)		**Max. Marks**	**50 %**	**Minimum Marks Required in CIE to Qualify for SEE (50% individually)**
Theory	**Weightage of CIE-1 and CIE-2 or CIE-3**	**30**	**15**	**25**
	Assignment	**10**	**05**	
	Group Activity	**05**	**2.5**	
	Seminar	**05**	**2.5**	
	TOTAL	**50**	**25**	

SEE QUESTION PAPER PATTERN										
Q. No.	**Q1**	**Q2**	**Q3**	**Q4**	**Q5**	**Q6**	**Q7**	**Q8**	**Q9**	**Q10**
UNIT	1		2		3		4		5	
1. Two full questions (each of 20 Marks) are to be set from each unit.										
2. Student shall answer five full questions selecting one full question from each unit.										

Course Outcomes: After the completion of the course, the student will be able to	
CO1	Students will demonstrate knowledge & use of probability and will be able to characterize probability models using probability mass (density) functions & cumulative distribution functions.

Semester: I **Scheme: 2026**

CO2	Students will be introduced to the techniques of developing discrete & continuous probability distributions and its applications.
CO3	Students will be able to describe a random process in terms of its mean and correlation functions.
CO4	Students will be introduced to methods of Hypothesis testing for goodness of fit.
CO5	Students will be able to understand the terminology & nomenclature appropriate queuing theory and also demonstrate the knowledge and understand the various queuing models.

COURSE SUSTAINABLE DEVELOPMENT GOALS (SDGs)		
Goal No.	**Goal Title**	**Description**
4	Quality Education	Operating Systems develops problem-solving, programming, and system design skills through concepts such as process management, memory management, synchronization, and file systems. It equips students with essential computing knowledge for higher education and lifelong learning.
8	Decent Work and Economic Growth	The course prepares students with industry-relevant skills in multitasking, multithreading, virtualization, resource management, and cloud computing, enhancing employability and supporting innovation and productivity in the IT sector.

Mapping of COs with POs
Strength of correlation: Strongly related-3, Moderately related-2, Weakly related-1, Not related-0

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**
CO1	3	2	0	0	2
CO2	3	3	2	2	2
CO3	3	3	3	2	2
CO4	3	3	2	2	3
CO5	3	2	2	1	3

Dr. Ambedkar Institute of Technology

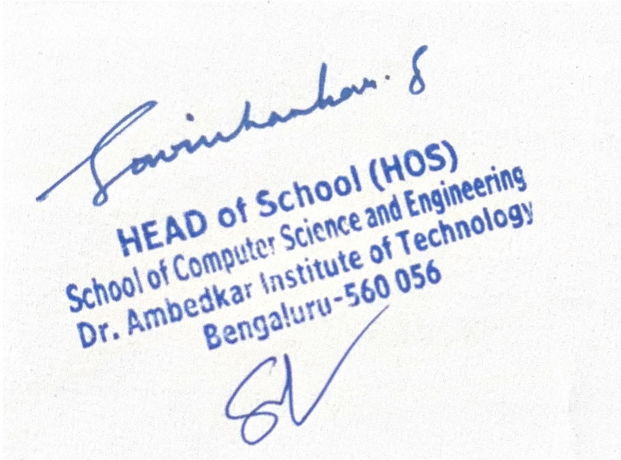
School of Computer Science and Engineering

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

Course Coordinators:

G.S.Shivaprasanna
Dr. Shivaprasanna G S Associate Professor Mathematics Programme



Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

Course Title	Advanced Data Structure and Algorithms				
Course Code	1MSCST102				
Course Category	PCC (Professional Core Course)				
Teaching - Learning Scheme & Credit	L	T	P	Total Credits	Total Teaching Hours
	3	0	0	3	Lecture: 42 + Tutorials: 0 + Practical: 0 + Self Study: 48 = 90
CIE Marks: 50	SEE Marks: 50		Total Max. Marks: 100		Duration of SEE: 03 Hours

Course Learning Objectives:	
CLO1	Implementing iterative and recursive optimized solutions for real time problems
CLO2	Developing graph search algorithms for real time applications.
CLO3	Adopting efficient heap operations and Hashing techniques to improve the efficiency of data access
CLO4	Implementing optimal algorithms for NLP

SYLLABUS	No. of Hours	BTLs
UNIT 1	**8**	**L1-L4**
Review of Analysis Techniques: Growth of Functions: Asymptotic notations; Standard notations and common functions; Recurrences and Solution of Recurrence equations- The Course substitution method, The recurrence – tree method, The master method		
SELF-STUDY COMPONENT	**10 Hours**	
Analysis of time complexity of real time applications, Amortized Analysis: Aggregate, Accounting and Potential Methods		
UNIT 2	**8**	**L1-L4**
Graph Algorithms: Bellman - Ford Algorithm; Single source shortest paths in a DAG; Johnson's Algorithm for sparse graphs; Flow networks and Ford-Fulkerson method.		
SELF-STUDY COMPONENT	**10 Hours**	
Time complexity analysis of any graph based real time application. Maximum bipartite matching. Floyd-Marshalls Algorithm		
UNIT 3	**9**	**L1-L4**
String-Matching Algorithms: Naïve string Matching; Rabin - Karp algorithm; String matching with finite automata; Knuth-Morris-Pratt algorithm; Boyer – Moore algorithms		
SELF-STUDY COMPONENT	**10 Hours**	

NP Completeness problems: Polynomial time, verification and NP completeness and reducibility.		
UNIT 4	**9**	**L1-L4**
Fundamental Heap Structures and Hashing: Heap Structures: Heaps, Maintaining the Heap Property, Building a Heap, Priority Queues. Hashing Fundamentals: Direct Address Tables, Hash Tables, Hash Functions		
SELF-STUDY COMPONENT	**9**	**Hours**
Binary Search Trees; B-Trees; Dynamic Hashing		
UNIT 5	**8**	**L1-L4**
Balanced Trees and Mergeable Heap Structures: Red-Black Trees: Properties of Red-Black Trees, Rotations, Insertion, Deletion. Mergeable Heap Structures: Leftist Trees, Binary Heaps		
SELF-STUDY COMPONENT	**9**	**Hours**
Binomial Heaps, Fibonacci Heaps; Data structures for disjoint sets		

TEXT BOOKS	
1.	Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, Introduction to Algorithms, 4th Edition, (2022), MIT Press.
2	Ellis Horowitz and Sartaj Sahni, Fundamentals of Data Structures in C, 2nd Edition (2023), Universities Press(India) Private limited.

REFERENCE BOOKS	
1.	Mark Allen Weiss, Data Structures and Algorithm Analysis in C++, 4th Edition, (2014) Pearson Education.
2.	Algorithm Design: Kleinberg and Tardos, Pearson Education India, 2006 - Algorithms
3.	Ellis Horowitz, SartajSahni, S.Rajasekharan: Fundamentals of Computer Algorithms, 2nd Edition, Universities press, 2007, ISBN 8173716129, 9788173716126

e-LEARNING RESOURCES
Introduction to Algorithms and Analysis By Prof. Sourav Mukhopadhyay IIT Kharagpur https://onlinecourses.nptel.ac.in/e-learning/preview/noc26_cs163
Algorithms by Jeff Erickson; https://jeffe.cs.illinois.edu/teaching/algorithms/
GeeksforGeeks – Advanced Data Structures Tutorials: https://www.geeksforgeeks.org/dsa/dsa-tutorial-learn-data-structures-and-algorithms/
MIT OpenCourseWare – Advanced Data Structures: https://ocw.mit.edu/courses/6-851-advanced-data-

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

structures-spring-2012/

	Assessment Details for both CIE and SEE
)	The weightage assigned to the Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) is 50% each. To pass the course, students must secure a minimum of 50% marks in CIE and 40% marks in SEE. **Continuous Internal Evaluation (CIE)** The CIE consists of Four components: 1. CIE – Theory Component 2. Assignment 3. Group Activity 4. Seminar Students must secure a minimum of 50% marks in each component CIE Theory, Assignment, and Group Activity components to qualify for appearing in the SEE. **Semester End Examination (SEE)** The Theory SEE is conducted for 100 marks with a duration of 3 hours. The marks obtained are subsequently scaled down to 50 marks for calculating the final course grade. The student must secure a minimum of 40% marks in the SEE to pass the course. ****Self-Study Component should be included for CIE.**

CONTINUOUS INTERNAL EVALUATION (CIE)		Max. Marks	50 %	Minimum Marks Required in CIE to Qualify for SEE (50% individually)
Theory	Weightage of CIE-1 and CIE-2 or CIE-3	30	15	25
	Assignment	10	05	
	Group Activity	05	2.5	
	Seminar	05	2.5	
	TOTAL	50	25	

SEE QUESTION PAPER PATTERN										
Q. No.	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
UNIT	1		2		3		4		5	
1. Two full questions (each of 20 Marks) are to be set from each unit.										
2. Student shall answer five full questions selecting one full question from each unit.										

Course Outcomes: After the completion of the course, the student will be able to	
CO1	Analyze and determine the time complexity of iterative algorithms using appropriate asymptotic notations.

Semester: I **Scheme: 2026**

CO2	Analyze and determine the time complexity of recursive algorithms using recurrence relations and appropriate analysis techniques.
CO3	Analyze and Design graph based algorithms to solve real-world computational problems.
CO4	Analyze and Design algorithms for string Based algorithms for real time problems
CO5	Analyze and implement appropriate data structures and their operations to enable efficient data retrieval.

COURSE SUSTAINABLE DEVELOPMENT GOALS (SDGs)		
Goal No.	**Goal Title**	**Description**
4	Quality Education	Develops advanced problem-solving, computational thinking, analytical skills, and algorithmic competency.
8	Decent Work and Economic Growth	Builds skills required for software development, AI, data science, optimization, and other technology-driven careers.
9	Industry, Innovation and Infrastructure	Provides foundations for designing efficient algorithms and scalable computational systems that support technological innovation.

Mapping of COs with POs

Strength of correlation: Strongly related-3, Moderately related-2, Weakly related-1, Not related-0

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**
CO1	3	1	3	3	2
CO2	3	1	3	2	2
CO3	3	2	3	3	3
CO4	3	2	3	2	3
CO5	3	1	3	2	3

Course Coordinators:

27/10/2026

Dr. K R Shylaja
Professor and DEAN (R&D)
CSE Programme

HEAD of School (HOS)
School of Computer Science and Engineering
Dr. Ambedkar Institute of Technology
Bengaluru-560 056

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

Course Title	**Data Science & Machine Learning Techniques**				
Course Code	**1MSCST103**				
Course Category	**PCC (Professional Core Course)**				
Teaching - Learning Scheme & Credit	**L**	**T**	**P**	**Total Credits**	**Total Teaching Hours**
	3	**0**	**0**	**3**	**Lecture: 42 + Tutorials: 0 + Practical: 0 + Self Study: 48 = 90**
CIE Marks: 50	**SEE Marks: 50**		**Total Max. Marks: 100**		**Duration of SEE: 03 Hours**

Course Learning Objectives:	
CLO1	Explain the principles of data science, analytics, artificial intelligence, data governance, and their role in data-driven decision making
CLO2	Explain the statistical analysis, visualization, predictive analytics to solve real-world data-driven problems
CLO3	Understanding the machine learning principles and their applications in real-world scenarios.
CLO4	Compare different machine learning approaches and justify the approach.
CLO5	Understand the current research trends in machine learning through literature review.

SYLLABUS	**No. of Hours**	**BTLs**
UNIT 1	**8**	**L1-L4**
Predictive Analytics and Decision Support: Nature of Data, Data Sources, Statistical Analysis, Descriptive Analytics, Data Visualization for Decision Making. **Design a Learning System** – Concept Learning Task – Concept Learning as Search – Finding a Maximally Specific Hypothesis – Version Spaces – Find S, Candidate Elimination Algorithm.		
SELF-STUDY COMPONENT	**9 Hours**	
Dashboard Concepts, Predictive Modeling, Model Evaluation and Cost-Benefit Analysis. Gradient Descendent, Maximally Approximate Hypothesis		
UNIT 2	**8**	**L1-L4**
Statistical Learning, What Is Statistical Learning? Why Estimate f?, How Do We Estimate f?, The Trade-Off Between Prediction Accuracy and Model Interpretability, Supervised Versus Unsupervised Learning, Regression Versus Classification Problems, Assessing Model Accuracy, Measuring the Quality of Fit , The Bias-Variance Trade-Off.		
SELF-STUDY COMPONENT	**10 Hours**	
Polynomial regression, Step Functions, Basis Functions evaluating regression fit, Ridge		

Regression, The Lasso		
UNIT 3	**9**	**L1-L4**
Simple Linear Regression, Estimating the Coefficients, Assessing the Accuracy of the Coefficient Estimates, Assessing the Accuracy of the Model, Multiple Linear Regression, Estimating the Regression Coefficients ***Logistic regression*:** Introduction, Model specification, Model fitting; Multi-class logistic regression		
SELF-STUDY COMPONENT	**10 Hours**	
Steepest descent, L2 regularization, Principal components analysis (PCA) Cross-Validation, The Validation Set Approach, Leave-One-Out Cross-Validation, *k*-Fold Cross-Validation, Curse of Dimensionality, Best Subset Selection, Forward and Backward Stepwise Selection.		
UNIT 4	**9**	**L1-L4**
Decision tree learning: Introduction, Decision tree representation, appropriate problems for decision tree learning, the basic decision tree algorithm, hypothesis space search in decision tree learning, inductive bias in decision tree learning, issues in decision tree learning, Instance based Learning: K nearest neighbor, Regression Trees, Classification Trees		
SELF-STUDY COMPONENT	**9 Hours**	
Trees Versus Linear Models, Advantages and Disadvantages of Trees; Random Forests, Clustering: Introduction, K-mean clustering, agglomerative hierarchical clustering		
UNIT 5	**8**	**L1-L4**
Probability and Bayes Learning: Bayesian Learning, Naïve Bayes, Python exercise on Naïve Bayes, **Support Vector Machine**: Introduction, the Dual formulation, Maximum margin with noise, nonlinear SVM and Kernel function.		
SELF-STUDY COMPONENT	**10 Hours**	
Using Bayes' Theorem for Classification, Linear Discriminant Analysis for p=1, Linear Discriminant Analysis for p >1, Quadratic Discriminant Analysis; Ensemble learning: Bagging, Stacking, Boosting, Error-correcting output codes		

TEXT BOOKS	
1.	Gareth James, Daniela Witten, Trevor Hastie and Robert Tibshirani, An Introduction to Statistical Learning with Applications in R, ISSN 1431-875X,ISBN 978-1-4614-7137-0 ISBN 978-1-4614-7138-7 (eBook), DOI 10.1007/978-1-4614-7138-7,2015,Springer Publication
2	Foster Provost and Tom Fawcett, Data Science for Business: What You Need to Know About Data Mining and Data-Analytic Thinking, ,2nd Edition, O'Reilly Media, USA, 2023
3	Machine Learning, Tom M. Mitchell, Indian Edition, 2013, McGraw-Hill Education, ISBN:978-1-25-909695-2

REFERENCE BOOKS	
1.	Alpaydin, Ethem. Introduction to machine learning. MIT press, 2020.

2.	Kevin P. Murphy, "Machine Learning: A Probabilistic Perspective", MIT Press,
3.	Ramesh Sharda, Dursun Delen and Efraim Turban Business Analytics, Data Science and AI: A Managerial Approach, ,6th Edition, Pearson Education, 2026.

e-LEARNING RESOURCES
Google Data Analytics: https://grow.google/intl/en_in/data-analytics-course/
Elements of AI: https://course.elementsofai.com/?utm_source=chatgpt.com
https://www.coursera.org/browse/data-science/machine-learning
https://nptel.ac.in/courses/106106139 NPTEL Course IIT Madras
http://www.digimat.in/nptel/courses/video/106105152/L01.html

	Assessment Details for both CIE and SEE
	The weightage assigned to the Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) is 50% each. To pass the course, students must secure a minimum of 50% marks separately in each component of CIE and 40% in SEE. **Continuous Internal Evaluation (CIE)** The CIE consists of four components: 1. CIE – Theory Component 2. Assignment 3. Group Activity 4. Seminar Students must secure a minimum of 50% marks in each component CIE Theory, Assignment, and Group Activity components to qualify for appearing in the SEE. **Semester End Examination (SEE)** The Theory SEE is conducted for 100 marks with a duration of 3 hours. The marks obtained are subsequently scaled down to 50 marks for calculating the final course grade. The student must secure a minimum of 40% marks in the SEE to pass the course. ****Self-Study Component should be included for CIE.**

CONTINUOUS INTERNAL EVALUATION (CIE)		**Max. Marks**	**50 %**	**Minimum Marks Required in CIE to Qualify for SEE (50% individually)**
Theory	**Weightage of CIE-1 and CIE-2 or CIE-3**	**30**	**15**	**25**
	Assignment	**10**	**05**	
	Group Activity	**05**	**2.5**	
	Seminar	**05**	**2.5**	
	TOTAL	**50**	**25**	

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

SEE QUESTION PAPER PATTERN										
Q. No.	**Q1**	**Q2**	**Q3**	**Q4**	**Q5**	**Q6**	**Q7**	**Q8**	**Q9**	**Q10**
UNIT	1		2		3		4		5	
1. Two full questions (each of 20 Marks) are to be set from each unit.										
2. Students shall answer five full questions selecting one full question from each unit.										

Course Outcomes: After the completion of the course, the student will be able to	
CO1	Analyze and apply the principles of data science, analytics in data-driven decision making
CO2	Analyze and apply statistical analysis, visualization and predictive analytics to solve real-world data-driven problems
CO3	Analyze and apply machine learning principles and their applications to address real-world scenarios.
CO4	Analyze the strengths and weaknesses of different machine learning approaches and justify the choice of methods for specific tasks

COURSE SUSTAINABLE DEVELOPMENT GOALS (SDGs)		
Goal No.	**Goal Title**	**Description**
4	Quality Education	Operating Systems develops problem-solving, programming, and system design skills through concepts such as process management, memory management, synchronization, and file systems. It equips students with essential computing knowledge for higher education and lifelong learning.
8	Decent Work and Economic Growth	The course prepares students with industry-relevant skills in multitasking, multithreading, virtualization, resource management, and cloud computing, enhancing employability and supporting innovation and productivity in the IT sector.
9	Industry, Innovation and Infrastructure	Enables students to develop innovative data-driven and AI/ML solutions for industrial and technological applications.

Mapping of COs with POs

Strength of correlation: Strongly related-3, Moderately related-2, Weakly related-1, Not related-0

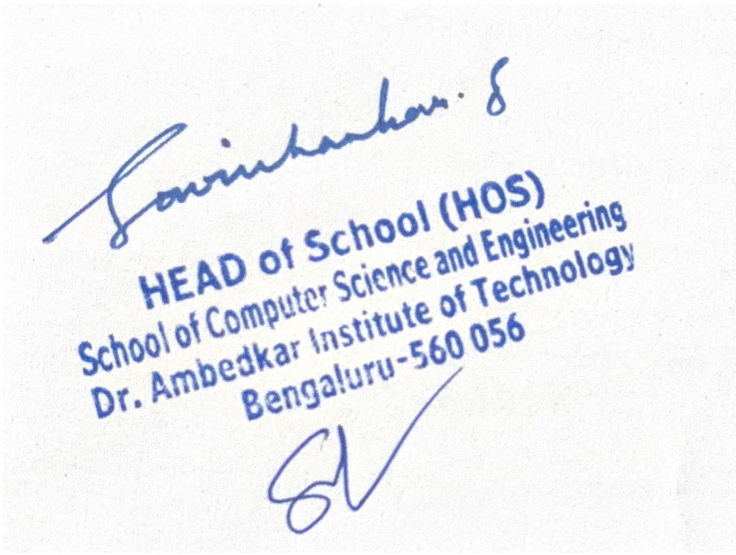
COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**
CO1	3	1	3	3	2

Semester: I **Scheme: 2026**

CO2	3	2	3	3	3
CO3	3	2	3	3	3
CO4	3	2	3	3	3
CO5	3	2	3	3	2

Course Coordinators:

27/10/2026
Dr. K R Shylaja Professor and DEAN (R&D) CSE Programme



Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

Course Title	**ADVANCED BLOCK CHAIN TECHNOLOGIES**				
Course Code	**1MSCSU104A**				
Course Category	**IPCC (Integrated Professional Core Course)**				
Teaching - Learning Scheme & Credit	**L**	**T**	**P**	**Total Credits**	**Total Teaching Hours**
	3	**0**	**1**	**4**	**Lecture: 42 + Tutorials: 0 + Practical: 28 + Self Study: 50 = 120**
CIE Marks: 50	**SEE Marks: 50**		**Total Max. Marks: 100**		**Duration of SEE: 04 Hours**

Course Learning Objectives:	
CLO1	Analyze blockchain architecture, decentralization principles, and cryptographic foundations.
CLO2	Evaluate advanced consensus protocols and blockchain scalability solutions.
CLO3	Implement smart contracts and decentralized applications using Ethereum.
CLO4	Compare enterprise blockchain frameworks and their application domains.
CLO5	Assess blockchain security, privacy-preserving mechanisms, and emerging decentralized technologies.

SYLLABUS	**No. of Hours**	**BTLs**
UNIT 1	**9**	**L1-L4**
Blockchain Fundamentals and Distributed Ledger Technology: Blockchain Fundamentals and Distributed Ledger Technology: Introduction to Blockchain, Distributed Systems, CAP Theorem and FLP Impossibility, Decentralization using Blockchain, Types of Blockchains, Consensus Mechanisms, Cryptographic Primitives.		
SELF-STUDY COMPONENT	**10 Hours**	
Hash Functions, Public Key Cryptography, Digital Signatures, Merkle Trees.		
UNIT 2	**9**	**L1-L4**
Advanced Consensus and Scalability Mechanisms: Byzantine Generals Problem, Proof of Work (PoW), Proof of Stake (PoS), Delegated Proof of Stake (DPoS), Practical Byzantine Fault Tolerance (PBFT), Proof of Authority (PoA), Federated Consensus, Blockchain Scalability Challenges.		

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

SELF-STUDY COMPONENT	**10 Hours**	
Segregated Witness (SegWit), State Channels, Sidechains.		
UNIT 3	**8**	**L1-L4**
Ethereum, Smart Contracts and Decentralized Applications: Ethereum Architecture, Ethereum Virtual Machine (EVM), Accounts and Transactions, Gas Mechanism, Smart Contracts, Solidity Fundamentals, Smart Contract Deployment, Decentralized Applications (DApps).		
SELF-STUDY COMPONENT	**10 Hours**	
Decentralized Autonomous Organizations (DAOs), Token Standards.		
UNIT 4	**8**	**L1-L4**
Enterprise Blockchain Platforms: Enterprise Blockchain Concepts, Hyperledger Project, Hyperledger Fabric Architecture, Membership Service Provider, Channels and Chaincode, Hyperledger Sawtooth.		
SELF-STUDY COMPONENT	**10 Hours**	
Corda Architecture, Quorum Platform, Enterprise Use Cases.		
UNIT 5	**8**	**L1-L4**
Security, Privacy and Emerging Trends: Blockchain Security, 51% Attacks, Sybil Attacks, Eclipse Attacks, Privacy in Blockchain, Zero-Knowledge Proofs, Ring Signatures, Confidential Transactions, Decentralized Finance (DeFi)		
SELF-STUDY COMPONENT	**10 Hours**	
Non-Fungible Tokens (NFTs), Web3 Ecosystem, Blockchain and IoT.		

Practical Component For Advanced Blockchain Technologies
1. Implement a Basic Blockchain. Create a Python program that implements a simple blockchain with at least three blocks. Each block must contain an index, timestamp, transaction data, hash of the previous block, nonce, and its own SHA-256 hash. Demonstrate that changing data in one block invalidates all subsequent blocks, illustrating blockchain immutability. 2. Build a Merkle Tree for Transaction Verification. Write a program to construct a Merkle tree from at least eight sample transactions using SHA-256 hashing. Compute and display the Merkle root, then modify one transaction and show how the root changes. Explain how Merkle roots enable efficient transaction verification without downloading the

entire block.

3. Token Transfer and Balance Verification

Develop and deploy an ERC-20 token contract in Remix IDE. Mint an initial supply of 1,000,000 tokens to the deployer's address, then transfer 500 tokens to a second address. Verify both balances using the balanceOf function and record the gas consumed for the transfer transaction ethereum+2

4. Asset Transfer Chaincode Deployment

Set up the Hyperledger Fabric test network using network.sh up createChannel, then deploy the asset-transfer-basic chaincode (Go or TypeScript) to the channel. Create a new asset, query it by ID, transfer ownership to another organization, and verify the updated ownership via a query.hyperledger-fabric.readthedocs+2

TEXT BOOKS:	
1	1. Imran Bashir, Mastering Blockchain, 4th Edition, Packt Publishing, 2023.

REFERENCE BOOKS:	
1	Andreas M. Antonopoulos and Gavin Wood, Mastering Ethereum: Building Smart Contracts and DApps, 1st Edition, O'Reilly Media, Inc., USA, 2018. ISBN: 978-1491971949
2	Daniel Drescher, Blockchain Basics: A Non-Technical Introduction in 25 Steps, 2nd Edition, Apress, Berkeley, CA, USA, 2020. ISBN: 978-1484254127.
3	Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller and Steven Goldfeder, Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction, 1st Edition, Princeton University Press, Princeton, New Jersey, USA, 2016. ISBN: 978-0691171692.
e-LEARNING RESOURCES:	
1	**NPTEL – Blockchain and Its Applications:** https://onlinecourses.nptel.ac.in/
2	**Ethereum Developer Documentation** – Ethereum architecture, EVM, smart contracts, Solidity, gas mechanism, and DApps: https://ethereum.org/en/developers/docs/

Assessment Details both (CIE and SEE)				
CONTINUOUS INTERNAL EVALUATION (CIE)		**Max. Marks**		Minimum Marks to be scored in CIE, to qualify to take SEE (50% individually)
Theory	**Weightage of CIE1 and CIE2 Tests or CIE3**	**30**		**15**
Laboratory	Rubrics for experiment are added on weekly basis	**10**	**20**	**10**

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

component s	and then taken average			
	Lab CIE	**10**		
	TOTAL	**50**		**25**

QUESTION PAPER PATTERN (SEE)										
Q. No.	**Q1**	**Q2**	**Q3**	**Q4**	**Q5**	**Q6**	**Q7**	**Q8**	**Q9**	**Q10**
UNIT	1		2		3		4		5	
1. Two full questions (each of 20 Marks) are to be set from each unit.										
2. Student shall answer five full questions selecting one full question from each unit.										

Course Outcomes: After the completion of the course, the student will be able to:	
CO1	Analyze blockchain architecture, decentralization principles, and cryptographic Foundations.
CO2	Evaluate advanced consensus protocols and blockchain scalability solutions.
CO3	Implement smart contracts and decentralized applications using Ethereum.
CO4	Compare enterprise blockchain frameworks and their application domains.
CO5	Assess blockchain security, privacy-preserving mechanisms, and emerging decentralized technologies.

COURSE SUSTAINABLE DEVELOPMENT GOALS (SDGs)		
Goal No.	**Goal Title**	**Description**
4	Quality Education	Ensure Inclusive and equitable quality education and promote lifelong learning opportunities for all
8	Decent Work and Economic Growth	Promote sustained, inclusive and sustainable economic growth, full and productive employment and decent work for all.
9	Industry, Innovation and Infrastructure	Build resilient infrastructure, promote inclusive and sustainable industrialization and foster innovation.

Mapping of COs with POs

Strength of correlation: Strongly related-3, Moderately related-2, Weakly related-1, Not related-0

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**
CO1	3	1	3	3	2
CO2	3	3	2	0	0
CO3	3	3	3	0	0

CO4	3	3	3	0	0
CO5	3	3	3	0	0

Course Coordinators:

Dr. Shamshekhar S. Patil Associate Professor CSE Programme

HEAD of School (HOS)
School of Computer Science and Engineering
Dr. Ambedkar Institute of Technology
Bengaluru-560 056

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

Course Title	Advanced Computer Networks					
Course Code	1MSCSU104B					
Course Category	IPCC (Integrated Professional Core Course)					
Teaching - Learning Scheme & Credit	L	T	P	Total credits	Total Teaching Hours	Credit
	3	0	1	4	Lecture: 42 + Tutorials: 0 + Practical: 28 + Self Study: 50 = 120	4
CIE Marks: 50	SEE Marks: 50		Total Max. Marks: 100		Duration of SEE: 03 Hrs.	

Course Learning Objectives:	
CLO1	Apply link-layer protocols, error detection and correction techniques, multiple access protocols, and data centre networking architectures to ensure reliable and efficient data communication
CLO2	Select wireless and mobile networking technologies for modern communication systems.
CLO3	Apply multimedia networking protocols to design quality of service aware real-time multimedia communication systems.
CLO4	Utilize network virtualization technologies on multiple operating systems enforcing security enhancements.
CLO5	Apply Voice and Video over IP protocols to ensure Quality of Service (QoS) based traffic management.

SYLLABUS	No. of Hours	BTLs
UNIT 1	**8**	**L1-L4**
The Link Layer: Links, Access Networks and LANs: Introduction to the Link Layer, The Services Provided by the Link Layer, Where Is the Link Layer Implemented? Error-Detection and -Correction Techniques: Parity Checks, Check summing Methods, Cyclic Redundancy Check (CRC), Multiple Access Links and Protocols: Channel Partitioning Protocols, Random Access Protocols, Taking-Turns Protocols, DOCSIS: The Link-Layer Protocol for Cable Internet Access, Data Center Networking, Retrospective: A Day in the Life of a Web Page Request: Getting Started: DHCP, UDP, IP, and Ethernet, Still Getting Started: DNS and ARP, Still Getting Started: Intra-Domain Routing to the DNS Server, Web Client-Server Interaction: TCP and HTTP.		
SELF-STUDY COMPONENT	**10 Hours**	
Numerical exercises on parity, checksum, and CRC., A flowchart illustrating the complete journey of a web page request.		
UNIT 2	**9**	**L1-L4**

Wireless and Mobile Networks: Introduction, Wireless Links and Network Characteristics, CDMA, Cellular Internet Access: An Overview of Cellular Network Architecture, 3G Cellular Data Networks: Extending the Internet to Cellular Subscribers, On to 4G: LTE, Mobility Management: Principles: Addressing, Routing to a Mobile Node, Mobile IP: Managing Mobility in Cellular Networks: Routing Calls to a Mobile User, Handoffs in GSM		
SELF-STUDY COMPONENT	**10 Hours**	
Different multimedia applications and classify them into streaming, interactive, and conversational applications. Video and audio characteristics, including bit rate, compression, delay sensitivity, and quality requirements. Analyze a real-world application such as YouTube, Netflix, or a video-conferencing platform, identifying its multimedia networking requirements.		
UNIT 3	**9**	**L1-L4**
Multimedia Networking: Multimedia Networking Applications, Properties of Video, Properties of Audio, Types of Multimedia Network Applications, Streaming Stored Video: UDP Streaming, HTTP Streaming, Adaptive Streaming and DASH, Content Distribution Networks, Protocols for Real-Time Conversational Applications: RTP, SIP		
SELF-STUDY COMPONENT	**10 Hours**	
Preparation of comparison charts, CDN architecture diagrams, short notes on RTP and SIP, and a case study of a multimedia streaming or video-conferencing application.		
UNIT 4	**8**	**L1-L4**
Network Virtualization: VPNs, NATs, And Overlays: Introduction, Virtualization, Virtual Private Networks (VPNs), VPN Tunnelling And IP-in-IP Encapsulation, VPN Addressing And Forwarding, Extending VPN Technology To Individual Hosts, Using A VPN With Private IP Addresses, Network Address Translation (NAT), NAT Translation Table Creation, Variant Of NAT, An Example Of NAT Translation, Interaction Between NAT And ICMP, Interaction Between NAT And Application.		
SELF-STUDY COMPONENT	**10 Hours**	
Draw and explain VPN tunneling and IP-in-IP encapsulation. Solve numerical or packet-flow exercises involving NAT translation tables. Prepare a comparison chart of VPNs and NAT.		
UNIT 5	**8**	**L1-L4**
Voice and Video Over IP: Introduction, Digitizing and Encoding, Audio and Video Transmission and Reproduction, Jitter And Playback Delay, Real-time Transport Protocol (RTP), Streams, Mixing, and Multicasting, RTP Encapsulation, RTP Control Protocol (RTCP), RTCP Operation, IP Telephony and Signaling, Quality of Service Controversy, QoS, Utilization, and Capacity, Emergency Services and Pre-emption.		
SELF-STUDY COMPONENT	**10 Hours**	
Illustrate traffic shaping and policing using suitable examples., Present a case study of QoS in an IP telephony or video-conferencing application		

Practical Component for Advances in Computer Networks
List of experiments
1. Implement Parity Check, Checksum, and Cyclic Redundancy Check (CRC) algorithms for error detection and compare their effectiveness. (**Software Tool:** C/C++ / Java / Python)
2. Capture and analyse Ethernet, ARP, IP, TCP, UDP, DNS, and HTTP packets using Wireshark. (**Software Tool:** Wireshark / Any other tools)
3. Configure an IEEE 802.11 Wireless LAN and evaluate throughput, delay, and packet loss under different traffic loads. (**Software Tool:** Cisco Packet Tracer / NS-3 / Any other tools)
4. Configure and analyze Bluetooth and ZigBee communication for short-range wireless networking. (**Software Tool:** Cisco Packet Tracer / NS-3 / Any other tools)
5. Configure VPN and NAT in a virtual network and study packet forwarding and address translation. (**Software Tool:** Cisco Packet Tracer / GNS3 / Linux / Any other tools)

TEXT BOOKS:	
1	James F. Kurose and Keith W. Ross, *Computer Networking: A Top-Down Approach*, 6th Edition, Pearson, 2021.
2	Douglas E. Comer, *Internetworking with TCP/IP, Volume I: Principles, Protocols, and Architecture*, 6th Edition, Pearson Education, 2014.

REFERENCE BOOKS:	
1	Andrew S. Tanenbaum, Nick Feamster, and David J. Wetherall, Computer Networks, 6th Edition, Pearson Education, 2021.
2	Network Warrior – by Gary A. Donahue, Practical guide for real-world network administration.
3	Larry Peterson and Bruce S Davis "Computer Networks :A System Approach" 5th Edition, Elsevier.

e-LEARNING RESOURCES:

Web links and Video Lectures (e-Resources):

https://nptel.ac.in/courses/106106243

https://onlinecourses.nptel.ac.in/e-learning/preview/noc26_cs35

https://www-net.cs.umass.edu/kurose_ross/index.php

http://www.wireshark.org/docs/wsug_html_chunked/

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

Assessment Details both (CIE and SEE)				
CONTINUOUS INTERNAL EVALUATION (CIE)		**Max. Marks**		Minimum Marks to be scored in CIE, to qualify to take SEE (50% individually)
Theory	**Weightage of CIE1 and CIE2 Tests or CIE3**	**30**		**15**
Laboratory components	Rubrics for experiment are added, then taken average	**10**	**20**	**10**
	Lab CIE	**10**		
	TOTAL	**50**		**25**

QUESTION PAPER PATTERN (SEE)										
Q. No.	**Q1**	**Q2**	**Q3**	**Q4**	**Q5**	**Q6**	**Q7**	**Q8**	**Q9**	**Q10**
UNIT	1		2		3		4		5	
1. Two full questions (each of 20 Marks) are to be set from each unit.										
2. Student shall answer five full questions selecting one full question from each unit.										

Course Outcomes: After the completion of the course, the student will be able to:	
CO1	Apply link-layer protocols, error detection and correction techniques, multiple access protocols, and data centre networking architectures to ensure reliable and efficient data communication
CO2	Select wireless and mobile networking technologies for modern communication systems.
CO3	Apply multimedia networking protocols to design quality of service aware real-time multimedia communication systems.
CO4	Utilize network virtualization technologies on multiple operating systems enforcing security enhancements.
CO5	Apply Voice and Video over IP protocols to ensure Quality of Service (QoS) based traffic management

COURSE SUSTAINABLE DEVELOPMENT GOALS (SDGs)		
Goal No.	**Goal Title**	**Description**
4	Quality Education	Ensure Inclusive and equitable quality education and promote lifelong learning opportunities for all
8	Decent Work and Economic Growth	Promote sustained, inclusive and sustainable economic growth, full and productive employment and decent work for all.

9	Industry, Innovation and Infrastructure	Build resilient infrastructure, promote inclusive and sustainable industrialization and foster innovation.

Mapping of COs with POs

Strength of correlation: Strongly related-3, Moderately related-2, Weakly related-1, Not related-0

COs	PO1	PO2	PO3	PO4	PO5
CO1	3	21	1	0	1
CO2	3	3	3	1	3
CO3	3	3	2	2	3
CO4	3	3	2	1	3
CO5	3	3	3	2	3

Course Coordinators:

Dr. Shamshekhar S Patil
Associate Professor
CSE Programme

HEAD of School (HOS)
School of Computer Science and Engineering
Dr. Ambedkar Institute of Technology
Bengaluru-560 056

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

Course Title	**Advanced Bigdata Analytics**					
Course Code	**1MSCSU104C**					
Course Category	**IPCC (Integrated Professional Core Course)**					
Teaching - Learning Scheme & Credit	**L**	**T**	**P**	**Total**	**Total Teaching Hours**	**Credit**
	3	**0**	**1**	**4**	**Lecture: 42 + Tutorials: 0 + Practical: 28 + Self Study: 50 = 120**	4
CIE Marks: 50	**SEE Marks: 50**		**Total Max. Marks: 100**		**Duration of SEE: 03 Hrs.**	

Course Learning Objectives:	
CLO1	Describe the need of Big data and visualization in real world applications.
CLO2	Apply big data analytics frameworks and visualization techniques for decision making
CLO3	Analyse the use of big data analytics and visualization for business applications
CLO4	Formulate optimized relational and scripting-based queries using Apache Hive and Pig to handle large-scale data ETL and reporting operations

SYLLABUS	**No. of Hours**	**RTLs**
UNIT 1	**8**	**L1-L4**
Introduction to Big Data Analytics: Definition, Need of Big Data and its characteristics, Classification of Data - Structured, semi structured, and unstructured, Scalability and Parallel Processing, Designing Data Architecture, Data Sources, Quality, Pre-Processing and Storing, Data Storage and Analysis, Big Data Analytics Applications and Case Studies.		
SELF-STUDY COMPONENT	**10 Hours**	
Investigate real-world applications of Big Data Analytics in healthcare, banking, education, retail, and social media. Analyze case studies to understand how Big Data Analytics supports decision-making and solves real-world problems.		
UNIT 2	**8**	**L1-L4**
Introduction to Hadoop: Introduction, Hadoop and its Ecosystem, Hadoop Distributed File System, MapReduce Framework and Programming Model, Hadoop Yarn, Hadoop Ecosystem Tools, Hadoop Distributed File System Basics: HDFS Design Features, Components, HDFS User Commands.		
SELF-STUDY COMPONENT	**10 Hours**	

Understand the role of YARN in resource management and job scheduling in Hadoop. Investigate Hadoop ecosystem tools such as Hive, Pig, HBase, Sqoop, and Flume. Practice basic HDFS user commands for creating directories, uploading, downloading, listing, and deleting files.		
UNIT 3	**9**	**L1-L4**
MapReduce, Hive and Pig: Introduction, MapReduce Map Tasks, Reduce Tasks and MapReduce Execution, Composing MapReduce for Calculations and Algorithms, Hive, HiveQL, Pig		
SELF-STUDY COMPONENT	**10 Hours**	
Explore the MapReduce framework, programming model, and execution phases using simple examples. Practice writing HiveQL queries to analyze large datasets. Study Apache Pig, its architecture, features, and applications in data processing.		
UNIT 4	**9**	**L1-L4**
Spark and Big Data Analytics: Introduction to Data Analysis with Spark-Spark SQL, Using Python with Spark SQL, Data Analysis Operations, Data ETL process, Introduction Analytics, Reporting and Visualization.		
SELF-STUDY COMPONENT	**10 Hours**	
Study the ETL (Extract, Transform, Load) process and its importance in Big Data processing. Practice extracting data from different sources, transforming datasets, and loading processed data into storage systems.		
UNIT 5	**8**	**L1-L4**
Machine Learning Algorithms for Big Data Analytics: Introduction, Estimating the relationships, Outliers, Variances, Probability Distributions, and Correlations, Regression analysis, Finding Similar Items, Similarity of Sets and Collaborative Filtering, Clustering Analysis.		
SELF-STUDY COMPONENT	**10 Hours**	
Study collaborative filtering techniques used in recommendation systems ,regression analysis techniques for predicting outcomes using large datasets		

PRACTICAL COMPONENTS OF IPCC
List of experiments
1. Given a raw, noisy dataset containing mixed formats (structured, semi-structured JSON, and unstructured text logs), write a script to handle missing values, filter outliers, detect data duplication, and normalize data types to prepare a clean schema for storage.

3. Design and implement a MapReduce program to process a large web-server access log file. The Mapper should parse the IP addresses and timestamps, and the Reducer should compute the hourly hit counts for each unique IP address.
4. Write a MapReduce program that executes a Reduce-Side Join or Map-Side Join on two separate datasets (e.g., a "Users" dataset and a "Transaction Logs" dataset) sharing a common key, simulating a relational database join operation over a distributed network.
5. Create an optimization pipeline in Apache Hive. Write a HiveQL script to create external tables for an e-commerce sales dataset. Implement dynamic partitioning and bucketing based on "Region" and "Year", and write analytical queries to extract top performing product categories.
6. Write an Apache Pig script (.pig) to load, filter, and transform an unstructured or semi-structured dataset (like a Twitter/X user mention dataset). Use Pig Latin operators (FILTER, GROUP, FOREACH, GENERATE, and ORDER BY) to isolate and count the most trending keywords.
7. Execute interactive SQL queries on distributed Spark clusters. Create temporary and global views from Spark Data Frames, run window functions to find rolling averages or rank records, and pass the final aggregate outputs to a Python visualization library (like Matplotlib or Seaborn) to generate analytics dashboards.
8. Design and implement an end-to-end ETL pipeline using Apache Spark and PySpark for an available dataset. Read data from CSV/JSON files, perform data cleaning and transformation operations, handle missing values and duplicates, apply aggregations and filtering, and store the processed data in an appropriate format. Create Spark SQL views and execute analytical queries to generate meaningful insights.
9. Implement a regression-based machine learning model using Apache Spark MLlib on available dataset. Perform data pre-processing and feature selection, divide the dataset into training and testing sets, build a Linear Regression model, evaluate its performance using suitable metrics such as MAE, RMSE and R^2, and interpret the results.
10. Implement a clustering and similarity analysis application using Apache Spark MLlib on available dataset. Apply K-Means clustering to group similar records, determine suitable clusters using an appropriate evaluation measure.

TEXT BOOKS:	
1	Raj Kamal and Preeti Saxena, "Big Data Analytics Introduction to Hadoop, Spark, and Machine-Learning", McGraw Hill Education, 2018 ISBN: 9789353164966, 9353164966.
2	Douglas Eadline, "Hadoop 2 Quick-Start Guide: Learn the Essentials of Big Data Computing in the Apache Hadoop 2 Ecosystem", 1st Edition, Pearson Education, 2016. ISBN-13: 978-9332570351

REFERENCE BOOKS:	
1	Tom White, "Hadoop: The Definitive Guide", 4th Edition, O"Reilly Media, 2015.ISBN 13: 978-

	9352130672.
2	Seema Acharya, Subhashini Chellappan, Big Data and analytics, Wiley Publications, 2015, ISBN-10: 8126554789, ISBN-13: 978-8126554782.
e-LEARNING RESOURCES:	
1	Essential for mastering HDFS commands, YARN configurations, and core MapReduce programming models: https://hadoop.apache.org/docs/stable/
2	Architecture scalability, Hadoop ecosystem tools, Pig, and Hive: https://nptel.ac.in/courses/106105186

Assessment Details both (CIE and SEE)

CONTINUOUS INTERNAL EVALUATION (CIE)		**Max. Marks**		Minimum Marks to be scored in CIE, to qualify to take SEE (50% individually)
Theory	**Weightage of CIE1 and CIE2 Tests or CIE3**	**30**		**15**
Laboratory components	Rubrics for experiment are added, then taken average	**10**	**20**	**10**
	Lab CIE	**10**		
	TOTAL	**50**		**25**

QUESTION PAPER PATTERN (SEE)

Q. No.	**Q1**	**Q2**	**Q3**	**Q4**	**Q5**	**Q6**	**Q7**	**Q8**	**Q9**	**Q10**
UNIT	1		2		3		4		5	
1. Two full questions (each of 20 Marks) are to be set from each unit.										
2. Student shall answer five full questions selecting one full question from each unit.										

Course Outcomes: After the completion of the course, the student will be able to:	
CO1	Describe the need of Big data and visualization in real world applications.
CO2	Apply big data analytics frameworks and visualization techniques for decision making
CO3	Analyse the use of big data analytics and visualization for business applications
CO4	Formulate optimized relational and scripting-based queries using Apache Hive and Pig to handle large-scale data ETL and reporting operations

4	Quality Education	Ensure Inclusive and equitable quality education and promote lifelong learning opportunities for all
8	Decent Work and Economic Growth	Promote sustained, inclusive and sustainable economic growth, full and productive employment and decent work for all.
9	Industry, Innovation and Infrastructure	Build resilient infrastructure, promote inclusive and sustainable industrialization and foster innovation.

Mapping of COs with POs

Strength of correlation: Strongly related-3, Moderately related-2, Weakly related-1, Not related-0

COs	PO1	PO2	PO3	PO4	PO5
CO1	3	2	1	0	1
CO2	3	3	3	1	3
CO3	3	3	2	2	3
CO4	3	3	2	1	3
CO5	3	3	3	2	3

Course Coordinators:

For 27/10/2026

Dr. Nandini N
Professor and HOP
CSE Programme

HEAD of School (HOS)
School of Computer Science and Engineering
Dr. Ambedkar Institute of Technology
Bengaluru-560 056

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

Course Title	**Deep Learning**					
Course Code	**1MSCSU104D**					
Course Category	**IPCC (Integrated Professional Core Course)**					
Teaching - Learning Scheme & Credit	**L**	**T**	**P**	**Total**	**Total Teaching Hours**	**Credit**
	3	**0**	**1**	**4**	**Lecture: 42 + Tutorials: 0 + Practical: 28 + Self Study: 50 = 120**	**4**
CIE Marks: 50	**SEE Marks: 50**		**Total Max. Marks: 100**		**Duration of SEE: 03 Hrs.**	

Course Learning Objectives:	
CLO1	To understand the basic concepts and techniques of Deep Learning and the need of Deep Learning techniques in real-world problems
CLO2	To understand CNN algorithms and the way to evaluate performance of the CNN architectures.
CLO3	To apply RNN and LSTM to learn, predict and classify the real-world problems in the paradigms of Deep Learning.
CLO4	To understand, learn and design GANs for the selected problems.
CLO5	To understand the concept of Auto-encoders and enhancing GANs using auto-encoders

SYLLABUS	**No. of Hours**	**BTL s**
UNIT 1	**8**	**L1-L4**
INTRODUCTION TO DEEP LEARNING:Historical Trends in Deep Learning,The Many Names and Changing Fortunes of Neural Networks, Deep Feed- forward networks Example: Learning XOR. Gradient Based learning, Hidden Units, Architecture Design.		
SELF-STUDY COMPONENT	10	Hours
Back-Propagation and Other Differentiation Algorithms , Computational Graphs, Chain Rule of Calculus, Recursively Applying the Chain Rule to Obtain Backprop, Back-Propagation Computation in Fully-Connected MLP, General Back-Propagation.		
UNIT 2	**8**	**L1-L4**
CONVOLUTION NEURAL NETWORK (CNN): The Convolution Operation, Motivation, Pooling, Convolution and Pooling as an Infinitely Strong Prior.		
SELF-STUDY COMPONENT	10	Hours
Variants of the Basic Convolution Function, Structured Outputs, Efficient Convolution Algorithms		

UNIT 3	9	L1-L4
Sequence Modeling: Recurrent and Recursive Nets: Recurrent Neural Networks, Teacher Forcing and Networks with Output Recurrence, Computing the Gradient in a Recurrent Neural Network, Recurrent Networks as Directed Graphical Models, Modeling Sequences Conditioned on Context with RNNs, Bidirectional RNNs		
SELF-STUDY COMPONENT	**10 Hours**	
Encoder-Decoder Sequence-to-Sequence Architectures, Deep Recurrent Networks, Recursive Neural Networks.		
UNIT 4	**9**	**L1-L4**
Deep Generative Models: Boltzmann Machines, Restricted Boltzmann Machines, Deep Belief Networks, Deep Boltzmann Machines.		
SELF-STUDY COMPONENT	**10 Hours**	
Boltzmann Machines for Real-Valued Data: Gaussian-Bernoulli RBMs, Undirected Models of Conditional Covariance, Convolutional Boltzmann Machines.		
UNIT 5	**8**	**L1-L4**
AUTO-ENCODERS: Undercomplete Autoencoders, Regularized Autoencoders, Representational Power, Layer Size and Depth, Stochastic Encoders and Decoders, Denoising Autoencoders		
SELF-STUDY COMPONENT	**10 Hours**	
Learning Manifolds with Autoencoders, Contractive Autoencoders, Applications of Autoencoders		

Practical Component
1. Implementation and Analysis of XOR using a Multi-Layer Perceptron
2. Design and implement a CNN model (with 2 layers of convolutions) to classify multi category image datasets. Record the accuracy corresponding to the number of epochs. Use the MNIST, CIFAR-10 datasets.
3. Implement the standard VGG CNN architecture model to classify the multi-category MNIST image dataset and evaluate its accuracy.
4. Implement a Recurrent Neural Network (RNN) for Sequence Prediction using Python and TensorFlow/Keras
5. Design and Implement GAN for Generating Synthetic Handwritten Digits
6. Implement an Autoencoder for Image Compression and Reconstruction using MNIST Dataset

TEXT BOOKS:

1	Ian Goodfellow and Yoshua Bengio Aaron Courville, Deep Learning: An MIT Press Book
2	Michael Nielson, Neural Networks and Deep Learning, Determination Press, 2015.
3	Satish kumar, Neural networks: A classroom Approach, Tata McGraw-Hill Education, 2004.

REFERENCE BOOKS:	
1	Deep Learning with Python, Francois Chollet, Manning publications, 2018.
2	Advanced Deep Learning with Keras, Rowel Atienza, PACKT Publications, 2018
e-LEARNING RESOURCES:	
1	https://nptel.ac.in/courses/106106184
2	https://cs231n.stanford.edu/

Assessment Details both (CIE and SEE)

CONTINUOUS INTERNAL EVALUATION (CIE)		**Max. Marks**		Minimum Marks to be scored in CIE, to qualify to take SEE (50% individually)
Theory	**Weightage of CIE1 and CIE2 Tests or CIE3**	**30**		15
Laboratory components	Rubrics for experiment are added, then taken average	**10**	**20**	10
	Lab CIE	**10**		
	TOTAL	**50**		25

QUESTION PAPER PATTERN (SEE)

Q. No.	**Q1**	**Q2**	**Q3**	**Q4**	**Q5**	**Q6**	**Q7**	**Q8**	**Q9**	**Q10**
UNIT	1		2		3		4		5	

1. Two full questions (each of 20 Marks) are to be set from each unit.
2. Student shall answer five full questions selecting one full question from each unit.

Course Outcomes: After the completion of the course, the student will be able to:	
CO1	Understand the architecture and training of deep Neural Networks.
CO2	Design and implement CNN architectures for image related tasks.
CO3	Apply RNN, LSTM and GRU models for sequential data processing.
CO4	Develop GANs for data generation tasks.
CO5	Utilize autoencoders for feature extraction and dimensionality reduction.

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

COURSE SUSTAINABLE DEVELOPMENT GOALS (SDGs)		
Goal No.	**Goal Title**	**Description**
4	Quality Education	Ensure Inclusive and equitable quality education and promote lifelong learning opportunities for all
8	Decent Work and Economic Growth	Promote sustained, inclusive and sustainable economic growth, full and productive employment and decent work for all.
9	Industry, Innovation and Infrastructure	Build resilient infrastructure, promote inclusive and sustainable industrialization and foster innovation.

Mapping of COs with POs with PSOs

Strength of correlation: Strongly related-3, Moderately related-2, Weakly related-1, Not related-0

COs	PO1	PO2	PO3	PO4	PO5
CO1	3	2	1	0	1
CO2	3	3	3	1	3
CO3	3	3	2	2	3
CO4	3	3	2	1	3
CO5	3	3	3	2	3

Course Coordinators:

2/Feb/26
Dr. Madhu B Associate Professor CSE Programme

HEAD of School (HOS)
School of Computer Science and Engineering
Dr. Ambedkar Institute of Technology
Bengaluru-560 056

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering
Computer Science and Engineering Programme
M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2025**

Course Title	**Data Analytics for Fraud Detection**				
Course Code	**1MSCS105A**				
Course Category	**PEC (Professional Elective Course-I)**				
Teaching - Learning Scheme & Credit	**L**	**T**	**P**	**Total Credits**	**Total Teaching Hours**
	3	**0**	**0**	**3**	**Lecture: 42 + Tutorials: 0 + Practical: 0 + Self Study: 48 = 90**
CIE Marks: 50	**SEE Marks: 50**		**Total Max. Marks: 100**		**Duration of SEE: 03 Hours**

Course Learning Objectives:	
CLO1	Explain the concepts, types, and challenges of fraud detection using data analytics.
CLO2	Apply data preprocessing, visualization, and analytical techniques for fraud analysis.
CLO3	Develop fraud detection models using statistical and machine learning techniques.
CLO4	Evaluate fraud detection solutions using predictive analytics, graph analytics, and real-world case studies.

SYLLABUS	**No. of Hours**	**RBTs**
UNIT 1	**8**	L1-L3
Introduction to Fraud Analytics: Fundamentals and characteristics of fraud, types of fraud, fraud detection and prevention, role of big data in fraud detection, data-driven fraud detection, fraud-detection techniques, fraud cycle, **fraud** analytics process model, role of fraud data scientists, data-analytic thinking, business problems and data science solutions, supervised and unsupervised analytical approaches, and overview of the data mining process.		
SELF-STUDY COMPONENT	10 Hours	
Comparative case study of two real-world fraud scenarios, examining the data sources, detection approaches, analytical methods and preventive measures used.		
UNIT 2	**8**	L1-L4
Data Collection, Sampling and Preprocessing: Types of data sources, merging data sources, sampling, types of data elements, visual data exploration and exploratory statistical analysis, Benford's Law, **descriptive** statistics, treatment of missing values and outliers, identification of red flags, data standardization and categorization, weights of evidence coding, variable selection, Principal Component Analysis, RIDIT and PRIDIT analysis, and segmentation.		
SELF-STUDY COMPONENT	10 Hours	
Independent analysis of a publicly available fraud dataset involving data quality assessment, sampling,		

descriptive statistics, visual exploration, identification of red flags, treatment of missing values and outliers, and **feature** transformation. Study of dimensionality reduction and fraud-risk scoring techniques.		
UNIT 3	**9**	**L1-L4**
Descriptive Analytics and Outlier Detection: Descriptive analytics for fraud detection, fundamentals of anomaly and outlier detection, graphical and statistical outlier detection procedures, probabilistic and statistical models for outlier detection, linear models, proximity-based outlier detection, clustering techniques for **fraud** detection, high dimensional outlier detection, and one-class Support Vector Machines for identifying anomalous observations.		
SELF-STUDY COMPONENT	**10 Hours**	
Comparative evaluation of two selected anomaly-detection methods using a suitable dataset, with emphasis on detection performance, false alarms and the identification of anomalous transactions.		
UNIT 4	**9**	**L1-L4**
Predictive Analytics for Fraud Detection: Fundamentals of predictive modelling, target definition, supervised learning, logistic regression, variable selection, decision trees, neural networks, Support Vector Machines, ensemble methods, multiclass classification, model generalization and overfitting, model validation, evaluation of classification models, handling skewed and imbalanced datasets, and performance evaluation of fraud detection models.		
SELF-STUDY COMPONENT	**9 Hours**	
Comparative evaluation of selected classification models using precision, recall, F1-score, precision–recall analysis and confusion matrices, with emphasis on false positives and false negatives in fraud detection for a publicly available dataset.		
UNIT 5	**8**	**L1-L4**
Social Network Analytics, Fraud Model Management and Governance: Social networks and their application to fraud detection, fraud as a social phenomenon, homophily, neighbourhood-based metrics, community mining for identifying groups of fraudsters, bipartite networks, fraud model lifecycle, model representation, selection of cases for investigation, fraud alert and case management, visual analytics, backtesting analytical fraud models, model design and documentation, data quality, privacy in fraud analytics, economic considerations, modelling extensions, Internet of Things in fraud analytics, and corporate fraud governance.		
SELF-STUDY COMPONENT	**9 Hours**	
Case-based study of social network analytics, neighbourhood-based metrics, community detection and bipartite networks for identifying potential fraud networks.		

TEXT BOOKS

1. Bart Baesens, Véronique Van Vlasselaer and Wouter Verbeke, Fraud Analytics Using Descriptive, Predictive, and Social Network Techniques: A Guide to Data Science for Fraud Detection, John

	Detection, John Wiley & Sons, 1st Edition, 2015.
2.	Foster Provost and Tom Fawcett, Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking, O'Reilly Media, 2013.ISBN 9781449374273.
3.	Charu C. Aggarwal, Outlier Analysis, 2nd Edition, Springer, 2017

REFERENCE BOOKS	
1.	Jiawei Han, Micheline Kamber, and Jian Pei, Data Mining: Concepts and Techniques, 4th Edition, Morgan Kaufmann, 2023.
2.	Trevor Hastie, Robert Tibshirani, and Jerome Friedman, The Elements of Statistical Learning, 2nd Edition (Corrected), Springer.
3.	Ian H. Witten, Eibe Frank, Mark A. Hall, and Christopher J. Pal, Data Mining: Practical Machine Learning Tools and Techniques, 4th Edition, Morgan Kaufmann, 2016.
4.	Christopher M. Bishop and Hugh Bishop, Deep Learning: Foundations and Concepts, Springer, 2024.
5.	Kevin Murphy, Probabilistic Machine Learning: An Introduction, MIT Press, 2022

e-LEARNING RESOURCES

- Kaggle Datasets (Fraud Detection) – Public datasets for credit card fraud, insurance fraud, transaction analytics, and machine learning projects.
- Scikit-learn Documentation – Documentation and tutorials for machine learning algorithms used in fraud detection.
- Scikit-learn MOOC (Inria) – Free hands-on course covering data preprocessing, classification, anomaly detection, and model evaluation.
- TensorFlow Tutorials – Official tutorials for deep learning models applicable to fraud analytics.
- FATF (Financial Action Task Force) – International standards, reports, and guidance on anti-money laundering (AML) and financial crime.
- WEKA Documentation – Machine learning toolkit useful for experimentation with fraud detection datasets.
- Python Data Science Handbook – Free online reference for NumPy, Pandas, Matplotlib, and Scikit-learn.
- Pandas Documentation – Official documentation for data preprocessing and analytics.
- NumPy Documentation – Numerical computing library used in data analytics.
- Matplotlib Documentation – Visualization techniques for exploratory fraud analysis.
- Video Lectures
- NPTEL – Data Analytics and Machine Learning Courses
- MIT OpenCourseWare – Data Science and Machine Learning

- Stanford Online – Machine Learning Resources
- DeepLearning.AI Machine Learning Specialization (Coursera) – Covers supervised learning, anomaly detection, model evaluation, and practical ML workflows.
- Google Developers – Machine Learning Crash Course

	Assessment Details for both CIE and SEE
	The weightage assigned to the Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) is 50% each. To pass the course, students must secure a minimum of 40% marks separately in both CIE and SEE. **Continuous Internal Evaluation (CIE)** The CIE consists of four components: 1. CIE – Theory Component 2. Assignment 3. Group Activity 4. Seminar Students must secure a minimum of 50% marks in each component CIE Theory, Assignment, and Group Activity components to qualify for appearing in the SEE. **Semester End Examination (SEE)** The Theory SEE is conducted for 100 marks with a duration of 3 hours. The marks obtained are subsequently scaled down to 50 marks for calculating the final course grade. The student must secure a minimum of 40% marks in the SEE to pass the course. ****Self-Study Component should be included for CIE.**

CONTINUOUS INTERNAL EVALUATION (CIE)		**Max. Marks**	**50%**	**Minimum Marks Required in CIE to Qualify for SEE (50% individually)**
Theory	**Weightage of CIE-1 and CIE-2 or CIE-3**	**30**	**15**	**25**
	Assignment	**10**	**05**	
	Group Activity	**05**	**2.5**	
	Seminar	**05**	**2.5**	
	TOTAL	**50**	**25**	

SEE QUESTION PAPER PATTERN										
Q. No.	**Q1**	**Q2**	**Q3**	**Q4**	**Q5**	**Q6**	**Q7**	**Q8**	**Q9**	**Q10**
UNIT	1		2		3		4		5	

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

1. Two full questions (each of 20 Marks) are to be set from each unit.
2. Students shall answer five full questions selecting one full question from each unit.

Course Outcomes: After the completion of the course, the student will be able to	
CO1	Explain the concepts, types, and challenges of fraud detection using data analytics.
CO2	Apply data preprocessing, visualization, and analytical techniques for fraud analysis.
CO3	Develop fraud detection models using statistical and machine learning techniques.
CO4	Evaluate fraud detection solutions using predictive analytics, graph analytics, and real-world case studies.

COURSE SUSTAINABLE DEVELOPMENT GOALS (SDGs)		
Goal No.	**Goal Title**	**Description**
4	Quality Education	Develops advanced analytical, statistical and machine-learning competencies through fraud analytics and data-driven problem-solving.
8	Decent Work and Economic Growth	Supports reliable financial transactions, fraud risk reduction and responsible use of analytics to strengthen economic activity.
9	Industry, Innovation and Infrastructure	Promotes innovative fraud detection systems using machine learning, anomaly detection, predictive analytics and graph-based methods.

Mapping of COs with POs

Strength of correlation: Strongly related-3, Moderately related-2, Weakly related-1, Not related-0

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**
CO1	3	2	0	1	2
CO2	3	3	2	2	2
CO3	3	3	3	3	2
CO4	3	3	2	3	3

Course Coordinators:

HEAD of School (HOS)
School of Computer Science and Engineering
Dr. Ambedkar Institute of Technology
Bengaluru-560 056

Dr. Veena A
Associate Professor and HOP
CSE (DS) Programme

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

Course Title	Data Engineering				
Course Code	1MSCS105B				
Course Category	PEC (Professional Elective Course-I)				
Teaching - Learning Scheme & Credit	L	T	P	Total Credits	Total Teaching Hours
	3	0	0	3	Lecture: 42 + Tutorials: 0 + Practical: 0 + Self Study: 48 = 90
CIE Marks: 50	SEE Marks: 50		Total Max. Marks: 100		Duration of SEE: 03 Hours

Course Learning Objectives:	
CLO1	Explain the concepts of data engineering, data architecture, data engineering lifecycle, and the roles and responsibilities of data engineers in modern organizations.
CLO2	Apply data engineering principles to select appropriate technologies, architectures, and lifecycle components for developing scalable, reliable, and maintainable data platforms
CLO3	Analyze data generation, storage, ingestion, transformation, and serving techniques to support efficient data processing and analytics requirements.
CLO4	Evaluate data engineering solutions by considering data quality, scalability, security, privacy, governance, technology selection, and operational requirements in modern data ecosystems
CLO5	Evaluate modern data engineering tools, platforms, and emerging technologies to recommend appropriate solutions for diverse data engineering applications and organizational requirements

SYLLABUS	No. of Hours	BTLs
UNIT 1	8	L1-L3
Foundations of Data Engineering and Data Architecture: Data Engineering Defined, The Data Engineering Lifecycle, Evolution of the Data Engineer, Data Engineering and Data Science, Data Engineering Skills and Activities, Data Maturity and the Data Engineer, Business Responsibilities, Technical Responsibilities, Data Engineers Inside an Organization, Internal-Facing versus External-Facing Data Engineers, Data Engineering Lifecycle Components, Designing Good Data Architecture, Principles of Good Data Architecture, Scalability, Security, Loosely Coupled Systems, Reversible Decisions, Architecture Leadership.		
SELF-STUDY COMPONENT	**10 Hours**	
Data **maturity**, business and technical responsibilities, and the role of data engineers within organizations. Understand data architecture principles, including scalability, security, loosely coupled systems, and reversible decisions.		
UNIT 2	8	L1-L4

Data Engineering Lifecycle and Technology Selection: The Data Engineering Lifecycle, Source Systems, Storage, Ingestion, Transformation, Serving Data, Security, Data Management, DataOps, Orchestration, Software Engineering, Choosing Technologies Across the Data Engineering Lifecycle, Evaluating Technology Choices, Build versus Buy Decisions, Cloud and On-Premises Considerations, Total Cost of Ownership, Vendor Selection and Technology Strategy.		
SELF-STUDY COMPONENT	**10 Hours**	
Data engineering lifecycle components, DataOps, orchestration, security, and data management. Explore technology selection, build-versus-buy decisions, cloud and on-premises solutions, and total cost of ownership.		
UNIT 3	**9**	**L1-L4**
Data Generation, Storage, and Ingestion: Data Generation in Source Systems, APIs, Application Databases, OLTP and OLAP Systems, Change Data Capture, Logs, Message Queues and Event Streams, Data Sharing, Storage Fundamentals, File Storage, Block Storage, Object Storage, Distributed Storage Systems, Data Warehouses, Data Lakes, Data Lakehouses, Data Platforms, Data Ingestion, Batch and Stream Ingestion, ETL and ELT, Serialization and Deserialization, Throughput, Reliability, Scalability, Data Migration.		
SELF-STUDY COMPONENT	**10 Hours**	
Study data sources, APIs, databases, OLTP/OLAP systems, logs, and event streams. Explore storage systems, data warehouses, data lakes, lakehouses, and batch/stream ingestion methods. Understand ETL/ELT, serialization, scalability, reliability, and data migration.		
UNIT 4	**9**	**L1-L4**
Queries, Transformation, and Data Serving: Queries, Data Modeling, Normalization and Denormalization, Data Transformation, Data Integration, Data Quality, Data Pipelines, Analytics Engineering, Serving Data for Analytics, Business Analytics, Operational Analytics, Embedded Analytics, Machine Learning Support, Data Products, Data Mesh, Query Federation, Semantic and Metrics Layers, Reverse ETL, Data Sharing and Self-Service Analytics.		
SELF-STUDY COMPONENT	**9**	**Hours**
Explore data pipelines, analytics engineering, business analytics, machine learning support, and data products. Understand data mesh, query federation, semantic layers, reverse ETL, and self-service analytics.		
UNIT 5	**8**	**L1-L4**
Security, Privacy, and Future Trends in Data Engineering: Security Fundamentals, Security Policies, Principle of Least Privilege, Encryption, Logging, Monitoring and Alerting, Network Access, Data Privacy, Governance Considerations, Shared Responsibility in Cloud Environments, Data Backup and Recovery, Emerging Data Engineering Trends, Modern Data Stack, Live Data Stack, Streaming Pipelines, Real-Time Analytics, Data and Machine Learning Integration, Future Roles and Responsibilities of Data Engineers.		

Roles and Responsibilities of Data Engineers.	
SELF-STUDY COMPONENT	**9 Hours**
Study data security, privacy, encryption, access control, monitoring, backup, and recovery practices. Explore data governance, cloud security responsibilities, modern data stacks, streaming pipelines, and real-time analytics. Understand data and machine learning integration and emerging roles of data engineers.	

TEXT BOOKS	
1.	Joe Reis and Matt Housley, Fundamentals of Data Engineering: Plan and Build Robust Data Systems, O'Reilly Media, 2022

REFERENCE BOOKS	
1.	Martin Kleppmann and Chris Riccomini, Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems, 2nd Edition, O'Reilly Media, 2026
2.	James Densmore, Data Pipelines Pocket Reference: Moving and Processing Data for Analytics, 1st Edition, O'Reilly Media, 2021
3.	Tyler Akidau, Slava Chernyak, and Reuven Lax, Streaming Systems: The What, Where, When, and How of Large-Scale Data Processing, 1st Edition, O'Reilly Media, 2018

	Assessment Details for both CIE and SEE
	The weightage assigned to the Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) is 50% each. To pass the course, students must secure a minimum of 40% marks separately in both CIE and SEE. **Continuous Internal Evaluation (CIE)** The CIE consists of four components: 1. CIE – Theory Component 2. Assignment 3. Group Activity 4. Seminar Students must secure a minimum of 50% marks in each component CIE Theory, Assignment, and Group Activity components to qualify for appearing in the SEE. **Semester End Examination (SEE)** The Theory SEE is conducted for 100 marks with a duration of 3 hours. The marks obtained are subsequently scaled down to 50 marks for calculating the final course grade.

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

	The student must secure a minimum of 40% marks in the SEE to pass the course. ****Self-Study Component should be included for CIE.**

CONTINUOUS INTERNAL EVALUATION (CIE)		Max. Marks	50%	Minimum Marks Required in CIE to Qualify for SEE (50% individually)
Theory	Weightage of CIE-1 and CIE-2 or CIE-3	30	15	25
	Assignment	10	05	
	Group Activity	05	2.5	
	Seminar	05	2.5	
	TOTAL	50	25	

SEE QUESTION PAPER PATTERN										
Q. No.	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
UNIT	1		2		3		4		5	
1. Two full questions (each of 20 Marks) are to be set from each unit.										
2. Student shall answer five full questions selecting one full question from each unit.										

Course Outcomes: After the completion of the course, the student will be able to	
CO1	Data engineering, data architecture, data engineering lifecycle, and the roles and responsibilities of data engineers in modern organizations
CO2	Apply data engineering principles to select appropriate technologies, architectures, and lifecycle components for developing scalable, reliable, and maintainable data platforms
CO3	Analyze data generation, storage, ingestion, transformation, and serving techniques to support efficient data processing and analytics requirements.
CO4	Evaluate data engineering solutions by considering data quality, scalability, security, privacy, governance, technology selection, and operational requirements in modern data ecosystems
CO5	Evaluate modern data engineering tools, platforms, and emerging technologies to recommend appropriate solutions for diverse data engineering applications and organizational requirements

COURSE SUSTAINABLE DEVELOPMENT GOALS (SDGs)		
Goal No.	**Goal Title**	**Description**
4	Quality Education	Operating Systems develops problem-solving, programming, and system design skills through concepts such as process management, memory management, synchronization, and file systems. It equips students with essential computing knowledge

		for higher education and lifelong learning.
8	Decent Work and Economic Growth	The course prepares students with industry-relevant skills in multitasking, multithreading, virtualization, resource management, and cloud computing, enhancing employability and supporting innovation and productivity in the IT sector.

Mapping of COs with POs
Strength of correlation: Strongly related-3, Moderately related-2, Weakly related-1, Not related-0

COs	PO1	PO2	PO3	PO4	PO5
CO1	3	2	0	0	2
CO2	3	3	2	2	2
CO3	3	3	3	2	2
CO4	3	3	2	2	3
CO5	3	2	2	1	3

Course Coordinators:

Dr. Veena A Associate Professor and HOP CSE(DS) Programme	Dr. K R Shylaja Professor and DEAN (R&D) CSE Programme

HEAD of School (HOS)
School of Computer Science and Engineering
Dr. Ambedkar Institute of Technology
Bengaluru-560 056

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

Course Title	Artificial Intelligence for Cyber Security				
Course Code	1MSCS105C				
Course Category	PEC (Professional Elective Course-I)				
Teaching - Learning Scheme & Credit	L	T	P	Total Credits	Total Teaching Hours
	3	0	0	3	Lecture: 42 + Tutorials: 0 + Practical: 0 + Self Study: 48 = 90
CIE Marks: 50	SEE Marks: 50		Total Max. Marks: 100		Duration of SEE: 03 Hours

Course Learning Objectives:	
CLO1	Explain the fundamental concepts of AI and ML and their applications in addressing cybersecurity challenges.
CLO2	Apply AI and ML techniques to detect, analyze, and mitigate cyber security threats and vulnerabilities.
CLO3	Utilize Natural Language Processing (NLP) methods in cybersecurity applications and analyze strategies for defending against adversarial attacks on AI systems.
CLO4	Evaluate the ethical, legal, and societal implications of deploying AI technologies in cybersecurity environments.

SYLLABUS	No. of Hours	BTLs
UNIT 1	**8**	**L1-L4**
Why Machine Learning and Security? : Cyber Threat Landscape, The Cyber Attacker's Economy, A Marketplace for Hacking Skills, Indirect Monetization, , What Machine Learning Is Not, Adversaries Using Machine Learning, Real-World Uses of Machine Learning in Security , Spam Fighting: An Iterative Approach and Limitations of Machine Learning in Security. Classifying and Clustering: Machine Learning: Problems and Approaches, Machine Learning in Practice: A Worked Example Training Algorithms to Learn.		
SELF-STUDY COMPONENT	**10 Hours**	
Cyber Attacker Economy & Threat Landscape: Hacking skill marketplaces, monetization models, and adversaries utilizing ML for automated targeting. ML in Security & Limitations: Spam fighting iteratively, limits of data-driven security (false positives, non-stationary environments, lack of ground truth). Classification & Clustering Basics: Mapping security problems to supervised vs. unsupervised paradigms and practical workflow for model training.		
UNIT 2	**8**	**L1-L4**
Classifying and Clustering: Supervised Classification Algorithms, Practical Considerations in Classification Clustering. Anomaly Detection : When to Use Anomaly Detection Versus Supervised Learning, Intrusion Detection with Heuristics, Data-Driven Methods, Feature Engineering for		

Anomaly Detection. Anomaly Detection with Data and Algorithms, Practical System Design Concerns, Malware Analysis.		
SELF-STUDY COMPONENT	**10 Hours**	
Supervised vs. Anomaly Detection: Choosing supervised classification (known threat patterns) vs. anomaly detection (zero-day/novel attacks). Heuristics & Feature Engineering: Moving from rule-based intrusion detection to data-driven features for network/host behaviors. Practical System Design & Malware Analysis: System architecture constraints (latency, throughput) and applying ML to static/dynamic malware analysis.		
UNIT 3	**9**	**L1-L4**
Network Traffic Analysis: Theory of Network Defense, Machine Learning and Network Security, Building a Predictive Model to Classify Network Attacks , Natural Language Processing in Cybersecurity. Protecting the Consumer Web: Types of Abuse and the Data That Can Stop Them, Authentication and Account Takeover, Supervised Learning for Abuse Problems , Clustering Abuse, Further Directions in Clustering.		
SELF-STUDY COMPONENT	**10 Hours**	
Network Defense & Predictive Modeling: Applying ML to packet flows, intrusion logs, and protocol analysis to classify cyber attacks. NLP in Cybersecurity: Using natural language processing for threat intelligence, command-line analysis, and phishing detection. Consumer Web Abuse & Account Protection: Detecting Account Takeover (ATO), credential stuffing, and bot activities via supervised and clustering approaches.		
UNIT 4	**9**	**L1-L4**
Production Systems: Defining Machine Learning System Maturity and Scalability, Data Quality, Model Quality, Performance, Maintainability, Monitoring and Alerting and Security and Reliability.		
SELF-STUDY COMPONENT	**9 Hours**	
System Maturity & Scalability: Scaling ML pipelines in high-throughput security environments while ensuring low latency. Data & Model Quality: Managing data drift, feature degradation, noisy security labels, and model performance metrics over time. Production Operations & Reliability: Monitoring, alerting, model maintainability, continuous retraining, and system fault tolerance.		
UNIT 5	**8**	**L1-L4**
Adversarial Machine Learning: The Importance of Adversarial ML, Security Vulnerabilities in Machine Learning Algorithms, Attack Transferability, Attack Technique: Model Poisoning, Attack Technique: Evasion Attack.		
SELF-STUDY COMPONENT	**9 Hours**	
Adversarial Vulnerabilities: Inherent security flaws in ML algorithms exposed to active, intelligent adversaries. **Attack Transferability:** Ability of adversarial attacks crafted against a surrogate model to succeed against a black-box production model. **Attack Techniques: Model Poisoning** (corrupting training data) and **Evasion Attacks** (crafting adversarial perturbations at inference time).		

Semester: I **Scheme: 2026**

TEXT BOOKS	
1.	Clarence Chio and David Freeman, Machine Learning and Security: Protecting Systems with Data and Algorithms, 2018, O'REILLY.

REFERENCE BOOKS	
1.	David Kim, Michael G. Solomon, Fundamentals of Information Systems Security, Jones & Bartlett Learning Publishers, 2013.
2.	Nina Godbole, Sunit Belapure, Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Wiley Publishers, 2011.
3.	Arun Kumar Krishna, AI in Cybersecurity: Transforming Threat Detection and Response, 2024.

	Assessment Details for both CIE and SEE
	The weightage assigned to the Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) is 50% each. To pass the course, students must secure a minimum of 40% marks separately in both CIE and SEE. **Continuous Internal Evaluation (CIE)** The CIE consists of four components: 1. CIE – Theory Component 2. Assignment 3. Group Activity 4. Seminar Students must secure a minimum of 50% marks in each component CIE Theory, Assignment, and Group Activity components to qualify for appearing in the SEE. **Semester End Examination (SEE)** The Theory SEE is conducted for 100 marks with a duration of 3 hours. The marks obtained are subsequently scaled down to 50 marks for calculating the final course grade. The student must secure a minimum of 40% marks in the SEE to pass the course. ****Self-Study Component should be included for CIE.**

CONTINUOUS INTERNAL EVALUATION (CIE)		**Max. Marks**	**50%**	**Minimum Marks Required in CIE to Qualify for SEE (50% individually)**
Theory	**Weightage of CIE-1 and CIE-2 or CIE-3**	**30**	**15**	**25**

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

Assignment	**10**	**05**	
Group Activity	**05**	**2.5**	
Seminar	**05**	**2.5**	
TOTAL	**50**	**25**	

SEE QUESTION PAPER PATTERN										
Q. No.	**Q1**	**Q2**	**Q3**	**Q4**	**Q5**	**Q6**	**Q7**	**Q8**	**Q9**	**Q10**
UNIT	1		2		3		4		5	
1. Two full questions (each of 20 Marks) are to be set from each unit.										
2. Student shall answer five full questions selecting one full question from each unit.										

Course Outcomes: After the completion of the course, the student will be able to	
CO1	Explain the fundamental concepts of Artificial Intelligence, Internet of Things (IoT), and their integration for developing intelligent IoT systems.
CO2	Apply machine learning, deep learning, and reinforcement learning techniques for processing and analyzing IoT-generated data
CO3	Develop AI-enabled IoT applications using cloud platforms and intelligent data analytics techniques
CO4	Analyze AI techniques for processing visual, text, audio, speech, and time-series data generated by IoT devices.
CO5	Apply AIoT concepts to develop solutions for smart homes, healthcare, smart cities, and industrial automation

COURSE SUSTAINABLE DEVELOPMENT GOALS (SDGs)		
Goal No.	**Goal Title**	**Description**
4	Quality Education	Operating Systems develops problem-solving, programming, and system design skills through concepts such as process management, memory management, synchronization, and file systems. It equips students with essential computing knowledge for higher education and lifelong learning.
8	Decent Work and Economic Growth	The course prepares students with industry-relevant skills in multitasking, multithreading, virtualization, resource management, and cloud computing, enhancing employability and supporting innovation and productivity in the IT sector.

Mapping of COs with POs

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

Strength of correlation: Strongly related-3, Moderately related-2, Weakly related-1, Not related-0

COs	PO1	PO2	PO3	PO4	PO5
CO1	3	2	0	0	2
CO2	3	3	2	2	2
CO3	3	3	3	2	2
CO4	3	3	2	2	3
CO5	3	2	2	1	3

Course Coordinators:

Aravinda.H.L

Dr. Aravind H L
Associate Professor
CSE Programme

Govindaraju.S

HEAD of School (HOS)
School of Computer Science and Engineering
Dr. Ambedkar Institute of Technology
Bengaluru-560 056

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

Course Title	**Artificial Intelligence for IoT**				
Course Code	**1MSCS105D**				
Course Category	**PEC (Professional Elective Course-I)**				
Teaching - Learning Scheme & Credit	**L**	**T**	**P**	**Total Credits**	**Total Teaching Hours**
	3	**0**	**0**	**3**	**Lecture: 42 + Tutorials: 0 + Practical: 0 + Self Study: 48 = 90**
CIE Marks: 50	**SEE Marks: 50**		**Total Max. Marks: 100**		**Duration of SEE: 03 Hours**

Course Learning Objectives:	
CLO1	Explain the fundamental concepts of Artificial Intelligence, Internet of Things (IoT), and their integration for developing intelligent IoT systems.
CLO2	Apply machine learning, deep learning, and reinforcement learning techniques for processing and analyzing IoT-generated data.
CLO3	Develop AI-enabled IoT applications using cloud platforms and intelligent data analytics techniques.
CLO4	Analyze AI techniques for processing visual, text, audio, speech, and time-series data generated by IoT devices.
CLO5	Apply AIoT concepts to develop solutions for smart homes, healthcare, smart cities, and industrial automation.

SYLLABUS	**No. of Hours**	**CLOs**
UNIT 1	**8**	CL-L3
Foundations and Basic Integrations of IoT and AI: Principles and Foundations of IoT and AI. What is IoT 101? Big data and IoT, Infusion of AI data science in IoT, Tools used. Machine Learning for IoT: Understanding ML and its Role in IoT, Learning paradigms, Prediction using linear regression, Logistic regression for classification, Classification using support vector machines, Naive Bayes, Decision trees, Ensemble learning.		
SELF-STUDY COMPONENT	**10 Hours**	
Techniques for improving the models: Feature Scaling to resolve uneven data scale Overfitting: Regularization, Cross validation. Hyperparameter tuning and grid search No Free Lunch theorem.		
UNIT 2	**8**	L1-L4
Advanced AI Techniques and Their Application in IoT: Deep Learning for IoT: Deep learning 101. Multilayered perceptron for regression and classification, Understanding CNNs, Recurrent neural networks. Reinforcement Learning for IoT: Introduction to RL, Simulated environments, Introducing Gymnasium, Q-learning, Q-Network.		

SELF-STUDY COMPONENT	10 Hours	
Auto Encoders: Denoising Auto Encoders, Variational Auto Encoders. Taxi drop-off using Q-network. DQN to play an Atari game, Double DQN, Dueling DQN Policy Gradients: Importance of Policy Gradients, Pong using Policy Gradients.		
UNIT 3	**9**	**L1-L4**
Implementing Intelligent IOT Solutions in Diverse Domains-1: AI Cloud Platforms for IoT: Introduction to Cloud Computing for IoT, Cloud Computing Basics, AI Services in Cloud Platforms, Training and Deploying AI Models for IoT on Cloud Platforms Deep Learning for Time-Series Data from IoT: Introduction to Time-Series Data, The Importance of Time-Series Analysis in IoT.		
SELF-STUDY COMPONENT	**10 Hours**	
Challenges in handling time-series data in IoT Working with Large Time-Series Data Using SparkML, Review of RNNs and LSTMs for Time-Series Analysis, LSTM for CO level prediction Pretrained Models for Time-Series Forecasting.		
UNIT 4	**9**	**L1-L4**
Implementing Intelligent IOT Solutions in Diverse Domains-2: Leveraging AI for Visual Data from IoT: A Brief Introduction to Visual Data in IoT Ecosystems, Handling Challenges in Visual Data in IoT. Leveraging AI Techniques for Visual Data in IoT. AI for Text, Audio and Speech Data from IoT: Introduction to Text, Audio, and Speech Data, Fundamentals of Text Data Processing, Working with Audio Data in IoT.		
SELF-STUDY COMPONENT	**9**	**Hours**
Audio processing and feature extraction, Audio command recognition, Speech recognition and transcription, Voice assistants and IoT integration, Using audio commands to control an IoT device		
UNIT 5	**8**	**L1-L4**
Applying AI and IoT in Real-World Scenarios:AI for Personal and Home IoT: Wearable and Fitness Devices, Smartwatch Health Monitoring in Action, Smart Home Devices, AI-driven home automation and intelligence, Case study: An Integrated smart home Implementation. AI for Smart Cities IoT: AI Applications in Smart Cities: Smart traffic management, Smart parking, Smart waste management, smart policing. AI-Powered Urban Analytics, Adapting IoT for Smart Cities and the Necessary Steps.		
SELF-STUDY COMPONENT	**9**	**Hours**
The smart camera breach (Ring cameras), The smart thermostat data leak, IoT data breach (ORVIBO Smart Home), IoT data breach (ORVIBO Smart Home)		

TEXT BOOKS

1. Dr. Amita Kapoor, *Hands-On Artificial Intelligence for IoT*, 2nd Edition, May 2025, Packt Publishing Ltd., Birmingham, UK, ISBN: 978-1-83546-718-3.

REFERENCE BOOKS	
1.	Yuxi (Hayden) Liu, Artificial Intelligence for IoT Cookbook, 1st Edition, March 2021, Packt Publishing Ltd., Birmingham, UK, ISBN: 978-1-83898-198-5. www.packtpub.com
2.	Rajkumar Buyya and Amir Vahid Dastjerdi (Editors), Internet of Things: Principles and Paradigms, 1st Edition, May 2016, Morgan Kaufmann (Elsevier), Cambridge, Massachusetts, USA, ISBN: 978-0-12-805395-9
3.	Perry Lea, Internet of Things for Architects: Architecting IoT Solutions by Implementing Sensors, Communication Infrastructure, Edge Computing, Analytics, and Security, 1st Edition, January 2018, Packt Publishing Ltd., Birmingham, UK, ISBN: 978-1-78847-059-9 (Paperback).

e-LEARNING RESOURCES

- NPTEL – Introduction to Internet of Things
- NPTEL – Artificial Intelligence and Machine Learning
- Google Cloud AI and IoT
- Coursera – AI and Machine Learning Courses
- edX – Artificial Intelligence and IoT Courses
- Artificial Intelligence for IoT

	Assessment Details for both CIE and SEE
	The weightage assigned to the Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) is 50% each. To pass the course, students must secure a minimum of 40% marks separately in both CIE and SEE. **Continuous Internal Evaluation (CIE)** The CIE consists of four components: 1. CIE – Theory Component 2. Assignment 3. Group Activity 4. Seminar Students must secure a minimum of 50% marks in each component CIE Theory, Assignment, and Group Activity components to qualify for appearing in the SEE. **Semester End Examination (SEE)** The Theory SEE is conducted for 100 marks with a duration of 3 hours. The marks obtained are subsequently scaled down to 50 marks for calculating the final course grade. The student must secure a minimum of 40% marks in the SEE to pass the course. ****Self-Study Component should be included for CIE.**

CONTINUOUS INTERNAL EVALUATION (CIE)	**Max. Marks**	**50%**	**Minimum Marks Required in CIE to Qualify for SEE (50%**

Semester: I **Scheme: 2026**

				individually)
Theory	**Weightage of CIE-1 and CIE-2 or CIE-3**	**30**	**15**	**25**
	Assignment	**10**	**05**	
	Group Activity	**05**	**2.5**	
	Seminar	**05**	**2.5**	
	TOTAL	**50**	**25**	

SEE QUESTION PAPER PATTERN										
Q. No.	**Q1**	**Q2**	**Q3**	**Q4**	**Q5**	**Q6**	**Q7**	**Q8**	**Q9**	**Q10**
UNIT	1		2		3		4		5	
1. Two full questions (each of 20 Marks) are to be set from each unit.										
2. Student shall answer five full questions selecting one full question from each unit.										

Course Outcomes: After the completion of the course, the student will be able to	
CO1	Explain the fundamental concepts of Artificial Intelligence, Internet of Things (IoT), and their integration for developing intelligent IoT systems.
CO2	Apply machine learning, deep learning, and reinforcement learning techniques for processing and analyzing IoT-generated data
CO3	Develop AI-enabled IoT applications using cloud platforms and intelligent data analytics techniques.
CO4	Analyze AI techniques for processing visual, text, audio, speech, and time-series data generated by IoT devices.
CO5	Apply AIoT concepts to develop solutions for smart homes, healthcare, smart cities, and industrial automation.

COURSE SUSTAINABLE DEVELOPMENT GOALS (SDGs)		
Goal No.	**Goal Title**	**Description**
4	Quality Education	Operating Systems develops problem-solving, programming, and system design skills through concepts such as process management, memory management, synchronization, and file systems. It equips students with essential computing knowledge for higher education and lifelong learning.
8	Decent Work and Economic Growth	The course prepares students with industry-relevant skills in multitasking, multithreading, virtualization, resource management, and cloud computing, enhancing employability and supporting innovation and productivity in the IT sector.

		management, and cloud computing, enhancing employability and supporting innovation and productivity in the IT sector.

Mapping of COs with POs
Strength of correlation: Strongly related-3, Moderately related-2, Weakly related-1, Not related-0

COs	PO1	PO2	PO3	PO4	PO5
CO1	3	2	1	0	2
CO2	3	3	2	2	2
CO3	3	3	3	2	2
CO4	3	3	2	2	3
CO5	3	2	2	1	3

Course Coordinators:

Aravinda.H.L

Dr. Aravinda H L
Associate Professor
CSE Programme

HEAD of School (HOS)
School of Computer Science and Engineering
Dr. Ambedkar Institute of Technology
Bengaluru-560 056

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

Course Title	**Generative Models**				
Course Code	**1MSCS106A**				
Course Category	**PEC II (Professional Elective Course II)**				
Teaching - Learning Scheme & Credit	**L**	**T**	**P**	**Total Credits**	**Total Teaching Hours**
	3	**0**	**0**	**3**	**Lecture: 42 + Tutorials: 0 + Practical: 0 + Self Study: 48 = 90**
CIE Marks: 50	**SEE Marks: 50**		**Total Max. Marks: 100**		**Duration of SEE: 03 Hours**

Course Learning Objectives:	
CLO1	Understand the fundamental concepts, principles and architectures of generative models and large language models.
CLO2	Analyze tokenization, embeddings, attention mechanisms and Transformer architectures used in modern generative models.
CLO3	Apply prompt engineering, semantic search, retrieval-augmented generation and agent-based techniques for generative AI applications.
CLO4	Analyze multimodal generative models and their applications involving text and image modalities.
CLO5	Apply fine-tuning, parameter-efficient adaptation, alignment and evaluation techniques for task-specific generative models.

SYLLABUS	**No. of Hours**	**BTLs**
UNIT 1	**8**	**L1-L4**
Foundations of Generative Models and Language Representation: Introduction to Generative AI and generative models; generative vs. discriminative models; applications and limitations. Language AI and evolution of language models; bag-of-words and dense vector representations. Tokenization. word, subword, character and byte tokens; tokenizer properties and vocabulary. Token embeddings. contextualized embeddings and text embeddings; Word2Vec and contrastive training. Attention-based language representation; encoder-only and decoder-only models; overview of the LLM training paradigm.		
SELF-STUDY COMPONENT	**10 Hours**	
Suggested study: Explore tokenizers and embeddings using a pretrained language model; compare word, subword and byte tokenization.		
UNIT 2	**8**	**L1-L4**
Transformers and Large Language Models: Transformer architecture: inputs and outputs of a Transformer LLM; overview of the forward pass. Self-attention and multi-head attention; Query, Key and Value; Transformer blocks. Positional embeddings; Rotary Positional Embeddings (RoPE); feed-forward layers, residual connections and normalization. Autoregressive text generation, probability distributions and sampling/decoding. Context size, parallel token processing, Key-Value caching and efficient attention; recent Transformer improvements.		

SELF-STUDY COMPONENT		10 Hours
Study a selected open Transformer LLM; compare decoding strategies, context size and KV caching.		
UNIT 3	**9**	**L1-L4**
Generative AI Techniques: Prompting, RAG and Agents: Text generation using pretrained models; controlling model output; prompt engineering and prompt components. Instruction-based prompting, few-shot prompting and in-context learning. Advanced prompting: chain prompting, reasoning with generative models, chain-of-thought, self-consistency and tree-of-thought. Constrained generation and output verification. Advanced text-generation tools: model I/O, prompt templates, chains and conversation memory. Agents and ReAct; semantic search, dense retrieval and reranking. Retrieval-Augmented Generation (RAG): pipeline, grounded generation, advanced RAG techniques and RAG evaluation.		
SELF-STUDY COMPONENT		**10 Hours**
Build a small prompt-engineering/RAG prototype; compare keyword search with dense retrieval.		
UNIT 4	**9**	**L1-L4**
Multimodal Generative Models: Introduction to multimodal large language models; Transformers for vision. Multimodal embedding models; CLIP and OpenCLIP for connecting text and images. Bridging the modality gap with BLIP-2; multimodal input preprocessing. Multimodal generation and prompting; image captioning and multimodal chat-based applications. Applications, challenges and evaluation considerations for multimodal generative systems.		
SELF-STUDY COMPONENT		**9 Hours**
Experiment with CLIP/image-text retrieval or image captioning using a pretrained multimodal model.		
UNIT 5	**8**	**L1-L4**
Training, Fine-Tuning, Alignment and Evaluation: Training paradigms for language models: pretraining, supervised fine-tuning (SFT) and preference tuning. Full fine-tuning and Parameter-Efficient Fine-Tuning (PEFT); instruction tuning with LoRA and QLoRA. Instruction-data templating; model quantization; training configuration and merging weights. Evaluation of generative models: word-level metrics, benchmarks, leaderboards, automated evaluation and human evaluation. Preference tuning and alignment; reward models, RLHF and Direct Preference Optimization (DPO). Responsible generative AI: hallucination, bias, safety, privacy and limitations of evaluation.		
SELF-STUDY COMPONENT		**9 Hours**
LoRA/QLoRA and generative-model evaluation; review current benchmarks and responsible-AI issues.		

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

TEXT BOOKS	
1.	Jay Alammar and Maarten Grootendorst, Hands-On Large Language Models: Language Understanding and Generation, O'Reilly Media, 2024, ISBN 9781098150952.

REFERENCE BOOKS	
1.	Sebastian Raschka, Build a Large Language Model (From Scratch), Manning Publications.
2.	Ian Goodfellow, Yoshua Bengio and Aaron Courville, Deep Learning, MIT Press.
3.	Jeremy Howard and Sylvain Gugger, Deep Learning for Coders with fastai and PyTorch, O'Reilly Media.

e-LEARNING RESOURCES
O'Reilly Learning: *Hands-On Large Language Models* – Jay Alammar and Maarten Grootendorst.
Hands-On LLM Official GitHub Repository: Code examples and notebooks accompanying the prescribed textbook.
Hugging Face Course: Transformers, Tokenizers, Datasets, Fine-Tuning and LLMs. Hugging Face Course
Hugging Face Documentation: Transformers and pretrained generative models. Hugging Face Transformers
Google Machine Learning Crash Course: Interactive modules, videos and exercises on ML and Generative AI. Google ML Crash Course
DeepLearning.AI: Online courses on Generative AI, Prompt Engineering, RAG, Agents, Fine-Tuning and Multimodal AI.
Google Colab: Cloud-based platform for implementing and experimenting with LLM and generative-AI models.

Assessment Details for both CIE and SEE

The weightage assigned to the Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) is 50% each.

To pass the course, students must secure a minimum of 40% marks separately in both CIE and SEE.

Continuous Internal Evaluation (CIE)

The CIE consists of four components:

1. CIE – Theory Component
2. Assignment
3. Group Activity
4. Seminar

Students must secure a minimum of 50% marks in each component CIE Theory, Assignment, and Group Activity components to qualify for appearing in the SEE.

Semester End Examination (SEE)

The Theory SEE is conducted for 100 marks with a duration of 3 hours. The marks obtained are

subsequently scaled down to 50 marks for calculating the final course grade.
The student must secure a minimum of 40% marks in the SEE to pass the course.
****Self-Study Component should be included for CIE.**

CONTINUOUS INTERNAL EVALUATION (CIE)		Max. Marks	50 %	Minimum Marks Required in CIE to Qualify for SEE (50% individually)
Theory	Weightage of CIE-1 and CIE-2 or CIE-3	30	15	25
	Assignment	10	05	
	Group Activity	05	2.5	
	Seminar	05	2.5	
	TOTAL	50	25	

SEE QUESTION PAPER PATTERN										
Q. No.	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
UNIT	1		2		3		4		5	
1. Two full questions (each of 20 Marks) are to be set from each unit.										
2. Student shall answer five full questions selecting one full question from each unit.										

Course Outcomes: After the completion of the course, the student will be able to	
CO1	Explain the principles, representations and architectures of generative models and language models.
CO2	Analyze Transformer architecture, attention, tokenization, embeddings and decoding mechanisms used in LLMs.
CO3	Develop generative-AI solutions using prompt engineering, semantic retrieval, RAG and agent-based techniques.
CO4	Analyze and apply multimodal generative models for text-image generation and understanding tasks.
CO5	Apply fine-tuning, PEFT, alignment and evaluation methods to adapt and assess generative models.

COURSE SUSTAINABLE DEVELOPMENT GOALS (SDGs)		
Goal No.	**Goal Title**	**Description**
4	Quality Education	Generative Models builds strong conceptual and practical foundations in large language models, Transformers, prompt engineering, retrieval-augmented generation and multimodal AI. It equips students with current AI knowledge and hands-on skills for research, higher education and lifelong learning, while fostering awareness of responsible and ethical AI use.

8	Decent Work and Economic Growth	The course prepares students with industry-relevant skills in building, fine-tuning, evaluating and deploying generative-AI applications such as LLM-based assistants, RAG systems and agents, enhancing employability and supporting innovation and productivity across the IT and AI sectors.

Mapping of COs with POs
Strength of correlation: Strongly related-3, Moderately related-2, Weakly related-1, Not related-0

COs	PO1	PO2	PO3	PO4	PO5
CO1	3	2	1	1	2
CO2	3	3	2	2	2
CO3	3	3	3	2	2
CO4	3	3	2	2	3
CO5	3	2	2	2	3

Course Coordinators:

27/10/2026	27/10/26
Dr. Shylaja K R Professor CSE Programme	Dr. Asha K N Assistant Professor CSE Programme

HEAD of School (HOS)
School of Computer Science and Engineering
Dr. Ambedkar Institute of Technology
Bengaluru-560 056

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

M.Tech CSE

Semester: I **Scheme: 2026**

Course Title	**Multi Modal Data Analytics**				
Course Code	**1MSCS106B**				
Course Category	**PEC II (Professional Elective Course II)**				
Teaching - Learning Scheme & Credit	**L**	**T**	**P**	**Total Credits**	**Total Teaching Hours**
	3	**0**	**0**	**3**	**Lecture: 42 + Tutorials: 0 + Practical: 0 + Self Study: 48 = 90**
CIE Marks: 50	**SEE Marks: 50**		**Total Max. Marks: 100**		**Duration of SEE: 03 Hours**

Course Learning Objectives:	
CLO1	Introduce the fundamental concepts, characteristics, sources, and applications of multimodal data and multimodal analytics.
CLO2	Familiarize students with techniques for multimodal data ingestion, preprocessing, transformation, synchronization, integration, and feature engineering.
CLO3	Develop proficiency in designing scalable data storage, retrieval, streaming, and analytics pipelines for multimodal AI applications.
CLO4	Enable students to design end-to-end multimodal data analytics solutions using appropriate AI, data engineering, cloud, and MLOps technologies for real-world applications.

SYLLABUS	**No. of Hours**	**BTLs**
UNIT 1	**8**	L1-L3
Introduction of the semantic layer, Data contracts, Multimodal data lifecycle, Storage and retrieval backbone with vector-native stores, composable lake houses, nearest neighbor (ANN) indexing, Kurbernetes orchestration.		
SELF-STUDY COMPONENT	**10 Hours**	
StreamBuy case study, Revault to compare the three platform archetypes among which every team chooses managed AI platforms, composable architectures, and real-time streaming infrastructure.		
UNIT 2	**8**	L1-L3
ByteEats and reframes ingestion as a set of trade-offs, comparison of compares traditional, hybrid AI, and agentic AI ingestion patterns, feature engineering across modality, tensors as a universal representation, fusion architectures, and the multimodal feature store.		
SELF-STUDY COMPONENT	**10 Hours**	

Storage as cognitive scaffolding, graph augmented retrieval, caching tiers, and edge-aware memory through TelMo case.		
UNIT 3	**9**	**L1-L4**
FieldSense the fusing of asynchronous modalities under event-time constraints, MLOps to the multimodal case with continuous integration, continuous delivery (CI/CD) across code and data, reproducible experiment triplets, and drift detection in latent space.		
SELF-STUDY COMPONENT	**10 Hours**	
ForgeSense Case Study: Study the design of a multimodal MLOps pipeline for asynchronous data fusion, focusing on event-time processing, CI/CD for code and data, reproducible experiments, model lifecycle management, and drift detection in multimodal representation spaces.		
UNIT 4	**9**	**L1-L4**
Stream-processing engines, batch frameworks, distributed AI runtimes on the same healthcare workload using Atlas Health AI, inference as engineering through the inference triangle, modal contracts, canary deployment, and observability.		
SELF-STUDY COMPONENT	**9**	**Hours**
Eigoteen Labs case study: Comparative study of stream-processing engines, batch frameworks, distributed AI runtimes, inference engineering, modal contracts, canary deployment, and observability for healthcare AI workloads.		
UNIT 5	**8**	**L1-L4**
Governance, security, and responsible AI intrinsic properties of the architecture, production-grade pipeline		
SELF-STUDY COMPONENT	**9**	**Hours**
StreamKit Case Study: Study of governance, security, privacy, responsible AI, and production-grade architecture for building scalable and trustworthy multimodal AI pipelines.		

TEXT BOOKS

1.	**Vasundra Srinivasan,** Data Engineering for Multimodal AI: Architecting Scalable Systems for Next-Generation AI Applications, *O'Reilly Media, 2026.*
2	Santosh Kumar and Sanjay Kumar Singh, Multimodal Machine Learning: Techniques and Applications, Elsevier, 2021

REFERENCE BOOKS

1.	Designing Machine Learning Systems: An Iterative Process for Production-Ready Applications, O'Reilly Media, 2022.
2.	Fundamentals of Data Engineering, O'Reilly Media, 2022.

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

M.Tech CSE

Semester: I **Scheme: 2026**

e-LEARNING RESOURCES
NPTEL – Multimodal Interaction, IIT Madras https://archive.nptel.ac.in/courses/106/106/106106200/
NPTEL – Neural Networks for Computer Vision and Natural Language Processing, IIT Guwahati https://nptel.ac.in/courses/106103843
NPTEL – Data Analytics / Learning Analytics Tools, IIT Bombay https://nptel.ac.in/courses/106101224
TensorFlow – Multimodal / Data Pipeline Resources
https://github.com/Data-Engineering-for-Multimodal-AI

	Assessment Details for both CIE and SEE
	The weightage assigned to the Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) is 50% each. To pass the course, students must secure a minimum of 50% marks in CIE and 40% marks in SEE. **Continuous Internal Evaluation (CIE)** The CIE consists of four components: 1. CIE – Theory Component 2. Assignment 3. Group Activity 4. Seminar Students must secure a minimum of 50% marks in each component CIE Theory, Assignment, and Group Activity components to qualify for appearing in the SEE. **Semester End Examination (SEE)** The Theory SEE is conducted for 100 marks with a duration of 3 hours. The marks obtained are subsequently scaled down to 50 marks for calculating the final course grade. The student must secure a minimum of 40% marks in the SEE to pass the course. ****Self-Study Component should be included for CIE.**

CONTINUOUS INTERNAL EVALUATION (CIE)		**Max. Marks**	**50 %**	**Minimum Marks Required in CIE to Qualify for SEE (50% individually)**
Theory	**Weightage of CIE-1 and CIE-2 or CIE-3**	**30**	**15**	**25**
	Assignment	**10**	**05**	
	Group Activity	**05**	**2.5**	
	Seminar	**05**	**2.5**	
	TOTAL	**50**	**25**	

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

M.Tech CSE

Semester: I **Scheme: 2026**

SEE QUESTION PAPER PATTERN										
Q. No.	**Q1**	**Q2**	**Q3**	**Q4**	**Q5**	**Q6**	**Q7**	**Q8**	**Q9**	**Q10**
UNIT	1		2		3		4		5	
1. Two full questions (each of 20 Marks) are to be set from each unit.										
2. Student shall answer five full questions selecting one full question from each unit.										

Course Outcomes: After the completion of the course, the student will be able to	
CO1	Explain the characteristics, sources, challenges, and applications of multimodal data and its role in data-driven AI systems.
CO2	Apply appropriate data ingestion, preprocessing, transformation, synchronization, and quality-management techniques to multimodal datasets.
CO3	Develop multimodal representations using feature engineering, embeddings, and data fusion techniques for analytics applications.
CO4	Design and implement scalable storage, retrieval, streaming, and analytics pipelines for multimodal data using suitable data engineering technologies.
CO5	Develop an end-to-end multimodal data analytics solution for a real-world application, considering scalability, deployment, monitoring, security, and responsible AI principles.

COURSE SUSTAINABLE DEVELOPMENT GOALS (SDGs)		
Goal No.	**Goal Title**	**Description**
4	Quality Education	Develops advanced skills in multimodal data analytics, AI, data engineering, MLOps and emerging technologies, supporting quality higher education and industry-relevant technical competencies.
8	Decent Work and Economic Growth	Builds employability and entrepreneurial skills in AI, data engineering, analytics and scalable computing, preparing students for technology-driven careers and innovation.
9	Industry, Innovation and Infrastructure	Enables students to design scalable multimodal data pipelines, distributed AI systems, cloud-based architectures and innovative data-driven solutions for real-world applications.

Mapping of COs with POs

Strength of correlation: Strongly related-3, Moderately related-2, Weakly related-1, Not related-0

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**
CO1	3	2	0	0	2
CO2	3	3	2	2	3

CO3	3	3	3	2	3
CO4	3	3	3	2	3
CO5	3	3	3	3	3

Course Coordinators:

27/10/2026	
Dr. K R Shylaja Professor and DEAN (R&D) CSE Programme	Dr. Veena A Associate Professor and HOP CSE (DS) Programme

HEAD of School (HOS)
School of Computer Science and Engineering
Dr. Ambedkar Institute of Technology
Bengaluru-560 056

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

Course Title	**Explainable AI**				
Course Code	**1MSCS106C**				
Course Category	**PEC II (Professional Elective Course II)**				
Teaching - Learning Scheme & Credit	**L**	**T**	**P**	**Total Credits**	**Total Teaching Hours**
	3	**0**	**0**	**3**	**Lecture: 42 + Tutorials: 0 + Practical: 0 + Self Study: 48 = 90**
CIE Marks: 50	**SEE Marks: 50**		**Total Max. Marks: 100**		**Duration of SEE: 03 Hours**

Course Learning Objectives:	
CLO1	Explain the fundamental concepts, significance, and limitations of Explainable AI and model interpretability.
CLO2	Describe intrinsic, model-agnostic, and post-hoc explainability techniques used in machine learning.
CLO3	Illustrate methods for interpreting deep learning models using visualization, feature attribution, counterfactual, and causal explanations.
CLO4	Explain the principles of fairness, bias detection, ethical AI, and adversarial robustness.
CLO5	Discuss explainable AI tools and their applications in image analysis and domain-specific problems.

SYLLABUS	**No. of Hours**	**RBTLs**
UNIT 1	**9**	L1-L4
Introduction to Machine Learning and Explainable AI, Purpose, need, and limits of explainability and interpretability, Explainability within the model development pipeline.		
SELF-STUDY COMPONENT	10 Hours	
Global and local explanations; intrinsic vs. post-hoc explainability; properties and evaluation of explanations. Case studies on explainability in healthcare and financial decision-making.		
UNIT 2	**9**	L1-L4
Intrinsic Explainable Models: Linear/Logistic Regression, Decision Trees, K-Nearest Neighbours. Model transparency and loss function behaviour. Model-Agnostic Techniques: Surrogate models.		
SELF-STUDY COMPONENT	10 Hours	
Feature importance, LIME and SHAP; explainability of bagging and boosting models. Case studies on loan approval and customer churn prediction using explainable ML models.		
UNIT 3	**8**	L1-L4

Explainability in Deep Neural Networks: Post-hoc and agnostic approaches: adversarial features, data augmentation, occlusion methods. Layer-wise analysis: CAM, Grad-CAM, DeepSHAP, DeepLIFT. Counterfactual and causal inference concepts; What-If Tool, Contrastive Explanation Method (CEM).		
SELF-STUDY COMPONENT	**10 Hours**	
Temporal and Domain-Specific Explainability: Time-series and LSTM attribution using Integrated Gradients. Explainability for Transformers and Large Language Models (LLMs). Case studies on medical image classification and sentiment analysis using deep learning and Transformers.		
UNIT 4	**8**	**L1-L4**
Bias detection and mitigation in AI models. Fairness and ethical aspects of explainable systems. Adversarial robustness: evasion attacks, pre-processing defences, adversarial training. Evaluation and certification of robustness.		
SELF-STUDY COMPONENT	**9 Hours**	
Feature engineering and its role in interpretability. Case studies on bias detection and mitigation in AI models and adversarial robustness. Review of current research directions in XAI.		
UNIT 5	**8**	**L1-L4**
Advanced Applications of Explainable AI: Explainability in complex and deep learning models, visualization and interpretation of model predictions, feature attribution methods, explanation of image classification models.		
SELF-STUDY COMPONENT	**9 Hours**	
Comparison of explanation methods, limitations and challenges of XAI. Case studies on explainability in healthcare, finance and image-based applications.		

TEXT BOOKS

1	Christoph Molnar, Interpretable Machine Learning, Lulu Press, 2022.
2	Explainable AI with Python, Antonio Di Cecco and Leonida Gianfagna, Springer, 2021
3	Ankur Taly et al., Explainable AI: Interpreting, Explaining and Visualizing Deep Learning, Springer, 2019.

REFERENCE BOOKS

1	Hands-On Explainable AI (XAI) with Python: Interpret, visualize, explain, and integrate reliable AI for fair. secure, and trustworthy AI apps, Denis Rothman, Packt publisher, 2020
2	Interpretable Machine Learning with Python: Learn to build interpretable high-performance models with hands-on real-world examples, by SergMasís , Packt publisher, 2021
3	Michael Munn, David Pitman, Explainable AI for Practitioners, O'Reilly Media, Inc., 2022
4	Leonida Gianfagna, Antonio Di Cecco, Explainable AI with Python, Springer, 2021

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

e-LEARNING RESOURCES	
1	Google Machine Learning – Explainability Resources
2	SHAP – Documentation and Tutorials
3	TensorFlow – Model Interpretability Tutorials
4	Google What-If Tool – Interactive ML Explainability Tool

	Assessment Details both (CIE and SEE)
	The weightage assigned to the Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) is 50% each. To pass the course, students must secure a minimum of 40% marks separately in both CIE and SEE. **Continuous Internal Evaluation (CIE)** The CIE consists of four components: 1. CIE – Theory Component 2. Assignment 3. Group Activity 4. Seminar Students must secure a minimum of 50% marks in each component CIE Theory, Assignment, and Group Activity components to qualify for appearing in the SEE. **Semester End Examination (SEE)** The Theory SEE is conducted for 100 marks with a duration of 3 hours. The marks obtained are subsequently scaled down to 50 marks for calculating the final course grade. The student must secure a minimum of 40% marks in the SEE to pass the course. ****Self-Study Component should be included for CIE.**

CONTINUOUS INTERNAL EVALUATION (CIE)		**Max. Marks**		Minimum Marks to be scored in CIE, to qualify to take SEE (40% individually)
Theory	**Weightage of CIE1 and CIE2 Tests or CIE3**	**30**	**15**	**25**
	Assignment	**10**	**05**	
	Group Activity	**05**	**2.5**	
	Seminar	**05**	**2.5**	
TOTAL		**50**	**25**	

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

SEE QUESTION PAPER PATTERN										
Q. No.	**Q1**	**Q2**	**Q3**	**Q4**	**Q5**	**Q6**	**Q7**	**Q8**	**Q9**	**Q10**
UNIT	1		2		3		4		5	
1. Two full questions (each of 20 Marks) are to be set from each unit.										
2. Student shall answer five full questions selecting one full question from each unit.										

Course Outcomes: After the completion of the course, the student will be able to

CO1	Explain the fundamental principles of Explainable AI and interpretability, and their role in the machine learning development pipeline.
CO2	Apply intrinsic and model-agnostic techniques to interpret machine learning models and explain their predictions.
CO3	Analyze deep learning models using visualization, layer-wise analysis, feature attribution, counterfactual, and causal explanation techniques.
CO4	Evaluate AI systems with respect to bias, fairness, ethical considerations, and adversarial robustness.
CO5	Develop and assess explainable AI approaches for deep learning and domain-specific applications, including image-based applications.

COURSE SUSTAINABLE DEVELOPMENT GOALS (SDGs)		
Goal No.	**Goal Title**	**Description**
3	Good Health and Well-being	Promotes the use of transparent and trustworthy AI for healthcare, medical diagnosis, and decision support.
6	Industry, Innovation and Infrastructure	Supports responsible AI innovation and the development of reliable, transparent, and robust intelligent systems.
10	Reduced Inequalities	Addresses bias and fairness in AI systems to promote equitable and non-discriminatory decision-making.
16	Peace, Justice and Strong Institutions	Promotes transparency, accountability, and responsible decision-making through explainable AI systems.

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

Mapping of COs with POs **Strength of correlation:** Strongly related-3, Moderately related-2, Weakly related-1, Not related-0					
COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**
CO1	2	3	3	2	3
CO2	3	2	3	2	3
CO3	3	2	3	3	3
CO4	2	3	3	3	3
CO5	3	3	2	3	3

Course Coordinators:

27/10/2026	27/10/26
Dr. Shylaja K R Professor and DEAN (R&D) CSE Programme	Arathi P Assistant Professor CSE Programme

HEAD of School (HOS)
School of Computer Science and Engineering
Dr. Ambedkar Institute of Technology
Bengaluru-560 056

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

Course Title	Software Engineering and Web Applications				
Course Code	1MSCS106D				
Course Category	PEC II (Professional Elective Course II)				
Teaching - Learning Scheme & Credit	L	T	P	Total Credits	Total Teaching Hours
	3	0	0	3	Lecture: 42 + Tutorials: 0 + Practical: 0 + Self Study: 48 = 90
CIE Marks: 50	SEE Marks: 50		Total Max. Marks: 100		Duration of SEE: 03 Hours

Course Learning Objectives:	
CLO1	Apply software architecture, component-based development, and service-oriented principles in designing large-scale software systems.
CLO2	Analyse software reliability, security, quality, and dependability using appropriate engineering techniques and metrics.
CLO3	Apply software evolution, configuration management, and process improvement practices to enhance software maintainability.
CLO4	Apply web engineering principles to design effective web application architecture, navigation, and user interfaces.
CLO5	Apply web engineering practices to develop, test, deploy, and maintain secure and high-quality web applications.

SYLLABUS	No. of Hours	BTLs
UNIT 1	8	L1-L4
Software Architecture and Reuse Engineering: Architectural Design – Architectural Design Decisions, Architectural Views, Architectural Patterns, Application Architectures, Component-Based Software Engineering – Components and Component Models, CBSE Process, Component Composition, Service-Oriented Software Engineering – Service Oriented Architecture (SOA), RESTful Services, Service engineering, Service Composition.		
SELF-STUDY COMPONENT	10 Hours	
Architectural Decisions & Views, Architectural Patterns & Types, CBSE & Composition, SOA & RESTful Services, Service Engineering. Quality Attribute Scenarios, View-Based Architecture Documentation, Microservices Architecture, Event-Driven Architecture, Event-Driven Service Integration, Service Discovery and Orchestration, Service-Level Agreement (SLA) Management, Distributed Transaction Management, Service Reliability Engineering, Observability and Distributed Tracing, Service Governance		
UNIT 2	8	L1-L4

Software Dependability and Security: Dependable systems: Dependability Properties, Sociotechnical systems, Redundancy and diversity, Dependable processes, Reliability Engineering – Availability and Reliability, Reliability Requirements, Fault-Tolerant Architectures, Reliability Measurement, Security Engineering – Security and dependability Security Requirements, Secure Systems Design, Security Testing and Assurance.		
SELF-STUDY COMPONENT	**10 Hours**	
Dependability & Sociotechnical Systems, Redundancy, Diversity & Fault Tolerance, Reliability Metrics, Security Engineering. Safety-Critical Systems, Resilience Engineering, Trustworthy AI Systems, Cyber-Physical Systems Safety, Failure Mode and Effects Analysis (FMEA), Security Assurance for AI-Enabled and Autonomous Systems, Digital Twins for Reliability Prediction and Failure Simulation, Post-Quantum Cryptography and Secure System Migration		
UNIT 3	**9**	**L1-L4**
Software Evolution and Configuration Management: Software Evolution – Evolution Processes, Legacy Systems, Software Maintenance, Quality Management – Software Quality, Quality Assurance, Software Standards, Reviews and Inspections, Software Measurement. Configuration Management – Version Management, System Building, Change Management, Release Management.		
SELF-STUDY COMPONENT	**10 Hours**	
Software Evolution & Maintenance, Quality Management & Reviews, Configuration & Release Management AI-Assisted Software Maintenance and Code Modernization, Legacy System Modernization and Migration to Microservices, Continuous Quality Engineering in DevOps, Software Quality Metrics for Cloud-Native Applications, Software Supply Chain Security and Software Bill of Materials (SBOM)		
UNIT 4	**9**	**L1-L4**
Web Application Analysis and Design: Introduction to Web Applications, Characteristics of Web Applications, Web Application Analysis, Content Design, Architectural Design for Web Applications, Navigation Design, Interface Design, User Experience Design, Responsive Web Design Principles, Modeling Web Applications.		
SELF-STUDY COMPONENT	**9**	**Hours**
Web App Characteristics & Analysis, Architecture & Design, Navigation & Responsive Design Progressive Web Applications (PWAs) and Offline-First Design, Micro-Frontend Architecture for Large-Scale Web Applications , Intelligent Navigation and Context-Aware User Interfaces		
UNIT 5	**8**	**L1-L4**
Web Application Development, Testing and Maintenance: Web Application Construction, Web Components, Web Services Integration, Testing Web Applications, Functional Testing, Performance Testing, Quality Assurance for Web Applications, Web Application Security, Deployment Strategies, Web Application Maintenance and Evolution.		
SELF-STUDY COMPONENT	**9**	**Hours**

Construction & Integration, Multi-Layer Web Testing, Security & Deployment. Micro-Frontend Integration and Module Federation, Containerization and Application Integration using Docker, Performance, Load, and Stress Testing of Web Applications, API Contract Testing and Automated Integration

TEXT BOOKS	
1.	Software Engineering, Ian Sommerville, Pearson Education, 10th Edition, 2016
2.	Web Engineering A Practitioner's Approach Roger S. Pressman and David Lowe, McGraw-Hill Higher Education, 1st Edition, 2008.

REFERENCE BOOKS	
1.	Software Architecture in Practice Len Bass, Paul Clements, and Rick Kazman, Addison-Wesley Professional, 4th Edition, 2021.
2.	Web Engineering Gerti Kappel, Birgit Pröll, Siegfried Reich, and Werner Retschitzegger, Wiley, 2006.
3.	DevOps A Software Architect's Perspective Len Bass, Ingo Weber, and Liming Zhu, Addison-Wesley Professional, 2015.

e-LEARNING RESOURCES
https://ocw.mit.edu/search/?q=software+engineering
https://developer.mozilla.org/en-US/docs/Learn_web_development
https://www.w3schools.com/
https://www.freecodecamp.org/learn/
https://www.youtube.com/@TraversyMedia
https://www.youtube.com/@NetNinja

Assessment Details for both CIE and SEE

The weightage assigned to the Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) is 50% each.

To pass the course, students must secure a minimum of 50 % marks separately in both CIE and SEE.

Continuous Internal Evaluation (CIE)

The CIE consists of four components:

1. CIE – Theory Component
2. Assignment
3. Group Activity
4. Seminar

Students must secure a minimum of 50% marks in each component CIE Theory, Assignment, and Group

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

Activity components to qualify for appearing in the SEE.

Semester End Examination (SEE)

The Theory SEE is conducted for 100 marks with a duration of 3 hours. The marks obtained are subsequently scaled down to 50 marks for calculating the final course grade.

The student must secure a minimum of 40% marks in the SEE to pass the course.

****Self-Study Component should be included for CIE.**

CONTINUOUS INTERNAL EVALUATION (CIE)		Max. Marks	50 %	Minimum Marks Required in CIE to Qualify for SEE (50% individually)
Theory	Weightage of CIE-1 and CIE-2 or CIE-3	30	15	25
	Assignment	10	05	
	Group Activity	05	2.5	
	Seminar	05	2.5	
	TOTAL	50	25	

SEE QUESTION PAPER PATTERN										
Q. No.	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
UNIT	1		2		3		4		5	
1. Two full questions (each of 20 Marks) are to be set from each unit.										
2. Student shall answer five full questions selecting one full question from each unit.										

Course Outcomes: After the completion of the course, the student will be able to	
CO1	Apply software architecture, component-based development, and service-oriented principles in designing large-scale software systems.
CO2	Analyse software reliability, security, quality, and dependability using appropriate engineering techniques and metrics.
CO3	Apply software evolution, configuration management, and process improvement practices to enhance software maintainability.
CO4	Apply web engineering principles to design effective web application architecture, navigation, and user interfaces.
CO5	Apply web engineering practices to develop, test, deploy, and maintain secure and high-quality web applications.

COURSE SUSTAINABLE DEVELOPMENT GOALS (SDGs)

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I Scheme: 2026

COURSE SUSTAINABLE DEVELOPMENT GOALS (SDGs)		
Goal No.	**Goal Title**	**Description**
4	Quality Education	Operating Systems develops problem-solving, programming, and system design skills through concepts such as process management, memory management, synchronization, and file systems. It equips students with essential computing knowledge for higher education and lifelong learning.
8	Decent Work and Economic Growth	The course prepares students with industry-relevant skills in multitasking, multithreading, virtualization, resource management, and cloud computing, enhancing employability and supporting innovation and productivity in the IT sector.

Mapping of COs with POs with PSOs

Strength of correlation: Strongly related-3, Moderately related-2, Weakly related-1, Not related-0

COs	PO1	PO2	PO3	PO4	PO5
CO1	2	1	3	1	3
CO2	2	2	3	3	2
CO3	2	2	2	2	3
CO4	1	2	3	1	2
CO5	2	2	3	2	3

Course Coordinators:

21/10/2026

Dr. K R Shylaja
Professor and DEAN (R&D)
CSE Programme

HEAD of School (HOS)
School of Computer Science and Engineering
Dr. Ambedkar Institute of Technology
Bengaluru-560 056

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

Course Title	Software Automation Testing Laboratory				
Course Code	1MSCSL107A				
Course Category	PCCL (Professional Core Course Laboratory)				
Teaching - Learning Scheme & Credit	L	T	P	Total Credits	Total Teaching Hours
	0	0	1	1	Lecture: 0 + Tutorials: 0 + Practical: 28 + Self Study: 02 = 30
CIE Marks: 50	SEE Marks: 50		Total Max. Marks: 100		Duration of SEE: 03 Hours

Course Learning Objectives:	
CLO1	Implement automated UI and API test scripts using Playwright and WireMock to validate software functionality.
CLO2	Analyze automation test results, identify defects, evaluate test coverage, and troubleshoot failures using appropriate testing tools and reports.
CLO3	Evaluate automation frameworks, testing strategies, and test artifacts to improve software quality and testing efficiency.
CLO4	Develop a scalable end-to-end automation testing framework by integrating UI testing, API mocking, reporting, and continuous testing practices for real-world applications.

SYLLABUS	
PART: A - List of Experiments	
1.	Develop and execute an automated test using Playwright to validate the login functionality of a web application with valid and invalid credentials. Analyze the execution results to identify failed scenarios and evaluate the effectiveness of the implemented automation script in improving software quality.
2.	Design and implement automated tests using Playwright to validate the search and filtering functionality of an e-commerce application with multiple input combinations. Analyze the test results for different scenarios and evaluate the robustness and maintainability of the automation scripts.
3.	Develop an end-to-end automated test suite using Playwright to validate shopping cart and checkout functionalities of an e-commerce application. Analyze the execution outcomes and evaluate the test suite based on coverage, reliability, and execution efficiency.
4.	Configure WireMock to simulate a REST API using predefined stub responses for GET requests. Analyze the mocked API responses under different request conditions and evaluate the effectiveness of service virtualization in supporting application testing.
5.	Create WireMock stubs to simulate POST requests with valid and invalid input data for a REST API. Analyze the application's behavior using mocked responses and evaluate the suitability of WireMock for testing applications when backend services are unavailable.
6.	Integrate Playwright with WireMock to automate the testing of a web application that

	interacts with mocked REST APIs. Analyze the interaction between the user interface and mocked services and evaluate the reliability, repeatability, and efficiency of the integrated automation testing approach.
PART: B - TYPICAL OPEN-ENDED EXPERIMENTS	
Design and Implement an End-to-End Automated Testing Framework for a Web Application: Develop a scalable automated testing framework for any open-source web application (banking, education, healthcare, etc.) using Playwright. Identify critical user workflows, automate the test scenarios, execute them under different conditions, and analyze the test outcomes.	
1.	Design and implement a data-driven automated testing framework using Playwright / Selenium / Available open source for a selected web application. Identify critical workflows and execute the same test scenarios using multiple sets of valid, invalid, boundary, and unexpected input data. Implement parameterization, test fixtures, reusable functions, and appropriate assertions. Analyze the test results to identify defects and evaluate the maintainability and reusability of the framework.
2.	Develop an automated testing solution using Playwright / Selenium / Available open source to evaluate a selected web application across multiple browsers and viewport configurations. Identify critical functional and UI scenarios, execute them across different browser engines and screen sizes, capture failures and screenshots, and compare the test results. Analyze browser-specific and responsive behaviour and evaluate the effectiveness of automated cross-browser testing.
3.	Design and implement a service virtualization framework using WireMock for a selected REST-based application. Identify different API response conditions such as successful responses, validation errors, authentication failures, timeouts, unavailable services, and server errors. Create suitable WireMock stubs and test how the application responds to these conditions. Analyze the results and evaluate the effectiveness of API contract and negative testing.
4.	Develop an automated testing framework using Playwright / Selenium / Available open source that generates comprehensive test execution reports for a selected web application. Execute a suitable set of functional test cases, capture screenshots and traces for failed scenarios, categorize failures, and analyze test execution time, pass/fail statistics, and coverage. Develop a mechanism to identify recurring failures and evaluate the effectiveness of automated reporting in software quality assessment.
5.	Design and implement a Continuous Testing pipeline for a web application by integrating Playwright-based automated tests with a CI/CD platform such as GitHub Actions or Jenkins. Configure automated test execution whenever application changes are introduced, collect test results and artifacts, and configure the pipeline to identify and report failures. Analyze the benefits and limitations of continuous testing in improving software quality and reducing regression testing effort.
6.	Design and implement a scalable end-to-end quality engineering framework for a selected open-source web application by integrating UI automation using Playwright, API simulation using WireMock, reusable test components, test data management, automated reporting, and continuous test execution. Identify critical business workflows, develop functional and negative test scenarios, execute the tests under different conditions, analyze failures and coverage, and evaluate the scalability, reliability, maintainability, and

	proposed framework.

Case Study on **OPEN ENDED PROJECTS**

1.	Paul C. Jorgensen, The Craft of Software Testing: Managing the Testing Process, 3rd Edition, Auerbach Publications.
2.	Alan Richardson, Selenium WebDriver Recipes in C#, 2nd Edition, Leanpub
3.	Srinivasan Desikan and Gopalaswamy Ramesh, Software Testing: Principles and Practices, Pearson Education.

1.	Evan Burchard, The Web Application Hacker's Handbook, Wiley.

https://playwright.dev

https://www.wiremock.io/webinars/developer-portals

https://playwright.dev/community/learn-videos

Assessment Details for CIE

CONTINUOUS INTERNAL EVALUATION (CIE)		Max. Marks	Minimum Marks Required in CIE to Qualify for SEE (50% in Each Component)
Laboratory Components	**Each experiment will be evaluated using the prescribed rubrics, and the average of the marks obtained across all experiments will be considered for the final evaluation.**	**10**	25
	LAB CIE	**40**	
	TOTAL	**50**	

- All laboratory experiments shall be included for the practical examination.
- The instructions and marks breakup printed on the cover page of the answer script shall be strictly adhered to by the examiners.
- Students shall be allowed to randomly select any one experiment from the given lot.
- Change of the selected experiment shall be permitted only once.

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

- Any change of experiment shall attract mark deduction as per the Dr. AIT Autonomous Examination Rules and Regulations

Course Outcomes: After the completion of the course, the student will be able to	
CO1	Apply software architecture, component-based development, and service-oriented principles in designing large-scale software systems.
CO2	Analyze software reliability, security, quality, and dependability using appropriate engineering techniques and metrics.
CO3	Apply software evolution, configuration management, and process improvement practices to enhance software maintainability.
CO4	Apply web engineering principles to design effective web application architecture, navigation, and user interfaces.
CO5	Apply web engineering practices to develop, test, deploy, and maintain secure and high-quality web applications.

COURSE SUSTAINABLE DEVELOPMENT GOALS (SDGs)		
Goal No.	**Goal Title**	**Description**
4	Quality Education	Ensure Inclusive and equitable quality education and promote lifelong learning opportunities for all
8	Decent Work and Economic Growth	Promote sustained, inclusive and sustainable economic growth, full and productive employment and decent work for all.

Mapping of COs with POs

Strength of correlation: Strongly related-3, Moderately related-2, Weakly related-1, Not related-0

COs	PO1	PO2	PO3	PO4	PO5
CO1	3	3	2	0	2
CO2	3	3	3	0	3
CO3	3	0	0	2	2
CO4	3	3	3	0	3
CO5	3	3	3	2	3

Course Coordinators:

HEAD of School (HOS)
School of Computer Science and Engineering
Dr. Ambedkar Institute of Technology
Bengaluru-560 056

22/10/2026

Dr. K R Shylaja
Professor and DEAN (R&D)
CSE Programme

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech Computer Science & Engineering

Semester: I **Scheme: 2026**

Course Title	**Data Science and Machine Learning Laboratory**				
Course Code	**1MSCSL107B**				
Course Category	**PCCL (Professional Core Course Laboratory)**				
Teaching - Learning Scheme & Credit	**L**	**T**	**P**	**Total Credits**	**Total Teaching Hours**
	0	**0**	**1**	**1**	**Lecture: 0 + Tutorials: 0 + Practical: 28 + Self Study: 02 = 30**
CIE Marks: 50	**SEE Marks: 50**		**Total Max. Marks: 100**		**Duration of SEE: 03 Hours**

Course Learning Objectives:	
CLO1	Provide hands-on experience in data preprocessing, analysis and visualization using Python
CLO2	Implement fundamental supervised and unsupervised machine learning algorithms
CLO3	Apply feature engineering and dimensionality reduction techniques to real-world datasets
CLO4	Evaluate machine learning models using appropriate performance measures
CLO5	Develop an end-to-end machine learning solution for a real-world problem

SYLLABUS	
PART: A - CONVENTIONAL EXPERIMENTS	
1.	**Data Preprocessing Using Python** Perform data preprocessing operations on the **Titanic Dataset**: ● Load and inspect the dataset. ● Handle missing values. ● Remove duplicate records. ● Detect and handle outliers. ● Encode categorical attributes. ● Perform feature scaling and normalization. **Suggested Dataset:** Titanic Dataset
2.	**Exploratory Data Analysis and Visualization** Perform exploratory data analysis on the **Iris Dataset**: ● Generate descriptive statistics. ● Analyze numerical attributes. ● Generate histograms, box plots and scatter plots. ● Generate a correlation matrix and heatmap. ● Identify relationships among features. **Suggested Dataset:** Iris Dataset
3.	**Linear Regression** Implement Simple and Multiple Linear Regression for predicting house prices.

	● Train-test split ● Model training ● Prediction ● MAE, MSE, RMSE and R^2 evaluation **Suggested Dataset:** California Housing Dataset
4.	**Logistic Regression** Implement Logistic Regression for binary classification and evaluate the model using Accuracy, Precision, Recall, F1-score and Confusion Matrix. **Suggested Dataset:** Breast Cancer Wisconsin Dataset
5.	**Decision Tree and Random Forest** Implement Decision Tree and Random Forest classifiers and compare their performance. **Suggested Dataset:** Wine Quality Dataset
6.	**SVM and KNN** Implement SVM and KNN classification and study the effect of different kernels and values of K. **Dataset:** Breast Cancer Wisconsin Dataset
7.	**K-Means Clustering** Implement K-Means clustering and determine the optimum number of clusters using the Elbow Method. **Suggested Dataset:** Mall Customer Segmentation Dataset
8.	**Principal Component Analysis** Apply PCA for dimensionality reduction, analyze explained variance and visualize the transformed dataset. **Suggested Dataset:** Wine Dataset
9	**Model Selection and Cross-Validation** Perform K-fold cross-validation and hyper parameter tuning using Grid Search/Random Search. Compare different machine learning models. **Suggested Dataset:** Breast Cancer Wisconsin Dataset
PART: B - TYPICAL OPEN-ENDED EXPERIMENTS	
1.	**Mini Project: End-to-End Machine Learning Application** Develop an end-to-end machine learning application involving: **Data Collection → Data Preprocessing → EDA → Feature Engineering → Model Selection → Model Training → Prediction → Evaluation → Visualization**

SELF STUDY COMPONENT	**02 Hours**
Case Study on **OPEN ENDED PROJECTS**	

Teaching and Learning Process (Innovative Delivery Methods)	
1.	Active Learning Techniques

2.	Problem-Based Learning (PBL)
3.	Team-Based Learning (TBL)
4.	Hands-On Experiments and Simulations

TEXT BOOKS	
1.	Aurélien Géron, **Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow**, O'Reilly Media
2.	Wes McKinney, **Python for Data Analysis**, O'Reilly Media.
3.	Andreas C. Müller and Sarah Guido, **Introduction to Machine Learning with Python**, O'Reilly Media

REFERENCE BOOKS	
1.	Christopher M. Bishop, **Pattern Recognition and Machine Learning**, Springer.
2.	Tom M. Mitchell, **Machine Learning**, McGraw-Hill
3.	Kevin P. Murphy, **Probabilistic Machine Learning: An Introduction**, MIT Press.
4.	Jake VanderPlas, **Python Data Science Handbook**, O'Reilly Media.

e-LEARNING RESOURCES
Google Data Analytics: https://grow.google/intl/en_in/data-analytics-course/
Elements of AI: https://course.elementsofai.com/?utm_source=chatgpt.com
https://www.coursera.org/browse/data-science/machine-learning
https://nptel.ac.in/courses/106106139 NPTEL Course IIT Madras
http://www.digimat.in/nptel/courses/video/106105152/L01.html

Assessment Details for CIE			
CONTINUOUS INTERNAL EVALUATION (CIE)		**Max. Marks**	**Minimum Marks Required in CIE to Qualify for SEE (50% in Each Component)**
Laboratory Components	**Each experiment will be evaluated using the prescribed rubrics, and the average of the marks obtained across all experiments will be considered for the final evaluation.**	**10**	**25**
	LAB CIE	**40**	
	TOTAL	**50**	

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech Computer Science & Engineering

Semester: I **Scheme: 2026**

Practical Examination Guidelines
● All laboratory experiments shall be included for the practical examination. ● The instructions and marks breakup printed on the cover page of the answer script shall be strictly adhered to by the examiners. ● Students shall be allowed to randomly select any one experiment from the given lot. ● Change of the selected experiment shall be permitted only once. ● Any change of experiment shall attract mark deduction as per the Dr. AIT Autonomous Examination Rules and Regulations

Course Outcomes: After the completion of the course, the student will be able to	
CO1	Apply Python-based tools for data preprocessing, exploratory analysis and visualization
CO2	Implement supervised machine learning algorithms for regression and classification problems.
CO3	Apply unsupervised learning and dimensionality reduction techniques to analyze datasets.
CO4	Evaluate and compare machine learning models using suitable performance metrics and validation techniques.
CO5	Develop an end-to-end machine learning solution for a real-world problem

COURSE SUSTAINABLE DEVELOPMENT GOALS (SDGs)		
Goal No.	**Goal Title**	**Description**
4	Quality Education	Ensure Inclusive and equitable quality education and promote lifelong learning opportunities for all
8	Decent Work and Economic Growth	Promote sustained, inclusive and sustainable economic growth, full and productive employment and decent work for all.

Mapping of COs with POs

Strength of correlation: Strongly related-3, Moderately related-2, Weakly related-1, Not related-0

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**
CO1	3	3	2	3	3
CO2	3	3	3	3	3
CO3	3	3	3	3	3
CO4	3	3	3	3	3
CO5	3	3	3	3	3

Course Coordinators:

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech Computer Science & Engineering

Semester: I **Scheme: 2026**

27/10/2026	27/10/26
Dr. K R Shylaja Professor and DEAN (R&D) CSE Programme	Dr. Asha K N Assistant Professor CSE Programme

HEAD of School (HOS)
School of Computer Science and Engineering
Dr. Ambedkar Institute of Technology
Bengaluru-560 056

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

Course Title	Advanced Algorithm and Data Structure Laboratory				
Course Code	1MSCSL107C				
Course Category	PCCL (Professional Core Course Laboratory)				
Teaching - Learning Scheme & Credit	L	T	P	Total Credits	Total Teaching Hours
	0	0	1	1	Lecture: 0 + Tutorials: 0 + Practical: 28 + Self Study: 02 = 30
CIE Marks: 50	SEE Marks: 50		Total Max. Marks: 100		Duration of SEE: 03 Hours

Course Learning Objectives:	
CLO1	Analyze the complexity of algorithms of real time problems to provide optimal solutions
CLO2	Provide solutions for graph based real time problems
CLO3	Build applications to using appropriate data structures to optimize the process
CLO4	Provide solutions for real time NLP based problems

SYLLABUS	
PART: A - CONVENTIONAL EXPERIMENTS	
1.	Write a program for implementing the following sorting methods: a) Merge sort b) Heap sort c) Quick sort Analyze the time complexity.
2.	Write code to find the shortest path in a directed weighted graph using Bellman ford algorithm
3.	Write a code to find a pattern in a given string using KMP algorithm
4.	Write a program to perform the following operations: a) Insert an element into a Min-Max heap b) Delete an element from a Min-Max heap c) Search for a key element in a Min-Max heap
5.	Write a program to perform the following operations: a) Insert an element into a Red-Black tree. b) Delete an element from a Red-Black tree. c) Search for a key element in a Red-Black tree
6.	Write a program to perform the following operations: a) Insert an element into a binomial heap b) Delete an element from a binomial heap. c) Search for a key element in a binomial heap
PART: B - TYPICAL OPEN-ENDED EXPERIMENTS	
1.	Design an algorithm to identify and report the vertices affected by a negative-weight cycle.
2.	Design an algorithm to generate plagiarism detection using KMP algorithm.
3.	Design a Red-Black Tree for a dictionary application where words are inserted, searched, updated, and deleted frequently
4.	Stock Monitoring System (Heap) Track minimum and maximum stock values dynamically

5.	Bucket Distribution System (Extendible Hashing) Observe record distribution across dynamically growing buckets

SELF STUDY COMPONENT	**02 Hours**
Case Study on OPEN ENDED PROJECTS	

Teaching and Learning Process (Innovative Delivery Methods)	
1.	Active Learning Techniques
2.	Problem-Based Learning (PBL)
3.	Team-Based Learning (TBL)
4.	Hands-On Experiments and Simulations

TEXT BOOKS	
1.	Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, Introduction to Algorithms, 3rd Edition, MIT Press.
2.	Fundamentals of Data structures in C, E. Horowitz, S. Sahni and Susan Anderson Freed, 2nd Edition, Universities Press

REFERENCE BOOKS	
1.	Peter Brass, Advanced Data Structures, 1st Edition, (2011), Cambridge University Press
2.	Mark Allen Weiss, Data Structures and Algorithm Analysis in C++, 4th Edition, (2014) Pearson Education.
3.	Data Structures Using C – A.S. Tanenbaum, Y. Langsam, and M.J. Augenstein, PHI/Pearson education
4.	Introduction to Data Structures in C, Ashok Kamthane, 1st Edition, Pearson

e-LEARNING RESOURCES
NPTEL – Advanced Data Structures: https://nptel.ac.in/courses/106102064
MIT OpenCourseWare – Advanced Data Structures: https://ocw.mit.edu/courses/6-851-advanced-data-structures-spring-2012/
GeeksforGeeks – Advanced Data Structures: https://www.geeksforgeeks.org/advanced-data-structures/
GeeksforGeeks – Heap, Trees, Hashing, DSU (C/C++ Implementations):
Programiz – Data Structures and Algorithms: https://www.programiz.com/dsa

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

Assessment Details for CIE			
CONTINUOUS INTERNAL EVALUATION (CIE)		**Max. Marks**	**Minimum Marks Required in CIE to Qualify for SEE (50% in Each Component)**
Laboratory Components	**Each experiment will be evaluated using the prescribed rubrics, and the average of the marks obtained across all experiments will be considered for the final evaluation.**	**10**	**25**
	LAB CIE	**40**	
	TOTAL	**50**	

Practical Examination Guidelines

- All laboratory experiments shall be included for the practical examination.
- The instructions and marks breakup printed on the cover page of the answer script shall be strictly adhered to by the examiners.
- Students shall be allowed to randomly select any one experiment from the given lot.
- Change of the selected experiment shall be permitted only once.
- Any change of experiment shall attract mark deduction as per the Dr. AIT Autonomous Examination Rules and Regulations

Course Outcomes: After the completion of the course, the student will be able to	
CO1	Analyze computational problems and select appropriate advanced data structures for developing efficient solutions to real-world problems
CO2	Implement and validate data operations using appropriate advanced data structures and programming techniques.
CO3	Apply suitable programming tools and development environments to implement, test, debug, and evaluate algorithms for real-world applications
CO4	Document the experimental methodology, design, implementation, results, and inferences in a structured technical report.

COURSE SUSTAINABLE DEVELOPMENT GOALS (SDGs)		
Goal No.	**Goal Title**	**Description**
4	Quality Education	Ensure Inclusive and equitable quality education and promote lifelong learning opportunities for all
8	Decent Work and Economic Growth	Promote sustained, inclusive and sustainable economic growth, full and productive employment and decent work for all.
9	Industry, Innovation and Infrastructure	Enables students to design efficient computational solutions and develop innovative software applications using advanced data structures and algorithms.

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

Computer Science and Engineering Programme

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

Mapping of COs with POs **Strength of correlation:** Strongly related-3, Moderately related-2, Weakly related-1, Not related-0					
COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**
CO1	3	3	2	0	2
CO2	3	3	3	0	3
CO3	3	2	3	2	2
CO4	3	3	3	2	3

Course Coordinators:

27/10/2026

Dr. K R Shylaja
Professor and DEAN (R&D)
CSE Programme

HEAD of School (HOS)
School of Computer Science and Engineering
Dr. Ambedkar Institute of Technology
Bengaluru-560 056

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

Course Title	Microservices Development Laboratory				
Course Code	1MSCSL107D				
Course Category	PCCL (Professional Core Course Laboratory)				
Teaching - Learning Scheme & Credit	L	T	P	Total Credits	Total Teaching Hours
	0	0	1	1	Lecture: 0 + Tutorials: 0 + Practical: 28 + Self Study: 02 = 30
CIE Marks: 50	SEE Marks: 50		Total Max. Marks: 100		Duration of SEE: 03 Hours

Course Learning Objectives:	
CLO1	Explain the architecture and fundamental components of microservices-based applications using the Spring Boot framework.
CLO2	Develop RESTful microservices with database persistence, input validation, and exception-handling mechanisms.
CLO3	Implement communication, service discovery, and API gateway-based routing among multiple microservices.
CLO4	Demonstrate the containerization, deployment, documentation, and testing of microservices using appropriate tools and technologies.
CLO5	Design distributed microservices applications to address real-world problems.

SYLLABUS

PART: A - CONVENTIONAL EXPERIMENTS

Development of a Product Management Microservice

Objective To develop a RESTful Product Management microservice by extending the reusable Spring Boot microservice framework with database persistence, validation, and exception handling. Description Extend the reusable Spring Boot microservice framework developed during the pre-requisite exercises to implement a Product Management microservice. Design the Product entity and corresponding repository, service, controller, and DTO classes following the layered architecture.

1. **Implement** RESTful CRUD operations for managing product information using Spring Data JPA and a relational database. Incorporate input validation and global exception handling to ensure reliable and maintainable service implementation.

Test the REST APIs using Postman and verify the correctness of all CRUD operations and error-handling mechanisms.

Students shall be able to

- Extend an existing Spring Boot microservice framework.

	● Design entity and DTO classes. ● Implement repository, service, and controller layers. ● Develop RESTful CRUD APIs. ● Configure database persistence using Spring Data JPA. ● Apply input validation. ● Implement global exception handling. ● Test REST APIs using Postman. **Deliverables** ● Product Management microservice. ● Database schema and entity classes. ● Functional CRUD REST APIs. ● Validated request handling. Tested API collection.
2.	**Development of a Customer Management Microservice** **Objective:** To develop an independent Customer Management microservice using the reusable Spring Boot framework and implement RESTful services for customer information management. **Description:** Develop a Customer Management microservice by extending the reusable framework created during the pre-requisite exercises. Design the Customer entity together with the corresponding repository, service, controller, and DTO components using the standard layered architecture. **Implement** RESTful CRUD APIs for customer management with persistent storage using Spring Data JPA. Apply validation and exception-handling mechanisms to ensure reliable processing of user requests. Verify all APIs using Postman and evaluate the correctness of the implemented microservice. **Students shall be able to** ● Develop an independent microservice. ● Design layered application components. ● Implement CRUD services. ● Manage persistent customer data. ● Apply validation techniques.

	● Handle application exceptions. ● Test REST APIs. **Deliverables** ● Customer Management microservice. ● Functional REST APIs. ● Persistent customer database. ● Validated API responses. ● Tested Postman collection.
3.	**Development of Communicating Microservices** **Objective**: To develop two independent microservices and implement synchronous inter-service communication using REST-based service invocation mechanisms. **Description**: Develop two lightweight microservices by extending the reusable Spring Boot framework. Design suitable REST APIs for each microservice and implement synchronous communication using Spring REST Client or OpenFeign to exchange information between them. Verify the correctness of service interactions by invoking APIs and exchanging JSON data. **Implement** appropriate exception handling to manage communication failures and validate the responses using Postman. **Students shall be able to** ● Develop multiple independent microservices. ● Implement synchronous inter-service communication. ● Exchange data between services. ● Handle communication exceptions. ● Test service interactions using Postman. **Deliverables** ● Two independent microservices. ● Functional REST-based service communication. ● Successful data exchange between services. ● Tested Postman collection.
4.	**Development of Service Discovery and API Gateway**

	Objective: To configure service registration, service discovery, and centralized API routing in a distributed microservices environment. Description: Develop two simple microservices using the reusable framework and configure them to register with a Service Discovery Server. Configure an API Gateway to route client requests dynamically to the registered services. **Verify** successful service registration, service discovery, and gateway-based request routing. Test different request scenarios using Postman and evaluate the behaviour of the distributed application. **Students shall be able to** ● Configure a Service Discovery Server. ● Register microservices dynamically. ● Configure API Gateway routing. ● Route client requests through the gateway. ● Verify service registration and discovery. **Deliverables** ● Service Discovery Server. ● Registered microservices. ● Configured API Gateway. ● Successful gateway-based routing.
5.	**Containerization and Deployment of Microservices.** **Objective**: To containerize and deploy multiple microservices using Docker and Docker Compose. **Description:** Develop one or more microservices using the reusable framework and containerize them by creating Docker images and Dockerfiles. Configure Docker Compose to deploy the microservices together with the required database services. **Verify** successful deployment, container networking, and application execution in the containerized environment. Evaluate the portability and deployment characteristics of the application. **Students shall be able to** ● Create Dockerfiles. Build Docker images. ● Configure Docker Compose.

	● Deploy multiple containers. ● Verify container networking and execution. **Deliverables** ● Dockerfiles. ● Docker images. ● Docker Compose configuration. ● Running containerized microservices.
6.	**API Documentation, Functional Testing, and Performance Evaluation.** **Objective**: To document, test, and evaluate RESTful microservices using API documentation and software testing tools. **Description**: Develop a microservice using the reusable framework and generate API documentation using Swagger/OpenAPI. Perform functional testing using Postman by executing various request scenarios and validating the correctness of API responses. **Conduct** basic performance evaluation using JMeter or an equivalent testing tool by measuring response time and throughput under different workloads. Analyse the observed performance characteristics and document the test results. **Students shall be able to** ● Generate API documentation. ● Perform functional API testing. ● Design API test cases. ● Conduct basic performance evaluation. ● Analyse application performance. **Deliverables** ● Swagger/OpenAPI documentation. ● Functional API test cases. ● Performance testing report. ● Performance analysis results.

PART: B - TYPICAL OPEN-ENDED EXPERIMENTS

Open-ended experiments are laboratory activities in which the outcome is not predetermined. Students are given the freedom to explore, design, and conduct experiments based on problem

statements related to the concepts defined by the Course Coordinator. Such experiments encourage creativity, critical thinking, problem-solving, and inquiry-based learning	
1.	**Design and Development of an E-Commerce Microservices Application** **Objective:** To design and develop a distributed e-commerce application using multiple microservices and contemporary microservices technologies. **Description**: Design and develop an e-commerce application by identifying suitable business services such as Product Management, Customer Management, Order Management, Inventory Management, and Payment Processing. Develop the required microservices using the reusable Spring Boot framework and implement REST-based communication among them. **Integrate** service discovery, API gateway, database persistence, and containerized deployment. Test the application using Postman and generate API documentation to validate the correctness of the distributed system. **Students shall be able to** ● Design a distributed microservices architecture. ● Identify service boundaries. ● Develop multiple collaborating microservices. ● Integrate service discovery and API gateway. ● Deploy and validate the complete application. **Deliverables** ● Complete e-commerce microservices application. ● Integrated distributed services. ● API documentation. ● Containerized deployment. ● Functional testing report.
2.	**Design and Development of a Smart Healthcare Management System** **Objective**: To design and develop a distributed healthcare management application using multiple collaborating microservices. **Description:** Design and develop a healthcare management system by identifying services such as Patient Management, Doctor Management, Appointment Scheduling, Billing, and Medical Records. Develop the required microservices using the reusable framework and implement communication among services.

	Configure service discovery, API gateway, database persistence, and deployment using Docker Compose. Test and document the developed application. **Students shall be able to** ● Model healthcare services as microservices. ● Develop integrated service-oriented applications. ● Configure distributed application infrastructure. ● Test and document the developed solution. ● Deliverables ● Healthcare microservices application. ● Integrated services. ● API documentation. ● Deployment configuration. ● Test report.
3.	**Design and Development of a Secure Microservices Application** **Objective**: To design and develop a secure microservices application by implementing authentication and authorization mechanisms. **Description**: Design and develop a distributed microservices application by integrating authentication and authorization using Spring Security and JSON Web Token (JWT) or equivalent security mechanisms. Protect REST APIs, manage user authentication, and authorize access to secured resources. **Test** secured APIs using Postman and verify authentication, authorization, and protected resource access under different user scenarios. **Students shall be able to** ● Design secure RESTful services. ● Implement authentication mechanisms. ● Configure authorization rules. ● Secure distributed microservices. ● Test secured APIs. Deliverables ● Secure microservices application.

	● Authentication and authorization implementation. ● Protected REST APIs. ● Security testing report.
4.	Design and Development of a Cloud-Native Microservices Solution **Objective**: To design, develop, deploy, and evaluate a complete cloud-native microservices solution for a real-world application. **Description:** Identify a suitable real-world problem and design a cloud-native microservices architecture by decomposing the application into independent services. Develop the required microservices using the reusable framework and integrate service communication, service discovery, API gateway, containerization, API documentation, and testing. **Deploy** the complete application using Docker Compose, perform functional testing, and evaluate the overall behaviour, scalability, and maintainability of the developed solution. **Students shall be able to** ● Design complete cloud-native applications. ● Develop integrated distributed systems. ● Apply microservices architectural principles. ● Deploy containerized applications. ● Evaluate the developed solution. ● Deliverables ● Complete cloud-native application. ● Distributed microservices architecture. ● Containerized deployment. ● API documentation. ● Functional testing report. ● Project report.
5.	**Design and Development of a Smart Campus Microservices Application** **Objective**: To design and develop a distributed smart-campus application using independently deployable microservices. **Description**: Design and develop a smart-campus system by identifying appropriate services such as Student Management, Attendance, Classroom/Laboratory Management,

	Asset Management, Notification, and Maintenance Request services. Develop suitable microservices and integrate them using REST-based or event-driven communication. Incorporate database persistence, service discovery, API gateway, security where appropriate, and containerized deployment. Test and document the complete system. **Students shall be able to**: ● Identify microservice boundaries from a real-world problem. ● Design collaborating services. ● Integrate multiple microservices ● Configure distributed infrastructure. ● Deploy and validate a complete application. **Deliverables:** ● Smart-campus microservices application ● Architecture diagram ● Integrated services ● API documentation ● Containerized deployment ● Testing report.
6	**Design and Development of a Library Management Microservices Application** **Objective:** To design and develop a simple Library Management application using multiple microservices. **Description**: Design and develop a Library Management application consisting of suitable services such as Book Management, Member Management, and Issue/Return Management. Develop the required microservices using Spring Boot and implement REST-based communication among them. Configure database persistence and test the integrated application using Postman. **Students shall be able to**: ● Identify services in a simple application. ● Develop independent Spring Boot microservices. Implement REST APIs for different ● services. Establish communication between microservices. Store and retrieve ● application data. Test the integrated application using Postman.

Semester: I **Scheme: 2026**

	Deliverables: ● Library Management microservices application. ● Minimum three functional microservices. ● REST APIs for the identified services. ● Database configuration. ● Postman test collection. ● Brief architecture diagram.

SELF STUDY COMPONENT	**02 Hours**
Students shall study the fundamental principles of microservices architecture, including service decomposition, independent deployment, scalability, and maintainability. They shall explore the Spring Boot framework for developing RESTful services and review the role of API gateways, service discovery, and containerization in distributed applications.	

Teaching and Learning Process (Innovative Delivery Methods)	
1.	Active Learning Techniques
2.	Problem-Based Learning (PBL)
3.	Team-Based Learning (TBL)
4.	Hands-On Experiments and Simulations

TEXT BOOKS	
1.	Chris Richardson, Microservices Patterns: With Examples in Java, Manning Publications, 2018.
2.	Sam Newman, Building Microservices: Designing Fine-Grained Systems, 2nd Edition, O'Reilly Media, 2021.
3.	Laurentiu Spilca, Spring Start Here: Learn What You Need and Learn It Well, Manning Publications, 2021.

REFERENCE BOOKS	
1.	Craig Walls, Spring Boot in Action, Manning Publications, 2016.
2.	Mark Richards and Neal Ford, Fundamentals of Software Architecture: An Engineering Approach, O'Reilly Media, 2020.
3.	Eberhard Wolff, Microservices: Flexible Software Architecture, Addison-Wesley Professional, 2017.
4.	Brendan Burns, Designing Distributed Systems: Patterns and Paradigms for Scalable, Reliable Services, O'Reilly Media, 2018.

5.	Spring Boot Reference Documentation (Official).
6.	Spring Cloud Reference Documentation (Official).
7.	Docker Documentation (Official).

e-LEARNING RESOURCES
https://docs.spring.io/spring-boot/documentation.html
https://spring.io/projects/spring-boot/
https://spring.io/guides/gs/spring-boot-docker/
https://docs.docker.com/guides/java/
https://docs.docker.com/reference/samples/spring/
https://spring.io/projects/spring-cloud
https://start.spring.io/
https://learning.postman.com/
https://jmeter.apache.org/usermanual/index.html
https://www.openapis.org

<table>
<tr><th colspan="4">Assessment Details for CIE</th></tr>
<tr><th colspan="2">CONTINUOUS INTERNAL EVALUATION (CIE)</th><th>Max. Marks</th><th>Minimum Marks Required in CIE to Qualify for SEE (40% in Each Component)</th></tr>
<tr><td rowspan="2">Laboratory Components</td><td>Each experiment will be evaluated using the prescribed rubrics, and the average of the marks obtained across all experiments will be considered for the final evaluation.</td><td>10</td><td rowspan="3">25</td></tr>
<tr><td>LAB CIE</td><td>40</td></tr>
<tr><td colspan="2">TOTAL</td><td>50</td></tr>
<tr><th colspan="4">Practical Examination Guidelines</th></tr>
<tr><td colspan="4">

All laboratory experiments shall be included for the practical examination.
The instructions and marks breakup printed on the cover page of the answer script shall be strictly adhered to by the examiners.
Students shall be allowed to randomly select any one experiment from the given lot.
Change of the selected experiment shall be permitted only once.
Any change of experiment shall attract mark deduction as per the Dr. AIT Autonomous Examination Rules and Regulations

</td></tr>
</table>

Dr. Ambedkar Institute of Technology

School of Computer Science and Engineering

M.Tech in Computer Science & Engineering

Semester: I **Scheme: 2026**

Course Outcomes: After the completion of the course, the student will be able to	
CO1	Explain the architecture and components of a Spring Boot microservice application and configure the development environment.
CO2	Develop RESTful microservices with database connectivity, validation, and exception handling.
CO3	Implement communication, service discovery, and request routing among microservices.
CO4	Deploy, document, and test microservices using containerization and API testing tools.
CO5	Develop a simple distributed microservices application by integrating multiple microservices.

COURSE SUSTAINABLE DEVELOPMENT GOALS (SDGs)		
Goal No.	**Goal Title**	**Description**
4	Quality Education	Ensure Inclusive and equitable quality education and promote lifelong learning opportunities for all
8	Decent Work and Economic Growth	Promote sustained, inclusive and sustainable economic growth, full and productive employment and decent work for all.

Mapping of COs with POs

Strength of correlation: Strongly related-3, Moderately related-2, Weakly related-1, Not related-0

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**
CO1	2	1	-	-	2
CO2	3	2	3	1	3
CO3	2	2	3	1	3
CO4	2	2	2	3	3
CO5	3	3	3	2	3

Course Coordinators:

Dr. Gowrishankar S Professor & HOS CSE School	Dr. Srinivasa A H Associate Professor CSE Programme

HEAD of School (HOS)
School of Computer Science and Engineering
Dr. Ambedkar Institute of Technology
Bengaluru-560 056