BMS **INSTITUTE OF TECHNOLOGY AND MANAGEMENT**

(An Autonomous institution affiliated to VTU, Belagavi)

Doddaballapur Main Road, Avalahalli, Yelahanka, Bengaluru 560064

M. Tech – Computer Science and Engineering

Scheme and Syllabus 2024

Year: 2024 - 2025

Institute Vision

To emerge as one of the finest technical institutions of higher learning, to develop engineering professionals who are technically competent, ethical and environment friendly for betterment of the society.

Institute Mission

Accomplish stimulating learning environment through high quality academic instruction, innovation and industry-institute interface.

Department of Computer Science and Engineering

VISION

To develop technical professionals acquainted with recent trends and technologies of computer science to serve as valuable resources for the nation/society.

MISSION

Facilitating and exposing the students to various learning opportunities through dedicated academic teaching, guidance and monitoring.

M. Tech in Computer Science and Engineering

Program Educational Objectives (PEOs)

PEO1	Apply analytical thinking to solve problems through research and development in the areas of Computer Science and Engineering and allied engineering domains.
PEO2	Adapt to changing technological trends through life-long learning by exh professional ethics, integrity and career growth.
PEO3	Develop skills to facilitate in providing sustainable solutions by addressing the growing challenges of the society.

Program Outcomes (POs)

PO1	Independently carry out research and development work to solve practical problems related to Computer Science and Engineering and allied engineering domains.
PO2	Write and present a substantial technical report/document.
PO3	Demonstrate a degree of mastery over the area as per the specialization of the program. The mastery should be at a level higher than the requirements in the appropriate bachelor program.
PO4	Analyze the acquired domain knowledge for providing feasible security solution(s).
PO5	Relate the learning outcomes to build requisite competency in professional environment.
PO6	Appraise the need for engaging in lifelong learning.

About the Department

The Department of Computer Science & Engineering was started in the year 2002 – 2003 with an intake of 60, the Year 2010 with an intake of 90 and 180 in the year 2017 – 2018. Department intake got enhanced to 240 in the year 2023 – 24, currently the intake is 900 for the academic year 2024 - 25. The Department is highly progressive and has a team of well qualified, experienced and dedicated faculty. 47 faculty members hold Doctoral degrees, and 22 Faculty members are currently pursuing their Doctorate degrees. The Department of Computer Science & Engineering continues to recruit faculty members with high experience in academics, industry, and research.

The Department has well-equipped computer laboratories for course work, teaching and to carry out projects. The servers and nodes are all connected in the network with all necessary licensed software. Exposure to cutting edge technologies is provided by means of Industrial Projects and technical talks from domain experts of reputed research organizations. The students are encouraged to involve themselves in creative, technical and research activities. Students have been performing well in the university examination.

The Department maintains a wireless network (Wi-Fi) with unlimited Internet access for use by staff and students. Our graduates are working in leading IT industries and many students have secured admissions to prestigious universities in India and abroad. The Department has an R&D Center as well as an Incubation Center which is facilitating the students to acquire practical knowledge. The Department also offers an M.Tech. Program (post-graduation) in Computer Science & Engineering, which was introduced in the year 2014 with an approved intake of 18.

About M.Tech in Computer Science and Engineering

M.Tech (Computer Science and Engineering) commenced in the year 2014 with an intake of 18 students. The Post Graduate Program in Computer Science & Engineering is an affiliated program offered by Visvesvaraya Technological University (VTU), Belagavi. The curriculum is designed by the university and has been common across all the institutions affiliated to it. Highly experienced faculty members with doctoral degrees handle the courses for this program. The program is accredited by National Board of

Accreditation (NBA) till Jun 2025. This is only PG program in the state of Karnataka to have got accreditation in the first attempt under Tier II. Students and faculty members are proactively involved in high end research activities and have published impetus research publications in domains of Computer Network, Network Security, AI and Data Science. Students undergo 8 weeks industrial internship in many reputed companies like Nokia, Siemens, Robert Bosch, Phillips and many more. Many students are aspirant of the higher studies (Ph.D.) in various domains

PREAMBLE

In keeping abreast with India's recent National Education Policy (NEP 2020), the Indian Institute of Science, Bengaluru, has designed the Master of Technology (Online) degree program, for practicing engineers and scientists. Towards the attainment of such a holistic and multidisciplinary education, the flexible and innovative curriculum has been provided at BMSIT&M with credit-based courses and projects/internships/special courses in the areas of community engagement and public service, environmental education, and value-based education.

The emphasis is more on the core competency in the curriculum of the program to enhance opportunities for placement through industry relevant courses as program core and program electives. This is effectively attained with proper design, operation and improvement in academic components inthe system with inclusive focus on Modern teaching methods, advanced curricula, innovative assessment methods, research temperament, industry associated curriculum. Implementation of academic autonomy can is with supportive governance and administrative structure is properly planned and put in place.

Curricular inputs for the framework are from all the stakeholders involved in the academic process and referring curriculum from standard and well- known universities/colleges. Input for the framework is also from Professional bodies like IEEE and CSI which recommends the advanced courses for the PG program of 2 years. The expected learning outcomes of autonomous curriculum of BMSIT&M cater to the aspiration of learner in- terms of higher education, research, industry requirements. Develop learner’s inquisitiveness and focus on research and development of disruptive technologies. Incorporation of ICT tools imperatively blended in the autonomous curricula reaching all class of learners.

With this preamble, the curriculum for the autonomous BMSIT&M has been designed to meet the contemporary needs (aspirations) of primary stake holders (students) with the following.

Salient features

1. **Inclusion of NEP 2020:** The aspiration of NEP 2020 and various levels has been incorporated in the M.Tech Computer Science Engineering (Autonomous) with inclusive focus on practical work, industrial internship, emphasis on research to solve the societal issues and latest trends as courses.
2. **Induction Programme:** There will be a week-long induction program for the PG students entering the institution. The incumbents learn about the institutional policies, processes, practices, culture and values.
3. **Post Graduate Program Outcomes (PO) Based Curriculum:** The curricula for the program is designed to meet the post graduate attributes (Program Outcomes) defined by National Board of Accreditation which are based on the knowledge, research, skill, ethics and higher learning.
4. **Emphasis on Research Project Based Learning:** To impart the skills to the prospective researcher, the emphasis on practical sessions is extended in the curricula for all the programs. At each semester, the adequate number of practical/laboratory courses is included. Further, some of the theory courses are blended with practical as integrated course.

5. **Industry exposure through Lab work, Mini projects and Internships:** The curricula include industry internships and mini projects for the students to expose them to the real-world experience at industrial environment. Mini projects expose to better technical articulation and project cycles.
6. **Self-Learning:** The curriculum provides with an opportunity for the students to take the initiative, with or without the assistance of others, in diagnosing their learning needs, formulating learning goals, identifying human and material resources for learning, and evaluating learning outcomes.
7. **Multiple avenues based on aspirations of the students**: The students will study the program specific courses for two years. There are three major avenues for the aspiring students to pursue:
    - **Industry/Placement:** The students who are aspiring to work as professional engineers in their core industrial domain have the option of studying the courses in the curriculum which are aligned towards the placement opportunities.
    - **Research:** The curriculum provides an opportunity for the students to pursue the courses which are in support of higher learning enabling the learner to do research work in the desired domain of interest.
    - **Presentation and Articulation:** The curriculum provides opportunities to present flexible assessment method for the course which improves communication and expect document this as report.

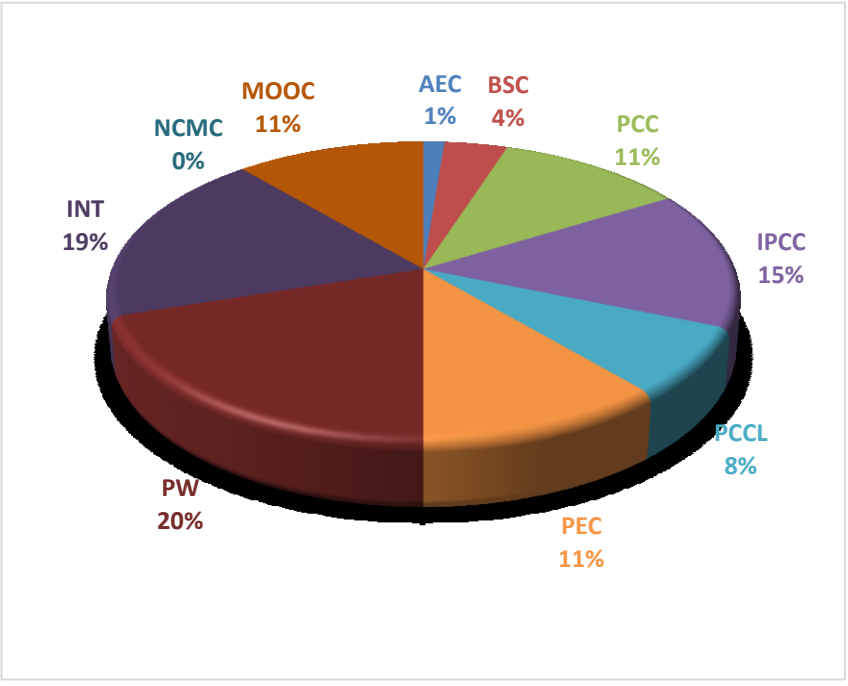
Credit Distribution of M. Tech Computer Science and Engineering

(Autonomous-2024)

SEM	AEC	BSC	PCC	IPCC	PCCL	PEC	MDC	PW	PBLC	INT	NCMC	MOOC	Total Credits
I		3	9	4	4						PP		20
II	1			8	2	9							20
III								4		5		9	18
IV								12		10			22
Total	**1**	**3**	**9**	**12**	**6**	**9**		**16**		**15**		**9**	**80**

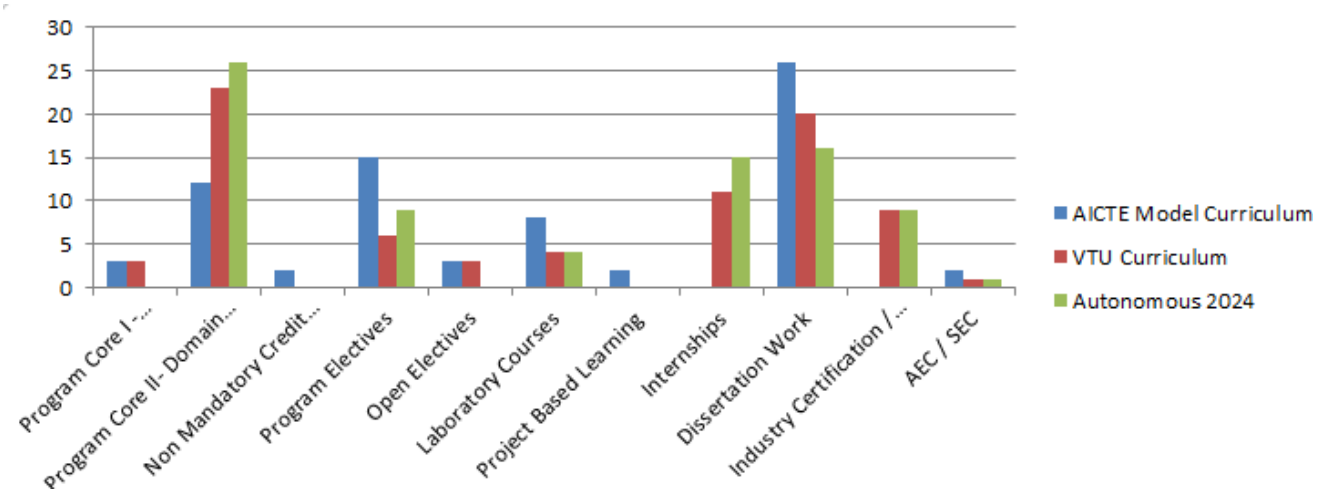
AEC/SEC	Ability/Skill Enhancement Course	**BSC**	Basic Science Course
PCC	Professional Core Course	**IPCC**	Integrated Professional Core Course
PCCL	Professional Core Course Lab	**PEC**	Professional Elective Course
MDC	Multi-Disciplinary Course	**PW**	Project Work
PBLC	Project Based Learning Course	**INT**	Internship
NCMC	None Credit Mandatory Course	**MOOC**	Massive Online Open Course

CREDIT DISTRIBUTION M. TECH - CSE



Statistical Comparison Curriculum Components with AICTE and VTU

Sl. No	Curriculum Components	AICTE Model Curriculum	VTU Curriculum	Autonomous 2024
1	Program Core I - Mathematics	3	3	-
2	Program Core II- Domain Specific	12	23	26
3	Non-Mandatory Credit Courses	2	-	PP
4	Program Electives	15	6	9
5	Open Electives	3	3	-
6	Laboratory Courses	8	4	4
7	Project Based Learning	2	-	-
8	Internships	-	11	15
9	Dissertation Work	26	20	16
10	Industry Certification / MOOC	-	9	9
11	AEC / SEC	2	1	1



Inclusion the autonomous curriculum is at par and above the standard prescribed

BMS INSTITUTE OF TECHNOLOGY AND MANAGEMENT

(Autonomous Institution Affiliated to VTU, Belagavi)
Scheme of Teaching and Examinations – 2024
Scheme
Outcome-Based Education (OBE)and Choice Based Credit System (CBCS)
(Effective from the academic year 2024 - 25 onwards)

I Semester M Tech COMPUTER SCIENCE AND ENGINEERING

Sl. No.	Course Category	Course Code	Course Name	Teaching Department	Credits Distribution				Examination				Contact Hours/week
					L	T	P	Total	CIE Marks	SEE Marks	Total Marks	SEE Duration (H)	
1	BSC	24MAT11	Applied Mathematics	MT	2	2	0	3	50	50	100	3	4
2	IPCC	24MCS12	Advanced Algorithms	CS	3	0	2	4	50	50	100	3	5
3	PCC	24MCS13	Artificial Intelligence	CS	3	0	0	3	50	50	100	3	3
4	PCC	24MCS14	Fundamentals of Data Science	CS	3	0	0	3	50	50	100	3	3
5	PCC	24MCS15	Cryptography and Network security	CS	2	2	0	3	50	50	100	3	4
6	PCCL	24MCSL16	Artificial Intelligence Laboratory	CS	0	1	2	2	50	50	100	3	4
7	PCCL	24MCSL17	No SQL Database Laboratory	CS	0	1	2	2	50	50	100	3	4
8	NCMC	24MRMI18	Research Methodology and IPR	-	-	-	-	PP	Online courses (online.vtu.ac.in)				
TOTAL					**13**	**4**	**3**	**20**	**350**	**350**	**700**		

Note: **BSC**-Basic Science Courses, **PCC**: Professional core. **IPCC**-Integrated Professional Core Courses, **PCC(PB):** Professional Core Courses (Project Based), **PCCL**-Professional Core Course lab **, NCMC**- None Credit Mandatory Course, ,**L**-Lecture**, P**-Practical**, T/SDA**-Tutorial / Skill Development Activities (Hours are for Interaction between faculty and students)
MRMI19- Research Methodology and IPR (**Online**) for the students who have **not studied** this course in the Undergraduate level. This course is not counted for vertical progression, Students have to qualify for the award of the master's degree.

M- Master program **xx** – **ME** for Mechanical Engineering Stream, **CV** for Civil Engineering Stream, **EE** – Electrical & Electronics Engineering Stream, **EC**-Electronics and Communication Engineering Stream, **CS**- Computer Science and Engineering **BA** - Business Administration **AR**- Architecture- etc.

BSC: Basic Science Courses: Courses like Mathematics/ Science are the prerequisite courses that the concerned engineering stream board of Studies will decide. **PCC: Professional Core Course:** Courses related to the stream of engineering, which will have both CIE and SEE components, students have to qualify in the course for the award of the degree. **Integrated Professional Core Course (IPCC)**: Refers to a Professional Theory Core Course Integrated with practical of the same course. The IPCC's theory part shall be evaluated by CIE and SEE. The practical part shall be evaluated by only CIE (no SEE). However, questions from the practical part of IPCC shall be included in the SEE question paper. **Project Based Learning Course (PCC(PB):** Project Based Learning course is a professional core Course only Students have to complete a project out of learning from the course and SEE will be viva voce on project work. **PCCL: Professional Core Course Laboratory:** Practical courses whose CIE will be evaluated by the class teacher and SEE will be evaluated by the two examiners.

Skill development activities: Under Skill development activities in a concerning course, the students should

1. Interact with industry (small, medium, and large).
2. Involve in research/testing/projects to understand their problems and help creative and innovative methods to solve the problem.
3. Involve in case studies and field visits/ fieldwork.
4. Accustom to the use of standards/codes etc., to narrow the gap between academia and industry.
5. Handle advanced instruments to enhance technical talent.
6. Gain confidence in the modelling of systems and algorithms for transient and steady-state operations, thermal study, etc. Work on different software/s (tools) to simulate, analyze and authenticate the output to interpret and conclude.

All activities should enhance student's abilities to employment and/or self-employment opportunities, management skills, Statistical analysis, fiscal expertise, etc. Students and the course instructor/s are to be involved either individually or in groups to interact together to enhance the learning and application skills of the study they have undertaken. The students with the help of the course teacher can take up relevant technical –activities that will enhance their skills. The prepared report shall be evaluated for CIE marks.

MRMI19-Research Methodology and IPR- None Credit Mandatory Course (NCMC) if students have not studied this course in their undergraduate program then he /she has to take this course at **http://online.vtu.ac.in** and to qualify for this course is compulsory before completion of the minimum duration of the program (Two years), however, this course will not be considered for vertical progression.

BMS INSTITUTE OF TECHNOLOGY AND MANAGEMENT

(Autonomous Institution Affiliated to VTU, Belagavi)

Scheme of Teaching and Examinations – 2024

Scheme

Outcome-Based Education (OBE) and Choice Based Credit System (CBCS)

(Effective from the academic year 2024 - 25 onwards)

II Semester M Tech Computer Science and Engineering

Sl. No.	Course Category	Course Code	Course Name	Teaching Department	Credits Distribution				Examination				Contact Hours/week
					L	T	P	Tot al	CIE Marks	SEE Marks	Total Marks	SEE Duration (H)	
1	IPCC	24MCS21	Machine Learning	CS	3	0	2	4	50	50	100	3	3
2	IPCC	24MCS22	Internet of Things	CS	3	0	2	4	50	50	100	3	4
3	PEC	24MCS23X	Specialization Course-I	CS	3	0	0	3	50	50	100	3	4
4	PEC	24MCS24X	Specialization Course-II	CS	3	0	0	3	50	50	100	3	4
5	PEC	24MCS25X	Specialization Course-III	CS	3	0	0	3	50	50	100	3	3
6	PCCL	24MCSL26	Web Applications Development Laboratory	CS	0	1	2	2	50	50	100	3	4
7	AEC/SEC	24MCS27X	Ability/Skill Enhancement Course (Offline/Online)	CS	0	0	2	1	50	50	100	3	2
TOTAL					**15**	**0**	**8**	**20**	**350**	**350**	**700**	-	-

Note: **PCC**: Professional core. **IPCC**-Integrated Professional Core Courses, **PCC(PB):** Professional Core Courses (Project Based), **PCCL**-Professional Core Course lab**,PEC**- Professional Elective Courses, **MDC-** Multi-Disciplinary Courses

, **L-Lecture, P-Practical, T/SDA-Tutorial / Skill Development Activities** (Hours are for Interaction between faculty and students)

L-Lecture, P-Practical, T/SDA-Tutorial / Skill Development Activities (Hours are for Interaction between faculty and students) **PBLC**: Project Based Learning Course,

Note: **xxx** means specialization code for example **MDE- Design** Engineering, **LDN-** Digital Communication and Networking, **SCE**- Computer Engineering, **CCT**- Construction Technology, **AUD**- Urban Design, **MBA**- Master of Business Administration, **MCA**-Master of Computer Application, etc

Ability / Skill Enhancement Courses					
Course Code	**Course title**	L	T/SDA	P	
24MCS27A	Mobile Application Development Laboratory	0	0	2	
24MCS27B	Requirement Analysis and Software Testing Tools	0	0	2	
24MCS27C	GIT for DEVOPS	0	0	2	
24MCS27D	Introduction to Kafka	0	0	2	
24MCS27E	Kubernetes and Docker Laboratory	0	0	2	

Ability Enhancement Courses (AEC):. These courses are designed to help students enhance their skills in communication, language, and personality development. They also promote a deeper understanding of subjects like social sciences and ethics, culture and human behavior, human rights, and the law.
Skill Enhancement Course (SEC): Skill Enhancement Course means a course designed to provide value-based or skill-based knowledge and should contain both theory and lab/hands- on/training/fieldwork. The main purpose of these courses is to provide students with life skills in the hands-on mode to increase their employability.
If AEC/SEC courses are ONLINE (MOOCs) courses suggested by the concerned board of studies. These courses will be made available on **www. online.vtu.ac.in**, however online courses are not considered for vertical progression, but qualifying in online courses is mandatory for the award of the degree.

Specialization Basket 1			
Course Code	**Course Title**	**Course Code**	**Course Title**
24MCS23A	Probability, Statistics and Queuing Theory	24MCS23B	Statistical Learning and Data Mining
24MCS23C	Queuing Theory in Network Communications	24MCS23D	Graph Algorithms and Mining
24MCS23E	Cyber Security and Forensics	24MCS23F	Secure Cloud Computing
24MCS23G	Blockchain Technologies	24MCS23H	Secure Software Development

Specialization Basket 2			
Course Code	**Course Title**	**Course Code**	**Course Title**
24MCS24A	Augmented and Virtual Reality	24MCS24B	Spatial Computing and Mixed Reality
24MCS24C	3D Modeling and Animation for AR/VR	24MCS24D	Human-Computer Interaction
24MCS24E	Natural Language Processing	24MCS24F	Web Based Information Retrieval
24MCS24G	Semantic Web	24MCS24H	Generative AI and Prompt Engineering

Specialization Basket 3			
Course Code	**Course Title**	**Course Code**	**Course Title**
24MCS25A	Agile Technology	24MCS25B	Web Engineering
24MCS25C	Object Oriented Analysis and Design	24MCS25D	Program Management

BMS INSTITUTE OF TECHNOLOGY AND MANAGEMENT

(Autonomous Institution Affiliated to VTU, Belagavi)
Scheme of Teaching and Examinations – 2024
Scheme
Outcome-Based Education (OBE)and Choice Based Credit System (CBCS)
(Effective from the academic year 2024 - 25 onwards)

III Semester M. Tech COMPUTER SCIENCE AND ENGINEERING

Sl. No.	Course Category	Course Code	Course Name	Teaching Department	Credits Distribution				Examination			
					L	T	P	Total	CIE Marks	SEE Marks	Total Marks	SEE Duration (H)
1	PCC	24MCS31	Online Course (12 weeks duration)	CS	3	0	0	3	25	75	100	-
2	PCC	24MCS32	Online Course (12 weeks duration)	CS	3	0	0	3	25	75	100	-
3	PCC	24MCS33	Online Course (12 weeks duration)	CS	3	0	0	3	25	75	100	-
4	INT	24MINT34	Internship Phase I (Research/Industry Internship leading to Project work)	CS	0	0	10	5	100	---	100	-
5	PW	24MPROJ35	Project Phase I	CS	0	0	8	4	100	---	100	-
TOTAL					**9**	**0**	**18**	**18**	**275**	**225**	**500**	

- **Professional Core Course (PCC):** It is an Online course and been proctored by the faculty throughout the semester. Students can choose the online course/certification programs from NPTEL (The courses are identified by BOS). They may present the final certificate for internal assessment. Student can take the course during II semester break and need to submit the course completion certificate before semester end examination for evaluation.
- **Project Work Phase-1:** Students in consultation with the guide/co-guide if any shall pursue literature survey and complete the preliminary requirements of selected Project work. Each student shall prepare relevant introductory project document and present a seminar. CIE marks shall be awarded by a committee comprising of HoD as Chairman, Guide/co-guide if any, and a senior faculty of the department. The CIE marks awarded for project work phase -1 shall be based on the evaluation of Project Report, Project Presentation skill and Question and Answer session in the ratio 50:25:25.
- **Internship Phase-1:** All the students shall have to undergo mandatory internship of 4-6 weeks during the semester. Those, who have not pursued /completed the internship, shall be declared as fail in internship course and have to complete the same during subsequent semester end examinations after satisfying the internship requirements.

BMS INSTITUTE OF TECHNOLOGY AND MANAGEMENT

(Autonomous Institution Affiliated to VTU, Belagavi)
Scheme of Teaching and Examinations – 2024
Scheme
Outcome-Based Education (OBE)and Choice Based Credit System (CBCS)
(Effective from the academic year 2024 - 25 onwards)

IV Semester M Tech COMPUTER SCIENCE AND ENGINEERING

Sl. No.	Course Category	Course Code	Course Name	Teaching Department	Credits Distribution				Examination				Contact Hours
					L	T	P	Total	CIE Marks	SEE Marks	Total Marks	SEE Duration	
1	INT	24MINT41	Internship Phase II (Research/Industry Internship leading to Project work)	CS	0	0	20	10	100	100	200	3	20
2	PW	24MPROJ42	Project Phase II	CS	0	0	24	12	100	100	200	3	24
TOTAL					**0**	**0**	**44**	**22**	**200**	**200**	**400**	**-**	**44**

- **Internship Phase-2:** All the students shall have to undergo mandatory internship of 6-8 weeks during the semester. Those, who have not pursued /completed the internship, shall be declared as fail in internship course and have to complete the same during subsequent semester end examinations after satisfying the internship requirements.
- **Project Work Phase-2:** CIE marks shall be awarded by a committee comprising of HoD as Chairman, Guide/co-guide, if any, and a senior faculty of the department. The CIE marks awarded for project work phase -2 shall be based on the evaluation of Project Report subjected to plagiarism check, Project Presentation skill and Question and Answer session in the ratio 50:25:25. SEE shall be at the end of IV semester. Project work evaluation and Viva-Voce examination (SEE), after satisfying the plagiarism check (below 20%).

Online Course Pool for M.TECH. COMPUTER SCIENCE AND ENGINEERING

NPTEL Courses (Currently Offered) July-2024

Sl no.	Course ID	Course Name	SME Name	Institute	UG/PG
1	noc24-cs86	Distributed Optimization and Machine Learning	Prof. Mayank Baranwal	IIT Bombay	PG
2	noc24-cs112	Secure Computation: Part II	Prof. Ashish Choudhury	IIIT Bangalore	PG
3	noc24-cs85	Practical Cyber Security for Cyber Security Practitioners	Prof. Sandeep K. Shukla	IIT Kanpur	UG/PG
4	noc24-cs100	Randomized Methods in Complexity	Prof. Nitin Saxena	IIT Kanpur	UG/PG
5	noc24-cs87	Computational Arithmetic - Geometry for Algebraic Curves	Prof. Nitin Saxena	IIT Kanpur	UG/PG
6	noc24-cs102	Reinforcement Learning	Prof. Balaraman Ravindran	IIT Madras	UG/PG
7	noc24-cs89	Deep Learning for Computer Vision	Prof. Vineeth N Balasubramanian	IIT Hyderabad	UG/PG
8	noc24-cs117	Parameterized Algorithms	Prof. Neeldhara Misra Prof. Saket Saurabh	IIT Gandhinagar	UG/PG
9	noc24-cs104	Applied Accelerated Artificial Intelligence	Prof. Satyajit Das Prof. Satyadhyan Chickerur Prof. Bharatkumar Sharma Prof. Adesuyi Tosin	IIT Palakkad KLE Technological University NVIDIA	PG
10	noc24-cs90	Social Network Analysis	Prof. Tanmoy Chakraborty	IIT Delhi	UG/PG
11	noc24-cs119	Software Engineering	Prof. Rajib Mall	IIT Kharagpur	PG
12	noc24-cs106	Computational Complexity	Prof. Subrahmanyam Kalyanasundaram	IIT Hyderabad	PG
13	noc24-cs114	Deep Learning - IIT Ropar	Prof. Sudarshan Iyengar Prof. Sukrit Gupta	IIT Ropar	UG/PG
14	noc24-cs121	Cyber Security and Privacy	Prof. Saji K Mathew	IIT Madras	PG
15	noc24-cs93	Multi-Core Computer Architecture	Prof. John Jose	IIT Guwahati	UG/PG
16	noc24-cs107	Statistical Learning for Reliability Analysis	Prof. Monalisa Sarma	IIT Kharagpur	UG/PG
17	noc24-cs94	Ethical Hacking	Prof. Indranil Sengupta	IIT Kharagpur	UG/PG
18	noc24-cs125	Programming in Modern C++	Prof. Partha Pratim Das	IIT Kharagpur	UG/PG
19	noc24-cs95	Introduction to Industry 4.0 and Industrial Internet of Things	Prof. Sudip Misra	IIT Kharagpur	PG
20	noc24-cs116	Introduction To Algorithms and Analysis	Prof. Sourav Mukhopadhyay	IIT Kharagpur	UG/PG
21	noc24-cs109	Algorithmic Game Theory	Prof. Palash Dey	IIT Kharagpur	PG
22	noc24-cs126	Design & Implementation of Human-Computer Interfaces	Prof. Samit Bhattacharya	IIT Guwahati	UG/PG
23	noc24-cs97	Approximation Algorithm	Prof. Palash Dey	IIT Kharagpur	UG/PG
24	noc24-cs132	Responsible & Safe AI Systems	Prof. Ponnurangam Kumaraguru Prof. Balaraman Ravindran Prof. Arun Rajkumar	IIIT Hyderabad and IIT Madras	UG/PG

SEMESTER – I

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS)

SEMESTER - I

Applied Mathematics (2:2:0) 3

(Effective from the academic year 2024-25)

Course Code	24MAT11	CIE Marks	50
Teaching Hours/Week (L:T:P)	2:2:0	SEE Marks	50
Total Number of Contact Hours	40	Exam Hours	3

Course Objectives: This course will enable students to:

1. Explore the foundational aspects of statistical methods and Linear Algebra.
2. Apply the concept of probability distribution of discrete and continuous random variables.
3. Apply the concept of Linear Algebra and Vector Spaces to solve real world problems.
4. Analyze the statistical data for testing of hypothesis and draw inferences using Number Theory

Preamble: Applied Mathematics is a branch of mathematics that focuses on the practical application of mathematical techniques to solve real-world problems across various fields such as science, engineering, economics, and industry. It involves using mathematical models, algorithms, and computational methods to analyze and predict the behavior of complex systems. Applied mathematicians work on problems like optimizing processes, modeling natural phenomena, and designing systems for efficiency and reliability. The field bridges the gap between theoretical mathematics and practical applications, enabling innovations in areas such as cryptography, data analysis, fluid dynamics, financial
modeling, and more.

Module – 1

Introduction: Understanding of Vector spaces, graph theory, Statistical models & their applications in Engineering, Economics and Statistics.

Linear Algebra-I

Vector Spaces: Vector spaces; subspaces Linearly independent and dependent vectors, Basis and dimension, coordinate vectors-Illustrative examples. Linear transformations, Representation of transformations by matrices.

(8 hours)

Module – 2

Linear Algebra-II

Computation of Eigen values and Eigen vectors of real symmetric matrices-Jacobi and Given's method. Orthogonal vectors and orthogonal basis. Gram-Schmidt Orthogonalization process. QR decomposition, singular value decomposition.

(8 hours)

Module – 3

Random Variables: Review of Random Variables, Probability Distributions: Binomial distribution, Poisson distribution, Normal distribution, Exponential distribution and Uniform distribution.

(8 hours)

Module – 4
Joint probability distribution and Stochastic Process: Joint probability distribution (both discrete and continuous). Stochastic Processes: Introduction, Classification of stochastic processes, discrete time processes, Stationary, Ergodicity, Autocorrelation. (8 hours)
Module – 5
Number Theory: Divisibility, GCD, Euclidean algorithm, Congruences, Linear Congruences, The Chinese Remainder theorem, Solving Polynomials, Linear Diophantine Equation, System of Linear Congruences, Euler's Theorem, Wilson Theorem and Fermat's little theorem. (8 hours)
Course Outcomes: The students will be able to CO1: Apply probability formulations for new predictions with discrete and continuous RV's. CO2: Solve the vector spaces and related topics arising in magnification and rotation of images. CO3: Interpret the probability distribution arising in the study of engineering problems and their applications. CO4: Apply the statistical tools in multi variable distributions. CO5: Demonstrate knowledge and critical understanding of the well-established principles within Number Theory.
Question paper pattern: • **SEE** will be conducted for 100 marks. • Each full question is for 20 marks. (Answer five full questions out of 10 questions with intra modular choice). In every question, there will be a maximum of three sub-questions. • **CIE** will be announced prior to the commencement of the course. • 25 marks for test. An average of three tests will be taken. • 25 marks for Flexible Assessment Method.

Textbooks:

1. David C.Lay, Steven R.Lay and J.J.McDonald, "Linear Algebra and its Applications", 5th Edition, Pearson Education Ltd., 2015.
2. T.Veerarajan, "Probability, Statistics and Random Process", 3rdEdition,Tata Mc-Graw Hill Co., 2016.
3. Neal Coblitz, "A Course in Number Theory and Cryptography", Springer Verlag, Second edition, 2012.

References:

1. B.S. Grewal, "Higher Engineering Mathematics", Khanna Publishers, 44th Edition, 2017.
2. John Vince, "Foundation Mathematics for Computer Science", Springer International Publishing, Switzerland, 2015.
3. Burton, David M. Elementary number theory. Second edition. W. C. Brown Publishers, Dubuque, IA, 1989.

Web Resources:

1. Introduction to Probability and Statistics by the University of London: University of London. "Introduction to Probability and Statistics." Coursera, www.coursera.org/learn/probability-statistics.
2. Introduction to Number Theory by Stanford University (offered through Stanford Online but listed on Coursera): Stanford University. "Introduction to Number Theory." Coursera, www.coursera.org/learn/number-theory

Textbooks:

1. William Stallings, Cryptography and Network Security Principles and Practice, 7 th edition, Pearson, 2019.

References:

1. Damien Vergnaud and Michel Abdalla, Applied Cryptography and Network Security, 7th International Conference, ACNS 2009, Paris-Rocquencourt, France, June 2-5, 2009, Proceedings.
2. B. Schneier, "Applied Cryptography: Protocols, Algorithms, and Source Code in C", 2nd Edition, John Wiley & Sons, 1995.
3. Mihir Bellare and Phillip Rogaway, "Introduction to Modern Cryptography", 2005.
4. Alfred J. Menezes, Paul C. van Oorschot and Scott A. Vanstone, "Handbook of Applied Cryptography" CRC Press.
5. Neal Koblitz, A Course in Number Theory and Cryptology, Springer 1987.

Web Resources:

1. "Cryptography" by Stanford University: Stanford University. "Cryptography." Coursera, www.coursera.org/learn/crypto.
2. "Applied Cryptography" by the University of Colorado System: University of Colorado System. "Applied Cryptography." Coursera, www.coursera.org/learn/applied-cryptography.
3. "Introduction to Cryptography" by the University of London: University of London. "Introduction to Cryptography." Coursera, www.coursera.org/learn/crypto-introduction.

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS)

SEMESTER - I

Advanced Algorithms (3:0:2) 4

(Effective from the academic year 2024-25)

Course Code	24MCS12	CIE Marks	50
Teaching Hours/Week (L:T:P)	3:0:2	SEE Marks	50
Total Number of Contact Hours	50	Exam Hours	3

Course Objectives:

This course will enable students to:

1. Develop skills in analysing and implementing advanced sorting algorithms and matrix multiplication techniques.
2. Master string-matching techniques and probabilistic algorithms for efficient data processing.
3. Explore graph algorithms and tackle NP-complete problems using various strategies.
4. Design parallel algorithms using OpenMP and MPI libraries.

Module – 1

Preamble: The Advanced Algorithms course provides learners with a comprehensive understanding of how to create efficient algorithms and analyze their performance. This course covers fundamental algorithmic techniques such as divide-and-conquer, dynamic programming, greedy algorithms, and graph algorithms, as well as complexity analysis to evaluate their efficiency in terms of time and space. Students learn to design algorithms to solve complex computational problems and gain the skills to critically assess the trade-offs between different approaches. By the end of the course, students are well-prepared to apply these principles in both academic and real-world scenarios, ensuring optimal solutions in software development and problem-solving tasks.

Algorithm Analysis: Algorithm complexity - Growth of functions, Master method solution, Sorting Techniques & their time Complexity: Insertion Sort, Merge Sort, Heap Sort and Quick Sort, Sorting in Linear Time, Amortized Analysis, aggregation method, counting method, Strassen's algorithm for matrix multiplication. The recurrence – tree method.

Lab Programs/Experiments:

1. Program to implement Merge Sort, Heap Sort and Quick Sort algorithms.
2. Program to implement Strassen's algorithm for matrix multiplication.
3. Solving Algorithm Analysis problems.

(10 hours)

Module – 2

String-Matching Algorithms: Naïve string Matching, Rabin - Karp algorithm; String matching with finite automata, Knuth-Morris-Pratt algorithm; Boyer – Moore algorithms.

Probabilistic and Randomized Algorithms: Probabilistic algorithms; Randomizing deterministic algorithms, Monte Carlo and Las Vegas algorithms; Probabilistic numeric algorithms.

Lab Programs:

1. Program to implement Naive algorithm.
2. Program to implement Rabin - Karp algorithm.
3. Program to implement Boyer – Moore algorithm.
4. Program to implement Monte Carlo algorithm.

(10 hours)

Module – 3

Graph Algorithms: Bellman - Ford Algorithm; Single source shortest paths in a DAG; Johnson's Algorithm for sparse graphs; Flow networks and Ford-Fulkerson method; Maximum bipartite matching. Polynomials and the FFT: Representation of polynomials; The DFT and FFT; Efficient

implementation of FFT.

Lab Programs:

Program to implement Ford-Fulkerson method.

Program to implement Johnson's Algorithm.

(10 hours)

Module – 4

Distributed Memory Programming with MPI: Distributed Memory Programming with MPI Getting Started, The Trapezoidal Rule in MPI, Dealing with I/O, Collective Communication, MPI Derived Data types, A Parallel Sorting Algorithm.

Shared Memory Programming with OpenMP: Introduction to OpenMP, The Trapezoidal Rule, Scope of Variables, The Reduction Clause, The Parallel For Directive.

Lab Programs:

Compare the speedup of the parallel implementation of Quick sort using MPI (On a cluster of 5 Nodes) and OpenMP (Shared Memory Implementation on multicore machine).

Compare the speedup of the parallel implementation of Merge sort using MPI (On a cluster of 5 Nodes) and OpenMP (Shared Memory Implementation on multicore machine).

(10 hours)

Module – 5

NP-Complete Problems: Polynomial-time solvable problems, NP-Completeness and Reducibility, NP-Complete problems, NP- Hard Problems, Cooke's theorem, Clique decision problem, Graph coloring problem, Directed Hamiltonian cycle problem, Traveling salesman problem.

Lab Programs:

Traveling Salesman Problem (TSP)

Description: Given a list of cities and the distances between each pair, find the shortest possible route that visits each city exactly once and returns to the origin city.

Implementation: Use a brute force approach to calculate all possible permutations of city tours and find the shortest one, or use heuristics like nearest neighbour, genetic algorithms, or dynamic programming (e.g., Held-Karp algorithm).

Graph Colouring Problem

Description: Assign colours to the vertices of a graph so that no two adjacent vertices share the same colour using the minimum number of colours.

Implementation: Use backtracking to try all possible colourings or employ heuristic algorithms like the greedy colouring algorithm.

(10 hours)

Course outcomes: The students will be able to

CO1: Understand advanced methods of designing and analyzing algorithms.

CO2: Discuss different areas such as network flows, number theory and utilize concepts therein to develop algorithms in various domains.

CO3: Design and implement various algorithms.

CO4: Evaluate the implementations with respect to time and space complexity.

CO5: Design parallel algorithms with MPI and OpenMP

Question paper pattern:

- **SEE** will be conducted for 100 marks.
- Each full question is for 20 marks. (Answer five full questions out of 10 questions with intra modular choice). In every question, there will be a maximum of three sub-questions.
- **CIE** will be announced prior to the commencement of the course.
- 25 marks for test. Average of three tests will be taken.
- 25 marks for Flexible Assessment Method.

Textbooks:

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, Introduction to Algorithms, MIT Press, 3rd Edition, 2009.
2. Michael T. Goodrich and Roberto Tamassia, Algorithm Design Foundations, Analysis, and Internet Examples, John Wiley & Sons, Inc., 2nd Edition, 2009.
3. Peter S Pacheco and Matthew Malensek An introduction to parallel programming, Second Edition, Morgan Kaufmann Publishers, 2021.
4. Grama, Ananth, Anshul Gupta, George Karypis, and Vipin Kumar. Principles of parallel algorithm design. Introduction to Parallel Computing, 2nd ed. Addison Wesley, Harlow (2003).

References:

1. Dave and Dave, Design and Analysis of Algorithms, Pearson Education.
2. A.V. Aho, J.E. Hopcroft, Design & Analysis of Computer Algorithms, 2nd Edition, PHI.

Web Resources:

1. "Algorithms Specialization." Coursera, offered by Stanford University, https://www.coursera.org/specializations/algorithms. Accessed 28 Aug. 2024.
2. "Advanced Algorithms and Complexity." Coursera, offered by University of California, San Diego, https://www.coursera.org/learn/advanced-algorithms. Accessed 28 Aug. 2024.

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS)

SEMESTER – I

Artificial Intelligence (3:0:0) 3

(Effective from the academic year 2024-25)

Course Code	24MCS13	CIE Marks	50
Teaching Hours/Week (L: T:P)	3:0:0	SEE Marks	50
Total Number of Contact Hours	40	Exam Hours	3

Course Objectives: This course will enable students to:

Apply the basic principles, models, and algorithms of AI

Recognize, model, and solve problems in the analysis and design of information systems.

Explore structures and algorithms of a selection of techniques related to searching, reasoning, machine learning, and language processing.

Implement AI models using industry-standard tools and frameworks.

Preamble: Studying Artificial Intelligence opens doors to a realm where human ingenuity intersects with cutting-edge technology. This field delves into the creation of intelligent systems that can perceive, reason, and learn, revolutionizing how we interact with machines and data. Its significance spans industries, from healthcare to finance, driving innovation and efficiency. As AI continues to advance, understanding its principles and applications becomes crucial for shaping the future of technology and society. Pursuing this field promises a journey of discovery, innovation, and limitless possibilities.

Module – 1

Intelligent agents: Introduction, How agents should act, mapping from percept sequences to actions, Agents and Environments, the concept of Rationality, The Nature of Environments, Rational Agent, Structure of intelligent agents: Agent programs Simple reflex agents, model-based, goal-based agents, utility-based agents and learning agents, behavior and environment in which a particular agent operates, properties of agent.

(8 Hours)

Module – 2

Problem-Solving and Agents: Uniform Cost Search, Best First Search, Hill Climbing, Simulated Annealing, Problem Solving Agents, Software Agents- Concepts, Definition and Characteristics, Designing Agents as if People Mattered, Agents from Direct Manipulation to Delegation, Agents for Information Gathering

(7 Hours)

Module – 3

Knowledge Representation: Representations and Mappings, Approaches to Knowledge Representation, Issues in Knowledge Representation, Ontologies and Ontological Engineering
Using Predicate Logic: Representing Simple Facts in Logic, Representing Instance and ISA Relationships, Computable Functions and Predicates, Forward Chaining, Parse Tree Representation of Knowledge

(9 Hours)

Module – 4

Uncertain Knowledge and Reasoning: Quantifying Uncertainty: Acting under Uncertainty, Bayes' Rule and Its Use. Probabilistic Reasoning: Representing Knowledge in an Uncertain Domain, The Semantics of Bayesian Networks, Efficient Representation of Conditional Distributions. Making Simple Decisions: Combining Beliefs and Desires under Uncertainty, The basis of Utility Theory.

(8 Hours)

Module – 5

Expert Systems and Knowledge Acquisition: Representing and Using Domain Knowledge, Expert system Shells, Knowledge Acquisition. **Genetic Algorithms: Copying Natures Approaches:** A peek into Biological World, Significance of Genetic Operators, Termination Parameter

Case Studies: KAoS Open Agent Architecture, Real Time Case Studies on Semantic Artificial Intelligence, eXplainable Intelligence

(8 Hours)

Course Outcomes: The students will be able to:

CO1: Illustrate basic principles and applications of Artificial Intelligence.
CO2: Apply the different search algorithms for problem solving.
CO3: Analyze the knowledge representation and reasoning with different decision-making theories.
CO4: Apply Artificial Intelligence techniques for real solving Real world problems.

Question paper pattern:

- **SEE** will be conducted for 100 marks.
- Each full question is for 20 marks. (Answer five full questions out of 10 questions with intra modular choice). In every question, there will be a maximum of three sub- questions.
- **CIE** will be announced prior to the commencement of the course.
- 25 marks for the test. Average of three tests will be taken.
- 25 marks for Flexible Assessment Method.

Textbooks:

1. Kevin Knight, Elaine Rich and B. Nair, Artificial Intelligence, Third Edition, 2017.
2. Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, Fourt Edition, 2020.
3. Jeffrey M Bradshaw, An introduction to software agents, Software agents 4 (2012): 3-46.

References:

1. Flasinski M, Introduction to Artificial Intelligence, SPRINGER 2017.
2. Elaine Rich and Kevin Knight, "Artificial Intelligence", 3rd Edition, Tata McGraw-Hill, 2003

Web Resources:

1. AI Foundations for Everyone by IBM: IBM. "AI Foundations for Everyone." Coursera, www.coursera.org/learn/ai-foundations-for-everyone.
2. Deep Learning Specialization by Andrew Ng (DeepLearning.AI): DeepLearning.AI. "Deep Learning Specialization." Coursera, www.coursera.org/specializations/deep-learning.
3. Artificial Intelligence: Principles and Techniques by Stanford University: Stanford University. "Artificial Intelligence: Principles and Techniques." Coursera, www.coursera.org/learn/artificial-intelligence.

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS)

SEMESTER – I

Fundamentals of Data Science (3:0:0) 3

(Effective from the academic year 2024-25)

Course Code	24MCS14	CIE Marks	50
Teaching Hours/Week (L:T:P)	3:0:0	SEE Marks	50
Total Number of Contact Hours	40	Exam Hours	3

Course Objectives: This course will enable students to:

1. Understand the foundational aspects of data science and the role of a data scientist in the analytics process.
2. Explore the mathematical concepts for Data Science.
3. Apply concepts of linear algebra and statistical models for data analytics.
4. Develop skills for interpretation of types of data Visualization techniques and Model Evaluations.
5. Explore Linear Algebra concepts and Big Data Algorithmics operations.

Preamble: In today's data-driven world, the field of data science has become a pivotal force in shaping decisions and strategies across various sectors. At its core, data science involves harnessing vast amounts of data to extract meaningful insights and drive informed decisions.

Module – 1

Introduction To Data Science: Definition, Big Data and Data Science Hype, Datafication , Data Science Profile, Meta-Definition, Data Scientist, Statistical Inference, Populations and Samples, Populations and Samples of Big Data, Big Data Can Mean Big Assumptions, Modeling, Philosophy of Exploratory Data Analysis, The Data Science Process , A Data Scientist's Role in this Process Case Study: RealDirect.

(8 hours)

Module – 2

Mathematical Preliminaries: Probability, Descriptive Statistics, Correlation Analysis. **Data Munging:** Properties of Data, Languages for Data Science, Collecting Data, Cleaning Data, Crowdsourcing.

(8 hours)

Module – 3

Scores and Rankings: Developing Scoring Systems, Z-scores and Normalization, Advanced Ranking Techniques. **Statistical Analysis:** Sampling from Distributions, Statistical Distributions, Statistical Significance, Permutation Tests and P-values.

(8 hours)

Module – 4

Visualizing Data: Exploratory Data Analysis, Developing a Visualization Aesthetic, Chart Types, Great Visualizations. **Mathematical Models:** Philosophies of Modeling, A Taxonomy of Models, Baseline Models, Evaluating Models, Evaluation Environment. **Linear Algebra:** The Power of Linear Algebra, Visualizing Matrix Operations, Factoring Matrices, Eigenvalues and Eigenvectors, Eigenvalue Decomposition.

(9 hours)

Module – 5

Big Data: Achieving Scale: What is Big Data?, Algorithmic for Big Data, Filtering and Sampling, Parallelism, MapReduce. **Big Data Programming models:** Introduction, Distributed File systems Case Study:HDFS/GFS, MapReduce Programming Model, Case Study: Hbase/BigTable, Matrixoperations.

(7 hours)

Course outcomes: The students will be able to

CO1: Design and implement innovative solutions by synthesizing various data science techniques to address complex, multidisciplinary problems.
CO2: Critically analyze real-world problems to identify and formulate them as data science problems, considering the appropriate methods and techniques for analysis.
CO3: Build, and prepare data for use with a variety of statistical methods and models
CO4: Analyze Data using various Visualization techniques.
CO5: Apply Linear Algebra operations and Big Data Algorithmic in data analytics.

Question paper pattern:

1. **SEE** will be conducted for 100 marks.
2. Each full question is for 20 marks. (Answer five full questions out of 10 questions with intra modular choice). In every question, there will be a maximum of three sub-questions.
3. **CIE** will be announced prior to the commencement of the course.
4. 25 marks for test. Average of three tests will be taken.
5. 25 marks for Flexible Assessment Method.

Textbooks:

1. Steven S. Skiena, "The Data Science Design Manual", Springer 2017.
2. Rachel Schutt & O'neil, "Doing Data Science", Straight Talk from The Frontline O'REILLY, ISBN:978-1-449-35865-5, 1st edition, October 2013.
3. "Hadoop: The Definitive Guide, Tom White", 4th Edition, O"Reilly,2015

References:

1. Joel Grus," Data Science from Scratch" First Edition, April 2015
2. Gareth James, Daniela Witten, Trevor Hatie, RoberstTibhirani , "An Introduction to Statistical Learning-with Applications in R", 2013
3. Jure Leskovek, Anand Rajaraman and Jeffrey Ullman. Mining of Massive Datasets. v2.1, Cambridge University Press. 2 edition (30 September 2014)
4. R Programming for Data Science, Roger D. Peng, LeanPub, 2015.

Web Resources:

- "Data science for engineers" https://nptel.ac.in/noc/courses/noc20/SEM1/noc20- cs28/
- " Python for Data Science"https://archive.nptel.ac.in/noc/courses/noc22/SEM1/noc22-cs32/

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS)

SEMESTER – I

Cryptography and Network Security (2:2:0) 3

(Effective from the academic year 2024-25)

Course Code	24MCS15	CIE Marks	50
Teaching Hours/Week (L: T:P)	2:2:0	SEE Marks	50
Total Number of Contact Hours	40	Exam Hours	3

Course Objectives:

This course will enable students to:

1. Explain standard algorithms used to provide confidentiality, integrity and authenticity.
2. Distinguish key distribution and management schemes.
3. Apply encryption techniques to secure data in transit across data networks
4. Implement security applications in the field of Information technology.

Preamble: Embarking on the study of "Advanced Cryptography" delves into the intricate world of securing digital communication and information. This field explores advanced cryptographic algorithms and protocols, essential for safeguarding sensitive data in an increasingly interconnected world. Its significance lies in thwarting cyber threats and ensuring privacy and integrity in digital transactions and communications. As cybersecurity concerns escalate, expertise in advanced cryptography becomes pivotal for protecting critical infrastructure and preserving digital trust. Pursuing this specialization promises a deep dive into the forefront of cryptographic techniques, offering opportunities to innovate and defend against evolving cyber threats.

Module – 1

Computer and Network Security: Computer Security Concepts, The OSI Security Architecture, Security Attacks, Security Services, Security Mechanisms, Fundamental Security Design Principles, Attack Surfaces and Attack Trees, A Model for Network Security, and Standards.

Symmetric Ciphers, Classical encryption techniques: Symmetric Cipher Model, Substitution Techniques, Transposition Techniques, Rotor Machines, Stenography. (Chapter 1 , 2 & 3) (8 Hours)

Module – 2

Block Ciphers and the Data Encryption Standard: Traditional Block Cipher Structure,The Data Encryption Standard, A DES Example, The Strength of DES, Block cipher design principles and modes of operation. **Advanced Encryption Standard:** Finite Field Arithmetic, AES Structure, AES Transformation Functions, AES Key Expansion, An AES Example and AES Implementation. **Asymmetric Ciphers: Public-Key Cryptography and RSA**: Principles of Public-Key Cryptosystems, The RSA Algorithm and Diffie-Hellman Key Exchange, Elliptic Curve Arithmetic, Elliptic Curve Cryptography.

(Topics from Chapter 4, 6, 9 & 10) (8 Hours)

Module – 3

Cryptographic Data Integrity Algorithms

Cryptographic Hash Functions : Applications of Cryptographic Hash Functions

Two Simple Hash Functions, Requirements and Security, Hash Functions Based on Cipher Block Chaining, Secure Hash Algorithm (SHA) , & SHA-3. **Message Authentication Codes**: Message Authentication Requirements, Message Authentication

Functions, Requirements for Message Authentication Codes, Security of MACs, MACs Based on Hash Functions: HMAC.

Digital Signatures: Digital Signatures, Elgamal Digital Signature Scheme, SchnorrDigital Signature Scheme.

(Topics from Chapter 11, 12, & 13) (8 Hours)

Module – 4

Mutual Trust

Key Management and Distribution: Symmetric Key Distribution Using Symmetric Encryption, Symmetric Key Distribution Using Asymmetric Encryption, Distribution of Public Keys, X.509 Certificates, public-Key Infrastructure.

User Authentication: Remote User-Authentication Principles, Remote User-authentication Using Symmetric Encryption, Kerberos, Remote User-Authentication Using Asymmetric Encryption.

Network Access Control and Cloud Security : Network Access Control, Extensible Authentication Protocol, IEEE 802.1X Port-Based Network Access Control, Cloud Computing , Cloud Security Risks and Countermeasures , Data Protection in the Cloud, Cloud Security as a Service, Addressing Cloud Computing Security Concerns.

(Topics from Chapter 14, 15, & 16) (8 Hours)

Module – 5

Transport Layer Security: HTTPS, Secure Shell (SSH), Wireless Security, Mobile Device Security, IEEE 802.11 Wireless LAN Overview, IEEE 802.11i Wireless LAN Security.

Electronic Mail Security: internet Mail Architecture, Email Formats, Email Threats and Comprehensive Email Security , S/MIME, Pretty Good Privacy, DNSSEC,
DNS-Based Authentication of Named Entities.

IP Security Overview: IP Security Policy, Encapsulating Security Payload, Combining Security Associations, Internet Key Exchange, Cryptographic Suite.

Recap/Summary of the course.

(Topics from Chapter 17, 18, & 19) (8 Hours)

Course Outcomes: The students will be able to:

CO1: Apply the OSI security architecture, number theory and cipher techniques for the given problem.

CO2: Compare the performance of various cryptographic data integrity techniques for the identified problem.

CO3: Analyze the vulnerabilities in any computing system and design a cryptographic solution for the given problem/ case study

CO4: Examine the working of the techniques used for Mutual trust and security on internet and compare their performance.

Question paper pattern:

- **SEE** will be conducted for 100 marks.
- Each full question is for 20 marks. (Answer five full questions out of 10 questions with intra modular choice). In every question, there will be a maximum of three sub-questions.
- **CIE** will be announced prior to the commencement of the course.
- 25 marks for test. Average of three tests will be taken.
- 25 marks for Flexible Assessment Method.

Textbooks:

1. William Stallings, Cryptography and Network Security Principles and Practice, 7 th edition, Pearson, 2019.

References:

1. Damien Vergnaud and Michel Abdalla, Applied Cryptography and Network Security, 7th International Conference, ACNS 2009, Paris-Rocquencourt, France, June 2-5, 2009, Proceedings.
2. B. Schneier, "Applied Cryptography: Protocols, Algorithms, and Source Code in C", 2nd Edition, John Wiley & Sons, 1995.
3. Mihir Bellare and Phillip Rogaway, "Introduction to Modern Cryptography", 2005.
4. Alfred J. Menezes, Paul C. van Oorschot and Scott A. Vanstone, "Handbook of Applied Cryptography" CRC Press.
5. Neal Koblitz, A Course in Number Theory and Cryptology, Springer 1987.

Web Resources:

1. "Cryptography" by Stanford University: Stanford University. "Cryptography." Coursera, www.coursera.org/learn/crypto.
2. "Applied Cryptography" by the University of Colorado System: University of Colorado System. "Applied Cryptography." Coursera, www.coursera.org/learn/applied-cryptography.
3. "Introduction to Cryptography" by the University of London: University of London. "Introduction to Cryptography." Coursera, www.coursera.org/learn/crypto-introduction.

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System

(CBCS) SEMESTER – I

Artificial Intelligence Laboratory (0:1:2) 2

(Effective from the academic year 2024-25)

Course Code	24MCSL16	CIE Marks	50
Teaching Hours/Week (L: T:P)	0:1:2	SEE Marks	50
Total Number of Contact Hours	30	Exam Hours	2

Course Objectives:

This course will enable students to:

1. To enable learners to implement AI algorithms using various programming languages and tools, emphasizing practical applications in areas such as search, optimization, and decision-making.
2. Implement AI search algorithms.
3. Develop the ability to apply AI methodologies to solve real-world problems in domains.

Preamble: This course offers an in-depth study of essential AI techniques, focusing on search algorithms, game-playing strategies, and optimization methods. Students will learn to implement algorithms such as DFS, BFS, and A*, explore game strategies like Minimax and Monte Carlo Tree Search, and apply heuristic methods like Hill Climbing and Simulated Annealing. Additionally, the course covers logical reasoning with First-Order Predicate Logic and resolution principles, providing a solid foundation for solving complex AI problems.

Part A- Tutorial

Fundamentals of AI: Definition and goals of AI, History and applications of AI Types of AI (Narrow AI, General AI, Strong AI).

Search Algorithms in AI: Uniform Cost Search, Informed Search Strategies (Best First Search), Optimization Search Strategies (Hill Climbing, Simulated Annealing).

Heuristic Search and Problem Solving: Concept of Heuristics, Heuristic Search Strategies, Solving Constraint Satisfaction Problems (e.g., N-Queens).

Logical Reasoning and Knowledge Representation: First-Order Predicate Logic (FOPL), Principle, Forward Chaining. Ontologies and their Concepts for Knowledge Representation, Web Protege.

PART-B: Experiments

1. Implement Best First Search Algorithm to find the shortest path in a city.
2. Implement Uniform Cost Search Algorithm for Finding the Shortest Path.
3. Implement the Hill Climbing Algorithm for Solving the N-Queens Problem.
4. Implement the Simulated Annealing Algorithm for Optimization.
5. Create a simple ontology representing medical conditions, symptoms, and treatments using Protégé or Web Protégé as a Tool of Choice. Introduce data properties to capture more detailed information. Develop a more complex hierarchy and explore inheritance. Introduce SWRL (Semantic Web Rule Language) to define rules for automatic inference.
6. Create a simple ontology representing key concepts in an e-commerce system, including products, customers, orders, and payments. Introduce data properties to capture more detailed information about products, customers, and orders. Develop a more complex hierarchy and explore inheritance within the e-commerce domain. Align your e-commerce ontology with an existing standard ontology (e.g., GoodRelations - an ontology for e-commerce).

PROJECT

Making use of the concepts learnt in Artificial Intelligence, develop an expert system for any prospective domain of your choice. The project must encompass Knowledge Representation, Inference Rules and Inferencing Engine, Decision Making, and Optimization if necessary.

Course Outcomes: The students will be able to:

CO1: Demonstrate the ability to solve practical problems using different AI search algorithms.
CO2: Create AI-driven game strategies for various board games.
CO3: Apply heuristic and optimization methods to effectively tackle complex computational problems.
CO4: Implement logical reasoning techniques to represent knowledge in AI systems using suitable tools.

Text Books

1. Stuart J. Russell and Peter Norvig, "Artificial Intelligence – A Modern Approach", 4th Edition, Pearson Education, 2021.
2. Saroj Kaushik, "Artificial Intelligence", 2nd Edition, Cengage Learning India Pvt. Ltd. 2023

Web Resources

1. https://onlinecourses.nptel.ac.in/noc22_cs56/preview
2. "AI For Everyone." Coursera, offered by Andrew Ng, https://www.coursera.org/learn/ai-for-everyone. Accessed 28 Aug. 2024.

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS)

SEMESTER – I

No SQL Database Laboratory (0:1:2) 2

(Effective from the academic year 2024-25)

Course Code	24MCSL17	CIE Marks	50
Teaching Hours/Week (L: T:P)	0:1:2	SEE Marks	50
Total Number of Contact Hours	30	Exam Hours	2

Course Objectives: This course will enable students to:

1. Understand various NoSQL database types and their applications.
2. Explore various concepts in NoSQL database technologies to develop advanced data models.
3. Apply NoSQL solutions to solve complex data storage and retrieval challenges.

Preamble: NoSQL databases have become essential for handling large-scale, distributed, and unstructured data in modern applications. This lab provides hands-on experience with various NoSQL databases, including key-value stores, document stores, column-family stores, and graph databases. Students will learn to design, implement, and manage NoSQL databases, focusing on scalability, performance, and data modeling. The lab emphasizes practical skills in solving real-world data storage and retrieval challenges across different domains such as e-commerce, social media, IoT, and big data analytics. By the end of the lab, students will be well-prepared to utilize NoSQL databases effectively in their future projects and professional roles.

Part A- Tutorial

Introduction to NoSQL Databases: Understand what NoSQL databases are and how they differ from traditional SQL databases. Types of NoSQL databases: document stores, key-value stores, column-family stores, and graph databases.

Data Models and Schema Design: different data models used in NoSQL databases. Basic schema design principles for document, key-value, column-family, and graph databases.

CRUD Operations: Basic CRUD operations (Create, Read, Update, Delete) in NoSQL databases. NoSQL database-specific commands for these operations.

Query Languages and APIs: Get acquainted with the query languages and APIs of various NoSQL databases (e.g., MongoDB Query Language, Redis commands, Cassandra Query Language (CQL), Cypher for Neo4j).

Data Storage and Retrieval: stored and retrieved in NoSQL databases. Understand the basics of data indexing and retrieval techniques. **Scaling and Performance**: Understand the concepts of scalability and performance in NoSQL databases. Learn about sharding, replication, and partitioning.

PART-B: Experiments	
1	Create and query a simple database using MongoDB.
2	Implement CRUD operations in MongoDB using a programming language (e.g., Python).
3	Perform aggregation operations in MongoDB.
4	Perform map-reduce queries in MongoDB.
5	Design a schema for a MongoDB database.
6	Implement indexing in MongoDB to optimize query performance.
7	Set up a replica set in MongoDB for high availability.
8	Use MongoDB Atlas to deploy a cloud-based MongoDB cluster.
9	Perform text search operations in MongoDB.
10	Implement transactions in MongoDB for multi-document operations.

Course Outcomes: The students will be able to:

CO1: Evaluate different NoSQL database technologies determining the most suitable choice for specific real-world scenarios.

CO2: Design complex, scalable NoSQL database architectures for applications requiring high availability, horizontal scaling, and distributed data storage.

CO3: Develop advanced data models tailored to NoSQL databases, optimizing for performance, consistency, and scalability across various application domains.

Text Books:

5. Sadalage, P. & Fowler, NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, Wiley Publications,1st Edition ,2019.
6. MongoDB:The Definitive Guide- Powerful and Scalable Data Storage Third Edition 2020

References:

1. Strauch, Christoph. NoSQL Databases: A Step-by-Step Guide for Beginners. Springer, 2018.
2. Sadalage, Patrick, and Martin Fowler. NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence. Addison-Wesley, 2012.
3. Harrison, C.J. Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement. Pragmatic Bookshelf, 2012.
4. Robinson, Ian, Jim Webber, and Emil Eifrem. Graph Databases: New Opportunities for Connected Data. O'Reilly Media, 2015.

Web Resources:

1. "Databases and SQL for Data Science with Python." Coursera, offered by IBM, https://www.coursera.org/learn/sql-data-science. Accessed 28 Aug. 2024.
2. "NoSQL Database Systems." Coursera, offered by University of California, San Diego, https://www.coursera.org/learn/nosql-database-systems. Accessed 28 Aug. 2024.
3. "Introduction to MongoDB." Coursera, offered by MongoDB University, https://www.coursera.org/learn/introduction-to-mongodb. Accessed 28 Aug. 2024.

SEMESTER – II

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS)

SEMESTER – II

Machine Learning (3:0:2) 4

(Effective from the academic year 2024-25)

Course Code	24MCS21	CIE Marks	50
Teaching Hours/Week (L: T:P)	3:0:2	SEE Marks	50
Total Number of Contact Hours	50	Exam Hours	3

Course Objectives: This course will enable students to:

1. Master foundational machine learning concepts including algorithms, overfitting, underfitting, and hyperparameter tuning, preparing for advanced applications.
2. Develop proficiency in designing and training deep feedforward networks, understanding regularization techniques, and implementing back-propagation for efficient learning.
3. Acquire expertise in convolutional neural networks for image analysis and recognition tasks, utilizing efficient convolution algorithms and understanding the neuroscientific basis of their design.
4. Explore practical methodologies for model evaluation, hyperparameter selection, and debugging strategies, enabling deployment in real-world applications such as computer vision and speech recognition.

Preamble: Machine Learning helps in accurately predicting or classifying outcomes for new data points by learning patterns from historical data. Machine Learning (ML) is a subset of artificial intelligence that enables computers to learn and make decisions or predictions based on data. Unlike traditional programming, where explicit instructions are provided, ML algorithms identify patterns and relationships within large datasets to improve their performance over time without being explicitly programmed for specific tasks. Data scientists can work with machine learning with equal ease. Data science allows data collected for other purposes to be applied to model problems related to various domains. The most popular programming languages among data scientists are open source tools that include or support pre-built statistical and machine learning capabilities. Machine learning is expanding across all fields such as banking and finance, information technology, media entertainment, gaming etc.

Module – 1

Machine Learning Basics: Learning Algorithms, Capacity, Overfitting and Underfitting, Hyper parameters and Validation Sets, Estimators, Bias and Variance, Maximum Likelihood Estimation, Bayesian Statistics, Supervised Learning Algorithms, Unsupervised Learning Algorithms, Stochastic Gradient Descent, Building a Machine Learning Algorithm, Challenges Motivating Deep Learning

(10 Hours)

Module – 2

Deep Feedforward Networks: Example: Learning XOR, Gradient-Based Learning, Hidden Units, Architecture Design, Back-Propagation and Other Differentiation Algorithms.
Regularization for Deep Learning: Parameter Norm Penalties, Norm Penalties as Constrained Optimization, Regularization and Under-Constrained Problems, Dataset Augmentation, Noise Robustness, Semi-Supervised Learning, Multi-Task Learning, Early Stopping, Parameter Tying and Parameter Sharing, Sparse Representations, Bagging and Other Ensemble Methods Dropout, Adversarial Training, Tangent Distance, Tangent Prop, and Manifold Tangent Classifier.

(10 Hours)

Module – 3

Convolutional Networks: The Convolution Operation, Motivation,Pooling, Convolution and Pooling as an Infinitely Strong Prior, Variants of the Basic Convolution Function, Structured Outputs ,Data Types, Efficient Convolution Algorithms, Random or Unsupervised Features, The Neuroscientific Basis for Convolutional Networks, Convolutional Networks and the History of Deep Learning.

(10 Hours)

Module – 4

Sequence Modeling: Recurrent and Recursive Nets, Unfolding Computational Graphs, Recurrent Neural Networks, Bidirectional RNNs, Encoder-Decoder Sequence-to-Sequence Architectures, Deep Recurrent Networks, Recursive Neural Networks, The Challenge of Long-Term Dependencies, Echo State Networks, Leaky Units and Other Strategies for Multiple Time Scales, The Long Short-Term Memory and Other Gated RNNs, Optimization for Long-Term Dependencies, Explicit Memory.

(10 Hours)

Module – 5

Practical Methodology: Performance Metrics, Default Baseline Models, Determining Whether to Gather More Data, Selecting Hyperparameters, Debugging Strategies, Example: Multi-Digit Number Recognition. **Applications:** Large-Scale Deep Learning, Computer Vision, Speech Recognition, Natural Language Processing, Other Applications.

(10 Hours)

Machine Learning Laboratory:

Part A: Exercises

1. Demonstrate the Principal Component Analysis for Dimensionality Reduction.
2. Design and Develop the AdaBoost classifier from scratch using Python.
3. Implement the concept of Decision Tree Learning on a suitable dataset of your choice.
4. Demonstrate the concept of Bagging using Random Forests for a dataset of your choice.
5. Demonstrate the concept of Logistic Regression for classification on a suitable dataset.

Part B: Project

Implement a machine learning project to address a specific problem using a provided dataset. The objective of the project is to develop a model that can make accurate predictions or classifications based on the data and evaluate its performance. The deliverables must include a comprehensive report documenting process and findings, Code files for your implementation and visualizations and results of your model evaluation.

Course Outcomes: The students will be able to:

CO1: Apply foundational machine learning principles to analyze and solve real-world problems, demonstrating understanding through algorithm selection and hyperparameter tuning.

CO2: Design deep feedforward networks proficiently, demonstrating knowledge of regularization techniques and back-propagation for efficient learning.

CO3: Analyze machine learning algorithms to understand their underlying principles, strengths, and weaknesses.

CO4: Design advanced machine learning models and architectures tailored to specific problems and datasets. Utilize frameworks and tools to build and train these models.

Question paper pattern:

- **SEE** will be conducted for 100 marks.
- Each full question is for 20 marks. (Answer five full questions out of 10 questions with intra modular choice). In every question, there will be a maximum of three sub-questions.
- **CIE** will be announced prior to the commencement of the course.
- 25 marks for test. Average of three tests will be taken.
- 25 marks for Flexible Assessment Method.

Textbooks:

1. Ian Goodfellow, Yoshua Bengio, and Aaron Courville. Deep learning. MIT press, 2016.
2. Alpaydin, Ethem. Introduction to machine learning. MIT press, 2020.

References:

1. Kevin P. Murphy, “Machine Learning: A Probabilistic Perspective”, MIT Press, 2012.
2. References
3. S. Raschka and V. Mirjalil, Python Machine Learning, 1 st ed, Packt Publishing, 2019.
4. C. M. Bishop, Pattern Recognition and Machine Learning, 1st ed, Springer, 2006.
5. R. O. Duda, P. E. Hart and D. G. Stork, Pattern Classification, 2nd ed, Wiley, 2000.

Web Resources:

1. Machine Learning with Python. Coursera, IBM, www.coursera.org/learn/machine-learning-with-python. Accessed 28 Aug. 2024.
2. Introduction to Machine Learning. Coursera, Duke University, www.coursera.org/learn/intro-to-machine-learning. Accessed 28 Aug. 2024.

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS)

Semester – II

Internet of Things (3:0:2) 4

(Effective from the academic year 2024-25)

Course Code	24MCS22	CIE Marks	50
Teaching Hours/Week (L:T:P)	3:0:2	SEE Marks	50
Total Number of Lecture Hours	50	Exam Hours	3

Course objectives: This course will enable students to:

1. Understand the fundamentals of Internet of Things.
2. Explore the working of basic IoT protocols.
3. Apply the concept of Internet of Things in the real-world scenario.
4. Build a small low-cost embedded system using Raspberry Pi/ Arduino.

Preamble: Internet of Things (IoT) is presently a hot technology worldwide. Government, academia, and industry are involved in different aspects of research, implementation, and business with IoT . The Internet of Things (IoT) is a course about the new paradigm of objects interacting with people, with information systems, and with other objects. Important to learn the fundamentals of this emerging technology.

Module – 1

INTRODUCTION TO IoT : What is The Internet of Things? Overview and Motivations, Examples of Applications, IPV6 Role, Areas of Development and Standardization, Scope of the Present Investigation. Internet of Things Definitions and frameworks-IoT Definitions, IoT Frameworks, Basic Nodal Capabilities. Internet of Things Application Examples-Overview, Smart Metering/Advanced Metering Infrastructure- Health/Body Area Networks, City Automation, Automotive Applications, Home Automation, Smart Cards, Tracking, Over-The-Air-Passive Surveillance/Ring of Steel, Control Application Examples, Myriad Other Applications.

(10 Hours)

Module – 2

Fundamental IoT Mechanism and Key Technologies-Identification of IoT Object and Services, Structural Aspects of the IoT, Key IoT Technologies. Evolving IoT Standards- Overview and Approaches, IETF IPV6 Routing Protocol for RPL Roll, Constrained Application Protocol, Representational State Transfer, ETSI M2M,Third Generation Partnership Project Service Requirements for Machine-Type Communications,
CENELEC,IETF IPv6 Over Low power WPAN, Zigbee IP(ZIP),IPSO

(10 Hours)

Module – 3

Layer ½ Connectivity: Wireless Technologies for the IoT-WPAN Technologies for IoT/M2M, Cellular and Mobile Network Technologies for IoT/M2M. Case Studies illustrating IoT Design-Introduction, Home Automation, Cities, Environment, Agriculture, Productivity Applications.

(10 Hours)

Module – 4

Data Analytics for IoT – Introduction, Apache Hadoop, Using Hadoop MapReduce for Batch Data Analysis, Apache Oozie, Apache Spark, Apache Storm, Using Apache Storm for Realtime Data Analysis, Structural Health Monitoring Case Study.

(10 Hours)

Module – 5

Building IoT with Raspberry pi/Arduino: IoT Physical Devices & Endpoints – What is IoT Device –Exemplary Devices: Raspberry Pi –About the Board - Linux on Raspberry Pi
- Raspberry Pi Interfaces -Programming Raspberry Pi with Python - Other IoT Platforms.

(10 Hours)

List of Experiments and Mini Project

Experiments:

1. Transmit a string using UART.
2. Point-to-Point communication of two Motes over the radio frequency.
3. Multi-point to single point communication of Motes over the radio frequency. LAN (Subnetting).
4. I2C protocol study: Reading Temperature and Relative Humidity value from the sensor.

Project:

- For any problem selected in the domain of IoT.
- Make sure that the application should use at least 3 or more sensors.
- Indicative areas like health care can be included.

Course Outcomes: The students will be able to

CO1: Understand the concepts of Internet of Things.
CO2: Illustrate basic framework used in IoT architecture.
CO3: Identify functionality and usage of various protocols in IoT architecture.
CO4: Analyze applications of IoT in real time scenarios.
CO5: Design a portable IoT application.

Question paper pattern:

- **SEE** will be conducted for 100 marks.
- Each full question is for 20 marks. (Answer five full questions out of 10 questions with intra modular choice). In every question, there will be a maximum of three sub-questions.
- **CIE** will be announced prior to the commencement of the course.
- 25 marks for the test. Average of three tests will be taken.
- 25 marks for Flexible Assessment Method.

Textbooks:

1. Arshdeep Bahga, Vijay Madisetti, Internet of Things: A hands-on approach, Universities Press, 2015.
2. Daniel Minoli,"Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2MCommunications", Wiley, 2013.

References:

1. Jan Ho¨ ller, Vlasios Tsiatsis , Catherine Mulligan, Stamatis , Karnouskos, Stefan Avesand. David Boyle, From Machine-to-Machine to the Internet of Things - Introduction to a New Age of Intelligence, Elsevier, 2014.
2. Olivier Hersent, David Boswarthick, Omar Elloumi,The Internet of Things – Key applications and Protocols,Wiley,2012.

Web Resources:

1. University of California, Irvine. An Introduction to Programming the Internet of Things (IoT) Specialization. Coursera, www.coursera.org/specializations/iot. Accessed 28 Aug. 2024.
2. University System of Georgia. Cybersecurity and the Internet of Things. Coursera, www.coursera.org/learn/iot-cybersecurity. Accessed 28 Aug. 2024.

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS)

SEMESTER – II

Probability, Statistics and Queuing Theory (3:0:0) 3

(Effective from the academic year 2024 -2025)

Course Code	24MCS23A	CIE Marks	50
Teaching Hours/Week (L:T:P)	3:0:0	SEE Marks	50
Total Number of Contact Hours	40	Exam Hours	3

Course Objectives: This course will enable students to:

1. Develop analytical capability and to impart knowledge of Probability, Statistics and Queuing.
2. Apply above concepts in Engineering and Technology.
3. Acquire knowledge of Hypothesis testing and Queuing methods and their applications so as to enable them to apply them for solving real world problems

Preamble: Embarking on the study of "Probability, Statistics, and Queuing Theory" delves into the mathematical foundations that underpin decision-making and system analysis. This field explores the probabilistic models and statistical methods essential for understanding uncertainty and variability in diverse domains. Its significance extends to optimizing resource allocation, predicting system performance, and informing strategic decision- making across industries. As the backbone of modern data science and operations research, expertise in these disciplines is indispensable for tackling complex real-world problems. Pursuing this specialization promises a profound exploration into the mathematical principles that drive innovation and efficiency in a data-driven world.

Module – 1

Introduction: Axioms of probability, Conditional probability, Total probability, Baye's theorem, Discrete Random variable, Probability mass function, Continuous Random variable. Probability density function, Cumulative Distribution Function, and its properties, Two-dimensional Random variables, Joint pdf / cdf and their properties.

(8 Hours)

Module – 2

Probability Distributions / Discrete distributions: Binomial, Poisson Geometric and Hyper-geometric distributions and their properties. Continuous distributions: Uniform, Normal, exponential distributions and their properties.

(8 Hours)

Module – 3

Random Processes: Classification, Methods of description, Special classes, Average values of Random Processes, Analytical representation of Random Process, Autocorrelation Function, Cross-correlation function and their properties, Ergodicity, Poisson process, Markov Process, Markov chain.

(8 Hours)

Module – 4

Testing Hypothesis: Testing of Hypothesis: Formulation of Null hypothesis, critical region, level of significance, errors in testing, Tests of significance for Large and Small Samples, t- distribution, its properties and uses, F-distribution, its properties and uses, Chi-square distribution, its properties and uses, $\chi 2$ – test for goodness of fit, $\chi 2$ test for Independence.

(8 Hours)

Module – 5

Symbolic Representation: 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, The M/M/s Queuing with Finite buffers.

(8 Hours)

Course Outcomes: The students should be able to:

CO1: Apply the use of probability functions/ models to solve problems.
CO2: Apply discrete and continuous probability distributions techniques.
CO3: Evaluate the impact of variability and uncertainty in stochastic systems.
CO4: Analyze queuing theory principles to solve real-world problems.

Question paper pattern:

- **SEE** will be conducted for 100 marks.
- Each full question is for 20 marks. (Answer five full questions out of 10 questions with intra modular choice). In every question, there will be a maximum of three sub-questions.
- **CIE** will be announced prior to the commencement of the course.
- 25 marks for the test. Average of three tests will be taken.
- 25 marks for Flexible Assessment Method.

Textbook:

1. Probability, Statistics and Queuing Theory, V. Sundarapandian, Eastern Economy Edition, PHI Learning Pvt. Ltd, 2009.

References:

1. Probability & Statistics with Reliability, Queuing and Computer Applications, 2nd Edition by Kishor. S. Trivedi, Prentice Hall of India ,2004.
2. Probability, Statistics and Random Processes, 1st Edition by P Kausalya, Pearson Education, 2013.

Web Resources:

1. "Probability Theory." Coursera, offered by University of London and London School of Economics, https://www.coursera.org/learn/probability-theory. Accessed 28 Aug. 2024.
2. "Introduction to Probability and Data." Coursera, offered by Duke University, https://www.coursera.org/learn/probability-data. Accessed 28 Aug. 2024.
3. "Stochastic Processes and Queueing Theory." Coursera, offered by National Research University Higher School of Economics, https://www.coursera.org/learn/stochastic- processes-queueing-theory. Accessed 28 Aug. 2024.

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS)

SEMESTER – II

Statistical Learning and Data Mining (3:0:0) 3

(Effective from the academic year 2024-25)

Course Code	24MCS23B	CIE Marks	50
Teaching Hours/Week (L:T:P)	3:0:0	SEE Marks	50
Total Number of Contact Hours	40	Exam Hours	3

Course Objectives: This course will enable students to:

- Explore advanced statistical learning and data mining techniques.
- Implement and evaluate sophisticated algorithms for regression, classification, clustering, and dimensionality reduction on real-world datasets.
- Apply advanced model assessment and resampling methods to enhance model performance and reliability.
- Utilize unsupervised learning methods to extract and interpret insights from complex data.

Preamble: This course delves into advanced statistical learning and data mining techniques, emphasizing complex data analysis, sophisticated model building, and predictive analytics. Students will integrate theoretical knowledge with practical applications using real-world datasets, preparing them for research or professional roles in data science and analytics.

Module – 1

Introduction: Foundations of Statistical Learning and Data Mining

Historical Evolution and Modern Applications, Differences and Synergies between Data Mining and Statistical Learning, Supervised vs Unsupervised Learning. Generalized Linear Models (GLMs), Regularization Techniques: Ridge and Lasso Regression.

(8 hours)

Module – 2

Model Assessment and Resampling Techniques: Advanced Cross-Validation Techniques (e.g., K-Fold, Leave-One-Out), Metrics for Model Evaluation: AUC, Log-Loss, Confusion Matrix, Bias-Variance Tradeoff Analysis

(8 hours)

Module – 3

Advanced Predictive Modeling: Decision Trees and Random Forests, Gradient Boosting Machines (GBM) and XGBoost. Kernel Methods for Nonlinear Classification, Neural Networks: Introduction to Deep Learning. Regularization and Optimization in Predictive Modeling.

(8 hours)

Module – 4

Data Preprocessing Techniques: Handling Missing Data, Outliers, and Scaling, Feature Engineering and Selection Strategies, Association Rule Mining (e.g., Apriori, Eclat), Density Estimation and Manifold Learning.

(8 hours)

Module – 5

Emerging Trends and Technologies in Data Mining: Advanced Data Mining Techniques-Data Mining in Big Data. Impact of AI and ML on data mining practices and applications. Quantum computing concepts and potential implications for data mining. Automated machine learning (AutoML), federated learning, and advanced analytics-Case Studies.

(8 hours)

Course Outcomes: The students will be able to

CO1: Understand advanced statistical learning and data mining techniques, including complex regression models, classification algorithms, and ensemble methods.

CO2: Apply sophisticated algorithms on real-world datasets and evaluate model performance using advanced assessment techniques and resampling methods.

CO3: Analyze advanced unsupervised learning methods, such as clustering and dimensionality reduction, to derive meaningful insights from complex data.

CO4: Examine various techniques to address complex case studies and critically assess emerging trends and future directions in data science.

Question paper pattern:

- SEE will be conducted for 100 marks.
- Each full question is for 20 marks. (Answer five full questions out of 10 questions with intra modular choice). In every question, there will be a maximum of three sub-questions.
- CIE will be announced prior to the commencement of the course.
- 25 marks for the test. Average of three tests will be taken.
- 25 marks for Flexible Assessment Method.

Textbooks:

1. James, Gareth, et al. An Introduction to Statistical Learning. Springer, 2021.
2. Witten, Ian H., et al. Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann, 2016.

References:

1. Hastie, Trevor, et al. *The Elements of Statistical Learning*. Springer, 2009.
2. Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2nd ed., Springer, 2009.
3. Bishop, Christopher M. Pattern Recognition and Machine Learning. Springer, 2006.
4. Kuhn, Max, and Kjell Johnson. Applied Predictive Modeling. Springer, 2013.
5. James, Gareth, Daniela Witten, Trevor Hastie, and Robert Tibshirani. An Introduction to Statistical Learning with Applications in R. Springer, 2013.
6. Han, Jiawei, Micheline Kamber, and Jian Pei. Data Mining: Concepts and Techniques. 3rd ed., Morgan Kaufmann, 2011.

Web Resources:

1. University of Colorado Boulder. Data Mining Foundations and Practice Specialization. Coursera, www.coursera.org/specializations/data-mining. Accessed 28 Aug. 2024.
2. Wesleyan University. Machine Learning for Data Analysis. Coursera, www.coursera.org/learn/machine-learning-data-analysis. Accessed 28 Aug. 2024.
3. DeepLearning.AI. Mathematics for Machine Learning and Data Science Specialization. Coursera, www.coursera.org/specializations/mathematics-machine-learning-data- science. Accessed 28 Aug. 2024.

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System(CBCS)

SEMESTER -II

Queuing Theory in Network Communication (3:0:0) 3

(Effective from the academic year 2024-25)

Course Code	24MCS23C	CIE Marks	50
Teaching Hours/Week (L:T:P)	3:0:0	SEE Marks	50
Total Number of Contact Hours	40	Exam Hours	3

Course Objectives: This course will enable students to:

1. Understand the concepts of queuing theory and its applications in telecommunications.
2. Analyze various queuing models and their performance measures.
3. Explore advanced topics in queuing theory related to modern network architectures and traffic management.
4. Analyze significance of advanced queuing theory in Communication Networks.

Preamble: This course provides a comprehensive understanding of queuing theory, focusing on its application in network traffic and telecommunications. The course includes theoretical concepts and advanced topics to equip students with the knowledge required for research and industry applications.

Module – 1

Introduction to Queuing Theory: Basics of Queuing Theory: Definitions, Terminologies, Queue Characteristics: Arrival Process, Service Process, Queue Discipline, Little's Law, Applications of Queuing Theory in Network Traffic and Telecommunications

(8 hours)

Module – 2

Markov Chains: The Exponential Distribution, he Poisson Process, Discrete Time Markov Chains, Continuous Time Markov Chains

(8 hours)

Module – 3

Markovian Queuing Models: Queues and stochastic processes, Poisson arrival process, Birth-death Markov chains, Notations for queuing systems, Little theorem and insensitivity property, M/M/1 Queue Analysis, M/M/1/K Queue Analysis, M/M/S Queue Analysis, M/M/S/S Queue Analysis, The M/M/S/S/P Queue Analysis, The M/M/∞ Queue Analysis, Solution of Markovian Queues Directly in the z- Domain, Multi-Dimensional Erlang-B Cases, Distribution of the Queuing Delays in the FIFO Case, Erlang-B Generalization for Non-Poisson Arrivals.

(8 hours)

Module – 4

Advanced Queuing Models: The M/G/1 Queuing Theory, M/G/1 System Delay Distribution, Numerical Inversion Method of the Laplace Transform of the Delay, Impact of the Service Time Distribution on M/G/1 Queue, M/G/1 Theory with State-Dependent Arrival Process, Applications of the M/G/1 Analysis to Slotted-Based Arrivals and Departures, Advanced M/G/1 Cases, Different Imbedding Options for M/G/1.

(8 hours)

Module –5

Network of Queues: Traffic Rate Equations, The Little Theorem Applied to the Whole Network, Tandem Queues and the Burke Theorem, The Jackson Theorem, Traffic Matrices, Network Planning Issues, Traffic Engineering and Network Optimization. Case studies.

(8 hours)

Course Outcomes: The students will be able to

CO1: Comprehend the basics of queuing theory and its relevance to network traffic and telecommunications.

CO2: Analyze and model different types of queues and evaluate their performance.

CO3: Apply queuing theory models to real-world network scenarios.

CO4: Understand advanced queuing models and techniques for efficient network traffic management.

Question paper pattern:

- **SEE** will be conducted for 100 marks.
- Each full question is for 20 marks. (Answer five full questions out of 10 questions with intra modular choice). In every question, there will be a maximum of three sub-questions.
- **CIE** will be announced prior to the commencement of the course.
- 25 marks for the test. Average of three tests will be taken.
- 25 marks for Flexible Assessment Method.

Textbooks:

1. D. Gross, J. F. Shortle, J. M. Thompson, C. M. Harris, "Fundamentals of Queueing Theory," 5th Edition, Wiley, 2018.
2. Giovanni Giambene, "Queuing Theory and Telecommunications: Networks and Applications", Springer, N.Y., 3rd Ed, 2021.

References:

1. L. Kleinrock, "Queueing Systems, Volume 1: Theory," Wiley, 1975.
2. Jeremiah F. Hayes, "Modeling and Analysis of Computer Communications Networks". Plenum Press.
3. H. Kobayashi, B. L. Mark, "System Modeling and Analysis: Foundations of System Performance Evaluation," Pearson.
4. R. Nelson, "Probability, Stochastic Processes, and Queueing Theory: The Mathematics of Computer Performance Modeling," Springer.
5. Kishore Trivedi, "Probability & Statistics with Reliability Queuing and Computer Science Applications", Wiley.
6. D. Bertsekas and R. Gallager, "Data Networks", Prentice Hall.
7. Haruo Akimaru and Konosuke Kawashima, "Tele traffic Theory and Applications", Springer.

Web Resources:

1. https://onlinecourses.nptel.ac.in/noc24_ee132/preview
2. https://onlinecourses.nptel.ac.in/noc24_ma97/preview
3. https://onlinecourses.nptel.ac.in/noc24_ee124/preview
4. https://onlinecourses.nptel.ac.in/noc24_ee119/preview

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System(CBCS)

SEMESTER -II

Graph Algorithms and Mining (3:0:0) 3

(Effective from the academic year 2024-25)

Course Code	24MCS23D	CIE Marks	50
Teaching Hours/Week (L: T: P)	3:0:0	SEE Marks	50
Total Number of Contact Hours	40	Exam Hours	3

Course Objectives: This course will enable students to:

1. To provide the basic concepts and important properties of graphs.
2. To learn and explore several methods on algorithms such as graph traversal, shortest paths, minimum spanning tree
3. To introduce students to the field of graph mining and its application in various domains.
4. To give the students an opportunity to obtain hands-on experience on applications of graph mining.

Preamble: Graph algorithms and mining focus on analyzing and extracting useful information from graph-structured data, where entities are represented as nodes and their relationships as edges. Graph algorithms, such as depth-first search, breadth-first search, shortest path, and PageRank, are used to traverse and analyze these structures, uncovering patterns like community detection, node centrality, and connectivity. Graph mining involves identifying frequent subgraphs, motifs, and anomalies within large and complex networks, such as social networks, biological networks, and communication networks. This field is essential in domains like social network analysis, bioinformatics, recommendation systems, and fraud detection, where understanding the relationships between entities is crucial.

Module – 1

Introduction to Graphs: Introduction to graphs and basic terminology, Representations of a graph, types of graphs, basic algorithms for decomposing graphs into parts, connectivity of graphs, matching on graphs.

(8 hours)

Module – 2

Graph Algorithms: Graph coloring, graphs on surface, directed graphs, Shortest path algorithms, algorithms to discover minimum spanning tree, Flows in Networks and some important flow algorithms, Searching Graphs and Related algorithms.

(8 hours)

Module – 3

Graph Mining: Motivation for Graph Mining, Applications of Graph Mining, Mining Frequent Subgraphs –Transactions, BFS/Apriori Approach (FSG and others), DFS Approach (gSpan and others), Diagonal and Greedy Approaches, Constraint-based mining and new algorithms, Mining Frequent Subgraphs, graph visualizations.

(8 hours)

Module – 4

Graph Clustering: Spectral Clustering-Community Detection Algorithms: Girvan-Newman Hierarchical Clustering: Agglomerative and Divisive Approaches-Evaluation Metrics: Modularity, Conductance, Silhouette ScoreS- Graph Embedding: Node2Vec, DeepWalk, Scalable Clustering Algorithms: Louvain Method, Infomap.

(8 hours)

Module - 5

Graph Classification and Knowledge Graphs: Hierarchical Graph Classification: Handling Graphs with Hierarchical Structures- Graph Contrastive Learning: Self-Supervised Learning, Contrastive Loss for Graphs- Semantic Networks- Biological Networks: Disease Gene Prediction, Drug Interaction Networks-Knowledge Graphs: Entity Classification, Relation Prediction-Case Study using Knowledge Graphs.

(8 hours)

Course outcomes: The students will be able to

CO1: Apply graph algorithms to solve problems in various domains and techniques.
CO2: Analyze graph mining methods.
CO3: Solve graph-related problems.
CO4: Apply graph mining algorithms for large-scale datasets on various domains.

Question paper pattern:

- **SEE** will be conducted for 100 marks.
- Each full question is for 20 marks. (Answer five full questions out of 10 questions with intra modular choice). In every question, there will be a maximum of three sub-questions.
- **CIE** will be announced prior to the commencement of the course.
- 25 marks for the test. Average of three tests will be taken.
- 25 marks for Flexible Assessment Method.

Textbooks:

1. Aggarwal, Charu C., and Haixun Wang, editors. Managing and Mining Graph Data. Springer, 2010.
2. Easley, David, and Jon Kleinberg. Networks, Crowds, and Markets: Reasoning about a Highly Connected World. Cambridge University Press, 2010.
3. Hamilton, William L. Graph Representation Learning. Morgan & Claypool Publishers, 2020.
4. Fortunato, Santo, and Markos Hric. "Community Detection in Networks: A User Guide." Physics Reports, vol. 659, 2016, pp. 1-44.

References:

1. Sarkar, Deepayan. Graph Mining: Laws, Tools, and Case Studies. Wiley, 2016.
2. Zhang, X., and H. Zhou. Graph-Based Semi-Supervised Learning. Springer, 2011.
3. Agerri, R., and M. Moens. Mining Graph Data. Wiley, 2015.
4. Barabási, Albert-László. Network Science. Cambridge UP, 2016.
5. Han, Jiawei, Micheline Kamber, and Jian Pei. Data Mining: Concepts and Techniques. Morgan Kaufmann, 2011.

Web Resources:

1. "Graph Theory." Coursera, University of California, San Diego, https://www.coursera.org/learn/graph-theory.
2. "Data Mining." Coursera, University of Illinois at Urbana-Champaign, https://www.coursera.org/learn/data-mining.

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS)

SEMESTER – II

Cyber Security and Forensics (3:0:0) 3

(Effective from the academic year 2024-25)

Course Code	24MCS23E	CIE Marks	50
Teaching Hours/Week (L:T:P)	3:0:0	SEE Marks	50
Total Number of Contact Hours	40	Exam Hours	3

Course Objectives: This course will enable students:

1. To interpret the concepts of cyber security, forensics and its applications in different contexts.
2. To investigate incidents and areas affected due to cybercrime.
3. To illustrate tools used in cyber security,forensic
4. To infer legal perspectives in cyber security
5. To apply the policies, security standards, and IPR issues on a cybercrime incident.

Preamble: The course aims to provide an overview of cyber law, security, tools, and approaches to secure resources and manage intellectual property for enhancing the competitiveness for organizations. Upon completion of this course, students should be able to accomplish the course outcomes defined. Cyber security and forensics have direct impact on the security systems, society, financial models and affecting the GDP.

Module – 1

Introduction to Cybercrime: Cybercrime Definition and Origins of the Word, Cybercrime and Information Security, Who are Cybercriminals?, Classifications of Cybercrimes, Cybercrime: The Legal Perspectives, Cybercrimes: An Indian Perspective, Cybercrime and the Indian ITA 2000, A Global Perspective on Cybercrimes, Cybercrime Era: Survival Mantra for the Netizens. Cyber offenses: How Criminals Plan Them: How Criminals Plan the Attacks, Social Engineering, Cyberstalking, Cybercafe and Cybercrimes, Botnets: The Fuel for Cybercrime, Attack Vector.

(8 Hours)

Module – 2

Cybercrime: Mobile and Wireless Devices: Introduction, Proliferation of Mobile and Wireless Devices, Trends in Mobility, Credit Card Frauds in Mobile and Wireless Computing Era, Security Challenges Posed by Mobile Devices, Registry Settings for Mobile Devices, Authentication Service Security, Attacks on Mobile/Cell Phones, Mobile Devices: Security Implications for organizations, Organizational Measures for Handling Mobile, Organizational Security Policies and Measures in Mobile Computing Era, Laptops.

(8 Hours)

Module – 3

Tools and Methods Used in Cybercrime: Introduction, Proxy Servers and Anonymizers, Phishing, Password Cracking, Keyloggers and Spywares, Virus and Worms, Trojan Horses and Backdoors, Steganography, DoS and DDoS Attacks, SQL Injection, Buffer Overflow, Attacks on Wireless Networks. Phishing and Identity Theft: Introduction, Phishing, Identity Theft (ID Theft).

(8 Hours)

Module – 4

Understanding Computer Forensics: Introduction, Historical Background of Cyberforensics, Digital Forensics Science, The Need for Computer Forensics, Cyber forensics and Digital Evidence, Forensics Analysis of E-Mail, Digital Forensics Life Cycle, Chain of Custody Concept, Network Forensics, Approaching a Computer Forensics Investigation, Setting up a Computer Forensics Laboratory: Understanding the Requirements, Computer Forensics and Steganography, Relevance of the OSI 7 Layer Model to Computer Forensics, Forensics and Social Networking Sites: The Security/Privacy Threats, Computer Forensics from Compliance Perspective, Challenges in Computer Forensics, Special Tools and Techniques, Forensics Auditing, Antiforensics.

(8 Hours)

Module – 5

Introduction to Security Policies and Cyber Laws: Need for An Information Security Policy, Information Security Standards – ISO, Introducing Various Security Policies and Their Review Process, Introduction to Indian Cyber Law, Objective and Scope of the IT Act, 2000, Intellectual Property Issues, Overview of 20082020 / 23 Intellectual - Property - Related Legislation in India, Patent, Copyright, Law Related to Semiconductor Layout and Design, Software License.

(8 Hours)

Course Outcomes: The students will be able to:

CO1: Identify and analyze the cyber security risks due to different cybercrimes and examine them from a legal perspective.
CO2: Illustrate the use of Cyber security and of cyber-forensics tools in investigating the given cybercrime.
CO3: Analyze legal issues and socio-economic impact due to cybercrime and forensics investigation approach
CO4: Examine relevant network defense / web application tools to solve given cyber security problems/ case study.
CO5: Design the security policy for an organization in line with IT ACT 2000 and based on ISO standard.

Question paper pattern:

- **SEE** will be conducted for 100 marks.
- Each full question is for 20 marks. (Answer five full questions out of 10 questions with intra modular choice). In every question, there will be a maximum of three sub-questions.
- **CIE** will be announced prior to the commencement of the course.
- 25 marks for test. Average of three tests will be taken.
- 25 marks for Flexible Assessment Method.

Textbooks:

1. Sunit Belapure, Nina Godbole, Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives Wiley India Pvt Ltd 2013.
2. Surya Prakash Tripathi, Ritendra Goyal, Praveen Kumar Shukla, Introduction to information security and cyber laws, Dreamtech Press 2015.

References:

1. Thomas J. Mowbray, Cybersecurity: Managing Systems, Conducting Testing, and Investigating Intrusions John Wiley & Sons 2013.
2. James Graham, Ryan Olson, Rick Howard, Cyber Security Essentials CRC Press 2010.

Web Resources:

1. Cybersecurity Specialization by the University of Maryland: University of Maryland. "Cybersecurity Specialization". www.coursera.org/specializations/cyber- security.
2. Introduction to Cyber Security Specialization by NYU: NYU. "Introduction to Cyber Security Specialization". www.coursera.org/specializations/intro-cyber- security.
3. Computer Forensics by the University of London: University of London. "Computer Forensics". www.coursera.org/learn/computer-forensics.
4. Digital Forensics and Cyber Investigation by IBM: IBM. "Digital Forensics and Cyber Investigation". www.coursera.org/learn/digital-forensics-cyber-

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS)

SEMESTER – II

Secure Cloud Computing (3:0:0) 3

(Effective from the academic year 2024-25)

Course Code	24MCS23F	CIE Marks	50
Teaching Hours/Week (L:T:P)	3:0:0	SEE Marks	50
Total Number of Contact Hours	50	Exam Hours	3

Course Objectives: This course will enable students to:

1. Explore Cloud Computing Models and Architectures
2. Identifying Cloud Security Risks and Threats
3. Interpret Compliance and Legal Issues
4. Developing Cloud Security Policies and Procedures
5. Evaluating Cloud Security Solutions and Vendors

Preamble: Cloud security refers to the set of policies, technologies, and controls designed to protect data, applications, and infrastructure associated with cloud computing. As organizations increasingly rely on cloud services for storing sensitive data and running critical applications, ensuring robust cloud security has become essential. Key aspects of cloud security include data encryption, identity and access management, threat detection, and compliance with regulatory standards. With the rise of sophisticated cyber threats, cloud security measures must continuously evolve to safeguard against data breaches, unauthorized access, and other vulnerabilities that could compromise the integrity and confidentiality of cloud-based systems.

Module – 1

Introduction to Cloud Computing: Overview of Cloud Computing, Cloud Computing Models and Services, Cloud Computing Architecture, Security Challenges in the Cloud. **Cloud Security Fundamentals:** - Basic Cloud Security Concepts, Cloud Security Standards and Frameworks, Key Security Considerations for Cloud Environments.

(8 hours)

Module – 2

Identity and Access Management (IAM): IAM Concepts and Technologies, Cloud-Based IAM Solutions, Authentication and Authorization in the Cloud, Managing User Identity and Access. **Data Security and Privacy:** Data Protection Techniques, Encryption and Key Management, Privacy Laws and Regulations, Managing Sensitive Data in the Cloud.

(8 hours)

Module – 3

Network Security: Cloud Network Architecture, Network Security Controls and Technologies, Protecting Data in Transit, Managing Network Security Threats. **Compliance and Risk Management:** Regulatory Compliance in the Cloud, Risk Management Strategies, Cloud Provider Compliance and Certifications, Auditing and Reporting.

(8 hours)

Module – 4

Incident Response and Forensics: Incident Response Planning, Forensic Techniques and Tool, Legal and Ethical Issues in Cloud Forensics, Case Studies and Practical Considerations. **Emerging Trends and Future Directions:** - Advances in Cloud Security Technologies, The Role of Artificial Intelligence and Machine Learning Future Challenges and Opportunities.

(8 hours)

Module – 5

Case Studies and Practical Applications: Case Studies of Cloud Security Incidents, Best Practices and Lessons Learned, Implementing Security Solutions in Real-World Scenarios.

(8 hours)

Course outcomes: The students will be able to

CO1: Understand various cloud computing models (IaaS, PaaS, SaaS) and deployment types (public, private, hybrid).
CO2: Analyze common security risks and threats in cloud environments.
CO3: Implement effective cloud security controls, including access controls, encryption, and data protection strategies.
CO4: Apply compliance requirements and legal considerations relevant to cloud security.
CO5: Apply learned concepts to real-world scenarios and case studies involving cloud security.

Question paper pattern:

- **SEE** will be conducted for 100 marks.
- Each full question is for 20 marks. (Answer five full questions out of 10 questions with intra modular choice). In every question, there will be a maximum of three sub-questions.
- **CIE** will be announced prior to the commencement of the course.
- 25 marks for the test. Average of three tests will be taken.
- 25 marks for Flexible Assessment Method.

Textbooks:

1. Ronald L. Krutz and Russell Dean Vines. Cloud Security: A Comprehensive Guide to Secure Cloud Computing. Wiley India Pvt. Ltd, 2010.
2. Mather, Tim, et al. Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance. O'Reilly Media, 2009.
3. Vacca, John R., editor. Cloud Computing Security: Foundations and Challenges. CRC Press, 2017.

References:

1. Kavis, Michael J. Architecting the Cloud: Design Decisions for Cloud Computing Success. Wiley India Pvt. Ltd, 2014.
2. Coombs, Ted, and Justin R. Smith. Cloud Security For Dummies. John Wiley & Sons, 2022.

Web Resources:

1. Diogenes, Yuri. Introduction to Cloud Security with Microsoft Azure. Coursera, Microsoft. Accessed 30 Aug. 2024.
2. Google Cloud Training. Google Cloud Platform Security Essentials. Coursera, Google Cloud. Accessed 30 Aug. 2024.

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS)

SEMESTER – II

Blockchain Technologies (3:0:0) 3

(Effective from the academic year 2024-25)

Course Code	24MCS23G	CIE Marks	50
Teaching Hours/Week (L:T:P)	3:0:0	SEE Marks	50
Total Number of Contact Hours	40	Exam Hours	3

Course Objectives: The objective of this course is to:

1. Provide conceptual understanding of block chain technology and how it can be used in Industry 4.0
2. Explore the operations of the Blockchain technology and its consensus mechanisms
3. Demonstrate and interpret working of Ethereum
4. Analyze different case studies of Blockchain beyond crypto currency

Preamble: Block chain technology has the potential to revolutionize interactions between governments, businesses and citizens. Block chain drew global attention in terms of the secured deployment of various services across multiple industries. National Governance facing unique challenges in various modalities are addressed by block chain leading improvement. Block chain technology has the potential to boost GDP over the next decade.

Module - 1

Introduction: The Growth of Blockchain Technology **Distributed Systems** ,The history of Blockchain and bitcoin ,Blockchain definition, architecture, Generic Elements, Benefits, features, limitations and types of Blockchain, Consensus, CAP Theorem and Blockchain. **Symmetric Cryptography:** Introduction, Cryptographic primitives: keyless primitives, Random numbers, hash functions, Design of SHA- 256 and Symmetric cryptography, HMACs. Asymmetric key Cryptography techniques: Digital signatures, and RSA digital signature, Applications of Cryptographic hash functions.

(8 Hours)

Module - 2

Decentralization: Decentralization using Blockchain, Methods of decentralization, Routes to decentralization, Blockchain and full ecosystem decentralization, pertinent terminologies, brief on Ethereum platform for decentralization. **Consensus:** Introducing the consensus problem, The Byzantine Generals Problem, fault tolerance, State Machine Replication, FLP impossibility, Practical Byzantine Fault Tolerance. Working of Different Consensus Mechanisms- Proof of Work, Proof of Stake, Proof of Elapsed Time, Proof of Importance, Proof of Activity, Proof of Capacity, Proof of Storage, Proof of Authority. Detailed discussion on Proof of Work and Proof of stake.

(8 Hours)

Module - 3
Introducing Bitcoin: Bitcoin definition, A user's perspective **Blockchain** The genesis block, What is a Double spending problem? **Mining**: Task of the miners, mining rewards, proof of work, The mining algorithm. **Ehereum:** Blocks and Blockchain, The genesis block, The block validation mechanism, Block finalization, Gas. **Wallets and client software:** Wallets, Geth-Installation and Usage, MetaMask-Installation, creating and funding an account using MetaMask, **Nodes and miners**. (8 Hours)
Module - 4
Ethereum Development Environment: Overview, Test Networks, Components of a private network, Network ID, the genesis file, data Directory, Starting up the private network, mining on the private network, Remix IDE, MetaMask, using Metamask and Remix IDE to deploy a smart contract. **Development Tools and Frameworks:** Languages, Compilers, Tools and Libraries, Frameworks, Contract development and deployment, The Layout of Solidity source code file, The Solidity language. (8 Hours)
Module - 5
Use Cases / Case Studies **IOT -** Architecture, Benefits of convergence. **Government** – Border control, Voting, Citizen identification (ID cards) **Health**, **Finance-**Insurance, Post-trade settlement, Financial crime prevention, Payments, Loans. **Challenges**-Scalability, Privacy and Security **Areas to Address**-Regulation, Illegal Activity, privacy or transparency, Blockchain and AI. (8 Hours)
Course Outcomes: The students will be able to: CO1: Summarize the fundamental components and functional aspects of the Blockchain. CO2: Apply the development Tool and Framework for the given problem. CO3: Compare among various consensus mechanisms and justify suitable mechanisms for a given problem. CO4: Analyze the given Use cases / case studies of the Blockchain technology.
Question paper pattern: • **SEE** will be conducted for 100 marks. • Each full question is for 20 marks. (Answer five full questions out of 10 questions with intra modular choice). In every question, there will be a maximum of three sub-questions. • **CIE** will be announced prior to the commencement of the course. • 25 marks for the test. Average of three tests will be taken. • 25 marks for Flexible Assessment Method

Textbooks:

1. Mastering Blockchain: A deep dive into distributed ledgers, consensus protocols, smart contracts, DApps, cryptocurrencies, Ethereum, and more, 3rd Edition, Imran Bashir, Packt Publishing, 2020, ISBN: 9781839213199

References:

1. Arshdeep Bahga, Vijay Madisetti, ―Blockchain Applications: A Hands-On Approach, Arshdeep Bahga, Vijay Madisetti publishers 2017.

Web Resources:

1. https://onlinecourses.swayam2.ac.in/aic21_ge01/preview
2. "Blockchain Basics." Coursera, ConsenSys Academy, https://www.coursera.org/learn/blockchain-basics.
3. "Blockchain Revolution." Coursera, INSEAD,
4. https://www.coursera.org/learn/blockchain- revolution.
5. "Blockchain and Cryptocurrency Explained." Coursera, University at Buffalo, https://www.coursera.org/learn/blockchain-cryptocurrency.

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS)

SEMESTER – II

Secure Software Development (3:0:0) 3

(Effective from the academic year 2024 -2025)

Course Code	24MCS23H	CIE Marks	50
Teaching Hours/Week (L:T:P)	3:0:0	SEE Marks	50
Total Number of Contact Hours	40	Exam Hours	3

Course Objectives: This course will enable students to:

1. To learn the development principles and process models of secure software engineering.
2. To study the requirements, modeling, design testing and validation procedures that ensure security.
3. To apply secure software engineering principles across cross-disciplines.

Preamble: Secure Software Development is essential due to the growing complexity and sophistication of cyber threats, which threaten national security and economic stability. Economically, it prevents costly data breaches and builds consumer trust, supporting the growth of digital markets. Technologically, it underpins innovation by ensuring systems are resilient against attacks, enabling the safe implementation of advanced technologies like smart cities and e-governance. By fostering secure software practices, nations can enhance technological advancement, protect critical infrastructure, and promote sustainable economic growth, making it a strategic component of nation-building.

Module - 1

Introduction: What is System engineering-Systems engineering and the systems-System engineering processes- Understanding Software systems engineering-The software system engineering processes-Steps in the software development processes-Functional and non-functional requirements Verification and validation. **Engineering secure and safe systems:** Introduction-The approach-security versus safety-Four approaches to develop critical systems- The dependability approach-The safety engineering approach-The secure systems approach- The real- time systems approach Security-critical and safety-critical systems

(9 Hours)

Module - 2

Architecting Secure Software Systems: Security Requirements Analysis, Threat Modelling, Security Design Patterns Anti-Patterns, Attack Patterns, Security Design Patterns, Authentication, Authorization -Security Coding Security Algorithm, Security Protocol, Key Generation

(7 Hours)

Module – 3

Validating Security: Generating the Executable, Security Testing vulnerability assessment, code coverage tools - Secured Deployment, Security Remediation, Security Documentation, Security Response Planning, Safety-Critical Systems. **Secure Coding Principles:** Coding in C String manipulation, vulnerabilities and exploits, Pointers based vulnerabilities. Memory management, common errors, Integer Security, Double free Vulnerabilities.

(8 Hours)

Module - 4

Security in web-facing applications: Overview of web security, Identity Management, public key infrastructure, Code injection, Parameter tampering, secured web programming, application vulnerability description language

(8 Hours)

Module - 5

Security and safety metrics: Defining metrics-Differentiating measures and metrics -Software Metrics-Measuring and reporting metrics- Metrics for meeting requirements-Risk metrics-Security metrics for software systems-Safety metrics for software systems

(8 Hours)

Course Outcomes: The students should be able to:

CO1: Evaluate a secure software development process including designing secure applications, writing secure code against attacks.
CO2: Assess the reports through security testing procedures.
CO3: Solve the security issues of vulnerabilities, flaws, and threats.
CO4: Identify and use the standard Secure Coding Principles for designing secure software systems.
CO5: Develop secure web programming to enhance the software code more resistant to attacks.
CO6: Identify the need of Security and safety metrics.

Question paper pattern:

- **SEE** will be conducted for 100 marks.
- Each full question is for 20 marks. (Answer five full questions out of 10 questions with intra modular choice). In every question, there will be a maximum of three sub-questions.
- **CIE** will be announced prior to the commencement of the course.
- 25 marks for the test. Average of three tests will be taken.
- 25 marks for Flexible Assessment Method.

Textbook:

1. Asoke K. Talukder, Manish Chaitanya, Architecting Secure Software Systems, ISBN 9781420087840, 2008.
2. Engineering Safe and Secure Software Systems Book by C. Warren Axelrod Artech House Publishers; Unabridged edition (30 November 2012) ISBN-13 : 978-1608074723

References:

1. Software Security Principles. Policies, and Protection Mathias Payer July 2021, v0.37.
2. Security Engineering: A Guide to Building Dependable Distributed Systems by Ross J. Anderson: Anderson, Ross J. Security Engineering: A Guide to Building Dependable Distributed Systems. Wiley, 2020.

Web Resources:

1. Software Security by the University of California, Irvine: University of California, Irvine. “Software Security.” Coursera, www.coursera.org/learn/software-security.
2. Secure Software Design by the University of Colorado Boulder: University of Colorado Boulder. “Secure Software Design.”Coursera, www.coursera.org/learn/secure-software-design.

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS)

SEMESTER - II

Augmented and Virtual Reality (3:0:0) 3

(Effective from the academic year 2024-25)

Course Code	24MCS24A	CIE Marks	50
Teaching Hours/Week (L:T:P)	3:0:0	SEE Marks	50
Total Number of Contact Hours	40	Exam Hours	3

Course Objectives: This course will enable students to:

1. Explore scientifically sound principles of Augmented and Virtual Reality.
2. Compare and Contrast technologies in the context of AR and VR systems design.
3. Demonstrate the knowledge of the input devices, tracking and output devices for both compositing and interactive applications.
4. Analyze the use of objects for managing large scale Virtual Reality environments in real time.

Preamble: In an era where digital innovation continuously reshapes our interaction with the world, Augmented Reality (AR) and Virtual Reality (VR) stand at the forefront of this technological evolution. These immersive technologies offer profound possibilities for both industry and daily life, blending the virtual and real worlds to create new experiences and solutions.

Module - 1

Introduction to Virtual and Augmented Reality: What is Virtual Reality (VR)? What is Augmented Reality (AR)? What is the purpose of VR/AR? What are the basic concepts? What are the hard- and software components of VR/AR systems? How has VR/AR developed historically**?**

(8 hours)

Module - 2

Perceptual Aspects of VR and Virtual World: VR phenomena-double vision and cybersickness. human perception processes, human information processing, different limitations of human perception, Virtual worlds, the contents of VR environments, dynamic behavior of 3D objects. interactions with 3D objects.

(8 hours)

Module - 3

VR/AR Input Devices, Tracking and Output Devices: How do Virtual Reality (VR) and Augmented Reality (AR) systems recognize the actions of users, know where the user is, track objects in their movement, input devices for VR and AR. Output devices and technologies for VR and AR. Devices for visual output play, stationary displays, acoustic and haptic outputs.

(8 hours)

Module - 4

Interaction in Virtual Worlds, Real-Time Aspects of VR Systems: Design and realization of interaction and the resulting user interface of a VR/AR system, system control, selection, manipulation and navigation, real-time capability of VR systems., types of latencies, efficient collision detection.

(8 hours)

Module - 5

Authoring and Mathematical Foundations of VR/AR Applications: Authoring of VR and AR applications, the authoring process and the use of the tools, mathematical methods offer fundamental principles to model three-dimensional space.

(8 hours)

Course outcomes: The students will be able to

CO1: Review the Fundamental concepts of Virtual and Augmented Reality with hard and soft components and history.
CO2: Design the Perceptual Aspects of VR and Virtual World.
CO3: Describe the input devices, tracking and output devices in AR-VR Applications.
CO4: Summarize the interaction and real aspect of AR VR systems.
CO5: Articulate and illustrate the applications in authorizing and mathematical aspects of AR- VR tools

Question paper pattern:

- **SEE** will be conducted for 100 marks.
- Each full question is for 20 marks. (Answer five full questions out of 10 questions with intra modular choice). In every question, there will be a maximum of three sub-questions.
- **CIE** will be announced prior to the commencement of the course.
- 25 marks for the test. Average of three tests will be taken.
- 25 marks for Flexible Assessment Method.

Textbooks:

1. Ralf Doerner, Wolfgang Broll, Paul Grimm, Bernhard Jung: Virtual and Augmented Reality (VR/AR)-Foundations and Methods of Extended Realities (XR)-springers-2022.

References:

1. Schmalstieg D. and Hollerer T., Augmented and Virtual Reality, Addison-Wesley (2016).
2. Aukstakalnis S., Practical Augmented Reality: A Guide to the Technologies, Applications, and Human Factors for AR and VR, Addison-Wesley (2016).
3. Erin Pangilinan, Steve Lukas, Vasanth Mohan: Creating Augmented and Virtual Realities: Theory and Practice for Next-Generation Spatial Computing.
4. Doug A. B., Kruijff E., LaViola J. J. and Poupyrev I. , 3D User Interfaces: Theory and Practice , Addison-Wesley (2005,201lp) 2nd ed.
5. Parisi T., Learning Virtual Reality, O'Reilly (2016) 1st ed.

Whyte J., Virtual Reality and the Built Environment, Architectural Press (2002).

Web Resources:

1. Introduction to Augmented Reality and ARCore by Google: Google. “Introduction to Augmented Reality and ARCore.” Coursera, www.coursera.org/learn/augmented-reality-arcore.
2. Virtual Reality Specialization by the University of London: University of London. “Virtual Reality Specialization.” Coursera, www.coursera.org/specializations/virtual-reality.
3. Building AR/VR Experiences with Unity" by the University of London: University of London. “Building AR/VR Experiences with Unity.” Coursera, www.coursera.org/learn/building-arvr-experiences-unity.

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS)

SEMESTER – II

Spatial Computing and Mixed Reality (3:0:0) 3

(Effective from the academic year 2024-25)

Course Code	24MCS24B	CIE Marks	50
Teaching Hours/Week (L:T:P)	3:0:0	SEE Marks	50
Total Number of Contact Hours	40	Exam Hours	3

Course Objectives: This course will enable students to:

1. Explore the fundamental technologies that drive spatial computing and how they differ from traditional computing paradigms.
2. Discuss the diverse applications of spatial computing technologies across various sectors, showcasing current innovations and their impact.
3. Examine the broader societal implications, including ethical considerations, privacy concerns, and the effects on workforce skills and accessibility.
4. Analyze the emerging trends, future directions, and the potential transformative effects of spatial computing technologies on various industries and society at large.

Preamble:. immersive experiences that integrate seamlessly with our physical environments. Augmented Reality (AR), Virtual Reality (VR), and Mixed Reality (MR) are not just technological advancements; they represent a fundamental shift in how we perceive and interact with information. These technologies have the potential to revolutionize industries, redefine entertainment, and enhance our daily lives in ways previously imagined only in science fiction.

Module – 1

The Evolution of Spatial Computing: Historical Background, Key Milestones and Technological Advances, The Rise of Immersive Technologies. **Understanding Spatial Computing**: Defining Spatial Computing, Core Technologies: AR, VR, and MR,- How Spatial Computing Differs from Traditional Computing.

(8 hours)

Module – 2

Augmented Reality (AR): Overview of AR Technologies, AR Hardware and Software, Applications and Use Cases, Future Trends in AR. **Virtual Reality (VR):** VR Fundamentals and Technology, Immersive VR Experiences, VR Hardware and Development, The Role of VR in Training and Simulation.

(8 hours)

Module – 3

Mixed Reality (MR): Defining MR and Its Components, MR Hardware and Interaction Models, Applications in Industry and Entertainment, Challenges and Opportunities in MR. **The Convergence of Spatial Computing Technologies:** Integrating AR, VR, and MR,Cross, Platform Experiences, Synergies and Technological Innovations. **Spatial Computing in Different Sectors:** Healthcare and Medicine, Education and Training, Entertainment and Gaming, Manufacturing and Industry, Urban Planning and Smart Cities.

(8 hours)

Module – 4

The Impact of Spatial Computing on Society: Ethical Considerations and Privacy Issues, Social and Psychological Impacts, Accessibility and Inclusivity, The Future Workforce and Skills. **Designing for Immersive Experiences:** Principles of Immersive Design, User Experience (UX) in Spatial Computing, Challenges in Designing for Immersive Environments, Case Studies of Successful Designs.

(8 hours)

Module – 5

Developing Spatial Computing Applications: Software Development Tools and Platforms, Programming for AR, VR, and MR , Best Practices and Development Strategies, Case Studies of Innovative Applications. **The Future of Spatial Computing:** Emerging Trends and Technologies, Predictions and Vision for the Future, Research and Development Directions, The Role of AI and Machine Learning.

(8 hours)

Course Outcomes: The students will be able to

CO1: Analyze spatial computing technologies to understand their capabilities and applications.
CO2: Evaluate mixed reality applications for effectiveness in enhancing user experiences.
CO3: Design immersive mixed reality experiences integrating virtual and real-world elements.
CO4: Implement spatial computing solutions using relevant tools and technologies.

Question paper pattern:

- **SEE** will be conducted for 100 marks.
- Each full question is for 20 marks. (Answer five full questions out of 10 questions with intra modular choice). In every question, there will be a maximum of three sub-questions.
- **CIE** will be announced prior to the commencement of the course.
- 25 marks for the test. Average of three tests will be taken.
- 25 marks for Flexible Assessment Method.

Textbooks:

1. Immersive Futures: Exploring Spatial Computing Technologies: Morgan Lee, Mar 2024.

References:

1. The Future of Technology in Education: How AI Will Transform the Learning and Teaching Process Forever Hardcover – Import, 6 May 2020, by Harib Shaqsy.
2. Spatial Computing: An Ai-driven Business Revolution, by Cathy Hackl and Irene Cronin | 10 June 2024.
3. Creating Augmented and Virtual Realities: Theory and Practice for Next-Generation Spatial Computing (Greyscale Indian Edition) by Erin Pangilinan , Steve Lukas, et al. | 31 March 2019.

Web Resources:

1. Spatial Computing and Augmented Reality" by the University of London: University of London. “Spatial Computing and Augmented Reality.” Coursera, www.coursera.org/learn/spatial-computing-augmented-reality.
2. www.coursera.org/learn/augmented- reality-arcore.
3. www.coursera.org/specializations/virtual-reality.

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS)

SEMESTER – I

3D Modeling and Animation for AR/VR (3:0:0) 3

(Effective from the academic year 2024-25)

Course Code	24MCS24C	CIE Marks	50
Teaching Hours/Week (L:T:P)	3:0:0	SEE Marks	50
Total Number of Contact Hours	40	Exam Hours	3

Course Objectives: This course will enable students:

1. To explore the concepts of 3D- Three Dimension and Animation.
2. To gain Theoretical knowledge of how to create a Three-dimensional (3D) Environment.
3. To demonstrate the ability to map detailed textures to 3D objects in a theoretical way.
4. Compare and Contrast on Lighting and Rendering for the 3D objects and 3D environment.

Preamble: to equip ourselves with the knowledge and skills necessary to excel in the field of 3D design and animation, setting the stage for both theoretical understanding and practical application. three-dimensional (3D) technology is essential for creating immersive and realistic virtual environments.

Module – 1

Understanding 3D Space: Cartesian Mapping and 3D Coordinates, The Grid, Global and Local Coordinate System, Transforms, Pivots and Snaps, Freezing and resetting transforms
Exercise: Transforming objects in space with MAYA.

(8 hours)

Module – 2

Polygon Geometry: Basic Polygon concepts, triangulation and polygons, polygon primitives, Sub object Editing, Chamfer and Bevel, Extrude, Advanced polygons modeling tools, smoothing.
Exercise: Modeling with polygons Tools.

(8 hours)

Module – 3

NURBS and Curve-Based Geometry: Curve, Nurbs Curve, Projected Curves and Trim Surfaces, NURBS into polygons.
Exercise: The wine glass.

(8 hours)

Module – 4

Lighting, Materials, Textures, and UVs: GPU vs CPU rendering, light, camera, Materials, Polygon normal, light and Lighting, Map Shadows, Textures, UV Mapping, Software Rendering.
Exercise: Creating a Complex Material.

(8 hours)

Module – 5

Animation: Definition and Basic Concepts, Keyframes and Keyframing, Pose-Based Animation, Rotoscoping and motion Capture Graphs Curves and Tangents.

(8 hours)

Course Outcomes: The students will be able to

CO1: Apply best practices in 3D modeling and animation to address specific AR/VR project requirements.

CO2: Analyze the impact of 3D assets on user immersion and interactivity in AR/VR applications.

CO3: Analyze principles of lighting and rendering, crucial for achieving realistic visual effects and enhancing the overall quality of 3D scenes.

CO4: Apply detailed 3D models for AR/VR applications to animate models to produce realistic and interactive animations for immersive environments.

Question paper pattern:

- **SEE** will be conducted for 100 marks.
- Each full question is for 20 marks. (Answer five full questions out of 10 questions with intra modular choice). In every question, there will be a maximum of three sub-questions.
- **CIE** will be announced prior to the commencement of the course.
- 25 marks for the test. Average of three tests will be taken.
- 25 marks for Flexible Assessment Method.

Textbooks:

1. Essential Skills for 3D Modeling, Rendering, and Animation by Zeman, Nicholas Bernhardt, Boca Raton : Taylor & Francis, Paperback – 6 November 2014.

References:

1. The Animator's Survival Kit: A Manual of Methods, Principles and Formulas for Classical, Computer, Games, Stop Motion and Internet Animators 4th Edition – 2009.
2. Jean Ann Wright, "Animation Writing and Development: From Script Development to Pitch (Focal Press Visual Effects and Animation) 1st Edition".
3. Preston J. Blair, "Animation 1: Learn to Animate Cartoons Step by Step" (Cartooning, Book 1) Paperback – 2003.
4. Russell Chun "Adobe Animate CC Classroom in a Book" 1st Edition, 2018.

Web Resources:

1. 3D Model Creation with Blender. Coursera, University of London, https://www.coursera.org/learn/3d-model-creation-blender.
2. Introduction to Virtual Reality. Coursera, University of London, https://www.coursera.org/learn/virtual-reality.
3. 3D Computer Graphics. Coursera, University of Tokyo, https://www.coursera.org/learn/3d-computer-graphics.
4. Interactive Computer Graphics. Coursera, University of Tokyo, https://www.coursera.org/learn/interactive-computer-graphics.

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS)

SEMESTER – II

Human-Computer Interaction (3:0:0) 3

(Effective from the academic year 2024-25)

Course Code	24MCS24D	CIE Marks	50
Teaching Hours/Week (L:T:P)	3:0:0	SEE Marks	50
Total Number of Contact Hours	40	Exam Hours	3

Course Objectives: This course will enable students:

1. Explore the importance of a good interface design.
2. To analyze the importance of human psychology in designing good interfaces.
3. To instill knowledge for applying Human Computer Interface in their day – to – day activities.
4. To encourage students to indulge into research in Machine Interface Design.

Preamble: Human-Computer Interaction (HCI) refers to the design and use of systems and technologies that facilitate interaction between people and computers. This field encompasses a wide range of elements, including hardware, software, and user experience design, aiming to make interactions intuitive, efficient, and effective.

Module – 1

HUMAN AND THE INTERACTION PARADIGMS

Introduction, Input–output channels, Getting noticed, Design Focus: Where's the middle, Human memory, Design Focus: Cashing in, Thinking: reasoning and problem solving, Human error and false memories, Emotion, Individual differences, Psychology and the design of interactive systems-The context of the interaction, Half the picture, Experience, engagement and fun-Introduction, Paradigms for interaction.

(8 hours)

Module – 2

INTERACTION DESIGN: Introduction, What is design, The process of design, User focus, Cultural probes, Scenarios, Navigation design, Beware the big button trap, Modes, Screen design andlayout, Alignment and layout matter, Checking screen colors, Iteration and prototyping. **HCI IN THE SOFTWARE PROCESS**: Introduction, The software life cycle, Usability engineering, Iterative design and prototyping, Design Focus: Prototyping in practice, Design rationale.

(8 hours)

Module – 3

DESIGN RULES AND UNIVERSAL DESIGN: Introduction, Principles to support usability, Standards, Guidelines, Golden rules and heuristics, HCI patterns. Universal design principles, Multi-modal interaction, Designing websites for screen readers, Choosing the right kind of speech, Apple Newton, Designing for diversity, Mathematics for the blind.

(8 hours)

Module – 4

COGNITIVE MODELS AND SOCIO-ORGANIZATIONAL ISSUES: Introduction, Goal and task hierarchies, GOMS saves money, Linguistic models, The Challenge of display-based systems, Physical and device models, Cognitive architectures-Organizational issues, Implementing workflow in Lotus Notes.

(8 hours)

Module – 5

Communication And Collaboration Models: Introduction, Face-to-face communication, Looking real – Avatar Conference, Conversation, Text-based communication, Group working.

Task Analysis: Introduction, Differences between task analysis and other techniques, Task decomposition, Knowledge-based analysis, Entity–relationship-based techniques, Sources of information and data collection, Uses of task analysis. (8 Hours)

Course outcomes: Upon completion of the course, Students will be able to:

CO1: Analyze HCI principles to understand user needs and design requirements.
CO2: Evaluate interactive systems for usability and user experience
CO3: Analyze and identify user models, user support, socio-organizational issues, and stakeholder requirements of HCI systems.
CO4: Apply an interactive design process and universal design principles to designing HCI systems.

Question paper pattern:

- **SEE** will be conducted for 100 marks.
- Each full question is for 20 marks. (Answer five full questions out of 10 questions with intra modular choice). In every question, there will be a maximum of three sub-questions.
- **CIE** will be announced prior to the commencement of the course.
- 25 marks for the test. Average of three tests will be taken.
- 25 marks for Flexible Assessment Method.

Textbooks:

1. Alan Dix, Janet Finlay,Gregory D. Abowd, Russell Beale, *Human–Computer Interaction*,(3e) Pearson 2012

References:

1. Donald A. Norman, The design of everyday things, Currency and Doubbleday (2e), 2012.
2. Rogers Sharp Preece, Interaction Design:Beyond Human Computer Interaction, Wiley (2e), 2012.
3. Guy A. Boy, The Handbook of Human Machine Interaction, Ashgate publishing Ltd, 2011.

Web Resources:

1. Human-Computer Interaction. Coursera, University of California, San Diego, https://www.coursera.org/learn/human-computer-interaction. Accessed 28 Aug. 2024.
2. Interaction Design Specialization. Coursera, University of California, San Diego, https://www.coursera.org/specializations/interaction-design. Accessed 28 Aug. 2024.
3. User Experience Research and Design. Coursera, University of Michigan, https://www.coursera.org/specializations/user-experience-research-design. Accessed 28 Aug.

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS)

SEMESTER – II

NATURAL LANGUAGE PROCESSING (3:0:0) 3

(Effective from the academic year 2024-25)

Course Code	24MCS24E	CIE Marks	50
Teaching Hours/Week (L:T:P)	3:0:0	SEE Marks	50
Total Number of Contact Hours	40	Exam Hours	3

Course Objectives: This course will enable students:

1. To explore language processing techniques to enable Text data processing
2. To impart knowledge on text data processing using Statistical and Machine learning models
3. To describe the various Embedding and Deep learning models for NLP
4. To introduce real world applications of Language processing

Module – 1

INTRODUCTION TO LANGUAGE PROCESSING TASKS: Natural Language Processing – Applications of NLP – Linguistic Background – NLP tasks –Ambiguities in NLP tasks – Finite state automata – Regular Expressions – Corpus – Text Normalization – Edit Distance – Boundary Determination – Tokenization – Stemming -Lemmatization.

(8 Hours)

Module – 2

MORPHOLOGICAL ANALYSIS AND WORD PROCESSING: Morphological Analysis – Part of speech tagging – Shallow parsing – Dependency parsing – WordNet- Semantic similarity measures – Semantic representation – Coreference Resolution – Tools – Natural Language Toolkit – Stanford CoreNLP.

(8 Hours)

Module – 3

LANGUAGE AND STATISTICAL MODELS FOR NLP: Language model – n-gram language models – Hidden Markov Model – Conditional random Fields – Topic models – Graph Models – Machine Learning for NLP – Language Features – Maximum Entropy classifier – Phrase Based clustering.

(8 Hours)

Module – 4

NAMED ENTITY RECOGNITION AND KNOWLEDGE REPRESENTATION FOR TEXT: Information retrieval and Information extraction - Named Entity Recognition - Relation Identification-Template filling- Knowledge Representation for Texts: Ontologies, Knowledge Graphs, Frames-Linked Open Data Cloud.

(8 Hours)

Module – 5

CASE STUDIES FOR NLP: Case Studies and Research Cases in Question Answering – Machine Translation- Applications in Social media - Life science - Legal Text Applications.

(8 Hours)

Course outcomes: Upon completion of the course, Students will be able to:

CO1: Apply Language processing for text data at syntactic and semantic level.
CO2: Analyze the text content to provide predictions related to a specific domain using language models.
CO3: Analyze the patterns in text and pre-process the large text corpus.
CO4: Apply NLP concepts for building real-world applications.

Question paper pattern:

- **SEE** will be conducted for 100 marks.
- Each full question is for 20 marks. (Answer five full questions out of 10 questions with intra modular choice). In every question, there will be a maximum of three sub-questions.
- **CIE** will be announced prior to the commencement of the course.
- 25 marks for the test. Average of three tests will be taken.
- 25 marks for Flexible Assessment Method.

Textbooks:

1. Anders Søgaard, Ivan Vulić, Sebastian Ruder, Manaal Faruqui , Cross-Lingual Word Embeddings (Synthesis Lectures on Human Language Technologies), Morgan & Claypool Publishers, 2019.
2. Delip Rao, Brian McMahan, Natural Language Processing with PyTorch: Build Intelligent Language Applications Using Deep Learning, O'Reilly, 2019
3. Daniel Jurfsky, James H. Martin Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition, Second Edition, Pearson 2013.
4. Christopher D. Manning and Hinrich Schutze, "Foundations of Statistical Natural Language Processing", MIT Press, 1999

References:

1. Bird, Steven, Ewan Klein, and Edward Loper. Natural Language Processing with Python: Analysis of Text with the Natural Language Toolkit. O'Reilly Media, 2009.
2. Goldberg, Yoav. Neural Network Methods for Natural Language Processing. Morgan & Claypool Publishers, 2017.
3. Hirschberg, Julia, and Christopher D. Manning. Advances in Natural Language Processing. Cambridge University Press, 2021.
4. Manning, Christopher D., and Prabhakar Raghavan. Introduction to Information Retrieval. Cambridge University Press, 2008.

Web Resources:

1. Deeplearning.ai. Natural Language Processing Specialization. Coursera, https://www.coursera.org/specializations/natural-language-processing.
2. National Research University Higher School of Economics. Natural Language Processing with Classification and Vector Spaces. Coursera, https://www.coursera.org/learn/classification-vector-spaces.
3. University of Michigan. Natural Language Processing with Deep Learning in PyTorch. Coursera, https://www.coursera.org/learn/deep-learning-pytorch.

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS)

SEMESTER – II

WEB BASED INFORMATION RETRIEVAL (3:0:0) 3

(Effective from the academic year 2024-25)

Course Code	24MCS24F	CIE Marks	50
Teaching Hours/Week (L: T: P)	3:0:0	SEE Marks	50
Total Number of Contact Hours	40	Exam Hours	3

Course Objectives: This course will enable students to:

1. Demonstrate knowledge of key concepts in information retrieval, such as indexing, querying, and ranking.
2. Explain the mechanisms of search engines, including crawling, indexing, and query processing.
3. Develop and optimize search queries to effectively retrieve relevant web-based information.

Module – 1

Introduction: Basic Concepts – Practical Issues - Retrieval Process – Architecture - Boolean Retrieval –Evolution of the World Wide Web– Open Source IR Systems–History of Web Search – Web Characteristics–The impact of the web on IR —IR Versus Web Search.

(8 Hours)

Module – 2

Conceptual IR on the Web: Components of a Search Engine-Issues in Web Based IR-Types of Information Retrieval on the Web-Personalized Web Search-Applications of Information Retrieval on the Web-Web Crawler Architecture.

(8 Hours)

Module – 3

Modelling: Taxonomy and Characterization of IR Models – Boolean Model – Vector Model - Term Weighting-TF-IDF, Scoring and Ranking Models –Language Models – Set Theoretic Models -Probabilistic Models-Models for Browsing.

(8 Hours)

Module – 4

Analytical Models and Indexing: Latent Semantic Analysis-Semantic Similarity Models: Distance Based Models, Similarity Based Models, Co-occurrence Based Models-Blocked Sort Based Indexing -Distributed and Dynamic Indexing.

(8 Hours)

Module – 5

Evaluation, Querying and Case Studies: The Page Rank Model-Tokenization and Stemming Technique- IR Evaluation Metrics: Precision, Recall and Other Derived Measures- Keyword Based Querying and its categorization- Case Study on Hash Table Clustering for Web People Search- Case Study on Extracting Metadata using Element Tree for Tag Recommendation.

(8 Hours)

Course Outcomes: Upon completion of the course, Students will be able to:

CO1: Apply Boolean models, vector space models, and probabilistic models to solve practical problems.

CO2: Analyze the issues related to relevance and ranking of search results to meet user needs.

CO3: Design Web-based information retrieval systems.

Question paper pattern:

- **SEE** will be conducted for 100 marks.
- Each full question is for 20 marks. (Answer five full questions out of 10 questions with intra modular choice). In every question, there will be a maximum of three sub-questions.
- **CIE** will be announced prior to the commencement of the course.
- 25 marks for the test. Average of three tests will be taken.
- 25 marks for Flexible Assessment Method.

Textbooks:

1. Ricardo Baeza – Yates, BerthierRibeiro – Neto, Modern Information Retrieval: The concepts and Technology behind Search (ACM Press Books), Second Edition 2011
2. Ricardo Baeza – Yates, BerthierRibeiro – Neto, Modern Information Retrieval, Pearson Education, Second Edition, reprint 2015.

References:

1. Christopher D. Manning, Prabhakar Raghavan, Hinrich Schutze, Introduction to Information Retrieval, Cambridge University Press, First South Asian Edition 2012.
2. Stefan Buttcher, Charles L. A. Clarke, Gordon V. Cormack, Information Retrieval Implementing and Evaluating Search Engines, The MIT Press, Cambridge, Massachusetts London, England, 2010.

Web Resources:

1. University of London. "Information Retrieval." Coursera, https://www.coursera.org/learn/information-retrieval. Accessed 6 Sep. 2024.
2. Stanford University. "Natural Language Processing." Coursera, https://www.coursera.org/learn/natural-language-processing. Accessed 6 Sep. 2024.
3. University of California, Davis. "Text Retrieval and Search Engines." Coursera, https://www.coursera.org/learn/text-retrieval. Accessed 6 Sep. 2024.

M.TECH. COMPUTER SCIENCE AND ENGINEERING
Choice Based Credit System (CBCS)
SEMESTER – II

Semantic Web (3:0:0) 3
(Effective from the academic year 2024-25)

Course Code	24MCS24G	CIE Marks	50
Teaching Hours/Week (L: T: P)	3:0:0	SEE Marks	50
Total Number of Contact Hours	40	Exam Hours	3

Course Objectives: This course will enable students:

1. To learn the importance of Semantic Web.
2. To explore various semantic knowledge representation strategies.
3. To comprehend the concepts of Ontology.
4. To explore the various Ontology development methods and methodologies

Module – 1

Introduction: Comparing the Syntactic Web and Semantic Web-Limitations of Today's Web-Goal of the Semantic Web-How the Semantic Web will work-What Semantic Web is Not- A Layered Approach: Architecture of the Semantic Web-Logic on the Semantic Web.

(8 Hours)

Module – 2

Ontology And Semantic Web: What is an Ontology-From Ontology Towards Ontology Engineering- Types and Categorization of Ontologies- Categorizing Ontologies based on richness of their Structure-Overview of Ontological Commitments- Ontologies, Classifying Ontologies, Principles for the Design of Ontologies, Differences Among Taxonomies, Thesauri.

(8 Hours)

Module – 3

Ontology Representations And Languages: XML Overview-RDF: Overview, Syntax Structure, Semantics, Pragmatics-RDF Schema Representations-Ontology Markup Languages: OWL- Requirements for Web Ontology Description Languages, Header Information, Versioning, and Annotation Properties, Datatype and Object Properties, Property Characteristics, Classes, Class Descriptions, Class Axioms, Individuals, Data Types, A Summary of the OWL Vocabulary.

(8 Hours)

Module – 4

Modeling Specific Ontologies: Domain Ontologies: Building Chemistry Ontologies-Modeling Medical Ontologies- Building Engineering Ontologies- The CYC Method-Uschold and King's method-The KACTUS Approach-Methontology- Ontology Re-engineering.

(8 Hours)

Module – 5

Ontology Sources and Semantic Agents: Metadata, Dublin Core, Warwick Framework, FOAF, Upper Ontologies: SUMO, KR Ontology, CYC- Linked Open Data Cloud-Semantic Wikis and Wikidata- Agents in the Semantic Web Context, The Role of Ontologies, Software Agent Communication in the Semantic Web.

(8 Hours)

Course Outcomes: Upon completion of the course, Students will be able to:

CO1: Distinguish conventional web with Semantic Web.
CO2: Design semantic knowledge representation modes.
CO3: Design Ontologies pertaining to real world domains.
CO4: Compare the methods and methodologies in Ontology Modeling.

Question paper pattern:

- **SEE** will be conducted for 100 marks.
- Each full question is for 20 marks. (Answer five full questions out of 10 questions with intra modular choice). In every question, there will be a maximum of three sub-questions.
- **CIE** will be announced prior to the commencement of the course.
- 25 marks for the test. Average of three tests will be taken.
- 25 marks for Flexible Assessment Method.

Textbooks:

1. Karin K. Breitman, Marco Antonio Casanova, and Walt Truszkowski. Semantic Web: Concepts, Technologies and Applications. Springer, 2010.
2. Grigoris Antoniou, Frank van Harmelen, "A Semantic Web Primer (Cooperative Information Systems)", The MIT Press, Reprint 2015.
3. Asuncion Gomez-Perez, Oscar Corcho, Mariano Fernandez-Lopez "Ontological Engineering: with examples from the areas of Knowledge Management, eCommerce and the Semantic Web" Springer, Revised 2015.

References:

1. Alexander Maedche, "Ontology Learning for the Semantic Web", Springer; 1 edition, 2002
2. John Davies, Dieter Fensel, Frank Van Harmelen, "Towards the Semantic Web: Ontology – Driven Knowledge Management", John Wiley & Sons Ltd., Reprint 2013.
3. John Davies (Editor), Rudi Studer (Co-Editor), Paul Warren (Co-Editor) "Semantic Web Technologies: Trends and Research in Ontology-based Systems"Wiley Publications, Reprint 2015.
4. Dieter Fensel (Editor), Wolfgang Wahlster, Henry Lieberman, James Hendler, "Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential", The MIT Press, 2012 revised edition.
5. Michael C. Daconta, Leo J. Obrst, Kevin T. Smith, "The Semantic Web: A Guide to the Future of XML, Web Services, and Knowledge Management", Wiley, 2008 reprint.
6. Steffen Staab (Editor), Rudi Studer, "Handbook on Ontologies (International Handbooks on Information Systems)", Springer, 2008 reprint.
7. Dean Allemang (Author), James Hendler (Author) "Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL" (Paperback), Morgan Kaufma.

Web Resources:

1. Eindhoven University of Technology. "Semantic Web Technologies." Coursera, https://www.coursera.org/learn/semantic-web. Accessed 6 Sep. 2024.
2. Johns Hopkins University. "Linked Data and the Semantic Web." EP Online, https://ep.jhu.edu/courses/605643-linked-data-and-the-semantic-web/. Accessed 6 Sep. 2024.
3. University of Southampton. "The Semantic Web." FutureLearn, https://www.futurelearn.com/courses/the-semantic-web. Accessed 6 Sep. 2024.

M.TECH. COMPUTER SCIENCE AND ENGINEERING
Choice Based Credit System (CBCS)
SEMESTER – II

Generative AI and Prompt Engineering (3:0:0) 3
(Effective from the academic year 2024-25)

Course Code	24MCS24H	CIE Marks	50
Teaching Hours/Week (L: T:P)	3:0:0	SEE Marks	50
Total Number of Contact Hours	40	Exam Hours	3

Course Objectives: This course will enable students to:

1. Apply advanced neural network architectures like GANs, VAEs, and Transformers in generative AI models.
2. Analyze the use of generative AI in various fields such as healthcare, finance, and creative industries, gaining insights into real-world applications.
3. Gain proficiency in working with large language models (LLMs) such as GPT and BERT, with a focus on attention mechanisms, pre-training strategies, and practical deployment.
4. Develop practical skills in prompt engineering to improve the efficiency and quality of text generation tasks.
5. Apply generative AI techniques to solve real-world problems in natural language processing (NLP), text generation, and image generation, through hands-on projects and case studies.

Preamble: This syllabus serves as a comprehensive roadmap for students to master Generative AI and Prompt Engineering. It begins with foundational knowledge of advanced neural network architectures and generative models, such as GANs, VAEs, Transformers, and LSTMs, providing a solid base for understanding how these technologies work. Students will then explore practical applications of generative AI across various industries, including healthcare, finance, and the creative arts. The syllabus delves into advanced techniques, focusing on large language models like GPT and BERT, and emphasizes prompt engineering for optimizing AI performance. Through hands-on projects and real-world case studies, students will apply their learning to solve complex problems, enhancing their problem-solving skills and innovation capabilities. This structured approach ensures students are well-prepared for both academic research and professional roles in the rapidly evolving field of AI.

Module – 1

Advanced Neural Network Architectures: Introduction to advanced architectures. Introduction to Generative AI Models: Generative Adversarial Networks (GANs), Variational Autoencoders (VAEs), Transformers, Attention Mechanism in detail Long Short-Term Memory Networks (LSTMs). Generative AI Applications: Applications in Various Fields: Art and Creativity, Image and Video Generation, Text Generation, Music Composition, Healthcare Finance. Real-world use cases and challenges in deploying generative AI models.

(8 Hours)

Module – 2

Introduction to Large Language Models: Overview of Generative AI and Large Language Models. Basics of attention mechanisms and Transformer architecture. Pre-training techniques and transfer learning strategies. Real-world applications of large language models.

(8 Hours)

Module – 3

GPT Models and Applications: Study of GPT architecture and variants. Applications of GPT models in text generation and dialogue systems. Case study based implementation of GPT-based tasks. GPT-based chatbot enhances E-Shop's customer support service. BERT and Advanced Techniques: Understanding BERT architecture and pre-training objectives. Fine-tuning BERT for downstream NLP tasks. Exploration of advanced Transformer architectures and techniques.

(8 Hours)

Module – 4

Prompt Engineering: Five Pillars of Prompting. Intro to Text Generation Models. Standard Practices for Text Generation. Advanced Techniques for Text Generation with Langchain.

(8 Hours)

Module – 5

Prompt Engineering Continued: Vector Databases. Autonomous Agents with Memory and Tools. Intro to Diffusion Models for Image Generation. Standard Practices for Image Generation. Advanced Techniques for Image Generation.

(8 Hours)

Course Outcomes: The students will be able to:

CO1: Explain the architecture and functioning of GPT and BERT, and understand the role of attention mechanisms.

CO2: Apply basic prompt engineering techniques to optimize LLM performance in text generation tasks.

CO3: Implement GPT and BERT models in chatbots, text summarization, and sentiment analysis.

Textbooks:

1. Rehmani, Altaf. Generative AI for Everyone: Understanding the Essentials and Applications of This Breakthrough Technology. 2024.
2. Dhamani, Numa. Introduction to Generative AI. Kindle ed., 2024.
3. Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. Deep Learning. MIT Press, 2016.
4. Phoenix, James, and Mike Taylor. Prompt Engineering for Generative AI. 2024.

References:

1. Kalin, Josh. Generative Adversarial Networks Cookbook: Over 100 Recipes to Build Generative Models Using Python, TensorFlow, and Keras. Packt Publishing, 2018.
2. Sprinter, Jesse. Generative AI in Software Development: Beyond the Limitations of

Traditional Coding. O'Reilly Media, 2024.
3. Karim, Mohamed. Prompt Engineering: The Complete Guide. Tech Publishing, 2023.

Web Resources:

1. https://www.coursera.org/learn/generative-ai-prompt-engineering-for-everyone
2. https://elearn.nptel.ac.in/shop/iit-workshops/completed/leveraging-generative-ai-for-teaching-programming-course s/?v=c86ee0d9d7ed
3. https://elearn.nptel.ac.in/shop/iit-workshops/completed/introduction-to-language-models/?v=c86ee0d9d7ed

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS)

SEMESTER – II

AGILE TECHNOLOGY(3:0:0) 3

(Effective from the academic year 2024-25)

Course Code	24MCS25A	CIE Marks	50
Teaching Hours/Week (L:T:P)	3:0:0	SEE Marks	50
Total Number of Contact Hours	40	Exam Hours	3

Course Objectives: This course will enable students to:

1. Understanding iterative, incremental development processes leads to faster delivery of more useful software.
2. Demonstrate the workflow of the Automating process.
3. Explain the development of software using the Agile method.
4. Explain the Mastering Agility.
5. Understand the ability of six sigma concepts with Agile.

Preamble: Agile Technology refers to a set of principles and practices used in software development and project management that emphasize flexibility, collaboration, and iterative progress. Rooted in the Agile Manifesto, it prioritizes customer feedback, adaptive planning, and rapid delivery of functional software. Agile methodologies, such as Scrum and Kanban, advocate for frequent, incremental improvements through short development cycles known as sprints, fostering continuous communication between cross-functional teams and stakeholders.

Module – 1

Why Agile?: Understanding Success, Beyond Deadlines, The Importance of Organizational Success, Enter Agility, **How to Be Agile?:** Agile Methods, Don't Make Your Own Method, The Road to Mastery, Find a Mentor.

(8 hours)

Module – 2

Automating the Agile ALM: Goals of Automating the Agile ALM, Why Automating the ALMIs Important, Where Do I Start? Tools, Do Tools Matter? Process over Tools, Understanding Tools in the Scope of ALM, Staying Tools Agnostic, Commercial versus Open Source, What Do I Do Today?, Automating the Workflow, Process Modelling Automation.

(8 hours)

Module – 3

Managing the Lifecycle with ALM: Broad Scope of ALM Tools, Achieving Seamless Integration, Managing Requirements of the ALM, Creating Epics and Stories, Systems and Driven Development, Environment Management, Gold Copies, Supporting the CMDB.

(8 hours)

Module – 4

Mastering Agility: Values and Principles: Commonalities, About Values, Principles, and Practices, Further Reading, **Improve the Process:** Understand Your Project, Tune and Adapt, Break the Rules, **Rely on People:** Build Effective Relationships, Let the Right People Do the Right Things, Build the Process for the People, **Eliminate Waste:** Work in Small, Reversible Steps, Fail Fast, Maximize Work Not Done, Pursue Throughput.

(8 hours)

Module – 5

Introduction to Six Sigma and Agile: Definition and History, Core Concepts: DMAIC (Define, Measure, Analyse, Improve, Control), **Tools and Techniques:** Statistical Process Control, Root Cause Analysis, Process Mapping. **S**ynergies Between Six Sigma and Agile.
Case Study 1: Real-Time Quality Control in Agile with Six Sigma Tools
Case Study 2: Enhancing Software Quality with Six Sigma in Agile.

(8 hours)

Course Outcomes: The students will be able to:

CO1: Apply software development process to solve complex problems of engineering.

CO2: Apply the knowledge of Agile principle for rapid software development.

CO3: Distinguish between the traditional SDLC and agile ALM model for efficient and effective product delivery.

CO4: Develop real-world applications using DevOps tools.

Textbooks:

1. The Art of Agile Development (Pragmatic guide to agile software development), James shore, Chromatic, O'Reilly Media, Shroff Publishers & Distributors, 2007.

References:

1. Agile Software Development, Principles, Patterns, and Practices,Robert C. Martin, Prentice Hall; 1st edition, 2002.
2. Agile and Iterative Development A Manger's Guide", Craig Larman Pearson Education, First Edition, India, 2004.
3. "The Lean Six Sigma Guide to Doing More With Less" by Mark Price, "Agile Six Sigma" by Scott M. Graffius.

Web Resources:

1. . https://www.geeksforgeeks.org/software-engineering-agile-software-development/

M.TECH. COMPUTER SCIENCE AND ENGINEERING
Choice Based Credit System (CBCS)
SEMESTER – II

WEB ENGINEERING (3:0:0) 3
(Effective from the academic year 2024-25)

Course Code	24MCS27B	CIE Marks	50
Teaching Hours/Week (L:T:P)	3:0:0	SEE Marks	50
Total Number of Contact Hours	40	Exam Hours	3

Course Objectives:
This course will enable students to:

1. Develop a comprehensive understanding of the processes involved in web engineering, from concept to deployment.
2. Learn the key principles of web design focusing on usability, and accessibility.
3. Explore the basics of project management specific to web application development, including planning, scheduling, and risk management.
4. Explore the importance of web application development and its life cycle.

Preamble: Web Engineering involves the disciplined application of engineering principles to the development and management of web applications and services. It encompasses the systematic processes of design, development, deployment, and maintenance to ensure that web solutions are robust, scalable, and user-centric. Key aspects include the use of established methodologies and tools to handle requirements, architecture, usability, and performance, ensuring that web systems meet both technical specifications and user needs. Web Engineering aims to produce high-quality web applications efficiently while adapting to evolving technologies and user expectations.
.

Module – 1

Introduction: Web Engineering Introduction - WebE Framework-Principles Should to Adapt the Framework-Is There Any Merit in an Old-School Approach-The Components of Web Engineering - How Does Software Engineering Come into Play- What WebE Methods Reside within the Process Framework-Isn't Web Engineering All about Tools and Technology- Web Engineering Best Practices **(8 hours)**

Module – 2

Process: Defining the Framework- Incremental Process Flow- How Are Framework Activities Conducted? -How Is the Framework Refined? -Generic Actions and Tasks for the WebE Framework -How Should the Communication Activity Be Refined? -What Tasks Are Required to Develop an Increment Plan?

(8 hours)

Module – 3

Modeling: What Is Modeling?- What Analysis Modeling Tasks Can Be Applied? - What Are the Elements of a Design Model? -What Design Modeling Tasks Can Be Applied? What Construction Tasks Should Be Applied? Umbrella Activities -How Should a WebE Team Manage Change? - Modeling as a Concept -Modeling Frameworks-Is There a Modeling Framework for the Web- - Existing Modeling Approaches . (8 hours)

Module – 4

Communication: Communication and Planning-The Communication Activity -Elicitation- What Happens Before an Elicitation Session- How Do Stakeholders Prepare- What Tasks Are Performed During an Elicitation Session-What Are the User Categories for the WebApp- How Are Content and Functional Requirements Identified-How Are Constraints and Performance Issues Isolated-What Are Usage Scenarios? -What Are Use Cases? - How Is a Use Case Created- Identifying WebApp Increments **(8 hours)**

Module – 5

Design: Design for WebApps - What Does a WebApp Designer Need to Know? --What Is Logical Design- What Is Physical Design? What Information Is Created because of Design? -Design Goals-Design Quality? -What Is a Quality Framework? -Design Process What Are the Elements of WebApp Design? -Characteristics of the Design Process? What Does an Incremental WebE Process Imply for the Design Activity- Initial Design of the Conceptual Architecture -Initial Design of the Technical Architecture – Case Study and Comparative Web Architectures. **(8 hours)**

Course Outcomes:

The students will be able to:

CO1: Apply fundamental web design principles to create intuitive, user centered interfaces that meet accessibility standards and enhance the overall user experience.

CO2: Analyze Requirements for Web Application Design

CO3: Analyze client and user requirements to produce detailed design specifications, including architectural designs, wireframes, and user flows that align with business objectives.

CO4: Create comprehensive project plans that include timelines, resource allocation, and risk management strategies tailored to the unique demands of web application development.

Textbooks:

1. Pressman, Roger S., and David Lowe. Web Engineering: A Practitioner's Approach. McGraw-Hill, 2009.

References:

1. Kappel, Gerti, et al., editors. Web Engineering: The Discipline of Systematic Development of Web Applications. Wiley, 2006.
2. Rosenfeld, Louis, and Peter Morville. Information Architecture: For the Web and Beyond. 4th ed., O'Reilly Media, 2015.

Web Resources:

1. Doe, Jane, and John Smith. Introduction to Web Software Engineering. Coursera, 12 Mar. 2023, www.coursera.org/learn/web-software-engineering. Accessed 10 Sept. 2024.

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS)

SEMESTER – II

OBJECT ORIENTED ANALYSIS AND DESIGN (3:0:0) 3

(Effective from the academic year 2024-25)

Course Code	24MCS25C	CIE Marks	50
Teaching Hours/Week (L: T: P)	3:0:0	SEE Marks	50
Total Number of Contact Hours	40	Exam Hours	3

Course Objectives: This course will enable students:

1. To provide a brief, hands-on overview of object-oriented concepts and its life cycle for software development.
2. To learn for modeling the software and to design them using UML diagrams
3. To Explore the problem domain and to identify the objects from the problem specification.
4. To Apply design axioms and corollaries for the classes and object relational systems.

Preamble: Object-Oriented Analysis and Design (OOAD) is a methodology used to analyze and design software systems by visualizing them as collections of interacting objects, each representing an entity with specific attributes and behaviors. In OOAD, analysis focuses on understanding the problem domain and identifying the objects that will be part of the system. Design then structures these objects into classes, defining their relationships, interactions, and the overall architecture of the system. Key concepts include encapsulation (bundling data and methods), inheritance (reusing code through hierarchical relationships), polymorphism (objects behaving differently based on context), and abstraction (simplifying complex systems by focusing on essential characteristics). OOAD promotes reusability, scalability, and maintainability in software development.

Module – 1

INTRODUCTION: An overview – Object basics – Object state and properties – Behaviour – Methods – Messages – Information hiding – Class hierarchy – Relationships – Associations – Aggregations- Identity – Dynamic binding – Persistence – Meta classes – Object oriented system development life cycle.

(8 hours)

Module – 2

METHODOLOGY AND UML: Introduction – Survey – Rumbaugh, Booch, Jacobson methods – Unified modelling language – Static and Dynamic models – Rational Rose Suite - UML diagrams – Static diagram : Class diagram – Use case diagrams – Behaviour Diagram : Interaction diagram – State chart diagram – Activity diagram - Implementation diagram: Component diagram – Deployment diagram – example - Design of online railway reservation system using UML diagrams - Dynamic modelling – Model organization – Extensibility.

(8 hours)

Module – 3

OBJECT ORIENTED ANALYSIS: Identifying Use case – Business object analysis – Use case driven object-oriented analysis – Use case model – Documentation – Classification – Identifying object, relationships, attributes, methods – Super-sub class – A part of relationships Identifying attributes and methods – Object responsibility – construction of class diagram for generalization, aggregation – example – vehicle class. (8 hours)

Module – 4

OBJECT ORIENTED DESIGN: Design process and benchmarking – Axioms – Corollaries – Designing classes – Class visibility – Refining attributes – Methods and protocols – Object storage and object interoperability – Databases – Object relational systems – Designing interface objects – Macro and Micro level processes – The purpose of a view layer interface-OOUI - MVC Architectural Pattern and Design – Designing the system.

(8 hours)

Module – 5

Railway domain: Platform assignment system for the trains in a railway station - Academic domain: Student Marks Analyzing System - ATM system - Stock maintenance - Quiz System - E-mail Client system - Cryptanalysis – Health Care Systems. Use Open-source CASE Tools: StarUML/ UML Graph for the above case studies.

(8 hours)

Course Outcomes: The students will be able to:

CO1: Apply object-oriented life cycle model for a project.
CO2: Design static and dynamic models using UML diagrams.
CO3: Apply object-oriented analysis to identify the objects from the problem specification.
CO4: Apply the open-source CASE tools in various domains.

Textbooks:

1. The Art of Agile Development (Pragmatic guide to agile software development), James shore, Chromatic, O'Reilly Media, Shroff Publishers & Distributors, 2007.

References:

1. Ali Bahrami, "Object Oriented System Development", McGraw Hill International Edition, 2008
2. Brahma Dathan, Sarnath Ramnath, "Object-Oriented Analysis, Design and Implementation", Universities Press, 2010.

Web Resources:

1. http://staruml.sourceforge.net/docs/StarUML_5.0_Developer_Guide.pdf

M.TECH. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System(CBCS)

SEMESTER - II

Program Management (3:0:0) 3

(Effective from the academic year 2024-25)

Course Code	24MCS25D	CIE Marks	50
Teaching Hours/Week (L:T:P)	3:0:0	SEE Marks	50
Total Number of Contact Hours	40	Exam Hours	3

Course Objectives: This course will enable students to:

1. Reflect on key concepts, theories, and methodologies applicable to program management.
2. Effectively communicate with stakeholders and lead program teams.
3. Align program objectives with organizational strategy and goals.

Preamble: This course provides a comprehensive overview of program management, focusing on the strategies and skills required to plan, execute, and monitor multiple projects within a program. Students will learn to manage complex, interrelated projects, ensuring they align with organizational goals. The course also emphasizes leadership, risk management, stakeholder communication, and the use of program management tools and methodologies.

Module – 1

Introduction to Program Management: What Is Program and Program Management. Difference between Project and Program Management. The Relationships among Portfolio, Program, and Project Management, and their Roles in Organizational Project Management. The Relationships among Organizational Strategy, Program Management, and Operations Management. Business Value. Role of the Program Manager, Program Sponsor and Program Management Office. Program Management Performance Domains.

(8 hours)

Module – 2

Program Life Cycle Management: The Program Life Cycle. Program Activities and Integration Management. Program Definition phase activities. Program delivery phase activities. Program closure phase activities.

(8 hours)

Module – 3

Program Strategy Alignment and Program Benefits Management: Program Business Case. Program Charter. Program Roadmap, Environmental Assessments. Program Risk Management Strategy. Benefits Identification. Benefits Analysis and Planning. Benefits Delivery, Transition and Sustainment.

(8 hours)

Module – 4

Program Stakeholder Engagement and Communication: Program Stakeholder Identification. Program Stakeholder Analysis. Program Stakeholder Engagement Planning. Program Stakeholder Engagement. Program Stakeholder Communications.

(8 hours)

Module –5

Program Governance: Program Governance Practices. Program Governance Roles. Program Governance Design and Implementation.

(8 hours)

Course outcomes: The students will be able to

CO1: Analyze and manage program risks and opportunities.

CO2: Apply advanced project management tools and techniques within a program management context.

CO3: Develop and implement strategies for managing multiple projects under a single program.

Question paper pattern:

- **SEE** will be conducted for 100 marks.
- Each full question is for 20 marks. (Answer five full questions out of 10 questions with intra modular choice). In every question, there will be a maximum of three sub-questions.
- **CIE** will be announced prior to the commencement of the course.
- 25 marks for the test. An average of three tests will be taken.
- 25 marks for Flexible Assessment Method.

Textbooks:

1. Michel Thiry, "Program Management: A Comprehensive Overview", Routledge (Taylor & Francis Group), London.
2. Project Management Institute, "The Standards for Project Management".

References:

1. Mark C. Bojeun, "Program Management Leadership: Creating successful team dynamics", CRC Press.
2. Mitchell L. Springer, "Project and Program Management: A Competency-Based Approach", Purdue University, US.
3. Ginger Levin, "Program Management: A Life Cycle Approach", CRC Press.
4. Russ J. Martinelli, Dragan Z. Milosevic, and James M. Waddell, "Program Management for Improved Business Results", John Wiley & Sons.
5. Lowell Dye and James Pennypacker, "Managing Multiple Projects: Planning, Scheduling, and Allocating Resources for Competitive Advantage", Taylor & Francis.

Web Resources:

1. https://www.pmi.org/certifications/program-management-pgmp/exam-prep
2. https://www.pmi.org/-/media/pmi/documents/public/pdf/certifications/program-management-professional-handbook-english.pdf?rev=9cebccb9ac324e789723bb82450844d3&sc_lang_temp=en
3. https://www.pmi.org/-/media/pmi/documents/public/pdf/certifications/pgmp-exam-content-outline.pdf?rev=67166f23d2794860865edbaeff922a59&sc_lang_temp=en
4. https://www.pmi.org/pmbok-guide-standards/foundational/pmbok
5. https://ccrs.pmi.org/search/courses?kwd=PgMP&_gl=1*14y642f*_gcl_au*MjM4MzgxMzIxLjE3MjU0NDI4OTE.

M.TECH. COMPUTER SCIENCE AND ENGINEERING
Choice Based Credit System (CBCS)
SEMESTER – II

Web Application Development Laboratory (0:1:2) 2

(Effective from the academic year 2024-25)

Course Code	24MCSL26	CIE Marks	50
Teaching Hours/Week (L: T:P)	0:1:2	SEE Marks	50
Total Number of Contact Hours	30	Exam Hours	3

Course Objectives: This course will enable students:

1. To develop the knowledge for the applications in dynamic and responsive user interfaces with Reactjs.
2. Learn to implement JSON Web Token (JWT) authentication to secure APIs, ensuring that only authorized users can access specific endpoints.
3. To explore building dynamic and responsive user interfaces using React.js, Nodejs with appropriate NOSQL Databases for a capstone project.

Preamble: This lab course is designed to provide students with a comprehensive understanding of React.js and Nodejs, a leading JavaScript library for building dynamic and responsive user interfaces. Through hands-on experiments, students will learn core concepts such as component-based architecture, state management with hooks and integrating external APIs. The course will also cover advanced topics like hooks, JSON web Token Authentication. By the end of this course, students will be proficient in React.js, Nodejs ready to design and implement high-performance web applications.

Description

React.js: Introduction to React.js-Basics of React.js-What is React?-The Virtual DOM-Setting up a React Project (using Create React App)-JSX Syntax-Embedding Expressions in JSX-Components in React.

XML: Introduction to XML-Understanding XML-XML Syntax and Structure-Differences between XML, HTML, and JSON-Working with XML in Web Application-Parsing XML in JavaScript-XML and React Integration-XML vs. JSON for Data Exchange-XML Schema (XSD)-Defining Structure and Constraints-Validating XML with XSD-XPath and XSLT-XPath Syntax for Navigating XML Documents-Transforming XML with XSLT.

Node.js: Basics of Node.js-What is Node.js?-Setting up a Node.js Environment-Introduction to npm and Node.js Modules-Building a Basic Web Server with Node.js-Using the HTTP Module-Handling Requests and Responses-Working with the File System in Node.js-Reading and Writing Files-Streaming Data-Asynchronous Programming in Node.js-Callbacks, Promises, and Async/Await-Error Handling in Asynchronous Code.

PART A: List of Experiments

1. Implement a to-do list application using React JS hooks such as useState and useEffect.
2. Create a dynamic form/ web page in React JS that includes validation for user inputs..
3. Design the Rest API's , GET,POST,PUT,DELETE and JSON Web Token Authentication in Nodejs.
4. Create an XML document representing a university database. The document should include:

   University Information:

2. University name
3. Address
4. List of departments

   Department Information:

5. Department name
6. List of courses

   Course Information:

5. Course name
6. Course code
7. Instructor
8. Credits

5. Create an XML Schema (XSD) to validate the XML document from Exercise 4. Define the structure and data types for each element and attribute.

PART-B: Project

Design & Develop a Capstone project for developing a web application using Reactjs, or Nodejs with XML for use cases in Automation Sciences as a domain of choice.

Course Outcomes: The students will be able to:

CO1: Create dynamic, user-friendly front-end, validation applications using React JS.

CO2: Design an integrated payment gateways into Node.js applications, enabling secure and efficient payment processing.

CO3: Develop a Web Application for Automation Sciences as a Domain of Choice.

Text Book

- Write Modern Web Apps with the MEAN Stack Mongo, Express, AngularJS, and Node.js DEVELOP AND DESIGN Jeff Dickey, 2019.

References

- J.D.Meier,Alex Homer, "Web Application Architecture guide, Patterns and Practices", Microsoft 2008.

Web Resources

1. https://www.coursera.org/learn/web-development

M. Tech COMPUTER SCIENCE AND ENGINEERING
Choice Based Credit System (CBCS)
SEMESTER – II

Mobile Application Development Laboratory (0:0:2) 1
(Effective from the academic year 2024-25)

Course Code	24MCS27A	CIE Marks	50
Teaching Hours/Week (L: T:P)	0:0:2	SEE Marks	50
Total Number of Contact Hours	24	Exam Hours	3

Course Objectives:

1. Learn and acquire the art of Android Programming.
2. Install and Configure Android studio and its development tools to run the applications.
3. Use User Interface components for android application development.
4. Create Android applications using mobile related server-less databases.
5. Inspect different methods of sharing data using services

List of Experiments-PART A

1. Create an app that displays weather information for a given location using data from an external API.
2. Create an app that shows the user's current location on a map, allows searching for places, and marking locations.
3. Develop a basic e-commerce app where users can browse products, add to cart, and place orders.
4. Develop an app that performs actions based on voice commands.
5. Create a custom view that draws basic shapes (circles, rectangles) and handles user touch events.

PART-B: Project

1. AI-Powered Chatbot for Customer Support.
2. Health Monitoring System with Wearable Integration.
3. Real-Time Language Translation App.
4. E-learning Platform with AR Integration.
5. Mental Health Support App with AI-Based Chatbot.
6. Expense Tracker with AI-based Budgeting.

Course Outcomes: The students will be able to:

CO1: Create, test and debug Android applications by setting up an Android development environment.
CO2: Implement adaptive, responsive user interfaces that work across a wide range of devices.
CO3: Infer long-running tasks and background work in Android applications.
CO4: Demonstrate methods in storing, sharing and retrieving data in Android applications. CO5: Infer the role of permissions and security for Android applications.

Textbooks:

1. Google Developer Training, "Android Developer Fundamentals Course – Concept Reference", Google Developer Training Team, 2017.

References:

1. Erik Hellman, "Android Programming – Pushing the Limits", 1st Edition, Wiley India Pvt Ltd, 2014, ISBN-13: 97 8-8126547197
2. Dawn Griffiths and David Griffiths, "Headfirst Android Development", 1st Edition, O'Reilly SPD Publishers, 2015. ISBN-13: 978-9352131341

3. Bill Phillips, Chris Stewart and Kristin Marsicano, “Android Programming: The Big Nerd Ranch Guide”, 3rd Edition, Big Nerd Ranch Guides, 2017. ISBN-13: 978-0134706054.

Web Resources:

1. https://www.gitbook.com/book/google-developer-training/android-developer-fundamentals course-concepts/detail

M.TECH. COMPUTER SCIENCE AND ENGINEERING
Choice Based Credit System (CBCS)
SEMESTER – II

Requirement Analysis and Software Testing Tools Lab (0:0:2) 1
(Effective from the academic year 2024-25)

Course Code	24MCS27B	CIE Marks	50
Teaching Hours/Week (L: T:P)	0:0:2	SEE Marks	50
Total Number of Contact Hours	24	Exam Hours	3

Course Objectives: This course will enable students:

1. To Explore a test suite to evaluate the process metrics involved in testing tools such as Selenium or Cypress.
2. To Design a model for planning, assessing and testing of requirement Analysis for E-bidding or Electric vehicle using UMLGraph/AI Automation tools.
3. Learn To create a Project and exhibit skills to work towards solution of Real-time test scenarios in e-commerce websites/Electric Vehicles.

Preamble: Automation software testing is vital for maintaining software quality, efficiency, and reliability. This lab provides practical experience with designing and executing automated test cases using key tools such as Selenium, Cypress and JIRA. Students will learn to develop robust testing strategies and effectively utilize these tools for web testing, unit testing, and integration testing. Additionally, the lab will cover the use of JIRA for managing and tracking bugs, enhancing the overall testing and quality assurance process. By integrating these tools, students will be equipped with the skills needed for real-world software development and quality assurance roles.

PART A: List of Experiments

1. Write a Selenium script to automate the login process for a sample website.
2. Design a selenium web driver program to handle pop ups. Go to student login page, click on login button without giving username and password, and handle that pop up message.
3. Write a Python script to manage and track bugs in an E-Commerce Platform application using the JIRA REST API. Include functionalities for creating, updating, and retrieving issues.
4. Demonstrate a requirement specification and Prototype Model analysis of E-Bidding System using a UML Based tool.
5. Demonstrate a requirement specification and Prototype Model for Electric vehicle as a domain on choice using a UML Based tool.

PART-B: Project

Create a capstone project on Electric Vehicles to Test plan, Software Requirement Specifications (SRS), Test cases of Executable Test suites using Selenium/ Cypress Tools**.**

Course Outcomes: The students will be able to:

CO1: Analyze how the Selenium and Python scripts are used, web driver program to handle pop ups and manage, track bugs in an E-Commerce Platform using JIRA Tool.

CO2: Design models for planning, analyzing, assessing, SRS and testing automated test cases for E-Bidding and Agile Methodologies for Robotics automation.

CO3: Develop a capstone project with complete Test solutions using Selenium/ Cypress Tools for Automation scenarios in Electric Vehicles.

Text Books:

1. Paul C. Jorgensen: Software Testing, A Craftsman‟s Approach, 3rd Edition, Auerbach Publications, 2008.
2. Aditya P Mathur: Foundations of Software Testing, Pearson Education, 2008.

References:

1. Roger S.Pressman, Software engineering- A practitioner's Approach, McGraw-Hill International Edition, 6th edition, 2001.
2. Arlow, Jim, and Ila Neustadt. UML 2 and the unified process: practical object-oriented analysis and design. Pearson Education, 2005.

M.TECH. COMPUTER SCIENCE AND ENGINEERING
Choice Based Credit System (CBCS)
SEMESTER – II

GIT for DEVOPS (0:0:2) 1

(Effective from the academic year 2024-25)

Course Code	24MCS27C	CIE Marks	50
Teaching Hours/Week (L: T:P)	0:0:2	SEE Marks	50
Total Number of Contact Hours	20	Exam Hours	2

Course Objectives: This course will enable students to:

1. Gain a comprehensive understanding of Git fundamentals and its role in version control.
2. Master the use of branching, merging, and conflict resolution in Git.
3. Develop skills to collaborate effectively using remote Git repositories.
4. Learn to implement advanced Git techniques and workflows for efficient code management.

Preamble: This course provides an in-depth understanding of Git, a vital version control system in modern software development and DevOps. Students will learn essential Git operations, branching and merging strategies, remote repository management, and advanced Git techniques. By mastering these skills, students will be equipped to efficiently manage code changes, collaborate on projects, and maintain organized and robust codebases.

PART A: List of Experiments

1. **Setting Up and Basic Commands.**
   - **a.** Initialize a new Git repository in a directory. Create multiple files, add them to the staging area, and commit the changes with an appropriate commit message.
   - **b.** Remove a file from the staging area without deleting the file itself.
2. **Creating and Managing Branches**
   - **a.** Create a new branch named "bug-fix" and switch to it. Make a few changes and commit them. Then, merge these changes back into the "master" branch without creating a merge commit.
   - **b.** Rename an existing branch from "development" to "dev".
3. **Stashing Changes**
   - **a.** Create several uncommitted changes in your working directory. Stash the changes with a custom stash message, switch to another branch, make some changes, and commit them. Then, return to the original branch and apply the stashed changes.
4. **Collaboration and Remote Repositories**
   - **a.** Add a new remote repository named "upstream" and fetch the changes from this remote. Rebase your current branch onto the "upstream/master" branch.
   - **b.** Push your local "feature" branch to a remote repository and set the upstream branch for tracking future changes.
5. **Handling Merge Conflicts**
   - **a.** Simulate a merge conflict by making different changes to the same file in two different branches. Attempt to merge these branches and resolve the conflict using

the Git command line.

6. **Git Tags and Releases**
   a. Create an annotated Git tag named "release-2.0" with a message describing the release details. List all tags in the repository.
7. **Analyzing and Changing Git History**
   a. Write the command to show the commit graph along with the author and commit messages.
   b. Use git filter-branch or git rebase to modify the author of multiple commits across different branches.
   c. Remove a specific file from the history of a repository and from all commits.
8. **Git Worktrees**
   a. Create a separate worktree for an existing branch to work on it independently without switching the branches in your main repository.
9. **Advanced Git Operations**
   a. Use the interactive rebase feature to squash several commits into a single commit on the "master" branch.
   b. Write the command to revert multiple consecutive commits without using git revert multiple times.
10. **Git Hooks and Git Bisect**
    a. Write a pre-commit hook that checks for a specific pattern in the commit message and rejects the commit if it doesn't match.
    b. Use git bisect to find the commit that introduced a bug, given that you know a range of commits where the bug first appeared.

Course outcomes: The students will be able to

CO1: Analyze basic Git operations to handle local repositories.
CO2: Demonstrate branching and merging strategies, including conflict resolution in Git repositories.
CO3: Implement remote repositories, including pushing, pulling, and handling pull requests.
CO4: Apply advanced Git commands and workflows to maintain and organize complex codebases.

Textbooks:

1. Pro Git second edition by Scott Chacon and Ben Straub-2014
2. Loeliger, Jon, and Matthew McCullough. Version Control with Git: Powerful Tools and Techniques for Collaborative Software Development. 2nd ed., O'Reilly Media, 2012.
3. Chacon, Scott, and Ben Straub. Pro Git. 2nd ed., Apress, 2014.

References:

1. Swicegood, Travis. Pragmatic Version Control Using Git. Pragmatic Bookshelf, 2008.
2. Bell, Brent Laster. Professional Git. John Wiley & Sons, 2016.
3. Boulton, Mike. Git for Teams: A User-Centered Approach to Creating Efficient Workflows in Git. O'Reilly Media, 2016.

Web Resources:

1. Davis, Neal. Version Control with Git. Coursera, LearnQuest. Accessed 30 Aug. 2024.
2. Willis, Morgan. DevOps on AWS: Code, Build, and Test. Coursera, Amazon Web Services. Accessed 30 Aug. 2024.
3. Park, Sangkyu. Continuous Integration and Continuous Delivery (CI/CD). Coursera, University of California, Irvine. Accessed 30 Aug. 2024.

M.TECH. COMPUTER SCIENCE AND ENGINEERING
Choice Based Credit System (CBCS)
SEMESTER – II

Introduction to Kafka (0:0:2) 1
(Effective from the academic year 2024 -25)

Course Code	24MCS27D	CIE Marks	50
Teaching Hours/Week (L:T:P:S)	0:0:2	SEE Marks	50
Total Number of Contact Hours	20	Exam Hours	3 Hours

Course Objectives: This course will enable students to:

1. Identify and describe the core components of Apache Kafka, including topics, producers, consumers, and brokers.
2. Configure and deploy a Kafka cluster, including setting up producers and consumers to handle a basic data stream.
3. Analyze the performance of a Kafka cluster under different load conditions and optimize configuration settings for improved throughput.

Preamble: Apache Kafka is a powerful distributed streaming platform that enables real-time data processing and integration across diverse systems. Kafka typically covers its architecture, including brokers, topics, partitions, and ZooKeeper, and guides participants through setting up and configuring Kafka clusters. It emphasizes practical skills such as producing and consuming messages, developing Kafka Streams applications, and leveraging Kafka Connect for integrations. Advanced topics include implementing exactly-once semantics, securing Kafka deployments, monitoring cluster performance, and designing robust data pipelines.

Descriptions

Design, develop, and implement the specified programs as given in the list given below Apache Kafka in Windows/Linux Environment.

Prerequisites: Set up a basic Kafka environment with ZooKeeper and Kafka brokers.

PART A: List of Experiments

1. Develop simple Kafka producer and consumer applications in Java.
2. Create a simple Kafka Streams application to process streaming data.
3. Build a more complex Kafka Streams application with stateful operations.
4. Use Kafka Streams' interactive queries to query state stores. Modify the Kafka Streams application to expose interactive queries.
5. Develop producer and consumer applications with various delivery guarantees.

PART-B: Project

Build a distributed real-time event processing pipeline for an e-commerce platform to track user activity such as page views and purchases.

Course Outcomes: At the end of the course the student will be able to:

CO1: Build scalable, real-time data processing solutions using Kafka.

CO2: List and define the fundamental components of Apache Kafka, such as topics, partitions, producers, consumers, and brokers.

CO3: Implement a Kafka cluster, including the creation and management of topics, and demonstrate data flow between producers and consumers.

CO4: Evaluate the performance of a Kafka cluster under varying loads and analyze the impact of different configuration settings on throughput and latency.

Textbooks

1. Narkhede, Neha, Gwen Shapira, and Todd Palino. Kafka: The Definitive Guide: Real-Time Data and Stream Processing at Scale. O'Reilly Media, 2017.
2. Rajput, Dinesh. Mastering Apache Kafka: Self-Learning Guide. Packt Publishing, 2021.
3. Kleppmann, Martin. Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems. O'Reilly Media, 2017

Reference Books

1. Scott, Dylan, Viktor Gamov, and Dave Klein. Kafka in Action. Manning Publications, 2022.
2. Bejeck, Bill. Kafka Streams in Action: Real-Time Apps and Microservices with the Kafka Streams API. Manning Publications, 2018.

Web Resources

1. The Complete Apache Kafka Course. Coursera, Confluent, https://www.coursera.org/specializations/complete-apache-kafka-course. Accessed 28 Aug. 2024.

M.TECH. COMPUTER SCIENCE AND ENGINEERING
Choice Based Credit System (CBCS)
SEMESTER – II

Kubernetes and Docker Laboratory (0:0:2) 2
(Effective from the academic year 2024-25)

Course Code	24MCS27E	CIE Marks	50
Teaching Hours/Week (L: T:P)	0:0:2	SEE Marks	50
Total Number of Contact Hours	20	Exam Hours	2

Course Objectives: This course will enable students to:

1. Apply Docker commands to build, manage, and run containerized applications, including multi-container setups using Docker Compose.
2. Deploy Kubernetes applications by defining Pods, Deployments, and Services to orchestrate multi-component applications.
3. Integrate persistent storage and networking solutions in Kubernetes for stateful applications and expose services externally for access.
4. Develop a multi-tier application on Kubernetes using best practices for containerization, service management, and application scaling.

Preamble: The Docker and Kubernetes Lab course is designed to provide students with practical, hands-on experience in containerization and orchestration technologies, which are essential for building, deploying, and managing modern cloud-native applications. The course begins by introducing Docker, a powerful platform for developing, shipping, and running applications in containers, enabling efficient and consistent environments. The course then transitions into Kubernetes, the leading orchestration platform for automating the deployment, scaling, and management of containerized applications. By the end of the course, students will be well-equipped with the skills required to work in DevOps environments and deploy, manage, and scale applications in cloud-native architectures using Docker and Kubernetes.

PART A: List of Experiments

1. Install and Set Up Docker: Install Docker on Linux/Windows, verify the installation using docker --version, and run a test container with docker run hello-world.
2. Working with Docker Containers: Execute Docker commands to manage containers by pulling an image from DockerHub, running it in both interactive and detached modes, and stopping, removing, and restarting containers.
3. Docker Compose for Multi-Container Applications: Use Docker Compose to manage multi-container applications by creating a docker-compose.yml file to run a web server (e.g., Nginx) and a database (e.g., MySQL), and manage the application with Docker Compose commands for starting, stopping, and scaling.
4. Setting Up Minikube:Install and configure a single-node Kubernetes cluster with Minikube by installing Minikube and kubectl, starting Minikube, verifying the cluster status using kubectl, and running a test pod with kubectl run nginx --image=nginx.

5. Managing Kubernetes Deployments and Services: Create and manage a Kubernetes deployment by writing a deployment YAML file to run multiple replicas of an application (e.g., Nginx), scale it to 3 replicas, and create a service to expose the deployment for access via a browser or curl.

PART-B: Project

Deploying a Multi-Tier Application on Kubernetes

Objective: Deploy a multi-tier web application using Kubernetes with persistent storage, services, and scaling.

Tasks:

- Containerize a multi-component application (e.g., web front-end, backend, and database).
- Create Kubernetes YAML files for each component (Pods, Deployments, Services).
- Set up persistent storage for the database using PersistentVolumes and PersistentVolumeClaims.
- Deploy the application and test its functionality.
- Implement Horizontal Pod Autoscaling (HPA) based on CPU usage.

Course Outcomes: The students will be able to:

CO1: Apply Docker commands to build and manage containerized applications, including the creation and orchestration of multi-container environments using Docker Compose.

CO2: Deploy Kubernetes applications with proficiency, including scaling deployments and exposing services, to ensure effective orchestration and high availability.

CO3: Implement persistent storage and configure networking within Kubernetes to support both stateful and stateless applications, ensuring reliable data management and connectivity.

CO4: Design a multi-tier application on Kubernetes, applying best practices for containerization, scaling, and service management to deliver robust, cloud-native solutions.

Text Books:

1. Nigel Poulton, “Docker Deep Dive: Zero to Docker in a single book”, Released October 2020, Packt Publishing.

References:

1. Brendan Burns, Joe Beda, and Kelsey Hightower, “Kubernetes Up & Running: Dive into The Future of Infrastructure”, Released October 2019, O'Reilly Media, Inc.

Web Resources:

1. https://www.docker.com/blog/docker-and-kubernetes/