Syllabus
III B.Tech I SEM

INFORMATION RETRIEVAL SYSTEMS
CSE(AI&ML)

Subject Code: UGAI5T1123		**L**	**T**	**P**	**C**
III Year / I Semester		**3**	**0**	**0**	**3**

COURSE OBJECTIVE:

- To provide foundational knowledge of Information Retrieval (IR) systems, including their domain, structure, and evaluation techniques.
- To introduce data structures, indexing mechanisms, and text processing algorithms used in IR, such as inverted files, signature files, PAT trees, and string searching methods.
- To develop the ability to apply stemming, lexical analysis, and thesaurus construction techniques for improving information retrieval efficiency and accuracy.

SYLLABUS:

UNIT I: **(10 hrs)**

Introduction to Information storage and retrieval systems:
Domain Analysis of IR systems, IR and other types of Information Systems, IR System Evaluation

Introduction to Data structures and algorithms related to Information Retrieval: Basic Concepts, Data structures, Algorithms

UNIT II: **(12 hrs)**

Inverted Files and Signature Files Introduction, Structures used in Inverted Files, building an Inverted file using a sorted array, Modifications to the Basic Techniques.

Signature Files: Concepts of Signature files, Compression, Vertical Partitioning, Horizontal Partitioning.

UNIT III: **(10 hrs)**

New Indices for Text, Lexical Analysis and Stoplists: PAT Trees and PAT Arrays: Introduction, PAT Tree structure, Algorithms on the PAT Trees, Building PAT Trees as PATRICA Trees, PAT representation as Arrays. Stoplists

UNIT IV: **(08 hrs)**

Stemming Algorithms and Thesaurus Construction: Types of Stemming algorithms, Experimental Evaluations of Stemming, stemming to Compress Inverted Files.

Thesaurus Construction: Features of Thesauri, Thesaurus Construction, Thesaurus construction from Texts, Merging existing Thesauri.

UNIT V: **(08 hrs)**

String Searching Algorithms: Introduction, Preliminaries, The Naive Algorithm, The Knutt-Morris-Pratt Algorithm, The Boyer-Moore Algorithm, The Shift-Or Algorithm, The Karp-Rabin Algorithm.

Course Outcomes:

CO1	Explain the structure, domain, and evaluation of Information Retrieval systems along with related data structures.(L2)
CO2	Construct and modify indexing techniques using inverted and signature files for document retrieval.(L3)
CO3	Analyze the working of PAT trees, arrays, lexical analysis, and stoplist techniques in IR systems.(L4)
CO4	Evaluate stemming algorithms and demonstrate thesaurus construction techniques for semantic text retrieval.(L5)
CO5	Compare various string searching algorithms for efficient information retrieval.(L4)

Mapping CO's to PO's

POs/ COs	**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PSO 1**	**PSO 2**
CO1	3	2	–	–	–	–	–	–	–	–	–	2	-
CO2	3	3	3	–	2	–	–	–	–	–	–	3	3
CO3	2	3	3	2	2	–	–	–	–	–	–	3	2
CO4	2	2	2	3	–	–	–	–	–	–	–	2	3
CO5	2	3	2	3	3	–	–	–	–	–	2	2	3

TEXT BOOKS:

1.	Modern Information Retrieval,Ricardo Baeza-Yates, Neto, PEA,2007.
2.	Information Storage and Retrieval Systems: Theory and Implementation, Kowalski, Gerald, Mark Academic Press, 2000.

COMPUTER NETWORKS
(Common to CSE-AI&DS, CSE-AI&ML)

Subject Code: UGAI5T0223		**L**	**T**	**P**	**C**
III Year / I Semester		**3**	**0**	**0**	**3**

COURSE OBJECTIVE:

- To provide fundamental knowledge of computer networks, layered architecture, and communication protocols.
- To explain the functionalities of various network layers, including data link, network, transport, and application layers.
- To analyze protocols and services such as Ethernet, IP addressing, TCP, UDP, DNS, HTTP, FTP, and email.

SYLLABUS:

UNIT I: **(09 hrs)**

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.

UNIT II: **(12 hrs)**

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

UNIT III: **(10 hrs)**

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.

UNIT IV: **(08 hrs)**

The Transport Layer

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

UNIT V: **(09 hrs)**

The Application Layer

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

TEXT BOOKS:

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

Course Outcomes:

CO1	Summarize various network types, models, and historical aspects of Internet communication.(L2)
CO2	Apply error control techniques and data link layer protocols for reliable node-to-node communication.(L3)
CO3	Analyze IP addressing schemes and routing algorithms used at the network layer.(L4)
CO4	Evaluate the reliability and flow control mechanisms of transport layer protocols like TCP, UDP, and SCTP.(L5)
CO5	Analyze application layer protocols such as HTTP, DNS, FTP, Telnet, and Email in practical networking scenarios.(L4)

Mapping CO's to PO's

POs/ COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 1	PSO 2
CO1	3	2	–	–	–	–	–	–	–	–	–	2	-
CO2	3	3	2	–	3	–	–	–	–	–	–	3	3
CO3	3	3	3	2	3	–	–	–	–	–	–	3	3
CO4	3	3	3	3	3	–	–	–	–	–	2	2	3
CO5	2	3	2	2	2	–	–	–	–	–	2	2	2

OPERATING SYSTEMS
(CSE(AI&ML))

Subject Code: UGAI5T1223		**L**	**T**	**P**	**C**
III Year / I Semester		**3**	**0**	**0**	**3**

COURSE OBJECTIVE:

- o provide fundamental knowledge of operating system architecture, functions, and design strategies.
- To explore process, thread, memory, and file management mechanisms in modern operating systems.
- To analyze and apply various OS concepts like synchronization, deadlocks, virtual memory, and protection mechanisms in computing environments.

SYLLABUS:

UNIT I: **(10 hrs)**

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: **(10 hrs)**

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: **(09 hrs)**

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: **(09 hrs)**

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:	**(10 hrs)**
File System: File System Interface: File concept, Access methods, Directory Structure; File system Implementation: File-system structure, File-system Operations, Directory implementation, Allocation method, 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.	

Course Outcomes:

CO1	Outline the structure, functions, and services of modern operating systems and open-source OS environments.(L2)
CO2	Apply process management, multithreading, and CPU scheduling algorithms to improve system performance. (L3)
CO3	Analyze synchronization mechanisms and deadlock handling techniques for resource management. (L4)
CO4	Apply memory and storage management techniques for efficient use of hardware resources.(L3)
CO5	Evaluate file system structures and protection models to enhance data security and system reliability. (L5)

Mapping CO's to PO's

POs/ COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 1	PSO 2
CO1	3	2	–	–	–	–	–	–	–	–	–	-	-
CO2	3	3	3	–	2	–	–	–	–	–	–	2	3
CO3	3	3	3	3	3	–	–	–	–	–	2	3	3
CO4	3	3	2	–	3	–	–	–	–	–	–	2	3
CO5	2	3	3	–	2	2	–	–	–	–	2	2	2

TEXT BOOKS:

1.	Operating System Concepts, Silberschatz A, GalvinPB, GagneG,10thEdition, Wiley, 2018
2.	Modern Operating Systems, Tanenbaum AS,4th Edition, Pearson ,2016

REFERENCE BOOKS:

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

3.	J. Nilsson, "Artificial Intelligence: A new Synthesis", Elsevier Publishers.
4.	Artificial Intelligence, SarojKaushik, CENGAGE Learning.

ONLINE LEARNING RESOURCES:

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

AUTOMATA THEORY AND COMPILER DESIGN
(Professional Elective – I)
(Common to CSE-AI&DS, CSE-AI&ML)

Subject Code: UGAI5T0423		**L**	**T**	**P**	**C**
III Year / I Semester		**3**	**0**	**0**	**3**

COURSE OBJECTIVE:	
The learning objectives of this course are to: ➢ To introduce the theoretical foundations of automata, formal languages, and Turing machines. ➢ To apply the principles of syntax analysis and translation used in compiler design. ➢ To analyze decidability, parsing strategies, and intermediate code generation mechanisms.	
SYLLABUS:	
UNIT I:	**(10 hrs)**
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:	**(10 hrs)**
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:	**(10 hrs)**
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:	**(09 hrs)**
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	
UNIT V:	**(09 hrs)**
Syntax-Directed Translation: Syntax-Directed Definitions, Evaluation Orders for SDD's, Syntax Directed 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	

Course Outcomes:

CO1	Illustrate the concepts of finite automata, regular languages, and conversions between DFAs, NFAs.(L2)
CO2	apply regular expressions and context-free grammars to represent and validate languages.(L3)
CO3	Analyze computational models including PDAs and Turing Machines to understand language classifications and undecidability.(L4)
CO4	Apply lexical and syntax analysis techniques for language parsing and grammar processing.(L3)
CO5	Evaluate syntax-directed translation schemes, intermediate code, and run-time environment(L5)

Mapping CO's to PO's

POs/ COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 1	PSO 2
CO1	3	2	–	–	–	–	–	–	–	–	–	2	-
CO2	3	3	2	–	–	–	–	–	–	–	–	2	2
CO3	3	3	3	2	–	–	–	–	–	–	2	3	3
CO4	3	3	2	–	2	–	–	–	–	–	–	2	3
CO5	3	3	3	–	3	–	–	–	–	–	2	2	3

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

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

INTERNET OF THINGS
(Professional Elective – I)
(Common to CSE-AI&DS, CSE-AI&ML)

Subject Code: UGAI5T0623		**L**	**T**	**P**	**C**
III Year / I Semester		**3**	**0**	**0**	**3**

COURSE OBJECTIVE:

The learning objectives of this course are to:

- 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.

SYLLABUS:

UNIT I: **(10 hrs)**

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: **(10 hrs)**

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: **(09 hrs)**

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: **(09 hrs)**

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: **(10 hrs)**

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, 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, CunoPfister ,Oreilly

Course Outcomes:

CO1	Explain core IoT concepts, M2M communication, and major application-layer protocols (HTTP, FTP, Telnet, etc.) (L2)
CO2	Apply IoT architecture, layered design models, and communication technologies in real-world systems.(L3)
CO3	Analyze web and message communication protocols for connected devices. (L4)
CO4	Apply data acquisition and integration techniques for IoT-enabled business processes. (L3)
CO5	Evaluate cloud platforms, sensing technologies, and service models used in IoT deployments.(L5)

Mapping CO's to PO's

POs/ COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 1	PSO 2
CO1	3	2	–	–	–	–	–	–	–	–	–	2	-
CO2	3	3	3	–	2	–	–	–	–	–	–	3	3
CO3	3	3	3	2	3	–	–	–	–	–	2	3	3
CO4	2	3	2	–	2	2	–	–	–	–	–	2	3
CO5	2	3	3	–	3	2	–	–	–	–	2	2	3

EXPLORATORY DATA ANALYSIS USING PYTHON
(Professional Elective – I)
(Common to CSE-AI&DS, CSE-AI&ML)

Subject Code: UGAI5T0723		**L**	**T**	**P**	**C**
III Year / I Semester		**3**	**0**	**0**	**3**

COURSE OBJECTIVE:

The learning objectives of this 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.

SYLLABUS:

UNIT I: **(09 hrs)**

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: **(09 hrs)**

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
5. Case Study: Perform Exploratory Data Analysis with Personal Email Data

UNIT III: **(12 hrs)**

Data Transformation: Merging database-style data frames, 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: **(09 hrs)**

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: **(09 hrs)**

Model Development and Evaluation: Unified machine learning workflow, Data preprocessing, 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.
3. Case Study: Perform Exploratory Data Analysis with Wine Quality Dataset

Course Outcomes:

CO1	Explain the steps and tools in the exploratory data analysis process using Python libraries. (L2).
CO2	Analyze datasets using various visualization techniques to uncover patterns and relationships. (L3).
CO3	Analyze and transform datasets using merging, reshaping, and missing value handling methods. (L4).
CO4	Apply descriptive statistics and time series methods to extract insights from data.(L3)
CO5	Evaluate machine learning models using suitable metrics after data preprocessing and EDA.(L5)

Mapping CO's to PO's

POs/ COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 1	PSO 2
CO1	3	2	–	–	–	–	–	–	–	–	–	2	-
CO2	3	3	2	–	2	–	–	–	–	–	–	2	2
CO3	3	3	3	2	3	–	–	–	–	–	2	3	3
CO4	2	3	2	–	2	–	–	–	–	–	–	2	3
CO5	2	3	3	3	3	–	–	–	–	–	2	2	3

TEXT BOOKS

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

SOFTWARE ENGINEERING
(Professional Elective – I)

Subject Code: UGAI5T1323		**L**	**T**	**P**	**C**
III Year / I Semester		**3**	**0**	**0**	**3**

COURSE OBJECTIVE:

The learning objectives of this course are to:

- 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.

SYLLABUS:

UNIT I: **(09 hrs)**

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: **(10 hrs)**

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: **(12 hrs)**

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: **(08 hrs)**

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.SEICapability maturity model. Few other Important quality standards, and Six Sigma.
UNIT V: **(09 hrs)**
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 no reuse so far, Basic issues in any reuse program, A reuse approach, and Reuse at organization level

Course Outcomes:

CO1	Summarize the evolution of software engineering and distinguish between various software life cycle models. (L2)
CO2	Apply project estimation techniques and requirements gathering methods to develop a software requirements specification (SRS). (L3)
CO3	Apply software design principles including modularity, cohesion, and agile design approaches to create system models. (L3)
CO4	Analyze testing strategies, quality standards, and debugging practices to ensure software reliability and correctness. (L4)
CO5	Evaluate the role of CASE tools, maintenance models, and software reuse in improving productivity and sustainability. (L5)

Mapping CO's to PO's

POs/ COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 1	PSO 2
CO1	3	2	–	–	–	–	–	–	–	–	–	2	-
CO2	3	3	2	–	2	–	–	–	–	–	–	2	2
CO3	3	3	3	–	3	–	–	–	–	–	–	2	3
CO4	3	3	2	2	3	–	–	–	–	–	2	2	3
CO5	2	3	3	2	3	–	–	–	–	–	2	2	3

TEXT BOOKS:

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

REFERENCE BOOKS:

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

ONLINE LEARNING RESOURCES:

1.	https://nptel.ac.in/courses/106/105/106105182/
2.	https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_012605895063871488 27_shared/overview
3.	https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013382690411003904735_shared/overview

CLOUD COMPUTING
(Professional Elective – I)

Subject Code: UGAI5T1423		**L**	**T**	**P**	**C**
III Year / I Semester		**3**	**0**	**0**	**3**

COURSE OBJECTIVE:

The learning objectives of this course are to:

- 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

SYLLABUS:

UNIT I: **(10 hrs)**

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 App Engine).

UNIT II: **(10 hrs)**

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: **(10 hrs)**

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: **(09 hrs)**

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:	**(09 hrs)**
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.	

Course Outcomes:

CO1	Summarize cloud computing architecture, service models (IaaS, PaaS, SaaS), deployment models, and cloud platforms. (L2)
CO2	Apply cloud-enabling technologies including distributed computing, SOA, and virtualization for scalable environments.(L3)
CO3	Apply containerization and orchestration tools such as Docker and Kubernetes for cloud-based deployment. (L3)
CO4	Analyze cloud interoperability, fault tolerance, energy efficiency, and security in various cloud environments. (L4)
CO5	Evaluate serverless platforms, cloud-IoT integration, edge/fog computing, DevOps, and quantum cloud architectures. (L5)

Mapping CO's to PO's

POs/ COs	**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PSO 1**	**PSO 2**
CO1	3	2	–	–	–	–	–	–	–	–	–	2	-
CO2	3	3	3	–	3	–	–	–	–	–	–	2	3
CO3	3	3	3	–	3	–	–	–	–	–	–	2	3
CO4	3	3	3	2	2	2	–	–	–	–	2	2	3
CO5	3	3	3	2	3	2	–	–	–	–	2	2	3

TEXT BOOKS:

1.	Mastering Cloud Computing, 2nd edition, RajkumarBuyya, Christian Vecchiola, ThamaraiSelvi, ShivanandaPoojara, 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)

INFORMATION RETRIEVAL LAB

Subject Code: UGAI5P1523		L	T	P	C
III Year / I Semester		0	0	3	1.5

COURSE OBJECTIVE:

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

- Compute the similarity bet ween text documents
- Apply all pre-processing steps for text-data
- Implement classification of text documents.
- Perform document clustering using different algorithms.
- Implement PageRank algorithm for any network.

Programming Language: Python/R

Sample Experiments:

1. Representation of a Text Document in Vector Space Model and Computing Similarity between two documents.
2. Pre-processing of a Text Document: stop word removal and stemming
3. Construction of an Inverted Index for a given document collection comprising of at least 50 documents with a total vocabulary size of at least 1000 words.
4. Classification of a set of Text Documents into known classes (You may use any of the Classification algorithms like Naive Bayes, Max Entropy, Rochio's, Support Vector Machine). Standard Datasets will have to be used to show the results.
5. Text Document Clustering using K-means. Demonstrate with a standard dataset and compute performance measures- Purity, Precision, Recall and F-measure.
6. Crawling/ Searching the Web to collect news stories on a specific topic (based on user input). The program should have an option to limit the crawling to certain selected websites only.
7. To parse XML text, generate Web graph and compute topic specific page rank
8. Implement Matrix Decomposition and LSI for a standard dataset.
9. Mining Twitter to identify tweets for a specific period (and/or from a geographical location) and identify trends and named entities.
10. Implementation of PageRank on Scholarly Citation Network.

Course Outcomes:

CO1	Apply text preprocessing, vectorization, and inverted indexing techniques to represent and compare documents. (L3)
CO2	Implement classification and clustering algorithms on text datasets and analyze their performance using standard metrics. (L3)
CO3	Apply crawling, XML parsing, and graph-based methods to compute PageRank and extract structured web data. (L4).
CO4	Evaluate advanced techniques like LSI and Twitter mining for trend analysis and topic modeling in real-world datasets. (L5).

Mapping CO's to PO's:

POs/ COs	**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PSO 1**	**PSO 2**
CO1	3	2	2	–	3	–	–	–	–	–	–	2	2
CO2	3	3	3	2	3	–	–	–	–	–	2	3	3
CO3	3	3	2	–	3	–	–	–	–	–	2	2	3
CO4	3	3	3	2	3	–	–	–	–	–	2	2	3

COMPUTER NETWORKS LAB
(Common to CSE-AI&DS, CSE-AI&ML)

Subject Code: UGAI5P0923		**L**	**T**	**P**	**C**
III Year / I Semester		**0**	**0**	**3**	**1.5**

COURSE OBJECTIVE:

The main objectives of the course are to

- To implement data link layer protocols and error control mechanisms in networking.
- To simulate flow control, congestion control, and routing protocols.
- To analyze network traffic using packet analyzers and security tools.
- To evaluate network performance metrics using simulators.

Sample 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.

Course Outcomes:

CO1	Apply framing and error control techniques such as character stuffing, CRC, and Hamming code. (L3)
CO2	Simulate and analyze flow control and congestion control protocols in network communication.(L4)
CO3	Implement and analyze routing algorithms like Dijkstra and Distance Vector for shortest path and broadcasting. (L4)
CO4	Evaluate network behavior using packet analyzers (Wireshark), scanners (Nmap), and simulators (NS2).

Mapping CO's to PO's

POs/ COs	**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PSO 1**	**PSO 2**
CO1	3	2	2	–	3	–	–	–	–	–	–	2	-
CO2	3	3	3	2	3	–	–	–	–	–	2	2	3
CO3	3	3	3	2	3	–	–	–	–	–	2	2	3
CO4	3	3	3	2	3	–	–	–	–	–	3	2	3

FULL STACK DEVELOPMENT – 2
(Common to CSE, CSE-AI&DS, CSE-AI&ML)

Subject Code: UGCS5K1023		**L**	**T**	**P**	**C**
III Year / I Semester		**0**	**1**	**2**	**2**

COURSE OBJECTIVE:

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:

- Typescript
  - 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. Typescript
   a. Write a program to understand simple and special types.
   b. Write a program to understand function parameter and return types.
   c. Write a program to show the importance with Arrow function. Use optional, default and REST parameters.
   d. Write a program to understand the working of typescript with class, constructor, properties, methods and access specifiers.
   e. Write a program to understand the working of namespaces and modules.
   f. Write a program to understand generics with variables, functions and constraints.
2. Express JS – 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.
3. Express JS – Templating, Form Data
   a. Write a program using templating engine.
   b. Write a program to work with form data.

4. Express JS – Cookies, Sessions, Authentication
   a. Write a program for session management using cookies and sessions.
   b. Write a program for user authentication.
5. 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.
6. 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).
7. 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.
8. 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.
9. 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.
10. 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.
11. ReactJS Applications – To-do list and Quiz
   a. Design to-do list application.
12. 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()
13. 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().
14. 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.

Course Outcomes:

CO1	Apply TypeScript language features to develop modular and strongly typed web programs. (L3)
CO2	Develop server-side applications using ExpressJS with routing, authentication, sessions, and MongoDB integration.(L3)
CO3	Build responsive client-side applications using ReactJS with routing, components, forms, and hooks. (L3)

CO4	Design and deploy full-stack applications integrating TypeScript, ExpressJS, ReactJS, and MongoDB. (L6)

Mapping CO's to PO's

POs/ COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 1	PSO 2
CO1	3	2	2	–	3	–	–	–	–	–	–	2	-
CO2	3	3	3	2	3	–	–	–	–	–	2	2	3
CO3	3	3	3	2	3	–	–	–	–	–	2	2	3
CO4	3	3	3	3	3	–	–	–	2	2	3	3	3

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

TINKERING LAB
(USER INTERFACE DESIGN USING FLUTTER)
(Common to CSE, IT, CSE-AI&DS, CSE-AI&ML)

Subject Code: UGCS5P1123		**L**	**T**	**P**	**C**
III Year / I Semester		**0**	**0**	**3**	**1.5**

COURSE OBJECTIVE:

The main objectives of the course are to

- 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.

Course Outcomes:

CO1	Apply Dart fundamentals and Flutter widgets to build user interfaces. (L3).
CO2	Implement responsive design, screen navigation, and state management in Flutter apps. (L3).
CO3	Create dynamic Flutter apps using custom widgets, themes, forms, and animations. (L4)
CO4	Integrate REST APIs and evaluate app performance through testing and debugging tools. (L5)

Mapping CO's to PO's

POs/ COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 1	PSO 2
CO1	3	2	2	–	3	–	–	–	–	–	–	2	2
CO2	3	3	3	2	3	–	–	–	–	–	2	2	3
CO3	3	3	3	2	3	–	–	–	–	–	3	2	3
CO4	3	3	3	2	3	–	–	–	–	–	3	2	3

Text Books:

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.

Syllabus
III B.Tech II SEM

NATURAL LANGAUGE PROCESSING
(Common to CSE, IT, CSE-AI&ML)

Subject Code: UGCS6T1023		**L**	**T**	**P**	**C**
III Year / II Semester		**3**	**0**	**0**	**3**

COURSE OBJECTIVE:

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

SYLLABUS:

UNIT I: **(10 hrs)**

INTRODUCTION: Origins and challenges of NLP – Language Modeling: 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: **(10 hrs)**

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: **(10 hrs)**

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: **(09 hrs)**

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: **(09 hrs)**
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).

COURSE OUTCOMES:

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

CO1:	Understand the fundamentals of natural language processing, including language models, morphology, and tokenization.
CO2:	Apply POS tagging, syntactic parsing, and semantic analysis techniques to process natural language text.
CO3:	Analyze ambiguity, grammar rules, feature structures, and probabilistic parsing approaches in NLP.
CO4:	Apply discourse analysis, word sense disambiguation techniques, and NLP lexical resources for solving complex language tasks.

Mapping CO's to PO's:

POs/ COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 1	PSO 2
CO1	3	2	-	-	-	-	-	-	-	-	3	2	-
CO2	3	-	3	-	3	-	-	-	-	-	-	2	3
CO3	3	3	-	3	3	2	-	-	-	-	3	2	3
CO4	3	2	3	-	3	-	-	-	-	2	2	2	3

TEXT BOOKS:

1.	Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech, 2nd Edition, 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, 1st Edition, Breck Baldwin, Atlantic Publisher, 2015.
2.	Natural Language Processing with Java, 2nd Edition, Richard M Reese, OReilly Media,2015.
3.	Handbook of Natural Language Processing, Second, Nitin Indurkhya and Fred J. Damerau, Chapman and Hall/CRC Press, 2010.Edition
4.	Natural Language Processing and Information Retrieval, 3rd Edition, Tanveer Siddiqui, U.S. Tiwary, Oxford University Press,2008

DEEP LEARNING

Subject Code: UGAI6T1323		L	T	P	C
III Year / II Semester		3	0	0	3

COURSE OBJECTIVE:	
➢ 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.	
SYLLABUS:	
UNIT I:	**(09 hrs)**
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:	**(10 hrs)**
Feed forward Networks-Multilayer Perceptron, Gradient Descent, Backpropagation, Empirical Risk Minimization, regularization, auto encoders**.** **Deep Neural Networks:** Difficulty of training deep neural networks, Greedylayer wise training.	
UNIT III:	**(10 hrs)**
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).	
UNIT IV:	**(10 hrs)**
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:	**(09 hrs)**
Recent trends-Variational Auto encoders, Transformers, GPT Applications: Vision, NLP, Speech	

COURSE OUTCOMES:

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

CO1:	Understand the basic building blocks of neural networks, including perceptrons, thresholding logic, and the perceptron learning algorithm.(L2)
CO2:	Apply multilayer perceptrons, backpropagation, and deep learning techniques for neural network training. (L3)
CO3:	Analyze the impact of optimization algorithms and regularization strategies on the performance of deep neural networks. (L4)
CO4:	Evaluate the effectiveness of recurrent and convolutional neural network architectures in modeling sequential and spatial data.(L5)
CO5:	Analyze advanced models like Variational Autoencoders, Transformers, and GPT in deep learning applications(L4)

Mapping CO's to PO's

POs/ COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 1	PSO 2
CO1	3	2	–	–	2	–	–	–	–	–	–	2	-
CO2	3	3	2	–	3	–	–	–	–	–	2	2	3
CO3	3	3	3	2	3	–	–	–	–	–	2	2	3
CO4	3	3	3	2	3	–	–	–	–	–	2	2	3
CO5	3	3	3	2	3	–	–	–	–	–	2	2	3

TEXT BOOKS:

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

REFERENCE BOOKS:

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

DATA VISUALIZATION
(Common to CSE-AI&DS, CSE-AI&ML)

Subject Code: UGAI6T0223		**L**	**T**	**P**	**C**
III Year / II Semester		**3**	**0**	**0**	**3**

COURSE OBJECTIVE:	
The objectives of the course is to ➢ 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	
SYLLABUS:	
UNIT I:	**(10 hrs)**
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:	**(09 hrs)**
Creating visual representations, visualization reference model, visual mapping, visual analytics, Design of visualization applications	
UNIT III:	**(10 hrs)**
Classification of visualization systems, Interaction and visualization techniques misleading, Visualization of one, two and multi-dimensional data, text and text documents	
UNIT IV:	**(09 hrs)**
Visualization of groups, trees, graphs, clusters, networks, software, Metaphorical visualization	
UNIT V:	**(10 hrs)**
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	

COURSE OUTCOMES:

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

CO1:	Understand the principles of visual perception, the visualization process, and their relation to data representation.(L2)
CO2:	Apply visualization techniques and mapping models to build effective visual representations for various data. (L3)
CO3:	Apply appropriate techniques for visualizing one-dimensional to multi-dimensional and textual data.(L3)
CO4:	Analyze tree, graph, network, and metaphor-based visualization approaches for complex data.(L4)
CO5:	Evaluate modern visualization tools and techniques for volumetric, spatial, and GIS data with recent trends.(L5)

Mapping CO's to PO's

POs/ COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 1	PSO 2
CO1	3	2	–	–	–	–	–	–	–	–	–	2	
CO2	3	3	3	–	2	–	–	–	–	–	–	2	3
CO3	3	3	3	–	3	–	–	–	–	–	2	2	3
CO4	3	3	3	2	3	–	–	–	–	–	2	2	3
CO5	3	3	3	2	3	2	–	–	–	–	3	2	3

TEXT BOOKS:

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

SOFTWARE TESTING METHODOLOGIES
(Professional Elective-II)
(Common to CSE, IT, CSE-AI&DS, CSE-AI&ML)

Subject Code: UGCS6T0423		**L**	**T**	**P**	**C**
III Year / II Semester		**3**	**0**	**0**	**3**

COURSE OBJECTIVE:	
The main objectives of the course is to ➢ 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	
SYLLABUS:	
UNIT I:	**(10 hrs)**
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:	**(10 hrs)**
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:	**(10 hrs)**
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:	**(09 hrs)**
State, State Graphs and Transition testing: state graphs, good & bad state graphs, state testing, Testability tips.	
UNIT V:	**(09 hrs)**
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).	

COURSE OUTCOMES:

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

CO1:	Understand the principles, models, and taxonomy of bugs in software testing.(L2)
CO2:	Apply various software testing techniques such as path testing, data flow, domain, and state-based testing. (L3)
CO3:	Analyze different test strategies using graphs, logic-based testing, and transition diagrams. (L4)
CO4:	Apply automated testing tools to improve software quality and reliability.(L3)

Mapping CO's to PO's

POs/ COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 1	PSO 2
CO1	3	2	-	-	-	2	-	-	-	-	3	-	-
CO2	3	3	2	-	3	2	-	-	-	-	3	-	-
CO3	2	3	2	3	2	-	2	-	-	-	3	-	-
CO4	3	2	3	-	3	-	-	-	-	-	2	-	-

TEXT BOOKS:

1.	1. Software Testing techniques - BarisBeizer, Dreamtech, second edition
2.	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

CRYPTOGRAPHY & NETWORK SECURITY
(Professional Elective-II)
(Common to CSE-AI&DS, CSE-AI&ML)

Subject Code: UGAI6T0323		**L**	**T**	**P**	**C**
III Year / II Semester		**3**	**0**	**0**	**3**

COURSE OBJECTIVE:

The main objectives of the course is to

- 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 LayerProtocols Enhanced security mechanisms.

SYLLABUS:

UNIT I: **(09 hrs)**

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: **(09 hrs)**

Introduction to Symmetric Cryptography: Algebraic Structures-Groups, Rings, Fields, GF( ) fields, Polynomials.

Mathematics of Asymmetric cryptography: Primes, checking for Primness, Euler's phi-functions, Fermat's Little Theorem, Euler's Theorem, Generating Primes, Primality Testing, Factorization, Chinese Remainder Theorem, Quadratic Congruence, Exponentiation and Logarithm.

UNIT III: **(11 hrs)**

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: **(10 hrs)**

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: **(09 hrs)**
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**.**

COURSE OUTCOMES:

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

CO1:	Describe fundamental concepts in network security and classical encryption techniques.(L2)
CO2:	Apply mathematical foundations to implement symmetric and asymmetric cryptographic algorithms.(L2)
CO3:	Apply encryption and decryption techniques using symmetric key and public key cryptosystems.(L3)
CO4:	Analyze cryptographic hash functions, message authentication codes, and digital signature schemes.(L4)
CO5:	Evaluate security protocols for web, transport, IP, and email communications.(L5)

Mapping CO's to PO's

POs/ COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 1	PSO 2
CO1	3	2	–	–	–	–	–	–	–	–	–	2	-
CO2	3	3	2	–	3	–	–	–	–	–	–	2	3
CO3	3	3	3	–	3	–	–	–	–	–	2	2	3
CO4	3	3	3	2	3	–	–	–	–	–	2	2	3
CO5	3	3	3	2	3	2	–	–	–	–	3	2	3

TEXT BOOKS:

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

REFERENCE BOOKS:

1.	Cryptography and Network Security: AtulKahate, 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

DEVOPS

(Professional Elective-II)

(Common to CSE, IT, CSE-AI&DS, CSE-AI&ML)

Subject Code: UGCS6T0623		**L**	**T**	**P**	**C**
III Year / II Semester		**3**	**0**	**0**	**3**

COURSE OBJECTIVE:	
The main objectives of the course is to ➢ Describe the agile relationship between development and IT operations. ➢ Understand the skill sets and high-functioning teams involved in DevOps and related methods to reach a continuous delivery capability. ➢ Implement automated system update and DevOps lifecycle.	
SYLLABUS:	
UNIT I:	**(10 hrs)**
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:	**(10 hrs)**
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, n Unit& Code Coverage with Sonar Qube, SonarQube - Code Quality Analysis**.**	
UNIT III:	**(10 hrs)**
Build Automation - Continuous Integration (CI):Build Automation, what is CI Why Cl is Required, CI tools, Introduction to Jenkins (With Architecture), Jenkins workflow, Jenkins master slave architecture, Jenkins Pipelines, **PIPELINE BASICS** - Jenkins Master, Node, Agent, and Executor Freestyle Projects& Pipelines, Jenkins for Continuous Integration, Create and Manage Builds, User Management in Jenkins Schedule Builds, Launch Builds on Slave Nodes.	
UNIT IV:	**(09 hrs)**
Continuous Delivery: Importance of Continuous Delivery, CONTINUOUS DEPLOYMENT CD Flow, Containerization with Docker: Introduction to Docker, Docker installation, Docker commands, Images & Containers, Docker File, running containers, working with containers and publish to DockerHub. **Testing Tools:** Introduction to Selenium and its features, Java Scripttesting	

UNIT V:	**(09 hrs)**
Configuration Management - ANSIBLE: Introduction to Ansible, Ansibletasks Roles, Jinja2 templating, Vaults, Deployments using Ansible**.** **CONTAINERIZATION USING KUBERNETES(OPENSHIFT):** Introduction to Kubernetes Namespace& Resources, CI/CD - On OCP, BC, DC & ConfigMaps, Deploying Apps on Openshift Container Pods. Introduction to Puppet master and 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. Developtestcasesfortheabovecontainerizedapplicationusingselenium

COURSE OUTCOMES:

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

CO1:	Understand DevOps concepts, lifecycle, and automation principles involved in modern software delivery.
CO2:	Apply tools like Git, Jenkins, Docker, and SonarQube to automate the build, test, and integration processes.
CO3:	Analyze CI/CD pipelines, containerization, configuration management, and deployment processes for efficiency and scalability.
CO4:	Apply DevOps practices in real-world projects to improve software quality, delivery speed, and collaboration

Mapping CO's to PO's

POs/ COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 1	PSO 2
CO1	3	2	-	-	-	-	-	-	-	-	3	2	2
CO2	3	-	3	-	3	-	-	-	-	-	-	3	3
CO3	3	3	3	3	3	2	-	-	-	-	3	3	3
CO4	3	2	3	-	3	-	-	-	3	2	2	3	3

TEXT BOOKS:

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

REFERENCE BOOKS:

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, Joakim Practical DevOps,1stEdition, Packt Publishing,2016.
4.	Joakim Verona. Practical Devops,2ndEdition.Ingramshorttitle;2ndedition (2018). ISBN10: 1788392574
5.	Deepak Gaikwad, Viral Thakkar. DevOps Tools from Practitioner's Viewpoint.Wiley publications.ISBN:9788126579952

RECOMMENDER SYSTEMS
(Professional Elective-II)
(Common to CSE-AI&DS, CSE-AI&ML)

Subject Code: UGAI6T0423		**L**	**T**	**P**	**C**
III Year / II Semester		**3**	**0**	**0**	**3**

COURSE OBJECTIVE:	
The main objectives of the course is to ➢ This course covers the basic concepts of recommender systems, including personalization algorithms, evaluation tools, and user experiences.	
SYLLABUS:	
UNIT I:	**(09 hrs)**
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:	**(09 hrs)**
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:	**(12 hrs)**
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	
UNIT IV:	**(08 hrs)**
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:	**(10 hrs)**
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	

COURSE OUTCOMES:

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

CO1:	Understand the fundamental concepts, mathematical background, and real-world applications of recommender systems.
CO2:	Apply collaborative filtering and content-based recommendation algorithms to suggest personalized items to users.
CO3:	Analyze hybrid recommender system designs and compare various strategies for recommendation enhancement.
CO4:	Evaluate recommender systems using different metrics and critically assess their role in communities and personalized systems.

Mapping CO's to PO's

POs/ COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 1	PSO 2
CO1	3	2	–	–	–	–	–	–	–	–	1	3	2
CO2	2	–	2	–	3	–	–	–	–	–	2	3	3
CO3	2	3	–	2	3	–	–	–	–	–	2	3	3
CO4	2	–	3	3	3	–	–	–	–	2	2	3	3

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.

REFERENCE BOOKS

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

SOFTWARE PROJECT MANAGEMENT

(Professional Elective-III)

(Common to CSE, IT, CSE-AI&DS, CSE-AI&ML)

Subject Code: UGCS6T0823		**L**	**T**	**P**	**C**
III Year / II Semester		**3**	**0**	**0**	**3**

COURSE OBJECTIVE:

The main objectives of the course is to

- To describe and determine the purpose and importance of project management from the perspectives of planning, tracking and completion of project
- To compare and differentiate organization structures and project structures
- To implement a project to manage project schedule, expenses and resources with the application of suitable project management tools.

SYLLABUS:

UNIT I: **(10 hrs)**

Conventional Software Management: The waterfall model, conventional software Management performance.

Evolution of Software Economics: Software Economics, pragmatic software cost estimation.

Improving Software Economics: Reducing Software product size, improving software processes, improving team effectiveness, improving automation, Achieving required quality, peer inspections**.**

The old way and the new: The principles of conventional software Engineering, principles of modern software management, transitioning to an iterative process

UNIT II: **(10 hrs)**

Life cycle phases: Engineering and production stages, inception, Elaboration, construction, transition phases. Artifacts of the process: The artifactsets, Management artifacts, Engineering artifacts, programm aticartifacts

UNIT III: **(10 hrs)**

Model based software architectures: A Management perspective and technical perspective.

Work Flows of the process: Software process work flows, Iteration work flows.

Check points of the process: Major milestones, Minor Mile stones, Periodic status assessments.

Iterative Process Planning: Work break down structures, planning guidelines, cost and schedule estimating, Iteration planning process, Pragmatic planning.	
UNIT IV:	**(09 hrs)**
Project Organizations and Responsibilities: Line-of-Business Organizations, Project Organizations, evolution of Organizations**.** **Process Automation:** Automation Building blocks, The Project Environment. Project Control and Process instrumentation: The seven core Metrics, Management indicators, quality indicators, life cycle expectations, pragmatic Software Metrics, Metrics automation	
UNIT V:	**(09 hrs)**
Agile Methodology, ADAPTing to Scrum, Patterns for Adopting Scrum, Iterating towards Agility. **Fundamentals of DevOps**: Architecture, Deployments, Orchestration, Need, Instance of applications, DevOps delivery pipeline, DevOp secosystem. DevOps adoption in projects: Technology aspects, Agiling capabilities, Tool stack implementation, Peopleaspect,processes	

COURSE OUTCOMES:

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

CO1:	Understand software project life cycles, economic aspects, and evolution of software management practices.
CO2:	Apply project planning techniques including workflows, milestones, and cost/schedule estimation.
CO3:	Analyze organizational roles, automation, metrics, and project control mechanisms in software projects.
CO4:	Apply modern methodologies such as Agile and DevOps for improved software project delivery.

Mapping CO's to PO's

POs/ COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 1	PSO 2
CO1	3	2	-	-	-	3	-	-	-	-	3	-	-
CO2	3	-	3	3	3	-	-	-	-	-	-	2	2
CO3	3	3	2	3	3	2	-	-	-	-	3	3	3
CO4	3	2	3	-	3	-	-	-	3	2	3	3	3

TEXT BOOKS

1.	Software Project Management, Walker Royce, PEA,2005.
2.	Succeeding with Agile: Software Development Using Scrum, Mike Cohn, Addison Wesley.

3.	The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations, Gene Kim, John Willis, Patrick Debois, Jez Humb,1st Edition, O'Reilly publications,2016

REFERENCE BOOKS

1.	Software Project Management, BobHughes,3/e,Mike Cotterell, TMH
2.	Software Project Management, Joel Henry, PEA
3.	Software Project Management in practice, PankajJalote, PEA, 2005,
4.	Effective Software Project Management, Robert K.Wysocki,Wiley,2006.
5.	Project Management in IT,Kathy Schwalbe, Cengage

MOBILE ADHOC AND SENSOR NETWORKS
(Professional Elective-III)

Subject Code: UGAI6T1423		**L**	**T**	**P**	**C**
III Year / II Semester		**3**	**0**	**0**	**3**

COURSE OBJECTIVE:	
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.	
SYLLABUS:	
UNIT I:	**(09 hrs)**
Introduction- Adhoc networks. Mobile Ad-Hoc networking with a View of 4G Wireless, Off-the-Shelf Enables of Ad Hoc, IEEE 802.11 in Ad Hoc Networks	
UNIT II:	**(09 hrs)**
Protocols, Performance and Open Issues, Scatternet Formation in Bluetooth Networks, Antenna Beamforming and Power Control for Ad Hoc Networks, Topology Control in Wireless Ad Hoc Networks, Broadcasting and Activity Scheduling in Ad Hoc Networks	
UNIT III:	**(12 hrs)**
Location Discovery, Routing Approaches in Mobile AdHoc Networks, Energy-Efficient Communication in AdHoc Wireless, AdHoc Networks Security, Self-Organized and Cooperative Ad Hoc Networking.	
UNIT IV:	**(09 hrs)**
Simulation and Modeling of Wireless, Mobile, and AdHoc Networks, Modeling Cross-Layering Interaction Using Inverse Optimization Algorithmic Challenges in Ad Hoc Networks.	
UNIT V:	**(09 hrs)**
Sensor Networks: Introduction to sensor network, Unique constraints and challenges, Localization and Tracking, Networking Sensors, Infrastructure establishment, Sensor Tasking and Control, Sensor network databases, Sensor Network Platforms and tools, Industrial Applications and Research directions	

COURSE OUTCOMES:

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

CO1:	Understand the fundamental concepts and architectures of mobile ad hoc networks and IEEE 802.11 standards.(L2)
CO2:	Apply suitable protocols and power control techniques to optimize performance in ad hoc networks.(L3)
CO3:	Analyze core protocols—routing, topology control, beamforming, scheduling, and power management—for reliable, energy-efficient operation. (L4)
CO4:	Design protocol mechanisms and strategies to achieve reliable, self-organizing, energy-efficient ad hoc and sensor-network solutions. (L6)
CO5:	Evaluate security, localization, and application-specific challenges and select suitable mechanisms to build industrial-grade cooperative networks. (L5)

Mapping CO's to PO's

POs/ COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 1	PSO 2
CO1	3	2	-	-	-	-	-	-	-	-	2	-	-
CO2	3	3	2	-	3	-	-	-	-	-	2	-	-
CO3	3	3	3	2	3	-	-	-	-	-	2	-	-
CO4	3	2	3	3	3	-	-	2	2	2	3	-	-
CO5	3	3	2	2	3	2	2	-	2	-	3	-	-

TEXT BOOKS

1.	Mobile Adhoc Networks–Aggelou, George(McGraw-Hill).
2.	Mobile Adhoc Networking –Stefano Basagni (Editor), Marco Conti(Editor), Silvia Giordano(Editor),Ivan Stojmenovi&Cacute (Editor) (Wiley-IEEE Press).

REFERENCE BOOKS

1.	Wireless Sensor Networks: An Information Processing Approach–Feng Zhao,Leonidas Guibas(Elsevier).
2.	Hand book of Sensor Networks: Algorithms and Architectures–Ivan Stojmenovi& Cacute(Wiley

COMPUTER VISION
(Professional Elective-III)
(Common to CSE-AI&DS, CSE-AI&ML)

Subject Code: UGAI6T0623		**L**	**T**	**P**	**C**
III Year / II Semester		**3**	**0**	**0**	**3**

COURSE OBJECTIVE:

The main objectives of the course is to

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

SYLLABUS:

UNIT I: **(09 hrs)**

CAMERAS: Pinhole Cameras Radiometry –Measuring Light: Lightin Space, Light Surfaces, Important Special Cases Sources, Shadows, And Shading: Qualitative Radiometry, Sources and Their Effects, Local Shading Models, Application: Photometric Stereo, inter reflections: 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: **(08 hrs)**

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 from Texture.

UNIT III: **(12 hrs)**

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 Get stalt, Applications: Shot Boundary Detection and Background Subtraction, Image Segmentation by Clustering Pixels, Segmentation by Graph-Theoretic Clustering,

UNIT IV: **(09 hrs)**

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:	**(10 hrs)**
Geometric Camera Models: Elements of Analytical Euclidean Geometry, Camera Parameters and the Perspective Projection, Affine Cameras and Affine Projection EquationsGeometricCameraCalibration:Least-SquaresParameterEstimation, 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**.**	

COURSE OUTCOMES:

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

CO1:	Understand the fundamentals of image formation, radiometry, shading, and color perception relevant to computer vision.(L2)
CO2:	Apply linear filters, Fourier transforms, and edge detection techniques to extract features from images. (L3)
CO3:	Analyze stereo vision, segmentation by clustering, and multiple view geometry for scene understanding. (L4)
CO4:	Evaluate segmentation models and tracking algorithms using probabilistic inference techniques. (L5)
CO5:	Create solutions for camera calibration, model-based vision, and medical image registration using geometric models. (L6)

Mapping CO's to PO's

POs/ COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 1	PSO 2
CO1	3	2	-	-	2	-	-	-	-	-	-	3	2
CO2	3	3	2	-	3	-	-	-	-	-	-	3	2
CO3	3	3	3	2	3	-	-	-	-	-	-	3	3
CO4	3	3	3	3	3	-	-	-	-	-	-	3	3
CO5	3	3	3	3	3	2	-	2	2	2	2	3	3

TEXT BOOKS

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

REFERENCE BOOKS

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

NOSQL DATABASES
(Professional Elective-III)
(Common to CSE-AI&DS, CSE-AI&ML)

Subject Code: UGAI6T0523		**L**	**T**	**P**	**C**
III Year / II Semester		**3**	**0**	**0**	**3**

COURSE OBJECTIVE:

The main objectives of the course is to

- Distinguish the four NoSQL families—key-value, document, column-family, and graph—and compare their design trade-offs with traditional relational databases.
- Model, store, and query data using representative platforms (MongoDB, Cassandra/HBase, Riak, Neo4j), applying replication, sharding, and MapReduce for scalability and fault-tolerance.
- Select the appropriate NoSQL solution for real-world use-cases such as event logging, analytics, e-commerce, and social graphs, balancing consistency, availability, and performance.
- Build simple applications that integrate NoSQL stores with existing systems, demonstrating CRUD operations, indexing, and basic transaction or consistency management.

SYLLABUS:

UNIT I: **(08 hrs)**

Overview and History of NoSQL Databases. Definition of the Four Types of NoSQL Database, The Value of Relational Databases, Getting at Persistent Data, Concurrency, Integration, Impedance Mismatch, Application and Integration Databases, Attack of the Clusters, The Emergence of NoSQL, Key Points

UNIT II: **(10 hrs)**

Comparison of relational databases to new NoSQL stores, MongoDB, Cassandra, HBASE, Neo4j use and deployment, Application, RDBMS approach, Challenges NoSQL approach, Key-Value and Document Data Models, Column-Family Stores, Aggregate-Oriented Databases. Replication and sharding, Map Reduce on databases. Distribution Models, Single Server, Sharding, Master-Slave Replication, Peer-to-Peer Replication, Combining Sharding and Replication

UNIT III: **(12 hrs)**

NoSQL Key/Value databases using MongoDB, Document Databases, Document oriented Database Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Event Logging, Content Management Systems, Blogging Platforms, Web Analytics or Real-Time Analytics, E-Commerce Applications, Complex

Transactions Spanning Different Operations, Queries against Varying Aggregate Structure.
UNIT IV: **(08 hrs)**
Column-oriented NoSQL databases using Apache HBASE, Column-oriented NoSQL databases using Apache Cassandra, Architecture of HBASE, Column-Family Data Store Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Event Logging, Content Management Systems, Blogging Platforms, Counters, Expiring Usage
UNIT V: **(10 hrs)**
NoSQL Key/Value databases using Riak, Key-Value Databases, Key-Value Store, Key-Value Store Features, Consistency, Transactions, Query Features, Structure of Data, Scaling, Suitable Use Cases, Storing Session Information, User Profiles, Preferences, Shopping Cart Data, Relationships among Data, Multi operation Transactions, Query by Data, Operations by Sets. Graph NoSQL databases using Neo4, NoSQL database development tools and programming languages, Graph Databases, Graph Database. Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases

COURSE OUTCOMES:

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

CO1:	Understand the fundamental concepts, evolution, and types of NoSQL databases and compare them with relational models.(L2)
CO2:	Apply different NoSQL data models such as key-value, document, column-family, and graph databases to practical use cases.(L3)
CO3:	Analyze scalability, replication, and distribution strategies in NoSQL databases like MongoDB, Cassandra, and HBase.(L4)
CO4:	Evaluate and select appropriate NoSQL solutions based on consistency, availability, and scalability requirements.(L5)

Mapping CO's to PO's

POs/ COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 1	PSO 2
CO1	3	2	-	-	2	-	-	-	-	-	-	3	2
CO2	3	3	2	-	3	-	-	-	-	-	-	3	3
CO3	3	3	3	2	3	-	-	-	-	-	-	3	3
CO4	3	3	3	3	3	2	-	2	2	2	2	3	3

TEXT BOOKS

1.	Sadalage,P.&Fowler,NoSQLDistilled:ABriefGuidetotheEmergingWorldof PolyglotPersistence, Wiley Publications,1st Edition,2019

REFERENCE BOOKS

1.	https://www.ibm.com/cloud/learn/nosql-databases
2.	https://www.coursera.org/lecture/nosql-databases/introduction-to-nosql-VdRNp
3	https://www.geeksforgeeks.org/introduction-to-nosql/
4	https://www.javatpoint.com/nosql-databa

DEEP LEARNING LAB

Subject Code: UGAI6P1523		L	T	P	C
III Year / II Semester		0	0	3	1.5

COURSE OBJECTIVES:

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

Experiments

1. Implement multi-layer perceptron algorithm for MNIST Handwritten 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 class classification) using Reuters dataset.
4. Design a neural network for predicting house prices using Boston Housing Price dataset.
5. Build a Convolution Neural Network for MNIST Handwritten 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 hoten coding of words or characters.
9. Implement word embedding's for IMDB dataset.
10. Implement a Recurrent Neural Network for IMDB movie review classification problem.

COURSE OUTCOMES:

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

CO1:	Apply deep neural network architectures such as MLP and CNN to perform classification and regression tasks on real-world datasets. (L2)
CO2:	Analyze the impact of various deep learning hyperparameters and architectures on model performance across different domains (image, text, and numeric datasets).(L4)
CO3:	Evaluate pre-trained deep learning models (e.g., VGG16) and compare their performance with custom-built architectures for image classification tasks.(L5)
CO4:	Create deep learning pipelines using word embeddings and RNNs to solve natural language processing problems such as sentiment classification.(L6)

Mapping CO's to PO's

POs/ COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 1	PSO 2
CO1	3	2	3	–	3	–	–	–	–	–	2	3	3
CO2	3	3	3	2	2	–	–	–	–	–	2	3	3
CO3	2	3	2	3	3	–	–	–	–	–	2	3	3
CO4	3	2	3	3	3	–	–	–	–	–	3	3	3

TEXT BOOKS

1.	Reza ZadehandBharath Ram sundar, "Tensorflow for Deep Learning", O'Reilly publishers, 2018

WEB REFERENCES:

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

DATA VISUALIZATION LAB

Subject Code: UGAI6P1623		L	T	P	C
III Year / II Semester		0	0	3	1.5

COURSE OBJECTIVE:

The main objectives of the course are to

- 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.

Experiments

1. Load VADeaths(Death Rates in Virginia)dataset in R and visualize the data using different histograms.
2. Load air quality dataset in R and visualize La Guardia Airport's dialy maximum temperature using histogram.
3. Load AirPassengers dataset in R and visualize the data using line chart that shows increase in air passengers over given time period.

a) Load iris dataset in R, visualize the data using different Bar Charts and also demonstrate the use of stacked plots.

4. b)Load air quality dataset in R and visualize ozone concentration in air.

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.

COURSE OUTCOMES:

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

CO1:	Apply appropriate R functions and libraries to visualize univariate, bivariate, and multivariate datasets using histograms, bar charts, line charts, and scatter plots.(L3)
CO2:	Analyze real-world datasets using advanced visualization techniques such as box plots, hexbin plots, 3D plots, heatmaps, and correlograms. (L4)
CO3:	Evaluate and compare the effectiveness of various visualization methods for representing large, categorical, and temporal datasets.(L5)
CO4:	Create interactive map visualizations using libraries such as leaflet and maps in R to represent spatial/geographical data. (L6)

Mapping CO's to PO's

POs/ COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 1	PSO 2
CO1	3	2	3	–	3	–	–	–	–	–	2	2	2
CO2	3	3	3	2	3	–	–	–	–	–	2	3	2
CO3	2	3	3	2	3	–	–	–	–	–	2	3	3
CO4	3	2	3	2	3	3	–	–	–	–	3	3	3

WEB REFRENCES:

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

SOFT SKILLS
(Common to All)

Subject Code: UGBS6K0123		L	T	P	C
III Year / II Semester		0	1	2	2

COURSE OBJECTIVE:

The objectives of the course are to

- 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 .

Syllabus:

UNIT I: **(10 hrs)**

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: **(10 hrs)**

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: **(10 hrs)**

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

UNIT IV: **(09 hrs)**

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

UNIT V: **(09 hrs)**

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

COURSE OUTCOMES:

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

CO1:	Understand the fundamentals of verbal, non-verbal communication, and self-presentation skills
CO2:	Apply professional etiquette, team management, and leadership skills for workplace success.
CO3:	Apply written and spoken communication techniques for academic and career-oriented contexts.
CO4:	Analyze interpersonal behaviors and refine job-oriented skills like interviews and group discussions.

Mapping CO's to PO's

POs/ COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 1	PSO 2
CO1	-	-	-	-	-	-	-	3	3	-	3	-	-
CO2	-	-	-	-	-	-	-	3	-	3	3	-	-
CO3	-	-	-	-	-	-	-	3	3	-	3	-	-
CO4	-	-	-	3	-	-	2	3	3	3	3	-	-

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

TECHNICAL PAPER WRITING & IPR

Subject Code: UGAI6A1223		L	T	P	C
III Year / II Semester		2	0	0	-

COURSE OBJECTIVE:

The objectives of the course are to

- 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.

Syllabus:

UNIT I: **(06 hrs)**

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: **(06 hrs)**

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: **(06 hrs)**

Proofreading and summaries: Proofreading, summaries, Activities on summaries.

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

UNIT IV: **(08 hrs)**

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:	**(06 hrs)**
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	

COURSE OUTCOMES:

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

CO1:	Understand the principles of technical writing and structure of professional documents.(L2)
CO2:	Apply drafting techniques, formatting styles, and editing tools to create effective technical documents.(L3)
CO3:	Use word processor features for technical documentation, citations, and report generation.(L3)
CO4:	Analyze the fundamentals of Intellectual Property Rights (IPR) and assess their relevance in research and innovation.(L4)

Mapping CO's to PO's

POs/ COs	**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PSO 1**	**PSO 2**
CO1	3	3	-	-	-	2	3	-	3	-	3	-	-
CO2	-	3	-	-	3	-	-	-	3	2	3	-	-
CO3	-	2	-	2	3	-	-	-	2	-	3	3	-
CO4	-	-	3	3	-	2	3	-	-	-	3	3	-

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/