

Dr. Babasaheb Ambedkar Technological University
(Established as a University of Technology in the State of Maharashtra)

(Under Maharashtra Act No. XXIX of 2014)

P.O. Lonere, Dist. Raigad, Pin 402 103, Maharashtra Telephone and Fax.

02140 - 275142

www.dbatu.ac.in



Curriculum (As per NEP)
UNDER GRADUATE PROGRAMME
(B. Tech. Information Technology)
THIRD YEAR ENGINEERING

WITH EFFECT FROM THE ACADEMIC YEAR 2025-2026



Vision of the University

"The University is committed to become a leading 'Center of Excellence' in the field of Engineering, Technology, and Science as a seat of learning with a national character and international outlook."

Mission of the University

"The University is committed to provide quality technical education, research, and development services to meet the needs of industry, business, service sector, and society, at large."

Vision of the Department

"To achieve excellence in teaching, learning, and research to develop quality engineers to meet the current trends in the emerging world of Information Technology."

Mission of the Department

M1: The department constantly aims at providing quality technical education.

M2: The department works in pace with modern scientific and technological development.

M3: The department strives to meet the needs of industry, business, the service sector, and society at large."

Semester V													
Sr. No	Course Title	Course Code	Course Type	Teaching Scheme – Hours per week				Examination Scheme - Marks					Total Marks
				L	T	P	CR	CA	MSE	ESE	TW	PR/OR	
1	Database Management Systems	25UD1246PC501	PCC	2	0	0	2	20	20	60	-	-	100
2	Machine Learning and Artificial Neural Network	25UD1246PC502		2	0	0	2	20	20	60	-	-	100
3	Cloud Computing and Security	25UD1246PC503		2	0	0	2	20	20	60	-	-	100
4	Software Engineering	25UD1246PC504		2	0	0	2	20	20	60	-	-	100
	PEC-I (Any Two)												
4	A. Information and Cybersecurity	25UD1246PEC505A	PEC-I	2	0	0	2	20	20	60	-	-	100
	B. Data Science and Visualization	25UD1246PEC505B		2	0	0	2	20	20	60	-	-	100
	C. Fundamentals of Hyperspectral Imaging	25UD1246PEC505C		2	0	0	2	20	20	60	-	-	100
	D. Introduction to IoT	25UD1246PEC505D		2	0	0	2	20	20	60	-	-	100
	E. Advanced Machine Learning	25UD1246PEC505E		2	0	0	2	20	20	60	-	-	100
	F. Introduction to Blockchain	25UD1246PEC505F		2	0	0	2	20	20	60	-	-	100
	G. Virtualization and Containers	25UD1246PEC505G		2	0	0	2	20	20	60	-	-	100
	H. DevSecOps Principles and Practices	25UD1246PEC505H		2	0	0	2	20	20	60	-	-	100
	I. Introduction to Quantum Computing	25UD1246PEC505I		2	0	0	2	20	20	60	-	-	100
	J. Human-Computer Interaction	25UD1246PEC505J		2	0	0	2	20	20	60	-	-	100
	K. Software Project Management	25UD1246PEC505K		2	0	0	2	20	20	60	-	-	100
L. Network Management	25UD1246PEC505L	2	0	0	2	20	20	60	-	-	100		
M. Finance Engineering	25UD1246PEC505M	2	0	0	2	20	20	60	-	-	100		
5	Multi-Disciplinary Minor #	25UD1246MD506	MD M#	3	0	2	4	20	20	60	-	-	100
6	Open Electives*	25UD1246OE507	OE*	2	0	0	2	20	20	60	-	-	100
7	Database Management Systems Lab	25UD1246PCL508		0	0	2	1	-	-	-	60	40	100

8	Machine Learning and Artificial Neural Network Lab	25UD1246PCL509	PCC	0	0	2	1	-	-	-	60	40	100
9	Cloud Computing and Security Lab	25UD1246PCL510		0	0	2	1	-	-	-	60	40	100
10	Software Engineering Lab	25UD1246PCL511		0	0	2	1	-	-	-	60	40	100
11	PEC-I Lab (Any Two)	25UD1246PECL511 A/B/C/D/E/F/G/H/I/J	PEC-I	0	0	4	2				60	40	100
Total				17	0	14	24	160	160	480	360	240	1400

* Students will have to choose one of the courses from a bucket of “Open Electives” given in Annexure I. The updated list is available on the University website.

4 Credits based on the choice of MD-M from another department.

Semester VI

Sr. No	Course Title	Course Code	Course Type	Teaching Scheme – Hours per week				Examination Scheme - Marks					
				L	T	P	CR	CA	MSE	ESE	TW	PR/OR	Total
1	Advanced Database Technology	25UD1246PC601	PCC	2	0	0	2	20	20	60	-	-	100
2	Advanced Web Technologies	25UD1246PC602		2	0	0	2	20	20	60	-	-	100
3	Compiler Design	25UD1246PC603		2	0	0	2	20	20	60	-	-	100
4	Deep Learning	25UD1246PEC604		2	0	0	2	20	20	60	-	-	100
5	PEC-II (Any Two)		PEC-II										
	A. Ethical Hacking	25UD1246PEC605A		2	0	0	2	20	20	60	-	-	100
	B. Data Warehousing and Data Mining	25UD1246PEC605B		2	0	0	2	20	20	60	-	-	100
	C. Image Processing Techniques for Hyperspectral Data	25UD1246PEC605C		2	0	0	2	20	20	60	-	-	100
	D. Wireless Sensor Networks	25UD1246PEC605D		2	0	0	2	20	20	60	-	-	100
	E. Natural Language Processing	25UD1246PEC605E		2	0	0	2	20	20	60	-	-	100
	F. Block chain in Finance	25UD1246PEC605F		2	0	0	2	20	20	60	-	-	100
	G. Cloud Platforms: AWS/Azure/GCP.	25UD1246PEC605G		2	0	0	2	20	20	60	-	-	100
	H. Version Control with Git	25UD1246PEC605H		2	0	0	2	20	20	60	-	-	100
	I. Quantum Computing Algorithms	25UD1246PEC605I		2	0	0	2	20	20	60	-	-	100
	J. User Interface Design	25UD1246PEC605J		2	0	0	2	20	20	60	-	-	100
	K. Software Testing	25UD1246PEC605K		2	0	0	2	20	20	60	-	-	100
	L. Wireless and Mobile Communication	25UD1246PEC605L		2	0	0	2	20	20	60	-	-	100
M. Algorithmic Finance	25UD1246PEC605M	2	0	0	2	20	20	60	-	-	100		
6	Multi-Disciplinary Minor #	25UD1246MD606	MD M#	2	0	0	2	20	20	60	-	-	100
7	Time Series Modelling and Forecasting	25UD1246VSL607	VSEC	1	0	2	2	-	-	-	60	40	100
8	Advanced Database Technologies Lab	25UD1246PCL608	PCC	0	0	2	1	-	-	-	60	40	100
9	Advanced Web Technology Lab	25UD1246PCL609		0	0	2	1	-	-	-	60	40	100
10	Compiler Design Lab	25UD1246PCL6010		0	0	2	1	-	-	-	60	40	100
11	Deep Learning Lab	25UD1246PCL6011		0	0	2	1	-	-	-	60	40	100
12	PEC-II Lab (Any Two)	25UD1246PECL612 A/B/C/D/E/F/G/H/I/J /K/L/M	PEC-II	0	0	4	2	-	-	-	60	40	100
	Total			15	0	14	22	140	140	420	420	280	1400

2 Credits based on the choice of MD-M from another department.

Abbreviations: Generic/ Open Electives: **OE**; Vocational Skill and Skill Enhancement Courses: **VSEC**; Vocational Skill Courses: **VSC**; Skill Enhancement Courses: **SEC**; Ability Enhancement Courses: **AEC**; Indian Knowledge System: **IKS**; Value Education Courses: **VEC**; OJT: On Job Training: Internship/ Apprenticeship; Field projects: **FP**; Community engagement project: **CEP**; Co-curricular Courses: **CC**; Research Methodology: **RM**; Research Project: **RP**, Liberal Learning Course: **Lib. Learn**, Courses on Humanities, Social Science, and Management: **HSSM**

Multidisciplinary Minor I [Semester V]

Sr. No	Course Title	Course Code	Course Type	Teaching Scheme – Hours per week				Examination Scheme - Marks					Total	
				L	T	P	CR	CA	MSE	ESE	TW	PR/OR		
	Any One													
1	Database Management Systems	25UD1246MD506A	MD-M	3	0	2	4	20	20	60	-	-	100	
2	Machine Learning and Artificial Neural Network	25UD1246MD506B												
3	Cloud Computing and Security	25UD1246MD506C												
4	Software Engineering	25UD1246MD506D												
5	Information and Cybersecurity	25UD1246MD506E												
6	Data Science and Visualization	25UD1246MD506F												
7	Fundamentals of Hyperspectral Imaging	25UD1246MD506G												
8	Introduction to IoT	25UD1246MD506H												
9	Advanced Machine Learning	25UD1246MD506I												
10	Introduction to Block chain	25UD1246MD506J												
11	Virtualization and Containers	25UD1246MD506K												
12	DevSecOps Principles and Practices	25UD1246MD506L												
13	Introduction to Quantum Computing	25UD1246MD506M												
14	Human-Computer Interaction	25UD1246MD506N												
15	Software Project Management	25UD1246MD506O												
16	Network Management	25UD1246MD506P												
17	Finance Engineering	25UD1246MD506Q												

Multidisciplinary Minor II [Semester VI]

Sr. No	Course Title	Course Code	Course Type	Teaching Scheme – Hours per week				Examination Scheme - Marks					Total
				L	T	P	CR	CA	MSE	ESE	TW	PR/OR	
	Any One												
1	Advanced Database Technology	25UD1246MD606A	MD-M	2	0	0	2	20	20	60	-	-	100
2	Advanced Web Technologies	25UD1246MD606B											
3	Compiler Design	25UD1246MD606C											
4	Deep Learning	25UD1246MD606D											
5	Ethical Hacking	25UD1246MD606E											
6	Data Warehousing and Data Mining	25UD1246MD606F											
7	Image Processing Techniques for Hyperspectral Data	25UD1246MD606G											
8	Wireless Sensor Networks	25UD1246MD606H											
9	Advanced Machine Learning	25UD1246MD606I											
10	Blockchain in Finance	25UD1246MD606J											
11	Virtualization and Containers	25UD1246MD606K											
12	Version Control with Git	25UD1246MD606L											
13	Quantum Computing Algorithms	25UD1246MD606M											
14	User Interface Design	25UD1246MD606N											
15	Software Testing	25UD1246MD606O											
16	Wireless and Mobile Communication	25UD1246MD606P											
17	Algorithmic Finance	25UD1246MD606Q											

B. Tech. IT (Honors)[Semester V]

Sr. No	Course Title	Course Code	Course Type	Teaching Scheme – Hours per week				Examination Scheme - Marks					Total
				L	T	P	CR	CA	MSE	ES E	T W	PR/ OR	
	Any One		Honors	03	00	00	03	20	20	60	-	-	100
1	Cyber Security for CPS	25UD1246HR501											
2	Business and IT Management	25UD1246HR502											

B. Tech. IT (Honors) [Semester VI]

Sr. No	Course Title	Course Code	Course Type	Teaching Scheme – Hours per week				Examination Scheme - Marks					Total
				L	T	P	CR	CA	MSE	ESE	TW	PR/ OR	
	Any One		Honors	03	00	00	03	20	20	60	-	-	100
1	Internet of Things (IoT) and Smart Systems	25UD1246HR601											
2	IT Laws, Ethics, and Digital Privacy	25UD1246HR602											
3	AR/VR Technologies and Applications	25UD1246HR603											

EXIT SCHEME:

The student will have to complete all courses from any one of the groups to be eligible to get a B. Voc. in Information Technology after second year.

Bucket of Courses: Group I

- Programming and Software Development
- Advanced Python Programming
- Full Stack Web Development (HTML, CSS, JavaScript, Node.js)
- DevSecOps and Scripting

Bucket of Courses: Group II

- Cisco Certified Network Associate (CCNA)
- Network Configuration and Troubleshooting
- Cybersecurity Essentials
- Ethical Hacking and Penetration Testing
- Firewall Management and Security Policies

Bucket of Courses: Group III

- SQL and Database Management Systems
- NoSQL Databases (MongoDB, Cassandra)
- Data Warehousing and ETL (Extract, Transform, Load)
- Database Administration

Bucket of Courses: Group IV

- Cloud Computing and Virtualization
- Cloud Security and Compliance
- Amazon Web Services (AWS) Certification
- Microsoft Azure Fundamentals

Course Title : Database Management Systems

Semester : V

Prerequisite :

L – T – P : 2-0-0

Course Code : 25UD1246PC501

Credits : 2

Course Type : PCC

Course Objective:

1. To introduce the fundamental concepts, architecture, and advantages of DBMS, along with E-R modeling and data abstraction.
2. To develop an understanding of relational data models and relational query languages including relational algebra and calculus.
3. To provide knowledge of advanced SQL operations, view management, nested queries, joins, and stored procedures.
4. To explain the process of normalization and relational design, and introduce storage and indexing techniques.
5. To study query optimization, transaction processing, concurrency control, and recovery mechanisms in DBMS

Course Outcomes:

After learning the course, the students will be able to:

1. Explain the architecture, components, and data models of a DBMS, and illustrate E-R modeling with real-world examples.
2. Apply relational algebra/calculus and SQL commands to retrieve and manipulate data from relational databases.
3. Analyze and construct complex SQL queries using joins, nested queries, aggregate functions, and views.
4. Evaluate database schemas using functional dependencies and normalization techniques to improve design efficiency.
5. Design and implement solutions involving query optimization, transaction control, and crash recovery strategies.

Course Contents:

UNIT I

Introduction to DBMS:

[06 Hours]

Basic concepts, advantages of DBMS over file systems, data abstraction and data independence, DBMS architecture, components, and data models. Introduction to E-R modeling: entities, attributes, relationships, constraints, keys, and E-R diagrams.

UNIT II

Relational Model:

[06 Hours]

Basic concepts, Attributes and domains, Concept of integrity and referential constraints, Schema diagram, Relational query languages, Relational Algebra and Relational Calculus: Tuple relational and domain relational calculus. Structured Query Language-I: Introduction,

Characteristics and advantages, Data types and literals, DDL, Tables: creating, modifying, deleting.

UNIT III

Views: [06 Hours]

Creating, Dropping, Updation using views, DML, Operators, SQL DML queries, SELECT query and clauses. Structured Query Language- II: Set operations, Predicates and joins, Set membership, Tuple variables, Set comparison, Ordering of tuples, Aggregate functions, Nested queries, Database modification using SQL Insert, Update and Delete queries, Dynamic and SQL and concept of stored procedures

UNIT IV

Relational Database Design: [06 Hours]

Notion of normalized relations, Functional dependency, Decomposition and properties of decomposition, Normalization using functional dependency, Multi-valued dependency and join dependency. Storage and File Systems: Secondary storage, RAID, File organization, Indices, Static and dynamic hashing, B-Trees and B+ Trees.

UNIT V

Query Processing and Transaction Management: [06 Hours]

Measures of query cost, Selection operation, Sorting and join operation, Transaction concept, Components of transaction management, Concurrency and recovery system, Different concurrency control protocols such as timestamps and locking, Validation, Multiple granularity, Deadlock handling, Different crash recovery methods such as log-based recovery, Shadow-paging, Buffer management and Remote backup system

Textbooks:

1. Abraham Silberschatz, Henry F. Korth, and S. Sudarshan, “*Database System Concepts*”, McGraw Hill Education, ISBN: 9789390727506, 7th Edition, 2021.
2. RamezElmasri and Shamkant B. Navathe, “*Fundamental Database Systems*”, Pearson Education, 7th Edition, 2017.
3. Raghu Ramkrishnan, Johannes Gehrke, “*Database Management Systems*”, McGraw Hill Education, 3rd Edition, 2007.

Reference books:

1. Peter Rob, Carlos Coronel, “*Database Systems: Design, Implementation, and Management*”, Thomson/ Course Technology, 2007
2. Hector Garcia-Molina , “*Database Systems: The Complete Book*”, Pearson Education, 2008.

Course Title : Machine Learning and Artificial Neural Network

Semester : V

Prerequisite : Probability and statistics

L – T – P : 2-0-0

Course Code : 25UD1246PC502

Credits : 2

Course Type : PCC

Course Objective:

1. **To introduce** fundamental concepts, definitions, and challenges in machine learning, including performance evaluation and model selection techniques.
2. **To develop** an understanding of supervised and unsupervised learning algorithms, as well as essential feature engineering techniques.
3. **To explain** the architecture, types, and mathematical foundations of artificial neural networks for solving classification and regression problems.
4. **To impart** knowledge of learning rules, activation functions, optimization methods, and backpropagation for effective neural network training.
5. **To enable** students to understand multilayer perceptron models using modern machine learning and deep learning libraries.

Course Outcomes:

After learning the course, the students will be able to:

1. To explain the basic concepts of machine learning, its types, performance metrics, and Python tools for ML.
2. To implement supervised and unsupervised learning algorithms along with effective feature engineering.
3. To describe artificial neuron models, neural network types, and their mathematical foundations.
4. To apply learning laws, activation functions, and backpropagation algorithms to train neural networks.
5. To understand MLP architectures with proper regularization and parameter tuning

Course Contents:

UNIT I

Introduction to Machine Learning:

[6 Hours]

Basic definitions, types of learning, Well-posed learning problems, Designing a Learning System, Perspectives and Issues in Machine learning, cross-validation, Bias-Variance Trade off, The VC Dimension, Occam's razor. Performance Measures, Introduction to Python, scientific computing packages (NumPy, SciPy, Matplotlib), and simple ML packages (Scikit-learn, Pytorch).

UNIT II

Supervised Learning and Feature Engineering Techniques: [6 Hours]

Linear regression, Maximum likelihood and least square error hypotheses.

Logistic Regression, gradient descent algorithm, Decision trees, Instance based learning, Bayes theorem and concept learning, Naïve Bay's classifier

Clustering: k- means, Partitional clustering, Hierarchical clustering.

UNIT III

Probabilistic Models and Support Vector Machines: [6 Hours]

Biological Neurons vs. Artificial Neurons, McCulloch–Pitts Model, Perceptron model, learning rule, convergence theorem, Limitations of the single-layer perceptron (XOR problem), Types of neural networks: Feedforward, Feedback, Radial Basis Function (RBF), Applications of ANN, Mathematical preliminaries: Vectors, matrices, dot product, activation functions basics.

UNIT IV

Learning laws: [6 Hours]

Hebbian learning, Delta rule, Widrow–Hoff rule, Activation functions: Step, Sigmoid, Tanh, ReLU, Leaky ReLU, Softmax, Gradient Descent and variants (SGD, Mini-batch, Momentum, Adam, RMSprop), Cost functions: MSE, Cross-Entropy, Backpropagation algorithm: Derivation & implementation, Epochs, batches, and learning rate scheduling.

UNIT V

Ensemble learning: [6 Hours]

Architecture of MLP (Input, Hidden, Output layers), Universal Approximation Theorem, Weight initialization strategies (Xavier, He initialization), Overfitting & underfitting in ANN, Regularization techniques: L1, L2, Dropout, Early stopping, Batch Normalization,.

Textbooks:

1. Tom M. Mitchell, “*Machine Learning*,” McGraw Hill, 1997.
2. Ian Goodfellow, Yoshua Bengio, Aaron Courville, “*Deep Learning*”, MIT Press, 2016.
3. Christopher M. Bishop, “*Pattern Recognition and Machine Learning*,” Springer, 2006.
4. Ethem Alpaydin, “*Introduction to Machine Learning*,” MIT Press, 2020.
5. Aurélien Géron, “*Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*,” O’Reilly, 2022.

Reference books:

1. Kevin P. Murphy, *Machine Learning: “A Probabilistic Perspective*,” MIT Press, 2012.
2. Trevor Hastie, Robert Tibshirani, Jerome Friedman, “*The Elements of Statistical Learning*,” Springer, 2009.
3. François Chollet, “*Deep Learning with Python*,” Manning, 2021.
4. Sebastian Raschka & Vahid Mirjalili, “*Python Machine Learning*,” Packt, 2022.

Course Title : Cloud Computing and Security

Semester : V

Prerequisite : Computer Networks

L – T – P : 2-0-0

Course Code : 25UD1246PC503

Credits : 2

Course Type : PCC

Course Objective:

1. Understand the fundamental concepts, evolution, and characteristics of cloud computing.
2. Explore cloud architecture, virtualization, and resource provisioning techniques.
3. Analyze the significance of big data processing using Hadoop and MapReduce in cloud environments.
4. Identify various types of cyber threats, attack vectors, and basic network security mechanisms.
5. Understand cloud security architecture, data protection methods, and risk management techniques.

Course Outcomes:

After learning the course, the students will be able to:

1. Explain cloud computing models, deployment strategies, and their practical applications.
2. Describe virtualization technologies and cloud resource management mechanisms.
3. Implement and evaluate big data processing using Hadoop and MapReduce frameworks.
4. Identify and analyze cyber threats and apply basic security controls to mitigate them.
5. Apply cloud security practices including IAM, encryption, and risk assessment in cloud environments.

Course Contents:

UNIT I

Introduction to Cloud Computing:

[08 Hours]

Definition and characteristics of cloud computing, Evolution from traditional computing to cloud, Cloud service models: IaaS (Infrastructure as a Service), PaaS (Platform as a Service), SaaS (Software as a Service), Cloud deployment models: Public, Private, Hybrid, and Community clouds, Benefits and challenges of cloud computing, Use cases and applications.

UNIT II

Cloud Architecture and Virtualization

[08 Hours]

Cloud architecture overview, Virtualization concepts: Hypervisors (Type 1 and Type 2), Virtual machines (VMs), Virtualization in cloud computing and its types, Types of storage

virtualization, Resource provisioning , Load Balancing and Auto-Scaling.

UNIT III

Cloud-Based Big Data Processing (Hadoop & MapReduce): [08 Hours]

Need for big data processing in the cloud, 5 Vs of Big Data: Volume, Variety, Velocity, Veracity, Value, Hadoop Ecosystem: HDFS, YARN, Components of Hadoop: HDFS – Hadoop Distributed File System, NameNode and DataNode architecture, Advantages of Hadoop in the cloud environment, Origin and purpose of MapReduce, Phases of MapReduce: Input Splitting, Mapping, Shuffling and Sorting, Reducing, Output generation.

UNIT IV

Cyber Security: [08 Hours]

Information Security Principles (CIA Triad), Types of Cyber Threats: Malware, Phishing, Ransomware, SQL Injection, DDoS, Attack Vectors and Exploits, Network Security Basics: Firewalls, IDS/IPS, VPN, Security Controls: Physical, Technical, Administrative

UNIT V

Cloud Security and Risk Management: [08 Hours]

Cloud Security Architecture, Identity and Access Management (IAM), Data Protection Techniques: Encryption, Tokenization, Key Management, Security Risks in Cloud Environments (Multi-tenancy, API vulnerabilities), Security in Virtualization and Containers.

Textbooks:

1. Buyya, Rajkumar; Broberg, James; Goscinski, Andrzej M., “*Cloud Computing: Principles and Paradigms*”, Wiley, 2011.
2. Brooks, Charles; Grow, Christopher; Craig, Philip; Short, Donald, “*Cybersecurity Essentials*”, Wiley, 2018.

Reference books:

1. Mather, Tim; Kumaraswamy, Subra; Latif, Shahed, “*Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance*”, O'Reilly Media, 2009.
2. Erl, Thomas; Mahmood, Zaigham; Puttini, Ricardo, “*Cloud Computing: Concepts, Technology & Architecture*”, Prentice Hall / Pearson, 2013.

Course Title : Software Engineering

Semester : V

Prerequisite : Nil

L – T – P : 2-0-0

Course Code : 25UD1246PC504

Credits : 2

Course Type : PCC

Course Objective:

1. Understand the fundamental concepts of software engineering and various process models.
2. Analyze user requirements and apply techniques for effective requirement engineering.
3. Learn system modeling and design using standard UML practices.
4. Apply various software testing strategies and techniques to ensure software reliability.
5. Develop skills for effective software project planning, management, and risk mitigation.

Course Outcomes:

After learning the course, the students should be able:

1. To describe and compare various software process models, and apply requirement engineering techniques to develop functional use-cases.
2. To construct analysis and design models using software design principles, architectural patterns, and standard UML diagrams.
3. To Apply validation and testing strategies, including white-box and black-box techniques, to detect and debug software defects.
4. To apply project planning techniques such as effort estimation, scheduling, and metric-based monitoring for software development.
5. To analyze project risks, implement quality assurance practices, and manage software changes to maintain project quality.

Course Contents:

UNIT I

Software Development Process: [06 Hours]

Software crisis and myths, Software process and development: Generic view of process, Software life cycle and models, Analysis and comparison of various models, an agile view of process, Requirement Engineering: Requirements engineering tasks, Initiating requirement engineering process, Eliciting requirement, developing use-cases.

UNIT II

Building the analysis model, Negotiating and validating requirement, System Design Overview: [06 Hours]

Design process and design quality, Design concepts, Design model, Pattern based software design, Architectural design, User interface design. UML: Different methods: Rumbaugh / Booch / Jacobson, Need for standardization, Developing diagrams in UML (Use CASE, Class, Interaction, State diagrams).

UNIT III

Validation and Testing:

[06 Hours]

Strategic approach to Software testing, Strategic issues, Test strategies for conventional software, Validation testing, System testing, Debugging, White box testing and Black box testing.

UNIT IV

Planning of Project:

[06 Hours]

Project management, Metrics for process and projects, Estimation, Project scheduling.

UNIT V

Management of Project:

[06 Hours]

Risk management, Importance of software quality and measurements, software engineering techniques for quality assurance, and Change management.

Textbooks:

1. Roger S. Pressman, “*Software Engineering*”, Tata McGraw-Hill, 9th Edition, 2020.
2. G. Booch, J. Rumbaugh, and I. Jacobson, “*The Unified Modeling Language User Guide*”, Addison Wesley, 2nd Edition, 2005.

Reference books:

1. Shari Pfleeger, “*Software Engineering*”, Pearson Education, 4th Edition, 2012.
2. Ian Sommerville, “*Software Engineering*”, Pearson Higher Education, 10th Edition, 2021.
3. Pankaj Jalote, “*An Integrated Approach to Software Engineering*”, Springer New York, 2nd Edition, 2012.

Course Title : Information and Cybersecurity

Semester: V

Course Code : 25UD1246PEC505A

L – T – P : 2-0-0

Prerequisite : Computer Networks

Credits : 2

Course Type : PEC

Course Objectives:

1. Understand the principles of cybersecurity including CIA triad and real-world security breaches.
2. Analyze security management frameworks such as GRC and develop appropriate cyber policies.
3. Implement risk management, contingency planning, and disaster recovery strategies.
4. Evaluate cybersecurity threats, technologies, and defense mechanisms in industry.
5. Understand privacy principles, regulations (GDPR, DPDP), and their socio-economic impact.

Course Outcomes:

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

1. Describe and explain the core principles of cybersecurity and real-world security incidents like data breaches.
2. Apply governance, risk, and compliance frameworks to manage and control enterprise cybersecurity policies.
3. Develop contingency plans, incident response strategies, and assess cyber risks within IT environments.
4. Evaluate current industry-level cybersecurity practices and defense mechanisms.
5. Analyze privacy regulations, strategies, and trade-offs between privacy, economics, and security.

Course Contents:

UNIT I

Introduction

[6 Hours]

Introduction to cyber security, Confidentiality, integrity, and availability.

Foundations - Fundamental concepts, CIA, CIA triangle, data breach at target.

Security management, Governance, risk, and compliance (GRC)- GRC framework, security standards.

UNIT II

Cybersecurity Policy, Risk, and Contingency Planning:

[6 Hours]

Contingency planning - Incidence response, Disaster Recovery, BCP.

Cyber security policy - ESSP, ISSP, SYSSP.

Risk Management - Cyber Risk Identification, Assessment, and Control.

UNIT III

Cybersecurity Technologies and Industry Practices: Cyber security:

[6 Hours]

Industry perspective - Defense Technologies, Attack, Exploits.

Cyber security technologies - Access control, Encryption, Standards.

UNIT IV

Foundations of Privacy and Data Protection Regulations:

[6 Hours]

Foundations of privacy - Information privacy, Measurement, Theories.

Privacy regulation - Privacy, Anonymity, Regulation, Data Breach.

UNIT V

Global Privacy Regulations and Strategic Perspectives:

[6 Hours]

Privacy regulation in Europe, Privacy: The Indian Way - Data Protection, GDPR, DPDP, Aadhar.

Information privacy: Economics and strategy, Economic value of privacy, privacy valuation, WTA and WTC, Business strategy and privacy, espionage, Privacy vs safety.

Textbooks:

1. Michael E. Whitman, Herbert J. Mattord, *“Principles of Information Security”*, 6th edition, Cengage Learning, N. Delhi, (2018).
2. Darktrace, *“Technology”* <https://www.darktrace.com/en/technology/#machine-learning>, accessed November 2018.
3. Van Kessel, *“Is cyber security about more than protection?”*, EY Global Information Security Survey 2018-2019.
4. Johnston, A.C. and Warkentin, *“Fear appeals and information security behaviors: An empirical study”* MIS Quarterly, 2010.
5. Arce I. et al., *“Avoiding the top 10 software security design flaws”*, IEEE Computer Society Center for Secure Design (CSD), 2014.

Reference books:

1. Smith, H. J., Dinev, T., & Xu, H. *“Information privacy research: an interdisciplinary review”*, MIS Quarterly, 2011.
2. Subramanian R. *“Security, privacy and politics in India: a historical review”* Journal of Information Systems Security (JISSec), 2010.
3. Acquisti, A., John, L. K., & Loewenstein, *“What is privacy worth?”* The Journal of Legal Studies, 2013
4. Xu H., Luo X.R., Carroll J.M., Rosson M.B. , *“The personalization privacy paradox: An exploratory study of decision making process for location-aware marketing.”*, Decision Support Systems, 2011.

Course Title : Data Science and Visualization

Semester : V

Prerequisite :

L – T – P : 2-0-0

Course Code : 25UD1246PEC505B

Credits : 2

Course Type : PEC

Course Objective:

1. To introduce the fundamental concepts, workflow, and roles in the field of data science.
2. To equip students with techniques for data collection, cleaning, and preprocessing.
3. To develop skills in exploratory data analysis and basic predictive modeling.
4. To familiarize students with visualization principles and tools for effective data communication.
5. To promote ethical awareness and storytelling approaches in real-world data science applications.

Course Outcomes:

After learning the course, the students will be able to:

1. Understand key concepts, roles, and tools in data science.
2. Perform data cleaning, transformation, and feature engineering.
3. Apply EDA techniques and build basic predictive models.
4. Create effective visualizations using Python and BI tools.
5. Communicate insights ethically through data storytelling.

Course Contents:

UNIT I

Fundamentals of Data Science:

[06 Hours]

What is Data Science? Evolution of Data Science and Its Applications, Data Science Workflow: Key Steps and Challenges, Roles in Data Science: Data Engineer, Data Analyst, Data Scientist, Tools and Technologies Overview: Jupyter Notebook, Anaconda, Programming Languages: Python, R, Introduction to Git and GitHub for Version Control, Cloud Platforms: Google Colab (Brief Overview), Types and Sources of Data: Structured, Semi-Structured, Unstructured, Introduction to Public Datasets: Kaggle, UCI ML Repo, WHO, etc.

UNIT II

Data Preparation and Preprocessing:

[06 Hours]

Data Collection Methods, Data Cleaning and Transformation, Handling: Missing Data, Outliers, Duplicates Feature Engineering Techniques, Feature Scaling: Standardization, Normalization, Encoding Categorical Variables: One-Hot Encoding, Label Encoding, Introduction to Pandas and NumPy for Data Manipulation

UNIT III

Exploratory Data Analysis (EDA) and Predictive Modeling: [06 Hours]

Descriptive Statistics, Correlation and Covariance ,Data Summarization and Grouping, Visualizing Distributions: Boxplots ,Histograms, Introduction to Supervised Learning,Linear Regression Logistic Regression k-Nearest Neighbors (k-NN), Basics of Model Validation: Train-Test Split Overfitting and Underfitting, Model Evaluation Metrics: Accuracy, Precision, Recall, F1 Score, Introduction to Unsupervised Learning,Clustering Techniques: K-Means, Hierarchical clustering,Dimensionality Reduction: Principal Component Analysis (PCA)

UNIT IV

Data Visualization Principles and Tools: [06 Hours]

Design Principles: Color, Size, Scale, Chart Types ,Visualization Libraries in Python: Matplotlib ,Seaborn Plotly, Overview of Interactive Dashboards: Tableau, Microsoft Power BI, Case Studies in Exploratory Visualization

UNIT V

Textbooks:

1. Seema Acharya, "*Reimagining Data Visualization using Python*", Wiley Publication,2022.
2. Ben Fry, "*Visualizing Data: Exploring and Explaining data with Processing Environment*", Shroff/O'Reilly Media, 2016.
3. Joel Grus – *Data Science from Scratch: First Principles with Python*, O'Reilly, 2nd Edition,2019.
4. Wes McKinney – *Python for Data Analysis*, O'Reilly Media, 2nd Edition, 2017.
5. Ben Jones – *Communicating Data with Tableau*, O'Reilly, 2014

Reference books:

1. Scott Murray, "*Interactive Data Visualization for the web*", Shroff/O'Reilly Media,2017.
2. Tamara Munzner, "*Visualization Analysis and Design*", CRC Press, 2014.
3. Julia Steele, Noah Lliinsky, "*Designing Data Visualizations*", Shroff/O'Reilly Media,2012.
4. Kyran Dale, "*Data Visualization with Python and JavaScript: Scrape, Clean, Explore & Transform your data*", Shroff/O'Reilly Media, 2022.
5. Julia Steele, Noah Lliinsky, "*Beautiful Visualization*", Shroff/O'Reilly Media, 2016.,
6. Jake VanderPlas – *Python Data Science Handbook*, O'Reilly, 2016
7. John W. Tukey – *Exploratory Data Analysis*, Pearson Education
8. Cole Nussbaumer Knaflic – *Storytelling with Data: A Data Visualization Guide for Business Professionals*, Wiley, 2015
9. Hadley Wickham and Garrett Golemund – *R for Data Science*, O'Reilly, 2017 (for R users)

Course Title : Fundamentals of Hyperspectral Imaging

Semester : V

Prerequisite :

L – T – P : 2-0-0

Course Code : 25UD1246PEC505C

Credits : 2

Course Type : PEC

Course Objective:

1. To understand the underlying physics of hyperspectral image formation.
2. To explore the working principles of hyperspectral sensors and imaging systems.
3. To familiarize students with hyperspectral data representation and cube structures.
4. To study atmospheric and radiometric correction techniques.
5. To examine real-world applications across agriculture, environment, and defense sectors.

Course Outcomes:

After learning the course, the students will be able to:

1. Explain electromagnetic spectrum usage in hyperspectral imaging.
2. Compare and select appropriate hyperspectral imaging sensors and platforms.
3. Preprocess and calibrate raw HSI data for analysis.
4. Interpret data cubes and spectral reflectance curves.
5. Identify and analyze application-specific hyperspectral solutions.

Course Contents:

UNIT I

Introduction to Hyperspectral Imaging: [06 Hours]

Principles of EM spectrum; Comparison: multispectral vs hyperspectral imaging; Spectral resolution; Spectral ranges (VNIR, SWIR); History and evolution of HSI.

UNIT II

Spectral Calibration and Image Cubes: [06 Hours]

Radiometric and geometric calibration; Image cube structure; Spatial-spectral mapping; Sampling; Quantization; File formats; Metadata interpretation

UNIT III

Hyperspectral Sensor Technologies: [06 Hours]

Pushbroom, Whiskbroom, Snapshot sensors; Platforms: UAVs, satellites; AVIRIS, Hyperion, PRISMA characteristics; Sensor performance criteria.

UNIT IV

Atmospheric Effects and Corrections:

[06 Hours]

Atmospheric scattering/absorption; Correction models (FLAASH, QUAC); Solar illumination; Sensor noise; Preprocessing workflows.

UNIT V

Applications of Hyperspectral Imaging:

[06 Hours]

Agriculture: crop stress & yield; Water quality, urban monitoring; Geology: mineral mapping; Disaster monitoring; Case studies (India & Global).

Textbooks:

1. Chein-I Chang, "*Hyperspectral Imaging: Techniques for Spectral Detection and Classification*", Springer, 1st Edition, 2003.
2. S. S. Shenoi, "*Hyperspectral Imaging for Remote Sensing*", McGraw-Hill Education, 1st Edition, 2021.

Reference books:

1. Chein-I Chang, "*Hyperspectral Data Processing: Algorithm Design and Analysis*", Wiley, 1st Edition, 2013.
2. Prasad S. Thenkabail, John G. Lyon, Alfredo Huete (Editors), "*Hyperspectral Remote Sensing of Vegetation*", RC Press (Taylor & Francis), 1st Edition, 2011.

Course Title : Introduction to IoT

Semester : V

Prerequisite :

L – T – P : 2-0-0

Course Code : 25UD1246PEC505D

Credits : 2

Course Type : PEC

Course Outcomes:

1. Understand the various concepts, terminologies and architecture of IoT systems.
2. Use sensors and actuators for the design of IoT.
3. Understand and apply various protocols for design of IoT systems.
4. Use various techniques of data storage and analytics in IoT.
5. Understand various applications of IoT

Course Objective:

After learning the course, the students will be able :

1. To study fundamental concepts of IoT.
2. To understand the roles of sensors in IoT.
3. To Learn different protocols used for IoT design.
4. To be familiar with data handling and analytics tools in IoT.
5. Appreciate the role of big data, cloud computing and data analytics in a typical IoT system.

Course Contents:

UNIT I

Fundamentals of IoT:

[06 Hours]

Introduction, Definitions & Characteristics of IoT, IoT Architectures, Physical & Logical Design of IoT, Enabling Technologies in IoT, History of IoT, About Things in IoT, The Identifiers in IoT, About the Internet in IoT, IoT frameworks, IoT and M2M.

UNIT II

Sensors Networks:

[06 Hours]

Definition, Types of Sensors, Types of Actuators, Examples and Working, IoT Development Boards: Arduino IDE and Board Types, RaspberriPi Development Kit, RFID Principles and components, Wireless Sensor Networks: History and Context, The node, Connecting nodes, Networking Nodes, WSN and IoT.

UNIT III

Wireless Technologies for IoT:

[06 Hours]

WPAN Technologies for IoT: IEEE 802.15.4, Zigbee, HART, NFC, Z-Wave, BLE, Bacnet, Modbus. IP Based Protocols for IoT IPv6, 6LowPAN, RPL, REST, AMPQ, CoAP, MQTT. Edge connectivity and protocols

UNIT IV

Data Handling & Analytics:

[06 Hours]

Introduction, Bigdata, Types of data, Characteristics of Big data, Data handling Technologies, Flow of data, Data acquisition, Data Storage, Introduction to Hadoop. Introduction to data Analytics, Types of Data analytics, Local Analytics, Cloud analytics and applications

UNIT V

Applications of IoT:

[06 Hours]

Home Automation, Smart Cities, Energy, Retail Management, Logistics, Agriculture, Health and Lifestyle, Industrial IoT, Legal challenges, IoT design Ethics, IoT in Environmental Protection.

Textbooks:

1. Hakima Chaouchi, *“The Internet of Things Connecting Objects to the Web”* ISBN : 978-1- 84821-140-7, Wiley Publications
2. Olivier Hersent, David Boswarthick, and Omar Elloumi, — *“The Internet of Things: Key Applications and Protocols”*, Wiley Publications
3. Vijay Madiseti and Arshdeep Bahga, — *“Internet of Things (A Hands-on-Approach)”*, 1st Edition, VPT, 2014.
4. J. Biron and J. Follett, *“Foundational Elements of an IoT Solution”*, O'Reilly Media, 2016.
5. Keysight Technologies, *“The Internet of Things: Enabling Technologies and Solutions for Design and Test”*, Application Note, 2016.

References:

1. Daniel Minoli, — *“Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2M Communications”*, ISBN: 978-1-118-47347-4, Wiley Publications
2. Pethuru Raj and Anupama C. Raman, *“The Internet of Things: Enabling Technologies, Platforms, and Use Cases”*, CRC Press
3. https://onlinecourses.nptel.ac.in/noc17_cs22/course
4. http://www.cse.wustl.edu/~jain/cse570-15/ftp/iot_prot/index.html

Course Title : Advanced Machine Learning

Semester : V

Prerequisite : Machine Learning and Artificial Neural Network

L – T – P : 2-0-0

Course Code : 25UD1246PEC505E

Credits : 2

Course Type : PEC

Course Objective:

1. To introduce probabilistic classifiers, kernel-based SVMs, ensemble methods, and sequential models for advanced supervised learning.
2. To develop understanding of advanced dimensionality reduction techniques and their applications in high-dimensional data analysis.
3. To explore mixture models, clustering methods, and density estimation techniques for complex data distributions.
4. To impart knowledge of feature selection, handling imbalanced datasets, and applying interpretability methods in machine learning.
5. To familiarize students with causal inference, graph-based learning, federated learning, and practical ML deployment strategies.

Course Outcomes:

After learning the course, the students will be able to:

1. Apply Bayesian methods, SVMs, ensemble models, and HMMs to classification and sequence modeling tasks.
2. Implement advanced dimensionality reduction and visualization techniques for high-dimensional datasets.
3. Perform clustering using mixture models, fuzzy, spectral, and density-based methods in varied data scenarios.
4. Select relevant features, address class imbalance, and interpret ML model outputs using state-of-the-art tools.
5. Design ML pipelines, incorporate federated learning, and deploy models in real-world environments.

Course Contents:

UNIT I

[06 Hours]

Bayes optimal classifier, Bayesian Belief Network, Support Vector Machine, Kernel function and Kernel SVM. Ensemble strategies, boosting and bagging, Random forest, Hidden Markov Models (HMM) and applications.

UNIT II

[06 Hours]

Dimensionality Problem, Principal Component Analysis (PCA) and its interpretations, Singular Value Decomposition, Autoencoders and relation to PCA, Advanced dimensionality reduction: t-SNE, UMAP, PCA variants (Kernel PCA, Sparse PCA), Linear Discriminant Analysis (LDA).

UNIT III

[06 Hours]

Concept of mixture model, Gaussian mixture model, Expectation Maximization Algorithm, Recap of Clustering, Fuzzy K-means clustering, Mean-shift clustering, Spectral clustering, DBSCAN, OPTICS, High-dimensional clustering challenges.

UNIT IV

[06 Hours]

Feature selection techniques: filter, wrapper, embedded methods, Handling imbalanced datasets: SMOTE, ADASYN, undersampling & oversampling methods, Cost-sensitive learning and evaluation metrics for imbalance (ROC-AUC, PR curve, Cohen's kappa, Matthews correlation coefficient), Model interpretability: LIME, SHAP, permutation importance.

UNIT V

Textbooks:

1. Christopher M. Bishop, "*Pattern Recognition and Machine Learning*," Springer.
2. Trevor Hastie, Robert Tibshirani, Jerome Friedman, "*The Elements of Statistical Learning*," Springer.
3. Kevin P. Murphy, "*Machine Learning: A Probabilistic Perspective*," MIT Press.

References:

1. Mehryar Mohri, Afshin Rostamizadeh, Ameet Talwalkar, Foundations of Machine Learning, MIT Press.
2. Ian H. Witten, Eibe Frank, Mark A. Hall, Data Mining: Practical Machine Learning Tools and Techniques, Morgan Kaufmann.
3. Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (Scikit-learn parts only), O'Reilly.
4. Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, An Introduction to Statistical Learning, Springer.
5. Charu C. Aggarwal, Machine Learning for Text, Springer (for graph-based and advanced ML concepts).

Course Title : Virtualization and Containers

Semester : V

Prerequisite :

L – T – P : 2-0-0

Course Code : 25UD1246PEC505G

Credits : 2

Course Type : PEC

Course Objective:

1. To introduce students to virtualization concepts and techniques in computing environments.
2. To explore the architecture and components of virtual machines and hypervisors.
3. To understand the evolution and working of container technologies like Docker.
4. To examine the role of container orchestration using Kubernetes.
5. To enable practical skills in deploying, managing, and scaling virtualized and containerized systems.

Course Outcomes:

After learning the course, the students will be able to:

1. Describe and differentiate types of virtualization and container technologies.
2. Configure and manage virtual machines using various hypervisors.
3. Deploy containerized applications using Docker and Docker Compose.
4. Implement container orchestration using Kubernetes and analyze performance.
5. Evaluate use cases and challenges of virtualization and containers in industry.

Course Contents:

UNIT I

Introduction to Virtualization :

[06 Hours]

History and evolution of virtualization, Benefits and challenges of virtualization, Types of virtualization: Full, Para, Hardware-assisted, Virtual Machine Monitor (VMM) and Hypervisors (Type 1 & Type 2), Virtual machine architecture

UNIT II

Virtualization Technologies:

[06 Hours]

Hypervisors: VMware, Microsoft Hyper-V, KVM, Xen; VM lifecycle: creation, snapshots, cloning, migration; Storage and network virtualization; Virtualization in cloud platforms (AWS EC2, Azure VMs), Performance monitoring of VMs

UNIT III

Introduction to Containers:

[06 Hours]

Concept and need for containerization, Containers vs Virtual Machines, Docker architecture and components, Docker images, containers, volumes, and networking, Docker Compose for multi-container applications

UNIT IV

Container Orchestration :

[06 Hours]

Introduction to Kubernetes, Kubernetes architecture: master and node components; Pods, ReplicaSets, Deployments, Services, Helm charts and configuration management, Monitoring and scaling containers

UNIT V

Virtualization & Containers in Practice:

[06 Hours]

Case studies: Netflix, Google, Spotify; Container security and sandboxing; Resource allocation and scheduling; CI/CD integration with Docker and Kubernetes, Future trends: Serverless, MicroVMs, unikernels

Textbooks:

1. Matthew Portnoy, "*Virtualization Essentials*", Wiley, 2016.
2. Nigel Poulton, "*The Docker Book: Containerization is the new virtualization*", Leanpub, Latest Edition 2021

Reference books:

1. Kelsey Hightower, Brendan Burns, Joe Beda, "*Kubernetes: Up and Running*", O'Reilly Media, 2nd Edition, 2019

Course Title : DevSecOps Principles and Practices

Semester : V

Prerequisite :

L – T – P : 2-0-0

Course Code : 25UD1246PEC505H

Credits : 2

Course Type : PEC

Course Objectives::

1. Understand the core principles and philosophy of DevSecOps.
2. Explain the role of CI/CD (Continuous Integration/Continuous Delivery) in modern software development.
3. Gain hands-on experience with key DevSecOps tools for version control, automation, and containerization.
4. Develop a foundational understanding of infrastructure as code (IaC) and its benefits.
5. Demonstrate the ability to implement a basic DevSecOps pipeline for a software project.

Course Outcomes:

After learning the course, the students will be able to:

1. Articulate the differences between traditional software development and the DevSecOps methodology, and explain the benefits of adopting a DevSecOps culture.
2. Apply version control principles using Git and perform essential operations like branching, merging, and collaboration.
3. Configure and utilize a CI/CD tool like Jenkins to automate the build, test, and deployment of a simple application.
4. Implement and manage containerized applications using Docker and understand the basics of container orchestration.
5. Describe and apply the concept of Infrastructure as Code using a tool like Terraform to provision and manage cloud resources.

Course Contents:

UNIT I

Introduction to DevSecOps

[06 Hours]

What is DevSecOps?: Definition, history, and core principles. The three ways of DevSecOps: Flow, Feedback, and Continuous Learning. DevSecOps Culture: Bridging the gap between Dev and Ops, cross-functional teams, and the importance of collaboration and communication. DevSecOps vs. Agile: Understanding the relationship and differences between Agile and DevSecOps methodologies.

UNIT II

Version Control System (VCS)

[06 Hours]

Introduction to Git: Core concepts of Git, local vs. remote repositories. Git Operations: Committing changes, branching, merging, and handling conflicts. Remote Repositories: Working with platforms like GitHub or GitLab to collaborate on projects.

UNIT III

CI/CD& Continuous Deployment

[06 Hours]

CI Concepts: What is CI? Its importance in the software development lifecycle. CI Tools: Introduction to Jenkins. Installing and configuring Jenkins. Building a Pipeline: Creating a basic CI pipeline to automate the build and testing of an application. CD Concepts: The difference between Continuous Delivery and Continuous Deployment. Pipeline Stages: Extending the CI pipeline to include stages for automated testing and deployment. Artifact Management: Storing and managing build artifacts.

UNIT IV

Containerization & Orchestration

[06 Hours]

Introduction to Docker: Understanding containers, Docker images, and Dockerfiles. Docker Commands: Creating and managing Docker images and containers. Docker Compose: Defining and running multi-container applications. Introduction to Kubernetes: Basic concepts of container orchestration and Kubernetes architecture (brief overview).

UNIT V

Infrastructure as Code (IaC)

[06 Hours]

IaC Principles: Why IaC is important. Managing infrastructure through code. IaC Tools: Introduction to Terraform. Terraform Concepts: Writing basic Terraform configurations to provision cloud resources (e.g., virtual machines).

Textbooks:

1. Kim, G., Humble, J., Debois, P., & Willis, J. (2021). *The DevSecOps handbook: How to create world-class agility, reliability, and security in technology organizations* (2nd ed.). IT Revolution Press.
2. Humble, J., & Farley, D. (2010). *Continuous delivery: Reliable software releases through build, test, and deployment automation* (1st ed.). Addison-Wesley.
3. Kim, G., Behr, K., & Spafford, G. (2018). *The Phoenix project: A novel about IT, DevSecOps, and helping your business win* (5th anniversary ed.). IT Revolution Press.
4. Poulton, N. (2020). *Docker deep dive* (4th ed.). Leanpub.
5. Brikman, Y. (2022). *Terraform: Up & running: Writing infrastructure as code* (3rd ed.). O'Reilly Media.

Course Title : Introduction to Quantum Computing

Semester : V

Prerequisite : Mathematics

L – T – P : 2-0-0

Course Code : 25UD1246PEC505I

Credits : 2

Course Type : PEC

Course Objectives:

1. To understand the fundamentals of quantum computing and compare classical and quantum computation models.
2. To explain the concept of qubits, their physical implementations, and the role of quantum gates in computation.
3. To analyze quantum entanglement, Bell’s theorems, and their implications for quantum information processing.
4. To Study key quantum algorithms such as Deutsch–Jozsa, Grover’s, and Shor’s, and understand their computational advantages.
5. To Explore quantum complexity classes and recent quantum algorithms relevant to optimization, simulation, and linear systems.

Course Outcomes:

After learning the course, the students will be able to:

1. Explain the fundamental concepts of quantum computing, including qubits, quantum gates, quantum hardware platforms, and quantum advantage.
2. Understand quantum correlations, entanglement, quantum logic circuits.
3. Understand quantum algorithms such as Deutsch, Deutsch–Jozsa, Bernstein–Vazirani, Simon’s, and Grover’s algorithm.
4. Understand advanced quantum algorithms like Shor’s, VQE, QAO
5. Evaluate quantum error correction, fault tolerance, and emerging technologies such as NISQ processors.

Course Contents:

UNIT I

[06 Hours]

Introduction to Quantum Computing, Applications, Quantum Computing Basics. Single and Two qubit gates – Solovay - Kitaev Theorem. Quantum advantage claims.

UNIT II

[06 Hours]

Quantum correlations: Entanglement and Bell's theorems, Universal quantum logic gates and circuits, Quantum parallelism

UNIT III

[06 Hours]

Quantum algorithms: Deutsch algorithm, Deutsch Josza algorithm, Bernstein - Vazirani algorithm, Simon's algorithm. Database search: Grover's algorithm.

UNIT IV

[06 Hours]

Additional Quantum Algorithms: Variational Quantum Eigensolver (VQE), HHL, QAOA. Quantum Fourier Transform and prime factorization: Shor's Algorithm.

UNIT V

[06 Hours]

Error correction: Fault-tolerance, Simple error correcting codes, Survey of current status , NISQ era processors.

Text Books:

1. M. A. Nielsen and I. Chuang, "*Quantum Computation and Quantum Information*", Cambridge University Press (2010).
2. Arthur O. Pittenger, "*An Introduction to Quantum Computing Algorithms*", Birkhäuser, 1st Edition, 2000.
3. Qiskit Textbook: <https://qiskit.org/textbook/preface.html>

Reference Books:

1. Michael A. Nielsen and Isaac L. Chuang, "*Quantum Computation and Quantum Information*", Cambridge University Press, 2010.
2. Chris Bernhardt, "*Quantum Computing for Everyone*", MIT Press, 2019.

Course Title : Human-Computer Interaction

Semester : V

Prerequisite :

L – T – P : 2-0-0

Course Code : 25UD1246PEC505J

Credits : 2

Course Type : PEC

Course Objective:

1. Understand the historical evolution and foundations of Human-Computer Interaction as a multidisciplinary field.
2. Explore the human, computer, and interaction components, and how they influence interface design.
3. Apply principles of interaction design, usability, and HCI within the software development process.
4. Evaluate user interfaces through implementation methods and usability testing techniques.
5. Understand cognitive and communication models, as well as task analysis and dialog design.
6. Explore modern systems including groupware, ubiquitous computing, augmented reality, multimedia, and the World Wide Web.

Course Outcomes:

After learning the course, the students should be able:

1. To describe the objectives, evolution, and key elements of Human-Computer Interaction.
2. To apply interaction design principles and usability concepts in the software development process.
3. To explain implementation support and apply evaluation techniques and universal design principles in user-centered systems.
4. To apply cognitive and communication models, and analyze stakeholder needs to design effective human-computer interactions.
5. To describe modern HCI systems including groupware, ubiquitous computing, AR, hypertext, multimedia, and the web.

Course Contents:

UNIT I

[06 Hours]

Introduction: Course objectives and overview, Historical evolution of the field, The Human, The Computer, The Interaction.

UNIT II

[06 Hours]

Design processes: Interaction Design basics, Concept of usability – definition and elaboration, HCI in the Software Process, Design Rules.

UNIT III

[06 Hours]

Implementation and Evaluation: Implementation Support, Evaluation Techniques, Universal Design, User Support.

UNIT IV

[06 Hours]

Models: Cognitive Models, Socio-Organizational Issues and Stakeholders Requirements, Communication and Collaboration models. Theories: Task Analysis Dialog notations and Design Models of the system Modelling Rich Interactions.

UNIT V

[06 Hours]

Modern Systems: Group ware, Ubiquitous Computing and Augmented Realities, Hypertext, Multimedia and World Wide Web.

Textbooks:

1. Alan Dix, Janet Finlay, "***Human Computer Interaction***", Pearson Education, 3rd edition, 2009
2. Ben Shneiderman, "***Designing the User Interface - Strategies for Effective Human Computer Interaction***", Pearson Education, 2016.

Reference books:

1. M. B. Rosson, J. M. Carroll, "***Usability Engineering: Scenario-Based Development of Human-Computer Interaction***", Elsevier, 2002.
2. Alan Cooper, "***The Essentials of Interaction Design***", Wiley Publishing, 2014.
3. Nielsen, J. Morgan Kaufmann, San Francisco, "***Usability Engineering***", 1993.
4. Heim, S., "***The Resonant Interface: HCI Foundations for Interaction Design***", Addison-Wesley, 2007.

Course Title : Software Project Management

Semester : V

Prerequisite :

L – T – P : 2-0-0

Course Code : 25UD1246PEC505K

Credits : 2

Course Type : PEC

Course Objectives

1. To gain knowledge about the concepts and methods required for construction of large software intensive system
2. To gain knowledge on the principles and techniques of Software Project Management.
3. To gain knowledge about organizational behavior and general Management techniques used for Project Management.

Course Outcomes

After learning the course, the students will be able to:

1. To apply the process to be followed in the software development life-cycle models.
2. To understand approaches for managing and optimizing the software development process.
3. To explain the quality management and different types of metrics used in software Development
4. To do the Project scheduling, Tracking, Risk Analysis, Quality Management and Project cost estimation using different techniques and tools.

UNIT I

Project Evaluation and Planning:

[06 Hours]

Activities in Software Project Management, Overview of Project Planning, Stepwise planning, Software processes and process models, Cost Benefit Analysis, Cash Flow Forecasting, Cost-Benefit Evaluation Techniques, Risk Evaluation. Software effort estimation, Activity Planning, Risk Management, Resource Allocation.

UNIT II

Monitoring and Control:

[06 Hours]

Collecting Data, Visualizing Progress, Cost Monitoring, Review techniques, Project termination review, Earned Value analysis, Change Control, Software Configuration Management (SCM).

UNIT III

Managing Contracts:**[06 Hours]**

Types of Contracts, Stages in Contract Placement, Typical Terms of a Contract, Contract Management and Acceptance.

UNIT IV**Quality Management and People Management:****[06 Hours]**

Introduction, Understanding Behavior, Organizational Behavior, Selecting The Right Person for the Job, Motivation, The Oldman – Hackman Job Characteristics Model, Working in Groups, Organization and team structures, Decision Making, Leadership, Organizational Structures, Stress, Health and Safety. ISO and CMMI models, Testing, and Software reliability, Test automation.

UNIT V**Overview of Project Management Tools.****Text Book:**

1. Bob Hughes, Mike Cotterell, *“Software Project Management”*, Tata McGraw Hill, 6thEdition, 2017.

Reference Books:

1. Wakker Royce, *“Software Project Management”*, Pearson Education, 2002.
2. Robert K. Wysocki, *“Effective Software Project Management”*, Wiley publication, 2006.

Course Title : Network Management

Semester : V

Prerequisite : Computer Networks

L – T – P : 2-0-0

Course Code : 25UD1246PEC505L

Credits : 2

Course Type : PEC

Course Objectives

1. introduce the fundamental concepts, components, and architecture of network management systems.
2. Familiarize students with the OSI network management model, including FCAPS functions and related organizational and information models.
3. Provide a detailed understanding of SNMP protocols, versions, and associated MIB structures for effective network monitoring and control.
4. Explore the architecture and standards of Telecommunication Management Networks (TMN) and their implementation aspects.
5. Develop knowledge of network management tools, security threats, and modern security management techniques to ensure network integrity and performance.

Course Outcomes

After learning the course, the students will be able to:

1. Explain the components, types, goals, and functions of network management, along with relevant standards and architectures.
2. Analyze the OSI network management model and apply FCAPS functions to practical network scenarios.
3. Demonstrate the working of SNMP and its versions, interpret MIB structures, and implement monitoring with RMON.
4. Evaluate TMN architecture, services, and integration approaches in telecommunication networks.
5. Apply network management tools and security techniques to detect, prevent, and mitigate network threats.

UNIT I

Introduction to Network Management:

[6 Hours]

Components of a Network, Network Types: LAN, MAN, WAN, PAN, OSI and TCP/IP Models: Layers and Functions, Transmission Media: Wired and Wireless, Definition and Goals of Network Management, Network Management Functions: FCAPS (Fault, Configuration, Accounting, Performance, Security), Current Status and future of Network Management, Network Management Standards: ISO, TMN, IEEE, ITU, Network Management System (NMS) Components and Architecture, Standard protocols.

UNIT II

OSI Network Management Model: [6 Hours]

Functional Model (FCAPS): Fault Management, Configuration Management, Accounting Management, Performance Management, Security Management; Organizational Model: Roles of users, administrators, and network elements; Information Model: Structure of Management Information (SMI), Management Information Base (MIB); Communication Model: Manager-agent communication, Message exchanges and protocol operations.

UNIT III

SNMP (Simple Network Management Protocol): [6 Hours]

SNMP Overview and Architecture, SNMP Protocol Operations: GET, SET, GET-NEXT, TRAP, INFORM; SNMP Versions: SNMPv1, SNMPv2c, SNMPv3 (Authentication & Encryption), Structure of Management Information (SMI), Management Information Base (MIB) – Concepts and Examples, SNMP Agent and Manager Communication; Remote Monitoring (RMON): Introduction and Motivation, RMON1 and RMON2: Differences.

UNIT IV

TMN – Telecommunication Management Network: [6 Hours]

Telecommunication Management Network: Why TMN? Operations Systems, TMN Conceptual Model, TMN Standards, TMN Architecture, TMN Management Service Architecture, TMN Integrated View, Implementation.

UNIT V

Network Management Tools and Security Management: [6 Hours]

Network Management Tools, Network Statistics Measurement Systems, Network Management systems, Commercial Network Management Systems, SNMP Tools, Network Security Management, Network security threats and risk management, Security management techniques: encryption, authentication, firewalls.

Text Book:

1. Mani Subrahmanian, “*Network Management Principles and Practice*”, Pearson Education, 2nd Edition, 2010.

Reference Books:

1. Morris, “*Network Management*”, Pearson Education, 1st Edition, 2008
2. Mark Burges, “*Principles of Network System Administration*”, Wiley DreamTech, 1st Edition, 2008.

Course Title : Finance Engineering

Semester : V

Prerequisite :

L – T – P : 2-0-0

Course Code : 25UD1246PEC505M

Credits : 2

Course Type : PEC

Course Objectives

1. Develop a strong mathematical foundation for financial engineering, focusing on matrix operations, eigenvalues, covariance, and orthogonal transformations.
2. Understand and apply quadratic programming (QP) techniques to solve optimization problems in finance under various constraints.
3. Analyze and implement factor models, including CAPM, Arbitrage Pricing Theory, and multi-factor models, for asset pricing and portfolio construction.
4. Apply Principal Component Analysis (PCA) and related dimensionality reduction methods to financial datasets for risk and factor analysis.
5. Integrate advanced optimization and statistical modeling techniques to design, evaluate, and optimize portfolios under realistic market conditions.

Course Outcomes

On completion of this course, students will be able to:

1. Perform advanced numerical computations and statistical analysis for financial datasets using vectors, matrices, and decomposition techniques.
2. Formulate and solve convex and non-convex quadratic programming problems relevant to portfolio optimization and risk management.
3. Apply and interpret single-factor, multi-factor, and smart beta models for investment decision-making.
4. Implement PCA and dynamic factor models to extract latent factors and improve portfolio construction.
5. Design, optimize, and evaluate portfolios using mean-variance, risk parity, transaction cost optimization, and index tracking strategies.

UNIT I

Mathematics for finance:

[6 Hours]

Vectors & Matrices Operations, Matrix Inversion and Pseudoinverse, Linear Systems and Solvers, Eigenvalues and Eigenvectors, Covariance and Correlation, Covariance/correlation matrix decomposition, Orthogonal transformations, Numerical Methods in Finance

UNIT II

Quadratic Programming and Its Applications in Finance:

[6 Hours]

Definition of QP, Types of QP :Convex vs. Non-convex, Equality and inequality constraints, Geometry of QP: contours of quadratic functions and feasible region, Convexity and optimality conditions, Robust Optimization using QP, Quadratically Constrained QP (QCQP), QP with Integer Constraints (Mixed Integer QP), Penalty Functions & Relaxations

UNIT III

Factor Models and Factor Analysis in Finance:

[6 Hours]

What is a factor, Single-factor vs. multi-factor models, Macroeconomic vs. statistical factors, Arbitrage Pricing Theory (APT), CAPM and Fama-French 3/5 Factor Models, Factor-based investing, Smart beta and risk premia strategies, Matrix representations of factor models, Common factor model: $R = \Lambda F + \epsilon R$, Covariance decomposition.

UNIT IV

Principal Component Analysis (PCA) in Finance:

[6 Hours]

PCA for dimensionality reduction, Scree plots and eigenvalue selection, Interpreting principal components, Extracting latent factors from asset returns, Building minimum-variance portfolios, Volatility and yield curve modeling Whitening and standardization, Dynamic Factor Models, Bayesian factor models, Shrinkage methods for large portfolios

UNIT V

Applications Portfolio Optimization:

[6 Hours]

Markowitz Mean-Variance Portfolio Optimization, Minimum Variance Portfolio, Risk Parity Portfolio Construction, Transaction Cost Optimization, Smart Beta Strategy Allocation, Tracking a Market Index with Fewer Stocks

Textbooks

1. Frank J. Fabozzi, Petter N. Kolm, Dessislava A. Pachamanova, and Sergio M. Focardi, "*Robust Portfolio Optimization and Management*", Wiley Publication, 2007.
2. Markowitz, Harry M., "*Portfolio Selection: Efficient Diversification of Investments*", Wiley Publication, 1991.

3. Attilio Meucci, *"Risk and Asset Allocation"*, Springer, 2005.
4. Marcos López de Prado, *"Advances in Financial Machine Learning"*, Wiley Publication, 2018.
5. Rupak Chatterjee, *"Practical Methods of Financial Engineering and Risk Management"*, CRC Press, 2014.

References

1. David Ruppert and David S. Matteson, *"Statistics and Data Analysis for Financial Engineering"*, Springer, 2015.
2. Roncalli, Thierry, *"Introduction to Risk Parity and Budgeting"*, CRC Press, 2013.
3. Marcos López de Prado, *"Machine Learning for Asset Managers"*, Cambridge University Press, 2020.
4. Yves Hilpisch, *"Python for Finance: Mastering Data-Driven Finance"*, O'Reilly Media, 2018.
5. Giuseppe A. Paleologo, *"Risk-Based and Factor Investing"*, Elsevier, 2017.

Course Title : Database Management Systems Lab

Semester : V

Prerequisite :

L – T – P : 0-0-2

Course Code : 25UD1246PCL508

Credits : 1

Course Type : PCC

List of Experiments:

1. **Database Creation and SQL Commands:** Create databases and practice the use of SQL commands including DDL (Data Definition Language), DML (Data Manipulation Language), and DCL (Data Control Language).

2. **Basic SQL Operations:** Perform SQL operations such as INSERT, UPDATE, and DELETE on relational tables through a set of suitable exercises.

3. **SQL with Exception Handling:** Develop a SQL procedure for an application that demonstrates exception handling mechanisms.

4. **Using Cursors in SQL:** Write a SQL procedure that utilizes cursors for row-by-row processing of query results.

5. **Nested Queries:** Implement complex SQL queries involving subqueries and nested operations.

6. **Join Operations:** Demonstrate different types of joins (INNER, OUTER, SELF, etc.) using appropriate SQL queries.

7. **Application Report Generation using Functions:** Write a DBMS program to generate reports using built-in and user-defined SQL functions.

9. **Triggers in SQL:** Create SQL blocks that include the use of triggers for automatic execution of actions in response to specific database events.

10. **Stored Procedures:** Develop and execute stored procedures to encapsulate complex business logic.

Database Project (Mini-Application): Design and develop a menu-driven, GUI-based database application in a selected domain such as Banking, Billing, Library Management, Payroll, Insurance, Inventory, or Healthcare. The application must integrate all the key features and concepts learned in the previous experiments.

**Course Title : Machine Learning and Artificial Neural Network
Lab**

Semester : V

Prerequisite :

L – T – P : 0-0-2

Course Code : 25UD1246PCL509

Credits : 1

Course Type : PCC

List of Experiments:

1. Python basics: NumPy, SciPy, Matplotlib – data loading and visualization.
2. Implement k-fold cross-validation and bias-variance analysis.
3. Linear regression and logistic regression with gradient descent.
4. Decision tree classifier and performance evaluation.
5. Naïve Bayes classifier on text classification dataset.
6. k-means clustering and hierarchical clustering.
7. Implement perceptron learning rule and XOR problem.
8. Activation function visualizations (Step, Sigmoid, Tanh, ReLU, Softmax).
9. Implement backpropagation for a simple feedforward network from scratch.
10. Train an MLP with dropout, batch normalization, and early stopping.

Course Title : Software Engineering Lab

Semester : V

Prerequisite :

L – T – P : 0-0-2

Course Code : 25UD1246PCL510

Credits : 1

Course Type : PCC

List of Experiments:

1. Prepare detailed statement of problem for the selected / allotted mini project and identify suitable process model for the same with justification.
2. Develop Software Requirement Specification (SRS) document in IEEE format for the project.
3. Use project management tool to prepare schedule for the project.
4. Analyze the risk related to the project and prepare RMMM plan for the project.
5. Identify scenarios & develop UML Use case and class Diagram for the project.
6. Draw DFD (upto 2 levels) and prepare Data Dictionary for the project.
7. Develop Activity / State Transition diagram for the project.
8. Develop Sequence and Collaboration diagram for the project.
9. Change specification and make different versions using any SCM Tool.
10. Develop test cases for the project using white box testing
11. Mini Project Report on Design and Development of selected / allotted mini project using Software Engineering Principles

Course Title : Information and Cybersecurity Lab

Semester : V

Prerequisite :

L – T – P : 0-0-2

Course Code : 25UD1246PECL511A

Credits : 1

Course Type : PEC

List of Experiments :

1. Password Cracking and Strength Analysis Test password security using tools like John the Ripper or Hashcat and recommend strong password policies.
2. File Encryption and Decryption Implement symmetric (AES) and asymmetric (RSA) encryption for secure file transmission.
3. Digital Signature Implementation Develop a program to generate and verify digital signatures for document integrity.
4. Network Packet Capture and Analysis Use Wireshark or tcpdump to analyze live network traffic and detect suspicious activities.
5. Firewall Configuration and Testing Configure a host-based or network firewall and test its effectiveness against simulated attacks.
6. Vulnerability Scanning Perform vulnerability scanning using OpenVAS/Nessus and prepare a security assessment report.
7. Intrusion Detection System (IDS) Setup Deploy Snort or Suricata IDS and analyze alerts for possible intrusions.
8. Steganography and Steganalysis Hide information inside an image/audio file and attempt detection/removal of hidden content.
9. Malware Analysis Basics Analyze a benign malware sample in a sandbox environment and document findings.
10. Web Application Security Testing Perform SQL Injection and XSS vulnerability testing on a sample web application using OWASP ZAP or Burp Suite.
11. Email Security & Phishing Detection Simulate a phishing attack and demonstrate detection methods, including email header analysis.
12. Incident Response Simulation Investigate a simulated cyber incident, collect digital evidence, and prepare an incident response report.

Course Title : Data Science and Visualization Lab

Semester : V

Prerequisite :

L – T – P : 0-0-2

Course Code : 25UD1246PECL511B

Credits : 1

Course Type : PEC

List of Experiments :

1. Introduction to Jupyter Notebook and Python Basics
Setup Jupyter using Anaconda; write simple Python programs for data types, loops, functions.
2. Working with Public Datasets
Load and explore datasets from Kaggle/UCI using Pandas.
3. Data Cleaning and Handling Missing Values
Perform null value handling, outlier removal, and duplicate detection on real datasets.
4. Data Transformation and Feature Engineering
Apply standardization, normalization, and encoding techniques using Scikit-learn and Pandas.
5. Exploratory Data Analysis (EDA) – Descriptive Statistics
Generate summary statistics, correlation matrix, and group-wise aggregations.
6. Data Visualization – Univariate Analysis
Create histograms, boxplots, and bar plots using Matplotlib and Seaborn.
7. Data Visualization – Multivariate Analysis
Create scatter plots, heatmaps, pair plots, and visualize relationships between variables.
8. Supervised Learning – Linear Regression Model
Implement linear regression for a simple prediction task and evaluate model performance.
9. Supervised Learning – Logistic Regression & k-NN
Build classification models using logistic regression and k-NN; evaluate using accuracy, precision, recall.
10. Unsupervised Learning – k-Means Clustering & PCA
Perform k-Means clustering and visualize cluster assignments; apply PCA for dimensionality reduction and visualize principal components.
11. Interactive Dashboards with Plotly/Streamli Create a simple web-based dashboard using Plotly/Streamlit to display dynamic visualizations.
12. Data Storytelling and Infographic Creation Create visual reports or infographics summarizing dataset insights using Power BI or Tableau.
13. Mini Project: End-to-End Data Science Workflow Select a dataset, perform preprocessing, EDA, model building (supervised or unsupervised), visualization, and present findings.

**Course Title : Hyperspectral Data Acquisition & Preprocessing
Lab**

Semester : V

Prerequisite :

L – T – P : 0-0-2

Course Code : 25UD1246PECL511C

Credits : 1

Course Type : PEC

List of Experiments:

1. Acquire hyperspectral image cubes of a given scene using a laboratory hyperspectral camera setup.
2. Perform dark current correction on raw hyperspectral data.
3. Apply flat-field correction to remove sensor and illumination non-uniformities.
4. Conduct radiometric calibration to convert raw digital numbers to radiance values.
5. Perform geometric correction to align hyperspectral bands spatially.
6. Remove atmospheric effects from airborne hyperspectral imagery using appropriate models.
7. Implement band-to-band registration for hyperspectral datasets from different sensors.
8. Crop and extract a region of interest (ROI) from hyperspectral data for further analysis.
9. Apply spectral resampling to match hyperspectral data to a given sensor's band configuration.
10. Remove noisy or bad bands from hyperspectral datasets using statistical criteria.
11. Normalize hyperspectral data for illumination variation correction.
12. Prepare preprocessed hyperspectral data for classification or target detection workflows.

Course Title : Introduction to IoT Lab

Semester : V

Prerequisite :

L – T – P : 0-0-2

Course Code : 25UD1246PECL511D

Credits : 1

Course Type : PEC

List of Experiments

Experiments using ESP32 / Arduino

Serial Monitor, LED, Servo Motor – Controlling:

1) Controlling actuators through Serial Monitor. Creating different led patterns and controlling them using push button switches. Controlling servo motor with the help of joystick.

Distance Measurement of an object:

2) Calculate the distance to an object with the help of an ultrasonic sensor and display it on an LCD LDR Sensor, Alarm and temperature, humidity measurement:

3) (a)Controlling relay state based on ambient light levels using LDR sensor.

(b)Basic Burglar alarm security system with the help of PIR sensor and buzzer.

(c)Displaying humidity and temperature values on LCD

Experiments using Raspberry Pi / Arduino

4 (a)Controlling relay state based on input from IR sensors

(b)Interfacing stepper motor with R-Pi

(c)Advanced burglar alarm security system with the help of PIR sensor, buzzer and keypad. (Alarm gets disabled if correct keypad password is entered)

(d)Automated LED light control based on input from PIR (to detect if people are present) and LDR (ambient light level)

IOT Framework:

5) Upload humidity & temperature data to Thing Speak, periodically logging ambient light level to Thing Speak

6) Controlling LEDs, relay & buzzer using Blynk app

HTTP Based:

7) Introduction to HTTP. Hosting a basic server from the ESP32 to control various digital based actuators (led, buzzer, relay) from a simple web page.

8) Displaying various sensor readings on a simple web page hosted on the ESP32

MQTT Based:

9) Controlling LEDs/Motors from an Android/Web app, Controlling AC Appliances from an android/web app with the help of relay.

10) Displaying humidity and temperature data on a web-based application

UAV / Drone:

11) (a) Demonstration of UAV elements, Flight Controller

(b) Mission Planner flight planning design

12) Python program to read GPS coordinates from Flight Controller

Course Title : Advanced Machine Learning Lab

Semester : V

Prerequisite :

L – T – P : 0-0-2

Course Code : 25UD1246PECL511E

Credits : 1

Course Type : PEC

List of Experiments

1. Study and implement Bayes Optimal Classifier for binary classification.
2. Study and implement SVM with linear, RBF, and polynomial kernels
3. Study and implement PCA with variance analysis and scree plots.
4. Study and implement Linear Discriminant Analysis (LDA) for classification tasks.
5. Study and implement Fuzzy K-means clustering.
6. Study and implement DBSCAN and OPTICS clustering methods.
7. Study and implement LIME for local model interpretability.
8. Study and implement SHAP for global feature importance
9. Study and implement SMOTE, ADASYN, and undersampling for imbalanced data.

Course Title : Introduction to Blockchain Lab

Semester : V

Prerequisite :

L – T – P :0-0-2

Course Code : 25UD1246PECL511F

Credits : 1

Course Type : PEC

List of Experiments:

3. To implement a simple blockchain in Python to record and validate transactions.
4. To create and verify a cryptographic hash using SHA-256.
5. To implement digital signature generation and verification for a blockchain transaction.
6. To simulate a Proof-of-Work consensus mechanism.
7. To write and deploy a basic smart contract using Solidity on Ethereum test network.
8. To create a simple cryptocurrency token using the ERC-20 standard.
9. To implement a Merkle tree and demonstrate proof of inclusion.
10. To simulate a peer-to-peer network for blockchain data sharing.
11. To develop a blockchain-based voting system prototype.
12. To analyze blockchain transaction data from Bitcoin/Ethereum explorers.

Course Title : Virtualization and Containers Lab

Semester : V

Prerequisite :

L – T – P : 0-0-2

Course Code : 25UD1246PECL511G

Credits : 1

Course Type : PEC

List of Experiments:

1. Introduction to Virtualization

- a) Install and configure a Type-2 hypervisor (VMware Workstation / VirtualBox)
- b) Create, configure, and run multiple virtual machines
- c) Practice VM snapshots, cloning, and resource allocation

2. Server Virtualization

- a) Install a Linux server VM and configure network bridging/NAT
- b) Host multiple services (e.g., web server, database server) on separate VMs
- c) Measure resource usage and performance differences

3. Type-1 Hypervisor Exploration

- a. Explore VMware ESXi / Microsoft Hyper-V
- b. Create and manage VMs using a Type-1 hypervisor interface
- c. Compare performance with Type-2 hypervisors

4. Introduction to Containers

- a. Install Docker on Linux
- b. Run a “Hello World” container and basic Docker commands (**pull, run, ps, stop, rm**)
- c. Understand image layers and container lifecycle

5. Docker Image Creation

- a) Create a custom Docker image using a Dockerfile
- b) Push the image to Docker Hub / private registry
- c) Deploy containers from the custom image

6. Multi-Container Applications

- a. Use Docker Compose to deploy a multi-service application (e.g., WordPress + MySQL)
- b. Scale services up and down using **docker-compose**

7. Container Networking

- a. Configure container-to-container communication
- b. Expose container ports to the host and external networks
- c. Test connectivity and isolation between containers

8. Container Storage

- a. Use Docker volumes for persistent storage
- b. Demonstrate data persistence after container removal

9. Introduction to Kubernetes

- a. Install Minikube or use Kubernetes on cloud (GCP/AWS/Azure)
- b. Deploy a simple application in Kubernetes using `kubectl`
- c. Scale pods and manage deployments

Course Title : DevOps Principles and Practices Lab

Semester : V

Prerequisite :

L – T – P : 0-0-2

Course Code : 25UD1246PECL511H

Credits : 1

Course Type : PEC

List of Experiments:

1. Exploring Git Commands: Setting up a local Git repository, performing basic commands (**commit**, **log**), and working with branches (**branch**, **checkout**, **merge**).
2. Collaborative Development with GitHub: Creating a remote repository on GitHub, pushing local changes, and performing a Pull Request to merge changes from a different branch.
3. Implementing a CI Pipeline with Jenkins: Installing Jenkins on a virtual machine or a container. Configuring a simple Freestyle project to build a Java or Python application from a GitHub repository.
4. Building a Jenkins Pipeline: Writing a basic Jenkinsfile to define a declarative pipeline with stages for **Checkout**, **Build**, and **Test**.
5. Automating Application Deployment: Extending the Jenkins pipeline to automatically deploy the application to a local web server (e.g., Tomcat, Nginx) after a successful build and test.
6. Containerizing a Web Application: Writing a Dockerfile to containerize a simple web application (e.g., a "Hello World" Flask or Node.js app).
7. Managing Containers with Docker: Using Docker commands to build, run, stop, and remove the container from the previous experiment. Publishing the image to a container registry like Docker Hub.
8. Multi-Container Application with Docker Compose: Creating a **docker-compose.yml** file to define and run a multi-container application, such as a web server and a database.
9. Provisioning a Virtual Machine with Terraform: Setting up Terraform and writing a basic configuration file to provision a virtual machine on a public cloud provider (e.g., AWS, Azure, GCP) or a local provider like VirtualBox.
10. Implementing IaC for a Web Server: Writing a Terraform script to provision a virtual machine, install a web server (e.g., Apache, Nginx), and deploy a static webpage.

Course Title : Introduction to Quantum Computing Lab

Semester : V

Prerequisite :

L – T – P : 0-0-2

Course Code : 25UD1245PECL511I

Credits : 1

Course Type : PCE

List of experiments

1. Introduction to Qiskit and IBM Quantum Experience
2. Simulation of Bit-flip and Phase-flip Errors
3. Implement the 3-Qubit Bit Flip Code
4. Implement the 3-Qubit Phase Flip Code
5. Run a Bell State Circuit on a Real NISQ Device
6. Explore Quantum Volume and Circuit Depth on NISQ Hardware
7. Design a Variational Quantum Classifier (VQC)
8. Intro to Quantum Computer Vision: Quantum Edge Detection

Course Title : Human Computer Interaction Lab

Semester : V

Prerequisite :

L – T – P : 0-0-2

Course Code : 25UD1246PECL511J

Credits : 1

Course Type : PEC

List of Experiments :

1. Introduction to HCI Tools
 - a) Introduction to tools like Figma, Balsamiq, Adobe XD, or similar prototyping software.
 - b) Basics of wireframing and interface sketching.
2. Designing a User Interface for a Web or Mobile App
 - a. Design a login and dashboard screen for a real-world application.
 - b. Apply principles of layout, color, typography, and feedback.
3. Developing Interactive Prototypes
 - a. Create an interactive prototype using Figma or Adobe XD.
 - b. Include navigation between screens, buttons, and form handling.
4. Usability Evaluation and Heuristic Analysis
 - a. Perform heuristic evaluation on an existing system.
 - b. Identify usability issues and provide redesign recommendations.
5. Task Analysis and Cognitive Walkthrough
 - a) Create a task model (e.g., GOMS or HTA) for a chosen task.
 - b) Conduct a cognitive walkthrough to evaluate the ease of task completion.
6. Accessibility and Universal Design Implementation

Design an accessible interface considering color contrast, keyboard navigation, and screen reader support.
7. Design for Ubiquitous or Augmented Reality Applications (Demo-Based)
 - a. Study and document the interface of a ubiquitous or AR application.
 - b. Suggest UI improvements based on interaction principles.

8. User Testing and Feedback Collection

- a. Conduct user testing with 3–5 participants.\
- b. Prepare a report analyzing feedback and proposing refinements.

9. Tools and Technologies (Suggested):

- a. Figma, Balsamiq, Adobe XD
- b. HTML/CSS + JavaScript (for basic UI prototyping)
- c. UsabilityHub, Maze, or similar user testing platforms

Course Title : Software Project Management Lab

Semester : V

Prerequisite : Software Project Management

L – T – P : 0-0-2

Course Code : 25UD1246PECL511K

Credits : 1

Course Type : PEC

List of Experiments:

1. Develop a Software Project Charter for a proposed application, defining objectives, scope, stakeholders, and deliverables.
2. Create a Work Breakdown Structure (WBS) for a medium-scale software project using MS Project or equivalent tool.
3. Perform Project Effort Estimation using Function Point Analysis (FPA) for a given system description.
4. Estimate Project Cost and Schedule using COCOMO II model for a proposed software application.
5. Prepare a Gantt Chart and Network Diagram (PERT/CPM) for project scheduling and identify the critical path.
6. Perform Risk Identification and Risk Assessment for a software development project, with mitigation plans.
7. Create a Resource Allocation Plan for team members, tools, and budget in a sample project scenario.
8. Develop a Project Communication Plan including reporting formats, meeting schedules, and stakeholder updates.
9. Create a Software Quality Assurance (SQA) Plan with standards, metrics, and review procedures.
10. Monitor and Control Project Progress by tracking planned vs. actual schedules in MS Project/JIRA.
11. Prepare a Change Management Plan for handling scope changes and feature requests during the project lifecycle.
12. Conduct a Post-Project Review & Lessons Learned Document summarizing successes, failures, and process improvements.

Course Title : Network Management Lab

Semester : V

Prerequisite : Network Management

L – T – P : 0-0-2

Course Code : 25UD1246PECL511L

Credits : 1

Course Type : PEC

1. Install and configure a network monitoring tool (e.g., Nagios, Zabbix, or SolarWinds) to track device availability.
2. Simulate a small network in Cisco Packet Tracer or GNS3 and configure SNMP agents and managers.
3. Retrieve and display MIB (Management Information Base) data using an SNMP client tool.
4. Develop a script (Python/Perl) to query SNMP-enabled devices and display system statistics (CPU, memory, uptime).
5. Configure and monitor traffic flows on a router or switch using NetFlow or sFlow.
6. Set up alerts in a network monitoring system for critical events such as high CPU usage or link failure.
7. Use Wireshark to capture and analyze SNMP packets for request/response behavior.
8. Perform bandwidth usage monitoring and generate a utilization report for a given time period.
9. Implement a fault management scenario where network downtime is detected and logged automatically.
10. Simulate network configuration management by backing up and restoring router/switch configurations.
11. Demonstrate remote network device management via SSH/Telnet and apply security best practices.
12. Generate and analyze network performance reports (latency, packet loss, jitter) using monitoring tools.

Course Title : Finance Engineering Lab

Semester : V

Prerequisite : Finance Engineering

L – T – P : 0-0-2

Course Code : 25UD1246PECL511M

Credits : 1

Course Type : PEC

1. Implementation of Matrix Operations, Eigenvalue/Eigenvector Computation, and Covariance Matrix Decomposition for Financial Data
2. Design of a Numerical Solver for Linear Systems in Portfolio Risk Modeling.
3. Implementation of Orthogonal Transformations for Portfolio Risk Diversification Analysis
4. Design of a Markowitz Mean-Variance Portfolio Optimization Model using Quadratic Programming (QP)
5. Implementation of Robust Portfolio Optimization with Equality and Inequality Constraints.
6. Design of a Quadratically Constrained Portfolio Optimization Model (QCQP) with Transaction Costs.
7. Implementation of a Mixed-Integer QP Model for Index Tracking with Limited Assets.
8. Design of a CAPM-Based Portfolio Return Prediction and Performance Attribution Model
9. Implementation of Fama-French 3/5 Factor Models for Multi-Factor Risk Analysis
10. Design of an Arbitrage Pricing Theory (APT) Model for Asset Pricing
11. Implementation of PCA for Dimensionality Reduction and Latent Factor Extraction from Asset Returns
12. Design of a Minimum-Variance Portfolio using Principal Components
13. Implementation of a Dynamic Factor Model for Volatility and Yield Curve Analysis
14. Design of a Risk Parity Portfolio and Performance Comparison with Mean-Variance Optimization
15. Implementation of a Market Index Tracking Portfolio with Fewer Assets

Multidisciplinary Minor I [Semester V]

Sr. No	Course Title	Course Code	Course Type	Teaching Scheme – Hours per week				Examination Scheme - Marks					Total	
				L	T	P	CR	CA	MSE	ESE	TW	PR/OR		
	Any One													
1	Database Management Systems	25UD1246MD506A	MD-M	3	0	2	4	20	20	60	-	-	100	
2	Introduction to Machine Learning	25UD1246MD506B												
3	Cloud Computing and Security	25UD1246MD506C												
4	Software Engineering	25UD1246MD506D												
5	Information and Cybersecurity	25UD1246MD506E												
6	Data Science and Visualization	25UD1246MD506F												
7	Fundamentals of Hyperspectral Imaging	25UD1246MD506G												
8	Introduction to IoT	25UD1246MD506H												
9	Artificial Neural Networks	25UD1246MD506I												
10	Introduction to Blockchain	25UD1246MD506J												
11	Virtualization and Containers	25UD1246MD506K												
12	DevSecOps Principles and Practices	25UD1246MD506L												
13	Introduction to Quantum Computing	25UD1246MD506M												
14	Human-Computer Interaction	25UD1246MD506N												
15	Software Project Management	25UD1246MD506O												
16	Network Management	25UD1246MD506P												
17	Finance Engineering	25UD1246MD506Q												

Course Title : Systematic Literature Review	Semester : V
Prerequisite :	L – T – P : 3-0-0
Course Code : 25UD1246RS501	Credits : 3
Stream : Research	

Course Objectives:

1. To introduce students to the principles and methodology of conducting systematic literature reviews (SLRs).
2. To enable students to identify, collect, evaluate, and synthesize scholarly articles in a systematic and replicable way.
3. To build capabilities for academic writing, critical thinking, and review-based research methodology.
4. To prepare students for future capstone projects, research-based electives, internships, and project proposals.

Course Outcomes:

After learning the course, the students will be able to:

1. Understand the methodology, scope, and structure of a systematic literature review.
2. Formulate a clear research question and design a review protocol.
3. Conduct comprehensive and reproducible searches in scientific databases.
4. Evaluate and synthesize scholarly research using quality assessment tools.
5. Prepare a structured literature review report with appropriate citations and critical analysis.

Course Contents:

UNIT I

Introduction to Systematic Literature Review: [8 Hours]

Literature review vs. systematic review vs. meta-analysis, Importance and applications of SLR in computer science and IT, Types of reviews: narrative, scoping, mapping, and systematic

UNIT II

Planning the Review:

[8 Hours]

Defining research problems and review objectives, Formulating research questions using PICO, SPICE, etc., Designing the review protocol, Inclusion and exclusion criteria, registration of protocol.

UNIT III

Search Strategy and Data Sources

[8 Hours]

Identifying digital libraries and databases: IEEE Xplore, ACM DL, Scopus, Web of Science, Google Scholar, Constructing effective search queries using Boolean logic, Snowballing and grey literature searches, Reference management using Mendeley/Zotero

UNIT IV

Screening and Quality Assessment:

[8 Hours]

Title and abstract screening, full-text review, PRISMA flow diagram and documentation, Quality assessment criteria (e.g., CASP, AMSTAR), Inter-rater reliability and resolving reviewer disagreements
Data Extraction and Synthesis:
Creating data extraction forms, Quantitative vs. qualitative synthesis, Thematic analysis, content analysis, vote counting, Writing a synthesis with critical insights

UNIT V

Reporting and Publishing the Review

[8 Hours]

Structuring the SLR report: IMRAD format, tables, figures, Referencing and citation styles (APA/IEEE), Common errors and pitfalls in literature reviews, Ethical considerations in secondary research, Introduction to publication process and journal selection

Text Books:

1. Barbara Kitchenham, “**Guidelines for Performing Systematic Literature Reviews in Software Engineering**”, EBSE Technical Report
2. Diana Ridley, “**The Literature Review: A Step-by-Step Guide for Students**”, Sage Publications
3. Booth, “**Sutton & Papaioannou, Systematic Approaches to a Successful Literature Review**”, Sage
4. Booth, A., Sutton, A., & Papaioannou, D. – “**Systematic Approaches to a Successful Literature, Review**”, SAGE.
5. Gough, D., Oliver, S., & Thomas, J. – “**An Introduction to Systematic Reviews**”, SAGE.

Reference Books:

1. Khan, K. S., Kunz, R., Kleijnen, J., & Antes, “*G. Systematic Reviews to Support Evidence-Based Medicine*”
2. Gough, Oliver, Thomas, *An Introduction to Systematic Reviews*, Sage
3. PRISMA Guidelines (www.prisma-statement.org)
4. IEEE/ACM Author Resources
5. Scopus, Web of Science, Google Scholar Tutorials

Course Title : Experimental Design

Semester : V

Prerequisite :

L – T – P : 3-0-0

Course Code : 25UD1246RS502

Credits : 3

Stream : Research

Course Objectives:

1. To understand the fundamental principles and techniques of designing experiments.
2. To apply statistical methods in planning, conducting, and analyzing experiments.
3. To recognize sources of variability and reduce experimental error.
4. To develop the ability to select appropriate experimental designs for various research scenarios.
5. To enable interpretation and reporting of experimental results effectively.

Course Outcomes:

After learning the course, the students should be able:

1. Explain the principles and importance of experimental design.
2. Identify and classify variables and sources of error.
3. Apply suitable experimental designs such as CRD, RBD, and LSD.
4. Analyze data using ANOVA and interpret the results.
5. Design real-world experiments with valid statistical approaches.

Course Contents:

UNIT I

Introduction to Experimental Design:

[8 Hours]

Objectives and Importance of Experimentation, Definitions: Experiment, Treatment, Experimental Unit, Experimental Error, Types of Variables: Independent, Dependent, Confounding, Principles of Experimental Design: Randomization, Replication, Blocking, Control, Types of Errors: Systematic vs. Random Errors, Overview of Statistical Hypothesis Testing

UNIT II

Completely Randomized Design (CRD):

[8 Hours]

Layout and Model of CRD, Analysis of Variance (ANOVA) for CRD, Advantages and Limitations, Applications and Examples.

UNIT III

Randomized Block Design (RBD):

[8 Hours]

Layout and Model of RBD, Analysis of Variance (ANOVA) for RBD, Efficiency of RBD over CRD, Missing Plot Technique in RBD

UNIT IV

Latin Square Design (LSD):

[8 Hours]

Structure and Layout of LSD, Assumptions and Model, Analysis of Variance (ANOVA) for LSD, Advantages, Limitations, and Examples.

UNIT V

Factorial Experiments and Other Designs:

[8 Hours]

Introduction to Factorial Designs (2^2 , 2^3 , etc.), Main Effects and Interaction Effects, Confounding in Factorial Designs, Split-Plot and Strip-Plot Designs (conceptual overview), Response Surface Methodology (basic idea)

Text Books:

1. Montgomery, D. C. – “*Design and Analysis of Experiments*,” Wiley.
2. Gomez, K.A. & Gomez, A.A. – “*Statistical Procedures for Agricultural Research*,” Wiley.

Reference Books:

3. Das, M. N. & Giri, N. – “*Design and Analysis of Experiments*,” New Age International.
4. Panse, V.G. & Sukhatme, P.V. – “*Statistical Methods for Agricultural Workers*,” ICAR.

Course Title : Cyber Security for CPS
Course Code : 25UD1246HR501
Prerequisite : Computer Network
Course Type : Honors

Semester : V
L – T – P : 2-0-0
Credits : 2

Course Objective:

1. Understand the architecture and models of Industrial Control Systems (ICS) and Cyber-Physical Systems (CPS).
2. Analyze cyber threat models, real-world incidents, and defensive mechanisms in CPS environments.
3. Evaluate vulnerabilities and protection techniques in ICS/SCADA systems using penetration testing and case studies.
4. Assess the security challenges and solutions related to IoT devices integrated within CPS.
5. Apply risk management principles, legal, and regulatory frameworks to protect and govern CPS infrastructures.

Course Outcomes:

After completion of this course, students will be able to

1. Explain the architecture, modeling techniques, and foundational principles of ICS and Cyber-Physical Systems.
2. Apply risk management and regulatory compliance strategies for securing CPS environments based on NIST and legal frameworks.
3. Analyze types of cyber threats, impacts of CPS attacks, and develop defense strategies based on real-world incidents.
4. Evaluate SCADA-based control systems, identify vulnerabilities, and apply security testing techniques, including case studies.
5. Assess IoT architecture and implement security mechanisms for safeguarding CPS-integrated IoT systems.

Course Contents:

UNIT I

Introduction to Industrial Control Systems and Cyber-Physical Systems [06 Hours]

Introduction to Industrial Control Systems (ICS) and Operations, Overview of Industrial Network Protocols, Cyber-Physical System (CPS) Modeling and Plant Models, Feedback Control Models and Anomaly Detection Techniques, Concepts and Principles of CPSS: Confidentiality, Integrity, Availability, Non-repudiation, Veracity, Plausibility, Comparison between ICT and CPS in terms of Performance, Reliability, and Risk Management

UNIT II

Cyber Threats, Security Incidents, and Defense Mechanisms: [06 Hours]
Cyber Threat Models: Types of Threats in ICS and CPS, 3-Dimensional Cyber-Attack Space for Industrial Critical Systems, Impacts of Cyber-Physical Attacks, Known Security Incidents in CPS: Smart Grids, ICS/SCADA, IoT Systems, Defense Mechanisms and Incident Response in CPSS, Cyberwarfare: Legal Frameworks, Use of Force, and International Conventions.

UNIT III

ICS/SCADA Systems and Vulnerabilities [06 Hours]
ICS/SCADA Architecture: PLCs, HMIs, IEDs, RTUs, Field Networks and Protocols: Modbus, Profibus, DNP3, Vulnerabilities in ICS/SCADA Systems, Security Assessment and Penetration Testing, Protection Configuration and Best Practices, Case Study: Stuxnet Attack, Trends in Industrial Cybersecurity Breaches.

UNIT IV

IoT Security in Cyber-Physical Systems [06 Hours]
Architecture and Components: Sensors, Actuators, LoPAN, Wi-Fi, Vulnerabilities in IoT Systems, Penetration Testing and Security Assessment in IoT Deployments, Protection Strategies for IoT-Based CPS Environments, IoT Integration in ICS/SCADA Systems, Privacy and Legal Aspects in IoT Security.

UNIT V

CPSS Risk Management, Privacy, and Regulatory Frameworks: [06 Hours]
Risk Management in CPSS, Application of NIST Cybersecurity Framework for CPS and Critical Infrastructure, Privacy Laws and Regulatory Aspects, FTC and FCC Guidelines for Unfair and Deceptive Security Practices, Project Closeout, Reporting, and Future Trends in CPSS

Textbooks:

1. Raj Rajkumar, Insup Lee, Lui Sha, and James Stankovic “*Cyber-Physical Systems: From Theory to Practice*” Springer.
2. Eric D. Knapp, Joel Thomas Langill, “*Industrial Network Security: Securing Critical Infrastructure Networks for Smart Grid*”, SCADA, and Other Industrial Control Systems” Syngress
3. Tyson Macaulay and Bryan Singer, “*Cybersecurity for Industrial Control Systems: SCADA, DCS, PLC, HMI, and SIS*”, CRC Press

Reference Books

1. Çetin Kaya Koç “*Cyber-Physical Systems Security: Foundations and Challenges*”, Springer
2. Robert Radvanovsky and Jacob Brodsky, “*Handbook of SCADA/Control Systems Security*”, CRC Press
3. Al-Sakib Khan Pathan, “*IoT Security Issues*”, CRC Press

- NIST Special Publication 800-82 Revision 2, “**Guide to Industrial Control Systems (ICS) Security**”, National Institute of Standards and Technology (NIST) *Publisher*
NIST Cybersecurity Framework (CSF), *Official Document*

Course Title : Business and IT Management	Semester : V
Prerequisite :	L – T – P : 0-0-2
Course Code : 25UD1246HR501	Credits : 1
Course Type : Honors	

Course Objectives:

- To understand the role and relevance of Information Systems (IS) in supporting business processes and strategic decision-making.
- To analyze how IT aligns with business strategies and supports organizational transformation.
- To explore enterprise applications like ERP, CRM, and SCM and their role in optimizing business performance.
- To examine IT governance, risk management, and compliance frameworks applicable in business contexts.
- To investigate current and emerging IT trends and assess their impact on business innovation and sustainability.

Course Outcomes:

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

- Explain the integration of information systems with business functions .
- Analyze and align business strategies with IT frameworks and evaluate their impact..
- Demonstrate understanding of enterprise applications and suggest appropriate IT.
- Apply IT governance models and risk mitigation strategies in business scenarios.
- Assess emerging technologies and trends for enhancing business value and sustainability.

Course Contents:

UNIT I

Introduction to Business and IT Management: [6 Hours]

Overview of Business and IT alignment, Role of Information Systems in Business, Types of Information Systems (TPS, MIS, DSS, ESS). IT as a business enabler, IT Infrastructure components

UNIT II

Business Strategy and IT Alignment: [6 Hours]

Strategic role of IT in business, IT strategy vs Business strategy, Porter’s Five Forces Model, Value Chain analysis using IT, Case studies on strategic IT implementation

UNIT III

Enterprise Applications and Business Process Management:**[6 Hours]**

Enterprise Resource Planning (ERP), Customer Relationship Management (CRM), Supply Chain Management (SCM), Business Process Reengineering (BPR), Workflow automation and optimization

UNIT IV**IT Governance and Risk Management:****[8 Hours]**

IT policies, compliance, and standards, IT Governance frameworks (COBIT, ITIL), Data privacy and cybersecurity basics, Risk assessment in IT projects, Business Continuity Planning (BCP) and Disaster Recovery

UNIT V**Emerging Trends and Future of Business IT:****[8 Hours]**

Cloud computing and SaaS in business, Business Intelligence (BI) and Data Analytics, Artificial Intelligence in business, Digital Transformation and Industry 4.0, E-commerce and mobile business strategies

Textbooks:

1. Laudon & Laudon , “*Management Information Systems*”, 18th Edition, Pearson.
2. James A. O'Brien, “*Introduction to Information Systems*”, 16th Edition, McGraw-Hill.
3. Rainer & Prince, “*Introduction to Information Systems*”, 10th Edition, Wiley.
4. Robert D. Austin et al., “*Information Systems: People, Technology, and Processes*”, 4th Edition, Pearson.

Course Title : Advanced Database Technology

Semester : VI

Prerequisite : Database Management Systems

L – T – P : 2-0-0

Course Code : 25UD1246PC601

Credits : 2

Course Type : PCC

Course Objective:

1. To introduce the fundamental concepts of parallel and distributed database architectures and their applications.
2. To provide knowledge on object-oriented and object-relational databases, their design, and querying mechanisms.
3. To familiarize students with XML data models, web-based databases, and their role in modern data integration and mining.
4. To explore the unique challenges of mobile databases, including transaction management, recovery, and location-aware services.
5. To expose students to intelligent databases, including active, deductive, and multimedia databases with real-world applications.

Course Outcomes:

After learning the course, the students should be able:

1. To explain the architectures and operations of parallel and distributed database systems.
2. To analyze object-oriented and object-relational database models, query languages, and standards.
3. To implement XML data models and perform querying and integration with web and data mining systems.
4. To evaluate mobile database architectures, transaction models, and recovery mechanisms.
5. To describe and design intelligent and multimedia databases with appropriate data structures and indexing techniques.

Course Contents:

UNIT I

Parallel And Distributed Databases:

[06 Hours]

Database System Architectures: Centralized and Client-Server Architectures – Server System, Architectures – Parallel Systems- Distributed Systems – Parallel Databases: I/O

Parallelism – Inter and Intra Query Parallelism – Inter and Intra operation Parallelism – Distributed Database

Concepts - Distributed Data Storage – Distributed Transactions – Commit Protocols – Concurrency Control –

Distributed Query Processing – Three Tier Client Server Architecture- Case Studies

UNIT II

Object and Object Relational Databases: [06 Hours]
Hours] Concepts for Object Databa
Constructors – Encapsulation of Operations – Methods – Persistence – Type an
Objects – Object Database Standards, Languages and Design: ODMG Model – OE
Relational Systems: Object Relational features in SQL /Oracle – Case Studies.

UNIT III

Xml Databases: XML Databases: [06 Hours]
XML Data Model – DTD - XML Schema - XML Querying – Web Databases– JDBC–
Information Retrieval – Data Warehousing – Data Mining

UNIT IV

Mobile Databases: [0 Hours]
Location and Handoff Management - Effect of Mobility on Data Management - Location
Dependent Data Distribution - Mobile Transaction Models - Concurrency Control - Transaction
Commit Protocols- Mobile Database Recovery Schemes.

UNIT V

Intelligent Databases: [06 Hours]
Active databases – Deductive Databases – Knowledge bases – Multimedia Databases
Multidimensional Data Structures – Image Databases – Text/Document Databases- Video
Databases – Audio Databases – Multimedia Database Design.

Textbooks:

1. Carlo Zaniolo, Stefano Ceri, “*Advanced Database Systems*”, Morgan Kauffmann Publishers.,1997
2. Subramaniam, “*Multimedia Databases*”, Morgan Kauffman Publishers, 2008
3. Rajesh Narang, “*Object Oriented Interfaces and Databases*”, Prentice-Hall of India, Pvt. Ltd., 2004
4. Thomas Cannolly and Carolyn Begg, “*Database Systems, A Practical Approach to Design, Implementation and Management*”, Pearson Education, 6th Edition, 2015
5. Jeffrey A. Hoffer, Mary B. Prescottand Fred R. McFadden, “*Modern Database Management*”, Prentice Hall, 2007

Reference books:

1. Henry F Korth, Abraham Silberschatz and S. Sudharshan, “*Database System Concepts*”, McGraw Hill, 7th Edition, 2019
2. C. J. Date, A. Kannan and S. Swamynathan, “*An Introduction to Database Systems*”, Pearson Education, 8th Edition, 2006
3. R. Elmasri, S. B. Navathe, “*Fundamentals of Database Systems*”, Pearson Education/Addison Wesley, 7th Edition, 2015
4. Ramakrishnan, Gehrke, “*Database Management System*”, Tata McGraw Hill Publications, 3rd Edition, 2022
5. Ramez Elmasri, Sham Navathe, “*Fundamentals of Database Systems*”, Addison-Wesley, 7th Edition 2015

Course Title : Advanced Web Technology

Semester : VI

Prerequisite : Introduction to Web Design

L – T – P : 2-0-0

Course Code : 25UD1246P C602

Credits : 2

Course Type : PCC

Course Objective:

1. To introduce modern web technologies, architectures, and frameworks used in full-stack web development.
2. To impart the principles of Object-Oriented Programming (OOP) using PHP, including core concepts and methodologies.
3. To explore advanced PHP functionalities such as cURL, regular expressions, mailing, and integration with web services/APIs.
4. To familiarize students with the Laravel framework for building scalable and modular web applications using the MVC pattern.
5. To enable students to build server-side web applications using Node.js and understand its runtime and module handling.

Course Outcomes:

After completion of this course, students will be able to

1. Identify and explain modern web components, serverless architecture, and frameworks used in web development.
2. Apply object-oriented programming concepts using PHP to build secure and modular code.
3. Utilize advanced PHP features such as cURL, mail functions, regular expressions, and APIs to enhance application functionality.
4. Develop full-stack web applications using the Laravel MVC framework and implement CRUD operations with database connectivity.
5. Build and test web servers using Node.js for handling HTTP requests, JSON responses, and routing mechanisms.

Course Contents:

UNIT I

Advance Web Technology:

[06 Hours]

Core Components, Frameworks and Libraries, APIs, PWAs, Server less Architecture, Web Assembly,

Backend Technologies, Frontend Framework , Database management .

UNIT II

Object Oriented PHP:

[06 Hours]

Object Oriented Programming with PHP –Classes, Properties, Methods, Magic Methods: Constructor, Destructor, Getter and Setter, Encapsulation, Inheritance, Data Abstraction, Polymorphism

UNIT III

Advance PHP

[06 Hours]

Constructor, Parameterized constructors, Web Scraping using cURL, Regular Expression, Mail function, Web Services & APIs

UNIT IV

PHP MVC Framework - Laravel

[06 Hours]

Introduction to Laravel and MVC, Environment Setup, Routes, Namespaces, Controllers, Views, Request Response, Redirections, Forms, Session, Cookie, Database Connectivity and CRUD operations

UNIT V

Node.js

[06 Hours]

Introduction to Node.js, Node Package Manager, REPL Terminal, Node.js Webserver – Server and Clients, Creating a simple server, Rendering HTML, Rendering JSON Data, Routing

Text Books:

1. L. Atkinson and M. Suraski, “**PHP Manua**”l, PHP Documentation Group, 2022. [Online]. Available: <https://www.php.net/manual/en/>
2. B. Tatro, K. MacIntyre, and P. Lerdorf, “**Programming PHP**”, 4th ed., Sebastopol, CA, USA: O’Reilly Media, 2020.
3. M. Stauffer, “**Laravel: Up and Running: A Framework for Building Modern PHP Apps**”, 2nd ed., Sebastopol, CA, USA: O’Reilly Media, 2019.

Reference Books:

1. E. Freeman and E. Robson, “**Head First HTML and CSS**”, 2nd ed., Sebastopol, CA, USA: O’Reilly Media, 2012.
2. M. Cantelon, M. Harter, B. Rajlich, and A. Holowaychuk, “**Node.js in Action**”, 2nd ed., Shelter Island, NY, USA: Manning Publications, 2017.
3. D. Flanagan, “**JavaScript: The Definitive Guide**”, 7th ed., Sebastopol, CA, USA: O’Reilly Media, 2020.

Course Title : Compiler Design

Semester : VI

Prerequisite :

L – T – P : 2-0-0

Course Code : 25UD1246PC603

Credits : 2

Course Type : PCC

Course Objectives

1. To introduce the fundamental concepts of formal languages, automata theory, and computational models.
2. To explain the structure, phases, and functions of a compiler, including lexical analysis, syntax analysis, semantic analysis, and code generation.
3. To develop the ability to design and implement lexical analyzers, parsers, and syntax-directed translators.
4. To provide knowledge of intermediate code generation, type checking, and control flow management.
5. To expose students to code optimization techniques and issues in code generation for efficient target programs.

Course Outcomes

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

1. Apply the concepts of regular expressions, context-free grammars, and automata theory to compiler design.
2. Design and implement lexical analyzers and parsers for programming languages using suitable algorithms and tools.
3. Construct syntax-directed translation schemes and generate intermediate representations such as three-address code.
4. Perform type checking, control flow analysis, and backpatching in intermediate code generation.
5. Implement basic code optimization techniques and generate efficient target code for given source programs.

Course Contents:

UNIT I

[06 Hours]

Regular expressions and finite automata. Context-free grammars and push-down automata. Regular and context-free languages, pumping lemma. Turing machines and undecidability.

UNIT II

Introduction to Compiling and Lexical Analysis:

[06 Hours]

Definition, analysis of the source program, the phases of a compiler, the grouping of phases, Compiler-Construction tools, Role of the Lexical analyzer, Input buffering, Specification of Tokens, A Language for Specifying Lexical Analyzers, Design of a Lexical Analyzer generator

UNIT III

Syntax Analysis:

[06 Hours]

The role of the Parser, Context-free grammars, Writing a Grammar, Top-Down Parsing, Bottom-Up Parsing, Operator-precedence Parsing, LR-Parsers, Using Ambiguous Grammars, Parser Generators.

UNIT IV

Syntax-Directed Translation:

[06 Hours]

Definitions, Construction of Syntax Trees, Bottom-Up Evaluation of S Attributed definitions, Top-Down Translation, Bottom-Up Evaluation of Inherited attributes.

Intermediate Code Generation: Variants of syntax trees, Three address code, Type checking, Control Flow and Backpatching, procedure calls.

UNIT V

Code Generation and Code Optimization:

[06 Hours]

Code Generation: Issues in the design of code generation, The target language, Code Optimization: Need of code optimization, Principal sources of optimization, Basic blocks and flow graphs, Optimization of basic blocks, Peephole optimization.

Textbooks:

1. Aho, Sethi, Ullman, "Compilers-Tools and Techniques", Pearson, 2nd Edition, 2015.
2. Tremblay, Sorenson, "Theory and Practice of Compiler Writing", McGraw Hill Publication.
3. Hopcroft, "Introduction to Automata Theory, Languages and Computation", Pearson Publication.

Reference books:

1. Paul G. Sorenson, "*Compiler Writing*", Tata McGraw Hill.
2. Robin Hunter, "*The Essence of Compilers*", Pearson Publication, 2005.

Course Title : Deep Learning

Semester : VI

Prerequisite : Machine Learning

L – T – P : 2-0-0

Course Code :25UD1246PEC604

Credits : 2

Course Type : PCC

Course Objective:

1. To Understand the motivation, evolution, and mathematical foundations behind deep learning models.
2. To Learn the architectures, training methods, and optimization techniques for deep feedforward networks.
3. To Gain the ability to design, train, and fine-tune convolutional neural networks for image-based applications.
4. To Explore and implement recurrent architectures for sequence and time-series data.
5. To Examine advanced deep learning topics, including generative models, transformers, reinforcement learning, and ethical AI considerations.

Course Outcomes:

After learning the course, the students will be able to:

1. Explain the evolution from AI → ML → ANN → Deep Learning, the limitations of shallow networks, and the role of mathematical foundations in DL.
2. Design, initialize, and train deep feedforward neural networks using appropriate activation functions, optimization algorithms, and regularization techniques.
3. Implement convolutional neural networks for image classification, perform transfer learning, and apply pre-trained models to solve real-world vision problems.
4. Develop and evaluate sequence models such as RNN, LSTM, and GRU for tasks in NLP and time-series forecasting.
5. Demonstrate understanding of advanced deep learning models such as autoencoders, GANs, transformers, and reinforcement learning agents, and discuss their applications.

Course Contents:

UNIT I

[06 Hours]

Evolution of AI → ML → ANN → Deep Learning, Limitations of shallow networks, Overview of deep architectures, Mathematical foundations: Vectors, matrices, tensors, derivatives, gradients, Training challenges: vanishing/exploding gradients, overfitting, computational cost, Overview of

DL frameworks (TensorFlow, PyTorch, Keras), Case Study: Simple deep network for image classification (MNIST)

UNIT II

[06 Hours]

Deep feedforward network architecture and training pipeline, Activation functions revisited (ReLU, Leaky ReLU, ELU, Swish), Initialization techniques for deep networks, Regularization: L1, L2, Dropout, Data augmentation, Batch Normalization & Layer Normalization, Optimization algorithms: Adam, RMSprop, Nadam, learning rate schedules, Gradient clipping and advanced training strategies.

UNIT III

[06 Hours]

CNN architecture: Convolution, pooling, fully connected layers, Feature maps, receptive fields, stride, padding, Popular architectures: LeNet, AlexNet, VGG, ResNet, DenseNet, EfficientNet, Transfer learning & fine-tuning pre-trained CNN models, Applications in image classification, object detection, segmentation, Introduction to detection models: YOLO, Faster R-CNN (concepts only)

UNIT IV

[06 Hours]

Limitations of feedforward & CNNs for sequence data, RNN architecture and Backpropagation Through Time (BPTT), Vanishing gradient problem in RNNs, LSTM and GRU networks: structure, working, applications, Bidirectional RNNs, Encoder–Decoder architecture, Applications in NLP (text generation, sentiment analysis) and time series forecasting

UNIT V

[06 Hours]

Generative models: Autoencoders, Variational Autoencoders (VAE), Generative Adversarial Networks (GAN) – concepts and applications, Attention mechanism and Transformers (BERT, GPT – introductory concepts), Reinforcement Learning with Deep Networks (Deep Q-Networks) – basics, DL in speech recognition, NLP, and recommendation systems, Ethical considerations: Bias, explainability, responsible AI in DL models, Future trends: Edge AI, Neuromorphic computing, Quantum Deep Learning

Text books:

1. Ian Goodfellow, “*Yoshua Bengio, Aaron Courville*” – Deep Learning, MIT Press.
2. Francois Chollet – “*Deep Learning with Python*”, Manning.
3. Rajalingappaa Shanmugamani – “*Deep Learning for Computer Vision*”, Packt.

Reference books:

1. Michael Nielsen – “*Neural Networks and Deep Learning*” (Free online book).
2. Aurélien Géron – “*Hands-On Machine Learning with Scikit-Learn*”, Keras &

Course Title : Ethical Hacking

Semester: VI

Course Code : 25UD1246PEC605A

L – T – P : 2-0-0

Prerequisite : Computer Networks

Credits : 2

Course Type : PEC

Course Objectives:

1. To provide students with foundational knowledge of ethical hacking principles and system/network vulnerabilities.
2. To develop skills in tools and techniques for penetration testing and secure environment setup.
3. To explore cryptographic methods, steganography, and biometric authentication in system defense.
4. To demonstrate hands-on ethical hacking practices such as sniffing, spoofing, and SQL injection.
5. To build awareness of hardware-level threats and protective techniques using frameworks like Metasploit.

Course outcomes:

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

1. Understand and explain the fundamentals of ethical hacking, networking concepts, communication protocols, attacker strategies, and reconnaissance techniques.
2. Apply various tools for information gathering, vulnerability analysis, penetration testing, and exploit development.
3. Demonstrate the application of cryptographic techniques, hashing, steganography, and biometric authentication for securing systems and communications.
4. Analyze and mitigate common cyberattacks such as spoofing, sniffing, malware injection, SQL injection, and session hijacking.
5. Evaluate advanced hardware and software-level security threats using tools like Wireshark and Metasploit, and reflect on ethical and professional responsibilities in cybersecurity practices

UNIT I

Foundations of Ethical Hacking and Computer Networking: [6 Hours]

Introduction to ethical hacking. Fundamentals of computer networking. TCP/IP protocol stack. IP addressing and routing. TCP and UDP. IP subnets. Routing protocols. IP version 6.

UNIT II

Ethical Hacking Tools and Techniques: Information Gathering, System Exploitation, and Cryptography: [6 Hours]

Installation of attacker and victim system. Information gathering using advanced google search, archive.org, netcraft, whois, host, dig, dnsenum and NMAP tool. Vulnerability scanning using NMAP and Nessus. Creating a secure hacking environment. System Hacking: password cracking, privilege escalation, application execution. Malware and Virus. ARP spoofing and MAC attack. Introduction to cryptography, private-key encryption, public-key encryption.

UNIT III

Cybersecurity Mechanisms: Cryptographic Hashes, Authentication, and Network Security: [6 Hours]

Cryptographic hash functions, digital signature and certificate, applications. Steganography, biometric authentication, network-based attacks, DNS and Email security.

UNIT IV

Practical Cyber Attacks and Hardware Security Mechanisms: [6 Hours]

Packet sniffing using wireshark and burpsuite, password attack using burp suite. Social engineering attacks and Denial of service attacks.

Elements of hardware security: side-channel attacks, physical inclinable functions, hardware trojans.

UNIT V

Cyber Attack Simulation and Case Studies using Metasploit: [6 Hours]

Different types of attacks using Metasploit framework: password cracking, privilege escalation, remote code execution, etc. Attack on web servers: password attack, SQL injection, cross site scripting. Case studies: various attacks scenarios and their remedies.

Textbooks:

1. W. Stallings, “*Data and Computer Communications*”, 10th ed., Pearson Education, 2013.
2. B. A. Forouzan, “*Data Communication and Networking*,” 5th ed., McGraw-Hill Education, 2012.
3. B. A. Forouzan, “*TCP/IP Protocol Suite*,” 4th ed., McGraw-Hill Education, 2009.

Reference Books

1. W. R. Stevens, “*UNIX Network Programming*,” Vol. 1: The Sockets Networking API, 3rd ed. Boston, MA: Addison-Wesley, 2003.
2. C.-H. Wu and J. D. Irwin, “*Introduction to Computer Networks and Cybersecurity*”. Boca Raton, FL: CRC Press, 2013.
3. W. Stallings, “*Cryptography and Network Security: Principles and Practice*”, 7th ed. Boston, MA: Pearson Education, 2017.

Course Title : Data Warehousing and Data Mining

Semester : VI

Prerequisite : Database Management Systems

L – T – P : 2-0-0

Course Code : 25UD1246PEC605B

Credits : 2

Course Type : PEC

Course Objectives:

1. Understand the architecture, components, and design of a data warehouse.
2. Apply OLAP concepts and differentiate between various OLAP tools and operations.
3. Explain data mining concepts, techniques, and their relation to databases and statistics
4. Implement data preprocessing techniques including data cleaning, integration, and transformation
5. Discover association rules and apply classification and clustering methods on datasets.
6. Evaluate the impact and applications of data mining in real-world scenarios, including spatial databases.

Course Outcomes:

After learning the course, the students will be able to:

1. To understand data warehouse architecture and components, and apply OLAP operational processes to support data analysis and decision-making.
2. To understand the data mining process, relate it to statistics and databases, and classify mining systems.
3. To apply data preprocessing techniques and mine association rules to extract meaningful patterns from datasets.
4. To compare and evaluate data mining techniques such as classification, prediction, and clustering for their effectiveness in extracting meaningful patterns from data.
5. To analyze real-world case studies to understand data mining applications in web, text domains, and evaluate the social impacts of data mining practices.

Course Contents:

UNIT I

[06 Hours]

Introduction to data warehousing, Evolution of decision support systems, Modeling a data warehouse, Granularity in the data warehouse, Data warehouse life cycle, Building a data warehouse, Warehousing Components, Data Warehousing Architecture, Online Analytical Processing operations: Slice, Dice, Rollup, Drilldown and Pivot , Categorization of OLAP Tools, Map-Reduce ETL process.

UNIT II

[06 Hours]

Introduction to Data mining and knowledge discovery, Relation to Statistics, Databases, Data Mining Functionalities, Steps in Data Mining Process, Architecture of a Typical Data Mining Systems, Classification of Data Mining Systems.

UNIT III

[06 Hours]

Overview of Data Mining Techniques, Data Preprocessing, Data Cleaning, Data Integration, Data Transformation and Data Reduction, Data Generalization and Summarization Based Characterization, Mining Association Rules in Large Databases.

UNIT IV

[06 Hours]

Classification and Prediction, Issues regarding Classification and Prediction, Classification by Decision Tree Induction, Bayesian Classification, Other Classification Methods.

Prediction, Clusters Analysis, Types of Data in Cluster Analysis, Categorization of Major Clustering Methods, Partitioning methods, Hierarchical methods.

UNIT V

[06 Hours]

Applications of Data Mining, Social Impacts of Data Mining, Case Studies, Mining WWW, Mining Text Databases, Mining Spatial Databases.

Text Books:

1. Paulraj Ponniah, “ *Data Warehousing: Fundamentals for IT Professionals*”, Wiley India, 2 nd edition, 2010
2. Han, Kamber, “*Data Mining Concepts and Techniques*”, Morgan Kaufmann, 3rd edition, 2011.
3. M.H. Dunham, “*Data Mining Introductory and Advanced Topics*”, Pearson Education, 2006.

Reference Books:

1. Reema Theraja, “*Data warehousing*”, Oxford University Press 2009.
2. Pang-Ning Tan, Michael Steinbach and Vipin Kumar, “*Introduction to Data Mining*”, Pearson Publisher 2 nd edition ,2018
3. Ian H. Witten, Eibe Frank and Mark A. Hall, “*Data Mining*”, Morgan Kaufmann edition,2023.

Course Title : Image Processing Techniques for Hyperspectral Data

Semester : VI

Prerequisite : Introduction to Hyperspectral Imaging

L – T – P : 2-0-0

Course Code : 25UD1246PEC605C

Credits : 2

Course Type :

Course Objective:

1. To understand advanced preprocessing and enhancement techniques for hyperspectral data.
2. To explore dimensionality reduction and feature extraction methods tailored for hyperspectral imagery.
3. To learn classification, detection, and unmixing techniques for hyperspectral applications.
4. To apply image fusion and advanced analysis methods for real-world case studies..

Course Outcomes:

After learning the course, the students should be able:

1. Apply advanced preprocessing methods to enhance hyperspectral data quality.
2. Reduce hyperspectral dimensionality while preserving important spectral information.
3. Classify hyperspectral data using both supervised and unsupervised methods.
4. Detect anomalies and targets in complex hyperspectral scenes.
5. Integrate hyperspectral data with other sources for improved decision-making.

Course Contents:

UNIT I

[06 Hours]

Hyperspectral Data Preprocessing

- Review of hyperspectral data characteristics.

- Noise sources in hyperspectral imaging.
- Radiometric calibration & atmospheric correction (FLAASH, QUAC).
- Bad band removal & band alignment.
- Geometric correction and spatial resampling.

UNIT II

[06 Hours]

Spectral Feature Extraction & Dimensionality Reduction: [8 Hours]

Spectral signatures and spectral libraries, Band selection vs. band extraction methods., Principal Component Analysis (PCA), Minimum Noise Fraction (MNF) transformation, Independent Component Analysis (ICA).

UNIT III

[06 Hours]

Classification Techniques

Supervised classification (SVM, Random Forest, k-NN), Unsupervised classification (k-means, ISODATA), Spectral Angle Mapper (SAM) & Mahalanobis distance classification, Validation methods: confusion matrix, kappa coefficient.

UNIT IV

[06 Hours]

Target & Anomaly Detection

Matched filter algorithms, RX anomaly detector, Sub-pixel target detection, Spectral unmixing and abundance estimation, Endmember extraction techniques (PPI, N-FINDR).

UNIT V

[06 Hours]

Advanced Applications & Data Fusion

Hyperspectral and multispectral data fusion, Hyperspectral-LiDAR data integration, Change detection in hyperspectral time-series, Case studies in agriculture, mineral exploration, and environmental monitoring, Recent trends and research challenges in hyperspectral image processing

Textbooks:

1. Chein-I Chang, “*Hyperspectral Data Processing: Algorithm Design and Analysis*”, Wiley, 2013.
2. H.B. Gunasena et al., “*Hyperspectral Remote Sensing: Fundamentals and Practices*”, CRC Press, 2020.

Reference books:

1. Qian Du, "*Hyperspectral Image Analysis*", Springer, 2016.
2. Naoto Yokoya et al., "*Hyperspectral Data Exploitation: Theory and Applications*", Elsevier, 2021.
3. J.M. Bioucas-Dias et al., "*Hyperspectral Remote Sensing Data Analysis and Future Perspectives*", IEEE, 2013

Course Title : Wireless Sensor Networks

Semester : VI

Prerequisite : IoT

L – T – P : 2-0-0

Course Code : 25UD1246PEC605D

Credits : 2

Course Type : PEC

Course Objective:

1. To understand the fundamentals of wireless sensor networks and its application to critical real time scenarios.
2. To study the various protocols at various layers and its differences with traditional
3. To understand the issues pertaining to sensor networks and the challenges involved in managing a sensor network.
4. To Understand the security protocols for wireless sensor networks.
5. To study the use of sensor networks in managing sensor area networks in various embedded systems.

Course Outcomes:

After learning the course, the students will be able to:

1. Describe the overview of wireless sensor networks and enabling technologies for wireless sensor networks
2. Apply the design principles of WSN architectures and operating systems for simulating environment situations.
3. Apply various concepts for assignment of MAC addresses.
4. Select the appropriate infrastructure, topology, joint routing and information aggregation for wireless sensor networks
5. Analyse the sensor network platform and tools state-centric programming.

Course Contents:

UNIT I

OVERVIEW OF WIRELESS SENSOR NETWORKS:

[06 Hours]

Single-Node Architecture - Hardware Components- Network Characteristics- unique constraints and challenges, Enabling Technologies for Wireless Sensor Networks- Types of wireless sensor networks.

UNIT II

ARCHITECTURES:

[06 Hours]

Network Architecture- Sensor Networks Scenarios- Design Principle, Physical Layer and Transceiver Design Considerations, Optimization Goals and Figures of Merit, Gateway Concepts, Operating Systems and Execution Environments- Introduction to TinyOS and nesC- Internet to WSN Communication

UNIT III

NETWORKING SENSORS:

[06 Hours]

MAC Protocols for Wireless Sensor Networks, Low Duty Cycle Protocols And Wakeup Concepts - SMAC, - B-MAC Protocol, IEEE 802.15.4 standard and ZigBee, the Mediation Device Protocol, Wakeup Radio Concepts, Address and Name Management, Assignment of MAC Addresses, Routing Protocols EnergyEfficient Routing, Geographic Routing.

UNIT IV

INFRASTRUCTURE ESTABLISHMENT:

[06 Hours]

Topology Control, Clustering, Time Synchronization, Localization and Positioning, Sensor Tasking and Control

UNIT V

SENSOR NETWORK PLATFORMS AND TOOLS:

[06 Hours]

Sensor Node Hardware – Berkeley Motes, Programming Challenges, Node-level software platforms, Node level Simulators, State-centric programming.

Text Books:

1. Holger Karl & Andreas Willig, "*Protocols and Architectures for Wireless Sensor Networks*", John Wiley, 2005.
2. Feng Zhao & Leonidas J.Guibas, "*Wireless Sensor Networks-An Information Processing Approach*", Elsevier, 2007
3. Walteneus Dargie , Christian Poellabauer, "*Fundamentals Of Wireless Sensor Networks - Theory And Practice*", John Wiley & Sons Publications, 2011

Reference Books:

1. KazemSohraby, Daniel Minoli, & TaiebZnati, "*Wireless Sensor Networks-Technology, Protocols, and Applications*", John Wiley, 2007.
 2. Anna Hac, "*Wireless Sensor Network Designs*", John Wiley, 2003
- e- Resources & other digital material
1. <http://pages.di.unipi.it/bonuccelli/sensori.pdf>

Course Title : Natural Language Processing

Semester : VI

Prerequisite : Introduction to Machine Learning

L – T – P : 2-0-0

Course Code : 25UD1246PEC605E

Credits : 2

Course Type : PEC

Course Objectives:

1. To learn the leading trends and systems in natural language processing.
2. To understand the concepts of morphology, syntax, semantics and pragmatics.
3. To recognize the significance of pragmatics for natural language understanding.
4. To describe a simple system based on logic and demonstrate the difference between generation and interpretation of that presentation.
5. To describe applications based on natural language processing and to show the pragmatic processing.

Course Outcomes:

After learning the course the student will be able to:

1. Understand the models, methods and algorithms of statistical Natural Language Processing.
2. Implement probabilistic models in code, estimate parameters for such models and conduct experiments to validate such models.
3. Apply core computer science concepts and algorithms, such as dynamic programming.
4. Understand linguistic phenomena and explore the linguistic features relevant to NLP.
5. Identify opportunities and conduct research in NLP.

UNIT I

[06 Hours]

Introduction to NLP: Definition, Issues, Application domain, Tools for NLP, Linguistic organization of NLP, NLP vs. PLP. Morphology: Inflectional, derivational, Parsing and parsing with FST.

UNIT II

[06 Hours]

Introduction and Basic Text Processing, Language Modeling N-grams: Simple N-grams, Counting words in Corpora, smoothing (Add One, Written-Bell, Good-Turing), N-grams for spelling and pronunciation

UNIT III

[06 Hours]

POS Tagging: Tagsets, Concept of HMM tagger, Rule based and stochastic POST, Algorithm for HMM tagging, Transformation based tagging, Models for Sequential tagging – MaxEnt, CRF.

UNIT IV

[06 Hours]

Parsing : CFG and different parsing techniques, Constituency Parsing, Dependency Parsing, Distributional Semantics, Lexical Semantics: Lexemes (homonymy, polysemy, synonymy, hyponymy), WordNet, Internal structure of words, Metaphor and metonymy and their computational approaches. Word Sense Disambiguation: Selectional restriction based, Machine learning based and dictionary based approaches.

UNIT V

[06 Hours]

Topic Models, Entity Linking, Information Extraction, Text Summarization, Text Classification, Sentiment Analysis and Opinion Mining.

Text Books:

1. Jurafsky & J. H. Martin, “Speech and Language Processing – An introduction to Language processing, Computational Linguistics, and Speech Recognition”, Pearson Education.
2. Allen, James, “Natural Language Understanding”, 2nd Edition, Benjamin/Cummings, 1996.

Reference Books:

1. Bharathi, A., Vineet Chaitanya and Rajeev Sangal, “*Natural Language Processing-A Paninian Perspective*”, Prentice Hall India, 1995.
2. Eugene Charniak, “*Statistical Language Learning*”, MIT Press, 1993.
3. Manning, Christopher and Heinrich Schütze, “*Foundations of Statistical Natural Language Processing*”, MIT Press, 1999.

Course Title : Blockchain in Finance

Semester : VI

Prerequisite :

L – T – P : 2-0-0

Course Code : 25UD1246PEC605F

Credits : 2

Course Type : PEC

Course Objectives

The course aims to:

1. Introduce the fundamentals of blockchain technology and its underlying cryptographic principles.
2. Explain blockchain architecture, consensus mechanisms, and smart contracts in the context of financial applications.
3. Explore the use of blockchain in payments, settlements, asset tokenization, and trade finance.
4. Familiarize students with blockchain-based financial platforms, cryptocurrencies, and decentralized finance (DeFi).
5. Develop skills to design, implement, and evaluate blockchain-based financial solutions.

Course Outcomes

Upon successful completion of this course, students will be able to:

1. Describe blockchain architecture, consensus algorithms, and cryptographic foundations.
2. Apply blockchain concepts to model secure and transparent financial transactions.
3. Analyze and design smart contracts for finance-related applications.
4. Evaluate the applicability of blockchain in banking, payments, trade finance, and asset management.
5. Implement blockchain-based prototypes for real-world financial use cases.

Course Contents:

UNIT I

Introduction to Blockchain and Cryptography

[06 Hours]

Overview of blockchain technology, History and evolution of blockchain in finance, Cryptographic primitives: hash functions, public/private key encryption, digital signatures, Distributed ledger technology (DLT) fundamentals, Blockchain vs. traditional financial system

UNIT II

Blockchain Architecture and Consensus Mechanisms

[06 Hours]

Structure of a blockchain: blocks, transactions, Merkle trees, Types of blockchains: public, private, consortium, Consensus algorithms: Proof of Work, Proof of Stake, Practical Byzantine Fault Tolerance, Scalability, interoperability, and security issues in blockchain

UNIT III

Smart Contracts and Financial Applications

[06 Hours]

Smart contract fundamentals, Platforms for smart contracts (Ethereum, Hyperledger Fabric, Corda), Payment systems and cross-border settlements, Asset tokenization and digital identity management, Blockchain in KYC/AML compliance

UNIT IV

Cryptocurrencies and DeFi

[06 Hours]

Bitcoin and Ethereum in financial markets, Stablecoins and Central Bank Digital Currencies (CBDCs), Decentralized Finance (DeFi) concepts: lending, borrowing, decentralized exchanges, yield farming, Risks and challenges in cryptocurrency adoption

UNIT V

Blockchain in Banking and Capital Markets

[06 Hours]

Blockchain-based trade finance and supply chain finance, Securities settlement and clearing systems, Blockchain in derivatives and commodity trading, Case studies from JP Morgan, Ripple, and SWIFT GPI

Textbooks

1. Narayanan, A., Bonneau, J., Felten, E., Miller, A., & Goldfeder, S. – “*Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction*”, Princeton University Press.
2. Swan, M. – “*Blockchain: Blueprint for a New Economy*”, O’Reilly Media.
3. Pilkington, M. – “*Blockchain Technology: Principles and Applications*”, Elsevier.

Reference Books

1. Antonopoulos, A.M. – “*Mastering Bitcoin: Unlocking Digital Cryptocurrencies*”, O’Reilly Media.
2. Antonopoulos, A.M., & Wood, G. – “*Mastering Ethereum: Building Smart Contracts and DApps*”, O’Reilly Media.
3. Tapscott, D., & Tapscott, A. – “*Blockchain Revolution: How the Technology Behind Bitcoin and Other Cryptocurrencies is Changing the World*”, Penguin.
4. Wattenhofer, R. – “*The Science of the Blockchain*”, Inverted Forest Publishing.
5. Drescher, D. – “*Blockchain Basics: A Non-Technical Introduction in 25 Steps*”, Apress.
6. Zhang, W., & Wang, L. – “*Blockchain and Distributed Ledger Technology Use Cases: Applications and Lessons Learned*”, Springer.

Course Title : Cloud Platforms: AWS/Azure/GCP

Semester : VI

Prerequisite : Cloud Computing

L – T – P : 2-0-0

Course Code : 25UD1246PEC605G

Credits : 2

Course Type : PEC

Course Objective:

1. To introduce major cloud service providers and their core architectures.
2. To provide hands-on understanding of AWS, Azure, and GCP platforms.
3. To enable students to create and manage cloud infrastructure and services.
4. To teach platform-specific tools for compute, storage, networking, and monitoring.
5. To familiarize students with deployment, security, and cost optimization techniques on cloud platforms

Course Outcomes:

After learning the course, the students will be able to:

1. Compare features, architecture, and service offerings across AWS, Azure, and GCP.
2. Deploy virtual machines, manage storage, and configure networks on cloud platforms.
3. Use platform-specific tools for monitoring, autoscaling, and resource optimization.
4. Apply best practices for securing cloud environments and managing IAM policies.
5. Design and deploy scalable and cost-effective applications using public cloud platforms.

Course Contents:

UNIT I

Overview of Public Cloud Platforms:

[06 Hours]

Introduction to Cloud Platforms, AWS, Microsoft Azure, and Google Cloud Platform: Features and Architecture, Global Infrastructure and Regions, Overview of Free Tiers and Service Categories, CLI, SDKs, Web Portals.

UNIT II

Compute and Storage Services:

[06 Hours]

AWS EC2, Azure Virtual Machines, GCP Compute Engine; Storage Services: AWS S3, Azure Blob Storage, GCP Cloud Storage, Block vs Object vs File Storage; Launching and managing instances; Auto-scaling, Load Balancers, Snapshots

UNIT III

Networking and Identity Management:

[06 Hours]

VPC, Subnets, Security Groups, Firewalls, Load Balancing and DNS: Route 53 (AWS),

Azure DNS, GCP Cloud DNS, Identity & Access Management (IAM), Role-Based Access Control (RBAC), Shared Responsibility Model

UNIT IV

Platform-specific Tools and DevSecOps Integration: Monitoring and Logging: [06 Hours]

CloudWatch, Azure Monitor, GCP Operations, Resource Management: CloudFormation (AWS), ARM Templates (Azure), Deployment Manager (GCP), CI/CD Tools and IntegrationContainers and Kubernetes Support on AWS (EKS), Azure (AKS), GCP (GKE)

UNIT V

Cloud Security, Billing, and Use Cases: [06 Hours]

Security Best Practices, Encryption: At rest and in transit, Cost Estimation and Billing Tools, Cloud Use Cases in Industry (Startups, Enterprises, Government), Final Capstone Discussion / Case Study

Textbooks:

1. Mark Wilkins, *“Learning Amazon Web Services (AWS): A Hands-On Guide to the Fundamentals of AWS Cloud”*, Pearson Education, 1st Edition, 2019.
2. Jack Hyman, *“Microsoft Azure for Dummies”*, Wiley, 2nd Edition, 2022.
3. Dan Sullivan, *“Official Google Cloud Certified Associate Cloud Engineer Study Guide”*, Wiley, 1st Edition, 2019.

Reference books:

1. Brian Casillas, *“AWS Certified Solutions Architect Official Study Guide”*, Wiley, 2nd Edition, 2020.

Course Title : Version Control with Git

Semester : VI

Prerequisite : DevOps Principles and Practices

L – T – P : 2-0-0

Course Code : 25UD1246PEC605H

Credits : 2

Course Type : PEC

Course Objective:

1. Understand the fundamental role of version control systems (VCS) in modern software development and collaboration.
2. Gain proficiency in using Git from the command line to manage projects.
3. Master essential Git commands for the basic workflow: adding, committing, and managing files.
4. Learn how to work with branches for isolated development and use merging and rebasing to integrate changes.
5. Utilize a remote repository platform like GitHub for collaborative projects and understand the pull request workflow.

Course Outcomes:

After learning the course, the students will be able to:

1. Explain the core concepts of Git's data model, including commits, trees, and blobs.
2. Apply the basic Git workflow to track changes in a local repository.
3. Create and manage branches for feature development and bug fixes.
4. Resolve merge conflicts and use rebasing to maintain a clean project history.
5. Collaborate with others using a remote repository by cloning, fetching, pushing, and creating pull requests

Course Contents:

UNIT I

Introduction to Version Control

[06 Hours]

What is Version Control? Importance and benefits. Types of VCS: Centralized vs. Distributed. Introduction to Git: History, philosophy, and basic architecture. Setting up Git: Installation and initial configuration (git config)

UNIT II

The Basic Git Workflow

[06 Hours]

Initializing a Git repository (git init). The three states of Git: Working directory, Staging area (Index), and Repository. Basic commands: git add, git commit, git status, git log. Viewing differences: git diff.

UNIT III

Working with Branches

[06 Hours]

The concept of branching in Git. Creating, switching, and deleting branches (git branch, git checkout). Basic branching strategies: main branch, feature branches. Merging branches (git merge).

UNIT IV

Resolving Conflicts and Advanced Branching

[06 Hours]

Understanding and resolving merge conflicts. The concept of rebasing and its use cases. Comparing git merge vs. git rebase. Stashing changes (git stash).

UNIT V

Remote Repositories and Collaboration

[06 Hours]

The role of remote repositories (e.g., GitHub). Connecting local and remote repositories. Pushing and pulling changes (git push, git pull, git fetch). The Pull Request workflow for collaborative development. Forking a repository.

Text Books

1. Chacon, S., & Straub, B. (2014). *“Pro Git”* (2nd ed.). Apress.
2. Silverman, R. E. (2013). *“Git pocket guide: A working introduction”*n (1st ed.). O'Reilly Media.
3. Skoulikari, A. (2023). *“Learning Git: A hands-on and visual guide to the basics of Git”* (1st ed.)

Reference Books

1. O'Reilly, C., & Heltzel, M. (2020). *Git and GitHub: “A Guide to the Tools You Need to Be a Professional Developer”*. Packt Publishing.
2. Kim, G., Humble, J., Debois, P., & Willis, J. (2021). *“The DevSecOps handbook: How to create world-class agility, reliability, and security in technology organizations”* (2nd ed.). IT Revolution Press

Course Title : Quantum Computing Algorithms

Semester : VI

Prerequisite : Quantum Computing

L – T – P : 2-0-0

Course Code : 25UD1246PEC605I

Credits : 2

Course Type : PEC

Course Objectives

The course aims to:

1. Introduce the foundational principles and postulates of quantum mechanics relevant to quantum computation.
2. Explain qubit representation, quantum gates, circuits, and fundamental quantum communication protocols.
3. Develop understanding of major quantum algorithms, including Deutsch-Jozsa, Simon's, Grover's, and Shor's algorithms.
4. Familiarize students with advanced topics such as density matrices, projective measurements, and Bell's inequalities.
5. Introduce the basics of Quantum Machine Learning (QML) and its applications in data processing and optimization.

Course Outcomes

Upon successful completion of the course, students will be able to:

1. Describe the postulates of quantum mechanics and represent quantum states using Bloch sphere and density matrices.
2. Apply basic quantum gates and build quantum circuits for computation and communication protocols such as teleportation and dense coding.
3. Analyze and implement foundational quantum algorithms for search, period finding, and factorization.
4. Interpret Bell's inequalities, EPR paradox, and measurement theory in quantum systems.
5. Implement basic QML algorithms such as QPCA and QNN, and design variational quantum circuits for simple machine learning tasks.

Course Contents:

UNIT I

[06 Hours]

Why Quantum Computing?, Postulates of Quantum Mechanics – I, Postulates of Quantum Mechanics – II, Qubits and Bloch Sphere, Basic Quantum Gates, Quantum Circuits , No Cloning Theorem and Teleportation , Dense coding

UNIT II

[06 Hours]

Density Matrix-I, Density Matrix – II, Projective measurement, POVM, EPR and Bell's Inequalities-I, Bell's Inequalities – II

UNIT III

[06 Hours]

Deutsch Algorithm , Deutsch-Jozsa Algorithm , Simon Problem , Grover's Search, Algorithm – I , Grover's Search Algorithm –II , Grover's Search Algorithm –III , Grover's, Search Algorithm –IV

UNIT IV

[06 Hours]

Quantum Fourier Transform –I , Quantum Fourier Transform –II , Period Finding, Method of Continued Fraction , Shor's Factorization Algorithm

UNIT V

[06 Hours]

Basics of Quantum Machine Learning (QML), Quantum data encoding and feature maps, Quantum Principal Component Analysis (QPCA), Quantum Neural Networks (QNN) basics Variational circuits for ML tasks.

Textbooks

1. Nielsen, M.A., & Chuang, I.L., "*Quantum Computation and Quantum Information*", Cambridge University Press.
2. Yanofsky, N.S., & Mannucci, M.A., "*Quantum Computing for Computer Scientists*", Cambridge University Press.
3. Schuld, M., & Petruccione, F., "*Supervised Learning with Quantum Computers*", Springer.

Reference Books

1. Kaye, P., Laflamme, R., & Mosca, M., “*An Introduction to Quantum Computing*”, Oxford University Press.
2. Rieffel, E.G., & Polak, W.H., “*Quantum Computing: A Gentle Introduction*”, MIT Press.
3. Benenti, G., Casati, G., & Strini, G., “*Principles of Quantum Computation and Information*”, Vols. 1 & 2, World Scientific.
4. Guerreschi, G.G., & Matsuura, A.Y., “*Quantum Computing: From Linear Algebra to Physical Realizations*”, Morgan & Claypool.
5. Schuld, M., “*Machine Learning with Quantum Computers*”, Springer.

Course Title : User Interface Design

Semester : VI

Prerequisite : Human Computer Interaction

L – T – P : 2-0-0

Course Code : 25UD1246PEC605J

Credits : 2

Course Type : PEC

Course Objectives:

1. Understand the fundamental principles and guidelines for designing effective user interfaces.
2. Apply design elements such as layout, color, typography, and visual hierarchy in UI creation.
3. Use modern UI design and prototyping tools (e.g., Figma, Adobe XD) to create user interfaces.
4. Integrate accessibility principles into user interface designs to cater to diverse user needs.
5. Evaluate user interfaces using usability testing methods and heuristics.

Course Outcomes:

After learning the course, the students will be able to:

1. Describe the core principles and elements of user interface design.
2. Analyze interface design patterns for usability and effectiveness.
3. Create wireframes and interactive prototypes using design tools.
4. Apply accessibility guidelines to build inclusive user interfaces.
5. Evaluate user interfaces using usability testing and heuristic methods.

Course Contents:

UNIT I

Introduction to User Interface Design

[06 Hours]

Importance of good UI in HCI, Difference between UI and UX, Principles of good interface design (consistency, feedback, visibility, simplicity)

UNIT II

Principles of UI Design:

[06 Hours]

Visual design principles: Alignment, Proximity, Contrast, Repetition, Layouts and grids in interface design, Color theory, typography, iconography, Feedback and affordance in UI

UNIT III

Prototyping and UI Tools:

[06 Hours]

Low-fidelity vs High-fidelity prototypes, Introduction to tools: Figma, Adobe XD
Wireframing techniques and best practices, Designing screens: Login, Dashboard
Navigation

UNIT IV

Accessibility in UI Design

[06 Hours]

Introduction to accessibility: WCAG Guidelines, Designing for visual, auditory, motor, and cognitive impairments, Inclusive design and ethical considerations, Tools for accessibility testing

UNIT V

UI Evaluation and Testing

[06 Hours]

Usability goals and UI evaluation methods, Cognitive walkthrough and heuristic evaluation, User feedback collection and analysis, Metrics: Efficiency, error rate, satisfaction, learnability

Text Books:

1. Alan Dix et al., “*Human-Computer Interaction*”, Pearson
2. Jenifer Tidwell, “*Designing Interfaces*”, O’Reilly
3. Don Norman, “*The Design of Everyday Things*”, Basic Books
4. WCAG 2.1 Guidelines, W3C
5. Online Docs: Figma, Adobe XD, Material Design Guidelines

Reference Books:

1. Ben Shneiderman et al., “*Designing the User Interface: Strategies for Effective Human-Computer Interaction*”, Pearson
2. Bill Buxton, “*Sketching User Experiences: Getting the Design Right and the Right Design*”, Morgan Kaufmann
3. Steve Krug, “*Don’t Make Me Think: A Common Sense Approach to Web Usability*”, New Riders
4. Jeff Johnson, “*Designing with the Mind in Mind: Simple Guide to Understanding User Interface Design Guidelines*”, Morgan Kaufmann
5. Bruce Tognazzini, “*Tog on Interface*”, Addison-Wesley

Course Title : Software Testing

Semester : VI

Prerequisite :

L – T – P : 2-0-0

Course Code : 25UD1246PEC605K

Credits : 2

Course Type : PEC

Course Objective:

1. To study fundamental concepts in software testing, including software testing objectives, processes, criteria, strategies, and methods.
2. To learn planning of a test project, designing test cases and test data, conducting test operations, managing software problems and defects, and generating a test report.
3. To develop an understanding of the meaning and importance of quality in relation to software systems and the software development process.
4. To study issues and techniques for implementing and managing software quality assurance processes and procedures.

Course Outcomes:

After learning the course, the students will be able to:

To apply software testing knowledge and its processes to software applications.

2. To identify various software testing problems.

3. To solve software testing problems by designing and selecting software test models, criteria, strategies and methods.

4. To apply the techniques learned to improve the quality of software development.

5. To prepare a software quality plan for a software project.

Course Contents:

UNIT I

Principles of Testing Software development life cycle model:

[06 Hours]

Phases of software project, Quality, Quality assurance and quality control, Testing, Verification

and validation, Process models to represent various phases, Life cycle models, Software testing

life cycle. White Box Testing (WBT) and Black Box Testing: Static testing, Structural testing, Challenges in WBT. Black box testing: Black box testing process.

UNIT II

Integration Testing:

[06 Hours]

Definition, As a type of testing: Top-down integration, Bottom-up integration, Bidirectional integration, System integration, Choosing integration method, As a phase of testing, Scenario testing: System scenarios, Use case scenarios, Defect bash.

UNIT III

System and Acceptance Testing: [06 Hours]

Functional Vs non Functional, Functional system testing, Nonfunctional system testing, Acceptance testing.

UNIT IV

Performance testing, Regression testing, Internationalization testing, Adhoc testing:

[06 Hours]

Factors governing performance of testing, Methodology, tools and process for performance testing. Regression Testing: Introduction, Types of Regression testing, Regression testing process. Adhoc testing: Introduction, Buddy testing, Pair testing, Exploratory testing, Iterative testing, Agile and Extreme testing, XP work flow, Defect seeding.

UNIT V

Testing Object Oriented Software: [06 Hours]

Introduction, Comparison of object oriented and procedural software, System testing example,

Unit testing of classes, Tools for testing object oriented software, Testing web applications.

Textbooks:

1. Srinivasan Desikan, Gopaldaswamy Ramesh, *“Software Testing: Principles and Practices”*, Pearson publication, 2nd Edition, 2006.

Reference books:

1. Louise Tamres, *“Introducing Software Testing”*, Pearson publication, 2002.
2. Boris Beizer, *“Software Testing Techniques”*, Dreamtech press, 2nd Edition, 2014.

Course Title : Wireless and Mobile Communication

Semester : VI

Prerequisite : Computer Networks

L – T – P : 2-0-0

Course Code : 25UD1246PEC605L

Credits : 2

Course Type : PEC

Course Objective:

1. Introduce the fundamentals of wireless communication systems and mobile network architectures
2. Explain radio propagation models, modulation techniques, and multiple access methods used in wireless systems.
3. Provide an understanding of cellular concepts, frequency reuse, handoff, and interference management.
4. Familiarize students with various wireless communication standards and mobile network generations (1G to 5G).
5. Develop knowledge of emerging wireless technologies, mobility management, and security issues in mobile networks.

Course Outcomes:

Upon successful completion of this course, students will be able to:

1. Describe wireless communication principles, spectrum allocation, and frequency management.
2. Analyze cellular system design parameters such as frequency reuse, capacity, and handoff strategies.
3. Compare different modulation and multiple access techniques used in wireless systems.
4. Evaluate the architecture, protocols, and services of various mobile communication generations.
5. Apply knowledge of mobility management, security, and emerging wireless standards to real-world scenarios

Course Contents:

UNIT I

Introduction to Wireless Communication

[06 Hours]

Overview of wireless communication systems, Evolution of mobile communication: 1G to 5G, Wireless network types: WLAN, WPAN, WMAN, WWAN; Frequency spectrum for wireless communication, Standards and regulatory bodies

UNIT II

Radio Propagation and Modulation Techniques

[06 Hours]

Characteristics of radio waves; Propagation models: Free space, two-ray ground reflection, fading; Doppler shift and multipath propagation; Modulation techniques: ASK, FSK, PSK, QAM; Spread

UNIT III

Cellular Concepts and Multiple Access Techniques

[06 Hours]

Cellular system architecture; Frequency reuse, cell splitting, sectoring, Handoff strategies and power control, Interference and capacity enhancement techniques; Multiple access techniques: FDMA, TDMA, CDMA, OFDMA

UNIT IV

Mobile Communication Systems and Standards

[06 Hours]

GSM architecture and services, GPRS and EDGE technologies, UMTS (3G) and LTE (4G) architectures, 5G New Radio (NR) basics and features, Introduction to mobile IP and VoLTE

UNIT V

Wireless Networks, Mobility, and Security

[06 Hours]

Wireless LAN (IEEE 802.11) architecture and protocols, Bluetooth and ZigBee overview, Mobility management in mobile networks, Security threats in wireless communication, Authentication, encryption, and secure handoff.

Textbooks:

1. Theodore S. Rappaport, “*Wireless Communications: Principles and Practice*,” Pearson Education, 2nd Edition, 2010.
2. Jochen Schiller, “*Mobile Communications*”, Pearson Education, 2nd Edition, 2012.

Reference books:

1. Kaveh Pahlavan & Prashant Krishnamurthy, “*Principles of Wireless Networks*”, Pearson Education, 2009.
2. Andrea Goldsmith, “*Wireless Communications*”, Cambridge University Press, 2005.
3. Vijay Garg, “*Wireless Communications and Networking*”, Morgan Kaufmann, 2007.
4. William Stallings, “*Wireless Communications & Networks*”, Pearson Education, 2nd Edition, 2005.

Course Title : Algorithmic Finance

Semester : VI

Prerequisite : Finance Engineering

L – T – P : 2-0-0

Course Code : 25UD1246PEC605M

Credits : 2

Course Type : PEC

Course Objectives

By the end of this course, students will be able to:

1. Understand modern portfolio theory and advanced optimization techniques for asset allocation.
2. Gain proficiency in financial data acquisition, preprocessing, and analysis from multiple sources.
3. Apply quantitative models for risk estimation, performance evaluation, and scenario analysis.
4. Integrate risk management strategies using statistical, optimization, and simulation methods.
5. Explore financial sentiment analysis techniques and their impact on investment decisions.

Course Outcomes

Upon successful completion of this course, students will be able to:

1. Formulate and solve portfolio optimization problems using Markowitz, CVaR, and Black-Litterman models.
2. Evaluate portfolio performance using advanced risk-adjusted metrics and drawdown analysis.
3. Model and simulate portfolio risk using Monte Carlo and copula-based dependency structures.
4. Analyze financial market sentiment and integrate it into investment strategies.
5. Present data-driven investment recommendations using quantitative and qualitative analysis.

Course Contents:

UNIT I

Portfolio optimization

[06 Hours]

Portfolio optimization using Markowitz, CVaR, Black-Litterman, Risk Parity with [Riskfolio-Lib](#), mean-variance optimization, Black-Litterman with [PyPortfolioOpt](#), portfolio performance tear sheets with risk metrics, drawdowns, and Sharpe/Sortino ratios with [QuantStats](#), Analyze and

visualize portfolio performance, risk metrics, and return series with `ffn`

UNIT II

Data Access & Preprocessing

[06 Hours]

Download historical market data from Yahoo Finance with `yfinance`, Access data from sources like FRED, World Bank, etc. with `pandas-datareader`, Core numerical computation (e.g., returns, covariance) and manipulation with `pandas` and `numpy`

UNIT III

Risk Modeling, Simulation & Monte Carlo:

[06 Hours]

Machine learning models for risk classification, prediction, and clustering with `scikit-learn`, Modeling dependency and tail risks using copula functions with `copulas`, Sensitivity analysis for risk modeling with `SALib`, Random number generation for Monte Carlo with `NumPy` / `SciPy`

UNIT IV

Optimization:

[06 Hours]

Convex optimization in mean-variance, CVaR, robust optimization with `cvxpy`, General-purpose optimization solving with `scipy.optimize`, Linear/Mixed Integer Programming with `pulp`, Advanced modeling of linear and nonlinear optimization problems with `pyomo`

UNIT V

Financial Sentiment Analysis Models:

[06 Hours]

financial news and earnings calls analysis with `FinBERT`, sentiment labeling and data integration for financial use cases with `FinRL-Meta` / `FinNLP`, Rule-based sentiment analysis with `VADER` (from `NLTK`), Basic polarity/subjectivity score — quick prototyping with `TextBlob`, Summarize, interpret or classify financial news sentiment using GPT with `GPT models via OpenAI API`.

Textbooks:

1. Jon Danielsson, “*Financial Risk Forecasting: The Theory and Practice of Forecasting Market Risk with Implementation in R and Matlab*”, Wiley, 2011.
2. Marcos López de Prado, “*Advances in Financial Machine Learning*”, Wiley, 2018.
3. Abraham Lioui and Patrice Poncet, “*Dynamic Asset Allocation with Forwards and Futures*”, Springer, 2013.
4. Guilherme Duarte, “*Hands-On Financial Trading with Python: A Practical Guide to Using Zipline and PyPortfolioOpt*”, Packt Publishing, 2021.
5. Attilio Meucci, “*Risk and Asset Allocation*”, Springer, 2009.

6.

Reference books:

1. Emilio Barucci and Claudio Fontana, "***Mathematical Models for Financial Derivatives***", Springer, 2017.
2. Jason Strimpel, *Portfolio Optimization with Python: "An Introduction to Risk and Return Analysis with Riskfolio-Lib,"* Independently Published, 2021.
3. Emmanuel Ameisen, *Building Machine Learning Powered Applications: "Going from Idea to Product"*, O'Reilly Media, 2020.
4. Jacob Borne, *Financial Data Science with Python:: "Data Analysis, Preprocessing, and Machine Learning for Quantitative Finance"*, Apress, 2022.
5. Yi Dong and Tianyi Zhang, *Mastering Financial Risk Modeling: "Practical Guide to Monte Carlo Simulation, Copulas, and Risk Metrics"*, Wiley, 2020.

Course Title : Advanced Database Technology Lab

Semester : VI

Prerequisite : Advanced Database Technology

L – T – P : 0-0-2

Course Code : 25UD1246PCL608

Credits : 1

Course Type : PCC

List of Experiments:

1. Write advanced SQL queries using nested subqueries, joins, and aggregation for a university database.
2. Develop stored procedures and functions to automate fee calculation for a student management system.
3. Create database triggers to enforce business rules such as logging employee salary updates.
4. Use indexing and query optimization techniques to improve performance of a product inventory search.
5. Design and implement a MongoDB database for an online bookstore and perform CRUD operations.
6. Use MongoDB's aggregation pipeline to generate a sales summary report by category and month.
7. Model a hierarchical document structure in MongoDB for a course management system and apply indexing.
8. Simulate distributed database concepts by partitioning customer data across multiple tables and replicating them.
9. Design star and snowflake schemas for a retail sales data warehouse and implement them using SQL.
10. Perform ETL operations to extract CSV data, transform it, and load it into a data warehouse using Python or an ETL tool.
11. Implement OLAP operations (roll-up, drill-down, slice, dice) on a sales cube using sample multidimensional data.
12. Build a mini-project integrating SQL and NoSQL databases to manage structured and unstructured data for an e-commerce application.

Course Title : Advanced Web Technology Lab

Semester : VI

Prerequisite : Advanced Database Technology

L – T – P : 0-0-2

Course Code : 25UD1246PCL609

Credits : 1

Course Type : PCC

List of Experiments:

1. Design a responsive multipage personal portfolio using HTML5 and CSS3.
2. Create a web page that uses JavaScript to dynamically update content and validate user input.
3. Implement AJAX to fetch and display live data (e.g., news or weather) without reloading the page.
4. Develop a Single Page Application (SPA) for a task manager using React or Angular.
5. Create a RESTful API using Node.js and Express to manage product inventory.
6. Build a contact form that stores submitted data in a MongoDB or MySQL database.
7. Implement user authentication (login/logout) and session handling using Passport.js or Flask-Login.
8. Design a registration form with client-side and server-side input validation and sanitization.
9. Integrate a third-party weather or location API and display results dynamically.
10. Develop a feature to upload profile pictures and store them securely on the server.
11. Demonstrate prevention of SQL Injection and Cross-Site Scripting (XSS) in a user input form.
12. Deploy a full-stack e-commerce demo website with database and user login on a cloud platform (e.g., Heroku, Vercel, Netlify).

Course Title : Compiler Design Lab

Semester : VI

Prerequisite : Compiler Design

L – T – P : 0-0-2

Course Code : 25UD1246PCL6010

Credits : 1

Course Type : PCC

List of Experiments:

1. Lexical Analyzer
Write a program to implement a lexical analyzer for a given input program, identifying tokens such as keywords, identifiers, operators, and literals.
2. Finite Automata Simulation
Implement a DFA (Deterministic Finite Automaton) to recognize valid identifiers, numbers, and reserved words from a given source code.
3. Regular Expression to NFA/DFA Conversion
Write a program to convert a regular expression into an NFA and then to an equivalent DFA.
4. Symbol Table Implementation
Design and implement a symbol table with operations like insertion, deletion, search, and modification.
5. Recursive Descent Parser
Implement a recursive descent parser for a given grammar to check the syntax of given statements.
6. Operator Precedence Parser
Develop an operator precedence parser for arithmetic expressions.
7. LL(1) Parser
Construct an LL(1) parsing table and implement an LL(1) parser for a given grammar.
8. Intermediate Code Generation
Generate intermediate code (like three-address code) for simple arithmetic and boolean expressions.
9. Code Optimization Techniques
Implement basic code optimization techniques such as constant folding, dead code elimination, and strength reduction.
10. Target Code Generation
Generate target machine code or assembly code from intermediate representation for a given input program.

Course Title : Deep Learning Lab

Semester : VI

Prerequisite :

L – T – P : 0-0-2

Course Code : 25UD1246PCL6011

Credits : 1

Course Type : PCC

List of Experiments:

1. Install and run a simple neural network in TensorFlow/Keras or PyTorch.
2. Perform basic vector, matrix, and tensor operations using NumPy and PyTorch/TensorFlow.
3. Build a simple fully connected network for handwritten digit classification.
4. Compare ReLU, Sigmoid, Tanh, and Swish activations in a small network.
5. Apply L1, L2, and Dropout to prevent overfitting on a small dataset.
6. Train a network with and without Batch Normalization and compare training speed and accuracy.
7. Implement a simple CNN on MNIST
8. Use a pre-trained model (e.g., VGG16 or ResNet) for classifying a small custom image dataset.
9. Implement an RNN for predicting a simple text or numeric sequence.

Course Title : Cyber Forensics and Incident Response Lab

Semester : VI

Prerequisite : Cyber Security

L – T – P : 0-0-2

Course Code : 25UD1246PECL621A

Credits : 1

Course Type : PEC

List of Experiments:

1. Perform footprinting and reconnaissance on a given target domain using OSINT tools.
2. Conduct network scanning and host discovery in a controlled lab network.
3. Identify vulnerabilities in a target system using automated vulnerability scanners.
4. Perform service enumeration and banner grabbing to identify running applications.
5. Test a web application for OWASP Top 10 vulnerabilities in a controlled environment.
6. Detect and exploit SQL Injection on a vulnerable web application (lab setup).
7. Perform Cross-Site Scripting (XSS) testing and suggest mitigation techniques.
8. Conduct password attacks and hash cracking in an authorized lab environment.
9. Test wireless network security on a lab-configured access point.
10. Demonstrate a Man-in-the-Middle (MitM) attack and analyze captured traffic (lab setup).
11. Exploit and perform post-exploitation tasks on a vulnerable virtual machine.
12. Prepare a penetration testing report with evidence, risk assessment, and remediation steps.

Course Title : Data Warehousing and Data Mining Lab

Semester : VI

Prerequisite :

L – T – P : 0-0-2

Course Code : 25UD1246PECL621B

Credits : 1

Course Type : PEC

Lab Objectives:

1. To design and build a Data Warehouse/Data Mart using appropriate tools for real-world business applications.
2. To implement multidimensional data models and perform OLAP operations for analytical processing.
3. To apply data preprocessing, discretization, and visualization techniques for effective data analysis.
4. To implement classification, clustering, and association rule mining algorithms using tools such as WEKA.
5. To analyze and interpret results from data mining techniques and extract actionable insights from structured and unstructured data.

Lab Outcomes:

After learning the course, the students should be able:

1. To design and implement a data warehouse with appropriate schemas and perform OLAP operations for business analytics.
2. To apply data preprocessing, discretization, and visualization techniques to discover patterns and trends from datasets.
3. To implement and evaluate classification and clustering techniques using tools such as WEKA.
4. To apply association rule mining and text mining methods to extract meaningful insights from structured and unstructured data.

List of Experiments :

1. Build a Data Warehouse/Data Mart

Implement for a real-world problem (e.g., auto sales analysis)

2. Design Multidimensional Data Models

Create Star, Snowflake, and Fact Constellation schemas
Use enterprise domains like Banking, Healthcare, or Sales

3. Perform OLAP Operations

Apply Slice, Dice, Roll-up, Drill-down, and Pivot operations on OLAP cubes

4. Apply Data Visualization and Discretization Techniques

Use tool-based visualizations for trend analysis
Apply discretization filters on numerical attributes

5. To Implement Association Rule Mining algorithm (Apriori)

6. To implement K-means Clustering algorithm

7. To implement Bayesian Classification Algorithm.

8. Implement Association Rule Mining (Apriori Algorithm) using WEKA

Run Apriori algorithm on datasets with varying support/confidence
Analyze generated rules using WEKA

9. Decision Tree Classification using WEKA

Use ID3 and J48 algorithms
Study classifier outputs, entropy, and decision rules

10 .K-Means Clustering using WEKA

Run K-means algorithm with varying k values
Analyze clusters, centroids

11 .Case Study in Text Mining

Apply text mining techniques on unstructured text datasets

Derive patterns or topics using text analysis tools

Course Title : HSI Image Processing Lab
Prerequisite : Hyperspectral Imaging

Course Code : 25UD1246PECL621C

Course Type : PEC

Semester : VI

L – T – P : 0-0-2

Credits : 1

List of Experiments:

1. Perform spectral signature extraction for selected pixels/regions in a hyperspectral image.
2. Implement dimensionality reduction on hyperspectral data using Principal Component Analysis (PCA).
3. Apply Minimum Noise Fraction (MNF) transform to improve signal-to-noise ratio in hyperspectral imagery.
4. Conduct band selection to retain the most informative spectral bands for analysis.
5. Implement spectral angle mapper (SAM) classification for material identification.
6. Perform supervised classification using Support Vector Machine (SVM) on hyperspectral data.
7. Implement unsupervised classification using k-means or ISODATA clustering.
8. Detect specific targets in hyperspectral imagery using matched filter algorithms.
9. Compute vegetation indices (e.g., NDVI, SAVI) from hyperspectral data for vegetation health assessment.
10. Perform anomaly detection in hyperspectral imagery using RX Detector.
11. Apply spectral unmixing to estimate material abundance in mixed pixels.
12. Fuse hyperspectral data with multispectral or LiDAR data for improved feature extraction.

Course Title : Wireless Sensor Networks Lab

Semester : VI

Prerequisite :

L – T – P : 0-0-2

Course Code : 25UD1246PECL621D

Credits : 1

Course Type : PEC

Course Objective:

- 1.To understand the fundamentals of wireless sensor networks and its application to critical real time scenarios.
2. To study the various protocols at various layers and its differences with traditional
3. This course introduces the features of wireless sensor networks and their architecture.
4. The protocols of MAC and network layer are discussed in detail.
5. The course emphasizes localization and positioning schemes for real-time applications.

Course Outcomes:

After learning the course, the students will be able to:

1. Understand the basic features of wireless sensor networks.
2. Understand and apply the features of different wireless sensor architectures for real-world scenarios.
3. Understand and apply the protocols of MAC and network layer for real-world wireless sensor networks.
4. Understand and apply localization and positioning schemes.
5. Analyze the design of wireless sensor networks for real-time applications.

LIST OF EXPERIMENTS

- 1) To Study different types of Network cables (Copper and Fiber) and prepare cables (Straight and Cross) to connect Two or more systems. Use crimping tool to connect jacks. Use LAN tester to connect the cables.
- 2) Work with the commands Ping, Tracert, Ipconfig, pathping, telnet, ftp, getmac, ARP, Hostname, Nbtstat, netdiag, and Nslookup
- 3) To Find all the IP addresses on your network. Unicast, Multicast, and Broadcast on your network.
- 4) To Use Packet tracer software to build network topology and configure using Distance vector routing protocol.
- 5) Use Packet tracer software to build network topology and configure using Link State routing protocol.

- 6) To using JAVA RMI write a program to implement Basic Calculator. RMI called as remote method innovaction, by using rmi to develop calculator program in java
- 7) Implement a Chatting application using JAVA TCP and UDP sockets.
- 8) Hello command is used to know whether the machine at the other end is working or not. Echo command is used to measure the round trip time to the neighbour. Implement Hello and Echo commands using JAVA.
- 9) Using Wireshark perform the following operations: -
 - Inspect HTTP Traffic
 - Inspect HTTP Traffic from a Given IP Address,
 - Inspect HTTP Traffic to a Given IP Address,
 - Reject Packets to Given IP Address,
 - Monitor Apache and MySQL Network Traffic.
- 10) Install Network Simulator 2/3. Create a wired network using dumbbell topology. Attach agents, generate both FTP and CBR traffic, and transmit the traffic. Vary the data rates and evaluate the performance using metric throughput, delay, jitter and packet loss.
- 11) Create a static wireless network. Attach agents, generate both FTP and CBR traffic, and transmit the traffic. Vary the data rates and evaluate the performance using metric throughput, delay, jitter and packet loss.
- 12) Create a mobile wireless network. Attach agents, generate both FTP and CBR traffic, and transmit the traffic. Vary the data rates and evaluate the performance using metric throughput, delay, jitter and packet loss.
- 13) To simulate a Mobile Adhoc network (MANET) using NS2.
- 14) To implement a Transport Control Protocol in sensor network through the simulator.

Course Title : Natural Language Processing Lab

Semester : VI

Prerequisite :

L – T – P : 0-0-2

Course Code :25UD1246PECL621E

Credits : 1

Course Type : PEC

List of Experiments:

1. Study and implement Morphology
2. Study and implement Word Counting
3. Study and implement Word Sense Disambiguation
4. Study and implement Part-of-Speech Tagging
5. Study and implement N-Gram Model
6. Study and implement Entity Linking
7. Study and implement Emotion Detection
8. Study and implement Text Summarization

Course Title : Blockchain in Finance Lab

Semester : VI

Prerequisite :

L – T – P : 0-0-2

Course Code :25UD1246PECL621F

Credits : 1

Course Type : PEC

List of Experiments:

1. To implement a simple blockchain structure with transaction recording using Python.
2. To create and validate digital signatures for financial transactions.
3. To develop a proof-of-work consensus mechanism simulation.
4. To write and deploy a smart contract on Ethereum testnet for a simple payment system.
5. To implement a blockchain-based KYC verification system prototype.
6. To simulate cryptocurrency wallet creation and transaction processing.
7. To create a token using Ethereum ERC-20 standard for asset tokenization.
8. To implement a decentralized voting system for financial governance using smart contracts.
9. To develop a blockchain-based cross-border remittance system simulation.
10. To analyze and visualize blockchain transaction data from Bitcoin/Ethereum public ledgers.

Course Title : Cloud Platforms: AWS/Azure/GCP Lab

Semester : VI

Prerequisite :

L – T – P : 0-0-2

Course Code : 25UD1246PECL621G

Credits : 1

Course Type : PEC

Experiment 1: Getting Started with Public Cloud Platforms

- a) Create accounts on AWS Free Tier, Azure Student Subscription, and GCP Free Trial.
- b) Explore global regions and zones on each platform.
- c) Document available free-tier services.

Experiment 2: Introduction to CLI, SDKs, and Web Portal

- a) Install AWS CLI, Azure CLI, GCloud SDK.
- b) List compute/storage resources via CLI.

Compare interface usability (portal vs CLI).

Experiment 3: Launching Virtual Machines on AWS, Azure, and GCP

- a) Launch EC2 (AWS), Azure VM, GCP Compute Engine.
- b) Connect via SSH, install sample app (e.g., Apache server)
- c) Stop/start/reboot and terminate instances.

Experiment 4: Working with Cloud Storage (S3, Blob, GCP Storage)

- a) Create buckets/containers and upload files.
- b) Set file access permissions.

Generate pre-signed URLs and explore versioning.

Experiment 5: Auto-scaling and Load Balancing

- a) Create and configure Auto Scaling Groups (AWS/GCP).
- b) Configure Load Balancer and test scaling.
- c) View logs and analyze traffic distribution.

Experiment 6: Creating and Configuring a VPC / Virtual Network

- a) Create VPC, subnets, and route tables (AWS).
- b) Define firewall rules and test SSH access.
- c) Implement NAT Gateway or Internet Gateway.

Experiment 7: Configuring IAM and RBAC Policies

- a) Create IAM users, groups, and roles.
- b) Assign custom policies (e.g., read-only S3 access).
- c) Explore RBAC in Azure and GCP.

Experiment 8: Resource Management using Templates

- a) Use AWS CloudFormation to deploy a web server stack.
- b) Deploy a VM and storage via Azure Resource Manager Template.
- c) GCP for Use Deployment Manager in the same task

Course Title : Version Control with Git Lab

Semester : VI

Prerequisite :

L – T – P : 0-0-2

Course Code : 25UD1246PECL621H

Credits : 1

Course Type : PEC

Lab Experiments

1. **Getting Started:** Install Git, configure user settings, initialize a local repository, and make the first commit of a simple text file.
2. **Basic Workflow:** Create a few files, stage them selectively, and make multiple commits. Use git status and git log to inspect the repository.
3. **Inspecting Changes:** Modify a committed file and use git diff to view the changes before staging and after staging.
4. **Branching for a New Feature:** Create a new branch named feature-A, add a new file to it, and commit the changes. Switch back to the main branch and observe the file disappearing.
5. **Merging Branches:** Merge the feature-A branch into the main branch. Use git log --graph to visualize the merge.
6. **Simulating a Merge Conflict:** Modify the same line of code in both main and feature-B branches. Try to merge feature-B into main and manually resolve the conflict.
7. **Working with Remote Repositories:** Create a new repository on GitHub. Clone it to your local machine, make a commit, and push the changes back to the remote.
8. **Pulling and Pushing:** Work with a partner (or a second local copy of the repository). One person pushes changes, and the other uses git pull to get the latest updates.
9. **The Pull Request Workflow:** Fork a public repository on GitHub. Clone your fork, make a small change, and submit a pull request to the original repository's maintainer.
10. **Advanced Commands:** Use git reset to undo commits, and git stash to temporarily save changes without committing them.

Course Title : Quantum Computing Algorithms Lab

Semester : VI

Prerequisite :

L – T – P : 0-0-2

Course Code : 25UD1246PECL621I

Credits : 1

Course Type : PEC

List of Experiments:

1. To implement single- and two-qubit gates and prepare common quantum states.
2. To implement the quantum teleportation protocol for transferring an unknown qubit state.
3. To implement dense coding and demonstrate the no-cloning theorem.
4. To simulate projective measurements, construct POVMs, and perform density matrix tomography.
5. To test Bell's inequality using entangled states and measure the CHSH parameter.
6. To implement Deutsch and Deutsch–Jozsa algorithms and analyze their efficiency.
7. To implement Grover's search algorithm and study amplitude amplification.
8. To implement Quantum Fourier Transform (QFT) and quantum phase estimation.
9. To demonstrate Shor's algorithm (period finding) for factoring small integers.
10. To implement a Variational Quantum Classifier (VQC) or Quantum PCA for a simple dataset.

Course Title : User Interface Design Lab

Semester : VI

Prerequisite :

L – T – P : 0-0-2

Course Code : 25UD1246PECL621J

Credits : 1

Course Type : PEC

1. To introduce UI/UX design and prototyping tools.
2. To apply design principles for creating effective user interfaces.
3. To develop low- and high-fidelity wireframes and prototypes.
4. To incorporate accessibility features and test UI usability.

Course Outcomes:

After learning the course, the students will be able to:

1. Use professional tools to create UI wireframes and interactive prototypes
2. Apply visual design principles in user interface creation
3. Design inclusive interfaces adhering to accessibility standards
4. Conduct basic usability evaluations
5. Develop a complete UI flow for a real-world application
6. Design and implement a responsive website layout using HTML5, CSS3, and media queries.
7. Develop an interactive web form with real-time validation and user-friendly error messages.
8. Create a mobile app UI prototype using Figma/Adobe XD with proper navigation flow.
9. Implement a theme switcher (light/dark mode) for a sample web application.
10. Design a dashboard interface with charts, tables, and widgets for a data analytics app.
11. Create a multi-step registration form with progress indicators and animations.
12. Design and implement a navigation menu with dropdowns and hover effects.
13. Develop an accessible UI for a web application following WCAG guidelines.
14. Build an image gallery interface with filtering and sorting functionality.
15. Create an e-commerce product listing and detail page with interactive elements.
16. Develop a chatbot interface with a clean conversation design and message animations.
17. Conduct usability testing for a designed UI prototype and prepare a user feedback report.

Course Title : Software Testing Lab

Prerequisite :

L – T – P : 1-0-2

Course Code :25UD1246PECL621K

Credits : 2

Course Type : VSEC

List of Experiments:

1. Design and execute black-box test cases for a given software module using equivalence class partitioning and boundary value analysis.
2. Perform white-box testing using statement coverage, branch coverage, and path coverage for a sample program.
3. Conduct unit testing of a Java/Python program using JUnit or PyTest.
4. Automate functional test cases for a web application using Selenium WebDriver.
5. Perform regression testing on an updated version of a given software application.
6. Execute integration testing for multiple interacting software modules.
7. Conduct performance testing of a web application using JMeter.
8. Perform load and stress testing of a given application and record system behavior.
9. Validate input form fields using client-side and server-side validation techniques.
10. Test database operations (CRUD) for a given application and verify data integrity.
11. Conduct usability testing for a desktop or mobile application and prepare a user feedback report.
12. Prepare a detailed bug report and defect life cycle documentation for a tested software project.

Course Title : Wireless and Mobile Communication Lab

Semester : VI

Prerequisite :

L – T – P : 0-0-2

Course Code : 25UD1246PECL621L

Credits : 1

Course Type : PEC

List of Experiments:

1. **Study of Wireless Communication Tools** – Introduction to MATLAB, NS2/NS3, OMNeT++, or equivalent.
2. **Simulation of Radio Wave Propagation Models** – Free space, two-ray ground reflection, and fading.
3. **Implementation of Digital Modulation Techniques** – ASK, FSK, PSK, QAM using MATLAB/Scilab.
4. **Spread Spectrum Techniques** – DSSS and FHSS implementation and analysis.
5. **Cellular System Concepts** – Simulation of frequency reuse and cell splitting using a simulator.
6. **Handoff Simulation** – Demonstrating soft and hard handoff scenarios.
7. **Multiple Access Techniques** – Simulation and comparison of FDMA, TDMA, CDMA, OFDMA.
8. **GSM Network Simulation** – Call setup, location update, and SMS transmission using open-source GSM tools.
9. **Wi-Fi Network Setup and Analysis** – Configuring WLAN and analyzing throughput using Wireshark.
10. **Bluetooth Communication** – Pairing and data exchange between devices.

Course Title : Algorithmic Finance Lab

Semester : VI

Prerequisite : Finance Engineering

L – T – P : 0-0-2

Course Code : 25UD1246PECL621M

Credits : 1

Course Type : PEC

List of Experiments

1. Design of a Portfolio Optimization Model using Markowitz Mean-Variance Theory with Riskfolio-Lib
2. Implementation of a Black-Litterman Portfolio Allocation Strategy using PyPortfolioOpt
3. Design of a CVaR-Based Portfolio Optimization Model using cvxpy
4. Implementation of a Risk Parity Portfolio Strategy and Performance Evaluation using QuantStats
5. Design of a Data Acquisition and Preprocessing Pipeline using yfinance and pandas-datareader
6. Implementation of Monte Carlo Simulation for Portfolio Risk and Return Analysis using NumPy/SciPy
7. Design of a Copula-Based Dependency and Tail Risk Modeling Framework using the copulas Library
8. Implementation of a Financial Sentiment Analysis Model using FinBERT and Integration with Portfolio Signals
9. Design of an Optimization Framework for Portfolio Allocation using Linear and Nonlinear Programming with pyomo
10. Implementation of a Portfolio Performance Analytics Dashboard using ffn and QuantStats
11. Design of a Robust Portfolio Optimization Model under Parameter Uncertainty using cvxpy
12. Implementation of Drawdown Analysis and Risk Metrics Computation with QuantStats
13. Design of an Automated Data Fetching and Storage System for Multi-Source Financial Data
14. Implementation of a Rolling-Window Portfolio Rebalancing Strategy using pandas and numpy

Course Title : Time Series Modeling and Forecasting

Course Code :25UD1246VSL607

Prerequisite :

Course Type : VSEC

Semester: VI

L – T – P : 1-0-2

Credits : 2

Course Objective:

1. To provide an understanding of the fundamental concepts of time series analysis and its historical context.
2. To teach the basics of Python programming and its application in time series analysis.
3. To equip students with techniques for exploratory time series data analysis, including autocorrelation and decomposition methods.
4. To explain advanced time series models, their properties, and relevant statistical tests.
5. To cover multivariate , state space and machine learning models, including model building, estimation methods, and forecasting techniques.

Course Outcomes:

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

1. Understand and apply univariate volatility models, including ARCH, GARCH, EGARCH, and TGARCH, to capture and forecast time-varying volatility in financial time series.
2. Analyze and model interdependencies between multiple time series using VAR and VARMA models, and evaluate causal relationships through Granger causality and impulse response analysis.
3. Formulate and estimate state space models, including Kalman filter applications and Markov switching models, for modeling time-varying parameters and regime shifts in time series data.
4. Diagnose nonstationarity in time series data, perform unit root and stationarity tests, and apply cointegration techniques to model long-run equilibrium relationships using Vector Error Correction Models (VECM).
5. Apply modern machine learning methods, including neural networks and deep learning algorithms, for time series forecasting, and implement solutions using tools like TensorFlow and Keras.

Course Contents:

UNIT I

Financial Time Series and Related Models

[6 Hours]

Stylized Facts of Financial Time Series , Examples of Return Series ,Stylized Facts of Financial Time Series , Contents ix 6.2 GARCH Models , ARCH Models ,GARCH , Estimation and Testing, Examples, TGARCH Models , An Example , Problems

UNIT II

Multivariate Time Series Analysis

[6 Hours]

Basic Concepts, Covariance and Correlation Matrix Functions, Stationarity and Vector White Noise, Sample Covariance and Correlation Matrices, Multivariate Portmanteau Test, VARMA Models,

Definitions, Properties, VAR Model Building and Analysis, Representation of VARMA Processes, VAR Model Building Steps, Granger Causality, Impulse Response Analysis, Examples, Problems

UNIT III

State Space Models and Markov Switching Models [6 Hours]

State Space Models and Representations, State Space, State Space Representations of Time Series, Kalman Recursions, Local-Level Model and SARIMAX Models, Local-Level Model, SARIMAX Models, Markov Switching Models, Definitions, Examples, Problems

UNIT IV

Non-stationarity and Cointegrations [6 Hours]

Stochastic Trend and Stochastic Seasonality, Deterministic Trend and Stochastic Trend, Deterministic Seasonality and Stochastic Seasonality, Contents, Brownian Motions and Simulation, Probability Space, Brownian Motions, Stationarity, Nonstationarity, and Unit Root Tests, Trend Stationarity and Difference Stationarity, Unit Root Tests, Stationarity Tests, Cointegrations and Granger's Representation Theorem, Spurious Regressions and I (d) Processes, Cointegrations, Granger's Representation Theorem, Estimation of Vector Error Correction Models, Real Case of Spurious Regression and Noncointegration, Problems ..

UNIT V

Modern Machine Learning Methods for Time Series Analysis [6 Hours]

Introduction, Brief History of Artificial Intelligence, AI in Time Series Analysis, Artificial Neural Networks, Artificial Neural Network Developments, Neural Network Models, Deep Learning and Backpropagation Algorithms, What Is Deep Learning, Gradient Descent and Backpropagation Algorithms, Time Series Forecasting and TensorFlow, Time Series Forecasting, TensorFlow and Keras, Implementation and Example, Implementation Steps, An Example, Concluding Remarks .

Text Books:

1. Changquan Huang, AllaPetukhina, "*Applied Time Series Analysis and Forecasting with Python*", Springer.
2. Brockwell, P.J. and R.A. Davis, "*Time Series: Theory and Methods*", 2nd edition, Springer Verlag, 2009.
3. Cowpertwait, Paul S.P and Andrew V. Metcalfe, "*Introductory Time Series with R*", Springer, 2008.
4. Kirchgassner, G. and J. Wolters, "Introduction to Modern Time Series Analysis", Springer, 2007.

Reference Books:

1. G.E.P., Jenkins, G.M. and Reinsel, G.C. , "*Time Series Analysis, Forecasting and Control, Box*", Prentice-Hall, 1994.
2. Chris Chatfield, and Haipeng Xing , "*The Analysis of Time Series: An Introduction with R*", CRC Press, London., 2019.
3. Galit Shmueli and Kenneth C. Lichtendahl Jr, "*Practical Time Series Forecasting with R: A hands-on Guide*", Axelrod Schnall Publishers, 2016.

Multidisciplinary Minor II [Semester VI]

Sr. No	Course Title	Course Code	Course Type	Teaching Scheme – Hours per week				Examination Scheme - Marks					Total
				L	T	P	CR	CA	MSE	ESE	TW	PR/OR	
	Any One												
1	Advanced Database Technology	25UD1246MD606A	MD-M	2	0	0	2	20	20	60	-	-	100
2	Advanced Web Technologies	25UD1246MD606B											
3	Compiler Design	25UD1246MD606C											
4	Deep Learning	25UD1246MD606D											
5	Ethical Hacking	25UD1246MD606E											
6	Data Warehousing and Data Mining	25UD1246MD606F											
7	Image Processing Techniques for Hyperspectral Data	25UD1246MD606G											
8	Wireless Sensor Networks	25UD1246MD606H											
9	Advanced Machine Learning	25UD1246MD606I											
10	Blockchain in Finance	25UD1246MD606J											
11	Virtualization and Containers	25UD1246MD606K											
12	Version Control with Git	25UD1246MD606L											
13	Quantum Computing Algorithms	25UD1246MD606M											
14	User Interface Design	25UD1246MD606N											
15	Software Testing	25UD1246MD606O											
16	Wireless and Mobile Communication	25UD1246MD606P											
17	Algorithmic Finance	25UD1246MD606Q											

B. Tech. IT (Honors) [Semester VI]

Sr. No	Course Title	Course Code	Course Type	Teaching Scheme – Hours per week				Examination Scheme - Marks					Total
				L	T	P	CR	CA	MSE	ESE	TW	PR/OR	
	Any One		Honors	03	00	00	03	20	20	60	-	-	100
1	Internet of Things (IoT) and Smart Systems	25UD1246HR601											
2	IT Laws, Ethics, and Digital Privacy	25UD1246HR602											
3	AR/VR Technologies and Applications	25UD1246HR603											

Course Title : Internet of Things (IoT) and Smart Systems

Semester: VI

Course Code : 25UD1246HR601

L – T – P : 3 – 0 – 0

Prerequisite : Microprocessor and Microcontroller Fundamentals

Credits : 3

Course Type : Honors

Course Objectives:

1. To introduce the fundamental concepts and architecture of the Internet of Things (IoT).
2. To familiarize students with the various communication protocols and networking technologies used in IoT systems.
3. To enable students to work with IoT hardware platforms and sensors for real-time data acquisition and control.
4. To explore the design and development of smart systems and IoT applications in domains such as healthcare, home automation, and energy.
5. To impart knowledge about the security, privacy, and ethical concerns related to IoT.
6. To introduce students to emerging trends and technologies in IoT, such as edge computing, AI integration, and 5G.

Course Outcomes:

After learning the course, the students will be able to:

1. To understand the basics and evolution of IoT and identify its real-world applications.
2. To analyze the architecture and communication protocols used in IoT systems.
3. To demonstrate working knowledge of IoT hardware and software platforms.
4. To explore and evaluate smart systems across various domains.
5. To evaluate security issues in IoT and assess future trends in smart systems.

Course Contents:

UNIT I

Introduction to IoT:

[08 Hours]

\Overview of IoT: Definition, Characteristics, and Scope, Evolution of IoT: M2M to IoT, IoT Applications: Smart Home, Smart City, Healthcare, Industrial IoT, Agriculture, Transportation, IoT Reference Architectures (ARM, ITU-T, etc.), Challenges and Issues in IoT

UNIT II

IoT Architecture and Protocols:

[08 Hours]

IoT Architecture: Perception, Network, Middleware, Application Layers, Communication Protocols: Physical & Link Layer: RFID, IEEE 802.15.4, BLE, NFC Network Layer: IPv4/IPv6, 6LoWPAN, Transport Layer: TCP, UDP, Application Layer: MQTT, CoAP, AMQP, XMPP, Fog and Edge Computing in IoT

UNIT III

IoT Hardware and Software Platforms:

[08 Hours]

Sensors and Actuators: Types, Working, Interfacing, Microcontrollers & Boards: Arduino, Raspberry Pi, ESP32, NodeMCU, Operating Systems for IoT: TinyOS, RIOT, Contiki, Programming for IoT: Python/C basics, Arduino IDE, IoT Middleware & Platforms: ThingsBoard, Node-RED, Blynk, Google Firebase

UNIT IV

Smart Systems and Applications:

[08 Hours]

Introduction to Smart Systems: Definition, Features, Components of Smart Systems: Sensing, Processing, Communication, Control, Smart Home and Building Automation, Smart Health and Wearables, Smart Energy Systems (Smart Grid, Smart Meters), Smart Cities and Infrastructure

UNIT V

Security, Privacy and Future Trends:

[08 Hours]

IoT Security Requirements and Challenges, Common Attacks:

Spoofing, Replay, Eavesdropping, DoS, Security Solutions: Authentication, Access Control, Encryption, Data Privacy Issues and Mitigation, Future Trends: AI in IoT, Digital Twins, 5G & IoT, Blockchain for IoT

Text Books:

1. Arshdeep Bahga, Vijay Madiseti, *“Internet of Things: A Hands-On Approach”*, Universities Press, 1st Edition, 2015.
2. Adrian McEwen, Hakim Cassimally, *“Designing the Internet of Things”*, Wiley India, 1st Edition, 2013.

Reference Books:

1. Rajkumar Buyya, Amir Vahid Dastjerdi, *“Internet of Things: Principles and Paradigms”*, Morgan Kaufmann (Elsevier), 1st Edition, 2016.
2. Jan Holler, Vlasios Tsiatsis, Catherine Mulligan, Stamatis Karnouskos, Stefan Avesand, David Boyle, *“From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence”*, Academic Press, 1st Edition, 2014.
3. Donald Norris, *“The Internet of Things: Do-It-Yourself Projects with Arduino, Raspberry Pi, and Beaglebone Black”*, McGraw-Hill Education, 1st Edition, 2015.
4. Ovidiu Vermesan, Peter Friess, *“Internet of Things – From Research and Innovation to Market Deployment”*, River Publishers, 1st Edition, 2014.

Course Title: IT Laws, Ethics and Digital Privacy
Course Code 25UD1246HR502
Prerequisite Internet and Information Security
Course Type Honors

Semister VI
L – T – P 3– 0 – 0
Credits 3

Course Objectives:

1. To provide understanding of legal and ethical issues related to information technology.
2. To educate students on Indian and international IT laws, data protection regulations, and cybercrime frameworks.
3. To create awareness of digital privacy, surveillance, data ownership, and ethical use of technology.
4. To prepare students for ethical decision-making in digital environments.

Course Outcomes (COs):

Upon completion of the course, students will be able to:

1. Understand and explain key provisions of Indian IT laws and global frameworks.
2. Analyze ethical dilemmas in professional IT practice and decision-making.
3. Evaluate issues related to digital rights, surveillance, and privacy.
4. Apply knowledge of cyber laws and ethical principles to real-world IT case studies.
5. Demonstrate awareness of data protection laws, policies, and governance models.

UNIT I

Introduction to IT Laws and Ethics: [8 Hours]

Definition and Scope of Cyber Law, Need for IT laws and digital ethics in India, Professional and Ethical Responsibilities of IT professionals, Intellectual Property Rights (IPR), Copyrights, Patents

UNIT II

Indian Legal Framework [8 Hours]

Information Technology Act 2000 (and Amendments), Key Sections: Hacking, Identity Theft, Cyber Terrorism, Obscenity, Roles of Cert-In, Adjudication Process, Cyber Appellate Tribunal, Case Laws and Judicial Interpretations in India

Global Framework and International Treaties :

Budapest Convention on Cybercrime, General Data Protection Regulation (GDPR) – Key Concepts, Cross-border Data Flow, Safe Harbor & Data Localization, Comparative Study: India vs. EU/US Laws

UNIT III

Ethics in Digital Society:

[8 Hours]

Digital Citizenship, Net Neutrality, Censorship, Ethics in Artificial Intelligence and Autonomous Systems, Misinformation, Deepfakes, Social Media Responsibility, Ethical Decision-Making Models in Technology Use

UNIT IV

Privacy and Data Protection

[8 Hours]

Concept of Digital Privacy and Consent, Indian Data Protection Bill – Key Provisions, Privacy by Design, Right to be Forgotten, Surveillance vs. Civil Liberties

UNIT V

Case Studies and Emerging Issues

[8 Hours]

Case Studies on Cybercrime, Data Breaches, Ethics Violations, Future Trends: Blockchain and Legal Challenges, Cybersecurity and National Sovereignty, Digital Trust and Ethics in Emerging Tech

Textbooks:

1. Pavan Duggal, *“Cyber Law: Indian and International Perspectives on Key Topics Including Data Protection and Privacy”*, Saakshar Law Publications, Latest Edition
2. Vakul Sharma, *“Legal Aspects of Information Technology”*, Universal Law Publishing, 6th Edition
3. George Reynolds, *“Ethics in Information Technology”*, Cengage Learning, 5th Edition
4. William Stallings, *“Information Privacy Engineering and Privacy by Design”*, Pearson, 2021

Reference Books:

1. *“Cybersecurity and Cyberlaw”* – Aparna Viswanathan, LexisNexis
2. *“Internet Law: Cases and Problems”* – James Grimmelman, Semaphore Press, 2022
3. *General Data Protection Regulation (GDPR)* – Official EU Publications
4. *Digital Personal Data Protection Act 2023* – Ministry of Electronics and Information Technology, Government of India
5. **IEEE & ACM Codes of Ethics and Professional Conduct**

Course Title: AR/VR Technologies and Applications

Semester

Course Code 25UD1246HR603

L – T – P 3– 0 – 0

Prerequisite Mathematics, Physics, Programming and Problem Solving, Engineering Graphics, Solid Modeling and Drafting, Numerical & Statistical Methods, Mechatronics, Artificial Intelligence & Machine Learning,

Course Objectives:

1. Learn the fundamental Computer Vision, Computer Graphics and Human-Computer interaction Techniques related to VR/AR
2. Review the Geometric Modeling Techniques
3. Review the Virtual Environment
4. Discuss and Examine VR/AR Technologies
5. Use of various types of Hardware and Software in Virtual Reality systems
6. Simulate and Apply Virtual/Augmented Reality to varieties of Applications

Course Outcomes:

On completion of the course the learner will be able to;

1. Understand fundamental Computer Vision, Computer Graphics and HumanComputer Interaction Techniques related to VR/AR
2. Understand Geometric Modeling Techniques
3. Understand the Virtual Environment
4. Analyze and Evaluate VR/AR Technologies
5. Apply various types of Hardware and Software in Virtual Reality systems
6. Design and Formulate Virtual/Augmented Reality Applications

Course Contents

UNIT I

Introduction to Virtual Reality (VR)

Virtual Reality and Virtual Environment, Computer graphics, Real time computer graphics, Flight Simulation, Virtual environment requirement, benefits of virtual reality, Historical development of VR, Scientific Landmark

UNIT II

Computer Graphics and Geometric Modelling

The Virtual world space, positioning the virtual observer, the perspective projection, human vision, stereo perspective projection, Color theory, Conversion From 2D to 3D, 3D space curves, 3D boundary representation, Simple 3D modelling, 3D clipping, Illumination models, Reflection models, Shading a

Igorithms, Geometrical Transformations: Introduction, Frames of reference, Modelling transformations, Instances, Picking, Flying, Scaling the VE, Collision detection

UNIT III

Virtual Environment

Input/Output Devices: Input (Tracker, Sensor, Digital Gloves, Movement Capture, Videobased Input, 3D Menus & 3D Scanner, etc.), Output (Visual/Auditory/Haptic Devices)

Generic VR system: Introduction, Virtual environment, Computer environment, VR technology, Model of interaction, VR Systems, Animating the Virtual Environment:

Introduction, The dynamics of numbers, Linear and Nonlinear interpolation, the animation of objects, linear and non-linear translation, shape & object in between, free from deformation, particle system Physical Simulation: Introduction, Objects falling in a gravitational field, Rotating wheels, Elastic collisions, projectiles, simple pendulum, springs, Flight dynamics of an aircraft

UNIT IV

Augmented Reality (AR)

Taxonomy, Technology and Features of Augmented Reality, AR Vs VR, Challenges with AR, AR systems and functionality, Augmented Reality Methods, Visualization Techniques for Augmented Reality, Enhancing interactivity in AR Environments, Evaluating AR systems

UNIT IV

Development Tools and Frameworks

Human factors: Introduction, the eye, the ear, the somatic senses Hardware: Introduction, sensor hardware, Head-coupled displays, Acoustic hardware, Integrated VR systems Software: Introduction, Modelling virtual world, Physical simulation, VR toolkits, Introduction to VRML

UNIT V

AR / VR Applications

Introduction, Engineering, Entertainment, Science, Training, Game Development

Books and other resources

Text Books:

1. Coiffet, P., Burdea, G. C., (2003), "*Virtual Reality Technology*," Wiley-IEEE Press, ISBN: 9780471360896
2. Schmalstieg, D., Höllerer, T., (2016), "*Augmented Reality: Principles & Practice*," Pearson, ISBN: 9789332578494
3. Norman, K., Kirakowski, J., (2018), "*Wiley Handbook of Human Computer Interaction*," Wiley-Blackwell, ISBN: 9781118976135
4. LaViola Jr., J. J., Kruijff, E., McMahan, R. P., Bowman, D. A., Poupyrev, I., (2017), "*3D User Interfaces: Theory and Practice*," Pearson, ISBN: 9780134034324
5. Fowler, A., (2019), "*Beginning iOS AR Game Development: Developing Augmented Reality Apps with Unity and C#*," Apress, ISBN: 9781484246672
6. Hassanien, A. E., Gupta, D., Khanna, A., Slowik, A., (2022), "*Virtual and Augmented Reality for Automobile Industry: Innovation Vision and Applications*," Springer, ISBN: 9783030941017

References Books:

1. Craig, A. B., (2013), ***“Understanding Augmented Reality, Concepts and Applications,”*** Morgan Kaufmann, ISBN: 9780240824086
2. Craig, A. B., Sherman, W. R., Will, J. D., (2009), ***“Developing Virtual Reality Applications, Foundations of Effective Design,”*** Morgan Kaufmann, ISBN: 9780123749437
3. John Vince, J., (2002), ***“Virtual Reality Systems,”*** Pearson, ISBN: 9788131708446
4. Anand, R., ***“Augmented and Virtual Reality,”*** Khanna Publishing House
5. Kim, G. J., (2005), ***“Designing Virtual Systems: The Structured Approach,”*** ISBN: 9781852339586
6. Bimber, O., Raskar, R., (2005), ***“Spatial Augmented Reality: Merging Real and Virtual Worlds,”*** CRC Press, ISBN: 9781568812304
7. O'Connell, K., (2019), ***“Designing for Mixed Reality: Blending Data, AR, and the Physical World,”*** O'Reilly, ISBN: 9789352138371
8. Sanni Siltanen, S., (2012), ***“Theory and applications of marker-based augmented reality,”*** Julkaisija –Utgivare Publisher, ISBN: 9789513874490

Web References:

1. Manivannan, M., (2018), ***“Virtual Reality Engineering,”*** IIT Madras, <https://nptel.ac.in/courses/121106013>
2. Misra, S., (2019), ***“Industry 4.0: Augmented Reality and Virtual Reality,”*** IIT Kharagpur, <https://www.youtube.com/watch?v=zLMgdYI82IE>
3. Dube, A., (2020), ***“Augmented Reality - Fundamentals and Development,”*** NPTEL Special Lecture Series, <https://www.youtube.com/watch?v=MGuSTAqlZ9Q>
4. <http://cambum.net/course-2.htm>

B. Tech. IT (By Research)[Semester VI]

Sr. No	Course Title	Course Code	Course Type	Teaching Scheme – Hours per week				Examination Scheme - Marks					Total
				L	T	P	CR	CA	MSE	ESE	TW	PR/OR	
	Any One		Research	03	00	00	03	20	20	60	-	-	100
1	Case Study Research	25UD1246RS601											
2	Entrepreneurship and Innovation	25UD1246RS602											

B. Tech. IT (By Research)[Semester VI]

Course Title:	Case Study Research	Semester	VI
Course Code	25UD1246RS601	Course Type	Honors
Prerequisite		L – T – P	3 – 0 – 0
Stream	Research	Credits	3

Course Objectives:

1. To introduce the foundations and philosophy of case study research.
2. To familiarize students with various types of case study designs.
3. To develop skills for conducting case study research including data collection and analysis.
4. To enable critical evaluation of case studies in academic and applied contexts.
5. To guide students in writing and presenting comprehensive case study reports.

Course Outcomes:

After learning the course, the students will be able to:

1. Explain the principles and types of case study research.
2. Design a case study aligned with research questions and objectives.
3. Apply appropriate data collection methods for case study research.
4. Analyze and interpret qualitative data in the context of case studies.
5. Critically evaluate and write structured case study reports.

Course Contents:

UNIT I

Introduction to Case Study Research: Definition, scope, and characteristics of case study research, Case study vs. other research methods (e.g., experiments, surveys, ethnography), When to use a case study, Types of case studies: Exploratory, Descriptive, Explanatory, Intrinsic, Instrumental, Strengths and limitations of the method.

UNIT II

Designing a Case Study: Developing a research problem and questions, Case selection: Single vs. multiple case studies, Units of analysis, Maintaining validity and reliability, Ethical considerations in case study research.

UNIT III

Data Collection Methods: Sources of data: Interviews, observations, documents, archival records, physical artifacts, Triangulation of data sources, Creating and using interview protocols, Maintaining a case study database, Recording and organizing field data.

UNIT IV

Data Analysis and Interpretation: Coding and categorizing qualitative data, Thematic analysis and pattern matching, Cross-case synthesis (for multiple case studies), Use of qualitative analysis tools (e.g., NVivo, MAXQDA, [ATLAS.ti](#)), Integrating theory and evidence

UNIT V

Writing and Reporting the Case Study: Structure of a case study report: Introduction, Method, Findings, Discussion, Conclusion, Writing for academic vs. industry audiences, Use of visuals: Timelines, flowcharts, matrices, Common errors in case writing, Referencing and citation styles (APA, IEEE, etc.)

Text Books:

1. Yin, Robert K. – “*Case Study Research and Applications: Design and Methods*”, SAGE Publications.
2. Stake, Robert E. “*The Art of Case Study Research*”, SAGE Publications.

Reference Books:

1. Creswell, John W. – “*Qualitative Inquiry and Research Design: Choosing Among Five Approaches*, SAGE”.

Course Title:	Entrepreneurship and Innovation	Semester	VI
Course Code:	25UD1246RS602	Course Type	Honors
Prerequisite		L – T – P	3 – 0 – 0
Stream	Research	Credits	3

Course Objectives:

1. Develop an entrepreneurial mindset among engineering students.
2. Introduce fundamentals of innovation, ideation, and opportunity recognition.
3. Equip students with tools to validate business models and pitch ideas.
4. Encourage application of technology in solving real-world problems.
5. To enable students to develop business models and pitch ideas.

Course Outcomes:

After learning the course, the students will be able to:

1. Understand the role of entrepreneurship in the economy.
2. Identify and evaluate business opportunities using innovation frameworks.
3. Develop a business model canvas for a startup idea.
4. Analyze startup case studies and understand funding mechanisms.
5. Present and pitch entrepreneurial ideas effectively.

Course Contents:

UNIT I

Introduction to Entrepreneurship Innovation and Creativity:

Definitions, types of entrepreneurs, entrepreneurial mindset, myths of entrepreneurship, role in economic development. Innovation vs. Invention, sources of innovation, disruptive innovation, TRIZ, idea generation techniques (SCAMPER, Brainstorming).

UNIT II

Identifying Opportunities:

Opportunity recognition, problem–solution fit, market research, trends and gap analysis, technology-driven opportunities.

UNIT III

Business Models: Elements of a business model, Business Model Canvas (BMC), Lean Startup methodology, value proposition design.

Financing the Startup Bootstrapping, angel investors, venture capital, crowdfunding, government schemes (e.g., Startup India, MSME), incubators and accelerators

UNIT IV

Entrepreneurial Ecosystem and Case Studies: National and Global Startup Ecosystem, Role of Incubators, Accelerators, and Startup Hubs, Case Studies of Successful Entrepreneurs and Startups, Challenges and Failures in Entrepreneurship, Ethics and Social Responsibility in Entrepreneurship.

Legal and Ethical Issues: Company formation, IP rights, NDAs, patents, ethics in entrepreneurship

UNIT V

Product Development and Prototyping:

MVPs, prototyping tools, design thinking principles, customer validation.

Pitching and Communication: Pitch deck structure, elevator pitch, storytelling for startups, communication skills.

Text Books:

1. **Timmons & Spinelli**, “*New Venture Creation: Entrepreneurship for the 21st Century*”, McGraw-Hill.
2. **Eric Ries**, “*The Lean Startup*”, Crown Business.
3. **Osterwalder & Pigneur**, “*Business Model Generation*”, Wiley.
4. **Peter Drucker**, “*Innovation and Entrepreneurship*”, Harper Business.
5. NASSCOM and DPIIT Startup Guides (Govt. of India)

Reference Books:

1. Hisrich, Peters, Shepherd – “*Entrepreneurship*”, McGraw-Hill Education.

EXIT SCHEME:

The student will have to complete all courses from any one of the groups to be eligible to get a B. Voc. in Information Technology after second year.

Bucket of Courses: Group I

- Programming and Software Development
- Advanced Python Programming
- Full Stack Web Development (HTML, CSS, JavaScript, Node.js)
- DevSecOps and Scripting

Bucket of Courses: Group II

- Cisco Certified Network Associate (CCNA)
- Network Configuration and Troubleshooting
- Cybersecurity Essentials
- Ethical Hacking and Penetration Testing
- Firewall Management and Security Policies

Bucket of Courses: Group III

- SQL and Database Management Systems
- NoSQL Databases (MongoDB, Cassandra)
- Data Warehousing and ETL (Extract, Transform, Load)
- Database Administration

Bucket of Courses: Group IV

- Cloud Computing and Virtualization
- Cloud Security and Compliance
- Amazon Web Services (AWS) Certification
- Microsoft Azure Fundamentals

Bucket of Courses: Group I

- Programming and Software Development
- Advanced Python Programming
- Full Stack Web Development (HTML, CSS, JavaScript, Node.js)
- Mobile App Development (Android, iOS)

Course Title:	Programming and Software Development	Semester	VI
Course Code:		Course Type	Exit
Prerequisite		L – T – P	3 – 0 – 0
Stream	Group - I	Credits	3

Course Objectives:

1. To strengthen students' understanding of advanced programming principles.
2. To introduce systematic software development processes and practices.
3. To develop skills for problem solving, debugging, and version control.
4. To enable students to develop, test, and document real-world software applications.

Course Outcomes:

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

1. Apply advanced programming concepts to develop efficient solutions.
2. Understand the software development lifecycle and related models.
3. Use version control tools and collaborative development platforms.
4. Design, develop, test, and document small- to medium-scale software.
5. Apply Agile principles in the development process.

Course Contents:

UNIT I

Advanced Programming Concepts: Recursion, file handling, and exception handling, Working with libraries and modules/packages, Introduction to object-oriented programming (classes, objects, inheritance, polymorphism), Code reusability and modularity, Case study: Implementing a small CLI-based application (e.g., inventory system).

UNIT II

Software Development Fundamentals: Introduction to software engineering and SDLC, Software process models: Waterfall, Agile, Spiral, V-Model, Requirements gathering, feasibility study, Introduction to UML diagrams: Use case, class, sequence, Tools for documentation: Doxygen, Markdown.

UNIT III

Version Control and Collaboration Tools: Introduction to Git and GitHub/GitLab, Git basics: clone, commit, push, pull, branches; Managing collaborative projects, Continuous

integration basics using GitHub Actions or similar tools, Best practices for source code management.

UNIT IV

Software Testing and Debugging: Importance of testing in development, Types of testing: Unit testing, Integration testing, System testing, Automated testing tools: PyTest, JUnit, Selenium (overview), Debugging techniques and strategies, Writing test cases and generating test reports.

UNIT V

Agile Software Development and Deployment: Agile principles and Scrum methodology, Roles: Product Owner, Scrum Master, Development Team, User stories and sprint planning, Software deployment basics: local, cloud, container-based (Docker overview), Final mini-project: Develop, test, and deploy a working application.

Text Books:

1. **“Software Engineering”** by Ian Sommerville
2. **“Clean Code: A Handbook of Agile Software Craftsmanship”** by Robert C. Martin

Reference Books:

1. **“Python Crash Course”** or **“Core Java”**, depending on programming language focus

Course Title:	Advanced Python Programming	Semester	VI
Course Code:		Course Type	
Prerequisite		L – T – P	3 – 0 – 0
Stream	Exit	Credits	3

Course Objectives:

1. To deepen understanding of Python programming constructs and object-oriented design.
2. To apply Python in real-world tasks including file handling, database operations, and API interaction.
3. To introduce students to modules, packages, regular expressions, and exception handling.
4. To prepare students for advanced application areas like web development, data handling, and scripting.

Course Outcomes:

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

1. Utilize advanced features of Python such as decorators, generators, and comprehensions.
2. Develop object-oriented programs using classes, inheritance, and polymorphism.
3. Handle data with files, databases, and regular expressions.
4. Build basic web applications and automate tasks using Python libraries.
5. Work with APIs and perform error handling effectively.

Course Contents:

UNIT I

Advanced Python Features: List, dictionary, and set comprehensions, Lambda functions, map, filter, and reduce; Iterators and generators, Decorators and closures, Modules and packages: creation and usage.

UNIT II

Object-Oriented Programming in Python: Classes and objects, Constructors, destructors; Inheritance and method overriding, Polymorphism and encapsulation, Special (magic/dunder) methods (`__str__`, `__repr__`, etc.)

UNIT III

File Handling and Exception Management: Reading and writing text and binary files, CSV and JSON file operations; Exception handling: try, except, finally, raise; Custom exceptions; Context managers and the with statement.

UNIT IV

Database and Regular Expressions: Introduction to SQLite/MySQL with Python (using sqlite3 / mysql.connector), CRUD operations from Python, Regular expressions: searching, matching, replacing patterns, Validation using regex, Real-world examples: email, phone number, password validation.

UNIT V

Web and Automation with Python: Working with requests and APIs (GET, POST), Basics of web scraping using BeautifulSoup, Automating Excel and PDFs (using openpyxl, PyPDF2), Basics of Flask (simple web app), Capstone Task: Create a mini project integrating database, web, and APIs

Text Books:

1. *“Python Programming: Using Problem Solving Approach”* – Reema Thareja
2. *“Fluent Python”* – Luciano Ramalho

Reference Books:

1. *“Automate the Boring Stuff with Python”* – Al Sweigart

Course Title:	Full Stack Web Development (HTML, CSS, JavaScript, Node.js)	Semester	VI
Course Code:		Course Type	
Prerequisite		L – T – P	3 – 0 – 0
Stream	Exit	Credits	3

Course Objectives:

1. To provide hands-on experience in designing and developing dynamic web applications.
2. To teach front-end and back-end technologies using industry-relevant tools.
3. To integrate client-side and server-side technologies to create full stack solutions.
4. To introduce concepts of RESTful APIs, database interaction, and deployment.

Course Outcomes:

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

1. Design responsive front-end web pages using HTML, CSS, and JavaScript.
2. Develop interactive applications using JavaScript and DOM manipulation.
3. Implement backend services using Node.js and [Express.js](#).
4. Integrate frontend with backend using REST APIs.
5. Deploy full stack applications using basic cloud and hosting platforms.

Course Contents:

UNIT I

Front-End Development Basics: Introduction to Web Development and Full Stack Concepts, HTML5: Semantic elements, forms, media tags; CSS3: Selectors, box model, positioning, flexbox, grid; Responsive design with media queries, Basic website layout project.

UNIT II

JavaScript Programming: JavaScript syntax, variables, data types, operators; Control structures, functions, arrays, objects; DOM manipulation and events, Form validation and client-side scripting, Using browser developer tools for debugging.

UNIT III

Backend with Node.js and Express: Introduction to Node.js: Features, installation, npm; Modules and packages, file system, events; Express.js framework: Routing, middleware, request handling, Handling JSON data and query parameters, REST API development basics.

UNIT IV

Database and Integration: Introduction to MongoDB or MySQL (based on availability), CRUD operations with backend integration, Connecting backend to database using Node.js, Building RESTful APIs to perform database operations, Postman for API testing.

UNIT V

Full Stack Application and Deployment: Integrating frontend and backend (AJAX/fetch API), Session and authentication basics (intro to JWT or cookies), Hosting static websites (GitHub Pages, Netlify), Hosting backend apps (Render, Railway, or simple cloud deployment), Mini Project: Complete full stack application (e.g., task manager, blog, product catalog).

Text Books:

1. *“Eloquent JavaScript”* – Marijn Haverbeke
2. *“Node.js Web Development”* – David Herron

Reference Books:

1. MDN Web Docs – <https://developer.mozilla.org>

Course Title:	DevOps and Scripting	Semester	VI
Course Code:		Course Type	
Prerequisite		L – T – P	3 – 0 – 0
Stream	Exit	Credits	3

Course Objectives:

1. To introduce the fundamental concepts of DevSecOps and its lifecycle.
2. To develop shell and Python scripting skills for automation.
3. To understand and implement version control and CI/CD pipelines.
4. To explore containerization using Docker and orchestration using Kubernetes.
5. To automate infrastructure and configuration management using tools like Ansible.

Course Outcomes:

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

1. Understand and explain DevSecOps concepts, lifecycle, and tools.
2. Develop shell and Python scripts for automation in DevSecOps.
3. Apply version control practices using Git and GitHub.
4. Build and manage CI/CD pipelines using Jenkins or similar tools.
5. Implement containerization using Docker and orchestration with Kubernetes.
6. Automate infrastructure provisioning using tools like Ansible or Terraform.

Course Contents:

UNIT I

Introduction to DevSecOps: What is DevSecOps?, DevSecOps Lifecycle, Benefits and Need for DevSecOps, Agile vs DevSecOps, DevSecOps Tools Overview (Git, Jenkins, Docker, Kubernetes, Ansible), Cultural and Technical Aspects of DevSecOps.

UNIT II

Linux Essentials and Shell Scripting: Linux Commands for File Handling, Process Management, Permissions and Environment Variables, Bash Shell Scripting Basics, Conditional Statements, Loops, Functions, Automating Tasks with Shell Scripts, Cron Jobs and Task Scheduling.

UNIT III

Version Control and Scripting for Automation: Git: Configuration, Repositories, Branching, Merging, GitHub/GitLab Workflow, Writing Python Scripts for DevSecOps Tasks, Automating File and Log Management using Python, Introduction to APIs and Using Python for API Calls.

UNIT IV

CI/CD Pipeline Implementation: Introduction to Continuous Integration and Deployment, Jenkins: Installation, Configuration, Job Creation, Jenkins Pipelines and Groovy DSL, Integrating Git with Jenkins, Build Tools: Maven, Gradle, Automated Testing and Deployment Practices.

UNIT V

Containerization and Configuration Management: Docker: Images, Containers, Dockerfile, Volumes, Networking, Docker Compose for Multi-container Applications, Kubernetes: Architecture, Pods, Services, Deployments, YAML Configuration Files, Infrastructure as Code (IaC): Ansible Playbooks, Introduction to Terraform (Optional Topic).

Text Books:

1. Len Bass, Ingo Weber, Liming Zhu, *“DevSecOps: A Software Architect’s Perspective”*, Addison-Wesley, 2015.
2. Kelsey Hightower, Brendan Burns, Joe Beda, *“Kubernetes: Up & Running”*, O'Reilly Media, 2nd Edition, 2019.

Reference Books:

1. Gene Kim, Jez Humble, Patrick Debois, *“The DevSecOps Handbook: How to Create World-Class Agility”*, IT Revolution Press, 2016.
2. Michael Hausenblas, Phil Estes, *“Docker for Developers”*, Packt Publishing, 2020.
3. Pethuru Raj, Jeeva S. Chelladhurai, *“Learning DevSecOps: Continuously Deliver Better Software”*, Packt Publishing, 2018.
4. Mark Bates, *“Programming in Bash”*, Wiley Publishing, 2014.

Bucket of Courses: Group II

- Cisco Certified Network Associate (CCNA)
- Network Configuration and Troubleshooting
- Cybersecurity Essentials
- Ethical Hacking and Penetration Testing
- Firewall Management and Security Policies

Course Title : Cisco Certified Network Associate
Prerequisite : Computer Networks
Course Code :
Course Type : Exit Course

Semester : V
L – T – P : 3-0-0
Credits : 3

Course Objective:

1. Understand fundamental networking concepts and the OSI and TCP/IP models to build a strong networking foundation.
2. Apply IP addressing and subnetting concepts to design and manage network addressing schemes.
3. Analyze routing principles and implement routing protocols to enable network communication across multiple networks.
4. Understand and configure switching techniques, VLANs, and protocols to manage network segmentation and redundancy.
5. Apply network security principles and troubleshooting methods to maintain and secure network infrastructure.

Course Outcomes:

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

1. Describe networking models, protocols, and device functions essential for network communication.
2. Calculate subnet masks and design IP addressing schemes for efficient network organization.
3. Configure and interpret routing protocols to ensure data delivery between networks.
4. Configure switches and VLANs to segment networks and prevent loops using STP.
5. Utilize security techniques and troubleshoot common network issues to ensure reliable operation.

Course Contents:

UNIT I

Fundamentals of Networking and OSI/TCP-IP Models: [8 Hours]

Introduction to Computer Networks, Network Types: LAN, WAN, MAN, PAN

OSI Model: Layers and Functions, TCP/IP Protocol Suite, Data Encapsulation and Decapsulation, Introduction to Network Devices: Routers, Switches, Hubs, Bridges

UNIT II

IP Addressing and Subnetting: [8 Hours]

IPv4 Addressing: Classes, Public vs Private Ips, Subnetting: Concepts and Calculations

Variable Length Subnet Masking (VLSM), IPv6 Addressing Overview, Address Resolution Protocol (ARP)

UNIT III

Routing Fundamentals:

[8 Hours]

Routing Concepts and Functions, Static Routing and Dynamic Routing Overview, Routing Protocols: RIP, OSPF, EIGRP, Routing Tables and Packet Forwarding, Introduction to Access Control Lists (ACLs)

UNIT IV

Switching Concepts and VLANs:

[8 Hours]

Introduction to Switching and Switch Operations, MAC Address Table and Frame Forwarding, VLANs: Concepts and Configuration, Inter-VLAN Routing, Spanning Tree Protocol (STP) and Redundancy

UNIT V

Network Security and Troubleshooting:

[8 Hours]

Network Security Basics: Firewalls, VPNs, IDS/IPS, Basic Network Troubleshooting Commands and Techniques, Troubleshooting Layer 1 and Layer 2 Issues, Device Configuration and Management Using Cisco IOS, Introduction to Wireless Networking

Textbooks :

1. T. Lammle, CCNA Routing and Switching Study Guide, 2nd ed., Sybex, 2013.
2. J. M. Kurose and K. W. Ross, Computer Networking: A Top-Down Approach, 7th ed., Pearson, 2017.

Reference Books:

1. W. Stallings, Data and Computer Communications, 10th ed., Pearson, 2013.
2. Cisco Systems, CCNA Official Cert Guide, 2nd ed., Cisco Press, 2010.
3. B. A. Forouzan, Data Communications and Networking, 5th ed., McGraw-Hill, 2012.

Course Title : Network Configuration and Troubling
Prerequisite : Computer Networks
Course Code :
Course Type : Exit Course

Semester : V
L – T – P : 3-0-0
Credits : 3

Course Objective:

1. To understand the fundamental process of network setup and basic configuration of networking devices.
2. To apply IP addressing techniques and configure DHCP/DNS for dynamic network management.
3. To analyze VLAN configurations and troubleshoot inter-VLAN communication.
4. To evaluate and troubleshoot static and dynamic routing protocols in WAN environments.
5. To implement basic security configurations and use network diagnostic tools for performance analysis.

Course Outcome:

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

1. Students will be able to identify and configure basic router and switch settings using command-line interfaces.
2. Students will configure and validate IP addressing, DHCP leases, and DNS resolutions in a network.
3. Students will troubleshoot VLAN assignments, trunking, and inter-VLAN routing problems.
4. Students will evaluate routing protocol configurations (OSPF/RIP) and resolve routing-related issues.
5. Students will configure security features (ACLs, Port Security) and use tools like Wireshark and Nmap for diagnostics.

UNIT I

Network Basics and Device Configuration: [8 Hours]

Introduction to Network Architecture, Physical and Data Link Layer Troubleshooting, Device Boot Process (Router, Switch), Basic Configuration Commands (hostname, password, interfaces), Network Topologies and Device Roles, Interface Status and Cable Testing

UNIT II

IP Addressing, DHCP, and DNS Configuration: [8 Hours]

IPv4/IPv6 Addressing and Subnetting, DHCP Configuration and Troubleshooting, Static vs Dynamic IP Allocation, DNS Server Configuration and Resolution Testing, Ping, Traceroute, IPconfig, nslookup Tools

UNIT III

VLANs and Inter-VLAN Troubleshooting: [8 Hours]

VLAN Creation and Management, Trunking and Native VLAN, Inter-VLAN Routing Configuration, Troubleshooting VLAN and Routing Issues, Layer 2 vs Layer 3 Devices

UNIT IV

Routing and WAN Connectivity Issues: [8 Hours]

Static and Dynamic Routing Configuration, OSPF and RIP Troubleshooting, Default Route and Route Summarization, Troubleshooting Routing Loops and Blackholes, WAN Setup Basics (PPP, HDLC, Frame Relay intro)

UNIT V

Security, Logs, and Network Diagnostic Tools: [8 Hours]

Port Security and ACL Troubleshooting, Firewall Basics and Configuration, Log Analysis and Syslog Server Setup, Packet Analysis using Wireshark, Network Monitoring Tools: Nmap, Netstat, SNM.

Textbooks

1. T. Lammle, “*CCNA Routing and Switching Study Guide*”, 2nd ed., Sybex, 2013.
2. R. Droms and T. Lemon, “*The DHCP Handbook*”, 2nd ed., Pearson, 2002.

Reference Books

1. J. M. Kurose and K. W. Ross, “*Computer Networking: A Top-Down Approach*”, 7th ed., Pearson, 2017.
2. W. Stallings, “*Data and Computer Communications*”, 10th ed., Pearson, 2013.
3. Chris Sanders, “*Practical Packet Analysis*”, 3rd ed., No Starch Press, 2017.

Course Title : Cybersecurity Essentials
Prerequisite : Computer Networks
Course Code :
Course Type : Exit Course

Semester : V
L – T – P : 3-0-0
Credits : 3

Course Objective:

1. Understand the importance and scope of cybersecurity in protecting digital assets.
2. Apply core network security mechanisms to identify and prevent common network-based attacks.
3. Analyze and secure systems and applications from known vulnerabilities.
4. Evaluate cryptographic techniques and authentication protocols for securing information.
5. Design and assess cybersecurity policies and incident response strategies within organizations.

Course Outcome

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

1. Explain key cybersecurity threats, attack vectors, and real-world implications.
2. Configure basic network defenses and identify intrusion patterns.
3. Analyze vulnerabilities in systems and applications and propose remediation strategies.
4. Evaluate and compare cryptographic methods and authentication protocols in various use cases.
5. Formulate a basic incident response plan aligned with cybersecurity standards and policies.

UNIT I

Introduction to Cybersecurity and Threat Landscape:

[8 Hours]

Fundamentals of cybersecurity, Importance of cybersecurity in the digital age, Types of cyber threats: malware, ransomware, phishing, social engineering, Cybersecurity objectives: Confidentiality, Integrity, Availability (CIA triad), Real-world attack case studies

UNIT II

Network Security Fundamentals:

[8 Hours]

Network architecture: LAN, WAN, Internet, IP addressing and TCP/IP model, Firewalls, IDS, IPS, VPNs, Network attack types: DoS, DDoS, MITM, ARP Spoofing, Secure communication protocols (HTTPS, SSH)

UNIT III

System and Application Security: [8 Hours]

Operating system vulnerabilities (Windows, Linux), Secure system configuration and patching, Application security basics (secure coding practices, buffer overflow), Web application attacks: SQL injection, XSS, CSRF, Tools: OWASP ZAP, Burp Suite basics

UNIT IV

Cryptography and Authentication: [8 Hours]

Symmetric and asymmetric encryption (AES, RSA), Hash functions (SHA, MD5), digital signatures, Public Key Infrastructure (PKI), Authentication protocols (2FA, Kerberos, OAuth), Security certificates (SSL/TLS)

UNIT V

Cybersecurity Policies, Risk Management and Incident Response: [8 Hours]

Cybersecurity policies and frameworks (ISO 27001, NIST), Risk assessment and mitigation, Cyber laws and data privacy (IT Act, GDPR), Incident detection and response plan, Cyber forensics basics

Text Books:

1. William Stallings – “*Network Security Essentials*”: Applications and Standards, Pearson
2. Michael T. Goodrich, Roberto Tamassia – “*Introduction to Computer Security*”, Pearson

References Books:

1. Charles P. Pfleeger, Shari Lawrence Pfleeger – “*Security in Computing*”, Pearson
2. NPTEL Course – “*Introduction to Cybersecurity*” by Prof. Madhu Belur (IIT Bombay)
3. Kamesh Namuduri – “*Cybersecurity Fundamentals*” (NPTEL and Swayam aligned content)
4. RFC 2196 – Site Security Handbook, IETF

Course Title : Ethical Hacking and Penetration testing
Prerequisite : Computer Networks
Course Code :
Course Type : Exit Course

Semester : V/VI
L – T – P : 3-0-0
Credits : 3

Course Objectives (COs):

1. Understand the fundamentals and legality of ethical hacking and how attackers exploit vulnerabilities.
2. Apply footprinting, scanning, and enumeration techniques in real-time lab environments.
3. Analyze vulnerabilities and simulate various types of attacks on systems and networks.
4. Evaluate web application vulnerabilities and bypass authentication and security controls.
5. Design a full penetration testing lifecycle with appropriate documentation and countermeasures.

Course Outcomes:

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

1. Describe ethical hacking principles, attack surfaces, and legal issues.
2. Demonstrate scanning and enumeration tools to gather system/network info.
3. Analyze and exploit system-level vulnerabilities using Kali Linux and open-source tools.
4. Evaluate the security of web applications through vulnerability assessment.
5. Create penetration test reports with proposed remediation strategies.

UNIT I

Introduction to Ethical Hacking:

[8 Hours]

What is hacking and ethical hacking?, Types of hackers and attacks, Ethical hacking process and rules of engagement, Legal framework and cyber laws (India + International), Role of penetration testing in cybersecurity

UNIT II

Footprinting, Scanning and Enumeration:

[8 Hours]

Reconnaissance (passive and active), WHOIS, nslookup, traceroute, email tracking, Network scanning: Nmap, Netcat, Banner grabbing and OS fingerprinting, Enumerating services and vulnerabilities

UNIT III

System Hacking and Malware Threats:

[8 Hours]

Password cracking (John the Ripper, Hydra), Privilege escalation, Keylogging, spyware, trojans, Buffer overflow exploitation, Introduction to Metasploit Framework

UNIT IV

Web Application Penetration Testing:

[8 Hours]

OWASP Top 10 vulnerabilities (SQLi, XSS, CSRF, etc.), Broken authentication and session management, Tools: Burp Suite, ZAP Proxy, Injection attacks and cross-site scripting, Securing applications and input validation

UNIT V

Unit 5: Wireless, Cloud and Report Writing:

[8 Hours]

Wireless encryption cracking (WEP, WPA2), Cloud security basics and vulnerabilities, Tools: Aircrack-ng, Wireshark, Writing professional pentest reports, Risk scoring and mitigation planning

Course Title : Firewall Management and Security Policies

Semester : V/VI

Prerequisite : Computer Networks

L – T – P : 3-0-0

Course Code :

Credits : 3

Course Type : Exit Course

Course Objectives (COs):

1. Understand the role of firewalls in network security and the underlying technologies.
2. Configure different types of firewalls and apply packet filtering and stateful inspection.
3. Design and implement robust security policies in alignment with organizational needs.
4. Evaluate firewall rules, detect misconfigurations, and manage policy updates.
5. Analyze modern approaches to firewall integration with SIEM, cloud, and zero-trust environments.

Course outcomes:

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

1. Explain the fundamental concepts of firewalls, types, and their role in network defense.
2. Configure and manage packet-filtering and stateful inspection firewalls using tools or appliances.
3. Design and write security policies based on organizational security requirements and risk assessment.
4. Analyze firewall logs, optimize rules, and identify policy misconfigurations or anomalies.
5. Evaluate advanced integrations such as Next-Gen Firewalls (NGFW), cloud firewalls, and policy automation.

Course Contents:

UNIT I

Introduction to Firewall Technologies:

[8 hours]

Basics of firewalls and their need, Types of firewalls: packet filtering, stateful inspection, proxy-based, NGFW, Placement of firewalls in network topology, Demilitarized Zones (DMZ), bastion hosts, dual-homed hosts, Introduction to UTM (Unified Threat Management).

UNIT II

Firewall Configuration and Rule Management:

[8 hours]

Packet filtering vs. stateful inspection, NAT (Network Address Translation) and firewall interaction, Default-deny and allow-all strategies, Firewall rule creation and optimization, Lab: Basic configuration on iptables, pfSense, or Cisco ASA.

UNIT III

Security Policies and Rule Design:

[8 hours]

Role of security policies in firewall implementation, Components of effective security policies, Access control policies: MAC, DAC, RBAC, Designing rules for specific services (HTTP, FTP, SSH, etc.), Policy documentation and version control.

UNIT IV

Monitoring and Policy Auditing:

[8 hours]

Log analysis and threat detection, Firewall monitoring tools: syslog, SNMP, Detecting shadow and redundant rules, Automated rule analysis tools (FireMon, AlgoSec), Change management and compliance checks.

UNIT V

Advanced Firewall Concepts and Trends:

[8 hours]

Next-Generation Firewalls (NGFW): DPI, application-aware filtering, Cloud-based firewalls (Azure Firewall, AWS WAF, GCP Firewall), Integrating firewalls with SIEM and threat intelligence, Policy as Code and Zero Trust principles, Future of firewall management in SASE architectures.

Textbooks

1. Stallings, William – “*Network Security Essentials: Applications and Standards*”, Pearson Education.
2. Cheswick, Bill, Bellovin, Steven, Rubin, Aviel D. – “*Firewalls and Internet Security: Repelling the Wily Hacker*”, Addison-Wesley.
3. Zscaler, Palo Alto Networks Documentation – (Latest vendor manuals for NGFW and cloud firewall practices)

Reference Books & Online Resources

1. NPTEL Course: *Network Security* by Prof. D. Mukhopadhyay, IIT Kharagpur <https://nptel.ac.in/courses/106105031>
2. “*Network Security and Firewalls*” – Open Learning Courseware by IGNOU
3. IETF RFCs for Firewall and NAT: RFC 2979, RFC 3022
4. “*Fortinet Firewall Best Practices Guide*” (latest version)
5. “*Cisco ASA and Firepower Configuration Guide*”

Bucket of Courses: Group III

- SQL and Database Management Systems
- NoSQL Databases (MongoDB, Cassandra)
- Data Warehousing and ETL (Extract, Transform, Load)
- Database Administration

Course Title : SQL and Database Management Systems

Semester : V/VI

Prerequisite : Database Management Systems

L – T – P : 3-0-0

Course Code :

Credits : 3

Course Type : Exit Course

Course Objectives:

1. Understand the basic concepts of databases and relational models.
2. Develop SQL skills to manage and query data.
3. Understand normalization, relational algebra, and data integrity.
4. Explore transaction management and concurrency control.
5. Familiarize with DBMS architectures and indexing techniques.

Course Outcomes:

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

1. Explain the architecture and components of a DBMS and relational database concepts.
2. Construct SQL queries to manage and retrieve data from databases.
3. Analyze and apply normalization techniques to remove data redundancy.
4. Evaluate transactions for concurrency control and recovery techniques.
5. Design a relational database application using appropriate indexing and normalization.

Course Contents:

UNIT I

Introduction to DBMS, Data Models, Relational Model, Schema, Instances, Architecture.

UNIT II

SQL Basics, DDL, DML, DCL, TCL; Joins, Subqueries, Views, Aggregate Functions.

UNIT III

Relational Algebra, Normalization (1NF to 5NF), Functional Dependencies.

UNIT IV

Transactions, ACID properties, Concurrency Control, Lock-based and Timestamp-based Protocols.

UNIT V

Indexing (B-Tree, Hashing), Query Optimization, Storage and File Structure.

Textbooks and References:

1. Korth, Silberschatz & Sudarshan – “*Database System Concepts*”, McGraw-Hill.
2. Elmasri & Navathe – “*Fundamentals of Database Systems*”, Pearson.
3. NPTEL Course: “*Database Management Systems*” by Prof. P. P. Chakrabarti (IIT Kharagpur)

Course Title : NoSQL Databases (MongoDB, Cassandra)

Semester : V/VI

Prerequisite : Database Management Systems

L – T – P : 3-0-0

Course Code :

Credits : 3

Course Type : Exit Course

Course Objectives:

1. Understand limitations of RDBMS for big data.
2. Learn data models in NoSQL systems (document, key-value, column, graph).
3. Work with MongoDB and Cassandra databases.
4. Perform CRUD operations, indexing, and aggregation in NoSQL.
5. Understand distributed NoSQL architecture and CAP theorem.

Course Outcomes:

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

1. Compare SQL and NoSQL databases.
2. Implement data operations using MongoDB.
3. Analyze Cassandra architecture and data partitioning.
4. Evaluate consistency and availability trade-offs.
5. Build a scalable NoSQL database system.

Course Contents:

UNIT I

Introduction to NoSQL, CAP Theorem, BASE vs ACID, Types of NoSQL.

UNIT II

MongoDB – Documents, Collections, CRUD Operations, Aggregation, Indexing.

UNIT III

Cassandra – Data Model, Architecture, Keyspaces, Tables, CQL.

UNIT IV

Consistency Models, Replication, Sharding, Availability.

UNIT V

Use Cases, Performance Tuning, Integration with Big Data Systems.

Textbooks and References:

1. Shashank Tiwari – “*Professional NoSQL*”, Wiley.
2. Kristina Chodorow – “*MongoDB: The Definitive Guide*”, O'Reilly.
3. NPTEL: “*Big Data Computing*” by Dr. Rajiv Misra (IIT Patna)

Course Title : Data Warehousing and ETL (Extract, Transform, Load)

Semester : V/VI

Prerequisite : Database Management Systems

L – T – P : 3-0-0

Course Code :

Credits : 3

Course Type : Exit Course

Course Objectives:

1. Understand the architecture of Data Warehousing.
2. Learn dimensional modeling and schema design.
3. Implement ETL processes using tools.
4. Analyze OLAP operations and data cubes.
5. Study data cleaning and transformation techniques.

Course Outcomes:

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

1. Describe the purpose and architecture of data warehouses.
2. Apply star and snowflake schema design for modeling.
3. Design and execute ETL pipelines.
4. Analyze OLAP operations and generate reports.
5. Evaluate data quality and transformation strategies.

Course Contents:

UNIT I

Data Warehouse Architecture, Components, Comparison with OLTP.

UNIT II

Dimensional Modeling, Star, Snowflake Schemas, Fact and Dimension Tables.

UNIT III

ETL Concepts, Data Extraction, Cleansing, Transformation, Loading.

UNIT IV

OLAP Operations (Roll-up, Drill-down, Slice, Dice), Data Cubes.

UNIT V

Data Integration, Metadata Management, Data Quality Issues.

Textbooks and References:

- Ralph Kimball – “*The Data Warehouse Toolkit*”, Wiley.
- Paulraj Ponniah – “*Data Warehousing Fundamentals*”, Wiley.
- NPTEL: “*Data Science for Engineers*” by Prof. S. S. Iyengar (IIT Madras)

Course Title : Database Administration

Semester : V/VI

Prerequisite : Database Management Systems

L – T – P : 3-0-0

Course Code :

Credits : 3

Course Type : Exit Course

Course Objectives:

1. Understand roles and responsibilities of a DBA.
2. Learn backup, restore, and recovery mechanisms.
3. Perform performance tuning and indexing.
4. Manage users, privileges, and security.
5. Administer DBMS installation and monitoring.

Course Outcomes:

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

1. Define roles and tasks of a database administrator.
2. Apply techniques for user management and access control.
3. Analyze database performance and tuning strategies.
4. Evaluate different backup and recovery procedures.
5. Set up and configure a DBMS instance for enterprise use.

Course Contents:

UNIT I

DBA Roles, Responsibilities, DBMS Installation, Configurations.

UNIT II

User Management, Roles, Privileges, Auditing, Access Control.

UNIT III

Performance Tuning, Query Optimization, Indexing, Storage.

UNIT IV

Backup and Recovery Strategies, RMAN, Point-in-time Recovery.

UNIT V

Monitoring Tools, Alerts, Log Files, Patching and Upgrades.

Textbooks and References:

1. Craig Mullins – “*Database Administration*”, Addison-Wesley.
2. Oracle Documentation for DBA.
3. NPTEL: “*Database Systems*” by Prof. S. Sudarshan (IIT Bombay)

Bucket of Courses: Group IV

- Cloud Computing and Virtualization
- Cloud Security and Compliance
- Amazon Web Services (AWS) Certification
- Microsoft Azure Fundamentals

Course Title:	Cloud Computing and Virtualization	Semester	VI
Course Code:		Course Type	
Prerequisite		L – T – P	3 – 0 – 0
Stream	Exit	Credits	3

Course Objectives:

1. To understand the fundamentals of cloud computing and its evolution.
2. To study virtualization technologies and their role in cloud deployment.
3. To explore cloud delivery models (IaaS, PaaS, SaaS).
4. To configure and manage virtual machines and virtual networks.
5. To apply virtualization techniques in real-world cloud environments.

Course Outcomes:

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

1. Explain the fundamentals of cloud computing and virtualization.
2. Compare different types of virtualization techniques and tools.
3. Configure and deploy virtual machines and containers.
4. Implement basic cloud services using AWS/Azure/GCP.
5. Design and manage virtualized cloud infrastructures.

Course Contents:

UNIT I

Introduction to Cloud Computing: Definition and characteristics of cloud computing, History and evolution of cloud technologies, Cloud deployment models: Public, Private, Hybrid, Community, Cloud service models: IaaS, PaaS, SaaS; Advantages, limitations, and use cases.

UNIT II

Virtualization Concepts: Definition and need for virtualization; Types of virtualization: Server, Storage, Network, Desktop, Application; Hypervisors: Type I and Type II; Virtual Machine Monitor (VMM); Benefits and challenges of virtualization.

UNIT III

Virtualization Technologies and Tools: VMware, Hyper-V, VirtualBox, KVM; Containers vs. Virtual Machines; Containerization using Docker; Introduction to Kubernetes (basics only).

UNIT IV

Cloud Infrastructure and Virtualization: Virtualized data centers; Software-defined networking (SDN) and software-defined storage (SDS); Elasticity, provisioning, and orchestration; Cloud migration basics.

UNIT V

Cloud Services and Virtualization in Practice: Provisioning cloud VMs (AWS EC2, Azure VM); Deploying containers in cloud environments; Load balancing and auto-scaling; Cloud storage with virtualization (Amazon S3, Azure Blob).

Text Books:

1. Kris Jamsa, “Cloud Computing: SaaS, PaaS, IaaS, Virtualization, Business Models, Mobile, Security and More”, Jones & Bartlett Publishers, Edition (1st), 2013.
2. Thomas Erl, Ricardo Puttini & Zaigham Mahmood, “Cloud Computing: Concepts, Technology & Architecture”, Prentice Hall / PHI, 1st Edition, 2013.

Reference Books:

1. Barrie Sosinsky, “Cloud Computing Bible”, Wiley, 1st Edition, 2010.

Course Title:	Cloud Security and Compliance	Semester	VI
Course Code:		Course Type	
Prerequisite		L – T – P	3 – 0 – 0
Stream	Exit	Credits	3

Course Objectives:

1. To understand the fundamentals of security in cloud environments.
2. To analyze risks and vulnerabilities associated with cloud computing.
3. To explore identity, access, and data protection mechanisms.
4. To study compliance frameworks relevant to cloud services.
5. To examine cloud provider security offerings and shared responsibility.

Course Outcomes:

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

1. Identify cloud-specific security challenges and threats.
2. Implement IAM policies and data encryption techniques.
3. Analyze security measures provided by major cloud providers.
4. Understand and apply key compliance frameworks (e.g., GDPR, ISO/IEC 27001).
5. Evaluate cloud service provider compliance responsibilities.

Course Contents:

UNIT I

Introduction to Cloud Security: Security challenges in cloud computing, Shared responsibility model, CIA triad in the cloud context (Confidentiality, Integrity, Availability); Cloud threat landscape: data breaches, insider threats, DoS; Security design principles in cloud.

UNIT II

Identity and Access Management (IAM): IAM concepts: Users, Groups, Roles, Policies, Principle of Least Privilege (PoLP), Federated identity and single sign-on (SSO), Multi-factor authentication (MFA), Case Study: IAM in AWS, Azure, and GCP.

UNIT III

Data Protection and Encryption: Data classification and lifecycle in the cloud; Encryption: at rest, in transit, in use; Key management services (KMS) in AWS, Azure Key Vault; Secure data storage and backup; Tokenization and anonymization techniques.

UNIT IV

Compliance and Legal Aspects: Overview of compliance requirements in the cloud, Major

standards and regulations: GDPR – General Data Protection Regulation, ISO/IEC 27001 – Information Security Management, HIPAA – Health Insurance Portability and Accountability Act, PCI DSS – Payment Card Industry Data Security Standard; Cloud audit and certification programs (e.g., SOC 2, CSA STAR); Vendor lock-in and data sovereignty.

UNIT V

Cloud Security Tools and Best Practices: Cloud security services: AWS Shield, Azure Defender, Google Security Command Center; Security monitoring, logging, and incident response; Vulnerability management and patching in cloud; Best practices: network segmentation, firewalls, bastion hosts; Capstone: Cloud security policy design project.

Text Books:

1. Eyal Estrin, *“Cloud Security Handbook – Second Edition: Effectively Secure Cloud Environments Using AWS, Azure, and GCP”*, Packt Publishing, 2nd Edition, 2025.

Reference Books:

1. HisrScott E. Donaldson, *“Cloud Security Essentials: Identity Management, Encryption, and Cloud Compliance”*, Independently Published, 1st Edition, 2024.
2. Christopher Hale, *“Practical Cloud Security: A Guide for Secure Design and Deployment”*, O'Reilly Media, 2nd Edition, 2023.

Course Title:	Amazon Web Services (AWS) Certification	Semester	VI
Course Code:		Course Type	
Prerequisite		L – T – P	3 – 0 – 0
Stream:	Exit	Credits	3

Course Objectives:

1. To introduce AWS services and global cloud infrastructure.
2. To provide hands-on experience in configuring core AWS services.
3. To develop skills for secure, scalable, and fault-tolerant architecture.
4. To prepare students for AWS foundational/associate-level certifications.
5. To understand AWS pricing, support plans, and compliance features.

Course Outcomes:

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

1. Understand AWS architectural principles and global infrastructure.
2. Deploy and manage compute, storage, and network services in AWS.
3. Secure cloud environments using IAM and encryption tools.
4. Monitor and optimize AWS service usage and costs.
5. Apply AWS best practices to design and implement cloud solutions.

Course Contents:

UNIT I

Introduction to AWS and Cloud Computing: Overview of Cloud Computing models: IaaS, PaaS, SaaS; AWS Global Infrastructure: Regions, Availability Zones, Edge Locations; AWS Management Console and CLI basics, AWS Shared Responsibility Model; Overview of AWS certification paths (Foundational, Associate, Professional).

UNIT II

Compute and Storage Services: Amazon EC2: Instance types, AMIs, Security Groups, Key Pairs; Elastic Load Balancing (ELB) and Auto Scaling; Amazon S3: Buckets, Object Storage, Lifecycle Policies; Amazon EBS: Volume types, snapshots; Amazon EFS: File storage for Linux workloads.

UNIT III

Networking and Security in AWS: Amazon VPC: Subnets, Route Tables, Internet Gateway, NAT Gateway; Security Groups vs Network ACLs; Identity and Access Management (IAM): Users, Groups, Roles, Policies, Multi-Factor Authentication (MFA) and IAM Best Practices.

UNIT IV

Databases, Monitoring, and Automation: Amazon RDS: Managed relational databases; Amazon DynamoDB: NoSQL database basics; Monitoring and Logging: Amazon CloudWatch, CloudTrail; AWS Auto Scaling and Elastic Beanstalk.

UNIT V

Architecture, Pricing, and Certification Readiness: AWS Well-Architected Framework (5 pillars); Designing for High Availability, Fault Tolerance, and Cost Optimization; AWS Pricing Models: On-demand, Reserved, Spot Instances; AWS Cost Calculator, Budgets, and Trusted Advisor; Certification exam domains and sample questions.

Text Books:

1. Stuart Scott, *“AWS Certified Cloud Practitioner Study Guide: CLF-C01 Exam”*, Wiley, 3rd Edition, 2024.

Reference Books:

1. Eric Frick, *“Introduction to Amazon AWS”*, Independently published, 1st Edition, 2023
2. Mark Wilkins (Pearson IT Certification), *“AWS Certified Solutions Architect – Associate (SAA-C02) Cert Guide”*, Pearson IT Certification, 1st Edition, 2021

Course Title:	Microsoft Azure Fundamentals	Semester	VI
Course Code:		Course Type	
Prerequisite		L – T – P	3 – 0 – 0
Stream		Credits	3

Course Objectives:

1. To introduce students to Microsoft Azure cloud platform and its services.
2. To understand cloud concepts, Azure architecture, and core services.
3. To explore security, compliance, and governance in Azure.
4. To learn cost management and Azure pricing models.
5. To prepare students for the **AZ-900: Microsoft Azure Fundamentals** certification.

Course Outcomes:

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

1. Understand basic cloud concepts and Azure architecture.
2. Identify core Azure services and their use cases.
3. Apply principles of Azure governance, security, and identity.
4. Analyze Azure pricing models and perform cost estimations.
5. Demonstrate readiness for the AZ-900 certification exam.

Course Contents:

UNIT I

Introduction to Cloud and Azure Concepts: Overview of Cloud Computing, Benefits of cloud computing (High Availability, Scalability, Elasticity, etc.), Cloud service models: IaaS, PaaS, SaaS; Public, Private, Hybrid cloud models, Introduction to Microsoft Azure and Azure Portal; Overview of global infrastructure: Regions, Availability Zones, Resource Groups.

UNIT II

Core Azure Services and Solutions: Azure Compute Services: Azure Virtual Machines, App Services, Azure Functions; Azure Storage Services: Blob, File, Queue, Disk Storage; Azure Networking: Virtual Networks, VPN Gateway, Load Balancer, DNS; Azure Databases: Azure SQL Database, Cosmos DB; Azure Marketplace and Resource Deployment (ARM Templates).

UNIT III

Azure Security, Identity, and Governance: Azure Identity Services: Azure AD, Role-Based Access Control (RBAC), Multi-Factor Authentication (MFA); Security Tools: Microsoft Defender for Cloud, Azure Firewall, Network Security Groups ; Governance: Azure Policy, Blueprints, Locks, and Tags; Monitoring: Azure Monitor, Azure Activity Logs, and Alerts; Lab: Creating users and assigning RBAC roles.

UNIT IV

Azure Pricing, SLA, and Lifecycle: Azure Pricing Calculator and Total Cost of Ownership (TCO), Azure Cost Management and Budgets, Azure SLAs and Lifecycle of Services, Billing Models: Pay-as-you-go, Reserved Instances, Spot Pricing.

UNIT V

Exam Readiness and Case Studies: Overview of AZ-900 Exam Domains, Case Study: Hosting a Web App on Azure, Case Study: Implementing Secure Access for an Azure Environment, Practice Questions and Mock Test, Final Project: Deploy a complete solution using Azure services (Web + Database + Monitoring).

Text Books:

1. Jim Cheshire, “*Exam Ref AZ-900 Microsoft Azure Fundamentals*”, Microsoft Press, 3rd Edition, 2023.
2. Daniel A. Seara, Jim Cheshire, “*Microsoft Azure Fundamentals AZ-900 Exam Guide*”, Microsoft Press / Pearson IT Certification, 3rd Edition, 2023.

Reference Books:

1. Peter De Tender, Greg Leonardo, Jason Milgram, “*Azure Strategy and Implementation Guide for Beginners*”, Packt Publishing, 3rd Edition, 2020.
2. Benjamin Perkins, William Panek, “Exam Ref AZ-900 Microsoft Azure Fundamentals (Study Guide)”, Microsoft Press, 1st Edition, 2020.

