



B. Tech AI & ML (IS-23) COURSE STRUCTURE & SYLLABUS

B. Tech. – III Year I Semester

S.No	Subject code	Title	L	T	P	Credits
1	23PC3110C	Deep Learning	3	0	0	3
2	23PC3102C	Computer Networks	3	0	0	3
3	23PC3111C	Natural Language Processing	3	0	0	3
4	23PE3105C	1. Automata Theory & Compiler Design	3	0	0	3
	23PE3101C	2. Operating Systems				
	23PE3106C	3. Internet of Things(IoT)				
	23PE3107C	4. Exploratory Data Analysis with Python				
		5. Object Oriented Analysis and Design				
5		Deep Learning Lab	3	0	0	3
6	23PC3116C	Natural Language Processing Lab	0	0	3	1.5
7	23PC3113C	Full Stack Development -2 / SWAYAM Plus – Data Engineer / AI Engineer	0	0	3	1.5
8	23SE3101C	Tinkering Lab (User Interface Design using Flutter) / SWAYAM Plus - Android Application Development (with Flutter)	0	1	2	2
9	23ES3101C	Deep Learning Lab	0	0	2	1
10		Evaluation of Community Service Internship	-	-	-	2
		Total	15	01	10	23
MC		Student may select from the same Minor Pool	3	0	0	3
MC		Minor Course through SWAYAM / NPTEL (Minimum 12 Week, 3 credit course)	3	0	0	3
HC		Student may select from the same Honor's Pool	3	0	0	3
HC		Student may select from the same Honor's Pool	3	0	0	3



B. Tech.– III Year II Semester

S. No.	Category	Title	L	T	P	Credits
1	23PC3207C	Reinforcement Learning	3	0	0	3
2	23PC3208C	Big Data Analytics	3	0	0	3
3	23PC3209C	Data Visualization	3	0	0	3
4	23PE3206C 23PE3211C 23PE3205C 23PE3218C 23PE3221C	1. Cryptography & Network Security 2. Recommender Systems 3. Software Engineering 4. Social Network Analysis 5. 12-Week SWAYAM /NPTEL Course suggested by the BoS	3	0	0	3
5	23PE3208C 23PE3207C 23PE3203C 23PE3220C 23PE3221C	1. Computer Vision 2. Cloud Computing 3. DevOps 4. Soft Computing 5. 12-Week SWAYAM /NPTEL Course suggested by the BoS	3	0	0	3
6		Open Elective – II	3	0	0	3
7	23PC3214C	Big Data Analytics Lab	0	0	3	1.5
8	23PC3215C	Data Visualization Lab	0	0	3	1.5
9	23SC3201C	Soft skills / SWAYAM Plus - 21st Century Employability Skills	0	1	2	2
10	23AC3201C	Technical PaperWriting & IPR	2	0	0	-
Total			20	1	8	23
*Mandatory Industry Internship of 08 weeks duration during summer vacation						
MC	Student may select from the same minor's pool		3	0	3	4.5
HC	Student may select from the same 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



III B. Tech 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:

1. Explore feed forward networks and Deep Neural networks
2. Mathematically understand the deep learning approaches and paradigms
3. Apply the deep learning techniques for various applications

UNIT-I: Basics- Biological Neuron, Idea of computational units, McCulloch–Pitts unit and Thres holding 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, autoencoders.

Deep Neural Networks:Difficulty of training deep neural networks,Greedylayer wise training.

UNIT-III: Better Training of Neural Networks-Newer optimization methods forneural networks (Adagrad, adadelta, rmsprop, adam, NAG), second order methods for training, Saddle point problem inneural networks, Regularization methods (dropout, drop connect, batch normalization).

UNIT-IV: 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.

UNIT-V: Recent trends-Variational Autoencoders, Transformers,GPT Applications: Vision, NLP, Speech

Text Books:

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 B. Tech 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

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 Layer Design 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 Logging, 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 B. Tech I Semester	NATURAL LANGUAGE PROCESSING	L	T	P	C
		3	0	0	3



Course Objectives: This course introduces the fundamental concepts and techniques of natural language processing (NLP).

- Students will gain an in-depth understanding of the computational properties of natural languages and the commonly used algorithms for processing linguistic information.
- The course examines NLP models and algorithms using both the traditional symbolic and the more recent statistical approaches.
- Enable students to be capable to describe the application based on natural language processing and to show the points of syntactic, semantic and pragmatic processing.

UNIT I:INTRODUCTION: Origins and challenges of NLP – Language Modelling: Grammar-based LM, Statistical LM – Regular Expressions, Finite-State Automata – English Morphology, Transducers for lexicon and rules, Tokenization, Detecting and Correcting Spelling Errors, Minimum Edit Distance.

UNIT II:WORD LEVEL ANALYSIS: Unsmoothed N-grams, Evaluating N-grams, Smoothing, Interpolation and Backoff – Word Classes, Part- of-Speech Tagging, Rule-based, Stochastic and Transformation-based tagging, Issues in PoS tagging – Hidden Markov and Maximum Entropy models.

UNIT III: SYNTACTIC ANALYSIS: Context-Free Grammars, Grammar rules for English, Treebanks, Normal Forms for grammar – Dependency Grammar – Syntactic Parsing, Ambiguity, Dynamic Programming parsing – Shallow parsing Probabilistic CFG, Probabilistic CYK, Probabilistic Lexicalized CFGs – Feature structures, Unification of feature structures

UNIT IV:SEMANTICS AND PRAGMATICS: Requirements for representation, First-Order Logic, Description Logics – Syntax-Driven Semantic analysis, Semantic attachments – Word Senses, Relations between Senses, Thematic Roles, selectional restrictions – Word Sense Disambiguation, WSD using Supervised, Dictionary & Thesaurus, Bootstrapping methods – Word Similarity using Thesaurus and Distributional methods.

UNIT V:DISCOURSE ANALYSIS AND LEXICAL RESOURCES: Discourse segmentation, Coherence – Reference Phenomena, Anaphora Resolution using Hobbs and Centering Algorithm – Coreference Resolution – Resources: Porter Stemmer, Lemmatizer, Penn Treebank, Brill’s Tagger, WordNet, PropBank, FrameNet, Brown Corpus, British National Corpus (BNC).



Text Books:

1. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech, 2ndEdition, Daniel Jurafsky, James H. Martin - Pearson Publication,2014.
2. Natural Language Processing with Python, First Edition, Steven Bird, Ewan Klein and Edward Loper, OReilly Media,2009.

Reference Books:

1. Language Processing with Java and Ling Pipe Cookbook, 1stEdition, Breck Baldwin, Atlantic Publisher, 2015.
2. Natural Language Processing with Java, 2ndEdition, Richard M Reese, OReilly Media,2015.
3. Handbook of Natural Language Processing, Second, Nitin Indurkha and Fred J. Damerau, Chapman and Hall/CRC Press, 2010.Edition
4. Natural Language Processing and Information Retrieval, 3rdEdition, Tanveer Siddiqui, U.S. Tiwary, Oxford University Press,2008.



III B. Tech I Semester	AUTOMATA THEORY AND COMPILER DESIGN	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 ϵ -transitions to NFA without ϵ -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, 3rd 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, Jeffrey D. Ullman, 2nd Edition, Pearson.
3. Theory of Computer Science – Automata languages and computation, Mishra and Chandrashekar, 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. Loudon, Thomson. Course Technology.



		3	0	0	3
--	--	----------	----------	----------	----------

Course Objectives: The main objectives of the course are to make students

- 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 system performance.
- Illustrate different conditions for deadlocks and their possible solutions.

UNIT-I: Operating Systems Overview: Introduction, operating system functions, operating system 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:** Multithreading 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, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock.

UNIT-IV: Memory-Management Strategies: Introduction, Contiguous Memory Allocation, Paging, Structure of the Page Table, Swapping.

Virtual Memory Management: Introduction, Demand Paging, Copy-on-Write, Page Replacement, Allocation of Frames, Thrashing.

Storage Management: Overview of Mass Storage Structure, HDD Scheduling.

UNIT-V: File System: File System Interface, File Concept, Access Methods, Directory Structure. File System Implementation: File-System Structure, File-System Operations, Directory Implementation, Allocation Methods, Free Space Management.

File-System Internals: File-System Mounting, Partitions and Mounting, File Sharing.

Protection: Goals of Protection, Principles of Protection, Protection Rings, Domain of Protection, Access Matrix.

Text Books:

1. Operating System Concepts, Silberschatz A, Galvin PB, Gagne G, 10th Edition, Wiley,



2018. Modern Operating Systems, Tanenbaum AS, 4th Edition, Pearson, 2016.

Reference Books:

1. Operating Systems - Internals and Design Principles, Stallings W, 9th Edition, Pearson, 2018.
2. Operating Systems: A Concept-Based Approach, D. M. Dhamdhare, 3rd Edition, McGraw-Hill, 2013.

Online Learning Resources:

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



III B. Tech I Semester	INTERNET OF THINGS	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.

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.



**INTERNATIONAL SCHOOL OF
TECHNOLOGY AND SCIENCES FOR WOMEN
(An Autonomous Institution)**

(Approved by AICTE New Delhi, Affiliated to JNTUK Kakinada, Accredited by NBA and NAAC with A+ Grade)

NH-16, East Gonagudem (V), Rajanagaram (M), Rajamahendravaram, East Godavari -533294

Mobile : 9505506119 / 9505506101 | Email ID : istswomens101@gmail.com | website : www.ists.ac.in



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, Univesity Press, 2015

Reference Books:

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



III B. Tech I Semester	EXPLORATORY DATA ANALYSIS WITH PYTHON (SKILL ENHANCEMENT COURSE)	L	T	P	C
		3	0	0	3

Course Objectives: The main objectives of the course are to

- Introduce the fundamentals of Exploratory Data Analysis
- Cover essential exploratory techniques for understanding multivariate data by summarizing it through statistical methods and graphical methods.
- Evaluate the Models and select the best model

UNIT-I: Exploratory Data Analysis Fundamentals: Understanding data science, The significance of EDA, Steps in EDA, Making sense of data, Numerical data, Categorical data, Measurement scales, Comparing EDA with classical and Bayesian analysis, Software tools available for EDA, Getting started with EDA.

Sample Experiments:

1. a) Download Dataset from Kaggle using the following link :
<https://www.kaggle.com/datasets/sukhmanibedi/cars4u>
- b) Install python libraries required for Exploratory Data Analysis (numpy, pandas, matplotlib, seaborn)
2. Perform Numpy Array basic operations and Explore Numpy Built-in functions.
3. Loading Dataset into pandas dataframe
4. Selecting rows and columns in the dataframe

UNIT-II: Visual Aids for EDA: Technical requirements, Line chart, Bar charts, Scatter plot using seaborn, Polar chart, Histogram, Choosing the best chart

Case Study: EDA with Personal Email, Technical requirements, Loading the dataset, Data transformation, Data cleansing, Applying descriptive statistics, Data refactoring, Data analysis.

Sample Experiments:

1. Apply different visualization techniques using sample dataset
 - a. Line Chart
 - b. Bar Chart
 - c. Scatter Plots
 - d. Bubble Plot
2. Generate Scatter Plot using seaborn library for iris dataset
3. Apply following visualization Techniques for a sample dataset
 - a. Area Plot
 - b. Stacked Plot
 - c. Pie chart
 - d. Table Chart
4. Generate the following charts for a dataset.
 - a. Polar Chart
 - b. Histogram
 - c. Lollipop chart

Case Study: Perform Exploratory Data Analysis with Personal Email Data



UNIT-III: Data Transformation: Merging database-style dataframes, Concatenating along with an axis, Merging on index, Reshaping and pivoting, Transformation techniques, Handling missing data, Mathematical operations with NaN, Filling missing values, Discretization and binning, Outlier detection and filtering, Permutation and random sampling, Benefits of data transformation, Challenges.

Sample Experiments:

1. Perform the following operations
 - a) Merging Dataframes
 - b) Reshaping with Hierarchical Indexing
 - c) Data Deduplication
 - d) Replacing Values
2. Apply different Missing Data handling techniques
 - a) NaN values in mathematical Operations
 - b) Filling in missing data
 - c) Forward and Backward filling of missing values
 - d) Filling with index values
 - e) Interpolation of missing values
3. Apply different data transformation techniques
 - a) Renaming axis indexes
 - b) Discretization and Binning
 - c) Permutation and Random Sampling
 - d) Dummy variables

UNIT-IV: Descriptive Statistics: Distribution function, Measures of central tendency, Measures of dispersion, Types of kurtosis, Calculating percentiles, Quartiles, Grouping Datasets, Correlation, Understanding univariate, bivariate, multivariate analysis, Time Series Analysis

Sample Experiments:

1. Study the following Distribution Techniques on a sample data
 - a) Uniform Distribution
 - b) Normal Distribution
 - c) Gamma Distribution
 - d) Exponential Distribution
 - e) Poisson Distribution
 - f) Binomial Distribution
2. Perform Data Cleaning on a sample dataset.
3. Compute measure of Central Tendency on a sample dataset
 - a) Mean
 - b) Median
 - c) Mode
4. Explore Measures of Dispersion on a sample dataset
 - a) Variance
 - b) Standard Deviation
 - c) Skewness
 - d) Kurtosis
5. a) Calculating percentiles on sample dataset
 - b) Calculate Inter Quartile Range(IQR) and Visualize using Box Plots
6. Perform the following analysis on automobile dataset.
 - a) Bivariate analysis
 - b) Multivariate analysis
7. Perform Time Series Analysis on Open Power systems dataset

UNIT-V: Model Development and Evaluation: Unified machine learning workflow, Data pre-processing, Data preparation, Training sets and corpus creation, Model creation and training, Model evaluation, Best model selection and evaluation, Model deployment
Case Study: EDA on Wine Quality Data Analysis

Sample Experiments:

1. Perform hypothesis testing using stats models library



a) Z-Test b)T-Test

2. Develop model and Perform Model Evaluation using different metrics such as prediction score, R2 Score, MAE Score, MSE Score.

Case Study: Perform Exploratory Data Analysis with Wine Quality Dataset

Text Book:

1. Suresh Kumar Mukhiya, Usman Ahmed, Hands-On Exploratory Data Analysis with Python, Packt Publishing, 2020.

Reference Books:

1. Ronald K. Pearson, Exploratory Data Analysis Using R, CRC Press, 2020
2. RadhikaDatar, Harish Garg, Hands-On Exploratory Data Analysis with R: Become an expert in exploratory data analysis using R packages, 1st Edition, Packt Publishing, 2019

Web References:

1. <https://github.com/PacktPublishing/Hands-on-Exploratory-Data-Analysis-with-Python>
2. <https://www.analyticsvidhya.com/blog/2022/07/step-by-step-exploratory-dataanalysis-eda-using-python/#h-conclusion>
3. <https://github.com/PacktPublishing/Exploratory-Data-Analysis-with-Python-Cookbook>

III B. Tech I Semester	OBJECT ORIENTED ANALYSIS	L	T	P	C
		3	0	0	3



	AND DESIGN				
--	-------------------	--	--	--	--

Course Objectives: The main objective is for students to

- Become familiar with all phases of Object-Oriented Analysis and Design (OOAD).
- Master the main features of UML,
- Understand Object Technologies and their applications, develop problem-solving skills in various domains
- Learn Object Design Principles and their implementation.

UNIT-I:Introduction: The Structure of Complex Systems, The Inherent Complexity of Software, Attributes of Complex Systems, Organized and Disorganized Complexity, Bringing Order to Chaos, Designing Complex Systems.

Case Study: System Architecture - Satellite-Based Navigation.

UNIT-II:Introduction to UML: Importance of modelling, principles of modelling, object-oriented modelling, conceptual model of UML, Architecture, and Software Development Life Cycle. Basic Structural Modelling: Classes, Relationships, Common Mechanisms, and Diagrams.

Case Study: Control System - Traffic Management.

UNIT-III:Class & Object Diagrams: Terms, concepts, and modelling techniques for Class & Object Diagrams. Advanced Structural Modelling: Advanced Classes, Advanced Relationships, Interfaces, Types and Roles, Packages.

Case Study: AI - Cryptanalysis.

UNIT-IV:Basic BehaviouralModelling-I: Interactions, Interaction Diagrams, Use Cases, Use Case Diagrams, Activity Diagrams.

Case Study: Web Application - Vacation Tracking System.

UNIT -V:Advanced BehaviouralModelling: Events and Signals, State Machines, Processes and Threads, Time and Space, State Chart Diagrams. Architectural Modelling: Component, Deployment, Component Diagrams, and Deployment Diagrams.

Case Study: Weather Forecasting.



Text Books:

1. Grady Booch, Robert A. Maksimchuk, Michael W. Engle, Bobbi J. Young, Jim Conallen, Kellia Houston, "Object-Oriented Analysis and Design with Applications", 3rd edition, 2013, PEARSON.
2. Grady Booch, James Rumbaugh, Ivar Jacobson: "The Unified Modeling Language User Guide", Pearson Education.

Reference Books:

1. Meilir Page-Jones: "Fundamentals of Object-Oriented Design in UML", Pearson Education.
2. Pascal Roques: "Modeling Software Systems Using UML2", WILEY-Dreamtech India Pvt. Ltd.
3. Atul Kahate: "Object-Oriented Analysis & Design", The McGraw-Hill Companies. Applying UML and Patterns: "An Introduction to Object-Oriented Analysis and Design and Unified Process", Craig Larman, Pearson Education.



III B. Tech 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 results of two 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 a neural network for classifying movie reviews (Binary Classification) using IMDB dataset.
3. Design a neural Network for classifying news wires (Multi classclassification) using Reuters dataset.
4. Design a 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. RezaZadeh and Bharath Ramsundar,“TensorflowforDeepLearning”, O’Reilly publishers, 2018

References Books:

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



III B. Tech I Semester	NATURAL LANGUAGE PROCESSING LAB	L	T	P	C
		0	0	3	1.5

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

- CO 1:** Use the NLTK and spaCy toolkit for NLP Programming. (L3)
- CO 2:** Analyze various corpora for developing programs. (L4)
- CO 3:** Develop various pre-processing techniques for a given corpus. (L6)
- CO4:** Develop programming logic using NLTK functions. (L6)
- CO 5:** Build applications using various NLP techniques for a given corpus. (L6)

List of Experiments:

1. Installation and exploring features of NLTK and spa Cytools. Download Word Cloud and few corpora.
2. (i) Write a program to implement word Tokenizer, Sentence and Paragraph Tokenizers.
(ii) Check how many words are there in any corpus. Also check how many distinct words are there?
3. (i) Write a program to implement both user-defined and pre-defined functions to generate
 - (a) Uni-grams
 - (b) Bi-grams
 - (c) Tri-grams
 - (d) N-grams
 (ii) Write a program to calculate the highest probability of a word (w2) occurring after another word (w1).
4. (i) Write a program to identify the word collocations.
(ii) Write a program to print all words beginning with a given sequence of letters.
(iii) Write a program to print all words longer than four characters.
5. (i) Write a program to identify the mathematical expression in a given sentence.
(ii) Write a program to identify different components of an email address.
6. (i) Write a program to identify all antonyms and synonyms of a word.
(ii) Write a program to find hyponymy, homonymy, polysemy for a given word.
7. (i) Write a program to find all the stop words in any given text.
(ii) Write a function that finds the 50 most frequently occurring words of a text that are not stop words.
8. Write a program to implement various stemming techniques and prepare a chart with the performance of each method.
9. Write a program to implement various lemmatization techniques and prepare a chart with the performance of each method.



10. (i) Write a program to implement Conditional Frequency Distributions (CFD) for any corpus.
 (ii) Find all the four-letter words in any corpus. With the help of a frequency distribution (FreqDist), show these words in decreasing order of frequency.
 (iii) Define a conditional frequency distribution over the names of a corpus that allows you to see which initial letters are more frequent for males versus females.

11. (i) Write a program to implement Part-of-Speech (PoS) tagging for any corpus.
 (ii) Write a program to identify which word has the greatest number of distinct tags? What are they, and what do they represent?
 (iii) Write a program to list tags in order of decreasing frequency and what do the 20 most frequent tags represent?
 (iv) Write a program to identify which tags are nouns most commonly found after? What do these tags represent?

12. Write a program to implement TF-IDF for any corpus.

13. Write a program to implement chunking and chunking for any corpus.

14. (i) Write a program to find all the misspelled words in a paragraph.
 (ii) Write a program to prepare a table with frequency of misspelled tags for any given text.

15. Write a program to implement all the NLP Pre-Processing Techniques required to perform further NLP tasks.

Case Studies: (At Least any one Case Study has to be performed)

Case Study-1. Write a program to implement Named Entity Recognition (NER) for any corpus.

Case Study-2. Write a program to perform Auto-Correction of spellings for any text.

Case Study-3. Check for all positive words in a news article/ any text.

REFERENCE BOOKS:

1. Toolkit Steven Bird, Ewan Klein, and Edward Loper, "Natural Language Processing with Python – Analyzing Text with the Natural Language".

WEB REFERENCES:

1. <http://www.nptelvideos.in/2012/11/natural-languageprocessing.html>



III B. Tech I Semester	FULL STACK DEVELOPMENT -2	L	T	P	C
		0	1	2	2

List of Experiments:

Experiment 11: Node.js

- Write a program to show the workflow of JavaScript code executable by creating web server in Node.js.
- Write a program to transfer data over http protocol using http module.
- Create a text file src.txt and add the following content to it. (HTML, CSS, Javascript, Typescript, MongoDB, Express.js, React.js, Node.js)
- Write a program to parse an URL using URL module.
- Write a program to create an user-defined module and show the workflow of Modularization of application using Node.js

Experiment 12: Typescript

- Write a program to understand simple and special types.
- Write a program to understand function parameter and return types.
- Write a program to show the importance with Arrow function. Use optional, default and REST parameters.
- Write a program to understand the working of typescript with class, constructor, properties, methods and access specifiers.
- Write a program to understand the working of namespaces and modules.
- Write a program to understand generics with variables, functions and constraints.

Experiment 13-15: Augmented Programs: (Any 2 must be completed)

- Write a CSS program, to apply 2D and 3D transformations in a web page.
- Design a web page with new features of HTML5 and CSS3.
- Design a to-do list application using Javascript.

Experiment 1: ExpressJS – Routing, HTTP Methods, Middleware.

- Write a program to define a route, Handling Routes, Route Parameters, Query Parameters and URL building.
- Write a program to accept data, retrieve data and delete a specified resource using http methods.
- Write a program to show the working of middleware.

Experiment 2: ExpressJS – Templating, Form Data

- Write a program using templating engine.
- Write a program to work with form data.



- Experiment 3: ExpressJS – Cookies, Sessions, Authentication**
- Write a program for session management using cookies and sessions.
 - Write a program for user authentication.
- Experiment 4: ExpressJS – Database, RESTful APIs**
- Write a program to connect MongoDB database using Mongoose and perform CRUD operations.
 - Write a program to develop a single page application using RESTful APIs.
- Experiment 5: ReactJS – Render HTML, JSX, Components – function & Class**
- Write a program to render HTML to a web page.
 - Write a program for writing markup with JSX.
 - Write a program for creating and nesting components (function and class).
- Experiment 6: ReactJS – Props and States, Styles, Respond to Events**
- Write a program to work with props and states.
 - Write a program to add styles (CSS & Sass Styling) and display data.
 - Write a program for responding to events.
- Experiment 7: ReactJS – Conditional Rendering, Rendering Lists, React Forms**
- Write a program for conditional rendering.
 - Write a program for rendering lists.
 - Write a program for working with different form fields using react forms.
- Experiment 8: ReactJS – React Router, Updating the Screen**
- Write a program for routing to different pages using react router.
 - Write a program for updating the screen.
- Experiment 9: ReactJS – Hooks, Sharing data between Components**
- Write a program to understand the importance of using hooks.
 - Write a program for sharing data between components.
- Experiment 10: ReactJS Applications – To-do list and Quiz**
- Design to-do list application.
- Experiment 11: MongoDB – Installation, Configuration, CRUD operations**
- Install MongoDB and configure ATLAS
 - Write MongoDB queries to perform CRUD operations on document using insert(), find(), update(), remove()
- Experiment 12: MongoDB – Databases, Collections and Records**
- Write MongoDB queries to Create and drop databases and collections.
 - Write MongoDB queries to work with records using find(), limit(),



sort(), createIndex(), aggregate().

Experiment 13-15: Augmented Programs: (Any 2 must be completed)

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

Text Books:

1. Programming the World Wide Web, 7th Edition, Robert W Sebesta, Pearson, 2013.
2. Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node, Vasam Subramanian, 2nd edition, APress, O'Reilly.

Reference Books

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 B. Tech I Semester	TINKERING LAB (UI DESIGN-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 Widges 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

1. a) Install Flutter and Dart SDK.
b) Write a simple Dart program to understand the language basics.
2. a) Explore various Flutter widgets (Text, Image, Container, etc.).
b) Implement different layout structures using Row, Column, and Stack widgets.
3. a) Design a responsive UI that adapts to different screen sizes.
b) Implement media queries and breakpoints for responsiveness.
4. a) Set up navigation between different screens using Navigator.
b) Implement navigation with named routes.
5. a) Learn about stateful and stateless widgets.
b) Implement state management using set State and Provider.
6. a) Create custom widgets for specific UI elements.
b) Apply styling using themes and custom styles.
7. a) Design a form with various input fields.
b) Implement form validation and error handling.
8. a) Add animations to UI elements using Flutter's animation framework.
b) Experiment with different types of animations (fade, slide, etc.).
9. a) Fetch data from a REST API.
b) Display the fetched data in a meaningful way in the UI.
10. 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 1stEdition, Apres



III B. Tech II Semester	REINFORCEMENT LEARNING	L	T	P	C
		3	0	0	3

Course Objective:

- To provide the fundamentals of Reinforcement learning.

Course Outcomes:

- Enumerate the elements of Reinforcement Learning
- Solve then-armed Bandit problem
- Compare different Finite Markov Decision Process
- Discuss about Monte Carlo Methods in solving real world problems
- List the Applications and Case Studies of Reinforcement Learning

UNIT-I: The Reinforcement Learning Problem: Reinforcement Learning, Examples, Elements of Reinforcement Learning, Limitations and Scope, An Extended Example: Tic-Tac-Toe, Summary, History of Reinforcement Learning.

UNIT-II: Multi-arm Bandits: An n-Armed Bandit Problem, Action-Value Methods, Incremental Implementation, tracking a Nonstationary Problem, Optimistic Initial Values, Upper-Confidence-Bound Action Selection, Gradient Bandits, Associative Search (Contextual Bandits)

UNIT-III: Finite Markov Decision Processes: The Agent–Environment Interface, Goals and Rewards, Returns, Unified Notation for Episodic and Continuing Tasks, The Markov Property, Markov Decision Processes, Value Functions, Optimal Value Functions, Optimality and Approximation.

UNIT-IV: Monte Carlo Methods: Monte Carlo Prediction, Monte Carlo Estimation of Action Values, Monte Carlo Control, Monte Carlo Control without Exploring Starts, Off-policy Prediction via Importance Sampling, Incremental Implementation, Off-Policy Monte Carlo Control, Importance Sampling on Truncated Returns

UNIT-V: Applications and Case Studies: TD-Gammon, Samuel’s Checkers Player, The Acrobot, Elevator Dispatching, Dynamic Channel Allocation, Job-Shop Scheduling.

Text Books:

1. Richard S. Sutton and Andrew G. Barto, “Reinforcement Learning- An Introduction”, 2nd Edition, The MIT Press, 2018
2. Marco Wiering, Martijn van Otterlo Reinforcement Learning: State-of-the-Art (Adaptation, Learning, and Optimization (12)) 2012th Edition

Reference Books:

1. Vincent François-Lavet, Peter Henderson, Riashat Islam, An Introduction to Deep Reinforcement Learning (Foundations and Trends (r) in Machine Learning) , 2019.

III B. Tech II Semester	BIG DATA ANALYTICS	L	T	P	C
--------------------------------	---------------------------	----------	----------	----------	----------



		3	0	0	3
--	--	----------	----------	----------	----------

COURSE OBJECTIVES:

- Optimize business decisions and create competitive advantage with Big Data analytics
- Introducing Java concepts required for developing map reduce programs
- Derive business benefit from unstructured data
- Imparting the architectural concepts of Hadoop and introducing map reduce paradigm
- To introduce programming tools PIG & HIVE in Hadoop echo system.

UNIT-I: Data structures in Java: Linked List, Stacks, Queues, Sets, Maps; Generics: Generic classes and Type parameters, Implementing Generic Types, Generic Methods, Wrapper Classes, Concept of Serialization

UNIT-II: Working with Big Data:Google File System, Hadoop Distributed File System (HDFS) Building blocks of Hadoop (Namenode, Datanode, Secondary Namenode, Job Tracker, Task Tracker), Introducing and Configuring Hadoop cluster (Local, Pseudo-distributed mode, Fully Distributed mode), Configuring XML files.

UNIT-III: Writing Map Reduce Programs: A Weather Dataset, Understanding Hadoop API for Map Reduce Framework (Old and New), Basic programs of Hadoop Map Reduce: Driver code, Mapper code, Reducer code, Record Reader, Combiner, Practitioner

UNIT-IV: StreamMemory and Spark: Introduction to Streams Concepts– Stream Data Model and Architecture , Stream computing, Sampling Data in a Stream , Filtering Streams , Counting Distinct Elements in a Stream , Introduction to Spark Concept , Spark Architecture and components , Spark installation , Spark RDD (Resilient Distributed Dataset) – Spark RDD operations.

UNIT-V: Pig: Hadoop Programming Made Easier Admiring the Pig Architecture, Going with the Pig Latin Application Flow, Working through the ABCs of Pig Latin, Evaluating Local and Distributed Modes of Running Pig Scripts, Checking out the Pig Script Interfaces, Scripting with Pig Latin.

Applying Structure to Hadoop Data with Hive:Saying Hello to Hive, Seeing How the Hive is Put Together, Getting Started with Apache Hive, Examining the Hive Clients, Working with Hive Data Types, Creating and Managing Databases and Tables, Seeing How the Hive Data Manipulation Language Works, Querying and Analysing data



TEXT BOOKS:

1. Wiley & Big Java 4th Edition, Cay Horstmann, Wiley John Sons, INC
2. Hadoop: The Definitive Guide by Tom White, 3rd Edition, O'reilly

REFERENCE BOOKS:

1. Hadoop in Action by Chuck Lam, MANNING Publ.
2. Hadoop for Dummies by Dirk deRoos, Paul C.Zikopoulos, Roman B.Melnyk, Bruce Brown, Rafael Coss
3. Hadoop in Practice by Alex Holmes, MANNING Publ.
4. Big Data Analytics by Dr. A.Krishna Mohan and Dr.E.Laxmi Lydia
5. Hadoop Map Reduce Cookbook, SrinathPerera, ThilinaGunarathne

Software Links:

1. Hadoop: <http://hadoop.apache.org/>
2. Hive: <https://cwiki.apache.org/confluence/display/Hive/Home>
3. Piglatin: <http://pig.apache.org/docs/r0.7.0/tutorial.html>



III B. Tech II Semester	DATA VISUALIZATION	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 visualization and 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

UNIT-1:Introduction: What Is Visualization?, History of Visualization, Relationship between Visualization and 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 B. Tech II Semester	CRYPTOGRAPHY & NETWORK SECURITY	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

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
Introduction to Cryptography with Coding Theory: Wade Trappe, Lawrence C.



III B. Tech II Semester	RECOMMENDER SYSTEMS	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

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 neighbour recommendation, Item-based nearest neighbour 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 based filtering, 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-Centred 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 Books:

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



III B. Tech 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, ISO 9000. 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, Software reverse engineering, Software maintenance process models and Estimation of maintenance cost.

Software Reuse: reuse-definition, introduction, reason behind reuse so far, Basic issues in any reuse program, A reuse approach, and Reuse at organization level.

Text Books:

1. Fundamentals of Software Engineering, Rajib Mall, 5th Edition, PHI.
2. Software Engineering A Practitioner's Approach, Roger S. Pressman, 9th Edition, McGraw Hill International Edition.

Reference Books:

1. Software Engineering, Ian Sommerville, 10th Edition, Pearson.
2. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.

e-Resources:

- 1) <https://nptel.ac.in/courses/106/105/106105182/>
- 2) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_0126058950_6387148827_shared/overview
- 3) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_0133826904_11003904735_shared/overview



III B. Tech II Semester	SOCIAL NETWORK ANALYSIS	L	T	P	C
		3	0	0	3

Unit-I: INTRODUCTION: Introduction to Semantic Web: Limitations of current Web – Development of Semantic Web – Emergence of the Social Web – Social Network analysis: Development of Social Network Analysis – Key concepts and measures in network analysis – Electronic sources for network analysis: Electronic discussion networks, Blogs and online communities – Web-based networks – Applications of Social Network Analysis

Unit-II: MODELLING, AGGREGATING AND KNOWLEDGE REPRESENTATION

Ontology and their role in the Semantic Web: Ontology-based knowledge Representation – Ontology languages for the Semantic Web: Resource Description Framework – Web Ontology Language – Modelling and aggregating social network data: State-of-the-art in network data representation – Ontological representation of social individuals – Ontological representation of social relationships – Aggregating and reasoning with social network data – Advanced representations.

Unit-III: EXTRACTION AND MINING COMMUNITIES IN WEB SOCIAL NETWORKS

Extracting evolution of Web Community from a Series of Web Archive – Detecting communities in social networks – Definition of community – Evaluating communities – Methods for community detection and mining – Applications of community mining algorithms – Tools for detecting communities social network infrastructures and communities – Decentralized online social networks – Multi-Relational characterization of dynamic social network communities.

Unit-IV: PREDICTING HUMAN BEHAVIOR AND PRIVACY ISSUES

Understanding and predicting human behavior for social communities – User data management – Inference and Distribution – Enabling new human experiences – Reality mining – Context – Awareness – Privacy in online social networks – Trust in online environment – Trust models based on subjective logic – Trust network analysis – Trust transitivity analysis – Combining trust and reputation – Trust derivation based on trust comparisons – Attack spectrum and counter measures.

Unit-V: VISUALIZATION AND APPLICATIONS OF SOCIAL NETWORKS

Graph theory – Centrality – Clustering – Node-Edge Diagrams – Matrix representation – Visualizing online social networks, visualizing social networks with matrix-based representations – Matrix and Node-Link Diagrams – Hybrid representations – Applications – Cover networks – Community welfare – Collaboration networks – Co-Citation networks.



TEXT BOOKS:

1. Peter Mika, —Social Networks and the Semantic Web, First Edition, Springer 2007.
2. Borko Furht, —Handbook of Social Network Technologies and Applications, 1st Edition, Springer, 2010.

REFERENCES:

1. Guandong Xu ,Yanchun Zhang and Lin Li,-Web Mining and Social Networking – Techniques and applications, First Edition, Springer, 2011.
2. Dion Goh and Schubert Foo,-Social information Retrieval Systems: Emerging Technologies and Applications for Searching the Web Effectively, IGI Global Snippet, 2008.
3. Max Chevalier, Christine Julien and Chantal Soulé-Dupuy, Collaborative and Social Information Retrieval and Access: Techniques for Improved user Modelling, IGI Global Snippet, 2009.
4. John G. Breslin, Alexander Passant and Stefan Decker, -The Social Semantic Web, Springer, 2009.



III B.Tech II Semester	COMPUTER VISION	L	T	P	C
		3	0	0	3

Course Objectives:

- To understand the fundamental concepts related to sources, shadows and shading
- To understand the geometry of multiple views

UNIT-I: CAMERAS: Pinhole Cameras Radiometry – Measuring Light: Light in Space, Light Surfaces, Important Special Cases Sources, Shadows, And Shading: Qualitative Radiometry, Sources and Their Effects, Local Shading Models, Application: Photometric Stereo, Interreflections: Global Shading Models Color: The Physics of Color, Human Color Perception, Representing Color, A Model for Image Color, Surface Color from Image Color.

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 Texture: Representing Texture, Analysis (and Synthesis) Using Oriented Pyramids, Application: Synthesis by Sampling Local Models, Shape from Texture.

UNIT-III: The Geometry of Multiple Views: Two Views Stereopsis: Reconstruction, Human Stereopsis, Binocular Fusion, Using More Cameras Segmentation by Clustering: What is Segmentation? Human Vision: Grouping and Gestalt, 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: Tracking as an Abstract Inference Problem, Linear Dynamic Models, Kalman Filtering, Data Association, Applications and Examples

UNIT-V: Geometric Camera Models: Elements of Analytical Euclidean Geometry, Camera Parameters and the Perspective Projection, Affine Cameras and Affine Projection Equations Geometric Camera Calibration: Least-Squares Parameter Estimation, A Linear Approach to Camera Calibration, Taking Radial Distortion into 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 B. Tech II Semester	CLOUD COMPUTING	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 Fundamentals:Cloud 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 of distributed computing, Inter-process communication, technologies for distributed computing, remote procedure calls (RPC), service-oriented architecture (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 and Kubernetes, 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 B. Tech II Semester	DevOps	L	T	P	C
		3	0	0	3

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

1. Describe the agile relationship between development and IT operations.
2. Understand the skill sets and high-functioning teams involved in DevOps and related methods to reach a continuous delivery capability.
3. Implement automated system update and DevOps lifecycle.

UNIT-II 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

Source Code Management(GIT):The need for source code control, The history 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, GIT staging and collaboration.**UNITTESTING-CODECOVERAGE:**Junit,nUnit & Code Coverage with Sonar Qube, SonarQube - Code Quality Analysis.

UNIT-III: Build Automation - Continuous Integration (CI):Build Automation, What isCI Why CI 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, JavaScript testing.

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.



List of Experiments:

1. Write code for a simple user registration form for an event.
2. Explore Git and GitHub commands.
3. Practice source code management on GitHub. Experiment with the source code written in exercise 1.
4. Jenkins installation and setup, explore the environment.
5. Demonstrate continuous integration and development using Jenkins.
6. Explore Docker commands for content management.
7. Develop a simple containerized application using Docker.
8. Integrate Kubernetes and Docker
9. Automate the process of running containerized application developed in exercise 7 using Kubernetes.
10. Install and explore Selenium for automated testing.
11. Write a simple program in JavaScript and perform testing using Selenium.
12. Develop test cases for the above containerized application using Selenium.

Text Books

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

Reference Books

1. Len Bass, Ingo Weber, Liming Zhu. DevOps: A Software Architect'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, Joakim Practical DevOps, 1st Edition, Packt Publishing, 2016.
4. Joakim Verona. Practical DevOps, Second Edition. Ingram short title; 2nd edition (2018). ISBN10: 1788392574
5. Deepak Gaikwad, Viral Thakkar. DevOps Tools from Practitioner's Viewpoint. Wiley publications. ISBN: 9788126579952



III B. Tech II Semester	SOFT COMPUTING	L	T	P	C
		3	0	0	3

Course Objectives:

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

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, Stopping condition for genetic algorithm flow, Genetic-neuro hybrid systems, Genetic Fuzzy rule based system

Text Books:

- S.N. Sivanandam and S.N. Deepa, Principles of soft computing – John Wiley & Sons, 2007.
- Timothy J. Ross, Fuzzy Logic with engineering applications, John Wiley & Sons, 2016.

Reference Books:

- N.K. Sinha and M.M. Gupta, Soft Computing & Intelligent Systems: Theory & Applications – Academic Press / Elsevier. 2009.
- Simon Haykin, Neural Network – A Comprehensive Foundation – Prentice Hall International, Inc. 1998
- R. Eberhart and Y. Shi, Computational Intelligence: Concepts to Implementation, Morgan Kaufman/Elsevier, 2007.
- Driankov D., Hellendoorn H. and Reinfrank M., An Introduction to Fuzzy Control – Narosa Pub., 2001.
- Bart Kosko, Neural Network and Fuzzy Systems – Prentice Hall, Inc., Englewood Cliffs, 1992
- Goldberg D.E., Genetic Algorithms in Search, Optimization, and Machine Learning – Addison Wesley, 1989



III B. Tech II Semester	BIG DATA ANALYTICS LAB	L	T	P	C
		0	0	3	1.5

Software Requirements:

1. **Hadoop:** <https://hadoop.apache.org/release/2.7.6.html>
2. **Java :** <https://www.oracle.com/java/technologies/javase/javase8u211-later-archive-downloads.html>
3. **Eclipse :** <https://www.eclipse.org/downloads/>

List of Experiments:

Experiment 1: Week 1, 2:

1. Implement the following Data structures in Java
a) Linked Lists b) Stacks c) Queues d) Set e) Map

Experiment 2: Week 3:

2. (i) Perform setting up and Installing Hadoop in its three operating modes:
Standalone, Pseudo distributed, Fully distributed
(ii) Use web based tools to monitor your Hadoop setup.

Experiment 3: Week 4:

3. Implement the following file management tasks in Hadoop:
 - Adding files and directories
 - Retrieving files
 - Deleting files

Hint: A typical Hadoop workflow creates data files (such as log files) elsewhere and copies them into HDFS using one of the above command line utilities.

Experiment 4: Week 5:

4. Run a basic Word Count MapReduce program to understand MapReduce Paradigm.

Experiment 5: Week 6:

5. Write a map reduce program that mines weather data. Weather sensors collecting data every hour at many locations across the globe gather a large volume of log data, which is a good candidate for analysis with Map Reduce, since it is semi structured and record-oriented.

Experiment 6: Week 7:

6. Use MapReduce to find the shortest path between two people in a
7. social graph.

Hint: Use an adjacency list to model a graph, and for each node store the distance from the original node, as well as a back pointer to the original node. Use the mappers to propagate the distance to the original node, and the reducer to restore the state of the graph. Iterate until the target node has been reached.

Experiment 7: Week 8:

8. Implement Friends-of-friends algorithm in MapReduce.

Hint: Two MapReduce jobs are required to calculate the FoFs for each user in a social network .The first job calculates the common friends for each user, and the second job sorts the common friends by the number of connections to your friends.



Experiment 8: Week 9:

9. Implement an iterative PageRank graph algorithm in MapReduce.

Hint: PageRank can be implemented by iterating a MapReduce job until the graph has converged. The mappers are responsible for propagating node PageRank values to their adjacent nodes, and the reducers are responsible for calculating new PageRank values for each node, and for re-creating the original graph with the updated PageRank values.

Experiment 9: Week 10:

1. Perform an efficient semi-join in MapReduce.

Hint: Perform a semi-join by having the mappers load a Bloom filter from the Distributed Cache, and then filter results from the actual MapReduce data source by performing membership queries against the Bloom filter to determine which data source records should be emitted to the reducers.

Experiment 10: Week 11:

2. Install and Run Pig then write Pig Latin scripts to sort, group, join, project, and filter your data.

Experiment 12: Week 12:

3. Install and Run Hive then use Hive to create, alter, and drop databases, tables, views, functions, and indexes



III B. Tech 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.
- Make use 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 daily 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 B. Tech II Semester	SOFT SKILLS	L	T	P	C
		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 B. Tech II Semester	TECHNICAL PAPER WRITING & IPR	L	T	P	C
		2	0	0	-

Course Objective:

1. 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”, 2nd Ed., 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/>