LORDS INSTITUTE OF ENGINEERING AND TECHNOLOGY

(An Autonomous Institution)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHEME OF INSTRUCTIONS & EXAMINATION [LR-23]

(W.e.f Academic Year 2026-27)

IV-B.E. VII-Semester

(Tentative)

S. No.	Course Code	Category	Course Title	Scheme of Instructions				Scheme of Examination			CREDITS
				L	T	P/D	Contact Hours/Week	CIE	SEE	Duration in Hours	
Theory Course											
1	U23CS701	PCC	Distributed Systems	2	1	-	3	40	60	3	3
2	U23CS702	PCC	Deep Learning Technique	2	-	-	3	40	60	3	3
3	U23CS703	PCC	Full Stack Web Development	2	-	-	3	40	60	3	2
4	U23CS704	PCC	Cryptography and Network Security	2	1	-	3	40	60	3	3
5	---	PEC	Professional Elective – III	3	-	-	3	40	60	3	3
6	---	OEC	Open Elective-III	3	-	-	3	40	60	3	3
Practical/ Laboratory Course											
7	U23CS7L1	PCC	Deep Learning Technique Lab	-	-	3	3	25	50	3	1.5
8	U23CS7L2	PCC	Full Stack Web Development Lab	-	-	3	3	25	50	3	1.5
Project											
9	U23CS7P1	PROJ	Project Work Phase-I	-	-	4	4	100	--	--	2
Total				**14**	**-**	**17**	**33**	**415**	**510**	**--**	**22**

L: Lecture *(Hrs/Wk/Sem)* **T**: Tutorial *(Hrs/Wk/Sem)* **P**: Practical **D**: Drawing *(Hrs/Wk/Sem)*

CIE: Continuous Internal Evaluation **SEE**: Semester End Examination
OEC: Open Elective Course **PCC**: Program Core Courses
PEC: Professional Elective Course **CS:** Computer Science
PROJ: Project

Note:
1. Each contact hour is a Clock Hour.
2. The duration of the practical class is three hours, however it can be extended wherever necessary, to enable the student to complete the experiment.

Course Code	Course Title						Core/Elective
U23CS701	DISTRIBUTED SYSTEMS						Core
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
OS	2	-	-	-	40	60	2

Course Objectives:

The objectives of this course are

1. To acquire an understanding of the issues in distributed systems.
2. To learn about Naming and synchronization with different algorithms.
3. To study architectures and working of Distributed file systems, Distributed web-based system
4. To expose the students to distributed transaction management, security issues and replication.
5. To introduce Emerging trends in distributed computing

Course Outcomes:

After Successful completion of this course, student will be able to

1. List the principles of distributed systems and describe the problems and challenges associated with these principles
2. To know about interposes communication and remote communication.
3. Understand Distributed Computing techniques, Synchronous and Processes.
4. Understand Distributed File Systems Apply Distributed web-based system. Understand the importance of security in distributed systems
5. Student will be able to know distributed service-oriented architecture.

UNIT – I

Introduction: Characteristics & Properties of Distributes Systems – Taxonomy - Types of Distributed Systems Design goals – Transparency Issues.

Architectures: Architectural Styles, System Architectures, Architectures versus Middleware, and Self- Management in Distributed Systems.

Processes: Threads, Virtualization, Software Agents, Clients, Servers, and Code Migration. **Communication:** Inter process communication Mechanisms, Remote Procedure Call, Remote Method Invocation, Message-Oriented Communication, Stream- Oriented Communication, and Multicast Communication.

UNIT – II

Naming: Names, Identifiers and Addresses, Flat Naming, Structured Naming and Attribute-Based Naming. **Synchronization:** Clock Synchronization, Logical Clocks, Mutual Exclusion, Global Positioning of Nodes, and Election Algorithms.

Consistency and Replication: Introduction, Data-Centric Consistency Models, Client-Centric Consistency Models, Replica Management, and Consistency Protocols.

UNIT-III

Fault Tolerance: Introduction to Fault Tolerance, Process Resilience, Reliable Client- Server Communication, Reliable Group Communication, Distributed Commit, and Recovery.

Distributed Object-Based Systems: CORBA, DCOM, GLOBE -Architecture, Processes, Communication, Naming, Synchronization, Consistency and Replication, Fault Tolerance, and Security

UNIT-IV

Distributed File Systems: File system, DFS- definition, Characteristics, Goals, SUN NFS-NFS Architecture, NFS Implementation, Protocols, The CODA file system-Design Overview, An Example, Design Rational, Implementation, The GOOGLE file system-Definition, Architectures, GFS Architecture

Distributed Web-Based Systems: Traditional Web-Based Systems, Web Services Fundamentals, The Apache Web Server, Web Server Clusters, Communication, HTTP Fundamentals, Simple Object Access Protocol SOAP, Web Proxy Caching, Replication for Web Hosting Systems-CDN‟S, Service-Oriented Architectures, REST and Web Services

UNIT -V

Distributed Coordination-Based Systems -- Architecture, Naming and Security

Emerging Trends in Distributed Systems - Emerging Trends Introduction, Grid Computing, Cloud Computing and its roots in distributed systems mechanisms and self-management of distributed systems, Virtualization, Service Oriented Architecture, The Future of Emerging Trends. **Map-Reduce**: Example, Scaling, Programming Model, Apache Hadoop, Amazon Elastic Map Reduce, Mapreduce.net, Pig and Hive.

Suggested Readings:

1. “Distributed Systems” Andrew S. Tanenbaum and Maarten Van Steen, PHI 2nd Edition, 2009.
2. “Distributed Computing”, Sunita Mahajan and Seema Shah, Oxford University
3.”Guide to Cloud Computing” R. Hill, L. Hirsch, P. Lake, S. Moshiri, Principles and Practice‖, Springer, 2013.
4. “Cloud Computing-Principles and Paradigms” R. Buyya, J. Borberg, A. Goscinski, , Wiley, 2013.
5. "Distributed Operating Systems” by P. K. Sinha, PHI

Course Code	Course Title					Core/Elective
U23CS702	DEEP LEARNING TECHNIQUE					Core Course
Prerequisites	Contact Hour per Week			CIE	SEE	Credit
	L	T	P			
Python programming	3	-	-	40	60	3

Course Objectives:

This course will enable students to

1. Understand the concept of neural networks, convolutional neural networks, and recurrent neural networks.
2. Implement deep learning algorithms, and learn how to train deep networks.
3. Gain in-depth knowledge of Tensor Flow along with its functions, operations, and the execution pipeline.
4. Understanding the major Architectures of Neural Networks and getting into the Convolutional neural Networks.
5. Understand the applications of implementing deep learning such as image processing, natural language processing

Course Outcomes:

On completion of this course, the students are able to

6. Understand the fundamentals of deep learning.
7. Understand deep learning algorithms and design neural network.
8. Train and implement a neural network.
9. Gain knowledge about convolutional neural networks.
5. Apply neural networks in various fields.

UNIT - I

Introduction:

What is deep learning? Artificial intelligence, Machine learning, and Deep learning - Artificial intelligence - Machine learning – Learning representations from data - The “deep” in deep learning -Understanding how deep learning works, in
three figures -What deep learning has achieved so far- The promise of AI.

UNIT - II

Neural Network:

Getting started with neural networks - Anatomy of a neural network - Layers: the building blocks of deep learning - Models: networks of layers - Loss functions and optimizers: keysto configuring the learning process The Neural Network-Building Intelligent Machines, The Limits of Traditional Computer Programs, The Mechanics of Machine Learning, The Neuron, Expressing Linear Perceptrons as Neurons, Feed-Forward Neural Networks, Linear Neurons and Their Limitations, Sigmoid, Tanh.

UNIT- III

Training Feed-Forward Neural Networks - The Fast-Food Problem - Gradient Descent - The Delta Rule and Learning Rates - Gradient Descent with Sigmoidal Neurons – The Back propagation Algorithm - Stochastic and Minibatch Gradient Descent - Test Sets, Validation Sets, and Overfitting - Preventing Overfitting in Deep Neural Networks

UNIT -IV

Introduction to Major Architectures of Deep Networks–Unsupervised Pretrained Networks (UPNs), Convolutional Neural Networks (CNNs), Recurrent Neural Networks, Recursive Neural Networks

UNIT -V

Convolutional Neural Networks -Neurons in Human Vision - The Shortcomings of Feature Selection - Vanilla Deep Neural Networks Don't Scale - Filters and Feature Maps – Full Description of the Convolutional Layer - Max Pooling - Full Architectural Description of Convolution Networks - Closing the Loop on MNIST with Convolutional Networks - Accelerating Training with Batch Normalization.

Suggested Readings:

1. Fundamentals of Deep Learning : Designing Next-Generation Machine Intelligence Algorithms, Nikhil Buduma and
2. Nicholas Locascio, First Edition - O'Reilly , 2017
3. Deep Learning with Python ,Francois Chollet, Second Edition, Manning Publications,2017.
4. Deep Learning: A Practitioner's Approach , Josh Patterson and Adam Gibson, First Edition - O'Reilly , 2017.

Course Code	Course Title					Core/Elective
U23CS703	FULL STACK WEB DEVELOPMENT					Core Course
Prerequisites	Contact Hour per Week			CIE	SEE	Credit
	L	T	P			
HTML/CSS Basics	3			40	60	3

Course Objectives:

1.Understand the fundamentals of web technologies and web architecture.
2.To implement the structure and design of web pages using HTML and CSS.
3.Develop interactive client-side applications using JavaScript and React.js.
4.Build server-side applications using Node.js and Express.js.
5.Integrate databases and APIs to develop full-stack web applications.

Course Outcomes:

At the end of the course, student would be able to

1.Explain web technologies, web architecture, and Internet-based applications.
2.Design responsive and structured web pages using HTML, CSS, Flexbox, and Grid.
3.Develop interactive client-side applications using JavaScript and React.js.
4.Build server-side applications and RESTful APIs using Node.js and Express.js.
5.Develop database-driven full-stack applications using MongoDB and modern web technologies.

UNIT – I

HTML – Introduction to HTML, HTML Versions, Basic Structure of HTML Page, Basic Tags, Types of Tags, Semantic HTML, Lists, Tables, Images, Forms, Buttons, and Moving Images.
CSS - Introduction, CSS Versions, A Simple Specification, Types of Style Sheets, Style Classes, Font Properties, Background properties, text properties, Box Model and its properties, Position properties, Layouts - flexbox, Grid, Media Query for Responsive design. Introduction to Tailwindcss and Shadcn.

UNIT – II

Javascript - Introduction, Usage of variables, Operators, Control structures, Looping structures, predefined keywords, arrays, predefined functions, arrays and functions, mathematical functions, String functions, expressions, pattern matching using RegEXp Class, String Class, Exception Handling, Date Object, Form Validation, DOM, Manipulation, Classes and Objects, Introduction to JSON.

UNIT- III

React.js – Introduction to Reactjs, NPM, NPX, Installing React Library, Simple React Structure, The Virtual DOM, State, Props, Components and its types, Lifecycle, Data and Data Flow, Stateful and Stateless Components, Events, Router, Forms validations, Hooks and its types, Single page Application.

UNIT – IV

Node.JS : Features and drawbacks, Environment setup for Node.js, Program Architecture, Node.js web server, Global objects, Error Handling, Event loop, File system, Database connection using Mongodb.
Express.JS : API Methods – GET, POST, PUT, DELETE, Request and response objects, URL and Query Parameters, Routing, Templates, Middleware and MVC pattern.

UNIT - V

MongoDB : Overview of different types of databases, MongoDB, Configuring Server and Client, MongoDB Compass, Creating Database, MongoDB Commands, MongoDB CRUD Operations, **REST API -**Introduction to REST API,REST Architecture, GraphQL.

Suggested Readings:

1. Thomas A Powel "The Complete Reference HTML & CSS", 5th Edition, McGraw Hill.
2. Vasan Subramanian, "Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node", 2nd Edition, Apress.
3. Ketan Agnihotri, Pranali Dahale, " React Development using Typescript: Modern web app development using advanced React techniques ", English Edition, 2024.
4. Nabendu Biswas, "Ultimate Full-Stack Web Development with MERN: Design, Build, Test and Deploy Production-Grade Applications with MongoDB, Express, React and NodeJS", 1st Edition, 2023.

Online Resources:

1. https://react.dev/
2. https://nodejs.org/learn/
3. https://expressjs.com/en/
4. https://www.mongodb.com/docs/
5. https://www.w3schools.com/

Course Code	Course Title					Core/Elective
U23CS704	CRYPTOGRAPHY AND NETWORK SECURITY					Core Course
Prerequisites	Contact Hour per Week			CIE	SEE	Credit
	L	T	P			
CN, IS	3			40	60	3

Course Objectives:

1. Familiarize students with basics of Network Security and cryptography.
2. To understand the network security, services, attacks, mechanisms, types of attacks on TCP/IP protocol suite.
3. To comprehend and apply network layer security protocols, Transport layer security protocols, Web security protocols
4. To comprehend and apply authentication services, authentication algorithms.
5. Provide a solid understanding of main issues related to network security

Course Outcomes:

Student will able to:

1. Understand the most common type of information and network threat sources.
2. Be able to determine appropriate mechanisms for protecting the network.
3. Design a security solution for a given application system with respect to security of the system.
4. Understand the information and network security issues and apply the related concepts for protection and communication privacy.
5. Comprehend various network security threats.

UNIT - I

Security Concepts: Introduction, The need for security, Security approaches, Principles of security, Types of Security attacks, Security services, Security Mechanisms, A model for Network Security Cryptography Concepts and Techniques: Introduction, plain text and cipher text, substitution techniques, transposition techniques, encryption and decryption, symmetric and asymmetric key cryptography.

UNIT - II

Symmetric public key Ciphers: Block Cipher principles, DES, AES, Blowfish, RC5, IDEA, Block cipher operation, Stream ciphers, RC4. Asymmetric key Ciphers: Principles of key cryptosystems, RSA algorithm, Diffie-Hellman Key Exchange.

UNIT - III

Cryptographic Hash RSA algorithm, Functions: Message Authentication, Secure Hash Algorithm (SHA-512), Message authentication codes: Authentication requirements, HMAC, CMAC, Digital signatures. Key Management and Distribution: Symmetric Key Distribution Using Symmetric & Asymmetric Encryption, Distribution of Public Keys, Kerberos, X.509 Authentication Service.

UNIT - IV

Layer Transport-level Security: Web security considerations, Secure Socket and Transport Layer Security, HTTPS, Secure Shell (SSH) Wireless Network Security: Wireless Security, Mobile Device Security, IEEE 802.11 Wireless LAN, IEEE 802.11i Wireless LAN Security

UNIT - V

E-Mail Security: Security: Pretty Good Privacy, S/MIME IP Security: IP Security overview, IP Security architecture, Authentication Header, Encapsulating security payload, Combining security associations.

Suggested reading:

1.Cryptography and Network Security Principles and Practice: William Stallings, Pearson Education, 6th Edition

2. Cryptography and Network Security: AtulKahate, McGraw Hill, 3rd Edition

Course Code	Course Title					Core/Elective
U23CS705	SECURE SOFTWARE ENGINEERING					PE-III
Prerequisites	Contact Hour per Week			CIE	SEE	Credit
	L	T	P			
SE	3	-	-	40	60	3

Course Objectives:

1. Understand developing and testing for secure software.
2. Learning sound security fundamentals from real-world case studies
3. Process model of software engineering with secure software development models.
4. Attain knowledge of the distinction between critical and non-critical systems
5. Planning, scheduling and risk assessment/management for secure software.

Course Outcomes:

Student will be able to

1. Apply contemporary formal mathematical modeling techniques to model and analyze the security of a software system
2. Identify project security risks & selecting risk management strategies.
3. Use statistical methods to collect and analyze metrics for assessing and improving the security of a product, process, and project objectives.
4. Describe and discuss security concerns designs at multiple levels of abstraction
5. Design a software solution for secure access and protection of data.

UNIT – I

Security a software Issue: Introduction, The Problem, Software Assurance and Software Security, Threats to software security, Sources of software insecurity, Benefits of detecting software security defects early, managing secure software development What Makes Software Secure: Defining Properties of secure software, Influencing the security properties of software, Asserting and specifying desired security properties?

UNIT – II

Requirements Engineering for secure software: Introduction, The SQUARE process Model, Requirements elicitation and prioritization.

UNIT- III

Secure Software Architecture and Design:Introduction, Software Security Practices for Architecture and Design: Architectural risk analysis, Software Security Knowledge for Architecture and Design: Security Principles, Security Guidelines, and Attack Patterns. Secure Coding and Testing: Introduction, Code analysis, Coding Practices, Software Security Testing, Security Testing considerations throughout the SDLC.

UNIT – IV

Security and Complexity: System Assembly Challenges: Introduction, Security Failures, Functional and Attacker Perspectives for Security Analysis, System Complexity Drivers and Security, Deep Technical Problem Complexity.

UNIT - V

Governance and Managing for More Secure Software: Governance and security, Adopting an Enterprise Software Security Framework, How much security is enough?, Security and project management, Maturity of Practice.

Suggested Readings:

1.Julia H Allen, Sean J Barnum, Robert J Ellison, Gary McGraw, Nancy R Mead, "Software Security Engineering: A Guide for Project Managers", Addison Wesley, 2008

2. Ross J Anderson, "Security Engineering: A Guide to Building Dependable Distributed Systems", 2nd Edition, Wiley, 2008.

3. Howard, M. and LeBlanc, D., "Writing Secure Code", 2nd Edition, Microsoft Press, 2003

4. Jason Grembi, "Developing Secure Software", First Edition, Cengage Learning, 2008.

5. Gary R. McGraw, "Software Security: Building Security", AddisonWesley Software Security Edition, 2006.

6. Richard Sinn, "Software Security: Theory, Programming and Practice", First Edition, Cengage Learning, 2009.

Course Code	Course Title					Core/Elective
U23CS706	DATA MINING					PE-III
Prerequisites	Contact Hour per Week			CIE	SEE	Credit
	L	T	P			
DBMS	3	-	-	40	60	3

Course Objectives:

1. To introduce the basic concepts of data Mining and its applications
2. To understand different data mining like classification, clustering and Frequent Pattern mining
3. To introduce current trends in data mining

Course Outcomes:

Student will be able to

1. Understand basic data mining concepts, data types, statistical measures, and data similarity techniques.
2. Apply methods to discover frequent patterns, associations, and correlations in data.
3. Use classification techniques like decision trees, Bayesian methods, and SVM for prediction.
4. Perform clustering using different methods and evaluate cluster quality.
5. Understand recent trends, applications, and societal impact of data mining.

UNIT – I

Introduction: introduction to Mining and Data Mining , What kinds of data can be mined? What kinds of patterns can be mined? Which technologies are used? Which kinds of applications are Targeted? Major issues in Data Mining. Getting to know your data: Data objects and attributed types. Basic statistical descriptions of data. Data visualization, Measuring data similarity and dissimilarity.

UNIT – II

Mining frequent patterns, Associations and correlations: Basic concepts and methods, Frequent Item set Mining Methods, Pattern evaluation methods.

UNIT – III

Classification: Basic concepts, Decision tree induction, Bayes classification methods, Advance methods, Bayesian Belief Network, Classification by back propagation, Support vector machine.

UNIT – IV

Cluster Analysis: Concepts and Methods: Cluster Analysis, Partitioning Methods, Hierarchical Methods, Density-Based Methods, Grid-Based Methods, Evaluation of clustering.

UNIT – V

Data Mining Trends and Research Frontiers: Mining Complex Data Types, Other Methodologies of Data Mining, Data Mining Applications, Data Mining and Society, Data Mining trends.

Suggested Readings:

1. Jiawei Han, Micheline Kamber, Jin Pei, Data Mining: Concepts & Techniques, 3rd Edition., Morgon Koffman, 2011
2. Vikram Pudi, P. Radha Krishna, DataMining, Oxford University Press, 1st Edition, 2009
3. Ning Tan, Michael Steinbach, Vipin KumalntroductiontoDataMining, Pearson Education, 2008

Course Code	Course Title					Core/Elective
U23CS707	BLOCKCHAIN TECHNOLOGY					PE-III
Prerequisites	Contact Hour per Week			CIE	SEE	Credit
	L	T	P			
CN	3	-	-	40	60	3

Course Objectives:

1.Understand how blockchain systems (mainly bitcoin, Ethereum) works
2.To securely interact with them
3.Design, build and deploy smart contracts and distributed applications
4.Integrate ideas from blockchain technology into their own projects

Course Outcomes:

Student will be able to

1.Explain design principles of bitcoin and Ethereum, Nakamoto consensus, simplified payment verification protocol
2. List and describe the differences between PoW and PoS consensus
3. Interact with a blockchain system by sending and reading transactions
4. Design, build and deploy smart contracts and distributed applications
5. Evaluate security, privacy and efficiency of a given block chain system.

UNIT – I

Basics: Distributed database, Two general problem,Byzantine general problem and Fault Tolerance, Hadoop Distributed File System, Distributed Hash Table, ASIC resistance, Turing complete. Cryptography: Hash function, Digital signature - ECDSA, Memory Hard Algorithm, Zero Knowledge Proof.

UNIT – II

Blockchain: Introduction, Advantages over conventional distributed database, blockchain network, mining mechanism, distributed consensus, Merkle Patricia Tree, Gas Limit, Transactions and Fee, Anonymity, Reward, Chain Policy, Life of Blockchain application, Soft & Hard Fork, Private and Public blockchain.

UNIT- III

Distributed Consensus: Nakamoto consensus, Proof of Work, Proof of Stake, Proof of Burn, Difficulty Level, Sybil attack, Energy Utilization and alternate.

UNIT – IV

Cryptocurrency: History, Distributed Ledger, Bitcoin Protocols - Mining Strategy and Rewards, Ethereum-Construction, DAO, Smart Contracts, GHOST, Vulnerability, Attacks, Sidechain, Namecoin

UNIT - V

Cryptocurrency Regulation: Stakeholders, Roots of Bitcoin, Legal Aspects- Crypto Currency Exchange, Black Market and Global Economy Applications: Internet of Things, Medical Record Management System, Domain Name Service and Future of Blockchain. Case Study: Naive Blockchain construction, Memory Hard Algorithm - Hash cash Implementation, Direct Acyclic Graph, Play with Go-Ethereum, Smart Contract Construction, Toy Application using Blockchain, Mining Puzzles.

Suggested Readings:

1. Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction, Princeton University Press, 2016 - Aravind Narayanan, Joseph Bonneou, Edward Felten, Andrew Miller and Steven Goldfeder

2. 'Ethereum- A Secure Decentralized Transaction Ledger'- Dr.Gavin Wood, Yellow Paper,2014

3. A survey of Attacks on Ethereum Smart Contracts, Nicola Atzei, Massimo Bartoletti and TizianaCimoli

4. Bitcoin: A Peer-to-Peer Electronic Cash System, Satoshi Nakamoto

Course Code	Course Title					Core/Elective
U23CS708	FUNDAMENTAL OF VIRTUAL REALITY					PE-III
Prerequisites	Contact Hour per Week			CIE	SEE	Credit
	L	T	P			
AI.	3	-	-	40	60	3

Course Objectives:

1. Understand basics of AR, VR, MR, and XR.
2. Learn AR tools and tracking methods.
3. Study VR devices and environments.
4. Know interaction and rendering techniques.
5. Analyze applications and challenges of AR/VR.

Course Outcomes:

After completing this course, the student will be able to

1. Explain AR, VR, MR, and XR concepts.
2. Develop simple AR applications.
3. Create basic VR environments.
4. Apply interaction and rendering methods.
5. Evaluate AR/VR applications and issues.

Unit–I: Fundamentals of AR and VR

Introduction to Augmented Reality (AR) and Virtual Reality (VR): evolution and history of immersive technologies; definitions of AR, VR, MR, and XR; differences between AR and VR; applications in engineering, healthcare, education, gaming, and industry; components of AR/VR systems; hardware requirements; sensors, cameras, displays, and head-mounted devices.

Unit–II: Augmented Reality Systems

Architecture of AR systems: marker-based and markerless tracking techniques; feature detection and object recognition; mobile AR concepts and applications; AR development tools and platforms (AR.js, Vuforia, Unity OpenXR plugin); designing and developing basic AR applications.

Unit–III: Virtual Reality Systems

Introduction to VR devices and controllers: motion tracking systems; concepts of locomotion, field of view, immersion, and presence; representation of virtual environments; geometric transformations in VR; viewing transformations; human vision and perception in VR; development of VR applications using A-Frame framework.

Unit–IV: Interaction and Rendering in AR/VR

User interaction techniques in AR/VR: gesture recognition; voice and controller-based interaction; UI/UX design principles for immersive systems; spatial audio and haptic feedback; visual perception of depth, motion, and color; rendering techniques including rasterization, ray tracing, shading models; latency reduction and performance optimization.

Unit–V: Ethics, Challenges, and Applications of AR/VR

Privacy, security, and ethical issues in AR/VR systems: motion sickness and user comfort; accessibility challenges; case studies of AR/VR in healthcare, education, military training, manufacturing, tourism, and smart cities; future trends in immersive technologies and career opportunities in AR/VR.

Suggested Readings:

1. Chetankumar G. Shetty, "Augmented Reality & Virtual Reality", Notion Press, 2022 (Open educational content)

2. Jason Jerald, "The VR Book: Human-Centered Design for Virtual Reality", ACM Books, 2015

3. Alan B. Craig, "Understanding Augmented Reality: Concepts and Applications", Morgan Kaufmann

4. Tony Parisi, "Learning Virtual Reality: Developing Immersive Experiences and Applications for Desktop, Web, and Mobile", O'Reilly Media

5. Joseph Hocking, "Unity in Action", Manning Publications (with focus on open-source compatible plugins)

Course Code	Course Title					Core/Elective
U23CS801	SOFTWARE QUALITY & ASSURANCE					Core Course
Prerequisites	Contact Hour per Week			CIE	SEE	Credit
	L	T	P			
SE	3	-	-	40	60	3

Course Objectives:

1.Introduces components of software quality assurance systems before, during, and after software development

2.Presents a framework for software quality assurance and explains the individual components of the framework.

Course Outcomes:

After completing this course, the student will be able to

1.Judge the software quality factors for a defined project.

2.Integrate the quality activities in the project life during implementation

3.Propose some corrective and preventive actions for maintaining software quality

4.Manage components of software quality in the project life cycle

5.Process SQA management standards in a project.

UNIT-I

Introduction to Software Quality: Definition, Software quality factors, Overview of the components of the Software Quality Assurance(SQA) System

UNIT-II

Pre-project software quality components: Contract review, Development and quality plans SQA components in the project life cycle: Integrating quality activities in the project life cycle, Reviews, Assuring the quality of software maintenance components, CASE tools.

UNIT-III

Software quality infrastructure components: Procedures and work instructions, Supporting quality devices, Staff training and certification, Corrective and preventive actions, Configuration management, Documentation control.

UNIT-IV

Management components of software quality: Project progress control, Software quality metrics, Costs of software quality.

UNIT-V

Standards, certification and assessment: Quality management standards (ISO 9001 and ISO 9000-3, CMM, CMMI, SPICE, ISO/IEC 15504), SQA project process standards – IEEE software engineering standards .

Suggested Readings:

1. Daniel Galin, Software Quality Assurance : From Theory to Implementation, Addison Wesley, 2003.

2. Stephen Kan, Metrics and Models in Software Quality Engineering (2nd Edition), Addison Wesley, 2002.

3. Schulmeyer, G. Gordon and McManus, James, (eds), Handbook of Software Quality Assurance, 3rd Ed. Prentice Hall, 1999.

Course Code	Course Title					Core/Elective
U23CS802	INFORMATION RETRIEVAL SYSTEM					PE-IV
Prerequisites	Contact Hour per Week			CIE	SEE	Credit
	L	T	P			
DS	3	-	-	40	60	3

Course Objectives:

1. To learn about the relationships to data base management systems and libraries.
2. To understand the important concepts, algorithms, and data/file structures
3. To learn about clusters, searching techniques and visualization.
4. To expose the students to find the searching algorithms and Evaluation systems.
5. To facilitate design, and implementation of Information Retrieval (IR) systems.

Course Outcomes:

1. Analyze the retrieval information systems and its capabilities.
2. Analyze and apply appropriate data structures and indexing concepts.
3. Use the various searching techniques for improving the information visualization.
4. Apply the searching algorithms to evaluate information systems.
5. Use the multimedia information retrieval of query languages

UNIT-I:

Introduction to Information Retrieval Systems: Definition of Information Retrieval System, Objectives of Information Retrieval Systems, Functional Overview: Item Normalization, Selective Dissemination of Information, Document, Index and Multimedia Database Search, Information Retrieval System Capabilities: Search Capabilities, Browse Capabilities, Miscellaneous Capabilities, Z39.50 and WAIS Standards

UNIT – II:

Cataloging and Indexing: History and Objectives of Indexing, Indexing Process, Scope of Indexing, Linkages and Precoordination, Automatic Indexing: Classes of Automatic Indexing, Statistical Indexing, Natural Language, Concept Indexing, Hypertext Linkages, Information Extraction, Index Compression: Dictionary Compression, Posting File Compression

UNIT – III:

User Search Techniques and Scoring: Search Statements and Binding, Similarity Measures and Ranking, Relevance Feedback, Selective Dissemination of Information Search, Weighted Searches of Boolean Systems, Searching the INTERNET and Hypertext, Computing Scores in a Complete Search System: Efficient Scoring and Ranking, Components of an Information Retrieval System

UNIT – IV:

Evaluation and Information Visualization: Evaluation in Information Retrieval, Standard Test Collections, Evaluation of Unranked Retrieval Sets, Evaluation of Ranked Retrieval Results, Assessing Relevance, System Quality

and User Utility, Information Visualization: Cognition and Perception, Aspects of Visualization Process, Information Visualization Technologies.

UNIT – V:

Multimedia Information Retrieval: Introduction to Text Search Techniques, Software Text Search Algorithms, Hardware Text Search Systems, Spoken Language Audio Retrieval, Non-Speech Audio Retrieval, Graph Retrieval, Image Retrieval, Video Retrieval, Challenges in Multimedia IR, Applications and Future Trends

Suggested Readings:

1. W.B. Frakes, Ricardo Baeza-Yates, Information Retrieval: Data Structures and Algorithms, Prentice Hall, 1992.

2. Robert Korfhage, Information Storage & Retrieval, John Wiley & Sons.

3.Ricardo Baeza-Yates and Berthier Ribeiro-Neto, Modern Information Retrieval, Pearson Education.

Course Code	Course Title					Core/Elective
U23CS803	CLOUD COMPUTING					PE-IV
Prerequisites	Contact Hour per Week			CIE	SEE	Credit
	L	T	P			
OS	3	-	-	40	60	3

Course Objectives:

1.To understand the various distributed system models and evolving computing paradigms
2. To gain knowledge in virtualization of computer resources
3. To realize the reasons for migrating into cloud
4. To introduce the various levels of services that can be achieved by a cloud.

Course Outcomes:

Student will be able to
1.Ability to understand various service delivery models of a cloud computing
2. Ability to understand the ways in which the cloud can be programmed and deployed.
3. Ability to understand the virtualization and cloud computing concepts
4. Assess the comparative advantages and disadvantages of Virtualization technology
5. Analyze authentication, confidentiality and privacy issues in cloud computing

UNIT – I

Cloud Computing Fundamentals: Definition of Cloud computing, Roots of Cloud Computing, Layers and Types of Clouds, Desired Features of a Cloud, Cloud Infrastructure Management, Infrastructure as a Service Providers, Platform as a Service Providers. Computing Paradigms: High-Performance Computing, Parallel Computing, Distributed Computing, Cluster Computing, Grid Computing.

UNIT – II

Migrating into a Cloud: Introduction, Broad Approaches to Migrating into the Cloud, the Seven-Step Model of Migration into a Cloud, Enriching the 'Integration as a Service' Paradigm for the Cloud Era, the Onset of Knowledge Era the Evolution of SaaS, Evolution of Saas.

UNIT- III

Infrastructure as a Service (IAAS) & Platform (PAAS): Virtual machines provisioning and Migration services, Virtual Machines Provisioning and Manageability, Virtual Machine Migration Services, VM Provisioning and Migration in Action. On the Management of Virtual machines for Cloud Infrastructures-Aneka—Integration of Private and Public Clouds.

UNIT – IV

Software as a Service (SAAS) &Data Security in the Cloud: Software as a Service SAAS), Google App Engine – Centralizing Email Communications- Collaborating via WebBased Communication Tools-An Introduction to the idea of Data Security.

UNIT - V

SLA Management in cloud computing: Traditional Approaches to SLO Management, Types of SLA, Life Cycle of SLA, SLA Management in Cloud.

Suggested Readings:

TEXTBOOKS:

1. Essentials of cloud Computing: K. Chandrasekhran, CRC press, 2014
2. Michael Miller, Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online, Que Publishing, August 2008.
3. Cloud Computing: A Practical Approach, Anthony T.Velte, Toby J.Velte, Robert Elsenpeter, Tata McGraw Hill,rp2011.
4. Enterprise Cloud Computing, Gautam Shroff, Cambridge University Press,2010.
5. Cloud Computing: Implementation, Management and Security, John W.Rittinghouse, James F.Ransome, CRC Press,rp2012.
6. Cloud Application Architectures: Building Applications and Infrastructure in the Cloud, George Reese, O'reilly, SPD,rp2011.
7. Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, Tim Mather, Subra Kumaraswamy, Shahed Latif, O'Reilly, SPD, rp2011

Course Code	Course Title					Core/Elective
U23CS804	ENTERPRISES RESOURCE PLANNING					PE-IV
Prerequisites	Contact Hour per Week			CIE	SEE	Credit
	L	T	P			
BEFA	3	-	-	40	60	3

Course Objectives:

1. To understand the fundamentals and architecture of ERP systems.
2. To learn the core functional modules of ERP.
3. To analyze and evaluate ERP packages and solutions.
4. To study ERP applications in different industries.
5. To understand ERP implementation and emerging trends.

Course Outcomes:

Student will be able to

1. Explain the concepts and benefits of ERP systems.
2. Identify and analyze major ERP functional modules.
3. Evaluate ERP packages for organizational needs.
4. Assess ERP applications across various industries.
5. Apply ERP implementation and integration techniques effectively.

UNIT – I: Introduction to ERP Systems

ERP Overview, Integrated Management Information Systems, Supply Chain Management, Integrated Data Model, Benefits of ERP, Evolution of ERP and Modern Enterprise, Business Process Reengineering (BPR) and ERP, Business Modeling for ERP.

UNIT – II: ERP Functional Areas

Customer Service Management, Customer Relationship Management, Production Planning and Execution, Manufacturing Processes, Purchasing Management, Procurement Cycle, Goods Receipt Process, Financial Management, Financial Accounting, Cost Management, Performance Measurement and Other Business Metrics.

UNIT – III: ERP Packages and Solutions

Introduction to ERP Packages, Characteristics of ERP Packages, Selection Criteria for ERP Packages, ERP Package Evaluation, ERP Vendors and Solutions, Implementation Considerations, Customization and Configuration of ERP Systems.

UNIT – IV: ERP Applications and Industry Case Studies

ERP in Insurance Industry, ERP in Banking Industry, ERP in Pharmaceutical Industry, ERP in Healthcare Industry, ERP in Consumer Products Industry, ERP in Retail Industry, ERP in Universities and Educational Institutions, ERP in Transport Industry, ERP in Telecom Industry, ERP in Public Sector Organizations, Industry-Specific ERP Case Studies.

UNIT – V: ERP Implementation and Emerging Trends

Current Trends in ERP Implementations, ERP Deployment Strategies, Hardware and Network Selection, Data Management Requirements, Data Migration and Data Quality, Integration Requirements and Techniques, Enterprise Application Integration, Security and Compliance Considerations, Non-Functional Requirements for ERP Implementations, Challenges and Best Practices in ERP Projects.

Suggested Books:

1. INTRODUCTION TO MATERIALS MANAGEMENT 6/E (English) 6th Edition Authors: Tony Arnold / Stephen Chapman Publishers: Pearson India
2. Manufacturing Planning and Control for Supply Chain Management (APICS / CPIM Certification Edition) Authors: Thomas E. Vollmann, CFPIM, William L. Berry, D. Clay Whybark, and F. Robert JacobsPublishers: McGraw Hill Education
3. Accounting Handbook 6th Edition Authors: Shim Siegal Publishers: Barrons Educational Series
4. Operations Strategy 4th Edition Authors: Nigel Slack and Mike Lewis Publishers: Pearson
5. CONCEPTS IN ENTERPRISE RESOURCE PLANNING Authors: Ellen F. Monk, Bret J. Wagner
6. Enterprise Resource Planning – Ashim Raj Singla, Cengage Learning

Course Code	Course Title					Core/Elective
U23CS805	MALWARE ANALYSIS & DETECTION					PE-V
Prerequisites	Contact Hour per Week			CIE	SEE	Credit
	L	T	P			
NS	3			40	60	3

Course Objectives:

1. To understand the fundamentals of malware, its evolution, types, and security threats.
2. To learn static malware analysis techniques using executable file structures and reverse engineering concepts.
3. To study dynamic malware analysis methods through monitoring system behavior and debugging tools.
4. To analyze malware functionalities such as persistence, privilege escalation, and process injection.
5. To understand modern malware detection techniques including signature-based and machine learning methods.

.

Course Outcomes:

At the end of the course, the student will be able to:

1. Identify different types of malware and explain their characteristics and impact on systems.
2. Perform static analysis of suspicious files using PE structure, assembly code, and antivirus tools.
3. Conduct dynamic malware analysis using debugging, sandboxing, and network monitoring tools.
4. Evaluate malware behavior such as backdoors, credential theft, and covert execution methods.
5. Apply appropriate malware detection and prevention techniques for cybersecurity defense.

Unit-I

INTRODUCTION: Introduction to malware, OS security concepts, malware threats, evolution of malware, malware types-viruses, worms, rootkits, Trojans, bots, spyware, adware, logic bombs, malware analysis, static malware analysis, dynamic malware analysis.

Unit-II

STATIC ANALYSIS:X86 Architecture- Main Memory, Instructions, Opcodes and Endianness, Operands, Registers, Simple Instructions, The Stack, Conditionals, Branching, Rep Instructions, C Main Method and Offsets. Antivirus Scanning, Fingerprint for Malware, Portable Executable File Format, The PE File Headers and Sections, The Structure of a Virtual Machine, Reverse-Engineering- x86 Architecture, recognizing c code constructs in assembly, c++ analysis, Analyzing Windows programs, Anti-static analysis techniques-obfuscation, packing, metamorphism, polymorphism.

Unit-III

DYNAMIC ANALYSIS: Live malware analysis, dead malware analysis, analyzing traces of malware- system-calls, api-calls, registries, network activities. Anti-dynamic analysis techniques-anti-vm, runtime-evasion techniques,, Malware Sandbox, Monitoring with Process Monitor, Packet Sniffing with Wireshark, Kernel vs. User-Mode Debugging, OllyDbg, Breakpoints, Tracing, Exception Handling, Patching

Unit-IV

Malware Functionality: Downloader, Backdoors, Credential Stealers, Persistence Mechanisms, Privilege Escalation, Covert malware launching- Launchers, Process Injection, Process Replacement, Hook Injection, Detours, APC injection.

Unit-V

Malware Detection Techniques: Signature-based techniques: malware signatures, packed malware signature, metamorphic and polymorphic malware signature Non-signature based techniques: similarity-based techniques, machine-learning methods, invariant inferences.

Suggested Books:

1. Practical malware analysis The Hands-On Guide to Dissecting Malicious Software by Michael Sikorski and Andrew Honig ISBN-10: 159327-290-1, ISBN-13: 978-1-59327-290-6,2012 2
2. Computer viruses: from theory to applications by Filiol, Eric Springer Science & Business Media, 2006
3. Android Malware by Xuxian Jiang and Yajin Zhou, Springer ISBN 978-1-4614-7393-0,2005
4. Hacking exposed™™ malware & rootkits: malware & rootkits security secrets & Solutions by Michael Davis, Sean Bodmer, Aaron Lemasters, McGraw-Hill, ISBN: 978-0-07-159119-5, 2010
5. Windows Malware Analysis Essentials by Victor Marak, Packt Publishing, 2015

Course Code	Course Title					Core/Elective
U23CS806	BIG DATA ANALYTICS					Core Course
Prerequisites	Contact Hour per Week			CIE	SEE	Credit
	L	T	P			
DM	3	-	-	40	60	3

Course Objectives:

1. Understand big data for business intelligence
2. Understand big data for business intelligence
3. Defend big data Without SQL
4. Discuss the process of data analytics using Hadoop and related tools.

Course Outcomes:

At the end of the course, the student will be able to

1. Understand Big Data concepts, sources, and Hadoop basics.
2. Explain HDFS architecture and Hadoop file operations.
3. Develop basic MapReduce programs and understand processing.
4. Use Pig and Hive for data processing and querying.
5. Understand HBase and ZooKeeper basics and operations.

UNIT – I

Big Data Analytics: Define Big Data, Affects of Big Data on our daily lives, Data Sizes, Source of Big Data, Challenges of Big Data, 5 V's of Big Data, Types of Digital Data, Structured/Unstructured data-Advantages and Sources, Business Intelligence, Comparison business intelligence from traditional big data, Architecture of data warehouse, Analytical tools used for big data analytics, Hadoop environment and its components.

UNIT – II

Apache Hadoop Distributed File System (HDFS): Applications for HDFS, Architecture of HDFS, Basic File System Operations of Hadoop, Anatomy of File read/File operations write in HDFS, Parallel Copying by distcp. The Hadoop I/O: Data Integrity in HDFS, Different file based data structures, Hadoop Compression formats, tools and algorithms in file compression, Serialization, Hadoop configuration, Apache Oozie.

UNIT – III

Developing a MapReduce Application: Execution of MapReduce job, Failures in MapReduce job, Shuffling in MapReduce, Sorting in MapReduce, MapReduce job scheduling, Hierarchy for InputFormat class, General form of map, combiner, reduce functions, Types of input formats in MapReduce, Output formats used in MapReduce. MapReduce Features: Built-in counter groups, Map side join, Reduce side join, Use of side data distribution.

UNIT – IV

Pig Latin: Installing and Running Pig, Comparison with Databases, Different Pig Latin expression, Ways of executing Pig Programs, Built-in functions in Pig, Data Processing Operators, Pig Latin commands/ Pig in Practice. Hive: Installing Hive, The Hive Shell, Comparison with Traditional Databases, HiveQL, Tables, User Defined Functions.

UNIT – V

Hbase: Hbase cluster members, Comman issues while running Hbase cluster (load), Characteristics of Hbase, Hbase table, Comparsion with Hbase with RDBMS. ZooKeeper: Characteristics of ZooKeeper, Zookeeper Commands, Operations in ZooKeeper.

Suggested Readings:

1. Tom White, "Hadoop: The Definitive Guide", Third Edition, O'Reilly, 2012.

2. Eric Sammer, "Hadoop Operations", O'Reilly, 2012.

3. VigneshPrajapati, Big data analytics with R and Hadoop, 2013. 4. E. Capriolo, D. Wampler, and J. Rutherglen, "Programming Hive", O'Reilly, 2012

Course Code	Course Title					Core/Elective
U23CS807	DIGITAL FORENSICS					PE-V
Prerequisites	Contact Hour per Week			CIE	SEE	Credit
	L	T	P			
Cyber Security	3	-	-	40	60	3

Course Objectives:

1.To understand the basic digital forensics and techniques for conducting the forensic examination on different digital devices.

2. To understand how to examine digital evidences such as the data acquisition, identification analysis.

Course Outcomes:

At the end of the course, the student will be able to

1.Apply forensic analysis tools to recover important evidence for identifying computer crime.

2. Be well-trained as next-generation computer crime investigators.

3. Analyzing Processing crimes and incident scenes.

4. Utilizing Current computer forensics tools- software, hardware tools, validating and testing forensic software, addressing data-hiding techniques.

5. Investigating email crime and violations, understanding E-Mail servers, specialized E-Mail forensics tool.

UNIT – I

Computer forensics fundamentals, Benefits of forensics, computer crimes, computer forensics evidence and courts, legal concerns and private issues.

UNIT – II

Understanding Computing Investigations – Procedure for corporate High-Tech investigations, understanding data recovery work station and software, conducting and investigations. Forensic Technology and Practices.

UNIT – III

Data acquisition- understanding storage formats and digital evidence, determining the best acquisition method, acquisition tools, validating data acquisitions, performing RAID data acquisitions, remote network acquisition tools, other forensics acquisitions tools.

UNIT – IV

Processing crimes and incident scenes, securing a computer incident or crime, seizing digital evidence at scene, storing digital evidence, obtaining digital hash, reviewing case.

UNIT – V

Current computer forensics tools- software, hardware tools, validating and testing forensic software, addressing data-hiding techniques, performing remote acquisitions, E-Mail investigations- investigating email crime and violations, understanding E-Mail servers, specialized E-Mail forensics tool.

Suggested Readings:

1. Warren G. Kruse II and Jay G. Heiser, “Computer Forensics: Incident Response Essentials”, Addison Wesley, 2002. 2. Nelson, B, Phillips, A, Enfinger, F, Stuart, C., “Guide to Computer Forensics and Investigations, 2nd ed., Thomson Course Technology, 2006.

3. Vacca, J, Computer Forensics, Computer Crime Scene Investigation, 2nd Edition, Charles River Media, 2005.

Course Code	Course Title					Core/Elective
U23CS808	FUNDAMENTALS OF COMPUTER INTERACTION					PE-V
Prerequisites	Contact Hour per Week			CIE	SEE	Credit
	L	T	P			
OS	3	-	-	40	60	3

Course Objectives:

1. To understand the fundamentals and importance of User Interface (UI) design and Human-Computer Interaction (HCI).
2. To study human factors and their impact on interface usability.
3. To learn principles of effective screen design and navigation.
4. To explore UI components, windows, multimedia, and color usage.
5. To gain knowledge of software tools and interaction devices used in UI development.

Course Outcomes:

At the end of the course, the student will be able to

1. Understand basics of UI and HCI.
2. Analyze human factors in interface design.
3. Design effective screen layouts and navigation.
4. Apply UI components, windows, and colors.
5. Use tools and devices for interface development.

UNIT I

Introduction: Importance of user Interface – definition, importance of good design. Benefits of good design. A brief history of Screen design, The graphical user interface – popularity of graphics, the concept of direct manipulation, graphical system, Characteristics, Web user – Interface popularity, characteristics- Principles of user interface.

UNIT II

Design process – Human interaction with computers, importance of human characteristics human consideration, Human interaction speeds, and understanding business junctions.

UNIT III

Screen Designing : Design goals – Screen planning and purpose, organizing screen elements, ordering of screen data and content – screen navigation and flow – Visually pleasing composition – amount of information – focus and emphasis – presentation information simply and meaningfully – information retrieval on web – statistical graphics – Technological consideration in interface design.

UNIT IV

Windows : New and Navigation schemes selection of window, selection of devices based and screen based controls, Components – text and messages, Icons and increases – Multimedia, colors, uses problems, choosing colors.

UNIT V

Software tools: Specification methods, interface – Building Tools, Interaction Devices – Keyboard and function keys – pointing devices – speech recognition digitization and generation – image and video displays – drivers.

TEXT BOOKS:

1. Developing Java Web Services, R. Nagappan, R. Skoczylas, R.P. Sriganesh, Wiley India, rp 2008.
2. Developing Enterprise Web Services, S. Chatterjee, J. Webber, Pearson Education, 2008.
3. XML, Web Services, and the Data Revolution, F.P.Coyle, Pearson Education.

REFERENCES:

1. Building Web Services with Java, Second Edition, S. Graham and others, Pearson Edn., 2008.
2. Java web services, D.A. Chappell and T.Jewell, O'Reilly,SPD.
3. Java Web Services Architecture, McGovern, Sameer Tyagi etal.., Elsevier.
4. Web Services, G. Alonso, F. Casati and others, Springer, 2005.

Course Code	Course Title						Core/Elective
U23CM7L2	DEEP LEARNING TECHNIQUE LAB						Core
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
Python Programming	-	-	-	3	25	50	1.5

Course Objectives:

Students will try:

1. Understand the concepts of Artificial Neural Networks and Deep Learning concepts.
2. Implement ANN and DL algorithms with Tensor flow and Keras.
3. Gain knowledge on Sequence learning with RNN.
4. Gain knowledge on Image processing and analysis with CNN
5. Get information on advanced concepts of computer vision.

Course Outcomes:

After completion of this course, the students are able to

1. Develop ANN without using Machine Learning/Deep learning libraries.
2. Understand the Training ANN model with back propagation.
3. Develop model for sequence learning using RNN.
4. Develop image classification model using ANN and CNN.
5. Generate a new image with auto-encoder and GAN.

List of Experiments:

1. Create Tensors and perform basic operations with tensors.
2. Create Tensors and apply split & merge operations and statistics operations.
3. Design single unit perceptron for classification of iris dataset without using predefined models
4. Design, train and test the MLP for tabular data and verify various activation functions and optimizers tensor flow.
5. Design and implement to classify 32x32 images using MLP using tensorflow/keras and check the accuracy.
6. Design and implement a simple RNN model with tensorflow / keras and check accuracy.
7. Design and implement LSTM model with tensorflow / keras and check accuracy.
8. Design and implement GRU model with tensorflow / keras and check accuracy.
9. Design and implement a CNN model to classify multi category JPG images with tensorflow / keras and check accuracy. Predict labels for new images.
10. Design and implement a CNN model to classify multi category tiff images with tensorflow / keras and check the accuracy. Check whether your model is overfit / underfit / perfect fit and apply the techniques to avoid overfit and underfit like regulizers, dropouts etc.
11. Implement a CNN architectures (LeNet, Alexnet, VGG, etc) model to classify multi category Satellite images with tensorflow / keras and check the accuracy. Check whether your model is overfit / underfit / perfect fit and apply the techniques to avoid overfit and underfit.
12. Implement an Auto encoder to de-noise image.
13. Implement a GAN application to convert images.

Suggested Readings:

1. Bishop, C.M., Pattern Recognition and Machine Learning, Springer, 2006.
2. Yegnanarayana, B., Artificial Neural Networks PHI Learning Pvt. Ltd, 2009.
3. Golub, G.H., and Van Loan, C.F., Matrix Computations, JHU Press, 2013.
4. Satish Kumar, Neural Networks: A Classroom Approach, Tata McGraw Hill Education, 2004.

Course Code	Course Title					Core/Elective
U23CS7L1	FULL STACK WEB DEVELOPMENT LAB					Core Course
Prerequisites	Contact Hour per Week			CIE	SEE	Credit
	L	T	P			
HTML/CSS Basic	-	-	3	40	60	1.5

Course Objectives:

1. Understand HTML, CSS, and JavaScript for web development
2. Develop responsive and interactive web applications.
3. Build front-end applications using React.js.
4. Create back-end services using Node.js, Express.js, and MongoDB.
5. Integrate and deploy full-stack MERN applications.

Course Outcomes:

At the end of the course, student would be able to

1. Design responsive web pages using Semantic HTML and CSS.
2. Develop interactive web applications using JavaScript.
3. Build dynamic user interfaces using React.js.
4. Develop RESTful APIs and database-driven applications using the MERN stack.
5. Integrate and deploy full-stack web applications on cloud platforms.

LAB EXPERIMENT

1. HTML

Building a Personal Portfolio Page with HTML using necessary tags, consists of image, form, table for display skills and ratings.

2. CSS

Styling the Portfolio with CSS – Box Model, Flexbox & Project Gallery using Grid. It should be responsive use media query tags.

3. **Java script** basics – explore operators, control flow, arrays, functions, and string manipulation .

4. Bring the portfolio to life: dynamically add/remove project cards, change themes, and respond to user events.

5. Validate the Contact form from Exp 01 using regex patterns; implement try-catch exception handling.

6. Fetch API - Get live data (e.g., GitHub user stats, etc) and render it on the portfolio, introducing async/await.

7. **React.js** - Re-build the portfolio as a React app with functional components, props, state (useState), and useEffect.

8. **Node.js** - Install Node.js, understand the event loop, create a basic HTTP server, and use core modules.

9. **Express.js** - Build a RESTful API that serves portfolio data (projects, messages) and handles Contact-form submissions.

10. **MongoDB & Mongoose** - Connect the Express API to MongoDB Atlas; perform full CRUD on Projects and Contact collections.

11. **Full-Stack Integration** - Integrate React (Exp 07) + Express (Exp 09) + MongoDB (Exp 10) into a single deployable MERN application.

12. **Deployment** - Deploy the full MERN portfolio application to the cloud so it is publicly accessible.

Suggested Readings:

1. Vasan Subramanian, “Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node”, 2nd Edition, Apress.
2. Thomas A Powel “The Complete Reference HTML & CSS”, 5th Edition, McGraw Hill.
3. Ketan Agnihotri, Pranali Dahale, " React Development using Typescript: Modern web app development using advanced React techniques ", English Edition, 2024.
4. Nabendu Biswas, “Ultimate Full-Stack Web Development with MERN: Design, Build, Test and Deploy Production-Grade Applications with MongoDB, Express, React and NodeJS”, 1st Edition, 2023.

Online Resources:

1. https://react.dev/
2. https://nodejs.org/learn/
3. https://expressjs.com/en/
4. https://www.mongodb.com/docs/
5. https://www.w3schools.com/