B. Tech – III Year I Semester

S.No.	Category	Title	L	T	P	C
1	Professional Core	Deep Learning	3	0	0	3
2	Professional Core	Computer Networks	3	0	0	3
3	Professional Core	Operating Systems	3	0	0	3
4	Professional Elective-I	1. Object Oriented Analysis and Design 2. Automata Theory & Compiler Design 3. Soft computing 4. Internet of Things	3	0	0	3
5	Open Elective- I	OR Entrepreneurship Development & Venture Creation	3	0	0	3
6	Professional Core	Deep learning Lab	0	0	3	1.5
7	Professional Core	Computer Networks Lab	0	0	3	1.5
8	Skill Enhancement course	Full Stack development-2 / SWAYAM Plus – Data Engineer / AI Engineer /	0	1	2	2
9	ES	Tinkering Lab (*User Interface Design using Flutter*) / SWAYAM Plus - Android Application Development (with Flutter)	0	0	2	1
10	Evaluation of Community Service Project Internship		-	-	-	2
		Total	**15**	**1**	**10**	**23**
MC	Student may select from the Same Minor Pool		3	0	3	4.5
MC	Minor Course through SWAYAM / NPTEL (Minimum 12 Week, 3 credit course)		3	0	0	3
HC	Student may select from the Same Honors Pool		3	0	0	3
HC	Student may select from the Same Honors Pool		3	0	0	3

B. Tech – III Year II Semester

S.No.	Category	Title	L	T	P	C
1	Professional Core	Software Engineering	3	0	0	3
2	Professional Core	Generative A.I.	3	0	0	3
3	Professional Core	Data Visualization	3	0	0	3
4	Professional Elective-II	1. Software Testing Methodology 2. Cryptography & Network Security 3. Recommender Systems 4. DevOps 5. Any of the 12-Week SWAYAM /NPTEL Course suggested by the BoS	3	0	0	3
5	Professional Elective-III	1. Software Project Management 2. Mobile Adhoc Networks 3. Computer Vision 4. Cloud Computing 5. Any of the 12-Week SWAYAM /NPTEL Course suggested by the BoS	3	0	0	3
6	Open Elective - II		3	0	0	3
7	Professional Core	Generative A.I. Lab	0	0	3	1.5
8	Professional Core	Data Visualization Lab	0	0	3	1.5
9	Skill Enhancement course	Soft skills / SWAYAM Plus - 21st Century Employability Skills	0	1	2	2
10	Audit Course	Technical Paper Writing & IPR	2	0	0	-
Total			**20**	**1**	**8**	**23**
*Mandatory Industry Internship/Mini Project of 08 weeks duration during summer vacation						
MC	Student may select from the same minors pool		3	0	3	4.5
MC	Minor Course (Student may select from the same specialized minors pool)		3	0	0	3
HC	Student may select from the same honors pool		3	0	0	3
HC	Honors Course (Student may select from the honors pool)		3	0	0	3

* Under Industry Internship interested students can pursue SWAYAM Plus courses viz., Hands-on Masterclass on Data Analytics OR Artificial Intelligence for Real-World Application

Open Electives, offered to other department students:

Open Elective I: Operating Systems / Computer Organization and Architecture
Open Elective II: Database Management Systems
Open Elective III: Object Oriented Programming Through Java
Open Elective IV: Computer Networks / Software Engineering / IOT Based Smart Systems

Minor Engineering

Note:

1. *To obtain Minor Engineering, student needs to obtain 18 credits by successfully completing any of the following courses in the concern stream.*
2. *During Minor/Honors Course selection, there should not be any overlapping with Regular/Major/OPEN Electives*

For Minor in AI:	**L-T-P-C**
1. Database Management Systems	3-0-3-4.5 (II-II)
2. Artificial Intelligence	3-0-0-3 (III-I)
3. Advanced Data Structures & Algorithm Analysis	3-0-3-4.5 (III-II)
4. Operating Systems	3-0-0-3 (IV-I)
Any of the following 12 Week 3 credit NPTEL MOOC Courses	
5. Introduction to Machine Learning	
6. Programming in Java	
7. Computer Networks and Internet Protocol	
8. Fundamentals of Object Oriented Programming	
9. Discrete Mathematics for CS	
10. Software Engineering	
11. Deep Learning	

Note: As per AICTE APH 2023-24 student can opt any one of the following streams:

1. Honors in CSE
2. Honors in AI&ML
3. Honors in Data Science

Suggested MOOC Courses for HONORS in CSE

Student need to obtain 18 Credits by successfully completing the following

Mandatory Course(s)

1. Parallel Computer Architecture (MOOCS- SWAYAM / NPTEL 12W)
2. Quantum Algorithms and Cryptography(MOOCS- SWAYAM / NPTEL 12W)

Any of the following for remaining 12 Credits

3. Deep Learning for Computer Vision (MOOCS- SWAYAM / NPTEL 12W)
4. Applied Linear Algebra in AI & ML (MOOCS- SWAYAM / NPTEL 12W)
5. Applied Time-Series Analysis (MOOCS- SWAYAM / NPTEL 12W)
6. Machine Learning for Engineering and Science Applications (MOOCS- SWAYAM / NPTEL 12W)
7. Practical High-Performance Computing(MOOCS- SWAYAM / NPTEL 12W)
8. Deep Learning for Natural Language Processing (MOOCS- SWAYAM / NPTEL 12W)
9. Privacy and Security in Online Social Media (MOOCS- SWAYAM / NPTEL 12W)
10. Natural Language Processing (MOOCS- SWAYAM / NPTEL 12W)

B. Tech III Year – II Semester

III Year I Semester	DEEP LEARNING	L	T	P	C
		3	0	0	3

Course Objectives:

The objective of this course is to cover the fundamentals of neural networks as well as some advanced topics such as recurrent neural networks, long short-term memory cells and convolution neural networks.

Course Outcomes:

After completion of course, students would be able to:

- Explore feed forward networks and Deep Neural networks
- Mathematically understand the deep learning approaches and paradigms
- Apply the deep learning techniques for various applications

UNIT-I:

Basics- Biological Neuron, Idea of computational units, McCulloch–Pitts unit and Thresholding logic, Linear Perceptron, Perceptron Learning Algorithm, Linear separability, Convergence theorem for Perceptron Learning Algorithm.

UNIT-II:

Feed forward Networks- Multilayer Perceptron, Gradient Descent, Back propagation, Empirical Risk Minimization, regularization, auto encoders. Deep Neural Networks: Difficulty of training deep neural networks, Greedylayer wise training.

UNIT-III:

Better Training of Neural Networks –Newer optimization methods for neural networks (Adagrad, adadelta, rmsprop, adam, NAG), second order methods for training, Saddle point problem in neural networks, Regularization methods (dropout, drop connect, batch normalization).

UNITIV:

Recurrent Neural Networks- Back propagation through time, Long Short-Term Memory, Gated Recurrent Units, Bidirectional LSTMs, Bidirectional RNNs.

Convolutional Neural Networks: LeNet, AlexNet. Generative models: Restrictive Boltzmann Machines (RBMs), Introduction to MCMC and Gibbs Sampling, gradient computations in RBMs, Deep Boltzmann Machines.

UNITV:

Recenttrends-VariationalAutoencoders,Transformers,GPT Applications: Vision, NLP, Speech

TextBooks:

1. DeepLearning, Ian Good fellow and Yoshua Bengio and Aaron Courville, MIT Press, 2016

Reference Books:

1. NeuralNetworks:ASystematicIntroduction,RaúlRojas,1996
2. PatternRecognitionandMachineLearning,ChristopherBishop,2007
3. DeepLearningwithPython,FrançoisChollet,ManningPublications,2017

III Year I Semester	COMPUTER NETWORKS	L	T	P	C
		3	0	0	3

Course Objectives:

The course is designed to

- To understand the different types of networks
- To develop an understanding the principles of computer networks.
- To familiarize with Reference model OSI and TCP/IP
- To understand various layers of Reference models functions
- To explore network protocols

Course Outcomes (CO):

After completion of the course, students will be able to

- Understand the reference models and network protocols
- Describe data transmission media and data link layer.
- Understand the network layer design issues and Network layer Protocols.
- Evaluate transport layer services and its protocols.
- Understand application layer protocols and their uses

UNIT I: Introduction

Types of Computer Networks, Reference Models- The OSI Reference Model, The TCP/IP Reference Model, A Critique of the OSI Model and Protocols, A Critique of the TCP/IP Reference Model. History of Internet.

UNITII: The Data Link Layer

Transmission Media, Guided and Un-guided media, Data Link Layer Design Issues, Services Provided to the Network Layer, Error detecting and Error Correcting codes, Elementary Data Link Protocols, Sliding Window Protocols, HDLC, PPP. Multiple Access Protocols Wired Lans: Ethernet, Fast Ethernet, Gigabit Ethernet

UNITIII: The Network Layer

Network LayerDesign Issues, Routing Algorithms, Congestion, Congestion control algorithms. The Network Layer in the Internet, The IP Version 4 Protocol, IP Addresses- Classful, CIDR, NAT, IP Version 6 Protocol, Transition from IPV4 to IPV6

UNITIV: The Transport Layer

The Transport Layer Services, Transport Layer Protocols: UDP, TCP and SCTP

UNITV: The Application Layer

The World Wide Web, HTTP, Domain Name Space, Remote Loging, Electronic Mail and File Transfer

Textbooks:

1. "Computer Networks", Andrew S Tanenbaum, David J Wetherall, 5th Edition, Pearson
2. "Data Communications and Networking", Behrouz A Forouzan, 4th Edition, Tata McGraw Hill Education

Reference Books:

1. "Data and Computer Communication", William Stallings, Pearson
2. "TCP/IP Protocol Suite", Behrouz Forouzan, McGraw Hill.

III Year I Semester	OPERATING SYSTEMS	L	T	P	C
		3	0	0	3

Course Objectives:

The main objectives of the course is to make student

- Understand the basic concepts and principles of operating systems, including process management, memory management, file systems, and Protection
- Make use of process scheduling algorithms and synchronization techniques to achieve better performance of a computer system.
- Illustrate different conditions for dead lock and their possible solutions.

UNIT-I:

Operating Systems Overview: Introduction, Operating system functions, Operating systems operations, Computing environments, Free and Open-Source Operating Systems System Structures: Operating System Services, User and Operating-System Interface, system calls, Types of System Calls, system programs, Operating system Design and Implementation, Operating system structure, Building and Booting an Operating System, Operating system debugging

UNIT-II:

Processes: Process Concept, Process scheduling, Operations on processes, Inter-process communication. Threads and Concurrency: Multi threading models, Thread libraries, Threading issues. CPU Scheduling: Basic concepts, Scheduling criteria, Scheduling algorithms, Multiple processor scheduling.

UNIT–III:

Synchronization Tools: The Critical Section Problem, Peterson's Solution, Mutex Locks, semaphores, Monitors, Classic problems of Synchronization. Deadlocks: system Model, Dead lock characterization, Methods for handling Dead locks,Deadlock prevention, Deadlock avoidance, Deadlock detection, Recovery from Deadlock.

UNIT-IV:

Memory-Management Strategies: Introduction, Contiguous memory allocation, Paging,Structureof the Page Table, Swapping.VirtualMemoryManagement:Introduction,Demandpaging, Copy-on-write,Pagereplacement Allocation of frames, Thrashing StorageManagement:OverviewofMassStorageStructure,HDDScheduling.

UNIT-V:

FileSystem:FileSystemInterface:Fileconcept,Accessmethods,DirectoryStructure;File system Implementation: File-system structure, File-system Operations, Directory implementation, Allocation method, Free space management; File-System Internals:File-SystemMounting, Partitions

and Mounting, File Sharing.Protection:Goalso ofprotection,Principlesofprotection,ProtectionRings,Domainof protection, Access matrix

TextBooks:

1. OperatingSystem Concepts,SilberschatzA,GalvinPB,GagneG,10th Edition,Wiley, 2018.
2. ModernOperatingSystems,TanenbaumAS,4th Edition,Pearson ,2016.

Reference Books:

1. Operating Systems -Internals and Design Principles,Stallings W, 9th edition, Pearson, 2018
2. OperatingSystems:A ConceptBasedApproach,D.MDhamdhere,3rd Edition, McGraw-Hill, 2013

OnlineLearningResources:

1. https://nptel.ac.in/courses/106/106/106106144/
2. http://peterindia.net/OperatingSystems.html

III Year I Semester	OBJECT ORIENTED ANALYSIS AND DESIGN (PROFESSIONAL ELECTIVE-I)	L	T	P	C
		3	0	0	3

Course Objectives: The main objective is the students to

- Become familiar withal phases of OOAD.
- Master the main features of the UML.
- Master the main concepts of Object Technologies and how to apply them at work and develop the ability to analyze and solve challenging problem in various domains.
- Learn the Object design Principles and understand how to apply them towards Implementation.

UNIT-I:

Introduction: The Structure of Complex systems, The Inherent Complexity of Software, Attributes ofComplexSystem,OrganizedandDisorganizedComplexity,BringingOrdertoChaos,DesigningComplexSystems.**Case Study:**SystemArchitecture:Satellite-BasedNavigation

UNIT-II:

IntroductiontoUML: Importanceofmodeling,principlesofmodeling,object-orientedmodeling,conceptual model of the UML, Architecture, and Software Development Life Cycle. **Basic StructuralModeling:** Classes, Relationships, common Mechanisms, and diagrams. **Case Study:** Control System:Traffic Management.

UNIT-III:

Class&ObjectDiagrams: Terms,concepts,modelingtechniquesforClass&ObjectDiagrams.**Advanced Structural Modeling:** Advanced classes, advanced relationships, Interfaces, Types and Roles,Packages.**CaseStudy:** AI:Cryptanalysis.

UNITIV:

Basic BehavioralModeling-I: Interactions, Interaction diagrams Use cases, Use case Diagrams, ActivityDiagrams.**Case Study:**WebApplication:VacationTrackingSystem

UNITV:

Advanced BehavioralModeling: Events and signals, state machines, processes and Threads, time andspace, state chart diagrams.**Architectural Modeling:** Component, Deployment, Component diagramsandDeploymentdiagrams**CaseStudy:**WeatherForecasting.

TextBooks:

1. GradyBOOCH,RobertA.Maksimchuk,MichaelW.ENGLE,BobbiJ.Young,JimConallen,

KelliaHouston, "Object- Oriented Analysis and Design with Applications", 3rd edition,2013,PEARSON.
2. GradyBooch,JamesRumbaugh,IvarJacobson:TheUnifiedModelingLanguageUserGuide,PearsonEducation.

ReferenceBooks:

1. MeilirPage-Jones:FundamentalsofObject-Oriented DesigninUML,PearsonEducation.
2. PascalRoques:ModelingSoftwareSystemsUsingUML2,WILEY- DreamtechIndiaPvt.Ltd.
3. AtulKahate:ObjectOrientedAnalysis &Design, TheMcGraw-HillCompanies.

ApplingUMLandPatterns:AnintroductiontoObject–OrientedAnalysisandDesignandUnifiedProcess,CraigLarman,PearsonEducation.

III Year I Semester	Automata Theory and Compiler Design (Professional Elective-I)	L	T	P	C
		3	0	0	3

COURSE OUTCOMES: After completion of this course

- Understand and apply formal language theory.
- Design and implement parsers.
- Understand the phases of a compiler.
- Apply semantic analysis and error handling.
- Optimize intermediate and target code.

UNIT – I:
Introduction to Finite Automata: Structural Representations, Automata and Complexity, the Central Concepts of Automata Theory – Alphabets, Strings, Languages, Problems. Nondeterministic Finite Automata: Formal Definition, an application, Text Search, Finite Automata with Epsilon-Transitions. Deterministic Finite Automata: Definition of DFA, How A DFA Process Strings, The language of DFA, Conversion of NFA with €-transitions to NFA without €-transitions. Conversion of NFA to DFA

UNIT – II:
Regular Expressions: Finite Automata and Regular Expressions, Applications of Regular Expressions, Algebraic Laws for Regular Expressions, Conversion of Finite Automata to Regular Expressions. Pumping Lemma for Regular Languages: Statement of the pumping lemma, Applications of the Pumping Lemma. Context-Free Grammars: Definition of Context-Free Grammars, Derivations Using a Grammar, Leftmost and Rightmost Derivations, the Language of a Grammar, Parse Trees, Ambiguity in Grammars and Languages.

UNIT – III:
Push Down Automata: Definition of the Pushdown Automaton, the Languages of a PDA, Equivalence of PDA's and CFG's, Acceptance by final state Turing Machines: Introduction to Turing Machine, Formal Description, Instantaneous description, The language of a Turing machine Undecidability: Undecidability, A Language that is Not Recursively Enumerable, An Undecidable Problem That is RE, Undecidable Problems about Turing Machines

UNIT - IV Introduction: The structure of a compiler, Lexical Analysis: The Role of the Lexical Analyzer, Input Buffering, Recognition of Tokens, The Lexical- Analyzer Generator Lex, Syntax Analysis: Introduction, Context-Free Grammars, Writing a Grammar, Top-Down Parsing, Bottom-Up Parsing, Introduction to LR Parsing: Simple LR, More Powerful LR Parsers R18 B.Tech. CS&D Syllabus JNTU Hyderabad

UNIT - V Syntax-Directed Translation: Syntax-Directed Definitions, Evaluation Orders for SDD's, SyntaxDirected Translation Schemes, Implementing L-Attributed SDD's. Intermediate-Code Generation: Variants of Syntax Trees, Three-Address Code Run-Time Environments: Stack Allocation of Space, Access to Nonlocal Data on the Stack, Heap Management

Text Books:

1. Introduction to Automata Theory, Languages, and Computation, 3nd Edition, John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, Pearson Education.
2. Compilers: Principles, Techniques and Tools, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffry D. Ullman, 2nd Edition, Pearson.
3. Theory of Computer Science – Automata languages and computation, Mishra and Chandrashekaran, 2nd Edition, PHI.

Reference Books:

1. Introduction to Formal languages Automata Theory and Computation, Kamala Krithivasan, Rama R, Pearson.
2. Introduction to Languages and The Theory of Computation, John C Martin, TMH.
3. lex &yacc – John R. Levine, Tony Mason, Doug Brown, O'reilly
4. Compiler Construction, Kenneth C. Louden, Thomson. Course Technology.

III Year I Semester	Soft Computing (Professional Elective-I)	L	T	P	C
		3	0	0	3

Course Objectives:

To introduce the concepts in Soft Computing such as Artificial Neural Networks, Fuzzylogic-based systems, genetic algorithm-based systems and their hybrids.

Course Outcomes:

The students will be able to

- Learn soft computing techniques and their applications.
- Analyze various neural network architectures.
- Define the fuzzy systems.
- Understand the genetic algorithm concepts and their applications.
- Identify and select a suitable Soft Computing technology to solve the problem; construct a solution and implement a Soft Computing solution

UNIT-I:

Introduction to Soft Computing, Artificial neural networks, biological neurons, Basic models of artificial neural networks, Connections, Learning, Activation Functions, McCulloch and Pitts Neuron, Hebb network.

UNIT-II:

Perceptron networks, learning rule, Training and testing algorithm, Adaptive Linear Neuron, Back propagation Network, Architecture, Training algorithm

UNIT-III:

Fuzzy logic, fuzzy sets, properties, operations on fuzzy sets, fuzzy relations, operations on fuzzy relations, Fuzzy membership functions, fuzzification, Methods of membership, value assignments, intuition, inference, rank ordering, Lambda –
Cuts for fuzzy sets, Defuzzification methods

UNIT-IV:

Truth values and Tables in Fuzzy Logic, Fuzzy propositions, Formation of fuzzy rules, Decomposition of rules, Aggregation of rules, Fuzzy Inference Systems, Mamdani and Sugeno types, Neuro-fuzzy hybrid systems, characteristics, classification

UNIT-V:

Introduction to genetic algorithm, operators in genetic algorithm, coding,

selection,crossover,mutation,stoppingconditionforgeneticalgorithmflow, Genetic-neuro hybrid systems, GeneticFuzzy rule-based system

Text Books:

1. S.N.SivanandamandS.N.Deepa,Principlesofsoftcomputing–JohnWiley & Sons,2007.
2. Timothy J.Ross,FuzzyLogicwithengineeringapplications, John Wiley &Sons, 2016.

ReferenceBooks:

1. N.K.SinhaandM.M.Gupta,SoftComputing&IntelligentSystems:Theory & Applications-Academic Press /Elsevier. 2009.
2. SimonHaykin,NeuralNetwork-AComprehensiveFoundation-Prentice Hall International, Inc.1998
3. R. Eberhart and Y. Shi, ComputationalIntelligence:
4. Concepts to Implementation, Morgan Kaufman/Elsevier, 2007.
5. DriankovD.,HellendoornH.andReinfrankM.AnIntroductiontoFuzzy ControlNarosa Pub., 2001.
6. BartKosko,NeuralNetworkandFuzzySystems-PrenticeHall,Inc.,Englewood Cliffs, 1992
7. GoldbergD.E.,GeneticAlgorithmsinSearch,Optimization,andMachine LearningAddison Wesley, 1989

III Year I Semester	Internet of Things (Professional Elective-I)	L	T	P	C
		3	0	0	3

Course Objectives:

- Vision and Introduction to Internet of Things (IoT).
- Understand IoT Market perspective.
- Data and Knowledge Management and use of Devices in IoT Technology.
- Understand State of the Art – IoT Architecture.
- Understand Real World IoT Design Constraints, Industrial Automation and Commercial.

Course Outcomes (COs): At the end of the course, student will be able to

- Explain in a concise manner how the general Internet as well as Internet of Things work.
- Understand constraints and opportunities of wireless and mobile networks for Internet of Things.
- Use basic sensing and measurement and tools to determine the real-time performance of network of devices.
- Develop prototype models for various applications using IoT technology.

UNIT-I:

The Internet of Things: An Overview of Internet of things, Internet of Things Technology, behind IoTs Sources of the IoTs, M2M Communication, Examples of IoTs, Design Principles For Connected Devices Internet Connectivity Principles, Internet connectivity, Application Layer Protocols: HTTP, HTTPS, FTP, Telnet.

UNIT-II:

Business Models for Business Processes in the Internet of Things, IoT/M2M systems LAYERS AND designs standardizations, Modified OSI Stack for the IoT/M2M Systems, ETSI M2M domains and High- level capabilities, Communication Technologies, Data Enrichment and Consolidation and Device Management Gateway Ease of designing and affordability

UNIT-III:

Design Principles for the Web Connectivity for connected-Devices, Web Communication protocols for Connected Devices, Message Communication protocols for Connected Devices, Web Connectivity for connected-Devices.

UNIT-IV:

Data Acquiring, Organizing and Analytics in IoT/M2M, Applications /Services /Business Processes, IOT/M2M Data Acquiring and Storage, Business Models for Business Processes in the Internet of Things, Organizing Data, Transactions, Business Processes, Integration and Enterprise Systems.

UNIT-V:

Data Collection, Storage and Computing Using a Cloud Platform for IoT/M2M Applications/Services, Data Collection, Storage and Computing Using cloud platform Everything as a service and Cloud Service Models, IOT cloud-based services using the Xively (Pachube/COSM), Nimbits and other platforms Sensor, Participatory Sensing, Actuator, Radio Frequency Identification, and Wireless, Sensor Network Technology, Sensors Technology, Sensing the World.

Text Books:

1. Internet of Things: Architecture, Design Principles and Applications, Rajkamal, McGraw Hill Higher Education
2. Internet of Things, A.Bahgya and V.Madisetti, University Press, 2015

Reference Books:

1. Designing the Internet of Things, Adrian McEwen and Hakim Cassimally,WileyGetting Started with the Internet of Things, Cuno Pfister, Oreilly

III Year I Semester	Deep Learning Lab	L	T	P	C
		0	0	3	1.5

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

- Implement deep neural networks to solve real world problems
- Choose appropriate pre-trained model to solve real time problem
- Interpret the result soft different deep learning models

Software Packages required:

- Keras
- Tensorflow
- PyTorch

List of Experiments:

1. Implement multi-layer perceptron algorithm for MNISTH and written Digit Classification.
2. Design neural network for classifying movie reviews(BinaryClassification) using IMDB dataset.
3. Design a neural Network for classifying news wires (Multiclassclassification) using Reuters dataset.
4. Design neural network for predicting house prices using Boston Housing Price dataset.
5. Build a Convolution Neural Network for MNISTH and written Digit Classification.
6. Build a Convolution Neural Network for simple image (dogs and Cats) Classification
7. Use a pre-trained convolution neural network (VGG16)for image classification.
8. Implement one hot encoding of words or characters.
9. Implement word embeddings for IMDB dataset.
10. Implement a Recurrent Neural Network for IMDB movie review classification problem.

Text Books:

1. RezaZadehandBharathRamsundar,“TensorflowforDeepLearning”, O’Reilly publishers, 2018

References:

1. https://github.com/fchollet/deep-learning-with-python-notebooks

III Year I Semester	Computer Networks Lab	L	T	P	C
		0	0	3	1.5

List of Experiments:

1. Study of Network devices in detail and connect the computers in Local Area Network.
2. Write a Program to implement the data link layer farming methods such as
   i) Character stuffing ii) bit stuffing.
3. Write a Program to implement data link layer farming method checksum.
4. Write a program for Hamming Code generation for error detection and correction.
5. Write a Program to implement on a data set of characters the three CRC polynomials – CRC 12, CRC 16 and CRC CCIP.
6. Write a Program to implement Sliding window protocol for Goback N.
7. Write a Program to implement Sliding window protocol for Selective repeat.
8. Write a Program to implement Stop and Wait Protocol.
9. Write a program for congestion control using leaky bucket algorithm
10. Write a Program to implement Dijkstra‘s algorithm to compute the Shortest path through a graph.
11. Write a Program to implement Distance vector routing algorithm by obtaining routing table at each node (Take an example subnet graph with weights indicating delay between nodes).
12. Write a Program to implement Broadcast tree by taking subnet of hosts.
13. Wireshark
    i. Packet Capture Using Wire shark
    ii. Starting Wire shark
    iii. Viewing Captured Traffic
    iv. Analysis and Statistics & Filters.
14. How to run Nmap scan
15. Operating System Detection using Nmap
16. Do the following using NS2 Simulator
    i. NS2 Simulator-Introduction
    ii. Simulate to Find the Number of Packets Dropped
    iii. Simulate to Find the Number of Packets Dropped by TCP/UDP
    iv. Simulate to Find the Number of Packets Dropped due to Congestion
    v. Simulate to Compare Data Rate& Throughput.

III Year I Semester	Full Stack Frontend Development – Module – II (Skill Oriented Course)	L	T	P	C
		0	1	2	2

Course Objectives:

The main objectives of the course are to

- Make use of router, template engine and authentication using sessions to develop application in ExpressJS.
- Build a single page application using RESTful APIs in ExpressJS
- Apply router and hooksin designing ReactJS application

- Make use of MongoDB queries to perform CRUD operations on document database

Experiments covering the Topics:

- ExpressJS – Routing, HTTP Methods, Middleware, Templating, Form Data
- ExpressJS – Cookies, Sessions, Authentication, Database, RESTful APIs
- ReactJS – Render HTML, JSX, Components – function & Class, Props and States, Styles, Respond to Events
- ReactJS – Conditional Rendering, Rendering Lists, React Forms, React Router, Updating the Screen
- ReactJS – Hooks, Sharing data between Components, Applications – To-do list and Quiz
- MongoDB – Installation, Configuration, CRUD operations, Databases, Collections and Records

Sample Experiments:

1. ExpressJS – Routing, HTTP Methods, Middleware.

a. Write a program to define a route, Handling Routes, Route Parameters, Query Parameters and URL building.

b. Write a program to accept data, retrieve data and delete a specified resource using http methods.

c. Write a program to show the working of middleware.

2. ExpressJS – Templating, Form Data

a. Write a program using templating engine.

b. Write a program to work with form data.

3. ExpressJS – Cookies, Sessions, Authentication

a. Write a program for session management using cookies and sessions.

b. Write a program for user authentication.

4. ExpressJS – Database, RESTful APIs

a. Write a program to connect MongoDB database using Mangoose and perform CRUD operations.

b. Write a program to develop a single page application using RESTful APIs.

5. ReactJS – Render HTML, JSX, Components – function & Class

a. Write a program to render HTML to a web page.

b. Write a program for writing markup with JSX.

c. Write a program for creating and nesting components (function and class).

d.

6. ReactJS – Props and States, Styles, Respond to Events

a. Write a program to work with props and states.
b. Write a program to add styles (CSS & Sass Styling) and display data.
c. Write a program for responding to events.

7. ReactJS – Conditional Rendering, Rendering Lists, React Forms

a. Write a program for conditional rendering.
b. Write a program for rendering lists.
c. Write a program for working with different form fields using react forms.

8. ReactJS – React Router, Updating the Screen

a. Write a program for routing to different pages using react router.
b. Write a program for updating the screen.

9. ReactJS – Hooks, Sharing data between Components

a. Write a program to understand the importance of using hooks.
b. Write a program for sharing data between components.

10. MongoDB – Installation, Configuration, CRUD operations

a. Install MongoDB and configure ATLAS
b. Write MongoDB queries to perform CRUD operations on document using insert(), find(), update(), remove()

11. MongoDB – Databases, Collections and Records

a. Write MongoDB queries to Create and drop databases and collections.
b. Write MongoDB queries to work with records using find(), limit(), sort(), createIndex(), aggregate().

12. Augmented Programs: (Any 2 must be completed)

a. Design a to-do list application using NodeJS and ExpressJS.
b. Design a Quiz app using ReactJS.
c. Complete the MongoDB certification from MongoDB University website.

Text Books:

1. Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node, Vasan Subramanian, 2nd edition, APress, O'Reilly.
2. Node.Js in Action, Mike Cantelon, Mark Harter, T.J. Holowaychuk, Nathan Rajlich,Manning Publications. (Chapters 1-11)
3. React Quickly, AzatMardan,Manning Publications(Chapters 1-8,12-14)

Web Links:

1. ExpressJS - https://www.tutorialspoint.com/expressjs
2. ReactJS - https://www.w3schools.com/REACT (and) https://react.dev/learn#
3. MongoDB -https://learn.mongodb.com/learning-paths/introduction-to-mongodb

III Year I Semester	Tinkering Lab (UI Design Using Flutter)	L	T	P	C
		0	0	2	1

Course Objectives:

- Learns to Implement Flutter Widgets and Layouts
- Understands Responsive UI Design and with Navigation in Flutter
- Knowledge on Wedges and customize widgets for specific UI elements, Themes
- Understand to include animation apart from fetching data

List of Experiments:

Students need to implement the following experiments

17. a) Install Flutter and Dart SDK.
    b) Write a simple Dart program to understand the language basics.

18. a) Explore various Flutter widgets (Text, Image, Container, etc.).
    b) Implement different layout structures using Row, Column, and Stack widgets.

19. a) Design a responsive UI that adapts to different screen sizes.
    b) Implement media queries and breakpoints for responsiveness.

20. a) Set up navigation between different screens using Navigator.
    b) Implement navigation with named routes.

21. a) Learn about stateful and stateless widgets.
    b) Implement state management using set State and Provider.

22. a) Create custom widgets for specific UI elements.
    b) Apply styling using themes and custom styles.

23. a) Design a form with various input fields.
    b) Implement form validation and error handling.

24. a) Add animations to UI elements using Flutter's animation framework.
    b) Experiment with different types of animations (fade, slide, etc.).

25. a) Fetch data from a REST API.
    b) Display the fetched data in a meaningful way in the UI.

26. a) Write unit tests for UI components.
    b) Use Flutter's debugging tools to identify and fix issues..

Text Book:

1. Marco L. Napoli, Beginning Flutter: A Hands-on Guide to App Development.
2. Rap Payne, Beginning App Development with Flutter: Create Cross-Platform Mobile Apps 1st Edition, Apres

B. Tech III Year - II Semester

III Year II Semester	SOFTWARE ENGINEERING	L	T	P	C
		3	0	0	3

Course Objectives:

The objectives of this course are to introduce

- Software life cycle models, Software requirements and SRS document.
- Project Planning, quality control and ensuring good quality software.
- Software Testing strategies, use of CASE tools, Implementation issues, validation &verification procedures.

UNIT-I:

Introduction: Evolution, Software development projects, Exploratory style of software developments, Emergence of software engineering, Notable changes in software development practices, Computer system engineering.

Software Life Cycle Models: Basic concepts, Waterfall model and its extensions, Rapid application development, Agile development model, Spiral model.

UNIT-II:

Software Project Management: Software project management complexities, Responsibilities of a software project manager, Metrics for project size estimation, Project estimation techniques, Empirical Estimation techniques, COCOMO, Halstead's software science, risk management.

Requirements Analysis and Specification: Requirements gathering and analysis, Software Requirements Specification (SRS), Formal system specification, Axiomatic specification, Algebraic specification, Executable specification and 4GL.

UNIT-III:

Software Design: Overview of the design process, how to characterize a good software design? Layered arrangement of modules, Cohesion and Coupling. approaches to software design.

Agility: Agility and the Cost of Change, Agile Process, Extreme Programming (XP), Other Agile Process Models, Tool Set for the Agile Process (Text Book 2)

Function-Oriented Software Design: Overview of SA/SD methodology, Structured analysis, Developing the DFD model of a system, Structured design, Detailed design, and Design Review.

User Interface Design: Characteristics of a good user interface, Basic concepts, Types of user interfaces, Fundamentals of component-based GUI development, and user interface design methodology.

UNIT-IV:

Coding And Testing: Coding, Code review, Software documentation, Testing, Black-box testing,

White-Box testing, Debugging, Program analysis tools, Integration testing, testing object-oriented programs, Smoke testing, and some general issues associated with testing.

Software Reliability and Quality Management: Software reliability. Statistical testing, Software quality, Software quality management system,ISO9000. SEI Capability maturity model. Few other important quality standards, and Six Sigma.

UNITV:

Computer-Aided Software Engineering (Case): CASE and its scope, CASE environment, CASE support in the software life cycle, other characteristics of CASE tools, Towards second generation CASE Tool, and Architecture of a CASE Environment.

Software Maintenance: Characteristics of software maintenance, Softwarereverse engineering, Software maintenance process models and Estimation of maintenance cost.

SoftwareReuse: Reuse-definition,introduction,reasonbehindnoreusesofar,Basicissuesin any reuse program, A reuse approach, and Reuse at organization level.

Text Books:

1. FundamentalsofSoftwareEngineering,RajibMall,5th Edition,PHI.
2. Software Engineering: A Practitioner's Approach, Roger S. Pressman, 9th Edition, McGraw Hill International Edition.

Reference Books:

1. SoftwareEngineering,IanSommerville,10th Edition,Pearson.
2. Software Engineering, Principles and Practices,DeepakJain,OxfordUniversityPress.

e- Resources:

1) https://nptel.ac.in/courses/106/105/106105182/
2) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01260589506387148827_shared/overview
3) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01338269041100390473 5_shared/overview

III Year II Semester	GENERATIVE AI	L	T	P	C
		3	0	0	3

Course Objectives:
• Understand the basics of Generative AI.
• Know the basics of Text Generation.
• Understand the process of generating videos.
• Know about GAN and its variants.

UNIT-I:
Introduction To Gen Ai: Historical Overview of Generative modelling, Difference between Gen AI and Discriminative Modeling, Importance of generative models in AI and Machine Learning, Types of Generative models, GANs, VAEs, autoregressive models and Vector quantized Diffusion models, understanding if probabilistic modeling and generative process, Challenges of Generative Modeling, Future of Gen AI, Ethical Aspects of AI, Responsible AI, Use Cases.

UNIT-II:
Generative Models for Text: Language Models Basics, building blocks of Language models, Transformer Architecture, Encoder and Decoder, Attention mechanisms, Generation of Text, Models like BERT and GPT models, Generation of Text, Autoencoding, Regression Models, Exploring ChatGPT, Prompt Engineering: Designing Prompts, Revising Prompts using Reinforcement Learning from Human Feedback (RLHF), Retrieval Augmented Generation, Multimodal LLM, Issues of LLM like hallucination.

UNIT-III:
Generation of Images: Introduction to Generative Adversarial Networks, Adversarial Training Process, Nash Equilibrium, VariationalAutoencoders, Encoder-Decoder Architectures, Stable Diffusion Models, Introduction to Transformer-based Image Generation, CLIP, Visual Transformers ViT- Dall-E2 and Dall-E3, GPT-4V, Issues of Image Generation models like Mode Collapse and Stability.

UNIT-IV:
Generation of Painting, Music, and Play: Variants of GAN, Types of GAN, Cyclic GAN, Using Cyclic GAN to Generate Paintings, Neural Style Transfer, Style Transfer, Music Generating RNN, MuseGAN, Autonomous agents, Deep Q Algorithm, Actor-critic Network.

UNIT-V:
Open-Source Models and Programming Frameworks: Training and Fine tuning of Generative models, GPT4All, Transfer learning and Pretrained models, Training vision models, Google Copilot, Programming LLM,LangChain, Open-Source Models, Llama, Programming for TimeSformer, Deployment, Hugging Face.

Text Books:

1. Denis Rothman, "Transformers for Natural Language Processing and Computer Vision", Third Edition,Packt Books, 2024

Reference Books:

1. David Foster," Generative Deep Learning", O'Reily Books, 2024.
2. Altaf Rehmani, "Generative AI for Everyone", BlueRose One, 2024.

III Year II Semester	Data and Visual Analytics	L	T	P	C
		3	0	0	3

Pre-Requisites: Computer Graphics, Image Processing

Course Objective:

- familiarize students with the basic and advanced techniques of information visualizationand scientific visualization
- learn key techniques of the visualization process
- a detailed view of visual perception, the visualized data and the actual visualization, interaction and distorting techniques

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

CO	Course Outcomes	Knowledge Level (K)#
CO1	**Explain** Visualization and representation of data	K6
CO2	**Creating** visual representations and visualization reference model of applications	K3
CO3	**Classify**the visualization systems in a data representation	K4
CO4	**Identify** Visualization of groups and trees	K3
CO5	**Determine** the visualization of volumetric different data sets in applications	K6

#Based on suggested Revised BTL

SYLLABUS:

UNIT-I:

Introduction: What Is Visualization? History of Visualization, Relationship between Visualizationand Other Fields. The Visualization Process, Introduction of visual perception, visual representation of data, Gestalt principles, information overloads.

UNIT-II:

Creating visual representations, visualization reference model, visual mapping, visual analytics, Design of visualization applications

UNIT-III:

Classification of visualization systems, Interaction and visualization techniques misleading, Visualization of one, two and multi-dimensional data, text and text documents.

UNIT-IV:
Visualization of groups, trees, graphs, clusters, networks, software, Metaphorical visualization

UNIT-V:
Visualization of volumetric data, vector fields, processes and simulations, Visualization of maps, geographic information, GIS systems, collaborative visualizations, Evaluating visualizations
Recent trends in various perception techniques, various visualization techniques, data structures used in data visualization.

Textbook:
1. WARD, GRINSTEIN, KEIM.Interactive Data Visualization: Foundations, Techniques, and Applications. Natick: A K Peters, Ltd.
2. E. Tufte, The Visual Display of Quantitative Information, Graphics Press.

Resources:
1. https://kdd.cs.ksu.edu/Courses/CIS536/Lectures/Slides/Lecture-34-Main_6up.pdf

III Year II Semester	Software Testing Methodology (Professional Elective -II)	L	T	P	C
		3	0	0	3

Course Objectives

- To provide knowledge of the concepts in software testing such as testing process, criteria, strategies, and methodologies.
- To develop skills in software test automation and management using the latest tools.

UNIT – I:

Introduction: Purpose of testing, Dichotomies, model for testing, consequences of bugs, taxonomy of bugs Flow graphs and Path testing: Basics concepts of path testing, predicates, path predicates and achievable paths, path sensitizing, path instrumentation, application of path testing.

UNIT – II:

Transaction Flow Testing: transaction flows, transaction flow testing techniques.
Data Flow testing: Basics of data flow testing, strategies in data flow testing, application of data flow testing.Domain Testing: domains and paths, Nice & ugly domains, domain testing, domains and interfaces testing, domain and interface testing, domains and testability.

UNIT – III:

Paths, Path products and Regular expressions: path products & path expression, reduction procedure, applications, regular expressions & flow anomaly detection.
Logic Based Testing: overview, decision tables, path expressions, kv charts, specifications.

UNIT – IV:

State, State Graphs and Transition testing: state graphs, good & bad state graphs, state testing, Testability tips.

UNIT – V:

Graph Matrices and Application: Motivational overview, matrix of graph, relations, power of a matrix, node reduction algorithm, building tools. (Student should be given an exposure to a tool like Jmeter/selenium/soapUI/Catalon).

Text Books:

1. Software Testing techniques - BarisBeizer, Dreamtech, second edition.
2. Software Testing Tools – Dr. K. V. K. K. Prasad, Dreamtech.

Reference Books:

1. The craft of software testing - Brian Marick, Pearson Education.
2. Software Testing Techniques – SPD(Oreille)
3. Software Testing in the Real World – Edward Kit, Pearson.

4. Effective methods of Software Testing, Perry, John Wiley.
5. Art of Software Testing – Meyers, John Wiley.

III Year II Semester	Cryptography & Network Security (Professional Elective -II)	L	T	P	C
		3	0	0	3

Course Objectives:

- Explain the objectives of information security
- Explain the importance and application of each of confidentiality, integrity, authentication and availability
- Understand the basic categories of threats to computers and networks
- Discusses the Mathematics of Cryptography
- Discuss the fundamental ideas of Symmetric and Asymmetric cryptographic Algorithms
- Discusses the Network layer, Transport Layer and Application layer Protocols Enhanced security mechanisms

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

CO	Course Outcomes	Knowledge Level (K)#
CO1	Student will be able to understand security issues related to computer networks and learn different symmetric key techniques	K2
CO2	Students will be able learn mathematic of cryptography for symmetric and Asymmetric algorithms and apply this knowledge to understand the Cryptographic algorithms	K3
CO3	Students will be able learn different types of symmetric and Asymmetric algorithms	K3
CO4	Students will be able learn different algorithms of Hash functions, message authentication and digital signature and their importance to the security	K4
CO5	Students will be able learn different Enhanced security protocols of Application Layer, Transport Layer and Network layer	K4

#Based on suggested Revised BTL

SYLLABUS:

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.Classical Encryption Techniques-symmetric cipher model, Substitution techniques, Transposition techniques, Rotor Machines, Steganography.

UNIT – II:

Introduction to Symmetric Cryptography: Algebraic Structures-Groups, Rings, Fields, GF(2^n) fields, Polynomials.**Mathematics of Asymmetric cryptography:** Primes, Checking for Primness, Eulers phi-functions, Fermat's Little Theorem, Euler's Theorem, Generating Primes, Primality Testing, Factorization, Chinese Remainder Theorem, Quadratic Congruence, Exponentiation and Logarithm.

UNIT – III:

Symmetric key Ciphers: Block Cipher principles, DES, AES, Blowfish, IDEA, Block cipher operation, Stream ciphers: RC4, RC5

Asymmetric key Ciphers: Principles of public key cryptosystems, RSA algorithm, Diffie-Hellman Key Exchange, Elgamal Cryptographic system, Elliptic Curve Arithmetic, Elliptic Curve Cryptography.

UNIT – IV:

Cryptographic Hash Functions: Applications of Cryptographic Hash Functions, Two Simple Hash Functions, Requirements and Security, Hash Functions Based on Cipher Block Chaining, Secure Hash Algorithms (SHA)

Message Authentication Codes: Message Authentication Requirements, Message Authentication Functions, Requirements for Message Authentication Codes, Security of MAC'S, MAC'S Based on Hash Functions: HMAC, MAC'S Based on Block Ciphers: DAA And CMAC

Digital Signatures: Digital Signatures, Elgamal Digital Signature Scheme, Elliptic Curve Digital Signature Algorithm, RSA-PSS Digital Signature Algorithm.

UNIT – V:

Network and Internet Security: Transport-Level Security: Web Security Considerations, Transport Level Security, HTTPS, SSH.

IP Security: IP Security Overview, IP Security Policy, Encapsulating Security Payload, Authentication Header Protocol.

Electronic-Mail Security: Internet-mail Security, Email Format, Email Threats and Comprehensive Email Security, S/MIME, PGP.

Text Books:

1. Cryptography and Network Security - Principles and Practice: William Stallings, Pearson Education, 7th Edition, 2017
2. Cryptography and Network Security: Behrouz A. Forouzan Debdeep, Mc Graw Hill, 3rd Edition, 2015

Reference Books:

1. Cryptography and Network Security: Atul Kahate, Mc Graw Hill, 3rd Edition
2. Introduction to Cryptography with Coding Theory: Wade Trappe, Lawrence C. Washington, Pearson.

3. Modern Cryptography: Theory and Practice byWenbo Mao. Pearson

III Year II Semester	Recommender Systems (Professional Elective -II)	L	T	P	C
		3	0	0	3

Course Objectives:
This course covers the basic concepts of recommender systems, including personalization algorithms, evaluation tools, and user experiences

Course Outcomes:
- Describe basic concepts behind recommender systems
- Explain a variety of approaches for building recommender systems
- Describe system evaluation methods from both algorithmic and users' perspectives
- Describe applications of recommender systems in various domains

UNIT-I:
Introduction: Recommender system functions, Linear Algebra notation: Matrix addition, Multiplication, transposition, and inverses, covariance matrices, Understanding ratings, Applications of recommendation systems, Issues with recommender system.

UNIT-II:
Collaborative Filtering: User-based nearest neighbor recommendation, Item-based nearest neighbor recommendation, Model based and pre-processing based approaches, Attacks on collaborative recommender systems.

UNIT-III:
Content-based recommendation: High level architecture of content-based systems, Advantages and drawbacks of content-basedfiltering, Item profiles, discovering features of documents, obtaining item features from tags, representing item profiles, Methods for learning user profiles, Similarity based retrieval, Classification algorithms.
Knowledge based recommendation: Knowledge representation and reasoning, Constraint based recommenders, Case based recommenders.

UNIT-IV:
Hybrid approaches: Opportunities for hybridization, Monolithic hybridization design: Feature combination, Feature augmentation, Parallelized hybridization design: Weighted, Switching, Mixed, Pipelined hybridization design: Cascade Meta-level, Limitations of hybridization strategies.

UNIT-V:
Evaluating Recommender System: Introduction, General properties of evaluation research, Evaluation designs, Evaluation on historical datasets, Error metrics, Decision-Support metrics, User-Centered metrics.

Recommender Systems and communities: Communities, collaboration and recommender systems in personalized web search, Social tagging recommender systems, Trust and recommendations

Text Books:

1. Jannach D., Zanker M. and FelFering A., Recommender Systems: An Introduction, Cambridge University Press(2011), 1st ed.
2. Ricci F., Rokach L., Shapira D., Kantor B.P., Recommender Systems Handbook, Springer(2011), 1st ed.

References:

1. Manouselis N., Drachsler H., Verbert K., Duval E., Recommender Systems For Learning, Springer (2013), 1st ed.

III Year II Semester	DevOps (Professional Elective -II)	L	T	P	C
		3	0	3	4.5

Course Objectives: The main objectives of this course are to:

1. DescribetheagilerelationshipbetweendevelopmentandIToperations.
2. Understand the skill sets and high-functioning teams involved in DevOps and related methods to reach a continuous delivery capability.
3. ImplementautomatedsystemupdateandDevOpslifecycle.

CourseOutcomes: Onsuccessfulcompletionofthiscourse,studentswillbe able to:

- IdentifytheDevOpsConceptsandToolsforeffectiveprojectmanagement.
- UtilizeGITtokeeptrackofdifferent versionsofthesourcecode.
- BuildandAutomateTestusingJenkins.
- ImplementcontainerizationwithDocker.
- UseANSIBLE,Kubernetesforautomationanddeployment.

UNIT-I:

Introduction to DevOps: Introduction to SDLC, Agile Model. Introduction to Devops. DevOps Features, DevOps Architecture, DevOps Lifecycle, Understanding Workflow and principles, Introduction to DevOps tools, Build Automation, Delivery Automation, Understanding Code Quality, Automation of CI/ CD.Release management, Scrum, Kanban, delivery pipeline,bottlenecks, examples

UNIT-II:

SourceCodeManagement(GIT): Theneedforsourcecodecontrol,Thehistory of source code management, Roles and code, source code management system and migrations. What is Version Control and GIT, GIT Installation, GIT features, GIT workflow, working with remote repository, GIT commands, GIT branching, GITstagingandcollaboration.Unittesting-Code Coverage:Junit,nUnit & Code Coverage with Sonar Qube, SonarQube - Code Quality Analysis.

UNIT-III:

Build Automation - Continuous Integration (CI): Build Automation, what isCI Why Cl is Required, CI tools, Introduction to Jenkins (With Architecture), jenkins workflow, jenkins master slave architecture, JenkinsPipelines,PIPELINE BASICS - Jenkins Master, Node, Agent, and Executor FreestyleProjects& Pipelines, Jenkins for Continuous Integration, Create and Manage Builds, User Management in Jenkins Schedule Builds, Launch Builds on Slave Nodes.

UNIT-IV:

Continuous Delivery: Importance of Continuous Delivery, Continuous Deployment CD Flow,

Containerization with Docker: Introduction to Docker, Dockerinstallation,Dockercommands,Images&Containers,DockerFile,runningcontainers,workingwithcontainersandpublishtoDockerHub.TestingTools:IntroductiontoSeleniumanditsfeatures,JavaScripttesting.

UNIT-V:

Configuration Management - ANSIBLE: IntroductiontoAnsible,Ansibletasks Roles,Jinja2 templating, Vaults, Deployments using Ansible.

Containerization Using Kubernetes (Openshift): IntroductiontoKubernetes Namespace& Resources, CI/CD - On OCP, BC, DC& ConfigMaps, Deploying Apps on Openshift Container Pods. Introduction toPuppet masterand Chef.

TextBooks:

1. Joyner, Joseph., Devops for Beginners: Devops Software Development Method Guide for Software Developers and It Professionals, 1st Edition Mihails Konoplows, 2015.
2. Alisson Machado de Menezes., Hands-on DevOps with Linux,1st Edition, BPB Publications, India, 2021.

ReferenceBooks:

1. LenBass,IngoWeber,LimingZhu.DevOps:ASoftwareArchitect's Perspective. Addison Wesley; ISBN-10
2. Gene Kim Je Humble, Patrick Debois, John Willis. The DevOps Handbook, 1st Edition, IT Revolution Press, 2016.
3. Verona,JoakimPracticalDevOps,1stEdition, Packet Publishing,2016.
4. JoakimVerona.PracticalDevops,SecondEdition.Ingramshorttitle;2ndedition (2018). ISBN10: 1788392574
5. Deepak Gaikwad, ViralThakkar.DevOpsToolsfromPractitioner's Viewpoint.Wileypublications.ISBN:9788126579952

III Year II Semester	Software Project Management (Professional Elective -III)	L	T	P	C
		3	0	0	3

Course Objectives:

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

- Todescribeanddeterminethepurposeandimportanceofprojectmanagementfromtheperspectives ofplanning,trackingandcompletionofproject
- Tocompareanddifferentiateorganizationstructuresandprojectstructures
- To implement a project to manage project schedule, expenses and resources with the application ofsuitable projectmanagementtools

UNIT-I:

ConventionalSoftwareManagement: Thewaterfallmodel,conventionalsoftwareManagementperformance.
EvolutionofSoftwareEconomics: SoftwareEconomics,pragmaticsoftwarecostestimation.
ImprovingSoftwareEconomics: ReducingSoftwareproductsize,improvingsoftwareprocesses,improvingteameffectiveness,improvingautomation, Achievingrequiredquality,peerinspections.
Theoldwayand the new: Theprinciples of conventional softwareEngineering,principles of modernsoftwaremanagement,transitioningtoaniterativeprocess.

UNIT-II:

Lifecyclephases: Engineeringandproductionstages,inception,Elaboration,construction,transitionphases.
Artifactsoftheprocess: Theartifactsets,Managementartifacts,Engineeringartifacts,programmaticartifacts.

UNIT-III:

Modelbasedsoftwarearchitectures:AManagementperspectiveandtechnicalperspective.
WorkFlowsoftheprocess: Softwareprocessworkflows, Iterationworkflows.
Checkpointsoftheprocess:Majormilestones,MinorMilestones,Periodicstatusassessments.
IterativeProcessPlanning: Workbreakdownstructures,planningguidelines,costandscheduleestimating,Iterationplanningprocess,Pragmatic planning.

UNIT-IV:

ProjectOrganizationsandResponsibilities: Line-of-BusinessOrganizations,ProjectOrganizations,evolutionofOrganizations.
ProcessAutomation: AutomationBuildingblocks,TheProject Environment.

ProjectControlandProcessinstrumentation: ThesevencoreMetrics,Managementindicators,qualityindicators,life cycle expectations,pragmaticSoftware Metrics,Metricsautomation.

UNIT-V:

Agile Methodology, ADAPTing to Scrum, Patterns for Adopting Scrum, Iterating towards Agility.**FundamentalsofDevOps**:Architecture,Deployments,Orchestration,Need,Instanceofapplications,DevOpsdeliverypipeline,DevOpsecosystem.DevOpsadoptioninprojects:Technologyaspects,Agiling capabilities,Toolstackimplementation,Peopleaspect,processes

TextBooks:

1. SoftwareProject Management,WalkerRoyce,PEA,2005.
2. SucceedingwithAgile:SoftwareDevelopmentUsingScrum,MikeCohn,AddisonWesley.
3. TheDevOpsHandbook:HowtoCreateWorld-ClassAgility,Reliability,andSecurityinTechnology Organizations, Gene Kim, John Willis, Patrick Debois, Jez Humb,1st Edition,O'Reillypublications,2016.

ReferenceBooks:

1. SoftwareProjectManagement,BobHughes,3/e,MikeCotterell,TMH
2. SoftwareProjectManagement,JoelHenry,PEA
3. SoftwareProjectManagementinpractice, PankajJalote,PEA, 2005,
4. EffectiveSoftwareProjectManagement,RobertK.Wysocki,Wiley,2006.
5. ProjectManagementinIT,KathySchwalbe,Cengage

III Year II Semester	Mobile Adhoc Networks (Professional Elective -III)	L	T	P	C
		3	0	0	3

Course Objectives:
From the course the student will learn

- Architect sensor networks for various application setups.
- Devise appropriate data dissemination protocols and model links cost.
- Understanding of the fundamental concepts of wireless sensor networks and has a basic knowledge of the various protocols at various layers.
- Evaluate the performance of sensor networks and identify bottlenecks.

UNIT I:
Introduction to Ad Hoc Wireless Networks- Cellular and Ad Hoc Wireless Networks, Characteristics of MANETs, Applications of MANETs, Issues and Challenges of MANETs, Ad Hoc Wireless Internet, MAC protocols for Ad hoc Wireless Networks-Issues, Design Goals and Classifications of the MAC Protocols.

UNIT II:
Routing Protocols for Ad Hoc Wireless Networks- Issues in Designing a Routing Protocol, Classifications of Routing Protocols, Topology-based versus Position-based Approaches, Issues and design goals of a Transport layer protocol, Classification of Transport layer solutions, TCP over Ad hoc Wireless Networks, Solutions for TCP over Ad Hoc Wireless Networks, Other Transport layer protocols.

UNIT III:
Security protocols for Ad hoc Wireless Networks- Security in Ad hoc Wireless Networks, Network Security Requirements, Issues and Challenges in Security Provisioning, Network Security Attacks, Key Management, Secure Routing in Ad hoc Wireless Networks, Cooperation in MANETs, Intrusion Detection Systems.

UNIT IV:
Basics of Wireless Sensors and Applications- The Mica Mote, Sensing and Communication Range, Design Issues, Energy Consumption, Clustering of Sensors, Applications, Data Retrieval in Sensor Networks-Classification of WSNs, MAC layer, Routing layer, Transport layer, High-level application layer support, Adapting to the inherent dynamic nature of WSNs.

UNIT V:
Security in WSNs- Security in WSNs, Key Management in WSNs, Secure Data Aggregation in WSNs, Sensor Network Hardware-Components of Sensor Mote, Sensor Network Operating Systems–TinyOS, LA-TinyOS, SOS, RETOS, Imperative Language-nesC, **Dataflow Style Language-**TinyGALS, Node-Level Simulators, NS-2 and its sensor network extension, TOSSIM.

Text Books:

1. Ad Hoc Wireless Networks – Architectures and Protocols, 1st edition, C. Siva Ram Murthy, B. S. Murthy, Pearson Education, 2004
2. Ad Hoc and Sensor Networks – Theory and Applications, 2nd edition *Carlos Corderio Dharma P.Aggarwal,* World Scientific Publications / Cambridge University Press, March 2006

Reference Books:

1. Wireless Sensor Networks: An Information Processing Approach, 1st edition*, Feng Zhao, Leonidas Guibas*, Elsevier Science imprint, Morgan Kauffman Publishers, 2005, rp2009
2. Wireless Ad hoc Mobile Wireless Networks – Principles, Protocols and Applications, 1st edition*,* Subir Kumar Sarkar, et al., Auerbach Publications, Taylor & Francis Group, 2008
3. Ad hoc Networking, 1st edition*,Charles E.Perkins*, Pearson Education, 2001
4. Wireless Ad hoc Networking, 1st edition*, Shih-Lin Wu, Yu-Chee Tseng,* Auerbach Publications, Taylor & Francis Group, 2007
5. Wireless Sensor Networks – Principles and Practice, 1st edition, Fei Hu, Xiaojun Cao, An Auerbach book, CRC Press, Taylor & Francis Group, 2010

III Year II Semester	Computer Vision (Professional Elective -III)	L	T	P	C
			0	0	3

Course Objectives:

- To understand the Fundamental Concepts related to sources,shadows and shading
- To understand the Geometry of Multiple Views

Course Outcomes:

1. Implement fundamental image processing techniques required for computer vision
2. Implement boundary tracking techniques
3. Apply chain codes and other region descriptors, Hough Transform for line, circle, and ellipse detections.
4. Apply 3Dvision techniques and Implement motion related techniques.
5. Develop applications using computer vision techniques.

UNIT–I:

Cameras: PinholeCamerasRadiometry–MeasuringLight:LightinSpace, LightSurfaces,ImportantSpecialCasesSources,Shadows, andShading: Qualitative Radiometry, Sources and Their Effects, Local Shading Models, Application:PhotometricStereo,interreflections:GlobalShadingModelsColor: The Physics of Color, Human Color Perception, Representing Color, A Model for ImageColor,SurfaceColorfromImageColor.

UNIT-II:

Linear Filters:Linear Filters and Convolution, Shift Invariant Linear Systems, Spatial Frequency and Fourier Transforms, Sampling and Aliasing, Filters as Templates, Edge Detection:Noise, Estimating Derivatives, Detecting Edges Texture0:Representing Texture, Analysis (and Synthesis) Using Oriented Pyramids, Application: Synthesis by Sampling Local Models, Shape fromTexture.

UNIT-III:

The Geometry of Multiple Views: Two Views Stereopsis: Reconstruction, Human Stereopsis, Binocular Fusion, Using More Cameras Segmentation by Clustering: WhatIsSegmentation?HumanVision:GroupingandGetstalt,Applications: Shot Boundary Detection and Background Subtraction, Image Segmentation by Clustering Pixels, Segmentation by Graph-Theoretic Clustering,

UNIT-IV:

Segmentation by Fitting a Model: The Hough Transform, Fitting Lines, Fitting Curves, fitting as a Probabilistic Inference Problem, Robustness Segmentation and Fitting Using Probabilistic Methods: Missing Data Problems, Fitting, and Segmentation, The EM Algorithm in Practice,

Tracking with Linear Dynamic Models:TrackingasanAbstractInferenceProblem,LinearDynamicModels, KalmanFiltering,DataAssociation,ApplicationsandExamples

UNIT-V:

Geometric Camera Models: Elements of Analytical Euclidean Geometry, Camera Parameters and the Perspective Projection, Affine Cameras and Affine Projection EquationsGeometricCameraCalibration:Least-SquaresParameterEstimation, ALinearApproachtoCameraCalibration,TakingRadialDistortioninto Account, Analytical Photogrammetry,

Case study: Mobile Robot Localization Model- Based Vision: Initial Assumptions, Obtaining Hypotheses by Pose Consistency, Obtaining Hypotheses by pose Clustering, Obtaining Hypotheses Using Invariants, Verification, Case study: Registration in Medical Imaging Systems, Curved Surfaces and Alignment.

TextBooks:

1. David A. Forsyth and Jean Ponce: Computer Vision – A Modern Approach, PHI Learning (Indian Edition), 2009.

ReferenceBooks:

1. E. R. Davies: Computer and Machine Vision – Theory, Algorithms and Practicalities, Elsevier (Academic Press), 4th edition, 2013.
2. R. C. Gonzalez and R. E. Woods “Digital Image Processing” Addison Wesley 2008. 3. Richard Szeliski “Computer Vision: Algorithms and Applications” Springer-Verlag London Limited 2011.

III Year – II Semester	Cloud Computing (Professional Elective -III)	L	T	P	C
		3	0	0	3

Course Objectives:

- To explain the evolving utility computing model called cloud computing.
- To introduce the various levels of services offered by cloud.
- To discuss the fundamentals of cloud enabling technologies such as distributed computing, service-oriented architecture and virtualization.
- To emphasize the security and other challenges in cloud computing.
- To introduce the advanced concepts such as containers, serverless computing and cloud-centric Internet of Things.

UNIT -I:

Introduction to Cloud Computing FundamentalsCloud computing at a glance, defining a cloud, cloud computing reference model, types of services (IaaS, PaaS, SaaS), cloud deployment models (public, private, hybrid), utility computing, cloud computing characteristics and benefits, cloud service providers (Amazon Web Services, Microsoft Azure, Google AppEngine).

UNIT-II:

Cloud Enabling Technologies: Ubiquitous Internet, parallel and distributed computing, elements of parallel computing, hardware architectures for parallel computing (SISD, SIMD, MISD, MIMD), elements ofdistributed computing, Inter-process communication, technologies for distributed computing, remote procedure calls (RPC), service-orientedarchitecture (SOA), Web services, virtualization.

UNIT-III:

Virtualization and Containers: Characteristics of virtualized environments, taxonomy of virtualization techniques,virtualization and cloud Computing, pros and cons of virtualization, technology examples (XEN, VMware),building blocks of containers, container platforms (LXC, Docker), container orchestration, Docker Swarm andKubernetes,public cloud VM (e.g. Amazon EC2) and container (e.g. Amazon Elastic Container Service) offerings.

UNIT-IV:

Cloud computing challenges: Economics of the cloud, cloud interoperability and standards, scalability and fault tolerance, energy efficiency in clouds, federated clouds, cloud computing security, fundamentals of computer security, cloud security architecture, cloud shared responsibility model, security in cloud deployment models.

UNIT -V:

Advanced concepts in cloud computing:Serverless computing, Function-as-a-Service, serverless computing architecture, public cloud (e.g. AWS Lambda) and open-source (e.g. OpenFaaS)serverless platforms, Internet of Things (IoT), applications, cloud-centric IoT and layers, edge and fog computing, DevOps, infrastructure-as-code, quantum cloud computing.

Text Books:

1. Mastering Cloud Computing, 2nd edition, Rajkumar Buyya, Christian Vecchiola, Thamarai Selvi, Shivananda Poojara, Satish N. Srirama, Mc Graw Hill, 2024.
2. Distributed and Cloud Computing, Kai Hwang, Geoffery C. Fox, Jack J. Dongarra, Elsevier, 2012.

Reference Books:

1. Cloud Computing, Theory and Practice, Dan C Marinescu, 2nd edition, MK Elsevier, 2018.
2. Essentials of cloud Computing, K. Chandrasekhran, CRC press, 2014.
3. Online documentation and tutorials from cloud service providers (e.g., AWS, Azure, GCP)

III Year II Semester	Generative AI Lab	L	T	P	C
		0	0	3	1.5

Course Objectives:

1. To learn Python and TensorFlow skills for Generative AI.

2. To study techniques for cleaning and preparing data for Generative AI tasks.

3. To implement generative AI models

4. To develop innovative applications using generative AI tools and techniques.

Course Outcomes:

After learning the course, students will be able to:

1. Implement Python and TensorFlow basics, including data handling and preprocessing techniques.

2. Implement Generative AI models such as GANs, VAEs, LSTM networks, and Transformer models for image text, and music generation tasks.

3. Evaluate model performance and experiment with hyperparameters and optimization techniques to enhance Generative AI outcomes.

4. Develop innovative applications in image, text, and music generation, showcasing practical skills

List of Experiments:

1. Write Python scripts to implement basic operations and TensorFlow 2 tensors
2. Implement a Generative Adversarial Network (GAN) architecture using TensorFlow 2. Train theGAN model on a dataset such as MNIST or CIFAR-10 for image generation tasks.
3. Train a GAN model on a custom dataset for image generation. Experiment with hyperparameters,loss functions, and optimization techniques to optimize GAN training.
4. Explore advanced techniques such as Wasserstein GANs, Progressive GANs, or StyleGANs forimage generation. Implement and compare these techniques for generating high-quality images.
5. Develop applications for image and video generation using trained Generative AI models. Use themodels to generate art, create deep fakes, or synthesize video content.
6. **Text Generation:** Implement a Long Short-Term Memory (LSTM) network using TensorFlow 2 fortext generation tasks. Train the LSTM model on a dataset of text sequences and generate new textsamples.

7. **Text generation:** Implement a Transformer-based language model (e.g., GPT) using TensorFlow 2for text generation. Fine-tune the model on a text corpus and generate coherent and contextuallyrelevant text.
8. **Text generation:** Fine-tune a pre-trained language model (e.g., GPT, BERT) using transfer learningtechniques. Fine-tune the model on a domain-specific dataset and evaluate its performance for textgeneration tasks.
9. **Text generation:** Develop applications for text generation tasks such as story generation, dialoguegeneration, or code generation using trained Generative AI models.
10. **Music Generation:**Preprocess music data and represent it in a suitable format for music generation tasks. Explore MIDIor audio representations for training Generative AI models.
11. **Music Generation:**Implement a Long Short-Term Memory (LSTM) network using TensorFlow 2 for music generation.Train the LSTM model on a dataset of music sequences and generate new musical compositions.
12. **Generate Novel Music Compositions:**Transformer-based Music Generation: Implement a Transformer-based architecture (e.g.,MusicBERT, MusicGPT) using TensorFlow 2 for music generation. Fine-tune the model on a musicdataset and generate novel music compositions.

References:

1. Responsible AI: Implementing Ethical and Unbiased Algorithms, by Shashin Mishra and Sray Agarwal
2. Generative AI in Practice: 100+ Amazing Ways Generative Artificial Intelligence is Changing Business andSociety, Bernard Marr
3. “Generative AI with Python and TensorFlow 2: Create images, text, and music with VAEs, GANs, LSTMs,Transformer models”, Joseph Babcock and Raghav Bali
4. "Generative Adversarial Networks: An Overview" by Vinod Nair and Geoffrey E. Hinton.
5. "Hands-On Generative Adversarial Networks with PyTorch 1.x" by Stefano Bosisio and Vijayabhaskar J.

III Year – II Semester	Data Visualization Lab	L	T	P	C
		0	0	3	1.5

Course Objectives:

- To visualize the different datasets using histograms, line charts.
- To understand the use of bar charts and box plots.
- To understand Scatter plots, mosaic plots
- To understand different Map visualizations
- To learn advanced graphs such as correlogram, heatmap and 3D graphs.

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

- Visualize the different datasets using histograms, line charts.
- Makeuse of bar charts and box plots on different datasets
- Apply Scatter plots, mosaic plots in R for different datasets
- Apply different Map visualizations in R
- Create advanced graphs such as correlogram, heatmap and 3D graphs.

List of Experiments:

1. a) Load VADeaths(Death Rates in Virginia)dataset in R and visualize the data using different histograms.
   b) Load air quality dataset in R and visualize La Guardia Airport's dialy maximum temperature using histogram.
2. Load AirPassengers dataset in R and visualize the data using line chart that shows increase in air passengers over given time period.
3. a) Load iris dataset in R, visualize the data using different Bar Charts and also demonstrate the use of stacked plots.
   b)Load air quality dataset in R and visualize ozone concentration in air.
4. a) Load iris dataset in R, visualize the data using different Box plots including group by option and also use color palette to represent species.
   b) Load air quality dataset in R and visualize air quality parameters using box plots.
5. Visualize iris dataset using simple scatter, multivariate scatter plot and also visualize scatter plot matrix to visualize multiple variables across each other.
6. Load diamonds dataset in R and visualize the structure in datasets with large data points using hexagon binning and also add color palette then use the
7. Load HairEyeColor dataset in R and plot categorical data using mosaic plot.
8. Load mtcars dataset in R and visualize data using heat map.
9. Install leaflet library in R and perform different map visualizations.
10. Visualize iris dataset using 3d graphs such as scatter3d, cloud, xyplot.
11. Make use of correlogram to visualize data in correlation matrices for iris dataset.
12. Install maps library in R and draw different map visualizations.

Web References:

1. https://www.analyticsvidhya.com/blog/2015/07/guide-data-visualization-r/
2. https://www.geeksforgeeks.org/data-visualization-in-r/

III Year II Semester	Soft skills	L	T	P	C
	(Skill Enhancement Course)	0	1	2	2

Course Objectives:

- To equip the students with the skills to effectively communicate in English
- To train the students in interview skills, group discussions and presentation skills
- To motivate the students to develop confidence
- To enhance the students' interpersonal skills
- To improve the students' writing skills

UNIT – I

Analytical Thinking & Listening Skills: Self-Introduction, Shaping Young Minds - A Talk by Azim Premji (Listening Activity), Self – Analysis, Developing Positive Attitude, Perception.

Communication Skills: Verbal Communication; Non-Verbal Communication (Body Language)

UNIT – II

Self-Management Skills: Anger Management, Stress Management, Time Management, Six Thinking Hats, Team Building, Leadership Qualities

Etiquette: Social Etiquette, Business Etiquette, Telephone Etiquette, Dining Etiquette

UNIT – III

Standard Operation Methods: Basic Grammars, Tenses, Prepositions, Pronunciation, Letter Writing; Note Making, Note Taking, Minutes Preparation, Email & Letter Writing

UNIT-IV

Job-Oriented Skills: Group Discussion, Mock Group Discussions, Resume Preparation, Interview Skills, Mock Interviews

UNIT-V

Interpersonal relationships: Introduction, Importance, Types, Uses, Factors affecting interpersonal relationships, Accommodating different styles, Consequences of interpersonal relationships

Text books:

1. Barun K. Mitra, Personality Development and Soft Skills, Oxford University Press, 2011.

2. S.P. Dhanavel, English and Soft Skills, Orient Blackswan, 2010.

Reference books:

1. R.S.Aggarwal, A Modern Approach to Verbal & Non-Verbal Reasoning, S.Chand& Company Ltd., 2018.
2. Raman, Meenakshi& Sharma, Sangeeta, Technical Communication Principles and Practice, Oxford University Press, 2011.

E-resources:

1. https://swayam-plus.swayam2.ac.in/courses/course-details?id=P_CAMBR_01

III Year II Semester	Technical Paper Writing & IPR	L	T	P	C
		2	0	0	-

Course Objective: The course will explain the basic related to writing the technical reports and understanding the concepts related to formatting and structuring the report. This will help students to comprehend the concept of proofreading, proposals and practice

Unit I:

Introduction: An introduction to writing technical reports, technical sentences formation, using transitions to join sentences, Using tenses for technical writing.

Planning and Structuring: Planning the report, identifying reader(s), Voice, Formatting and structuring the report, Sections of a technical report, Minutes of meeting writing.

Unit II:

Drafting report and design issues: The use of drafts, Illustrations and graphics.

Final edits: Grammar, spelling, readability and writing in plain English: Writing in plain English, Jargon and final layout issues, Spelling, punctuation and Grammar, Padding, Paragraphs, Ambiguity.

Unit III:

Proofreading and summaries: Proofreading, summaries, Activities on summaries. **Presenting final reports:** Printed presentation, Verbal presentation skills, Introduction to proposals and practice.

Unit IV:

Using word processor: Adding a Table of Contents, Updating the Table of Contents, Deleting the Table of Contents, Adding an Index, Creating an Outline, Adding Comments, Tracking Changes, Viewing Changes, Additions, and Comments, Accepting and Rejecting Changes , Working with Footnotes and Endnotes, Inserting citations and Bibliography, Comparing Documents, Combining Documents, Mark documents final and make them read only., Password protect Microsoft Word documents., Using Macros,

Unit V:

Nature of Intellectual Property: Patents, Designs, Trade and Copyright. Process of **Patenting and Development:** technological research, innovation, patenting, development. International Scenario: International cooperation on Intellectual Property

Text Books:

1. Kompal Bansal & Parshit Bansal, "Fundamentals of IPR for Beginner's", 1st Ed., BS Publications, 2016.
2. William S. Pfeiffer and Kaye A. Adkins, "Technical Communication: A Practical Approach", Pearson.
3. Ramappa,T., "Intellectual Property Rights Under WTO", 2ndEd., S Chand, 2015.

Reference Books:

1. Adrian Wallwork, English for Writing Research Papers, Springer New York Dordrecht Heidelberg London, 2011.
2. Day R, How to Write and Publish a Scientific Paper, Cambridge University Press(2006)

E-resources:

1. https://www.udemy.com/course/reportwriting/
2. https://www.udemy.com/course/professional-business-english-and-technical-report-writing/
3. https://www.udemy.com/course/betterbusinesswriting/