Kerala University of Digital Sciences, Innovation and Technology

M.Tech Computer Science and Engineering

(Cyber Security Engineering/ Artificial Intelligence)

M.Sc Computer Science (Two Year)

(Cyber Security/Artificial Intelligence)

M.Sc Cyber Security (One Year)

M.Sc Advanced Artificial Intelligence (One Year)

Scheme and Syllabus

2026 Admission Onwards

School of Computer Science and Engineering (SoCSE)

School of Computer Science and Engineering

The School of Computer Science and Engineering (SoCSE) at the Kerala University of Digital Sciences, Innovation, and Technology (KUDSIT) was established in 2020 at Technopark Phase IV, Thiruvananthapuram. The School offers postgraduate and doctoral programmes in various areas of computer science and engineering.

The School currently offers the following academic programmes:

- **M.Tech. in Computer Science and Engineering** (Specializations: Cyber Security Engineering /Artificial Intelligence)
- **M.Sc. in Computer Science** (Specializations: Cyber Security / Artificial Intelligence)
- **M.Sc. in Cyber Security** (1 Year)
- **M.Sc. in Advanced Artificial Intelligence** (1 Year)
- **Ph.D. Programmes**

Master of Technology (M.Tech.) in Computer Science and Engineering

The M.Tech. in Computer Science and Engineering is a two-year programme with two specializations: Artificial Intelligence and Cyber Security Engineering. Students are required to choose one of these specializations.

Master of Science (M.Sc.) Programmes

The M.Sc. programmes are offered in two formats: two-year programmes and one-year programmes.

The two-year M.Sc. in Computer Science programme includes two specializations: Cyber Security and Artificial Intelligence. Students are required to choose one of these specializations. The admission criteria and eligibility requirements will be the same for both specializations.

The one-year M.Sc. programmes are offered as M.Sc. in Cyber Security and M.Sc. in Advanced Artificial Intelligence. These programmes are intended for candidates seeking advanced specialization in these areas after completing a four-year undergraduate degree or a two-year master's degree or a five-year integrated master's degree.

Programme Structure for Two-Year Programmes

The minimum credit requirement for the award of a two-year master's degree is 80 credits. However, students may be allowed to take up to 90 credits by enrolling in additional courses or projects.

One credit is defined as one hour of contact time (lectures or tutorials) per week, or two hours of student workload (laboratory work, projects, assignments, etc.) per week. Since each semester consists of 15 teaching weeks, one credit corresponds to 15 hours of contact instruction per semester or 30 hours of student workload per semester.

A postgraduate degree programme may be completed in a minimum of two years and a maximum of four years. This period may include a zero year, if recommended by the Academic Committee and approved by the Vice Chancellor, in accordance with the Postgraduate Regulations, 2026 (KUDSIT-PG Regulations 2026).

The course levels and credits distribution across the various 2 year PG programmes are as follows:

Programme description for Two-year PG Programmes

	Mode of Delivery	**Minimum Credits**			
		Course Level	**Course work**	**Capstone Project/Thesis**	**Total Credits**
1st Year (S1 & S2)	Coursework	400 500	24 16	--	40
2nd Year (S3 & S4)	Coursework & Research	500	20	20	40
	Coursework	500	40	..	40
	Research	500	..	40	40

The credit distribution requirements across various types of courses:

Two-year PG programmes (M.Sc)

Core courses			Elective Courses		Additional courses
Programme Core	University Core	Capstone Project/ Thesis	Programme elective	Open elective	Additional Credits [Optional-beyond the mandatory coursework and project]
25 Credits	5 Credits	20 Credits	18 Credits	12 Credits	10 Credits

Two-year PG programmes (M.Tech)

Core courses			Elective Courses		Additional courses
Programme Core	University Core	Capstone Project/ Thesis	Programme elective	Open elective	Additional Credits [Optional-beyond the mandatory coursework and project]
16 Credits	5 Credits	40 Credits	12 Credits	8 Credits	10 Credits

Credit Requirements for the Two-Year PG Programmes

A. For a two-year PG programme, students may take a maximum of 45 credits per year, including both credit and audit courses. For one-year PG programmes, the maximum limit is 45 credits. However, to be eligible for the degree, students must complete the required credits across the various course categories as specified in the tables above. This credit limit restriction may be waived to allow repeat attempts for failed courses, subject to approval by the Competent Authority.

B. The minimum attendance requirement in a semester is 75%. This requirement does not apply to project-based courses, fieldwork-based courses, or research conducted outside the university. Any exemptions must be specified by the course instructor in the course description document and approved by the Competent Authority.

C. Students may take a maximum of 12 credits through audit courses. These credits do not count toward the total credit requirement of the programme.

D. Students may earn up to 12 credits through challenge examinations. These credits will count toward the total credit requirement of the programme.

E. A minimum of 35 credits (including programme electives and project/thesis work) is required for a major specialization, and 9 credits are required for a minor specialization (equivalent to three 3-credit courses or equivalent). Minor specialization credits must come from defined specialization streams specified through programme or open electives in the programme outline. Specialization credits may include courses, projects, activities, and related academic work. This clause applies only to programmes that offer dual specializations.

F. Up to 12 credits under elective courses (Programme Electives or Open Electives), as approved by the University, may be earned through the SWAYAM platform during the programme.

G. Up to 8 credits within the Open Elective category may be designated for training in emerging technologies, such as Artificial Intelligence, Cybersecurity, Internet of Things (IoT), Big Data Analytics, Blockchain, Intelligent Automation, Augmented Reality, and 3D Printing.

H. Individual or group mini-projects may be offered as Programme Electives or Open Electives. However, a maximum of 15 credits may be earned through this mode. Mini-projects are intended for research exploration or practical implementation in specialized subject areas relevant to the master's programme specialization.

Mode of Instruction for Two-Year PG Programmes

A wide range of modern teaching and learning methods is expected to be used in delivering the courses.

Course Level	Teaching	Evaluations
400 level	Advanced courses include lecture courses with practicums, seminar-based courses, term papers, research methodology, advanced laboratory experiments or software training, research projects, hands-on training, internship or apprenticeship projects at the undergraduate level, and first-year postgraduate theoretical and practical courses.	As per the University Examination Manual and as decided by the Controller of the Examinations of the University.
500 level	It provides an opportunity for original study or investigation in the major or field of specialization, conducted on an individual and more autonomous basis at the postgraduate level. All 500-level courses must include a course project with mandatory report submission and evaluation.	As per the University Examination Manual and as decided by the Controller of the Examinations of the University.

Pass Criteria

As stipulated in the University Examination Manual.

Programme Structure for One-Year PG Programmes

The minimum credit requirement for the award of a one-year master's programme is 40 credits. However, students may be allowed to take up to 45 credits by enrolling in additional courses or projects.

One credit is defined as one hour of contact time (lectures or tutorials) per week or two hours of student workload (labs, projects, assignments, etc.) per week. Since there are 15 teaching weeks in a semester, one credit corresponds to 15 hours of contact instruction per semester or 30 hours of student workload per semester.

A postgraduate degree programme may be completed in a minimum of one year and a maximum of two years.

The course levels and credit distribution across the various PG programmes are as follows:

Programme description for a one-year PG Programmes

	Mode of Delivery	Minimum Credits			
		Course Level	**Coursework**	**Research/thesis/ Project/Patent**	**Total Credits (Min)**
S1 & S2	Coursework + Research	500	20	20	40
	Coursework Only	500	40	-	40
	Research Only	-	-	40	40

The credit distribution requirements across various types of courses:

One-year PG programmes

Core courses		Elective Courses	
Programme Core	Capstone Project/ Thesis	Programme elective	Open elective
8 Credits	20 Credits	8 Credits	4 Credits

Mode of Instructions for 1 Year PG

A wide variety of modern teaching and learning methods are expected to be used in the delivery of the courses.

Course Level	Teaching	Evaluations
500 level	Original study or investigation in a major field of specialization, by an individual or a group	As per the University Examination Manual and as decided by the Controller of the Examinations of the University.

Pass Criteria

As stipulated in the University Examination Manual

2 Year M.Tech in Computer Science and Engineering with Specialization in Artificial Intelligence / Cyber Security Engineering
(AY 2026-27 Onwards)

Semester 1				
Course Code	**Title of the Course**	**Credits**	**Credit Split Lecture/Lab/ Seminar/Project**	**Level**
	Digital Access for Community Empowerment	5		500
M4010102	Advanced Data Structures and Algorithms	4	2-1-0-1	400
M4010103	Mathematical Foundations for Computer Science	4	2-1-0-1	400
M4010104	Programming in Python	4	2-1-0-1	400
M4010100/ M4010108	Programme Elective AI and Machine Learning /Computer Networks and Security	5	3-1-0-1	400
Total Credits		22		

Semester 2				
Course Code	**Title of the Course**	**Credits**	**Credit Split Lecture/Lab/ Seminar/Project**	**Level**
M4010116/ M4010112	Data and Intelligence / Cryptography	4	3-0-1-0/2-1-0-1	400
	Programme Elective	4		400
	Programme Elective	4		500
	Open Elective	4		500
	Open Elective	4		500
Total Credits		20		

Semester 3				
Course Code	**Title of the Course**	**Credits**	**Credit Split Lecture/Lab/ Seminar/Project**	**Level**
M5010151	Thesis I	20	0-0-0-20	500
Total Credits		20		

Semester 4				
Course Code	**Title of the Course**	**Credits**	**Credit Split Lecture/Lab/ Seminar/Project**	**Level**
M5010152	Thesis II	20	0-0-0-20	500
Total Credits		20		

Programme Electives for Artificial Intelligence (Minimum 15 Credits Required)				
Course Code	**Title of the Course**	**Credits**	**Credit Split Lecture/Lab/ Seminar/Project**	**Level**
M4010107	Data Analytics	4	2-1-0-1	400
M4010109	Robotics	4	2-1-0-1	400
M5010110	Deep Learning	4	2-1-0-1	500
M5010111	Natural Language Processing	4	2-1-0-1	500
M5010106	Reinforcement Learning	4	2-1-0-1	500
M5010107	Computer Vision	4	2-1-0-1	500
M5010108	Soft Computing	4	2-1-0-1	500
M5010109	Speech Processing	4	2-1-0-1	500
M5010143	Cognitive Computing	4	2-1-0-1	500
M5010135	Big Data Technologies	4	2-1-0-1	500
M5010136	Optimization Techniques	4	2-1-0-1	500
M5010130	Federated Learning	4	2-1-0-1	500
M5010145	Emerging Topics in Artificial Intelligence	4	2-1-0-1	500
	Approved Swayam Courses			

Programme Electives for Cyber Security Engineering (Minimum 15 Credits Required)				
Course Code	**Title of the Course**	**Credits**	**Credit Split Lecture/Lab/Seminar/Project**	**Level**
M4010114	Ethical Hacking and Penetration Testing	4	2-1-0-1	400
M4010137	Hardware Security	4	2-1-0-1	400
M5010118	AI for Cyber Security	4	2-1-0-1	500
M5010113	Cloud Security	4	2-1-0-1	500
M5010138	IoT Networks and Endpoint Security	4	2-1-0-1	500
M5010115	Systems Security and Risk Analysis	4	2-1-0-1	500
M5010140	Cyber Analytics	4	2-1-0-1	500
M5010117	Malware Analysis and Reverse Engineering	4	2-1-0-1	500
M5010120	Database Security	4	2-1-0-1	500
M5010121	Mobile Application Security	4	2-1-0-1	500
M5010122	Information Security Management System	4	2-1-0-1	500
M5010146	Emerging Topics in Cyber Security	4	2-1-0-1	500
	Approved Swayam Courses			

2 Year M. Sc. in Computer Science with Specialization in Cyber Security/Artificial Intelligence (AY 2026-27 Onwards)

Semester 1				
Course Code	**Title of the Course**	**Credits**	**Credit Split Lecture/Lab/ Seminar/Project**	**Level**
	Digital Access for Community Empowerment	5		500
M4020100/ M4020108	AI and Machine Learning / Computer Networks and Security	5	3-1-0-1	400
M4020102/ M4020105	Advanced Data Structures and Algorithms / Data Structures and Algorithms	4	2-1-0-1	400
M4020103	Mathematical Foundations for Computer Science	4	2-1-0-1	400
M4020104	Programming in Python	4	2-1-0-1	400
Total Credits		22		

Semester 2				
Course Code	**Title of the Course**	**Credits**	**Credit Split Lecture/Lab/ Seminar/Project**	**Level**
M4020116/ M4020112	Data and Intelligence / Cryptography	4	3-0-1-0/2-1-0-1	400
M4020131/ M4020114/	Data Analytics / Ethical Hacking and Penetration Testing	4	2-1-0-1	400
	Programme Elective	4		500
	Programme Elective	4		500
	Open Elective	4		500
Total Credits		20		

Semester 3				
Course Code	**Title of the Course**	**Credits**	**Credit Split Lecture/Lab/ Seminar/Project**	**Level**
	Programme Elective	4		500
	Programme Elective	4		500
	Programme Elective	4		500
	Open Elective	4		500
	Open Elective	4		500
Total Credits		20		

Semester 4				
Course Code	**Title of the Course**	**Credits**	**Credit Split Lecture/Lab/ Seminar/Project**	**Level**
M5020151	Project	20	0-0-0-20	500
Total Credits		20		

Programme Electives for Artificial Intelligence (Minimum 15 Credits Required)				
Course Code	**Title of the Course**	**Credits**	**Credit Split Lecture/Lab/ Seminar/ Project**	**Level**
M5020110	Deep Learning	4	2-1-0-1	500
M5020111	Natural Language Processing	4	2-1-0-1	500
M5020132	Stochastic Processes and Models	4	2-1-0-1	500
M5020133	Digital Image and Video Processing	4	2-1-0-1	500
M5020106	Reinforcement Learning	4	2-1-0-1	500
M5020107	Computer Vision	4	2-1-0-1	500
M5020108	Soft Computing	4	2-1-0-1	500
M5020109	Speech Processing	4	2-1-0-1	500
M5020143	Cognitive Computing	4	2-1-0-1	500
M5020135	Big Data Technologies	4	2-1-0-1	500
M5020136	Optimization Techniques	4	2-1-0-1	500
M5020130	Federated Learning	4	2-1-0-1	500
M5020145	Emerging Topics in Artificial Intelligence	4	2-1-0-1	500
	Approved Swayam Courses			
	Individual/Group Mini-projects/Research Activity	4	0-0-0-4	500

.

Programme Electives for Cyber Security (Minimum 15 Credits Required)				
Course Code	**Title of the Course**	**Credits**	**Credit Split Lecture/Lab/ Seminar/Project**	**Level**
M5020139	Digital Forensics	4	2-1-0-1	500
M5020118	AI for Cyber Security	4	2-1-0-1	500
M5020113	Cloud Security	4	2-1-0-1	500
M5020138	IoT Networks and Endpoint Security	4	2-1-0-1	500
M5020115	Systems Security and Risk Analysis	4	2-1-0-1	500
M5020140	Cyber Analytics	4	2-1-0-1	500

M5020117	Malware Analysis and Reverse Engineering	4	2-1-0-1	500
M5020120	Database Security	4	2-1-0-1	500
M5020121	Mobile Application Security	4	2-1-0-1	500
M5020122	Information Security Management System	4	2-1-0-1	500
M5020146	Emerging Topics in Cyber Security	4	2-1-0-1	500
	Approved Swayam Courses			
	Individual/Group Mini-projects/Research Activity	4	0-0-0-4	500

1 Year M. Sc. in Cyber Security (AY 2026-27 Onwards)

Semester 1				
Course Code	**Title of the Course**	**Credits**	**Credit Split Lecture/Lab/ Seminar/Project**	**Level**
M5050108	Computer Networks and Security	5	3-1-0-1	500
	Programme Elective	4		500
	Open Elective	4		
M5050150	Minor Project	8	0-0-0-8	500
Total Credits		21		

Semester 2				
Course Code	**Title of the Course**	**Credits**	**Credit Split Lecture/ Lab/ Seminar/Project**	**Level**
M5050112	Cryptography	4	2-1-0-1	500
	Programme Elective	4		500
M5050151	Major Project/Research Work/Internship	12	0-0-0-12	500
Total Credits		20		

Programme Electives				
Course Code	**Title of the Course**	**Credits**	**Credit Split Lecture/ Lab/ Seminar/Project**	**Level**
M5050114	Ethical Hacking and Penetration Testing	4	2-1-0-1	500
M5050118	AI for Cyber Security	4	2-1-0-1	500
M5050115	Systems Security and Risk Analysis	4	2-1-0-1	500
M5050117	Malware Analysis and Reverse Engineering	4	2-1-0-1	500
M5050121	Mobile Application Security	4	2-1-0-1	500
M5050146	Emerging Topics in Cyber Security	4	2-1-0-1	500
	Approved Swayam Courses			500
	Individual/Group Mini-projects/Research Activity	4	0-0-0-4	500

1 Year M. Sc. in Advanced Artificial Intelligence (AY 2026-27 Onwards)

Semester 1				
Course Code	**Title of the Course**	**Credits**	**Credit Split Lecture/Lab/Seminar/Project**	**Level**
M5050100	Artificial Intelligence	4	2-1-0-1	500
M5050131	Computational Data Analytics	4	2-1-0-1	500
M5050103	Mathematical Foundations for Artificial Intelligence	2	1-1-0-0	500
	Programme Elective	4		500
	Programme Elective	4		500
	Open Elective	4		500
Total Credits		22		

Semester 2				
Course Code	**Title of the Course**	**Credits**	**Credit Split Lecture/Lab/Seminar/Project**	**Level**
M5050152	Major Project/Research Work/Internship	20	0-0-0-20	500
Total Credits		20		

Programme Electives				
Course Code	**Title of the Course**	**Credits**	**Credit Split Lecture/Lab/Seminar/Project**	**Level**
M5050116	Data & Responsible Intelligent Systems	4	3-0-1-0	500
M5050110	Deep Learning and Generative AI	4	2-1-0-1	500
M5050111	NLP and Agentic AI	4	2-1-0-1	500
M5050107	Computer Vision	4	2-1-0-1	500
M5050109	Speech Processing	4	2-1-0-1	500
M5050106	Reinforcement Learning	4	2-1-0-1	500
M5050144	AI Driven Software Engineering	4	2-1-0-1	500
M5050145	Emerging Topics in Artificial Intelligence	4	2-1-0-1	500
	Approved Swayam Courses			500

Open Electives offered by SoCSE				
Course Code M. Tech/M. Sc	**Title of the Course**	**Credits**	**Credit Split Lecture/Lab/ Seminar/Project**	**Level**
M5010141	Technical Communication	2	1-1-0-0	500
M4020101	Introduction to Cyber Security	4	2-1-0-1	400
M4020142	Operating Systems	4	2-1-0-1	400
M4020119	Computer Architecture	4	2-1-0-1	400
M5020124	OOPS and JAVA	4	2-1-0-1	500
M5020123	Web Technology	4	2-1-0-1	500
M5020125	Object Oriented Software Engineering	4	2-1-0-1	500
M5010126 / M5020126	Blockchain Technology	4	2-1-0-1	500
M5010127 / M5020127	Augmented and Virtual Reality	4	2-1-0-1	500
M5010128 / M5020128/ M5050128	Quantum Computing	4	2-1-0-1	500
M5010129 / M5020129	Cloud and Edge Computing	4	2-1-0-1	500
M5010130/ M5020130	Federated Learning	4	2-1-1-0	500
M5010143/ M5020143	Cognitive Computing	4	2-1-0-1	500
M5010108/ M5020108	Soft Computing	4	2-1-0-1	500
M5050144	AI Driven Software Engineering	4	2-1-0-1	500
	Courses from other specializations (within the school)			
M5020147	Emerging Topics in Computer Science	4	2-1-0-1	500
	Approved SWAYAM Courses			
	Individual/Group Mini-projects/Research Activity	4	0-0-0-4	500

Course Syllabus

ADVANCED DATA STRUCTURES AND ALGORITHMS

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M4010102/ M4020102	Advanced Data Structures and Algorithms	2-1-0-1	2026

Prerequisites: Students should possess the fundamental programming skills in Computer Programming Languages such as Python.

Course Objective:

1. Understand fundamental data structures and algorithms and the tradeoffs between various implementations of these abstractions.

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

CO1: Analyse existing algorithms to identify their efficiency and evaluate the suitability of advanced data structures in problem-solving.

CO2: Design and create algorithms for novel computational problems while evaluating their correctness and performance.

CO3: Evaluate the complexity of algorithms using advanced concepts, including NP-completeness, randomization, approximation, and parameterized techniques.

CO4: Create optimized algorithmic solutions tailored for specific applications, justifying the choice of data structures and techniques.

CO5: Analyse algorithmic trade-offs and design strategies to address computational challenges with improved efficiency and scalability.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	1	2	1	3	1	1
CO2	1	2	2	2	3	2
CO3	2	1	3	3	2	2
CO4	2	1	3	3	3	2
CO5	1	2	2	2	3	1

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	Various Algorithm Design Strategies. Revising Asymptotic Complexity Analysis, Sorting, Searching and Divide and Conquer Algorithm strategy.
2	Trees: Balanced Binary Search Trees (AVL trees) and Splay Trees. Graphs: Weighted graphs, Basic Graph Algorithms (BFS, DFS and applications), Strongly Connected Components.
3	Single-Source Shortest Paths and Minimum Spanning Trees: implementation through heaps, Greedy Algorithm design. All Pairs Shortest Paths and other Dynamic Programming examples.
4	Overview of P, NP Problems, NP-Completeness and a brief introduction to Randomization, Approximation and Parameterized Complexity.

Lab Exercises

Solving various problems using appropriate data structures. Use of recursion. Solving problems with Greedy, Divide and Conquer, and Dynamic programming techniques. Solving other interesting problems where data structures need to be used in an intelligent way.

Text Books

1. T.H. Cormenet al., Introduction to Algorithms, MIT Press, 2009.
2. B. N. Miller and D. L. Ranum, Problem Solving with Algorithms and Data Structures Using Python, Franklin, Beedle and Associates, 2011.

References

1. Y. Langsamet al., Data Structures using C, Asia: Pearson Education, 2004.
2. A. Drozdek, Data Structures and Algorithms in JAVA, 2nd ed., Brooks/Cole, 2002.
3. J. Kleinberg and E. Tardos, Algorithm Design, Pearson Education, 2006.
4. S. Dasgupta et al., Algorithms, New York: McGraw-Hill Higher Education, 2008.

AI AND MACHINE LEARNING

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M4010100 /M4020100	**AI and Machine Learning**	**3-1-0-1**	**2026**

Prerequisites: Nil

Course Objectives:

1. To impart algorithmic skills for designing AI and machine learning techniques and solutions.
2. To equip the students to identify and analyse problems solvable with AI/machine learning algorithms/techniques.
3. To impart solution design capability with AI/machine learning techniques.

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

CO1: Algorithm design capability in AI/Machine Learning

CO2: Algorithm analysis capability in AI/Machine Learning

CO3: Problem analysis skills in AI/Machine Learning Applications

CO4: Solution design capability in AI/machine learning Applications

CO5: Solution illustration capability in AI/Machine Learning Applications

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	3	3	2	2	2
CO2	3	3	3	2	2	2
CO3	2	3	3	3	3	3
CO4	2	3	3	2	3	3
CO5	2	3	3	3	3	3

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus	
Module	**Content**
1	Artificial Intelligence - Turing Test, Knowledge Representation, Rule/Logic based AI and Machine Learning Based AI, Importance of search in AI - gradient descent, modelling the brain - Perceptron, Back Propagation Algorithm, Narrow and General AI.
2	Machine Learning Paradigms - Supervised, Unsupervised and reinforcement Learning. Generalization performance, Bias-Variance tradeoffs, Feature Engineering - relevance, feature extraction - PCA. Supervised Learning:- Classification - Bayesian, Decision Tree and Random Forests, Ensemble Methods - Boosting, Regression - linear, logistic.

3	Unsupervised Learning - Density Estimation - Maximum Likelihood and Parzen Windows, Clustering- Partition Based, Subspace Clustering - CLIQUE, Incremental Clustering - Leader and BIRCH. Sequence Modelling - Hidden Markov Models.
4	Statistical Learning theory – Empirical Risk Minimization, and Structural Risk Minimisation: VC Dimension. Kernel Machines - Support Vector Machines, Support Vector Clustering.

Lab Exercises

Module 1:

Experiments on Google AI Experiments platform, Implementation of Perceptron

Module 2:

Implementation of PCA, Nave Bayes Classifier, Logistic Regression

Module 3:

Implementation of ML Estimation, K-Means and HMM

Module 4:

Experiments with SVM Libraries - SVM and Deep SVM

Text Books

1. S. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, 4th ed., Pearson, 2020.
2. S. Shalev-Shwartz and S. Ben-David, Understanding Machine Learning: From Theory toAlgorithms, Cambridge University Press, 2014.
3. I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning, MIT Press, 2016.

References

1. S. Haykin, Neural Networks and Learning Machines, 3rd ed., Pearson, 2009.
2. G. Bonaccorso, Mastering Machine Learning Algorithms, Packt Publishing, 2018.

AI FOR CYBER SECURITY

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5010118/ M5020118/ M5050118	**AI for Cyber Security**	**2-1-0-1**	**2026**

Prerequisites: A basic understanding of algebra, linear algebra, modular arithmetic

Course Objectives:

- To equip students with a solid understanding of rule-based AI, machine learning, and deep learning techniques for addressing various cybersecurity challenges.
- To enable students to apply machine learning and AI algorithms effectively in real-world cybersecurity applications such as malware detection, phishing prevention, and threat mitigation.

- To enhance students' ability to design and implement autonomous systems for cyber defense using advanced AI techniques such as reinforcement learning, and deep learning.

Course Outcomes: After completion of this course, the students would be able to:

CO1: Analyze the effectiveness of rule-based AI and machine learning models in detecting cybersecurity threats and mitigating risks.

CO2: Distinguish between supervised and unsupervised learning approaches for phishing detection, intrusion detection, and anomaly analysis in cybersecurity.

CO3: Evaluate classification algorithms such as Bayesian models, SVMs, Decision Trees, and ensemble methods for malware detection and threat prediction.

CO4: Design deep learning models, including CNNs, RNNs, LSTMs, and GANs, for advanced applications like attack pattern generation and encrypted traffic analysis.

CO5: Integrate feature engineering techniques, such as PCA and reinforcement learning, to construct adaptive and scalable solutions for real-time cyber threat mitigation.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	2	2	1	1	2
CO2	3	3	2	1	1	1
CO3	3	3	3	1	1	2
CO4	3	3	3	1	2	3
CO5	3	3	3	1	1	2

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus:

Module	Content
1	Rule-Based AI vs. Machine Learning in Threat Detection, Neural Networks and Learning Algorithms for Cybersecurity Applications, Bias-Variance Tradeoffs in Cybersecurity Models, Reinforcement Learning for Adaptive Threat Mitigation, PCA-Based Feature Engineering for Anomaly Detection, Supervised Learning in Phishing

	Detection, Unsupervised Learning for Intrusion Detection, Automated Cyber Defense with Reinforcement Learning.
2	Effectiveness of Bayesian, SVM, Decision Tree, and Random Forest classifiers in detecting cyber threats, Performance of ensemble methods in improving malware detection accuracy, Application of linear and logistic regression models for predicting security risks, Use of supervised learning algorithms in phishing and APT detection.
3	Partition-Based and Subspace Clustering for anomaly detection in network traffic, Incremental Clustering techniques for real-time cyber threat detection, Spectral Clustering in identifying hidden attack patterns, Hidden Markov Models for detecting rare cyber-attacks and advanced persistent threats.
4	Deep neural networks for cyber threat classification, Deep Feed Forward Networks for malware detection, Convolutional Neural Networks for image-based phishing page detection, Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) for time-series anomaly detection, Generative Adversarial Networks (GANs) for generating attack patterns, Autoencoders for unsupervised anomaly detection in network traffic.

Text Books:

1. Tony Thomas, Athira P. Vijayaraghavan, Sabu Emmanuel, Machine Learning Approaches in Cybersecurity Analytics, Springer 2020.
2. Tony Thomas, RoopakSurendran, Teenu S John, MamounAlazab, Intelligent Mobile Malware Detection, CRC Press, Taylor and Francis, 2023.
3. Artificial Intelligence: A Modern Approach 4th Edition, Stuart Russell and Peter Norvig, Pearson, 2020.
4. Understanding Machine Learning: From Theory to Algorithms, ShaiShalevShwartz, ShaiBen-David, Cambridge University Press, 2014.
5. Clarence Chio, David Freeman, Machine Learning & Security, O Reilly, 2018.
6. Deep Learning Applications for Cyber Security, Alazab, Mamoun, Tang, MingJian (Eds.), Springer, 2019.
7. Rakesh M. Verma, David J. Marchette, Cybersecurity Analytics, by Chapman and Hall/CRC, 2019.
8. Charu C. Aggarwal, Neural Networks and Deep Learning: A Textbook (Second Edition), Springer, 2023.

References:

1. Alexey Kleymenov, AmrThabet , Mastering Malware Analysis: The complete malware analyst's guide to combating malicious software, APT, cybercrime, and IoT attacks, 2019.
2. Monappa KA, Learning Malware Analysis: Explore the concepts, tools, and techniques to analyze and investigate Windows malware, Packt Publication, 2018.
3. Xin et al, Machine Learning and Deep Learning Methods for Cybersecurity, IEEE Access 2018.
4. Bowei Xi, Adversarial machine learning for cybersecurity and computer vision: Current developments and challenges, WIREs Computational Statistics, April 2020.
5. Mohammad Al-Rubaie, Privacy Preserving Machine Learning: Threats and Solutions, n IEEE Security and Privacy Magazine, 2018.
6. Aiyanyo et al, A Systematic Review of Defensive and Offensive Cybersecurity with Machine Learning, Applied Sciences, MDPI, Aug 2020.
7. Shaukat et al, A Survey on Machine Learning Techniques for Cyber Security in the Last Decade, IEEE Access, Dec 2020.
8. Anthony D. Joseph, Blaine Nelson, Benjamin I. P. Rubinstein, Adversarial Machine Learning, Cambridge University Press, 2019.

AI-DRIVEN SOFTWARE ENGINEERING

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5050144	**AI-driven Software Engineering**	**2-1-0-1**	**2026**

Prerequisites: Machine learning fundamentals and Basic Software Engineering Concepts.

Course Objectives:

1.To understand the evolution of Software Engineering and the role of Artificial Intelligence in transforming the SDLC.
2.Toapply machine learning techniques for software analytics such as effort estimation, defect prediction, and code analysis.
3.To utilize Generative AI models for code generation, documentation, testing, and refactoring tasks.
4.To integrate AI techniques into DevOps practices, AIOps, and autonomous software engineering systems.
5. To analyze security, ethical, and reliability challenges in AI-driven software development and evaluate emerging research trends.
6. To design and optimize predictive, generative, and agentic AI components within scalable software engineering frameworks.

Course Outcomes: After completion of this course, the students will be ableto,

CO1:Analyze limitations of traditional software engineering and explain AI integration across SDLC phases
CO2:Apply machine learning techniques for software analytics including defect prediction, effort estimation, and fault localization.
CO3:Design and evaluate generative AI-based solutions for code generation, testing, documentation, and refactoring.
CO4:Develop AI-driven DevOps and agent-based automation strategies for intelligent software deployment and monitoring.
CO5:Assess security, bias, explainability, and governance issues in AI-assisted software engineering systems.
CO6:Critically analyze current research trends and propose improvements in AI-driven software engineering practices.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.
PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.
PLO 3 Apply for a scholarship to conduct independent and innovative research.
PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.
PLO 5 Practice ethical standards of professional conduct and research.
PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	2	-	-	-	-
CO2	3	3	2	-	-	1
CO3	3	3	3	2	-	3
CO4	2	2	2	1	-	3
CO5	2	1	-	1	3	1
CO6	2	3	3	2	1	3

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus:

Module	Content
1	**Module I: Foundations of AI-Driven Software Engineering** Evolution of Software Engineering (SE) and its Limitations - Introduction to AI in Software Engineering - AI integration across SDLC - Machine Learning for software analytics: Effort estimation -Defect prediction -Fault localization – ML based Static and

	Dynamic code analysis - NLP for requirements engineering - Intelligent Software architecture support - Search-based software engineering.
2	**Module II: Generative AI for Software Development** Large Language Models for code generation - Prompt engineering for software tasks - Retrieval-Augmented Generation (RAG) for software artifacts - AI-assisted documentation and specification generation - Automated code review using LLMs -Test case generation using Generative AI - Programme synthesis and automated refactoring - Evaluation metrics and benchmarking framework for generative code systems.
3	**Module III: DevOps, AIOps, Agentic AI and Autonomous Software Engineering** **DevOps Foundations:** CI/CD pipelines - Infrastructure as Code – Monitoring and logging - AI-Driven DevOps (**AIOps**) - Log analysis and anomaly detection - Deployment failure prediction - Data-driven release decision support- Agile and Lean Practices in DevOps. **Introduction to Agentic AI** - AI agents in SDLC phases-Tool augmented LLMs and function calling-multi-agent orchestration frameworks-Autonomous CI/CD systems-Self-healing systems-Human-in-the-loop AI supervision.
4	**Module IV: Secure and Responsible AI driven Software Engineering** Security risks in AI generated code - Prompt injection and adversarial threats - Bias and reliability in AI-assisted development - Explainable AI for software systems - Responsible AI in SDLC- Federated learning for secure software analytics- Evaluation frameworks for AI-driven SE systems-Research trends in AI in Software Engineering-Case Studies from Industry and Academia.

Text Books and References

1) Zhang, D., & Tsai, J. J. (Eds.). (2005). Machine learning applications in software engineering (Vol. 16). World Scientific.
2) Maria Virvou, George A. Tsihrintzis, Nikolaos G. Bourbakis, Lakhmi C. Jain. (Eds.). (2022), Handbook on Artificial Intelligence-Empowered Applied Software Engineering, VOL.1: Novel Methodologies to Engineering Smart Software Systems, Springer Cham
3) Romero, J. R., Medina-Bulo, I., & Chicano, F. (Eds.). (2023). Optimising the software development process with artificial intelligence. Berlin: Springer.
4) Benala, T. R., Dehuri, S., Mall, R., & Favorskaya, M. N. (Eds.). (2025). Boosting Software Development Using Machine Learning. Springer.
5) Kadry, S., &Balasubramaniam, S. (Eds.). (2025). Generative AI for Software Development: Code Generation, Error Detection, Software Testing. Walter de Gruyter GmbH & Co KG.
6) Valentina Alto(2024), Building LLM Powered Applications, Packt Publishing Ltd.
7) Paul Iusztin, Maxime Labonne(2024), LLM Engineer's Handbook, Packt Publishing Ltd.
8) Chip Huyen(2024), AI Engineering: Building Applications with Foundation Models, O'Reilly Media
9) Humble, J., & Farley, D. (2018). The DevOps Handbook: How to Create World-Class Agility, Reliability, & Security in Technology Organizations. IT Revolution.

10) Navin Sabharwal and Gaurav Bhardwaj(2022). Hands-on AIOps: Best Practices Guide to Implementing AIOps,Apress Standard
11) Shaila Rana, Rhonda Chicone(2025), Generative AI Security: Defense, Threats, and Vulnerabilities, Wiley-IEEE Press.
12) Vaishnavi Gudur, Bishwajeet Pandey, Advait Patel (Eds.). (2026). Trustworthy AI Systems: Engineering Secure, Scalable, and Responsible Intelligence for Real Applications, Springer Cham

ARTIFICIAL INTELLIGENCE

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5050100	**Artificial Intelligence**	**2-1-0-1**	**2026**

Prerequisites: Prior knowledge of Programming and Computer Science

Course Objectives is to:

1. Understand the theoretical foundations and evolution of AI.
2. Design and analyze AI systems across the lifecycle.
3. Implement and evaluate machine learning and neural models.
4. Apply statistical learning and kernel methods for robust modeling.
5. Develop and assess explainable and interpretable AI systems.

Course Outcomes: After completion of this course, the students would be able to:

CO1: Explain the theoretical foundations and evolution of Artificial Intelligence.

CO2: Analyze and design AI systems using knowledge engineering and learning frameworks.

CO3: Apply machine learning paradigms and neural architectures to real-world problems.

CO4: Evaluate model performance using statistical learning theory and kernel methods.

CO5: Implement and assess explainability techniques for interpreting AI models.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.
PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.
PLO 3 Apply for a scholarship to conduct independent and innovative research.
PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.
PLO 5 Practice ethical standards of professional conduct and research.
PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	2	1	1	2	1
CO2	3	3	2	2	2	2
CO3	3	3	3	2	2	2
CO4	3	3	3	3	2	3
CO5	3	3	3	2	3	3

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	Natural Intelligence, Artificial Intelligence, Rational Agent Framework, Turing Test and its modern variants, Strong AI vs Weak AI, Narrow AI vs General AI, Subfields of AI and their interrelationships, Symbolic AI and Logic-Based Systems, Knowledge-Based Systems, Data-Driven AI, Neuro-Symbolic Integration, Evolution from Rule-Based Systems to Foundation Models, Contemporary AI systems and capabilities, Applications, Performance and Limitations, Advancements of AI in the new era.
2	AI life cycle, Problem Solving and Searching Techniques: Problem characteristics, breadth-first search, depth-first search, hill climbing and its variations, heuristics search techniques: best-first search, A* algorithm, constraint satisfaction problem. Learning in problem-solving, learning from examples (Induction), Explanation-Based Learning (EBL), Need for Explainable AI, Interpretability vs Explainability, Black-box vs Interpretable models. modify the below content with recent trends in AI.
3	Supervised, Unsupervised and Reinforcement Learning, Generalization performance, Classification: Bayesian methods, Decision Trees, Random Forests, Ensemble methods – Boosting and Bootstrap. Density Estimation: Maximum Likelihood and Parzen Windows, Clustering: Partition-Based, Subspace Clustering, Incremental Clustering, Spectral Clustering, Hidden Markov Models.
4	Empirical Risk Minimization, Structural Risk Minimization, VC Dimension, Support Vector Machines, Multi layered perceptron, Neural Networks, Loss functions, Vanishing Gradient Problem, Intrinsic vs Post-hoc explanations, Model-specific vs Model-agnostic approaches, Local vs Global explanations, Feature importance methods, Explainable AI-LIME, SHAP, Partial Dependence Plots, Saliency maps, Counterfactual explanations, Performance Analysis.

Text Books

1. S. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, 4th ed., Pearson, 2020.
2. S. Shalev-Shwartz and S. Ben-David, Understanding Machine Learning: From Theory to Algorithms, Cambridge University Press, 2014.

References

1. S. Haykin, Neural Networks and Learning Machines, 3rd ed., Pearson, 2009.
2. G. Bonaccorso, Mastering Machine Learning Algorithms, Packt Publishing, 2018.
3. W. Ertel, Introduction to Artificial Intelligence, Springer, 2018.
4. M. Coeckelbergh, AI Ethics, The MIT Press Essential Knowledge Series, 2020.
5. M. Wooldridge, A Brief History of Artificial Intelligence: What It Is, Where We Are, and Where We Are Going, Flatiron Books, 2021.

AUGMENTED AND VIRTUAL REALITY

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5010127/ M5020127	Augmented and Virtual Reality	2-1-0-1	2026

Prerequisites: Nil

Course Objectives:

1. To provide students with an understanding of concepts and frameworks of immersive technologies.
2. To help students get familiarized with the hardware and software of AR/VR systems.
3. To help the students develop immersive technology applications.

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

CO1: Design and manage large-scale immersive virtual environments for real-time applications.
CO2: Evaluate research gaps in AR/VR systems and propose innovative solutions for emerging challenges.
CO3: Develop and integrate AR/VR technologies for diverse interdisciplinary applications.
CO4: Justify the selection of AR/VR tools and techniques for specific real-world problem-solving scenarios.
CO5: Critique existing AR/VR systems to recommend enhancements and optimize performance.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.
PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.
PLO 3 Apply for a scholarship to conduct independent and innovative research.
PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.
PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	**3**	**1**	**3**	**3**	**1**	**1**
CO2	**1**	**3**	**3**		**3**	**1**
CO3	**3**		**3**	**3**	**3**	**3**
CO4	**3**	**3**	**1**	**1**	**1**	**2**
CO5	**3**	**3**	**2**	**3**		**1**

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus	
Module	**Content**
1	Real vs Virtual World, Degree/Level of Immersion (3-DoF vs 6-DoF), Sensory Immersion, Cognitive Immersion, Emotional Immersion, Physical Immersion, Virtual World Coordinate System, Real and Virtual Experiences (Aural, Visual, and Haptic), Historical Background of VR and AR, Importance and Applications of VR and AR, Evolution of VR, Three I's of Virtual Reality, VR-Continuum, XR and Immersive Technologies, Difference between VR, MR, and AR, Methods of Interaction, Key Milestones in VR Development, Concepts in Virtual Reality, Motion Tracking, Rendering Techniques, Depth Perception in the Virtual World, Gestures and Cues, Challenges.
2	Components of VR, VR Display, VR Development Tools and Technologies, Sensors and Input Devices, VR System Integration, VR System Performance Optimization, Components of XR, Mixed Reality and AR in XR, AR Display, Marker-Based and Markerless AR, AR Development Tools and Technologies, AR System Integration, AR System Performance Optimization, Best Practices in XR Development, Development Practices, AR/VR Development Lifecycle, Prototyping and Storytelling, Understanding the Production Pipeline (Sensing, Rendering, Mobile, Stand-Alone, and High-End Computing Platforms), Testing and Debugging in VR/AR, Project Management in VR/AR Development.
3	Web-based VR Development, ARML, X3D Standard, WebXR API, Performance Optimization for Web-based VR, Mobile-based VR Development, Using ARCore Framework, Mobile VR Performance Optimization, Mobile VR Interaction Design, Cross-Platform Development Considerations, AR Development, GLTF (and Other Similar Standards), Web AR and Mobile AR, ARKit Framework, AR Content Creation and Management, AR User Interface Design.

4	Advanced Tools, Unity/Unreal/Blender, Basic VR Solution with Unity, Creating Custom Markers with Unity/Unreal, Marker-Based AR Development, Markerless AR Development, Scripting and Programming in Unity/Unreal, Applications and Trends in XR, Application of XR, Case Studies, Recent Developments in XR, Future Trends in XR, XR and Metaverse, Ethical Considerations in XR, XR Impact on Society and Culture, Security and Privacy in XR, Security in Immersive Technologies, Privacy Concerns in VR/AR, Regulatory and Compliance Issues.

Text Books

1. T. Thomas, R. Prakash, and A. Kaushik, Augmented and Virtual Reality: Theory and Practice. Wiley, Nov. 2025.
2. G.C. Burdea and P. Coiffet, Virtual Reality Technology, 2nd ed., Wiley-IEEE Press, 2003/2006.
3. A. B. Craig, Understanding Augmented Reality, Concepts and Applications, Morgan Kaufmann, 2013.
4. A. B. Craig and W. R. Sherman, Understanding Virtual Reality: Interface, Application, and Design, 2002.
5. S. M. LaValle, Virtual Reality, Cambridge University Press, 2017.
6. J. G. Tromp et al., Emerging Extended Reality Technologies for Industry 4.0 Early Experiences with Conception, Design, Implementation, Evaluation and Deployment, Wiley 2020.
7. S. Aukstakalnis, Practical Augmented Reality: A Guide to the Technologies, Applications, and Human Factors for AR and VR, Pearson Education, 2016.

References

1. A. B. Craig et al., Developing Virtual Reality Applications: Foundations of Effective Design, Morgan Kaufmann, 2009.
2. T. Jung and M. Cluaudia, Augmented Reality and Virtual Reality, Empowering Human, Place and Business, Springer International Publishing, 2018.
3. D. Schmalstieg and T. Höllerer, Augmented Reality: Principles and Practice, Boston: Addison-Wesley, 2016.
4. S. Greengard, Virtual Reality, MIT Press, 2019.
5. D. Vroegop, Microsoft HoloLens Developer's Guide, Packt Publishing, 2017.
6. M. Lanham, Learn ARCore-Fundamentals of Google ARCore: Learn to Build Augmented Reality Apps for Android, Unity, and the Web with Google ARCore 1.0, Packt Publishing, 2018.
7. S. Ong, Beginning Windows Mixed Reality Programming: For HoloLens and Mixed Reality Headsets, Springer, 2021.
8. P. Fuchs, Virtual Reality Headsets - A Theoretical and Pragmatic Approach, CRC Press, 2017.

BIG DATA TECHNOLOGIES

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5010135/ M5020135	Big Data Technologies	2-1-0-1	2026

Prerequisites: Nil

Course Objectives:

1. Introduce the fundamental concepts, challenges, and tools of Big Data technologies.
2. Equip students with skills to design, develop, and deploy Big Data solutions.
3. Familiarize students with distributed computing frameworks like Hadoop and Spark.
4. Prepare students to solve real-world problems using scalable Big Data architectures.

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

CO1: Analyse the key concepts and ecosystem of Big Data technologies.
CO2: Develop and deploy applications using distributed computing frameworks such as Hadoop and Spark.
CO3: Apply data preprocessing, storage, and retrieval techniques in large-scale data management.
CO4: Implement machine learning algorithms and data analytics on Big Data platforms.
CO5: Evaluate and optimize the performance of Big Data systems for real-world applications.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.
PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.
PLO 3 Apply for a scholarship to conduct independent and innovative research.
PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.
PLO 5 Practice ethical standards of professional conduct and research.
PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PL02	PL03	PL04	PLO5	PL06
C01	3	3	2	1	2	1
CO2	3	2	1	1	1	1
CO3	3	3	1	1	1	2
CO4	3	3	2	1	2	2
CO5	3	3	2	2	2	2

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content

1	**Introduction:** Overview of Big Data - Characteristics, challenges, and applications - Big Data ecosystem and architecture - Overview of Hadoop: HDFS, MapReduce, and YARN- Introduction to NoSQL databases: Types, examples, and use cases.
2	**Distributed Computing Frameworks**: Deep dive into Hadoop: Advanced HDFS and MapReduce concepts - Apache Spark: Architecture, RDDs, DataFrames, and Datasets. Spark Streaming and GraphX - Introduction to Apache Kafka and its integration with Spark.
3	**Data Analytics and Machine Learning in Big Data** - Data preprocessing and exploratory data analysis (EDA) on Big Data -Machine learning with Spark MLlib: Classification, regression, and clustering -Introduction to deep learning frameworks for Big Data - Visualization tools for Big Data analytics (e.g., Tableau, Power BI).
4	**Big Data Platforms and Case Studies:** Overview of cloud-based Big Data platforms (e.g., AWS, Google Cloud, Azure). Data governance, security, and ethical considerations in Big Data. Performance optimization and benchmarking. **Case studies:** Big Data solutions in industry (e.g., finance, healthcare, e-commerce).

Reference Books

1. White, T. Hadoop: The Definitive Guide. O'Reilly Media, 2015.
2. Lam, C. Hadoop in Action. Manning Publications, 2011.
3. Karau, H., Konwinski, A., Wendell, P., &Zaharia, M. Learning Spark: Lightning-Fast Big Data Analysis. O'Reilly Media, 2015.
4. Hurwitz, J., Nugent, A., Halper, F., & Kaufman, M..Big Data For Dummies. Wiley, 2013.
5. Maheshwari, A. Data Analytics Made Accessible. Kindle Edition, 2017.
6. Berman, J. J. Big Data: Principles and Best Practices of Scalable Real-Time Data Systems. Elsevier, 2018.

BLOCKCHAIN TECHNOLOGY

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5010126 / M5020126	**Blockchain Technology**	**2-1-0-1**	**2026**
Prerequisites: Nil			

Course Objectives:

1. To provide students with a deeper understanding of the concepts of blockchain technology with due focus on decentralized computing and distributed systems.
2. To help the students develop the ability to address real-world problems using the learned concepts of smart contracts and Dapps.
3. To connect the learned concepts with other business domains having opportunities for disruptive innovation with blockchain.
4. To make students aware of the existing challenges of blockchain and focus on contributing revolutionary solutions of the same.

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

CO1: Analyse the security challenges in blockchain systems and evaluate various strategies.
CO2: Analyze the variants of blockchain/DLT and their adoption in respective domains.
CO3: Evaluate the scalability and performance of different programmemable blockchains.
CO4: Develop and evaluate blockchain-based frameworks to ensure transparency and efficiency in supply chain management.
CO5: Analyse the integration of blockchain with emerging technologies to create advanced systems for data integrity and automation.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.
PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.
PLO 3 Apply for a scholarship to conduct independent and innovative research.
PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.
PLO 5 Practice ethical standards of professional conduct and research.
PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	1	1	2	2	2	3
CO2	1	2	3	3	3	2
CO3	1	1	3	3	3	2
CO4	1	1	2	2	3	2
CO5	1	2	2	2	3	3

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content

1	Fundamentals of Blockchain technology: Centralized Vs Decentralized Computing, Concept of Distributed Ledger. Cryptographic principles - Encryption Techniques, Block Ciphers, Hash Functions (SHA), Digital Signatures, Public-Key Cryptography (RSA, ECDSA), Merkle Trees, DAG, PKI. Distributed Systems - Basic principle, design, architecture, Inter-process communication, peer-to-peer networks. Features of Blockchain. Blockchain vs Database, Blockchain vs Internet.
2	Blockchain network: Byzantine Generals Problem, Consensus Approach - PoW, PoS, pBFT. Working of Bitcoin network - Nodes, Forks, Mining, Wallets, UTXO Model. Challenges of Blockchain Technology. Blockchain Architectures: Public, Private, Hybrid. Potential Threats. - 51% attack, Sybil and Eclipse attacks.
3	Programmemable Blockchains - Smart Contracts, Dapps. Introduction to Ethereum - Architecture, EVM. Token Standards - Fungible and Non-fungible (ERC). Hyperledger Umbrella Projects. Corda DLT. Why or Why Not Blockchain. Next Generation Blockchains - Cardano, Algorand, Polkadot. Application of Blockchain- Banking, Supply chain, Governance.
4	Advanced Concepts - ZKPs, Sharding and sidechains, Layer-2 Protocols solving Blockchain Trilemma. Decentralized Finance (DeFi), Decentralized Autonomous Organizations (DAO). SegWit. BIP and EIP.

Lab Experiments

Experiments will be done with Ethereum and Hyperledger Fabric

Text Books

1. I. Bashir, Mastering Blockchain: A Deep Dive into Distributed Ledgers, Consensus Protocols, Smart Contracts, DApps, Cryptocurrencies, Ethereum, and More, 3rd ed., Packt Publishing, 2020.
2. D. Tapscott and A. Tapscott, *Blockchain Revolution: How the Technology Behind Bitcoin and Other Cryptocurrencies is Changing the World*, Portfolio Penguin, 2018.
3. A. M. Antonopoulos and G. Wood, *Mastering Ethereum: Building Smart Contracts and DApps*, O′Reilly 2018.
4. Asharaf S, et. al, Blockchain Technology: Algorithms and Applications, Wiley, 2023.
5. Asharaf S, et. al, Advanced Blockchain Technologies: A Comprehensive Textbook on Ethereum Smart Contracts to Decentralised Applications, Wiley, 2025.

References

1. S. Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System, 2009.
2. A. Lewis, The Basics of Bitcoins and Blockchains: An Introduction to Cryptocurrencies and the Technology that Powers Them (Cryptography, Crypto Trading, Digital Assets, NFT), Mango Media, 2018.

CLOUD AND EDGE COMPUTING

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction

M5010129 / M5020129	Cloud and Edge Computing	2-1-0-1	2026

Prerequisites: Basic knowledge in Computer Systems and Concepts

Course Objectives:

1. To impart a comprehensive and in-depth understanding of Cloud and Edge Computing basics, technologies and applications to students by introducing and researching cutting-edge topics, technologies, applications and implementations.
2. To expose the students to frontier areas of Cloud and Edge Computing while providing sufficient foundations for further study and research.

Course Outcomes: After completion of this course, the students would be able to:

CO1: Analyze cloud computing architectures, service models, and deployment strategies to assess their applications and limitations.
CO2: Design cloud and edge computing solutions using the tools such as Cloud SIM ,AWS and Google Cloud.
CO3: Evaluate cloud technologies and edge computing frameworks by exploring real-world case studies and emerging trends.
CO4: Integrate concepts like IoT services to develop interdisciplinary solutions for computing challenges.
CO5: Develop and present a research project demonstrating expertise in Cloud application design, implementation, and critical analysis.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.
PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.
PLO 3 Apply for a scholarship to conduct independent and innovative research.
PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.
PLO 5 Practice ethical standards of professional conduct and research.
PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

		PLO1	PLO2	PLO3	PLO4	PLO 5	PLO6	
	CO1	3	2	1	2	1	2	
	CO2	3	2	2	2	1	2	
	CO3	2	2	2	2	2	1	
	CO4	2	2	2	3	3	1	
	CO5	2	2	2	2	2	2	

(Correlation: 1: Slight (Low) 2: Moderate (Medium)3: Substantial (High))

Syllabus	
Module	**Content**
1	**Introduction to cloud computing** – Definition – Characteristics – Cloud Service Models: IaaS, PaaS and SaaS – Deployment Models: Public, Private, Hybrid and Community – Cloud Services Examples – Cloud Architecture – Cloud Reference Model (NIST Architecture) - Cloud Computing Characteristics and Applications
2	**Cloud Concepts and Technologies**: Virtualization – Load Balancing – Scalability – Elasticity – Deployment – Replication – Monitoring – Software Defined Networking – Network Function Virtualization – Map Reduce – Identity and Access Management – Service Level Agreements – Billing – Service Oriented Architecture – RESTful Web Services – Publish Subscribe Model
3	**Exploring Cloud Services and Platforms**: Compute Services – Storage Services – Database Services – Application Services – Content Delivery Services – Analytics Services – Deployment and Management Services – Identity and Access Management Services – Open-Source Private Cloud Softwares.
4	**Edge Computing**: Definition – Scope – Key Difference between Cloud and Edge Computing – Evolution – Edge Devices and Hardware – Communication Protocols and Networking in Edge Computing – Overview of Edge Computing Frameworks: Kubernetes, Open Horizon, AWS IoT Greengrass - Programmeming Models for Edge Computing Applications – Future Research Directions in Edge Computing. **Case Studies and Use Cases of Cloud and Edge Computing**

Text Books

1. K. Chandrasekaran, Essentials of Cloud Computing, CRC Press, 2015.
2. ArshdeepBahga, Vijay Madisetti, Cloud Computing: A Hands-On Approach, Universities Press (India) Private Limited, 2014.
3. R. Buyya et al., Mastering Cloud Computing, McGraw-Hill, 2013.
4. C. Surianarayanan and P. Chelliah, Essentials of Cloud Computing: A Holistic Perspective, 1st ed., Springer, 2019.
5. Rajkumar Buyya and Satish Narayana Srirama, Fog and Edge Computing: Principles and Paradigms, Wiley, 2019.
6. Elias Krishnasamy et al., Edge Computing: An Overview of Framework and Applications, PRACE Technical Report, 2020.
7. R. Buyya, S. N. Srirama, Fog and Edge Computing: Principles and Paradigms, Wiley, 2019.
8. J. R. Vacca, Cloud Computing Security: Foundations and Challenges, CRC Press, 2016.
9. B. Burns et al., Kubernetes: Up and Running: Dive Into the Future of Infrastructure, O'Reilly, 2019.
10. A. A. A. Donovan and B. W. Kernighan, The Go Programmeming Language, Addison-Wesley, 2015.

11. S. Klabnik, C. Nichols, The Rust Programmeming Language, No Starch Press, 2018.
12. J. S. Chelladhurai, V. Singh, and P. Raj, Learning Docker, 2nd ed., Packt Publishing, 2017.
13. A. Kurniawan, Learning AWS IoT, Packt Publishing, 2018.
14. E. Krishnasamya et al., Edge Computing: An Overview of Framework and Applications, PRACE Technical Report, 2020.

Online Resources:

1. AWS Whitepapers and Technical Guides: https://aws.amazon.com/whitepapers/
2. AWS Documentation : https://docs.aws.amazon.com/
3. AWS Architecture Center : https://aws.amazon.com/architecture/

CLOUD SECURITY

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5010113/ M5020113	**Cloud Security**	**2-1-0-1**	**2026**

Prerequisites: Cloud Computing

Course Objectives:

1. Understand the principles and challenges of cloud computing security, including risk assessment, legal considerations.
2. Learn about the Key strategies and best practices for securing the cloud environment.
3. Gain practical skills in securing cloud environments using Amazon Web Services (AWS), including monitoring, encryption, key management, threat detection, and compliance management.

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

CO1: Analyse security concerns, risks, and legal aspects of cloud computing, focusing on virtualization, provisioning, storage, and networking, to identify and mitigate vulnerabilities.

CO2: Evaluate cloud security architectures and strategies, including encryption methods, sensitive data categorization, and solutions for vendor lock-in, to ensure secure data storage and compliance.

CO3: Apply advanced monitoring and auditing practices using AWS tools like CloudWatch, CloudTrail, and AWS Config to gain actionable insights into cloud operations and enhance compliance.

CO4: Utilize AWS security tools, including KMS, Web Application Firewall, and Shield, to address modern threats such as DDoS attacks and data breaches.

CO5: Create optimized and secure cloud environments by integrating AWS features and adhering to best practices for cloud data protection and system monitoring.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.
PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.
PLO 3 Apply for a scholarship to conduct independent and innovative research.
PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.
PLO 5 Practice ethical standards of professional conduct and research.
PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	2	1	1	3	1
CO2	3	1	2	1	3	2
CO3	3	2	2	1	3	2
CO4	3	1	2	1	3	2
CO5	2	1	2	2	2	2

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus:

Module	Content
1	Cloud Computing Security Concerns, Risk issues and Legal Aspects: Virtualization, Provisioning, Storage, Operation, Security and Networking – Risk Tolerance – Legal and Regulatory Issues. Key Strategies and Best Practices.
2	Cloud Architecture security – Security Requirements, Security Patterns and Architectural elements, Cloud Security Architecture Cloud Data security – Overview – Data Encryption – Sensitive Data Categorization – Cloud Data Storage – Cloud Lock in
3	Amazon AWS Monitoring & Auditing: AWS Monitoring Introduction, Cloud watch metrics – Cloud watch Logs, Live Tail, Agents -Cloud watch Alarms, Even Bridge – Cloud Trail – AWS Config
4	AWS Security – Introduction, Encryption, KMS, Multi-Region Keys, S3 Replication with encryption, Encrypted AMI Sharing process, SSM Parameter Store, AWS secrets Manager, AWS certificate Manager, Web Application Firewall, Shield – DDoS protection, Amazon Guard Duty, Amazon Inspector, Amazon Macie

Text Books

1. J.R. Winkler, Securing the Cloud: Cloud Computer Security Techniques and Tactics, 1st ed., Elsevier, 2011.
2. T. Mather, S. Kumaraswamy, and S. Latif, Cloud Security and Privacy, 1st ed., O'Reilly, 2009.

References

3. R. Vacca, Cloud Computing Security Foundations and Challenges, CRC Press, 2017.
4. Ronald L. Krutz and Russell Dean Vines, Cloud Security: A Comprehensive Guide to Secure Cloud Computing", Wiley, 2010.
5. Michael J. Kavis, Architecting the Cloud: Design Decisions for Cloud Computing Service Models (IaaS, PaaS, and SaaS), Wiley, 2014.

Online Resources:

1. AWS Security: https://docs.aws.amazon.com/security/
2. AWS Monitoring and Observability: https://docs.aws.amazon.com/monitoring/
3. AWS Compliance and Legal Resources: https://aws.amazon.com/compliance/

COGNITIVE COMPUTING

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5010143 / M5020143	**Cognitive Computing**	**2-1-0-1**	**2026**

Prerequisites: 10th class biology and chemistry, basic back ground in simple differential equations and probability theory, interest in neuroscience and cognitive science.

Course Objectives:

1. To provide students with a basic understanding of the concepts of neuroscience, cognitive science, and cognitive computing.
2. To help them understand how to connect the concepts of cognitive science and neuroscience to the computing domain.
3. To inform students of current research trends in cognitive computing and artificial emotional intelligence.

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

CO1: Analyse various cognitive and emotional processes in the brain/mind and evaluate their applications in the computing domain.
CO2: Evaluate cognitive and affective computing models and systems for their design, implementation, and effectiveness.
CO3: Create innovative research ideas by integrating concepts from cognitive science and computing to address complex challenges.
CO4: Analyse the interdisciplinary relationship between cognitive science and computing to identify opportunities for advancing the field.
CO5: Create cognitive and affective computing prototypes or frameworks through collaborative research and development initiatives.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.
PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.
PLO 3 Apply for a scholarship to conduct independent and innovative research.
PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.
PLO 5 Practice ethical standards of professional conduct and research.
PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	2	1	1	1	1
CO2	3	3	2	1	1	2
CO3	2	3	3	2	2	3
CO4	3	2	2	2	1	2
CO5	2	3	2	3	2	3

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	Basic neuroscience: Neurons, Dendrites and Axons, Synapses, Synaptic and Action Potentials, Action Potential generation and propagation, Brain organization, anatomy and functions, Synaptic integration and plasticity, the Concept of a Basic Circuit, Abstractions of Cortical Basic Circuits, Neocortical Brain Organization. Neuron models - McCulloch-Pitts, Integrate-and-Fire, Hodgkin-Huxley.
2	Cognitive psychology of decision making, neural basis, Scientific theories and measures of Consciousness, Cognitive models of memory, Mental Imagery, Understanding a problem, a cybernetic view of cognition consciousness and free will. Hierarchical temporal memories, Brain Simulations, Eye Tracking and other modalities for data acquisition. Scope of Realization of Cognition in Artificial Intelligence.
3	Brain Computer Interface: Types – Synchronous and Asynchronous, Invasive- Partially Invasive - Non-Invasive BCI, Structure of BCI System, BCI Monitoring Hardware-EEG, EEG Pre-processing Techniques, Analysis -time, spatial and frequency domains, fMRI, neuro imaging tools, Brain Response useful for Building BCIs, BCI applications. Emotions and Machines; Theories, models and neural basis of emotions, computational models for synthetic emotion simulation and dynamics, application of artificial emotional intelligence in healthcare, video surveillance.

4	Introduction to Brain networks, graph models for complex systems, graph theory and brain, connectivity at microscale. Clinical applications of brain network analysis, network visualization, case studies. Demonstration and tools for computing different connectivity measures and their visualizations.

References

1. E. Kandel et al., Principles of Neural Science, McGraw-Hill Professional, 2012.
2. E. Bruce Goldstein, Cognitive Psychology: Connecting Mind, Research, and Everyday Experience, 4th ed., Cengage Learning, 2014.
3. Rao, R. P. N., Brain Computer Interfacing: An Introduction, Cambridge University Press, 2013.
4. N. Panigrahi and S. P. Mohanty, Brain Computer Interface EEG Signal Processing, CRC Press, 2022.
5. A. Ortony, G. L. Clore, and A Collins, The Cognitive Structure of Emotions, Cambridge University Press, 2011.
6. J. Friedenberg and G. Silverman, Cognitive Science: An Introduction to the Study of Mind, Sage Publications, 2021.
7. M. Gazzaniga, Cognitive Neuroscience: The Biology of the Mind, W. W. Norton, 2018.

Computational Data Analytics

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5050131	**Computational Data Analytics**	**2-1-0-1**	**2026**

Prerequisites: Basic knowledge in Machine learning, statistics and Python

Course Objectives is to:

1. Provide students with a good understanding of the concepts of data analytics.
2. Help the students develop the ability to solve problems using the learned concepts.
3. Connect the concepts to other domains, such as machine learning and pattern recognition.

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

CO1: Analyze diverse data types and construct robust data preprocessing and wrangling pipelines using Python-based tools for reliable analytical workflows.
CO2: Apply statistical inference techniques, including estimation, hypothesis testing, and information-theoretic measures, to derive meaningful conclusions from data.
CO3: Develop and optimize predictive models using regression, simulation methods, neural networks, and ensemble techniques with appropriate performance evaluation and interpretability strategies.
CO4: Design scalable analytical solutions for high-dimensional, graph-based, and streaming data while addressing challenges such as concept drift and feature evolution.
CO5: Integrate computational, inferential, and modeling techniques to develop end-to-end data analytics solutions and effectively communicate analytical insights for research and real-world applications.

Programme Learning Outcomes:

PLO 1: Develop strong fundamental disciplinary knowledge.
PLO 2: Demonstrate research skills that are of experimental, computational, or theoretical nature.
PLO 3: Apply for a scholarship to conduct independent and innovative research.
PLO 4: Show communication skills in various formats (oral, written) and to expert and non-expert audiences.
PLO 5: Practice ethical standards of professional conduct and research.
PLO 6: Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	3	2	2	2	2
CO2	3	3	3	2	2	2
CO3	3	3	2	2	2	3
CO4	2	3	2	2	2	2
CO5	2	2	2	3	3	3

Mapping of course outcomes with programme learning outcomes:

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	Data Science and Python-based Data Handling Introduction to Data Science fundamentals, Nature of data and its characteristics, Total Information Awareness, Bonfoni's Principle, Rhine's paradox, Data preprocessing, Data wrangling, Data aggregation, Data exploration, Handling missing data: single and multiple imputation; Correcting inconsistent data: deduplication, entity resolution, pairwise matching; NumPy fundamentals, Pandas data structures and operations, Matplotlib visualization techniques.
2	Statistical Inference and Information-Theoretic Methods Sampling distributions, Central Limit Theorem, Point estimation, Properties of estimators, Minimum variance unbiased estimation, Maximum likelihood estimation, Method of moments, Consistency of estimators, Interval estimation, Hypothesis testing, Critical regions, Likelihood ratio tests, Correlation analysis, Covariance analysis, Inference analysis, Entropy, KL divergence, Entropy-based techniques, Statistical implementation using Python.
3	Predictive Modeling, Simulation, and Optimization Linear regression, Logistic regression, Regularization (L1, L2), Monte Carlo simulation, Markov Chain Monte Carlo (MCMC), Modelling the brain - Perceptron, Neural networks for analytics,, Back Propagation Algorithm, Advanced ensemble techniques, Hyperparameter optimization, Model tuning strategies, Model interpretability, Performance benchmarking, Predictive modeling using scikit-learn, Simulation using Python.

4	High-Dimensional, Streaming, and Emerging Analytics Dimensionality reduction, Bias-Variance tradeoffs, Feature Engineering - relevance, feature extraction - PCA, Feature selection, Graph data analysis, Network metrics, Stream processing, Online analytics, Infinite-length data handling, Concept drift, Concept evolution, Feature evolution, Visual analytics, Deployment strategies, Scalable analytics pipelines, Current trends and research directions.

Text Books

1. J. Leskovec, A. Rajaraman and J. Ullman, Mining of Massive Datasets, Cambridge University Press, 2014.
2. S. Ozdemir, Principles of Data Science, 2nd ed., Packt Publishing, 2018.
3. S. Lau, J. Gonzalez, and D. Nolan, Principles and Techniques of Data Science, UC Berkeley.
4. J. S. Saltz and J. M. Stanton, An Introduction to Data Science, Sage Publications, 2017.

References

1. R. V. Hogg, J. W. McKean and A. Craig, Introduction to Mathematical Statistics, 6th ed., India: Pearson Education, 2006.
2. D. Cielen, A. D.B. Meysman, and M. Ali, Introducing Data Science: Big Data, Machine Learning, and More, Using Python Tools, Simon and Schuster, 2016.
3. G. Grolemund and H. Wickham, R for Data Science, O'Reilly, 2017.
4. N. Zumel and J. Mount, Practical Data Science with R, Simon and Schuster, 2014.

COMPUTER ARCHITECTURE

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M4020119	**Computer Architecture**	**2-1-0-1**	**2026**

Prerequisites: Nil

Course Objectives:

1. To help students understand the fundamentals behind a computer and its architecture.
2. To explore the working principles of a computer's essential building blocks.
3. To understand how these building blocks are assembled to design a so-called computer.
4. To explore a few advanced topics in computer architecture.

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

CO1: Analyse the functioning of various components in a computer system and evaluate their interactions.

CO2: Apply knowledge of computer architecture to model and analyse systems for security vulnerabilities.

CO3: Compare various types of computer architectures and analyse their design principles, identifying strengths and weaknesses.

CO4: Create efficient models of computer systems based on an understanding of their components and architectures.

CO5: Evaluate the performance and functionality of a computer system, applying acquired knowledge of its components to optimise usage.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.
PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.
PLO 3 Apply for a scholarship to conduct independent and innovative research.
PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.
PLO 5 Practice ethical standards of professional conduct and research.
PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	**3**	**1**	**1**	**2**	**2**	**2**
CO2	**3**	**3**	**2**	**2**	**3**	**2**
CO3	**2**	**3**	**2**	**2**	**3**	**1**
CO4	**2**	**2**	**1**	**3**	**2**	**3**
CO5	**3**	**3**	**3**	**2**	**3**	**2**

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	Computer Fundamentals: Computer types, functional units, Basic concepts. Von Neumann Architecture Instruction Sets: Machine instructions, Memory operations, addressing modes, Instructions sets, Stacks, Subroutines, RISC & CISC architectures.
2	Processing Unit: Components (Registers, ALU, Datapath), Instruction execution, Control signals, Operations of control unit. Computer Arithmetic: Basic operations on signed numbers, Floating point operations.
3	Memory Management: Memory Hierarchy, Semiconductor based memory (Internal Organization, SRAM, DRAM), Read only memory, Cache memories – mapping techniques, performance, locality of reference, Cache hit / miss, Cache coherence problem Input/output: Accessing I/O devices, Bus Operations, I/O Modules, I/O Control mechanisms – Programmemed I/O, Interrupt controlled, Direct Memory Access, I/O Interface (Serial, Parallel), I/O interconnection Standards.

4	Pipelining: Pipeline concept, Speedup, Throughput, Hazards inpipeline –structural hazard, datahazard, control hazard: Branch hazard; Dealing with hazards - Register Renaming, Branch Prediction. Advanced Computer Architecture: Parallel Processing- Flynn's classification, Amdahl's law, Characteristics of Multiprocessors, Interconnection Structures, Interprocessor Arbitration, Interprocessor Communication and Synchronization, Cache Coherence, Vector/Array Processing.

Text Books

1. C. Hamacher et al., *Computer Organization*, 6th ed., McGraw-Hill Higher Education, 2011.
2. D. A. Patterson and J. L. Hennessy, *Computer Organization and Design – The Hardware/Software Interface*, 6th ed., Morgan Kaufmann, 2020.
3. W. Stallings, *Computer Organization and Architecture: Designing for Performance*, 8th ed., Pearson, 2009.
4. P. P. Chaudhuri, *Computer Organization and Design*, 3rd ed., PHI Learning, 2008.
5. A. S. Tanenbaum, *Structured Computer Organization*, 6th ed., Pearson, 2012.

References

1. William Stallings, *Computer Organization and Architecture: Designing for Performance,* 7th ed., Prentice-Hall India, 2005.
2. C. Hamacher, Z. Vranesic, and S. Zaky, *Computer Organization*, 6th ed., McGraw Hill, 2012.
3. M. M. Mano, *Digital Logic and Computer Design*, 4th ed., Pearson Education, 1979.

COMPUTER NETWORKS AND SECURITY

Course Code	Course Name	Credit Split	Year of
		Lecture/Lab/Seminar/Project	**Introduction**
M4010108/ M4020108/ M5050108	**Computer Networks and Security**	**3-1-0-1**	**2026**

Prerequisites: Nil

Course Objectives:

1. To introduce the fundamental aspects of computer networks.
2. To enable the students to understand various cyber-attacks targeted on computer networks.
3. To enable the students to develop various security mechanisms for computer networks.
4. To enable the students to simulate various network attacks.

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

CO1: Apply network protocols (TCP, UDP, HTTP) to design and implement basic applications and evaluate their performance using metrics such as delay, throughput, and loss.
CO2: Analyze the functionalities of the transport, network, and data link layers, including routing algorithms, congestion control, and error correction methods, to solve communication challenges.
CO3: Evaluate error detection, correction techniques, and multiple access protocols for improving network efficiency and reliability.
CO4: Analyze and simulate network vulnerabilities and attacks in traditional and IoT environments using tools like NS2/NS3 or Contiki, and propose security solutions.
CO5: Analyse the ability to design and implement secure network solutions by applying key management, user authentication protocols, and endpoint security techniques.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.
PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.
PLO 3 Apply for a scholarship to conduct independent and innovative research.
PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.
PLO 5 Practice ethical standards of professional conduct and research.
PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	2	1	1	3	1
CO2	3	1	2	1	2	2
CO3	3	2	2	1	2	2
CO4	3	1	2	1	2	2
CO5	2	2	3	2	2	2

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	Network Basics: The Network Edge, The Network Core, Access Networks, Delay, Loss and Throughput, Protocol Layers and Their Service Models, Application Layer: RPC, P2P, HTTP, FTP, DNS, DHCP, Electronic Mail, WLAN, Socket, Programming with TCP and UDP
2	Transport Layer: Services, TCP, UDP, Network Layer: Functions, design issues, Internet Protocol (IP), IPV4 and IPv6, Routers, Routing algorithms, Congestion Control Algorithms

3	Data Link Layer: Design issues, framing methods, Error Detection and Correction, PPP, Sliding Window Protocols, Multiple Access Protocols, Address Resolution, Protocol (ARP), Ethernet, Link Layer Switches, Spanning Tree Protocol, VLAN
4	Security Attacks, Security Services, Security Mechanisms, Key Management and Distribution, User Authentication Protocols, SSL, TLS, Wireless Network Security, Electronic Mail Security, Vulnerability Analysis, Attacks in sensor and IoT networks, Endpoint Security, familiarization of Network simulators - NS2/NS3 or Cooja/Contiki and simulation of attacks and analyze network performance.

Text Books

1. J. Kurose and K. Ross, Computer Networking: A Top-Down Approach, 7th ed., Pearson, 2016.
2. A. S. Tanenbaum, Computer Networks, 5th ed., Pearson, 2013.
3. W. Stallings, Cryptography and Network Security Principles and Practice, Prentice Hall, 1998.
4. V. Tsiatsiset al., Internet of Things: Technologies and Applications for a New Age of Intelligence, Elsevier Academic press, 2018.
5. Z. Mahmood, ConnectedVehiclesintheInternet of Things: Concepts, Technologies and Frameworks for IoT, Springer, 2020.
6. I. F. Akyildiz and M. Can Vuran, Wireless Sensor Networks, Wiley, 2010.

References

1. L. L. Peterson and B. S. Davie, Computer Networks: A Systems Approach, Morgan Kaufmann, 2011.
2. S. Keshav, An Engineering Approach to Computer Networking, Pearson Education, 2000.
3. S. S. Shinde, Computer Network, New Age International, 2009.
4. P. Raj and A. C. Raman, The Internet of Things: Enabling Technologies, Platforms, and Use Cases, 1st ed., Auerbach Publications, 2017.
5. A. McEwen, Designing the Internet of Things, Wiley, 2013.

COMPUTER VISION

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5010107/ M5020107/ M5050107	**Computer Vision**	**2-1-0-1**	**2026**

Prerequisites: Mathematics

Course Objectives:

1. To provide students with a good understanding of computer vision concepts.
2. To help the students develop the ability to solve problems using the learned concepts.
3. Connect the concepts to other domains, such as machine learning and pattern recognition, within and without computer vision.

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

CO1: Analyse the foundations of modern computer vision theory, identify key challenges, and evaluate state-of-the-art solutions.
CO2: Evaluate the design, development, and integration of computer vision algorithms and systems for their effectiveness and applicability.
CO3: Analyse real-world scenarios to identify opportunities for applying computer vision techniques to solve complex problems.
CO4: Create and demonstrate a functional computer vision system through collaborative research projects, detailed reports, and presentations.
CO5: Create innovative computer vision models and frameworks by leveraging advanced computational and mathematical techniques.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.
PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.
PLO 3 Apply for a scholarship to conduct independent and innovative research.
PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.
PLO 5 Practice ethical standards of professional conduct and research.
PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	2	1	1	1	1
CO2	3	3	2	1	1	2
CO3	2	3	2	2	1	2
CO4	2	3	3	3	2	3
CO5	2	3	3	3	2	3

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	The Four Rs of Computer Vision, Geometry of Image Formation and Sensing, Single/Two View Geometry, Camera Calibration, Vanishing Points, Planar Scenes and Homography, Interest Point Detection, Robust Correspondence Estimation
2	Feature Extraction: Edges - Canny, LoG, DoG; Line detectors (Hough Transform), Corners - Harris and Hessian Affine, Orientation Histogram, SIFT, SURF, HOG, GLOH, Scale-Space Analysis- Image Pyramids and Gaussian derivative filters, Gabor Filters and DWT.
3	Image Segmentation: Region Growing, Edge Based approaches to segmentation, Graph-Cut, Mean-Shift, MRFs, Texture Segmentation; Object detection
4	Motion Analysis: Background Subtraction and Modelling, Optical Flow, KLT, Spatio-Temporal Analysis, Dynamic Stereo; Motion parameter estimation.

Text Books

1. R. Szeliski, *Computer Vision: Algorithms and Applications*, London: Springer, 2011.
2. D. A. Forsyth and J. Ponce, *Computer Vision: A Modern Approach*, Pearson Education, 2003.
3. R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed., Cambridge University Press, 2004.

References

1. S. J. D. Prince, *Computer Vision: Models, Learning, and Inference*, 1st ed. USA: Cambridge University Press, 2012.
2. E. R. Davies, *Computer Vision: Principles, Algorithms, Applications, Learning*, 5th ed. USA: Academic Press, 2017.

CRYPTOGRAPHY

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M4010112/ M4020112/ M5050112	**Cryptography**	**2-1-0-1**	**2026**

Prerequisites: A basic understanding of algebra, linear algebra, modular arithmetic

Course Objectives:

1. To explore and apply modern cryptographic standards and protocols, including symmetric encryption, public-key cryptography, and key management systems, for securing communications and data.
2. To introduce and assess post-quantum cryptographic techniques, including lattice-based, code-based, and hash-based algorithms, to ensure secure systems in the face of quantum computing threats.
3. To develop the skills necessary for implementing cryptographic solutions in real-world applications, evaluate their security, and recommend strategies for transitioning to quantum-safe cryptography.

Course Outcomes: After completion of this course, the students would be able to:

CO1: Identify and compare modern cryptographic standards, including AES, ChaCha20, RSA, and ECC, assessing their applications and security strengths.
CO2: Evaluate the performance of modern cryptographic standards and recommend strategies for their implementation.
CO3: Design secure cryptographic systems using hybrid encryption, SHA-2, SHA-3, and HMAC for data integrity and authentication.
CO4: Appraise quantum-safe cryptography algorithms (CRYSTALS-Kyber, Dilithium, SPHINCS+) and predict their impact on future security.
CO5: Evaluate the transition to post-quantum cryptographic systems and recommend strategies for their implementation.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.
PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.
PLO 3 Apply for a scholarship to conduct independent and innovative research.
PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.
PLO 5 Practice ethical standards of professional conduct and research.
PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	2	2	1	1	2
CO2	2	3	3	2	2	2
CO3	3	2	3	2	2	1
CO4	2	3	3	1	1	2
CO5	3	3	3	2	1	2

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus:

Module	Content
1	Overview of cryptographic standards and protocols, Symmetric encryption: AES and its applications, ChaCha20 and its applications
2	Key management systems and protocols: Diffie-Hellman, ECDH, Multi-factor authentication (MFA), Public-key cryptography: RSA, ECC, Hybrid encryption systems
3	Cryptographic hashing and message authentication: SHA-2, SHA-3, HMAC Digital signatures and authentication: RSA, ECDSA,
4	Post-quantum cryptography, Lattice-based cryptography: CRYSTALS-Kyber, Quantum-safe digital signatures: CRYSTALS-Dilithium, SPHINCS+, Transitioning to post-quantum cryptographic systems

Text Books:

1. William Stallings, Cryptography and Network Security: Principles and Practice, 8th edition, Pearson, 2024.
2. Christof Paar, Jan Pelzl, Tim Güneysu, Understanding Cryptography: From Established Symmetric and Asymmetric Ciphers to Post-Quantum Algorithms, Springer-Verlag Berlin and Heidelberg GmbH & Co. K; Second Edition, 2024.
3. Jean-Philippe Aumasson, Serious Cryptography: A Practical Introduction to Modern Encryption, No Starch Press, 2017.
4. Douglas Robert Stinson, Cryptography: Theory and Practice, Chapman and Hall/CRC; Standard Edition, 2018.
5. Zhiyong Zheng, Kun Tian, Fengxia Liu, Modern Cryptography Volume 2: A Classical Introduction to Informational and Mathematical Principle, Springer; 1st ed. 2023.

References:

1. Ross Anderson, Security Engineering: A Guide to Building Dependable Distributed Systems, Wiley, 2020.
2. Thomas R. Shemanske, A Beginner's Guide, Modern Cryptography and Elliptic Curves, American Mathematical Society, 2017.
3. NIST Standards, FIPS 203, FIPS 204 and FIPS 205.

CYBER ANALYTICS

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5010140/ M5020140	**Cyber Analytics**	**2-1-0-1**	**2026**

Prerequisites: Nil

Course Objectives:

1. To Introduce various supervises, unsupervised and reinforcement learning algorithms.
2. To enable the students to apply ML techniques to analyze cyber data.
3. To enable the students to perform cyber threat detection, risk estimation, vulnerability detection, and cyber-attack detection.
4. To make the students design ML-based cyber security solutions.

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

CO1: Demonstrate a comprehensive understanding of the concepts and importance of cybersecurity analytics in modern cyber defense.

CO2: Apply various data collection and preprocessing techniques to extract valuable insights from cybersecurity data.

CO3: Utilize data analysis techniques and machine learning algorithms for effective threat detection and categorization.

CO4: Employ artificial intelligence approaches, including deep learning, natural language processing, and generative models, for analyzing complex cybersecurity challenges.

CO5: Utilize a comprehensive data engineering and machine learning tool/platform to explore advanced techniques in cybersecurity analytics, including deep learning and GPT.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	2	3	2		
CO2	3	3	3	2	2	2
CO3	2	3	3	2	1	1
CO4	2	2	2	1	2	3
CO5	1	2	3	1	2	3

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	**Cyber Threat Intelligence and Data Collection** Understanding Cyber Threat Intelligence and its Significance, Effective Data Collection for Cybersecurity Insights, Data Preprocessing Techniques for Enhanced Analysis, Exploratory Data Analysis for Identifying Threat Indicators, Leveraging Machine Learning in Cybersecurity: Concepts and Techniques Use Case: Network Intrusion Detection using Machine Learning
2	**Advanced Threat Detection and Profiling** Advanced Techniques for Threat Detection and Categorization, Clustering and Classification Methods for Effective Analysis, Feature Engineering and Selection for Improved Detection, Profiling User and Entity Behavior for Insider Threat Detection, Real-time Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS) Use Case: Identifying Suspicious Insider Activities using Behavioral Analysis
3	**Machine Learning and AI for Threat Analysis** Harnessing Deep Learning for Intrusion Detection, Leveraging Natural Language Processing for Threat Analysis, Synthetic Data Generation using Generative Adversarial Networks (GANs), Explainable AI Models for Transparent Cybersecurity Analysis, Utilizing Machine Learning in Security Information and Event Management (SIEM) Use Case: Detecting Zero-Day Attacks with Deep Learning Techniques
4	**Incident Response and Cyber Big Data Analytics** Effective Incident Response Strategies: Analytics-driven Incident Handling, Incorporating Analytics into Incident Response Workflow, Monitoring Key Performance Indicators (KPIs) for Cyber Defense Use Cases: • Detecting and Responding to Advanced Threats with Analytics • Analyzing Insider Threats and Unauthorized Data Exfiltration Cyber Big Data Analytics: Role of Cyber Big Data in Identifying Emerging Threat Patterns, Scalable Storage and Processing Solutions for Large-scale Security Data Use Case: Predictive Analysis of Cyber Threats using Big Data Techniques

Text Books

1. T. Thomas et al., Machine Learning Approaches in Cybersecurity Analytics, Springer, 2020.
2. K. Harbott, Cybersecurity Analytics: The Evolution of Threat and Risk Management, Wiley, 2015.
3. M. Panella, R. Setola, and E. Bertino, Cybersecurity Analytics and Decision Support in Smart Grids, Springer, 2021.
4. R. Chandel and P. Sharma, Cybersecurity Analytics: A Hands-On Approach, Apress, 2020.
5. I. Santos, C. Laorden, and X. Ugarte-Pedrero, Data Science for Cybersecurity, Springer, 2018.
6. O. Savas and Y. Karaca, Big Data Analytics for Cybersecurity, CRC Press, 2018.

References

1. H. Xiong, Shekhar, and W. B. Croft, Applied Data Analytics: Principles and Applications, CRC Press, 2018.
2. S. Chen, J. Yan, and D. Z. Du, Big Data Analytics for Cyber-Physical Systems: Machine Learning for the Internet of Things, CRC Press, 2019.
3. E. D. Knapp and R. Samani, Applied Cyber Security and the Smart Grid: Implementing Security Controls into the Modern Power Infrastructure, Syngress, 2013

Data & Responsible Intelligence Systems

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5050116	**Data & Responsible Intelligence Systems**	**3-0-1-0**	**2026**
Prerequisites: Nil			

Course Objectives:

1. Impart the knowledge to understand the relevance of data and scope for building intelligent systems in various domains
2. Impart analytical skills needed to identify the need for responsible data intelligence
3. Equip with the relevant computational and algorithmic knowledge
4. To impart responsible data intelligent system design capability
5. Equip with knowledge to analyse the impact of responsible data intelligent systems in various domains

Course Outcomes: After completion of this course, the students would be able to:

CO1: Data Intelligence Problem identification capability in various domains

CO2: Analysis capability in responsible data intelligence problems

CO3: Knowledge in relevant Computational Methods/Algorithms

CO4: Solution design capability in responsible data intelligence systems

CO5: Knowledge in solution analysis and impact study.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	3	2	3	2	3
CO2	3	3	3	2	2	3
CO3	3	3	2	2	2	3
CO4	3	2	3	3	2	3
CO5	2	2	3	2	3	3

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	Data Intelligence and Decision Making, Data Architecture, Data Profiling, Data Storage - Data Lake, Data Quality and Integration, ETL process. Data Analytics Thinking, Exploratory Data Analysis, Multidimensional Analysis, Data Visualisation.

2	Decision Analytic Thinking, Association Mining and Applications. Big Data Environments and Knowledge Extraction. Collaborative Filtering - Amazon Case study, Enterprise Data Management - Collibra case study.
3	Data Governance Laws - GDPR, DPDP and Implications, Responsible Computing - Responsible AI - Technology Support for building responsible data intelligence systems. Bias in Algorithmic Fairness Case Study - Compas.
4	Collaborative Intelligence - Humans and AI-Copilots. AI for Scientific Discovery - Robotic Labs - ethical concerns, Intelligence in CRM - Telenor case study, Healthcare Intelligence - VideaHealth Case study, Retail Intelligence - Vispera case study, Manufacturing Intelligence - Dow Chemicals case study.

Lab/Assignment:

A few experiments with real world data sets and open source tools (individual/group) Case study presentations and discussions (group of maximum three)

Text Books:

1. Daniel T. Larose, Chantal D. Larose, Data Mining and Predictive Analytics, John Wiley and Sons, 2016.
2. Arvind Narayanan, Sayash Kapoor, AI Snake Oil, Penguin Random House India Pvt Ltd, 2024
3. Provost, F. and Fawcett, T., Data Science for Business, Shroff Publishers and Distributors Pvt. Ltd, 2014
4. HBR Case Studies

References:

1. Erl, T., Khattak, W. and Buhler, P., Big Data Fundamentals: Concepts, Drivers and Techniques, Pearson Education India, July 2016
2. Seth Stephens-Davidowitz, Everybody Lies: Big Data, New Data, and What the Internet Can Tell Us About Who We Really Are, HarperLuxe, 2017.
3. Ravi Bapna, Anindya Ghose, Thrive: Maximizing Well-being in the Age of AI, MIT Press, 2024.

DATA ANALYTICS

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M4020131	**Data Analytics**	**2-1-0-1**	**2026**
Prerequisites: Basic knowledge in Machine learning, statistics and Python			

Course Objectives:

1. To provide students with a good understanding of the concepts of data analytics.
2. To help the students develop the ability to solve problems using the learned concepts.
3. Connect the concepts to other domains, such as machine learning and pattern recognition, within and without data analytics.

Course Outcomes: After completion of this course, the students will be able to: **CO1**: Interpret the data analytics techniques and evaluate state-of-the-art solutions.
CO2: Infer real-life problems and adapt data analytics techniques to solve them.
CO3: Design, organize, and demonstrate data analytics solutions through team research projects.
CO4: Outline project outcomes and prioritize key findings for effective reporting.
CO5: Integrate analyses and present project reports to connect analytical insights with actionable recommendations.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.
PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.
PLO 3 Apply for a scholarship to conduct independent and innovative research.
PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.
PLO 5 Practice ethical standards of professional conduct and research.
PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course out comes with programme learning out comes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	3	2	2	2	2
CO2	3	3	3	2	1	2
CO3	2	1	1	2	3	3
CO4	2	2	1	2	2	2
CO5	1	2	1	2	2	2

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	Introduction to Data science fundamentals, Nature of Data and its characteristics, Total information awareness, Bonferroni's Principle, Rhine's paradox, Recap of Statistical and Inferential Analysis, Data preprocessing, Data wrangling, Data exploration, Dealing with missing data – single and multiple data imputation, Entropy based techniques.

2	Sampling distributions; Point estimation - estimators, minimum variance unbiased estimation, maximum likelihood estimation, method of moments, consistency; Interval estimation; Testing of hypotheses - tests and critical regions, likelihood ratio tests; Linear regression.
3	Monte Carlo and MCMC simulations; Correcting inconsistent data – Deduplication, Entity resolution, Pairwise Matching; Fellegi-Sunter Model, Advanced processing- Regression, Correlation, Covariance analysis, Aggregation, Sampling.
4	Dimensionality Reduction; Feature extraction and feature selection; Graph data analysis, Stream processing and online analytics, Dealing with infinite length, concept drift, concept/feature evolution, Visual analytics, Current trends and research.

Text Books

1. J. Leskovec, A. Rajaraman and J. Ullman, *Mining of Massive Datasets*, Cambridge University Press, 2014.
2. S. Ozdemir, *Principles of Data Science*, 2nd ed., Packt Publishing, 2018.
3. S. Lau, J. Gonzalez, and D. Nolan, *Principles and Techniques of Data Science*, UC Berkeley.
4. J. S. Saltz and J. M. Stanton, *An Introduction to Data Science*, Sage Publications, 2017.

References

1. R. V. Hogg, J. W. McKean and A. Craig, *Introduction to Mathematical Statistics*, 6th ed., India: Pearson Education, 2006.
2. D. Cielen, A. D.B. Meysman, and M. Ali, Introducing Data Science: Big Data, Machine Learning, and More, Using Python Tools, Simon and Schuster, 2016.
3. G. Grolemund and H. Wickham, R for Data Science, O'Reilly, 2017.
4. N. Zumel and J. Mount, Practical Data Science with R, Simon and Schuster, 2014.

DATA AND INTELLIGENCE

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M4010116/ M4020116	**Data and Intelligence**	**3-0-1-0**	**2026**

Prerequisites: Nil

Course Objectives:

1. To impart skills needed to identify and understand data problems.
2. To equip with analytical thinking on problems solvable with data intelligence.
3. To impart solution design capability with data intelligence.

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

CO1: Data Problem identification capability in various domains
CO2: Analysis capability in data problems
CO3: Knowledge in Computational Methods/Algorithms used to solve data problems
CO4: Solution design capability in data problems
CO5: Knowledge in solution analysis and impact study in data problems

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.
PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.
PLO 3 Apply for a scholarship to conduct independent and innovative research.
PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.
PLO 5 Practice ethical standards of professional conduct and research.
PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	3	2	3	2	3
CO2	3	3	3	2	2	3
CO3	3	3	3	2	3	3
CO4	3	3	3	2	3	3
CO5	2	2	3	3	3	3

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus:

Module	Content
1	Data Intelligence and Decision Making, Collaborative Intelligence - Humans and AI. Data Architecture, Data Profiling and Storage, Data Quality and Integration, ETL process.
2	Data Analytics Thinking, Exploratory Analysis, Multidimensional Analysis, OLAP, Data Visualization, Data Modelling, Overfitting and Underfitting.
3	Decision Analytic Thinking -Applications of Clustering, Classification and Association Mining. Big Data Environments and Knowledge Extraction. Enterprise Data Management - Collibra case study.
4	Responsible Data Intelligence - Digital Personal Data Protection Bill 2023, Intelligence in CRM - Telenor case study, Healthcare Intelligence - Videa Health Case study, Retail Intelligence - Vispera case study, Manufacturing Intelligence - Dow Chemical's case study.

Lab/Assignment

A case study presentation and discussion (by a group of three)

Text Books

1. F. Provost and T. Fawcett, Data Science for Business, Shroff Publishers and Distributors, 2014.
2. D. T. Larose and C. D. Larose, Data Mining and Predictive Analytics, John Wiley and Sons, 2016.
3. HBR Case Studies

References

1. T. Erl et al., Big Data Fundamentals: Concepts, Drivers and Techniques, India: Pearson Education, 2016.
2. S. Stephens-Davidowitz, Everybody Lies: Big Data, New Data, and What the Internet Can Tell Us About Who We Really Are, Harper Luxe, 2017.
3. Ravi Bapna, Anindya Ghose, Thrive: Maximizing Well-being in the Age of AI, MIT Press, 2024

DATABASE SECURITY

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5010120/ M5020120	Database Security	2-1-0-1	2026

Prerequisites: Nil

Course Objectives:

1. To teach different types of databases.
2. To teach the security aspects of databases
3. To perform data auditing.

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

CO1: Analyse various database models, design processes, and the ACID properties to evaluate their impact on data consistency and security.
CO2: Create advanced SQL queries, including secure practices to prevent SQL injection and ensure data integrity.
CO3: Compare and evaluate traditional RDBMS and NoSQL databases for handling structured, semi-structured, and unstructured data.
CO4: Apply access control mechanisms like Role-Based Access Control (RBAC) and Attribute-Based Encryption in database systems to ensure secure data management.
CO5: Create comprehensive database security strategies incorporating authentication, encryption, and auditing techniques to address emerging security challenges.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.
PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.
PLO 3 Apply for a scholarship to conduct independent and innovative research.
PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.
PLO 5 Practice ethical standards of professional conduct and research.
PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	2	1	3	2	
CO2	3	2	2	2	2	1
CO3	2	2	2	2	2	1
CO4	3	1	2	1	3	1
CO5	2	2	2	2	2	1

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content

1	Different Types of Databases, Entity Relationship Models, Relational Models, Relational Algebra, Calculus, ACID Properties, Relational Databases, Concurrency Control, Process of Database Design, Dependencies and Normalization for Relational Databases, Object-oriented/Object-Relational Models, Threats to the Database, Principles of Database Security, Levels of Database Security, Database Security Issues.
2	Introduction to SQL, SQL Features, SQL Operators, SQL Datatypes, SQL Parsing, Types of SQL Commands, Advanced Study of Structured Query Language, Querying Data from the database, Correlated Sub-queries, Joins, Hierarchical Queries, Bind Variables, Cursors, Functions, Stored Procedures, MySQL, Basics of New SQL Databases, SQL Injection and Mitigation.
3	Structured Data, Unstructured Data, Semi-Structured Data, Limitations of Traditional RDBMSs, SQL and Structured Data, SQL and Semi-Structured Data, SQL and Unstructured Data, The Emergence of NoSQL, NoSQL Database features, Types of NoSQL Databases, Search Engine Databases, Basics of MongoDB and Neo4j, Data Auditing, Statistical Database Security, Semantic Integrity Control, Privilege Analysis, Virtual Private Database (VPD), Data Redaction, Sensitive Data Protection.
4	Authentication and Authorization in DBMS, Properties and Basic Principles of Access Control Mechanisms, Views for Access Control, Classical Database Access Control: Discretionary Access Control, Role-Based Access Control and Mandatory Access Control; Access Control in Open Environments such as Attribute Based Encryption and Identity Based Encryption, Access Control in SQL, Network Data Encryption, Strong Authentication, Private Data Aggregation, Search in Encrypted Data : Searchable Encryption Overview, Selected Schemes on Searchable Encryption.

Text Books

1.A. Silberschatz et al., Database System Concepts, 6th ed., McGraw-Hill, 2011.

2.A. Meier and M. Kaufmann, SQL and NoSQL Databases: Models, Languages, Consistency Options and Architectures for Big Data Management, Springer, 2019.

3.G. Harrison, Next Generation Databases: NoSQL, NewSQL, and Big Data, Apress, 2015.

4.R. Elmasri and S. B. Navathe, Fundamentals of Database Systems, 6th ed., Pearson Education, 2011.

5.R. B. Vatan, Implementing Database Security and Auditing, Digital Press, 1st ed., 2005.

References

1. C. J. Date *et al.*, *An Introduction to Database Systems*, 8th ed., Pearson Education, 2006.
2. R. Elmasri and S. Navathe, *Fundamentals of Database Systems*, Pearson, 2000.
3. G. K. Gupta, *Database Management Systems*, McGraw-Hill, 2011.
4. J. Hellerstein and M. Stonebraker, *Readings in Database Systems (The Red Book)*, 4th ed., MIT Press, 2005.
5. J. L. Harrington, *Object Oriented Database Design Clearly Explained*, Harcourt, 2000.
6. R. Ramakrishnan, *Database Management Systems*, 4th ed., McGraw-Hill, 2015.
7. R. Ramakrishan and J. Gehrke, *Database Management Systems*, 3rd ed. McGraw-Hill, 2002.
8. S. Ceri and G. Pelagatti, *Distributed Databases: Principles and Systems*, Universities Press, 2000.
9. V. Atluri and P. Samarati, *Security of Data and Transaction Processing*, Springer, 2000.

DEEP LEARNING

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5010110/ M5020110	**Deep Learning**	**2-1-0-1**	**2026**

Prerequisites: AI and Machine Learning

Course Objectives:

1. To provide students with a good understanding of the concepts of the deep learning.
2. To help the students develop the ability to solve problems using the learned concepts.
3. To connect the concepts to other domains.

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

CO1: Analyse the foundations of modern deep learning theory, identify key challenges, and evaluate state-of-the-art solutions.
CO2: Apply deep learning algorithms and techniques to solve complex computational problems.
CO3: Evaluate the design, development, and integration of deep learning models and systems.
CO4: Create and demonstrate a functional deep learning system through collaborative research projects and comprehensive presentations.
CO5: Analyse real-world applications of deep learning to explore innovative opportunities for system development.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.
PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.
PLO 3 Apply for a scholarship to conduct independent and innovative research.
PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.
PLO 5 Practice ethical standards of professional conduct and research.
PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	2	1	1	1	1
CO2	3	3	2	1	1	2
CO3	2	3	3	3	2	3
CO4	2	3	3	3	2	3
CO5	3	2	2	2	1	2

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus	
Module	**Content**
1	Deep Feed forward Networks, Regularization for Deep Learning
2	Optimization for Training Deep Models. Convolutional Neural Networks, Sequence Modeling - Recurrent and Recursive Nets
3	Practical Methodology, Autoencoders, Representation Learning
4	Deep Generative Models, Applications of Deep Learning

Text Books

1. J. Patterson and A. Gibson, *Deep learning: A Practitioner's Approach*, O'Reilly, 2017.
2. I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016.
3. M. A. Nielsen, *Neural Networks and Deep Learning*, Determination Press, 2015.

References

1. L. Deng and D. Yu, *Deep Learning: Methods and Applications*, Now Publishers, 2013.
2. D. Koller and N. Friedman, *Probabilistic Graphical Models*, MIT Press, 2009.

Deep Learning and Generative AI

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5050110	**Deep Learning and Generative AI**	**2-1-0-1**	**2026**

Prerequisites:

Course Objectives is to:

1. Provide students with a good understanding of the concepts of deep learning.
2. Help the students develop the ability to solve problems using the learned concepts.
3. Connect the concepts to other domains.

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

CO1: Analyse the foundations of modern deep learning theory, identify key challenges, and evaluate state-of-the-art solutions.
CO2: Apply deep learning algorithms and techniques to solve complex computational problems.
CO3: Evaluate the design, development, and integration of deep learning models and systems.
CO4: Create and demonstrate a functional deep learning system through collaborative research projects and comprehensive presentations.
CO5: Analyse real-world applications of deep learning to explore innovative opportunities for system development.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.
PLO 2 Demonstrate research skills that are of experimental, computational, or theoretical nature.
PLO 3 Apply for a scholarship to conduct independent and innovative research.
PLO 4 Show communication skills in various formats (oral, written) and to expert and non-expert audiences.
PLO 5 Practice ethical standards of professional conduct and research.
PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	2	1	1	1	1
CO2	3	3	2	1	1	2
CO3	2	3	3	3	2	3
CO4	2	3	3	3	2	3
CO5	3	2	2	2	1	2

Mapping of course outcomes with programme learning outcomes:

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	Deep Feed forward Networks, Regularization for Deep Learning,Optimization for Training Deep Models.
2	Convolutional Neural Networks, Sequence Modeling - Recurrent and Recursive Nets.
3	Deep Generative Models – Autoencoders, Generative Adversarial Networks and Diffusion models
4	Generative AI-Prompt Engineering, Hallucinations, Bias and Model drift, prompt structure, chain of thought, Fine-Tuning and Model Adaptation, Pre-trained vs fine-tuned models, Parameter- efficient tuning- LoRA, adapter, Generative AI Models and tools, Performance -BLEU, ROUGE, FID, METEOR, Human evaluation methods, Ethics, Regulations and Responsible AI – Bias and fairness, data privacy and security, AI regulations, Responsible AI frameworks – Case studies

Text Books

1. Christopher M. Bishop and Hugh Bishop, *Deep Learning: Foundations and Concepts*, 1st ed. Springer, 2023.
2. I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning, MIT Press, 2016.
3. S. Prince, *Understanding Deep Learning*, MIT Press, 2023.

References

1. J. Patterson and A. Gibson, Deep learning: A Practitioner's Approach, O'Reilly, 2017.
2. M. A. Nielsen, Neural Networks and Deep Learning, Determination Press, 2015.
3. Elizabeth Sherly, Leena Pillai, Kavya Manohar, John Mc Crae, Machine Translation: Best Practices using Deep Learning and Generative AI, CRC Press , Taylor and Francis, 2025 (Preview copy)

DIGITAL FORENSICS

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5020139	Digital Forensics	2-1-0-1	2026

Prerequisites: Nil

Course Objectives:

1. To provide students with a solid understanding of digital forensics principles and evidence analysis techniques.
2. To equip students with skills for applying forensic methods in real-world investigations across various platforms.
3. To develop critical thinking and communication skills in reporting forensic findings and collaborating on investigations.

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

CO1: Solve complex digital forensic cases by employing systematic investigation processes, including evidence handling, data acquisition, and advanced forensic analysis techniques.

CO2: Demonstrate proficiency in analyzing storage media and operating systems to identify, extract, and interpret critical digital evidence across diverse platforms.

CO3: Identify patterns and anomalies in network and mobile device data to assess security incidents, reconstruct events, and support intrusion detection efforts.

CO4: Appraise the legal, ethical, and technical implications of forensic practices, including evidence admissibility, collaboration with law enforcement, and privacy concerns.

CO5: Develop innovative approaches and frameworks, such as blockchain-based custody or AI-driven forensic tools, to enhance the accuracy and efficiency of digital forensic investigations.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	3	3	2	2	2
CO2	3	2	3	2	2	1
CO3	3	3	3	2	2	1
CO4	2	2	2	3	3	2
CO5	3	3	3	2	1	3

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus:

Module	Content
1	**Foundations of Digital Forensics**

	Computer Forensics Fundamentals, Digital Evidence and Handling, Forensic Investigation Process, Legal and Ethical Considerations, Collaboration with Law Enforcement, Reporting and Communication, Blockchain for Custody, AI in Analysis, Forensic Visualization, Evidence Admissibility Frameworks.
2	**Storage Media Analysis** Characteristics of Disk Drive Types, Logical Disk Structure, Booting Processes of Windows, Linux, and Mac OS; File Systems and Examination Techniques for Windows, Linux, and Mac OS; Fundamentals of Data Acquisition and Duplication, Acquisition Formats and Methodologies, Advanced Disk Drive Forensics, Cross-Platform File System Analysis, Novel Data Acquisition Techniques, Automation in Forensic Duplication.
3	**Operating System Forensics** Volatile and Non-Volatile Information, Windows Memory Forensics, Registry Analysis, Web Browser Cache, Cookie, and History Analysis, Windows File and Metadata Analysis, Hibernation File Analysis, Crash Dump Analysis, File System Analysis. Linux and Mac Forensics: Volatile and Non-Volatile Data in Linux, Filesystem Image Analysis Using Sleuth Kit, Memory Forensics, Mac Forensics, Advanced Memory Forensics Techniques, Cross-Platform Volatile Data Analysis, Automation in Filesystem Image Examination, Novel Methods in Metadata and Registry Analysis.
4	**Network and Mobile Forensics** Network Forensics: Protocols and Packet Capturing with Wireshark, tshark, tcpdump; PCAP Filtering, Network Logs Analysis (Apache, IIS, System), Event Correlation, IoCs Detection, Traffic Investigation, and Intrusion Detection. Mobile Forensics: Data Extraction, Call Logs, Messages, Emails, Media, App Data, and Social Media Analysis, Advanced PCAP Techniques, IoC Detection, Social Media Forensics, Mobile Data Integrity.

Text Books:

1. B. Nelson *et al*., *Guide to Computer Forensics and Investigations*, Sixth Edition, 2020.
2. Nour Moustafa, Digital Forensics in the Era of Artificial Intelligence, Taylor & Francis; 1st edition, 2024
3. J. Sammons, *The Basics of Digital Forensics: The Primer for Getting Started in Digital Forensics*, Elsevier, 2014.
4. A. M. Marshall, *Digital Forensics: Digital Evidence in Criminal Investigation*, John – Wiley and Sons, 2008.
5. N. Reddy, *Practical Cyber Forensics: An Incident-Based Approach to Forensic Investigations*, New York, Apress, 1st Edition, 2019.
6. L. E. Daniel and P. R. Johnson, *Digital Forensics for Legal Professionals: Understanding Digital Evidence from the Warrant to the Courtroom*, Syngress, 2012.

References:

1. T. J. Holt *et al*., *Cybercrime and Digital Forensics: An Introduction*, Routledge, 2nd Edition, 2017.
2. S. Widup and J. Sammons, *Computer Forensics and Digital Investigation with EnCase Forensic*, Syngress, 2014.
3. M. H. Ligh*et al.*, *The Art of Memory Forensics: Detecting Malware and Threats in Windows, Linux, and Mac Memory*, Wiley, 2014.
4. EC-Council, *Computer Forensics: Investigating Network Intrusions and Cyber Crime*, EC Council Press Series: Computer Forensics, 2016.

DIGITAL IMAGE AND VIDEO PROCESSING

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5020133	**Digital Image and Video Processing**	**2-1-0-1**	**2026**

Prerequisites: Nil

Course Objectives:

1. To provide students with a good understanding of the concepts of image and video processing tasks.
2. To help the students develop the ability to solve problems using the learned concepts.
3. Connect the concepts to other domains, such as machine learning and pattern recognition, within and without image and video processing.

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

CO1: Assess the effectiveness of various image enhancement, restoration, and segmentation techniques to support informed decision-making in specific applications.

CO2: Design and formulate innovative solutions using morphological image processing techniques for tasks like object detection and region-based segmentation.

CO3: Design and formulate innovative solutions using morphological image processing techniques for tasks like object detection and region-based segmentation.

CO4: Invent and build systems that integrate advanced image and video processing techniques with emerging technologies like machine learning for applications such as object recognition and pattern analysis.

CO5: Derive and compose solutions by connecting digital image and video processing with interdisciplinary fields, including human visual perception and computational modeling, to address complex real-world problems.

Program Learning Outcomes

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with program learning outcomes:

	PLO1	PL02	PL03	PL04	PLO5	PL06
C01	3	2	3	2	2	1
CO2	3	3	3	2	2	2
CO3	3	3	2	2	2	2
CO4	2	2	2	2	2	2
CO5	2	2	2	2	2	2

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content

1	Introduction to Image Processing Systems, Image Acquisition, Sampling and Quantization, Pixel Relationships, Color Fundamentals and Modules, File Formats, Image Enhancement and Restoration, Spatial Domain Gray Level Transformations, Histogram Processing, Spatial Filtering, Smoothing and Sharpening.
2	Frequency Domain: Filtering in Frequency Domain, DFT, FFT, DCT, Smoothing and Sharpening Filters, Homomorphic Filtering. Noise Models: Spatial and Frequency Properties of Noise, Important Noise Probability Density Functions, Periodic Noise, Estimation of Noise Parameters, Constrained and Unconstrained.
3	Restoration Models, Image Deblurring Algorithms. Morphological Image Processing: Erosion and Dilation, Opening and closing, Hit or miss transformation, basic morphological algorithms, gray scale morphology. Image Segmentation and Feature Analysis, Detection of Discontinuities, Edge Operators, Edge Linking and Boundary Detection, Thresholding, Region based Segmentation: Region Growing, Region Splitting and Merging. Representation and description: boundary and regional descriptors, Image Compression: classification of lossy and lossless image compression schemes.
4	Video Formation, Perception and Representation: Video Capture and Display, Analog Video Raster, Digital Video, Fourier Analysis of Video Signals and Frequency Response of the Human Visual System. Video Sampling: Basics of the Lattice Theory, Sampling of Video Signals Over Lattices, Filtering Operations in Cameras and Display Devices. Video Sampling Rate Conversion, Different Video Modeling. Video Object Tracking and segmentation. Object recognition, pattern and pattern classes, recognition based on decision- theoretic methods, structural methods, case studies –image analysis, image coding.

Text Books

1. R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Upper Saddle River, N.J: Prentice Hall, 2008.
2. A. K. Jain, *Fundamentals of Digital Image Processing*, USA: Prentice Hall, 1989.
3. J. W. Woods, *Multidimensional Signal, Image, and Video Processing and Coding*, 2nd ed. USA: Academic Press, 2011.
4. Y. Wang *et al.*, *Video Processing and Communications*, Signal Proc. Series, Prentice Hall, 2002.

References

1. W. K. Pratt, *Digital Image Processing: PIKS Scientific Inside*, USA: Wiley-Inter Science, 2007.
2. S. E. Umbaugh, *Digital Image Processing and Analysis: Human and Computer Vision Applications with CVIP Tools, 2nd ed. USA:* CRC Press, 2010.
3. A. M. Tekalp, *Digital Video Processing, 2nd ed.* USA: Prentice Hall Press, 2015.
4. A. C. Bovik, *Handbook of Image and Video Processing (Communications, Networking and Multimedia)*, USA: Academic Press, 2005.

ETHICAL HACKING AND PENETRATION TESTING

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M4010114/ M4020114/ M5050114	Ethical Hacking and Penetration Testing	2-1-0-1	2026

Prerequisites: Nil

Course Objectives:

1. To help the students apply tools and techniques to explore cyber security breaches.
2. To provide students with a knowledge of the need for protecting the cyber assets from an adversary.
3. To provide students with a knowledge of employing machine learning techniques for vulnerability assessment.

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

CO1: Apply the fundamental principles and legal aspects of ethical hacking and penetration testing to real-world scenarios.

CO2: Analyse various information security threats and vulnerabilities, and assess their potential impact using appropriate techniques.

CO3: Apply techniques such as password cracking, social engineering, and authentication mechanisms to enhance system security.

CO4: Evaluate and counter network-level attacks, web application vulnerabilities, and insider threats through practical methods.

CO5: Create effective security solutions and strategies to mitigate identified vulnerabilities and security risks in systems.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	2	1	2	2	1
CO2	3	3	1	2	2	2
CO3	3	3	1	2	2	3
CO4	2	3	2	2	3	3
CO5	3	3	2	2	3	3

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	**Ethical Hacking Fundamentals and Information Security Threats** Understanding Ethical Hacking: Principles, Importance, and Legal Aspects, Basics of Cybersecurity: Threats, Attacks, and Defense Mechanisms, Information Security Laws, Standards, and Regulatory Compliance, Footprinting and Reconnaissance: Gathering Information for Assessments, Network Scanning and Enumeration: Identifying Targets and Services, Vulnerability Assessment and Analysis: Identifying Weaknesses, Developing Comprehensive Vulnerability Assessment Reports
2	**Password Cracking and Social Engineering Techniques** Password Cracking Techniques: Brute Force Attack, Dictionary or Word List Attack and Rainbow Table Attack, Password Cracking Tools and Countermeasures; Strengthening Authentication: Multi-Factor Authentication (MFA), Social Engineering Concepts and Techniques, Countermeasures to Social Engineering and Identity Theft, Insider Threats and Countermeasures Hands-on Password Cracking and Social Engineering Simulations
3	**Network and Web Application Attacks** Network Level Attacks: DoS, DDoS, Session Hijacking, and Mitigation, Hacking Web Applications: Common Vulnerabilities and Attack Surfaces, OWASP Top 10: Understanding and Mitigating Web App Threats, Countermeasures to Web App Attacks: Security Best Practices, Network Intrusion Detection and Prevention Systems (IDS/IPS), Firewalls and Network Infra Devices: Concepts and Configurations, Practical Penetration Testing: Network and Web Application Targets
4	**Wireless, Mobile, and Cloud Security Assessment** Wireless Network Security: Threats, Attacks, and Mitigation, Hacking Wireless Networks: Techniques and Countermeasures, Mobile Device Security: Vulnerabilities and Exploits, Assessing Mobile Apps: Identifying Security Flaws, Cloud Computing Security: Risks, Benefits, and Best Practices, IoT and OT Security: Attacks and Countermeasures Hands-on Wireless Hacking, Mobile Exploitation, and Cloud Assessment

Text Books

1. M. Walker, Certified Ethical Hacker All-in-One Exam Guide, 4th ed., McGraw-Hill Education, 2020.
2. J. Erickson, Hacking: The Art of Exploitation, 2nd ed., No Starch Press, 2021.
3. W. Stallings, Network Security Essentials: Applications and Standards, 7th ed., Pearson, 2021.
4. P. L. Wylie, The Pentester Blue Print, Wiley Publication, 2021.

References

1. P. Kim, The Hacker Playbook 2: Practical Guide to Penetration Testing, Createspace Independent Publishing, 2015
2. M. T. Simpson, Hands-On Ethical Hacking and Network Defense, 2nd ed., Cengage Learning, 2012.
3. M. Meucci and A. Muller, Owasp Testing Guide v. 4.0, Open Web Application Security Project, 2014.
4. D. Kennedy et al., Metasploit: The Penetration Tester's Guide, 4th ed., No Starch Press, 2018.

FEDERATED LEARNING

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5010130 / M5020130	**Federated Learning**	**2-1-0-1**	**2026**

Prerequisites: The prerequisite knowledge for this course includes machine learning, basic computer systems and basic programming skills.

Course Objectives:

1. To know the basics of Federated Learning
2. To explore the components of Federated Learning systems.
3. Learn to customize, tune and orchestrate them for better model training.
4. To learn the applications of Federated Learning.

Course Outcomes: After completion of this course, the students would be able to:

CO1: Evaluate the fundamental concepts, architecture, and categories of federated learning by comparing them with traditional AI and big data approaches.

CO2: Design and implement federated learning algorithms using state-of-the-art frameworks such as TensorFlow Federated, PySyft, and Flower to address diverse problem domains.

CO3: Critically analyze privacy-preserving techniques like differential privacy, secure multi-party computation, and homomorphic encryption for federated learning systems.

CO4: Develop simulated federated learning scenarios to compare FL and non-FL approaches and explore their applications in domains like healthcare, finance, and edge computing.

CO5: Create real-world federated learning projects, including workflow design, deployment, and presentation of findings in professional formats.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	2	2	1	1	1	1
CO2	2	2	3	1	1	2
CO3	2	3	2	1	1	1
CO4	2	3	2	1	1	2
CO5	2	3	2	1	1	1

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus:

Module	Content
1	**Introduction:** Motivation – Challenges in Bigdata and Traditional AI - Federated Learning (FL) as a Solution – Comparison between Traditional Big data ML and FL systems. FL: Definitions – Process – Architecture - Categories – Current Developments – Research Issues – Open-Source Projects – The Federated AI Eco system - Security for FL Systems – Decentralized FL and Blockchain – Overall architecture of FL Systems.
2	**Federated Learning:** Work Flow and Key Elements - Horizontal FL – Definition – Architecture – Federated Averaging Algorithm; Vertical FL – Definition – Architecture – VFL Algorithms; Federated Transfer Learning (FTL) – Heterogeneous FL – Privacy Preserving Techniques: Secure Multi-Party Computation – Homomorphic Encryption – Differential Privacy.
3	**Federated Datasets and Software Frameworks** PySyft, TensorFlow Federated, FATE, Paddle FL, Flower, Xaynet, IBM FL, Substra, OpenFL, FedML, FedJax, Backdoors 101, FedLab, SimFL, easyFL, TorchFL. Methodology for Designing a Simulated Federated Learning Scenario - Comparative Study of FL and Non-FL Scenarios.
4	**Case Studies with Use Cases of FL Applications** Federated Computer Vision (CV) – Federated Natural Language Processing (NLP) – Federated Recommendation Systems. Applying FL - Healthcare Sector – Financial Sector – FL meets Edge Computing – Internet of Intelligence – Distributed Learning for Bigdata.

	Future Trends and Developments in Federated Learning – Looking at Future AI Trends.

Text Books and References

1. Qiang Yang, Yang Liu, Yong Cheng, Yan Kang, Tianjian Chen, Han Yu, Federated Learning, Synthesis Lectures on Artificial Intelligence and Machine Learning, Lecture 43, Springer Series, 2020.
2. K. N., Jeno, G. Federated Learning with Python: Design and Implement a Federated Learning System and Develop Applications Using Existing Frameworks. Germany: Packt Publishing, 2022.
3. Heiko Ludwig, Nathalie Baracaldo, Federated Learning: A Comprehensive Overview of Methods and Applications, Springer International Publishing, 2022.

HARDWARE SECURITY

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M4010137	**Hardware Security**	**2-1-0-1**	**2026**

Prerequisites: Prior knowledge of computer networks, cryptography, sensor networks and basics of computer hardware.

Course Objectives:

1. Provide knowledge of state-of-the-art security methods and devices.
2. Familiarize the range of hardware-level attack techniques and countermeasures.
3. Make students aware of potential hardware vulnerabilities and provide them with the knowledge and skills to build trustworthy hardware.

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

CO1: Analyse the vulnerabilities in the current digital system design flow and evaluate the physical attacks on these systems.

CO2: Apply knowledge to evaluate hardware security issues and understand their implications.

CO3: Create secure and trusted hardware solutions using the relevant tools and techniques.

CO4: Analyse recent trends in hardware security and apply this knowledge in research and development.

CO5: Evaluate the effectiveness of various hardware security methods in addressing modern challenges.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	2	3	1	2	2	1
CO2	2	3	1	2	2	2
CO3	3	3	2	2	3	3
CO4	2	3	3	2	2	3
CO5	3	3	1	2	3	3

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	Hardware Security threats, Vulnerabilities, and Attacks. Challenges in Securing Hardware, Threats to Hardware. Hardware Security Vulnerability Assessment. Hardware-Assisted Computer Security: ARM Trust Zone, Intel SGX. Hardware Root of Trust, Trusted Platform Module (TPMs), TPM Cryptographic Hardware, Hardware Accelerators, Cryptographic Coprocessors. Implementing Security in Reprogrammemable Hardware. FPGA Basics, Applications and Uses, FPGA Based Security Solutions.
2	Modern IC Design and Manufacturing Practices and Their Implications: Hardware Intellectual Property (IP) Piracy and IC Piracy, Design Techniques to Prevent IP and IC Piracy, Physically Unclonable Functions (PUFs), PUF Implementations and using PUFs to prevent Hardware Piracy, Model Building Attacks on PUFs (Case Study: SVM Modeling of Arbiter PUFs, Genetic Programming based Modeling of Ring Oscillator PUF). JTAG Protection**.**
3	Side-channel Attacks (SCA) on Cryptographic Hardware: Current-measurement based Side-channel Attacks, power, electromagnetic SCA. Design Techniques to Prevent Side-channel Attacks, Improved Side-channel Attack Algorithms and Cache Attacks. Fault-tolerance of Cryptographic Hardware, Fault Attacks. Hardware Trojan based SCA.

4	Hardware Trojans: Hardware Trojan Nomenclature and Operating Modes, Countermeasures-Design and Manufacturing Techniques to Prevent/Detect Hardware Trojans, Logic Testing and Side-channel Analysis based Techniques for Trojan Detection. Case study: Hardware security issues and solutions in vehicles, hardware security of fog end-devices for the internet of things.

Text Books

1. D. Mukhopadhyay and R. S. Chakraborty, *Hardware Security: Design, Threats, and Safeguards*, Chapman and Hall/CRC, 2014.
2. Y. Jin, Introduction to Hardware Security, Electronics, MDPI, 2015.
3. S. Sidhu et al., Hardware Security in IoT Devices with Emphasis on Hardware Trojans, Journal of Sensor and Actuator Networks, 2019.
4. I. Butun et al., Hardware Security of Fog End-Devices for the Internet of Things, Sensors, 2020.
5. P. Prinetto and G. Roascio, Hardware Security, Vulnerabilities, and Attacks: A Comprehensive Taxonomy, ITASEC, 2020.

INFORMATION SECURITY MANAGEMENT SYSTEM

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5010122/ M5020122	**Information Security Management Systems**	**2-1-0-1**	**2026**

Prerequisites: Nil

Course Objectives:

1. To impart an in-depth understanding of information security management systems.
2. To prepare students for managing all the aspects of security of any large organization.

Course Outcomes: After completion of this course, the students would be able to:

CO1: Analyse the security requirements of an organization to identify vulnerabilities and manage the overall security strategy.

CO2: Evaluate risk management frameworks and create a comprehensive risk treatment plan to address organizational security needs.

CO3: Create security policies, procedures, and guidelines to support the organization's risk management and compliance efforts.

CO4: Analyse security systems and evaluate their effectiveness through regular audits to ensure compliance with industry standards.

CO5: Evaluate and create effective security response strategies, ensuring they align with organizational goals and regulatory requirements.

PLO 1 Develop strong fundamental disciplinary knowledge.
PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.
PLO 3 Apply for a scholarship to conduct independent and innovative research.
PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.
PLO 5 Practice ethical standards of professional conduct and research.
PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	3	2	2	3	3
CO2	3	3	2	2	3	3
CO3	3	3	2	2	3	3
CO4	3	3	2	2	3	3
CO5	3	3	2	2	3	3

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	Introduction to Information Security: Fundamentals of Information Security and Risk Management, Introduction to ISMS, Introduction to Information Security Standards
2	Information Security Management Systems: Identification of Information Security Requirements, Application of Risk Assessment Techniques, Risk Treatment and Security Control Identification, Statement of Applicability. Practical: Assessing the risk for an IT organization, Preparation of the risk treatment plan.
3	Information Security Policies: Selection of Protective Measures, Preparation of Documented Information (ISMS Manual, Information Security Policies, Information Security Procedures, Information Security Guidelines, Forms and Records) Practical: Preparation of the security policy and guidelines for an IT organization.

4	Implementation Techniques and Measuring Effectiveness: Asset Management, Information Security Incident Management, Business Continuity Management, Measuring Effectiveness of ISMS, Internal Audit and Compliance Checking. Practical: Preparation of security audit report for an IT organization.

Text Books

1. A. Calder and S. Watkins, *ISO 27001: 2013 - A Pocket Guide*, IT Governance Publishing, 2017.
2. D. Alexander and A.Finch, *Information Security Management Principles*, BCS, The Chartered Institute for IT, 2020.
3. W. Siler, Information Security Management Systems: A Novel Framework and Software as a Tool for Compliance with Information Security Standard, CRC Press, 2013.
4. A. Nair, G. M. R., Mastering Information Security Compliance Management, Packt Publication, 2023.
5. K. C. Laudon and J. P. Laudon, Management Information System, Pearson Education, 2022.
6. S. Nadkarni, Fundamentals of Information Security, BPB Publications, 2020.

References

1. H. F. Tipton and M. Krause, *Information Security Management Handbook*, Auerbach Publications, 2019.
2. P. H. Gregory, *CISM Certified Information Security Manager All-in-One Exam Guide*, McGraw-Hill Education, 2018.
3. A. Kohnke and D. Shoemaker, *The Complete Guide to Cybersecurity Risks and Controls*, Apress, 2017.
4. D. Kosutic, *ISO 27001 -Risk Management in Plain English*, Advisera, 2015.
5. *ISO, ISO/IEC 27001:2022 - Information Security Management System - Requirements, ISO*, 2013.

INTRODUCTION TO CYBER SECURITY

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M4020101	**Introduction to Cyber** Security	**2-1-0-1**	**2026**
Prerequisites: Nil			

Course Objectives:

1. Establish foundational cybersecurity knowledge across digital systems.
2. Introduce threat models, vulnerabilities, and protective mechanisms.
3. Build early hands-on familiarity with security tools and protocols.
4. Embed awareness of future-oriented topics like post-quantum cryptography and AI threats.
5. Prepare students for industry-aligned roles and advanced specialization in later semesters.

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

CO1: Articulate key principles of cybersecurity, threat models, and security goals.

CO2: Classify common cyberattacks and evaluate their impact in real-world contexts.

CO3: Apply core security techniques (encryption, access control, backups) to simulated environments.

CO4: Analyze privacy and ethical considerations, including emerging Indian cyber laws.

CO5: Use baseline diagnostic and security tools to audit and secure digital systems.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	2	3	2	1	1
CO2	3	3	3	2	3	1
CO3	2	3	3	3	2	1
CO4	1	2	2	3	3	2
CO5	1	2	3	3	2	3

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	**Foundations of Cybersecurity and Threat Landscape** CIA triad and core security principles, Threats, vulnerabilities, attack surfaces, and risk Cyber adversaries: cybercrime, hacktivists, insiders, APTs Security in the Indian context: IT Act, CERT-IN, DPDP Act 2023 Post-Quantum Context: Why quantum computers threaten today's cryptography (RSA/ECC) Case study: Data breaches (Equifax), ransomware (WannaCry), Aadhaar implications

2	**Systems, Network, and Web Security** Operating System Security: file permissions, privilege management Network fundamentals: TCP/IP, DNS, ports, firewalls Common attacks: sniffing, spoofing, scanning, MITM Web vulnerabilities: OWASP Top 10, XSS, SQLi Hands-on tools: nmap, netstat, iptables, browser security settings Zero Trust Principle: Perimeter vs. identity-based defense
3	**Cryptography and Secure Communication** Foundations of cryptography: classical to modern Symmetric (AES) and Asymmetric (RSA, ECC) encryption Hashing, digital signatures, PKI, TLS/SSL Hands-on Labs: GPG, encrypting messages, verifying signatures Intro to PQC: quantum threat to RSA/ECC, NIST PQC, Kyber/Dilithium India's PQC readiness and hybrid crypto adoption
4	**Cyber Hygiene, Privacy, and Emerging Security Domains** Email and phishing security, password managers, 2FA Mobile & IoT Security: app permissions, insecure defaults, update hygiene Cloud security basics: shared responsibility model, IAM AI in Security: role of ML in threat detection, adversarial attacks Blockchain and Security: cryptographic primitives, tamper-proof logs Career pathways: Blue team vs Red team, certifications, CTFs

Lab Topics:

- OS-level permissions & hardening (Linux)
- Basic packet sniffing and port scanning
- Password audit and manager setup
- GPG encryption & digital signatures
- Simulated phishing and secure browsing exercise

Text Books

1. William Stallings, Network Security Essentials, 6th Ed., Pearson, 2021.
2. Charles J. Brooks et al., Cybersecurity Essentials, 2nd Ed., Pearson, 2020.
3. Mark Ciampa, Security+ Guide to Network Security Fundamentals, Cengage.
4. Daniel Bernstein et al., Post-Quantum Cryptography, Springer (selected chapters).

IOT NETWORKS AND ENDPOINT SECURITY

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5010138/M5020138	**IoT Networks and Endpoint Security**	**2-1-0-1**	**2026**

Prerequisites: Prior knowledge of distributed systems, computer networks, cryptography, sensor networks and basics of connected systems.

Course Objectives:

1. To impart a comprehensive and in-depth understanding of network security, IoT Networks, endpoint security, and various security mechanisms.
2. To expose the students to frontier areas of IoT security while providing sufficient foundations for further study and research.

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

CO1: Analyze the vulnerabilities in traditional and modern networks, including IoT networks, and evaluate the effectiveness of various security mechanisms in defending critical infrastructure.

CO2: Evaluate IoT architecture, communication protocols, and security challenges, applying advanced techniques to predict and mitigate threats to IoT endpoints.

CO3: Create and implement security frameworks for IoT networks, leveraging lightweight cryptography, key management, and privacy-enhancing techniques to address complex challenges.

CO4: Evaluate real-world case studies of IoT network security (e.g., IoV, UAV networks, Industrial IoT), identifying research opportunities and advancing solutions for future security needs.

CO5: Design and simulate secure IoT environments using advanced tools and frameworks, assessing the impact of simulated attacks on network performance and security mechanisms.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	2	1	3	2	
CO2	3	2	2	2	2	1
CO3	2	2	2	2	2	1
CO4	3	1	2	1	2	1
CO5	2	2	2	2	2	1

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content

1	Overview of TCP/IP, TCP/IP networks, Network Vulnerabilities, Zero-day vulnerabilities, Malwares, Threat and Risk Assessment, Network Vulnerability Assessment, Vulnerability Naming Schemes, Information Infrastructure Defense, Reverse Engineering and Code Obfuscation. Network Access Control. Firewalls. DMZ Network. Router Security. Enterprise Wireless Network Security Protocols. Security in 5G and 6G. Endpoint Devices, Security of Endpoint Devices, Endpoint Device Security Challenges. Case Studies: Cyber Attacks on Critical Infrastructure.
2	IoT Architecture, Resource Management, Interoperability in IoT, IoT Communication Protocols, Network and Transport Layer Challenges, IoT Threats and Security Challenges, Attacks on Different Layers and Categorization of IoT Attacks, IoT Gateway Security, IoT Routing Attacks, Secure Data Aggregation Mechanisms, *Security Analytics and Threat Prediction.* IoT Endpoint Devices, Threats to IoT Endpoints, General Attacks on IoT Endpoint Devices, IoT Endpoint Security Mechanisms, Security of AIOT Devices. Endpoint Security Best Practices. Case Studies.
3	Security Frameworks for IoT networks, Intrusion Detection and Prevention, Lightweight Cryptography, Key Management and Authentication, Privacy Enhancing and Anonymization Techniques, Trust and Identity Management, Access Control, IoT Simulators to simulate IoT Networks and Attacks on IoT networks, IoT Operating Systems and Security, IoT Forensics. IoT Security Standards.
4	Case Studies: Internet of Vehicles (IoV), Unmanned Aerial Vehicle (UAV) Networks, Industrial IoT Networks. Future Research Direction/Opportunity in the IoT Networks and Endpoint Security.

Text Books

1. C. H. Gebotys, *Security in Embedded Devices*, Springer, 2010.
2. C. H. John, Wu, and J. David Irwin, *Introduction to Computer Networks and Cybersecurity*, CRC Press, 2013.
3. E. A. Lee and S. A. Seshia, *Introduction to Embedded Systems, A Cyber-Physical Systems Approach*, 2nd ed., MIT Press, 2017.
4. F. Hu, *Security and Privacy in Internet of Things (IoTs): Models, Algorithms, and Implementations*, CRC Press, 2020.
5. K. Namuduri*et al.*, *UAV Networks and Communications*, Cambridge University Press, 2017.
6. N. Gupta *et al.*, *Internet of Vehicles and its Applications in Autonomous Driving*, Springer, 2021.
7. R. Buyya and A. V. Dastjerdi, *Internet of Things: Principles and Paradigms*, Elsevier, 2016.
8. R. Buyya and S. N. Srirama, *Fog and Edge Computing: Principles and Paradigms*, Wiley, 2019.
9. W. Stallings, *Cryptography and Network Security: Principles and Practice*, Pearson education, 2013.
10. Z. Mahmood, *Connected Vehicles in the Internet of Things: Concepts, Technologies and Frameworks for the IoV*, Springer, 2020.

MALWARE ANALYSIS AND REVERSE ENGINEERING

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5010117/ M5020117/ M5050117	**Malware Analysis and Reverse Engineering**	**2-1-0-1**	**2026**

Prerequisites: Nil

Course Objectives:

1. To provide students with a knowledge of various malware types and families.
2. To help the students apply tools and techniques to detect malware.
3. To provide the students with an understanding of the need for protecting computer systems against malware attacks.

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

CO1: Analyse the fundamentals of malware analysis, including various types of malware and their families across different operating systems.

CO2: Evaluate proficiency in static analysis and reverse engineering techniques for detecting and analysing obfuscated and packed malware.

CO3: Analyse malware behavior and evasion techniques through dynamic analysis to understand their operational impact.

CO4: Explore and evaluate advanced topics such as IoT malware analysis and using machine learning and deep learning for automated malware detection.

CO5: Create an awareness of adversarial evasion techniques and their impact on malware detection mechanisms.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	3	1	2	2	1
CO2	3	3	2	2	3	3
CO3	3	3	2	2	3	3
CO4	3	3	3	2	3	3
CO5	3	3	2	2	3	3

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	**Introduction to Malware and Operating Systems** Understanding Android Malware: Source Code, Security Assessment Tools; Types and Families of Android Malware, Reverse Engineering Android Applications, Windows Operating System and Malware Types, Reverse Engineering Windows Applications, Security Assessment Tools for Windows, Types of Linux and IoT Malware and Families, Linux Operating System Overview, Reverse Engineering Linux OS and IoT Firmware, Security Assessment Tools for Linux and IoT.
2	**Static Analysis and Reverse Engineering** Static Analysis of Android Malware, Detection of Obfuscated and Packed Android Malware, Dalvik Opcode Analysis, Static Analysis Tools for Android Malware, Static Analysis of Windows Malware, Reverse Engineering Windows Malware, Detection of Obfuscated and Packed Windows Malware, Static Analysis Tools for Windows Malware, Static Analysis of Linux and IoT Malware, Reverse Engineering Linux and IoT Malware, Detection of Obfuscated and Packed Linux and IoT Malware, Static Analysis Tools for Linux and IoT Malware, IoT Implant Toolkit for Malware Implantation.
3	**Dynamic Analysis and Evading Malware** Dynamic Analysis of Android Malware, Investigating Android Malware Obfuscation, Dynamic Analysis Tools for Android Malware, Android Malware Evasion and Current Trends, Dynamic Analysis of Windows Malware, Process Monitoring for Dynamic Analysis of Windows Malware, Windows Registry Monitoring, Investigating Windows Malware Obfuscation, Dynamic Analysis Tools for Windows Malware, Dynamic Analysis of Linux and IoT Malware, Examining Memory Snapshots for Linux Malware, Investigating Security of Linux Kernel Against Malware Attacks, Detecting IoT Malware Using Network Traffic Analysis.
4	**Machine Learning and Deep Learning in Malware Detection** Machine Learning for Malware Detection: Static and Dynamic Features, Deep Learning for Automated Malware Analysis, Introduction to Adversarial Malware Evasion, Adversarial Evasion in Various OS Malware Detection Mechanisms.

Text Books

1. A. Kleymenov and A. Thabet, Mastering Malware Analysis: The Complete Malware Analyst's Guide to Combating Malicious Software, APT, Cybercrime, and IoT Attacks, Packt Publication, 2019.
2. 2. K. A. Monappa, Learning Malware Analysis: Explore the Concepts, Tools, and Techniques to Analyze and Investigate Windows Malware, Packt Publication, 2018.
3. A. D. Joseph et al., Adversarial Machine Learning, Cambridge University Press, 2019.
4. T. Thomas et al., Machine Learning Approaches in Cybersecurity Analytics, Springer, 2020.
5. K. Dunham, Android Malware and Analysis, 1st ed., Auerbach Publications, 2014.
6. M. Sikorski and A. Honig, *Practical Malware Analysis: The Hands-On Guide to Dissecting Malicious Software*, 1st ed., No Starch Press, 2012.

References

1. M. H. Ligh et al., *The Art of Memory Forensics: Detecting Malware and Threats in Windows, Linux, and Mac Memory*, 1st ed., Wiley, 2014.
2. C. Chio and D. Freeman, *Machine Learning and Security*, O'Reilly, 2018.

MATHEMATICAL FOUNDATIONS FOR ARTIFICIAL INTELLIGENCE

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5050103	**Mathematical Foundations for Artificial Intelligence**	**1-1-0-0**	**2026**

Prerequisites: Nil

Course Objectives:

1. Build a solid foundation in probability, linear algebra, and optimization as core mathematical tools for designing and analyzing AI algorithms.
2. Apply mathematical concepts and techniques to formulate models, perform rigorous analysis, and solve real-world problems in AI and related domains.

Bridge theory and practice through structured problem-solving and hands-on exercises, enabling readiness for advanced research and industry-level AI challenges.

Course Outcomes: After completion of this course, the students would be able to:

CO1: Apply concepts of vector spaces, eigenvalues, inner product spaces, probability theory, and optimization techniques to formulate and solve computational problems in AI and related domains.

CO2: Evaluate probabilistic models and optimization strategies, including convexity conditions and gradient-based methods, to analyze the performance and stability of mathematical solutions.

CO3: Create mathematical and computational models using linear algebra, probability distributions, and unconstrained optimization methods to address real-world machine learning and data-driven challenges.

CO4: Apply constrained and unconstrained optimization principles, including necessary and sufficient optimality conditions, to design efficient algorithms under equality and inequality constraints.

CO5: Evaluate and create integrated solutions by combining linear algebraic structures, probabilistic reasoning, and optimization frameworks to solve advanced problems in computer science and artificial intelligence.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge

PLO 2 Demonstrate research skills that are of experimental, computational, or theoretical nature

PLO 3 Apply scholarship to conduct independent and innovative research

PLO 4 Show communication skills in a variety of formats (oral, written) and to expert and non-expert audiences;

PLO 5 Practice ethical standards of professional conduct and research;

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and write articles for scholarly journals if it is taught by faculty in the department.

Mapping of course outcomes with programme learning outcomes:

	PLO 1	PLO 2	PLO 3	PLO 4	PLO 5	PLO 6
CO1	3	2	2	1	1	1
CO2	3	2	2	1	1	2
CO3	2	3	2	2	1	1
CO4	3	2	2	2	1	2
CO5	3	3	3	3	2	2

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus:

Module	Content
1	Vector spaces, Bases, Dimension, Rank and Nullity – Row and Column space of a matrix. Eigenvalues and Eigenvectors. Inner product spaces. Normed linear spaces.

2	Introduction to Probability Theory – Notion and Axioms of probability – Equally likely events – Conditional probability – Independent events. Bayes' theorem, Random Variables. Probability Distributions.
3	Optimization – Sequences and limits, derivative matrix, gradients. Unconstrained optimization – Necessary and sufficient conditions for optima, convex sets, convex functions, optima of convex functions, steepest descent, Newton and quasi-Newton methods, conjugate direction methods.
4	Constrained optimization – Linear and non-linear constraints, equality and inequality constraints, optimality conditions.

Text Books

1. Hsu HP, *Theory and Problems of Probability, Random Variables, and Random Processes*. New York: McGraw-Hill; May 2014.
2. S. Lipschutz, *Schaum's Outline of Theory and Problems of Linear Algebra*, New York: McGraw-Hill, 1968.
3. E. K. P. Chong and S. H. Zak, *An Introduction to Optimisation*, 2nd ed. India: Wiley, 2010.

References

1. S. M. Ross, *Introduction to Probability Models*, 11th ed., Academic Press, 2014.
2. G. Strang, *Linear Algebra and its Applications*, 4th ed. India: Cengage Learning, 2005.
3. S. Sra, S. Nowozin, and S. J. Wright, *Optimization for Machine Learning*, MIT Press, 2012.
4. D. Bertsimas and J. Tsitsiklis, *Introduction to Linear Optimization*, Athena Scientific, 1998.

MATHEMATICAL FOUNDATIONS FOR COMPUTER SCIENCE

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M4010103/ M4020103	**Mathematical Foundations for Computer Science**	**2-1-0-1**	**2026**
Prerequisites: Nil			

Course Objectives:

• Develop a strong foundational understanding of discrete mathematics, probability, linear algebra, and graph theory to address complex computational problems.

• Apply mathematical principles and techniques to model, analyze, and solve real-world problems in computer science and related fields.

• Integrate theoretical concepts with practical applications through problem-solving and hands-on exercises to prepare for advanced research and industry challenges.

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

CO1: Apply mathematical principles from discrete mathematics, probability, linear algebra, and graph theory to solve real-world computational problems.

CO2: Construct mathematical models for analyzing complex problems in computer science, such as network optimization, machine learning, and cryptographic systems.

CO3: Illustrate key concepts in probability, linear algebra, and graph theory through practical experimentation and computational techniques.

CO4: Analyze algorithms and their performance using graph theory, combinatorics, and probability-based approaches.

CO5: Develop innovative solutions for computer science challenges by integrating concepts from different areas of mathematics and articulating their implications.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	2	2	1	1	1
CO2	3	3	2	2	1	2
CO3	2	3	2	2	1	1
CO4	3	2	2	2	1	2
CO5	3	3	3	3	2	2

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	Propositional logic, predicate logic, set theory, relations and functions, combinatorics, recurrence relations, modular arithmetic Practicum: Symbolic computation, solving recurrence relations, applications in cryptography and algorithm design
2	Probability axioms, conditional probability, Bayes' theorem, random variables, probability distributions, expectation, variance, hypothesis testing, entropy, and mutual information. Practicum: Simulating random variables, data analysis, applications of entropy
3	Matrix operations, vector spaces, eigenvalues and eigenvectors, diagonalization, singular value decomposition (SVD), orthogonality Practicum: Matrix manipulations, applications of eigenvalues and eigenvectors applications of SVD, vector space modeling.
4	Graphs, Graph representations, special types of graphs, connectivity, isomorphism of graphs, graph centrality values, Euler and Hamilton paths Practicum: Graph implementation and modeling with graphs, analyzing algorithms and their performance using graphs, applications of graph centrality values

Text Books

1. Kenneth H. Rosen, Discrete Mathematics and Its Applications, McGraw Hill; 8th edition, 2021
2. Neal Koblitz,. A Course in Number Theory and Cryptography, Springer Verlag (low price edition), 2nd Edition, 1994
3. Kenneth Hoffman, Ray Kunze, Linear Algebra, Prentice-Hall of India Pvt.Ltd.
4. Hsu HP. Theory and problems of probability, random variables, and random processes. New York: McGraw-Hill; May 2014.
5. M. Mignotte, Mathematics for computer algebra, Springer-Verlag, 1992.
6. SuvritSra, Sebastian Nowozin, and Stephen J. Wright. Optimization for Machine Learning. The MIT Press, 2011.

References

1. Bertsimas, D. &Tsitsiklis, J., Introduction to linear optimization, Athena Scientific, 1997.
2. An Introduction to Optimization- E. Chong, S. Zak, Wiley, 2013.
3. Hastie, T.; Tibshirani, R. & Friedman, J., The Elements of Statistical Learning, Springer New York Inc., New York, NY, USA, 2001.
4. Donald F. Stanat and David F. McAllister, Discrete mathematics in Computer Science,Prentice-Hall, 1977.
5. Thomas Koshy, Elementary number theory with Applications, Elsevier, 2007.
6. G. Chartrand and P. Zhang, Introduction to Graph Theory, McGraw-Hill Companies, 2005.
7. Douglas B. West, Introduction to Graph Theory, Prentice Hall of India, 2001.

MOBILE APPLICATION SECURITY

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5010121/ M5020121/ M5050121	**Mobile Application Security**	**2-1-0-1**	**2026**

Prerequisites: Nil.

Course Objectives:

1. To impart a comprehensive and in-depth understanding of mobile application security, mobile OS security, and various security mechanisms.
2. To expose the students to frontier areas of mobile security while providing sufficient foundations for further study and research.

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

CO1: Analyse the fundamental concepts of mobile application security, the importance of securing smartphone devices, and the various types of mobile applications.

CO2: Evaluate the architecture and components of the Android OS, including activities, services, content providers, broadcast receivers, fragments, and intents.

CO3: Analyse Android security models, app sandboxing, permissions, and data encryption techniques to assess their effectiveness in protecting mobile applications.

CO4: Create secure Android applications using best practices, including app signing, secure communication, and root protection mechanisms.

CO5: Analyse and evaluate mobile application vulnerabilities through static and dynamic analysis, malware analysis, and runtime manipulation, and create mitigation techniques.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	3	1	2	2	1
CO2	3	3	1	2	2	2
CO3	3	3	2	2	3	3
CO4	2	3	2	2	3	3
CO5	3	3	2	2	3	3

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	**Introduction to Mobile Application Security and Android Basics** Importance of Smartphone Security, Types and Categories of Mobile Applications, History of Android and its Evolution, Features and Architecture of Android OS, Components of Android: Activity, Service, Content Provider, Broadcast Receiver, Fragment, Intent, Resources.

2	**Android Security and Development Environment** Android Security Models: App Sandboxing, App Signing, App Permissions; Data Encryption and Secure Coding Practices, Securing Android Devices: Best Practices and Configuration, Certificate/SSL Pinning for Secure Communication, Android Software Development Kit (SDK) Tools, Android Emulator and Debugging with Android Debug Bridge (adb), Using Android Studio for Application Development.
3	**Mobile Application Vulnerabilities and Analysis** Common Mobile Vulnerabilities and Avoidance Techniques, Identifying Vulnerable Features in Android Applications, Decompiling Android Applications: Smali Files and Java Code Recovery, Risk Analysis and Classification of Android Applications, Tools for Mobile Malware Analysis, Android Malware Analysis Approaches: Static, Dynamic, Network, Hybrid Analysis; Bypassing Root Detection and Certificate/SSL Pinning, Application Patching and Runtime Manipulation using Frida and Objection, Introduction to OWASP Top 10 Mobile Security Risks.
4	**iOS and Windows Phone Security** iOS Security Model and Architecture, Introduction to Jailbreaking and its Implications, Xcode and iOS Application Development Environment, File System and Device Interaction in iOS, Decompiling iOS Applications and Reverse Engineering, Intercepting Network Traffic for Analysis, Security Model of Windows Phone OS, Comparative Analysis of Mobile Security across Platforms.

Text Books

1. M. Swamynathan and J. Mannino, *Mobile Security and Privacy: A Hands-On Guide*, O'Reilly, 2019.
2. H. Dwivedi, *Mobile Application Security*, Packt Publishing, 2019.
3. Tim Speed *et al.*, *Mobile Security: How to Secure, Privatize, and Recover Your Devices* ,Apress, 2019.
4. V. K. Velu, *Mobile Application Penetration Testing*, Packt Publishing, 2020.
5. N. Elenkov, *Android Security Internals: An In-Depth Guide to Android's Security Architecture*, 1st ed., No Starch Press, 2014.
6. D. Thiel, *iOS Application Security: The Definitive Guide for Hackers and Developers*, 1st ed., O'Reilly, 2016.
7. N. Bergman *et al.*, *Hacking Exposed Mobile: Security Secrets and Solutions, 2nd ed.*, McGraw-Hill Education, 2020.

References

1. A. Hoog and K. Strzempka, *Android Forensics: Investigation, Analysis, and Mobile Security for Google Android*, 1st ed., Elsevier, 2011.
2. C. Miller *et al.*, *iOS Hacker's Handbook*, 1st ed., Wiley, 2012.
3. D. Chell*et al.*, *The Mobile Application Hacker's Handbook*, 1st ed., Wiley, 2015.

NATURAL LANGUAGE PROCESSING

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5010111/ M5020111	Natural Language Processing	2-1-0-1	2026

Prerequisites: Prior knowledge of Python, Probability, Statistics and Machine Learning

Course Objectives:

1. To create significance of Natural Language Processing and to process text and data
2. To infer and analyses the theory behind Language Modeling from an algorithmic point of view.
3. To create application levels of Natural Language Processing.
4. To gain Practical experience using NLP toolkits,

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

CO1: Analyse linguistic essentials and implement text preprocessing techniques for Natural Language Processing.

CO2: Apply various text representation techniques to extract features from text data.

CO3: Evaluate deep learning models, Large Language Models (LLMs), and Transformer-based architectures for solving complex NLP tasks, such as sentiment analysis and language generation.

CO4: Create practical NLP applications by integrating ethical considerations during the development process.

CO5: Apply NLP toolkits and State-of-the-Art models, including LLMs and Generative AI, to demonstrate case studies and real-world applications.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	3	3	1	1	1
CO2	3	3	3	1	1	1
CO3	3	3	3	1	1	1
CO4	3	3	3	1	1	2
CO5	2	2 3	3	2	3	2

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus	
Module	**Content**
1	Introduction to Natural Language Processing: Definition, Scope and Applications. Essential Linguistics: Syntax, Semantics and Pragmatics, Morphology, Corpus, Token, Lexicon, Stop Words, Multilingualism, Script Diversity. Classical NLP Paradigms and Methods: Semantic Analysis, Syntactic Analysis. Constituency Grammar, Dependency Grammar, Context Free Grammar, Regular Expressions, Finite-State Automata, Morphological Parsing and analysis, Dependency Parsing, Tokenization, Stemming, Lemmatization, Stop word removal, Parts-of-Speech (POS) Tagging, Named Entity Recognition, Hidden Markov Model (HMM) - Viterbi algorithm and Conditional Random Fields (CRF), Maximum Entropy models.
2	Language Modeling: Defining language models, distributional semantics, probabilistic n-gram language models, smoothing, interpolation, entropy, perplexity. Text Representation Techniques: Word Embedding - Word2Vec, GloVe, Fastext. Document Embedding- Bag of Words (BoW), Count Vectors, TF-IDF. Topic Modeling, LDA. Word Sense Disambiguation (WSD), Information Retrieval, Practice Sentiment Analysis, Text summarization, Text Classification, Question Answering, Topic Modeling- LDA.
3	NLP using Deep Learning: RNN, CNN, LSTM, Multilingual Seq2seq Deep Neural Network, Encode-decoder Model, Attention, Self Attention, Transformer Models - BERT, GPT, XLNet, GAN, Generative AI and LLM. Ethics and Bias in NLP.
4	Applications and Case Studies: Practice Machine Translation- Rule-Based Machine Translation (RBMT), Statistical Machine Translation (SMT), Hybrid Machine Translation, Neural Machine Translation (NMT), LLM based, Machine learning of cross-lingual mappings, learning representations using cross-lingual supervision, Challenges in using NLP with multilingual resources

Text Books

1. Daniel Jurafsky and James H. Martin, *Speech & Language Processing*, Pearson Education India, 2000.
2. S. Vajjala et al., *Practical Natural Language Processing: A Comprehensive Guideto Building Real-World NLP Systems*, O'Reilly, 2020.
3. Christpher Manning, HinrichSchutze, Foundations of Statistical Natural Language Processing, MIT press, Cambridge, 1999.
4. Li Deng, Yang Liu, Deep Learning in Natural Language Processing, Springer, 2015.

NLP and Agentic AI

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5050111	**NLP and Agentic AI**	**2-1-0-1**	**2026**

Prerequisites: Prior knowledge of Python, Probability, Statistics, AI and Machine Learning

Course Objectives:

1. To create significance of Natural Language Processing and to process text and data
2. To infer and analyses the theory behind Language Modeling from an algorithmic point of view.
3. To create application level using different Computational Models and Deep learning
4. To study Agentic approach in NLP for workflow by building framework for easy deployment and to gain Practical exposure using NLP toolkits.

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

CLO1: Understand the linguistic essentials, and text preprocessing techniques in Natural Language Processing.

CLO2: Apply various text representation techniques for extracting features from text data.

CLO3: Explore deep learning models, LLM and Transformer-based architectures to tackle complex NLP tasks, such as sentiment analysis and language generation.

CLO4: Develop AI orchestration using Agentic AI for framework and workflow defining for easy product deployment.

CLO5: Demonstrate case studies and applications using NLP toolkits and State-of-the-art models like LLM, GenAI Agentic AI etc.

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	3	3	1	1	1
CO2	3	3	3	1	1	1
CO3	3	3	3	1	1	1
CO4	3	1	1	2	2	2
CO5	2	2	3	2	3	2

Mapping of course outcomes with programme learning outcomes:

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	Introduction to Natural Language Processing: Definition, Scope and Applications. Essential Linguistics: Syntax, Semantics, Discourse and Pragmatics, Morphology, Corpus, syntactic and Semantic Analysis, Morphological Analysis, Parts of Speech (POS)Tagging, Context Free Grammar, Dependency parsing, Named Entity Recognition (NER), HMM, Viterbi Algorithm, Conditional Random Fields (CRF), Maximum Entropy Models.

2	Language Modeling: Defining language models, distributional semantics, probabilistic n-gram language models, entropy, perplexity. Text Representation Techniques: Word Embedding - Word2Vec, GloVe, Fastext. Document Embedding- Bag of Words (BoW), TF-IDF. Topic Modeling, LDA. Word Sense Disambiguation (WSD), Information Retrieval, RAG based, Text summarization, Text Classification, Question Answering, Topic Modeling- LDA.
3	Machine Translation – Five Waves of Machine Translation -RBMT, SMT, NMT, LLM based, NLP using Deep Learning: RNN, CNN, LSTM, Multilingual Seq2seq Deep Neural Network, Encode-decoder Model, Attention, Self-Attention, Transformer Models, - BERT, GPT, GAN, Generative AI and LLM. Cognitivism, Ethics and Bias in LLM and NLP
4	Agentic AI- Foundations of Agentic AI, Intelligent AIs and its behaviors, Multi Agent Systems-coordination, cooperative, communication and collaborations, Learning Agents, LangChain and LangGraph - Agentic Workflows using LangChain, Graph-Based Agent Design with LangGraph, Human-in-the-loop workflows. Future trends in Agentic and Multi-Agent AI, case studies.

Text Books

1. Christpher Manning, HinrichSchutze, Foundations of Statistical Natural Language Processing , MIT press, Cambridge, 1999
2. Li Deng, Yang Liu, Deep Learning in Natural Language Processing, Springer, 2015
3. Elizabeth Sherly, Leena Pillai, Kavya Manohar, John Mc Crae, Machine Translation: Best Practices using Deep Learning and Generative AI, CRC Press , Taylor and Francis, 2025 (preview copy)
4. Ellot R Stround, Agentic AI with Langchain&Langgraph: Build Reliable Multi-Agent LLM Systems with Python, RAG Workflows, and Production- November 2025

Reference Books

1. Vajjala et al., Practical Natural Language Processing: A Comprehensive Guide to Building Real-World NLP Systems, O'Reilly, 2020.
2. Machine Learning: Concepts, Techniques and Applications Paperback – 27 June 2025 by T. V. Geetha (Author), S. Sendhilkumar (Author), CRC Press, Taylor and Francis, Atlantic Publishers (India)

OBJECT ORIENTED SOFTWARE ENGINEERING

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5020125	**Object Oriented Software Engineering**	**2-1-0-1**	**2026**
Prerequisites: Nil			

Course Objectives:

1. To introduce the fundamental concepts of software engineering and various phases of Software development
2. To introduce various software process models and Object-Oriented Technology.
3. To build an understanding of various SE models, Object Oriented Designs, and Models.
4. To familiarize testing, Maintenance, and Deployment Models of Software Systems.

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

CO1: Evaluate suitable software development life cycle models to be used for a project.

CO2: Analyze a problem, identify and define the system requirements to solve the problem, and prepare the Software Requirements Specification.

CO3: Translate the Software Requirement Specification to a design using an appropriate software design methodology and prepare a Software Design Description, including Object Oriented Modeling.

CO4: Design software systems based on appropriate technology and programming language by adhering to coding standards, ensuring code quality, and managing resources economically.

CO5: Plan appropriate testing strategies for validating the developed software system.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	2	1	1	1	1
CO2	3	3	2	1	1	2
CO3	3	3	2	2	1	2
CO4	3	3	3	3	2	3
CO5	3	3	3	2	3	3

(Correlation: 1: Slight (Low) 2: Moderate (Medium)3: Substantial (High))

Syllabus

Module	Content

1	**Introduction to Software Engineering** History of Software and Software Engineering, Software Crisis and Retrospection, Software Engineering Layers, Software Process, A Generic Process Framework, Software Process Models – Waterfall Model, V-Model, Incremental Model, Spiral Model, Prototyping Model, Rational Unified Process, Iterative Models, Agile Software Development, Software Engineering Ethics.
2	**Requirement Analysis and Specification** Requirement Engineering processes: Requirement elicitation – Functional and non-functional requirements, Requirement Analysis, Object Oriented Modelling, Developing use cases and Use Case Models, Use case Analysis, Interaction Diagrams. Requirement Specification, IEEE Std 830-1998 Software Requirement Specification (SRS) Preparation, Requirement verification, Requirement Traceability Matrix, Requirement change control.
3	**Software Design** Design Principles and Concepts, Design methodologies – Structured System Analysis and Design or Function Oriented Design and Object-Oriented Analysis and Design Domain Model, Design Classes, subsystems and Packages, Software Architectural Styles and Design Patterns, Architectural Design-4+1 view Architecture, Data Model, IEEE Std 1016-2009 Software Design Description (SDD) Template. Case Study: Library Management System – Object Oriented Analysis and Design using UML.
4	**Coding, Testing and Deployment** Introduction to Coding, Selection of Technology/Programming Language, Programming Practices, Coding Standards, Code Verification - Code Review and Static Analysis, Size Measures, Complexity Analysis, Software Verification and Validation, Testing Fundamentals, Software Testing Strategies, Black Box and White Box Testing, Unit Testing, Integration Testing, System Testing, User Acceptance Testing, Testing Process and Test Documentation, Test Case Design Techniques for Black Box and White Box Testing, Software Maintenance. Deployment Diagram

Text Books

1. I. Sommerville, *Software Engineering*, 10th ed., Pearson Education, 2015.
2. R. S. Pressman, *Software Engineering: A Practitioner's Approach*, 8th ed., McGraw-Hill, 2014.
3. G. Booch*et al.*, *The United Modeling Language User Guide*, Addison-Wesley, 2005.
4. B. Bruegge and A. H. Dutoit, *Object-Oriented Software Engineering*, 2nd ed., Pearson Education, 2004.
5. A. Cockburn, *Agile Software Development*, 2nd ed., Pearson Education, 2007.

References

1. R. Mall, Fundamentals of Software Engineering, PHI Learning, 2014.
2. P. Jalote, An Integrated Approach to Software Engineering, 3rd ed., Narosa Publishing House, 2009.
3. I. Jacobson et al., The Unified Software Development Process, Pearson Education, 1999.
4. IEEE Std 830-1998 – IEEE Recommended Practice forSoftware Requirements Specifications.
5. IEEE Std 1016-2009 – IEEE Standard for Information Technology – Systems Design – Software Design Descriptions.

OOPS AND JAVA

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5020124	**OOPS and JAVA**	**2-1-0-1**	**2025**

Prerequisites: Basic programming concept.

Course Objectives:

1. To introduce object-oriented concepts through Java language.
2. To use object-oriented programming in building simple software tools.

Course Outcomes: After completion of this course, the students would be able to:

CO1: Analyse the key principles of object-oriented programming (OOP) and evaluate their application in real-world software development.

CO2: Apply Java programming language concepts to develop software solutions, demonstrating proficiency in coding and problem-solving.

CO3: Evaluate the use of multi-threading and network programming concepts in Java to solve concurrent programming challenges.

CO4: Create Java-based applications incorporating object-oriented principles, multi-threading, and network programming for improved system functionality.

CO5: Analyse the performance and efficiency of multi-threaded Java applications and evaluate optimizations to improve scalability and resource management.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.
PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.
PLO 3 Apply for a scholarship to conduct independent and innovative research.
PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.
PLO 5 Practice ethical standards of professional conduct and research.
PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	2	1	2	3	2
CO2	3	3	2	2	3	3
CO3	2	3	2	3	2	3
CO4	3	3	2	3	3	3
CO5	3	3	3	2	3	2

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	Object Oriented Paradigm and JAVA overview: Object oriented Concepts: Introduction to OOPS, Abstraction, Encapsulation, Objects and Classes, Constructors Inheritance, Polymorphism, Abstract Classes, Interfaces, Introduction to Java, JVM, Primitive data types, Control Statements, Methods, Classes Introduction to Java Compilers and Lab.

2	JAVA statements: selection statements, iteration statements, jump statements, Introduction to classes: Class fundamentals, declaring object reference variable, Introducing methods, constructors, the key word, garbage collection, the finalize (), method. Methods and Classes Overloading methods, using objects as parameters.
3	Java Arrays, Utilities and Packages: Java Arrays, Wrapper Classes, Java IO, Inheritance, Super class, Polymorphism, java Packages, class libraries, Interfaces, Exception Handling, JAVA Strings.
4	Multithreading and JAVA Networking: The Java thread model, the main thread, creating thread, creating multiple thread, using is alive () and join (). Thread priorities, synchronization, Inter thread communications, suspending resuming and stopping thread using multithreading Networking: Networking basics, Java and the Internet Address, TCP/IP client Sockets, URL, URL connection, TCP/IP server Sockets The Applet Class.

Text Books

1. P. Naughton and H. Schildt, The Complete Reference JAVA 2, McGraw-Hill, 1999.
2. C. T. Wu, Introduction to JAVA Programmeming, 2nd ed., John Wiley and Sons, 2000.
3. M. T. Somashekaraet al., Object Oriented Programmeming with JAVA, PHI Learning, 2017.

References

1. B. Eckel and C. Allison, Thinking in JAVA, 2nd ed., Prentice Hall, 2000.
2. C. Horstmann, Computing Concepts with JAVA 2 Essentials, 2nd ed. India: Wiley, 2006.
3. H. Schildt, Java: a Beginner Guide Essential Skills Made Easy, 4th ed., McGraw- Hill Professional, 2007.

OPERATING SYSTEMS

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M4020142	**Operating Systems**	**2-1-0-1**	**2026**
Prerequisites: Nil			

Course Objectives:

1. To help students understand the necessity and fundamental concepts of an Operating System.
2. To explore all the essential building blocks in an Operating System.
3. To build practical skills for developing application programming in an Operating System.
4. Explore the different types of Operating Systems in different domains and analyse the security aspects.

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

CO1: Analyze various concepts and building blocks associated with Operating Systems.

CO2: Apply the concepts, building blocks, principles, and best practices tothe software development.

CO3: Illustrate security aspects in the Operating System through its predefined features.

CO4: Design application programming with multi-processing concepts.

CO5: Analyze different types of Operating Systems available and develop applications.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	3			2	
CO2	2	3	3		3	3
CO3	3			3	3	
CO4	2	3	3		3	3
CO5	3	3	3		3	3

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	**Introduction**: Basic OS functions, evaluation of OS, different types of OS, computer system operation, I/O structure, system protection, OS services,

	Processor and user modes, kernels, system calls and system programmes. **Process Management:** Concept of processes, I/O and CPU bound process, process hierarchy, co-operating processes, inter-process communication. **Process scheduling:** Scheduling criteria, preemptive and non-preemptive scheduling, scheduling algorithms, multiprocessor scheduling. **Threads:** Overview, benefits of threads, user and kernel threads. **Process Synchronization:** Background, concurrent processes, critical section problem, classical problems of synchronization, semaphores.
2	**Deadlocks:** Characterization, detection, prevention, avoidance, recovery. **Memory Management:** Background, logical vs. physical address, swapping, paging, segmentation. **Virtual Memory:** Background, demand paging, page replacement algorithms, thrashing. **Disk Management:** Disk structure, disk scheduling, boot block and bad blocks. Characteristics of Embedded Systems, Embedded Linux, and Application specific OS. Basic services of NACH Operating System, Principles of protection, domain of protection, access matrix, access control, language-based protection, programme threats, system and network threats, user authentication, implementing security defenses, firewalling, exercises- man-in-the middle attacks.
3	**File Systems:** File concept, access methods, file system structure, allocation methods, free-space management, directory structure, efficiency and performance. **I/O Management:** I/O hardware, polling, interrupts, DMA, application I/O interface, performance. **Protection and Security:** Goals of protection, security problem, authentication, programme threats, system threats, threat monitoring, encryption.
4	**Free RTOS:** architecture, distribution, management of heap memory, task, queue, software timer, interrupt, resource management, memory management, task notification, low power support, porting.

Text Books

1. W. Stallings, Operating System: Internals and Design Principles, 8th ed., Prentice Hall, 2014.
2. A. Silberschatzet al., Operating System Concepts, 9th ed., John Wiley and Sons, 2012.
3. M. J. Bach, The Design of the Unix Operating System, People's Posts and Telecommunications Publishing House, 2003.
4. L. Qing and C. Yao, Real-time Concepts for Embedded Systems, CRC press, 2003.
5. R. Barry, Mastering the Free RTOS™ Real Time Kernel -A Hands-On Tutorial Guide, Real Time Engineers, 2016.

6. W. Mauerer, Professional Linux® Kernel Architecture, O'Reilly, 2010.

References

1. E. Sieveret al., Linux in a Nutshell, O'Reilly Media, 2005.
2. D. P. Bovet and M. Cesati, Understanding the Linux Kernel, O'Reilly, 2005.
3. F. Mayer et al., SELinux by Example: Using SecurityEnhancedLinux, Pearson Education, 2006.

OPTIMIZATION TECHNIQUES

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5010136, M5020136	**Optimization Techniques**	**2-1-0-1**	**2026**

Prerequisites: Nil

Course Objectives:

1. To provide students with a good understanding of optimization techniques.
2. To help the students develop the ability to solve problems using the learned concepts.
3. Connect the concepts to other domains, such as machine learning and pattern recognition, within and without optimization techniques.

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

CO1: Evaluate optimization problems and critically assess state-of-the-art solutions in the context of optimization techniques.

CO2: Design and create optimization algorithms through the integration of classical and modern optimization methods.

CO3: Develop, test, solve optimization techniques within team research projects, and effectively present the findings.

CO4: Critically analyze and compare optimization methods to identify suitable techniques for specific real-world problems.

CO5: Evaluate and defend the selection of optimization techniques for problem solving and research applications.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	1	3	3	1	1
CO2	2	3	3		3	1
CO3	2		3	3	3	
CO4	3	3	1	1	1	
CO5	2	3	2	3		2

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	Optimization - sequences and limits, derivative matrix, level sets and gradients, Taylor series.
2	Unconstrained optimization - necessary and sufficient conditions for optima, convex sets, convex functions, optima of convex functions, steepest descent, Newton and quasi-Newton methods, conjugate direction methods.
3	Constrained optimization- linearand non-linear constraints, equality and inequality constraints, optimality conditions.
4	Constrained convex optimization, projected gradient methods, penalty methods.

Text Books

1. E. K. P. Chong and S. H. Zak, An Introduction to Optimisation, 2nd ed. India: Wiley, 2010.

2. D. G. Luenberger and Y. Ye, *Linear and Nonlinear Programmeming*, 3rd ed., Springer, 2010.

References

1. S. Sra, S. Nowozin, and S. J. Wright, Optimization for Machine Learning, MIT Press, 2012.

2. R. Battiti and M. Brunato, The LION Way: Machine Learning Plus Intelligent Optimization, Createspace Independent Publishing, 2014.

PROGRAMMEMING IN PYTHON

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M4010104/ M4020104	**Programming in Python**	**2-1-0-1**	**2026**

Prerequisites: Nil

Course Objectives:

1. To help students learn problem-solving techniques.
2. To help students understand the fundamental concepts of programming using the Python. programming language and introduce the basic concepts of Object-Oriented programming in Python.
3. To introduce students to database concepts and simple data science tools.
4. To help students build practical skills in computational problem-solving through Programming in Python, Capstone Projects and Case Studies.

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

CO1: Apply foundational Python programming constructs—including data types, control flow, and collections—to implement structured and efficient code for solving computational problems.
CO2: Design modular programmes using user-defined functions, object-oriented principles, and file operations to enhance code reusability, abstraction, and data handling.
CO3: Perform advanced programming using NumPy and Pandas by creating, manipulating, and aggregating arrays and data frames to extract meaningful insights from structured data.
CO4: Develop and customize interactive data visualizations using libraries such as Matplotlib, Seaborn, and Plotly, and integrate data from relational databases using SQL and Python.
CO5: Create end-to-end Python applications by combining programming constructs, data analysis tools, and visualization techniques to solve real-world problems, as demonstrated through capstone projects.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.
PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.
PLO 3 Apply for a scholarship to conduct independent and innovative research.
PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.
PLO 5 Practice ethical standards of professional conduct and research.
PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	3	1	1	2	1
CO2	3	3	2	1	1	2
CO3	3	3	2	1	1	2
CO4	3	2	2	1	2	2
CO5	3	3	1	3	2	2

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus:

Module	Content
1	Introduction to Python, Real-world Applications of Python, Features and Implementations of Python, Running and Using Jupyter Notebooks, Basic Data Types, Input and Output Operations, Basic Operators, Comparison, and Logical Operators, Lists and Indexing, Advanced Indexing and Updating Data in a List, Tuples and Dictionaries, Conditional Execution, Loops, Looping through a Dictionary.
2	Writing and Using Functions: Function Arguments, User-Defined Functions, Methods, Global vs Local Scope, Nested Functions, Default and Flexible Arguments, Lambda Functions; Modules and Packages, String and List Methods, Exception Handling; File Handling in Python; Object-Oriented Programming: Classes, Methods, and Objects, Introduction to Object-Oriented Design.
3	Introduction to NumPy: 1D and 2D Arrays, Looping over Arrays, Statistical Tools; Introduction to Pandas: Series and Data Frames, Creating, Slicing, and Filtering, Importing and Exporting Data, Indexing and Selecting Data; Editing Data Frames: Setting and Transforming Columns; Combining and Reshaping. Data Frames: Grouping and Aggregating, Common Functionalities
4	Introduction to Matplotlib: Line, Bar, Scatter, Histogram, Box, Pair, and Time Series Plots, 3D Plotting, Line of Best Fit, Customising and Exporting Graphs, Introduction to Seaborn, Introduction to Plotly (for interactive visualizations), Database Concepts and SQL, SQL using Python, Capstone Projects and Case Studies

Lab Exercises

Module 1

1. Write programmes to practice basic data types, input/output operations, and operators. Include arithmetic, logical, and comparison operations.
2. Implement scripts using conditional execution. Add meaningful comments for readability and maintainability.
3. Practice loops and nested loops with control statements (break, continue). Examples: check for prime numbers, generate Fibonacci series, compute factorial, check for Armstrong and palindrome numbers.
4. Explore built-in data structures such as lists, tuples, sets, and dictionaries. Perform operations like insertion, deletion, and searching.

Module 2

5. Programmes to write user-defined functions to compute factorial, Fibonacci series, and sum of squares. Use default and keyword arguments.
6. Use lambda functions with map() and filter() to process lists. Examples: square even numbers, filter prime numbers.
7. Define a class Employee with methods to set and display employee details. Demonstrate class variables, constructors, and method overriding.
8. Create a Python script for file handling. Read from and write to a text file, and include exception handling for file-not-found and incorrect format.
9. Create a module with reusable mathematical functions. Import the module into another script and use its functions in computations.

Module 3

10. Use NumPy to create 1D and 2D arrays. Perform reshaping, slicing, and compute statistical measures such as mean, median, and standard deviation.
11. Load a CSV file using Pandas and perform operations such as filtering rows, slicing columns, and handling missing values.
12. Group and aggregate data in a DataFrame. Compute aggregate statistics like mean, count, and max; then sort the results.
13. Edit and transform a DataFrame. Add, rename, or drop columns, and apply filters based on conditions.
14. Combine multiple DataFrames using concatenation and merging. Reshape data using pivot and melt operations.

Module 4

15. Create plots using Matplotlib: line, histogram, and scatter plots. Customize axis labels, legends, titles, and styles.
16. Visualize data using Seaborn: generate pair plots, box plots, and heatmaps using built-in datasets.
17. Use Plotly to build interactive visualizations such as bar charts and time series plots. Export the results as HTML.
18. Connect to an SQLite database using Python. Retrieve student data with SQL queries and display results using Pandas.

19. Capstone Mini Project: Load a real-world dataset (e.g., COVID-19, sales, or student performance), clean and analyze it using Pandas and NumPy, visualize insights using at least two plotting libraries, and summarize findings in output or comments.

Text Books

1. C. Dierbach, Introduction to Computer Science Using Python: A Computational Problem-Solving Focus, Wiley, 2017.
2. A. N. Kamthane and A. A. Kamthane, Programming and Problem Solving with Python, McGraw-Hill Education, 2018.
3. S. F. Lott, Object Oriented Python, Packt Publishing, 2014.
4. W. McKinney, Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython, O'Reilly, 2012.

References

1. R. Thareja, Python Programming Using Problem Solving Approach, Oxford Higher Education, 2017.
2. B. N. Miller and D. L. Ranum, Problem Solving with Algorithms and Data Structures Using Python, Franklin, Beedle and Associates, 2011.
3. D. D. Riley and K. A. Hunt, Computational Thinking for the Modern Problem Solver, CRC Press, 2014.
4. J. VanderPlas, Python Data Science Handbook: Essential Tools for Working with Data , O'Reilly, 2016.
5. F. Nelli, Python Data Analytics: With Pandas, NumPy, and Matplotlib , 2nd ed., Apress, 2018.

QUANTUM COMPUTING

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5010128/ M5020128/ M5050128	**Quantum Computing**	**2-1-0-1**	**2026**
Prerequisites: Basic linear algebra			

Course Objectives:

1. To provide students with a solid foundation in quantum mechanics, including wave-particle duality, quantum tunneling, and the Schrödinger equation.
2. To enable students to effectively manipulate and analyze qubits, quantum gates, and circuits using foundational principles and visualization techniques.
3. To equip students with the skills to apply quantum algorithms to solve real-world problems in computation, cryptography, and other domains.

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

CO1. Evaluate the fundamental principles of quantum mechanics, to understand their role in quantum computation.

CO 2. Analyze and evaluate the efficiency of quantum algorithms, including Grover's search, Shor's factoring, and Deutsch-Jozsa algorithms, in solving computational problems.

CO3. Design and create quantum circuits and gates using qubits, Bloch sphere representation, and quantum entanglement.

CO4. Integrate and develop solutions involving advanced quantum concepts like teleportation, superdense coding, and the no-cloning theorem.

CO5. Formulate and create quantum computing models to address challenges in classical and quantum problem-solving.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	2	2	1	1	1
CO2	3	3	2	2	1	2
CO3	3	3	3	1	1	2
CO4	3	3	3	2	1	2
CO5	3	3	3	1	2	3

Syllabus

Module	Content

1	Elements of quantum mechanics, Wave-particle duality, Wave functions and probability amplitude, Heisenberg's uncertainty principle, Schrodinger equation, postulates of quantum mechanics, Quantum tunneling Research topics: quantum tunneling in nanoscale devices, uncertainty in quantum sensors, wave function interpretations.
2	Qubits, combining qubits using the tensor product, measuring qubits, Performing operations on qubits, Bra-ket notation, Bloch sphere representation, Qubit rotations, Projective measurements, Qubit modalities. Research topics: error mitigation in qubits, novel tensor product models, enhanced qubit visualizations.
3	Quantum gates, Quantum circuits, Quantum entanglement, No cloning theorem, Quantum teleportation, Super dense coding, Quantum parallelism, DiVincenzo's criteria for quantum computation Research topics: circuit optimization, non-standard gates, practical teleportation methods
4	Quantum Fourier transform, Deutsch's Algorithm, Deutsch-Jozsa Algorithm, Simon's periodicity algorithm, Grover's search algorithm, Shor's Factoring algorithm. Research topics: hybrid quantum-classical algorithms, advanced factorization methods, cryptographic applications.

Text Books

1. M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information*, Cambridge University Press, 2000.
2. V. Kasirajan, *Fundamentals of Quantum Computing, Theory and Practice*, Springer, 2021.
3. M. Nakahara and T. Ohmi, *Quantum Computing*, CRC Press, 2008.
4. M. Mosca, *An Introduction to Quantum Computing*, New York: Oxford University Press, 2007.

References

1. M. L. Bellac, *A Short Introduction to Quantum Information and Quantum Computation*, Cambridge University Press, 2006.
2. P. Kaye et al., *An Introduction to Quantum Computing,* Oxford, 2007.
3. A. Peres, *Quantum Theory: Concepts and Methods,* New York: Springer, 1993.
4. N. D. Mermin, *Quantum Computer Science*, Cambridge University Press, 2007.

REINFORCEMENT LEARNING

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5010106/ M5020106/ M5050106	**Reinforcement Learning**	**2-1-0-1**	**2026**

Prerequisites: Mathematics

Course Objectives:

1. To provide students with a good understanding of the concepts of the reinforcement learning.
2. To help the students develop the ability to solve problems using the learned concepts.
3. To connect the concepts to other domains.

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

CO1: Analyse the foundations of modern reinforcement learning theory, identify key challenges, and evaluate state-of-the-art solutions.

CO2: Evaluate the design, implementation, and integration of reinforcement learning algorithms and systems.

CO3: Analyse real-world scenarios to identify opportunities for applying reinforcement learning techniques effectively.

CO4: Create and demonstrate a functional reinforcement learning system through collaborative research projects and detailed presentations.

CO5: Create innovative reinforcement learning models and frameworks to address complex problems by leveraging advanced computational techniques.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	2	1	1	1	1
CO2	3	3	2	1	1	2
CO3	2	3	2	2	1	2
CO4	2	3	3	3	2	3
CO5	2	3	3	3	2	3

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	Introduction to Reinforcement Learning, Markov Processes Markov Reward Processes (MRPs) Markov Decision Processes (MDPs), MDP Policies, Policy Evaluation, Policy Improvement, Policy Iteration, Value operators.
2	Model-free learning - Q-learning, SARSA, Scaling up: RL with function approximation, RL with function approximation.
3	Imitation learning in large spaces, Policy search, Exploration/Exploitation, Meta-Learning, Batch Reinforcement Learning, Bandit problems and online learning.
4	Solution methods: dynamic programming, Monte Carlo learning, Temporal difference learning, Eligibility traces, Value function approximation, Models and planning.

Text Books

1. R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, MIT Press, 1998.
2. C. Szepesvari, *Algorithms for Reinforcement Learning*, Morgan and Claypool Publishers, 2010.

References

1. K. P. Murphy, Machine *Learning: A Probabilistic Perspective*, MIT Press, 2012.
2. M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programmeming, 1st ed. USA:* John Wiley and Sons, 1994.

ROBOTICS

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M4010109	**Robotics**	**2-1-0-1**	**2026**

Prerequisites: Prior knowledge of undergraduate level Mathematics, Programming

Course Objectives:

1. To impart a comprehensive and in-depth understanding of Robotics, technologies and applications to students by introducing and researching cutting-edge topics, technologies, applications and implementations.
2. To expose the students to frontier areas of Robotics while providing sufficient foundations for further study and research.

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

CO1: Analyze the foundations of robotics, including mechanics, planning, and control, and apply knowledge through assessments such as quizzes and examinations.

CO2: Design and implement robotics projects to prepare for industrial automation environments.

CO3: Evaluate and critique current literature in robotics to identify advancements and trends in the field.

CO4: Conduct research on robotics algorithms and demonstrate findings.

CO5: Integrate and apply theoretical and practical knowledge to solve real-world robotics challenges.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	1	3	2	1	1
CO2	1	3	3		3	
CO3	4		1	3	2	
CO4	3	3	4	1	1	
CO5	2	1	2	2		2

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus:

Module	Content

1	Introduction to the subject, automation, Industrial robots, Serial, parallel robots, configuration space, rigid body motion, kinematics, Inverse and forward kinematics, dynamics, trajectory planning, Applications. Configuration space: degrees off reedom (dof), robot joints, Grubler, s formula, Open and close chains, configuration space and its representation. Configuration and velocity constraints, task space and workspace
2	Rigid body motion: rigid body motion in plane, mathematical description of position and orientation, rotation and angular velocities, changing the coordinate frames, representations of rotation, homogenous transformation matrix, rotating and translating a vector or frame, twists, wrenches. Forward Kinematics: DH parameters, examples of DH parameters, product of exponential formulas,
3	Velocity, kinematics and statics: Jacobian, space Jacobian, body Jacobian, singularity analysis, manipulability, Inverse kinematics: Analytical and numerical inverse kinematics, kinematics of closed chains (parallel robots), Stewart-Gaugh platform. Manipulator dynamics: Acceleration of rigid body, mass distribution, Newton's equation, Example of close form dynamic equations, Lagrangian formulation of manipulator dynamics.
4	Trajectory generation: Point to point trajectory, polynomial via point trajectory, time scaling, Manipulator-mechanism design: basing design on task requirements, kinematic configuration, redundant and closed-chain structures, Actuation schemes, gears, shafts, links, Actuators, pneumatic cylinders, motors, position sensing, force sensing

Text Books

1. J. J. Craig, Introduction to Robotics: Mechanics and Control, 3rded., Pearson, 2004.
2. K. M. Lynch and F. C. Park, Modern Robotics: Mechanics, Planning, and Control, Cambridge University Press, 2017.
3. D. Fox and S. Thrun, Probabilistic Robotics, MIT Press, 2005.
4. S. K. Saha, Introduction to Robotics, McGraw-Hill Education, 2008.

SOFT COMPUTING

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5010108 / M5020108	**Soft Computing**	**2-1-0-1**	**2026**
Prerequisites: Nil			

Course Objectives:

1. To impart algorithmic skills needed for designing soft computing techniques and solutions.
2. To equip the students to identify and analyze problems solvable with soft computing techniques.
3. To impart solution design capability with soft computing techniques.

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

CO1: Inter application domains that require soft computing techniques.

CO2: Analyze algorithms to evaluate their effectiveness in solving soft computing problems.

CO3: Focus and construct solutions by applying appropriate soft computing methods.

CO4: Consider problem-solving approaches and justify the use of soft computing in specific scenarios.

CO5: Organize and illustrate the design and analysis of algorithms for soft computing through projects and presentations.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	2	1	2	1	2
CO2	3	3	1	2	1	3
CO3	3	3	2	1	2	3
CO4	2	3	1	1	1	3
CO5	3	3	2	1	1	2

Mapping of course outcomes with programme learning outcomes:

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	Difference between Soft and Hard computing, Overview of different components of soft computing techniques - Fuzzy Logic, Rough Logic, ANNs, Genetic Algorithms, Swarm Intelligence
2	Introduction to Fuzzy logic, Fuzzy membership functions, Operations on Fuzzy sets, Fuzzy relations, Fuzzy propositions, Fuzzy implications, Fuzzy inferences, Defuzzification, Fuzzy logic controller.

3	Genetic algorithms basic concepts, encoding, fitness function, Parent Selection - Roulette wheel, Rank, Tournament, Mutation and Crossover operators, Convergence of GA, Applications of GA, Case studies.
4	Swarm Intelligence - agent systems, social agents, Particle Swarm Optimization - path planning applications, Ant Colony Optimization - solving traveling salesman problem with ACO, introduction to Artificial Immune Systems

Text Books

1. R. Rajasekaranet al., Neural Networks, Fuzzy Logic, and Genetic Algorithms: Synthesis and Applications, India: Prentice Hall, 2011.
2. T. Ross, Fuzzy Logic with Engineering Applications, McGraw-Hill, 1997.
3. A. Slowik, Swarm Intelligence Algorithms, CRC press, 2020.

References

1. D. E. Goldberg, Genetic Algorithms in Search, Optimisation, and Machine Learning, Addison-Wesley, 1989.
2. E. Bonabeauet al., Swarm Intelligence: From Natural to Artificial Systems, Oxford University Press, 1999.
3. L. Polkowski and P. Verlag, Rough Sets: Mathematical Foundations, Heidelberg, 2002.

SPEECH PROCESSING

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5010109/ M5020109/ M5050109	**Speech Processing**	**2-1-0-1**	**2026**

Prerequisites: Mathematics

Course Objectives:

1. To give students a good understanding of speech processing tasks.
2. To help the students develop the ability to solve problems using the learned concepts.
3. Connect the concepts to other domains, such as machine learning and pattern recognition, within and without speech.

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

CO1: Analyse the foundations of modern speech processing theory, identify key problems, and explore state-of-the-art solutions.

CO2: Evaluate the effectiveness of speech signal processing algorithms and their integration into complete systems.

CO3: Create a functional speech signal processing system through a team-based research project, supported by a detailed project report and presentation.

CO4: Analyse the challenges in developing robust speech processing systems for diverse applications and environments.

CO5: Create innovative solutions by integrating speech signal processing techniques with advanced tools and technologies to address real-world problems.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.
PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.
PLO 3 Apply for a scholarship to conduct independent and innovative research.
PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.
PLO 5 Practice ethical standards of professional conduct and research.
PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	2	1	1	1	1
CO2	3	3	2	1	1	2
CO3	2	3	3	3	2	3
CO4	2	2	2	3	3	3
CO5	2	3	2	3	3	3

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	The human vocal and auditory systems. Characteristics of speech signals: phonemes, prosody, IPA notation. Lossless tube model of speech production. Time and frequency domain representations of speech; window characteristics and time/frequency resolution tradeoffs. Properties of digital filters: mean log response, resonance gain and bandwidth relations, bandwidth expansion transformation, all-pass filter characteristics.
2	Autocorrelation and covariance linear prediction of speech; optimality criteria in time and frequency domains; alternate LPC parametrisation. Speech coding: PCM, ADPCM, CELP. Speech synthesis: language processing, prosody, diphone and formant synthesis; time domain pitch and speech modification.
3	Speech recognition: hidden Markov models and associated recognition and training algorithms. Language modelling. Large vocabulary recognition. Acoustic preprocessing for speech recognition.
4	Speech Processing: Spectral and non-spectral analysis techniques, Model- based coding techniques, Noise reduction and echo cancellation, Synthetic and coded speech quality assessment. Selection of recognition unit, Model-based recognition, Language modeling, Speaker Identification, Text analysis and text-to- speech synthesis.

Text Books

1. L. Rabiner and R. Schafer, Theory and Applications of Digital Speech Processing, 1st ed. USA: Prentice Hall Press, 2010.
2. B. Gold et al., Speech and Audio Signal Processing: Processing and Perception of Speech and Music, 2nd ed. USA: Wiley-Inderscience, 2011.

References

1. D. O'Shaughnessy, Speech Communication: Human and Machine, Addison-Wesley, 1987.
2. T. Ogunfunmiet al., Speech and Audio Processing for Coding, Enhancement and Recognition, Springer, 2014.
3. J. Benestyet al., Springer Handbook of Speech Processing, Berlin: Springer, 2008.

STOCHASTIC PROCESSES AND MODELS

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5020132	**Stochastic Processes and Models**	**2-1-0-1**	**2026**

Prerequisites: Mathematics

Course Objectives:

1. To give students a good understanding of the concepts of information-theoretic methods, stochastic processes, and models.
2. To help the students develop the ability to solve problems using the learned concepts.
3. Connect the concepts to other domains, such as machine learning and pattern recognition, within and without stochastic processes and models.

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

CO1: Analyse the mathematical foundations of modern stochastic models, their associated problems, and state-of-the-art solutions.

CO2: Evaluate the design, development, and integration of stochastic models, algorithms, and systems.

CO3: Create and demonstrate a functional stochastic model through collaborative research projects and project report presentations.

CO4: Analyse real-world scenarios to identify opportunities for applying stochastic modelling techniques.

CO5: Create innovative stochastic models and algorithms by integrating advanced mathematical concepts and computational techniques.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.
PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.
PLO 3 Apply for a scholarship to conduct independent and innovative research.
PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.
PLO 5 Practice ethical standards of professional conduct and research.
PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	2	1	1	1	1
CO2	3	3	2	1	1	2
CO3	2	3	3	3	2	3
CO4	3	2	2	2	1	2
CO5	2	3	3	2	2	3

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	Concepts of multiple random variables. Bayesian belief networks (BBN): Representation, Independence and conditional independence, Partial independence and other structure. Exact inference in BBN: Variable elimination, Pearl's algorithm, Junction tree, Recursive decomposition, Using additional structure.
2	Approximate inference: Monte Carlo approximations, Loopy belief propagation, Variational methods. Learning of BBNs: learning parameters, learning structure, Bayesian averaging, EM (learning with hidden variables and missing values), structural EM.
3	Dynamic belief networks: Particle filtering. Markov random fields (Markov networks): Representation (potentials), Independence and conditional independence, Trees, Boltzman machines, Conditional Markov random fields.
4	Inference in Markov networks. Learning Markov networks: Iterative proportional fitting, Cluster variational methods, Other approximations. Relational graphical models.

Text Books

1. D. Koller and N. Friedman, Probabilistic Graphical Models: Principles and Techniques, MIT Press, 2009.
2. D. Barber, Bayesian Reasoning and Machine Learning, Cambridge University Press, 2012.
3. D. J. C. Mackay, Information Theory, Inference, and Learning Algorithms, UK: Cam- bridge University Press, 2003
4. J. Pearl, Probabilistic Reasoning in Intelligent Systems, Morgan Kaufman, 1997.

SYSTEMS SECURITY AND RISK ANALYSIS

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5010115/ M5020115/ M5050115	**Systems Security and Risk Analysis**	**2-1-0-1**	**2026**

Prerequisites: Prior Knowledge of operating systems, computer networks, web technology, DBMS, security fundamentals, mathematics.

Course Objectives:

1. To impart a comprehensive and in-depth understanding of systems security and risk analysis.
2. To enable the students to study an organization, model security, measure risk, and design security strategy.

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

CO1: Analyse the threat landscape to identify and evaluate potential security risks in an IT organization.

CO2: Analyse the risk factors and vulnerabilities within an IT organization to assess potential threats.

CO3: Create comprehensive defense strategies to mitigate identified risks and enhance the organization's security posture.

CO4: Evaluate security emergency scenarios and respond with appropriate countermeasures and recovery strategies.

CO5: Evaluate the effectiveness of security protocols and response strategies to ensure organizational resilience.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	3	3	1	2	2	1
CO2	3	3	1	2	2	2
CO3	3	3	2	2	3	3
CO4	3	3	2	2	3	3
CO5	3	3	2	2	3	3

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	Discussion of fundamental Network and Systems security issues: Various Attacks on - Network Protocols, Systems, Web Infrastructure
2	Phases of YACRAF Risk Analysis: Phase 0: Scope and delimitations Phase 1: Business Analysis Phase 2: System Definition and Decomposition
3	Phase 3: Threat Analysis Phase 4: Attack and Resilience Analysis Phase 5: Risk Assessment and Recommendations
4	Main Assignment: Think like a CISO!

Text Books

1. T. UcedaVelez and M. Morana, *PASTA: Risk Centric Threat Modeling: Process for At- tack Simulation and Threat Analysis*,John Wiley and Sons, 2015.
2. J. Freund and J. Jones, *Measuring and Managing Information Risk: A FAIR Approach*, Butterworth-Heinemann., 2014.
3. W. Du, *Computer Security: A Hands-on Approach*, CreateSpace Independent Publishing, 2017.

References

1. A. Hoffman, *Web Application Security*, O'Reilly, 2020.
2. P. Ackerman, *Industrial Cybersecurity: Efficiently Secure Critical Infrastructure Systems*, Packt Publishing, 2017.
3. W. Stallings, *Cryptography and Network Security: Principles and Practice*, Prentice Hall, 2017.
4. M. Ekstedt, Z. Afzal, P. Mukherjee et al., *Yet Another Cybersecurity Risk Assessment Framework*, International Journal of Information Security, Springer, 2023.

TECHNICAL COMMUNICATION

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5010141	**Technical Communication**	**1-1-0-0**	**2026**

Prerequisites: Basic English, Grammar rules.

Course Objectives:

1. Get the fundamental knowledge of technical communication.
2. Write technical documents in proper format and structure.
3. Communicate effectively in a professional context, using appropriate rhetorical approaches.
4. Adapt content and rhetorical strategies according to the audience and purpose of each document.
5. Create and deliver technical briefings tailored to specific audiences, purposes, and media.

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

CO1: Interpret the nature, objective, and importance of Technical Communication.
C02: Develop technical write-ups effectively.
CO3: Boost their confidence in public speaking.
C04: Plan and deliver presentations in front of a diverse audience.
CO5: Evaluate and improve communication efficiency through mastery of voice dynamics.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	2	2	1	3	3	3
CO2	3	3	1	2	2	3
CO3	3	3	3	2	1	3
CO4	3	2	1	3	2	3
CO5	3	1	1	1	2	3

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content
1	**Fundamentals of Technical Communication:** Features of technical communication, The distinction between General and Technical Communication, Language as a tool of Communication, Dynamics of Communication: Definition and process, Kinesics, Proxemics, Paralinguistic features, Importance of Interpersonal and Intercultural Communication in today's organization, The flow of Communication: Downward; upward, Lateral or Horizontal, Barriers to Communication, Code and Content, Stimulus and Response, Encoding process, Decoding process, Professional Personality Attributes
2	**Forms of Technical Writing** Synopsis writing, Technical Report, Thesis/ Project writing, Technical research Paper writing, Seminar and Conference paper writing, Expert Technical Lecture, 7 Cs of effective business writing: concreteness, completeness, clarity, conciseness, courtesy, correctness, consideration, C.V./Resume writing, Technical Proposal, Email writing, Agenda of meeting, Minutes of meeting

3	**Voice Dynamics and Oral Communication** Pronunciation Etiquette; Syllables; Vowel sounds; Consonant sounds; Tone: Rising tone; Falling Tone; Flow in Speaking, Speaking with a purpose, Speech and personality, Professional Personality Attributes: Empathy; Considerateness; Leadership; Competence. Public speaking, Overcoming Stage Fear: Confident speaking; Audience Analysis and retention of audience interest, Presentation strategies, Interview skills, Negotiation skills Critical and Creative thinking in communication.
4	**Technical Presentation: Case Studies Using Learnt Strategies and Techniques** Presentation Skills for Technical Paper/Project Reports/ Professional Reports based on proper Stress and Intonation Mechanics, Comprehension Skills based on Reading and Listening Practicals on a model AudioVisual Usage, Role Play, Group Discussion, Extempore, Mock Interview, Conducting meetings and minutes of meeting.

Text Books

1. M. Raman and S. Sharma, *Technical Communication – Principles and Practices*, Oxford Univ. Press, 2007.
2. R.C. Sharma and K. Mohan, *Business Correspondence and Report Writing*, McGraw- Hill, 2001.
3. L. U. B. Pandey, *Practical Communication: Process and Practice*, India: A.I.T.B.S. Publications, 2014.
4. T. A. Sherman *et al.*, *Modern Technical Writing*, Apprentice Hall, 2015.
5. S.D. Sharma, *A Text Book of Scientific and Technical Writing*, Vikas Publication, 2008.
6. M. Murphy, *Skills for Effective Business Communication*, Harvard University, 2014.
7. P. Mehra, *Business Communication for Managers*, Pearson Publication, 2011

WEB TECHNOLOGY

Course Code	Course Name	Credit Split Lecture/Lab/Seminar/Project	Year of Introduction
M5020123	**Web Technology**	**2-1-0-1**	**2026**
Prerequisites: Nil			

Course Objectives:

1. To help students understand the web application fundamentals.
2. To explore the architecture and design principles of web-based applications.
3. To understand the most suitable application stack for a requirement and its implementation.
4. To explore a few related concepts like Microservices, common web application security issues, REST API.

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

CO1: Assess and compare different web server architectures, transmission protocols, and web application structures to justify their use in diverse scenarios.

CO2: Design and develop dynamic web pages using CSS frameworks and advanced JavaScript techniques, incorporating modern libraries and frameworks like React.

CO3: Critically analyze and justify the application of creational design patterns, such as Factory and Singleton, to address complex web application challenges.

CO4: Formulate and implement scalable web applications using Django's MTV architecture, integrating REST APIs and data visualization techniques for effective solutions.

CO5: Derive and create microservices-based solutions by contrasting them with monolithic designs, leveraging MVC and N-tier architecture principles.

Programme Learning Outcomes:

PLO 1 Develop strong fundamental disciplinary knowledge.

PLO 2 Demonstrate research skills that are of an experimental, computational, or theoretical nature.

PLO 3 Apply for a scholarship to conduct independent and innovative research.

PLO 4 Show communication skills in various formats (oral, written) and to expert and non- expert audiences.

PLO 5 Practice ethical standards of professional conduct and research.

PLO 6 Acquire professional skills such as collaborative skills, ability to write grants, entrepreneurial skills, and writing articles for scholarly journals if it is taught by faculty in the School.

Mapping of course outcomes with programme learning outcomes:

	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6
CO1	1	1	3	3	1	3
CO2	2	1	3	2	1	2
CO3	2	1	3	3	1	3
CO4	1	1	2	3	2	3
CO5	1	1	2	2	2	1

(Correlation: 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High))

Syllabus

Module	Content

1	Design, HTML5 Elements, Attributes and elements, Type of Style sheets: Internal Style Sheet, Inline Style sheet, External Style Sheet, CSS3 Elements and features, CSS frameworks, Content delivery network, Selectors, XML Schema, Presenting XML Using XML Processors: DOM and SAX.
2	Introduction to Java Script, Object in JavaScript, Dynamic HTML with Java Script, JavaScript Object Notation, Data types, Arrays, Decisions and Loops, Functions and scope, JavaScript libraries, JavaScript Frameworks, ECMAScript, TypeScript, Single page applications (SPA), Basics of React Web Framework, Introduction to MERN framework.
3	Creational Design Patterns, Factory Pattern, Abstract Factory Pattern, Prototype pattern, Singleton Pattern, Builder Pattern, Dependency Injection pattern, The Web Services based on technologies such as SOAP, REST, WSDL, Django Framework: Architecture, MTV Architecture Pattern in Django Structure.
4	Data Access with Django and Python, CRUD Operations with DJango, Models, Templates, Controllers, Sample Django MTV Web Application, REST API with Django - Advanced, Cache and Sessions with Django, Data Visualization Techniques for small and large data, Fundamentals of web application architecture (1Tier, 2-Tier, 3-Tier, NTier and MVC) and components, User interface app components, Structural components, Microservices, Monolithic vs. Microservices.

Text Books

1. J. C. Jackson, *Web Technologies - A Computer Science Perspective*, Pearson Education, 2009.
2. William S Vincent, *Django for Professionals: Production Websites with Python & Django* Paperback, Import - 2019.
3. J. B. Mille, *Internet Technologies and Information Services,* ABC-CLIO, 2014.