Semester VI

Course Code	Course Title	Credits	Lectures /Week
USCS601	**Data Science**	**2**	**3**

About the Course: This course introduces the field of Data Science, covering the fundamental concepts, techniques, and tools used for data analysis, machine learning, and data visualization. Students will learn how to preprocess and analyze data, build predictive models, evaluate model performance, and effectively communicate insights through visualizations. The course also explores data management principles and practices. Practical hands-on exercises and projects using popular Data Science libraries and technologies are included to reinforce the concepts learned.

Course Objectives:

- Understand the foundations and scope of Data Science, including its applications and comparison to related fields like Business Intelligence and Artificial Intelligence.
- Develop skills in data preprocessing, including cleaning, transforming, selecting, and merging data, to ensure data quality and suitability for analysis.
- Gain knowledge of machine learning algorithms and techniques, such as regression, classification, clustering, and ensemble learning, to build predictive models and make data-driven decisions.
- Learn how to evaluate and select models using appropriate evaluation metrics and cross-validation techniques to ensure reliable and robust model performance.
- Develop proficiency in data visualization techniques and tools to effectively communicate insights and tell compelling stories using data.

Learning Outcomes:

After successful completion of this course, students would be able to

- Apply data preprocessing techniques to clean and transform raw data, handle missing values and outliers, and merge datasets.
- Implement machine-learning algorithms to perform tasks such as regression, classification, clustering, and ensemble learning.
- Evaluate and compare different machine learning models using appropriate evaluation metrics and cross-validation techniques.
- Create informative and visually appealing data visualizations to communicate insights and patterns in data.
- Understand the principles and practices of data management, including data governance, data quality assurance, and data privacy considerations.

Unit	Topics	No of Lectures
I	**Introduction to Data Science and Data Preprocessing** **What is Data Science?:** Definition and scope of Data Science, Applications and domains of Data Science, Comparison with other fields like Business Intelligence (BI), Artificial Intelligence (AI), Machine Learning (ML), and	**15**

	Data Warehousing/Data Mining (DW-DM) **Data Types and Sources:** Different types of data: structured, unstructured, semi-structured, Data sources: databases, files, APIs, web scraping, sensors, social media **Data Preprocessing:** Data cleaning: handling missing values, outliers, duplicates, Data transformation: scaling, normalization, encoding categorical variables, Feature selection: selecting relevant features/columns, Data merging: combining multiple datasets **Data Wrangling and Feature Engineering:** Data wrangling techniques: reshaping, pivoting, aggregating, Feature engineering: creating new features, handling time-series data Dummification: converting categorical variables into binary indicators, Feature scaling: standardization, normalization **Tools and Libraries:** Introduction to popular libraries and technologies used in Data Science like Pandas, NumPy, Sci-kit Learn, etc.	
II	**Data Analysis and Machine Learning** **Exploratory Data Analysis (EDA):** Data visualization techniques: histograms, scatter plots, box plots, etc., Descriptive statistics: mean, median, mode, standard deviation, etc., Hypothesis testing: t-tests, chi-square tests, ANOVA, etc. **Introduction to Machine Learning:** Supervised learning: classification and regression, Unsupervised learning: clustering and dimensionality reduction, Bias-variance tradeoff, underfitting, and overfitting **Regression Analysis:** Simple linear regression, Multiple linear regression, Stepwise regression, Logistic regression for classification **Model Evaluation and Selection:** Techniques for evaluating model performance: accuracy, precision, recall, F1-score, Confusion matrix and ROC curve analysis, Cross-validation: k-fold cross-validation, stratified cross-validation, Hyperparameter tuning and model selection **Machine Learning Algorithms:** Decision Trees and Random Forests, Support Vector Machines (SVM), Artificial Neural Networks (ANN), Ensemble Learning: Boosting and Bagging, K-Nearest Neighbors (K-NN), Gradient Descent for optimization	**15**
III	**Model Evaluation, Data Visualization, and Management** **Model Evaluation Metrics:** Accuracy, precision, recall, F1-score, Area Under the Curve (AUC), Evaluating models for imbalanced datasets **Data Visualization and Communication:** Principles of effective data visualization, Types of visualizations: bar charts, line charts, scatter plots, etc. Visualization tools: matplotlib, seaborn, Tableau, etc. Data storytelling: communicating insights through visualizations **Data Management:** Introduction to data management activities, Data pipelines: data extraction, transformation, and loading (ETL), Data governance and data quality assurance, Data privacy and security considerations	**15**

Textbook(s):

1. Data Science from Scratch First Principles with Python- Joel Grus O'reilly, 2nd Edition
2. Advancing into Analytics From Excel to Python and R, George Mount, Oreilly, First Edition
3. Introduction to Machine Learning with Python, Andreas C. Muller, Sarah Guido, Oreilly, First Edition

Additional Reference(s):

1. Doing Data Science, Rachel Schutt and Cathy O'Neil, O'Reilly,2013
2. Mastering Machine Learning with R, Cory Lesmeister, PACKT Publication,2015
3. Hands-On Programming with R, Garrett Grolemund,1st Edition, 2014
4. An Introduction to Statistical Learning, James, G., Witten, D., Hastie, T., Tibshirani, R.,Springer,2015

Course Code	Course Title	Credits	Lectures /Week
USCSP601	**Data Science – Practical**	**1**	**3**
1	Introduction to Excel • Perform conditional formatting on a dataset using various criteria. • Create a pivot table to analyze and summarize data. • Use VLOOKUP function to retrieve information from a different worksheet or table. • Perform what-if analysis using Goal Seek to determine input values for desired output.		
2	Data Frames and Basic Data Pre-processing • Read data from CSV and JSON files into a data frame. • Perform basic data pre-processing tasks such as handling missing values and outliers. • Manipulate and transform data using functions like filtering, sorting, and grouping.		
3	Feature Scaling and Dummification • Apply feature-scaling techniques like standardization and normalization to numerical features. • Perform feature dummification to convert categorical variables into numerical representations.		
4	Hypothesis Testing • Formulate null and alternative hypotheses for a given problem. • Conduct a hypothesis test using appropriate statistical tests (e.g., t-test, chi-square test). • Interpret the results and draw conclusions based on the test outcomes.		
5	ANOVA (Analysis of Variance) • Perform one-way ANOVA to compare means across multiple groups. • Conduct post-hoc tests to identify significant differences between group means.		
6	Regression and Its Types • Implement simple linear regression using a dataset. • Explore and interpret the regression model coefficients and goodness-of-fit measures. • Extend the analysis to multiple linear regression and assess the impact of additional predictors.		
7	Logistic Regression and Decision Tree • Build a logistic regression model to predict a binary outcome. • Evaluate the model's performance using classification metrics (e.g., accuracy, precision, recall). • Construct a decision tree model and interpret the decision rules for classification.		

8	K-Means Clustering • Apply the K-Means algorithm to group similar data points into clusters. • Determine the optimal number of clusters using elbow method or silhouette analysis. • Visualize the clustering results and analyze the cluster characteristics.
9	Principal Component Analysis (PCA) • Perform PCA on a dataset to reduce dimensionality. • Evaluate the explained variance and select the appropriate number of principal components. • Visualize the data in the reduced-dimensional space.
10	Data Visualization and Storytelling • Create meaningful visualizations using data visualization tools • Combine multiple visualizations to tell a compelling data story. • Present the findings and insights in a clear and concise manner.

Course Code	Course Title	Credits	Lectures /Week
USCS602	**Cloud Computing and Web Services**	**2**	**3**

About the Course: The course "Cloud Computing and Web Services" provides an in-depth understanding of cloud computing fundamentals and web service technologies. Students will learn about different types of clouds, cloud deployment models, and cloud platforms. They will also explore key concepts of virtualization, security in cloud computing, and popular cloud computing platforms such as OpenStack and AWS. Through practical exercises and hands-on projects, students will gain the skills required to design, deploy, and manage cloud-based applications and services.

Course Objectives:

- Understand the basics of cloud computing, including types of clouds, deployment models, and essential characteristics of cloud platforms.
- Explore web services technologies such as SOAP and REST and understand their role in distributed computing and parallel computing.
- Gain proficiency in utilizing virtualization technologies, including creating virtual machines and managing virtualized environments using tools like KVM and oVirt.
- Explore and utilize popular cloud computing platforms such as OpenStack and AWS to architect, deploy, and manage cloud-based applications and services.
- Learn about cloud security fundamentals, including confidentiality, integrity, availability, and secure development practices.

Learning Outcomes:

After successful completion of this course, students would be able to

- Demonstrate a comprehensive understanding of cloud computing concepts, including different types of clouds and their characteristics.
- Implement and utilize web service technologies, such as SOAP and REST, to develop distributed and parallel computing applications.
- Design, deploy, and manage cloud-based applications and services using popular cloud computing platforms such as OpenStack and AWS.
- Apply secure development practices and implement cloud security policies to ensure the confidentiality, integrity, and availability of cloud software solutions.
- Utilize virtualization technologies to create and manage virtualized environments, considering the benefits and drawbacks of virtualization.

Unit	Topics	No of Lectures
I	**Cloud Computing Basics** **Web Services** – Distributed Computing, Parallel Computing, WSDL structure, SOAP- Structure of SOAP Message (In JAX-WS), SOAP Messaging Architecture, SOAP Header, Client-side SOAP Handler, REST- What is REST? HTTP methods, Java API for RESTful Web Services (JAX-RS)	**15**

	Virtulization:- Characteristics of Virtualized Environments.. Pros and Cons of Virtualization. Virtualization using KVM, Creating virtual machines, oVirt - management tool for virtualization environment.	
II	**Introduction to Cloud Computing:** Definition, Types of Clouds, Deployment of software solutions and web applications, Types of Cloud Platforms, Essential characteristics – On-demand self-service, Broad network access, Location independent resource pooling ,Rapid elasticity , Measured service, Comparing cloud providers with traditional IT service providers **Cloud Computing Software Security fundamentals:** Cloud Information Security Objectives, Confidentiality, Integrity, Availability, Cloud Security Services, Relevant Cloud Security Design Principles, Secure Cloud Software Requirements, Secure Development practices, Approaches to Cloud Software Requirement Engineering, Cloud Security Policy Implementation.	**15**
III	**Cloud Applications** **CloudSim:** Introduction to Simulator, understanding CloudSim simulator, CloudSim Architecture(User code, CloudSim, GridSim, SimJava) Understanding Working platform for CloudSim, **OpenStack:** Introduction to OpenStack, OpenStack test-drive, Basic OpenStack operations, OpenStack CLI and APIs, Tenant model operations, Quotas, Private cloud building blocks, Controller deployment, Networking deployment, Block Storage deployment, Compute deployment, deploying and utilizing OpenStack in production environments, Building a production environment, Application orchestration using OpenStack Heat **AWS:** Architecting on AWS, Building complex solutions with Amazon Virtual Private Cloud (Amazon VPC)	**15**

Textbook(s):

1. Java Web Services Up and Running 2nd edition, Martin Kalin, O'Reilly (2013)
2. Pro Power Shell for Amazon Web Services, Brian Beach, Apress, 2014
3. Enterprise Cloud Computing Technology, Architecture, Applications, Gautam Shroff, Cambridge University Press, 2010
4. Mastering Cloud Computing, Rajkumar Buyya, Christian Vecchiola, S Thamarai Selvi, Tata McGraw Hill Education Private Limited, 2013
5. OpenStack in Action, V. K. CODY BUMGARDNER, Manning Publications Co, 2016

Additional Reference(s):

1. OpenStack Essentials, Dan Radez, PACKT Publishing, 2015 2
2. OpenStack Operations Guide, Tom Fifield, Diane Fleming, Anne Gentle, Lorin Hochstein, Jonathan Proulx, Everett Toews, and Joe Topjian, O'Reilly Media, Inc., 2014
3. https://www.openstack.org

Course Code	Course Title	Credits	Lectures /Week
USCSP602	**Cloud Computing and Web Services – Practical**	**1**	**3**
1	Define a simple services like Converting Rs into Dollar and Call it from different platform like JAVA and .NET		
2	Create a Simple SOAP service.		
3	Create a Simple REST Service.		
4	Develop application to consume Google's search / Google's Map RESTful Web service.		
5	Installation and Configuration of virtualization using KVM.		
6	Develop application to download image/video from server or upload image/video to server using MTOM techniques		
7	Implement FOSS-Cloud Functionality VSI (Virtual Server Infrastructure) Infrastructure as a Service (IaaS), Storage		
8	Implement FOSS-Cloud Functionality - VSI Platform as a Service (PaaS),		
9	Using AWS Flow Framework develop application that includes a simple workflow. Workflow calls an activity to print hello world to the console. It must define the basic usage of AWS Flow Framework, including defining contracts, implementation of activities and workflow coordination logic and worker programs to host them		
10	Implementation of Openstack with user and private network creation.		

Course Code	Course Title	Credits	Lectures /Week
USCS6031	**Wireless Sensor Networks**	**2**	**3**

About the Course: This course provides a comprehensive understanding of Wireless Sensor Networks (WSNs) and their applications. It covers the fundamental concepts, architectural elements, advantages, and challenges of WSNs. Students will explore sensor node technology, network architecture, optimization goals, and design principles for WSNs. The course also delves into wireless transmission, telecommunication systems, and introduces the concepts of WSN operating systems and ad-hoc networks. Through practical examples and case studies, students will gain hands-on experience in medium access control protocols, routing strategies, transport control protocols, and WSN middleware architecture.

Course Objectives:

- Provide students with a comprehensive understanding of Wireless Sensor Networks (WSNs), including their basic architectural elements, advantages, and challenges.
- Introduce students to the key technologies and protocols used in WSNs, such as medium access control (MAC) protocols, routing strategies, and transport control protocols.
- Familiarize students with wireless transmission principles and telecommunication systems relevant to WSNs, including frequency, signals, antennas, and satellite systems.
- Develop students' practical skills in designing and implementing WSN solutions by exploring WSN operating systems, ad-hoc networks, and optimization goals.

Learning Outcomes:

After successful completion of this course, students would be able to

- Understand the fundamental concepts, architectural elements, and optimization goals of Wireless Sensor Networks (WSNs) and apply this knowledge to analyze and design WSN solutions.
- Evaluate and compare different medium access control protocols and routing strategies in WSNs, and make informed decisions to ensure efficient and reliable communication.
- Demonstrate knowledge of wireless transmission technologies, such as frequency, signals, antennas, and propagation, and analyze their impact on WSN performance.
- Assess the role of telecommunication systems, satellite, broadcast systems in WSNs, and understand their applications and implications for WSN deployments.

Unit	Topics	No of Lectures
I	**Introduction and Overview of Wireless Sensor Networks:** Basic Sensor Network Architectural Elements, Advantage and challenges, Applications, Sensor Node Technology, Sensor Taxonomy, WN Operating Environment, Radio Technology, Network architecture, Optimization goals and figures of merit, Design principles for WSNs, Service interfaces of WSNs, Gateway concepts. **Wireless Sensor Network Operating Systems and Ad-hoc Networks:** Overview of Wireless Sensor Network Operating Systems, Examples of WSN Operating Systems Ad-hoc Networks in Wireless Sensor Networks, Characteristics and Challenges of Ad-hoc Networks in WSNs, Energy	**15**

	Efficiency Considerations in Ad-hoc Networks, Security and Privacy in Ad-hoc Networks, Examples of WSN OS, Ad-hoc Network.	
II	**Medium Access Control Protocol:** Fundamentals of MAC Protocols, Sensor-MAC Case Study **Routing in WSN:** Routing Challenges and Design Issues in Wireless Sensor Networks, , IEEE 802.15.4 LR-WPANs Standard Case Study, Routing Strategies in Wireless Sensor Networks, **Transport Control Protocol:** Traditional Transport Control Protocols, Transport Protocol Design Issues, WSN Middleware Architecture	**15**
III	**Wireless Transmission:** Frequency for radio transmission, Signals, Antennas, Signal propagation, Multiplexing, Modulation, Spread spectrum, Cellular systems. **Telecommunication, Satellite and Broadcast Systems:** Satellite and Broadcast Systems: GSM: Mobile services, System architecture, Radio interface, Protocols, Localization And Calling, Handover, security, New data services; DECT: System architecture, Protocol architecture; ETRA, UMTS and IMT- 2000. Satellite Systems: History, Applications, Basics: GEO, LEO, MEO; Routing, Localization, Handover.	**15**

Textbook(s):

1. Wireless Sensor Networks Technology, Protocols, and Applications ,Kazem Sohraby, Daniel Minoli and TaiebZnati, John Wiley & Sons, 2017
2. Protocols and Architectures for Wireless Sensor Network, Holger Kerl, Andreas Willig, John Wiley and Sons, 2015

Additional Reference(s):

1. Fundamentals of Wireless Sensor Networks, Theory and Practice, Waltenegus Dargie, Christian Poellabauer , Wiley Series on wireless Communication and Mobile Computing, 2011
2. Networking Wireless Sensors, Bhaskar Krishnamachari , Cambridge University Press, 2005

Course Code	Course Title	Credits	Lectures /Week
USCSP6031	**Wireless Sensor Networks – Practical**	**1**	**3**
1	Understanding the Sensor Node Hardware. (For Eg. Sensors, Nodes(Sensor mote), Base Station,Graphical User Interface.)		
2	Exploring and understanding TinyOS computational concepts: - Events, Commands and Task. - nesC model - nesC Components		
3	Understanding TOSSIM for - Mote-mote radio communication - Mote-PC serial communication		
4	Create and simulate a simple adhoc network		
5	Understanding, Reading and Analyzing Routing Table of a network.		
6	Create a basic MANET implementation simulation for Packet animation and Packet Trace		
7	Implement a Wireless sensor network simulation.		
8	Create MAC protocol simulation implementation for wireless sensor Network.		
9	Simulate Mobile Adhoc Network with Directional Antenna		
10	Create a mobile network using Cell Tower, Central Office Server, Web browser and Web Server. Simulate connection between them		

Course Code	Course Title	Credits	Lectures /Week
USCS6032	**Information Retrieval**	**2**	**3**

About the Course: This course introduces the principles, techniques, and technologies underlying information retrieval (IR) systems. Students will explore the fundamental concepts of document indexing, storage, and retrieval, as well as advanced topics such as retrieval models, text categorization, web information retrieval, and evaluation techniques. Through a combination of theoretical study, practical exercises, and reference to industry-standard books, students will gain a solid foundation in the field of information retrieval.

Course Objectives:

- To understand the fundamental principles and components of information retrieval systems.
- To explore various techniques for document indexing, storage, and retrieval.
- To analyze and compare different retrieval models and understand their strengths and limitations.
- To gain practical experience in implementing and evaluating information retrieval systems.
- To explore advanced topics in information retrieval, such as web information retrieval and machine learning techniques.

Learning Outcomes:

After successful completion of this course, students would be able to

- Explain the key components and principles of information retrieval systems.
- Apply indexing, storage, and retrieval techniques to efficiently retrieve relevant documents.
- Compare and contrast different retrieval models and select appropriate models for specific search scenarios.
- Develop practical skills in implementing and evaluating information retrieval systems.
- Demonstrate an understanding of advanced topics in information retrieval, including web search and machine learning techniques.

Unit	Topics	No of Lectures
I	**Foundations of Information Retrieval** **Introduction to Information Retrieval (IR) systems:** Definition and goals of information retrieval, Components of an IR system, Challenges and applications of IR **Document Indexing, Storage, and Compression:** Inverted index construction and compression techniques, Document representation and term weighting, Storage and retrieval of indexed documents, **Retrieval Models:** Boolean model: Boolean operators, query processing, Vector space model: TF-IDF, cosine similarity, query-document matching, Probabilistic model: Bayesian retrieval, relevance feedback **Spelling Correction in IR Systems**: Challenges of spelling errors in queries and documents, Edit distance and string similarity measures, Techniques for	**15**

	spelling correction in IR systems **Performance Evaluation:** Evaluation metrics: precision, recall, F-measure, average precision, Test collections and relevance judgments, Experimental design and significance testing	
II	**Advanced Topics in Information Retrieval** **Text Categorization and Filtering:** Text classification algorithms: Naive Bayes, Support Vector Machines, Feature selection and dimensionality reduction, Applications of text categorization and filtering **Text Clustering for Information Retrieval:** Clustering techniques: K-means, hierarchical clustering, Evaluation of clustering results, Clustering for query expansion and result grouping **Web Information Retrieval:** Web search architecture and challenges, Crawling and indexing web pages, Link analysis and PageRank algorithm **Learning to Rank:** Algorithms and Techniques, Supervised learning for ranking: RankSVM, RankBoost, Pairwise and listwise learning to rank approaches Evaluation metrics for learning to rank **Link Analysis and its Role in IR Systems:** Web graph representation and link analysis algorithms, HITS and PageRank algorithms, Applications of link analysis in IR systems	**15**
III	**Advanced Topics in Information Retrieval** **Crawling and Near-Duplicate Page Detection:** Web page crawling techniques: breadth-first, depth-first, focused crawling, Near-duplicate page detection algorithms, Handling dynamic web content during crawling **Advanced Topics in IR:** Text Summarization: extractive and abstractive methods, Question Answering: approaches for finding precise answers, Recommender Systems: collaborative filtering, content-based filtering **Cross-Lingual and Multilingual Retrieval:** Challenges and techniques for cross-lingual retrieval, Machine translation for IR, Multilingual document representations and query translation, Evaluation Techniques for IR Systems **User-based evaluation:** user studies, surveys, Test collections and benchmarking, Online evaluation methods: A/B testing, interleaving experiments	**15**

Textbook(s):

1. Ricardo Baeza-Yates and Berthier Ribeiro-Neto, ―Modern Information Retrieval: The Concepts and Technology behind Search, Second Edition, ACM Press Books
2. C. Manning, P. Raghavan, and H. Schütze, ―Introduction to Information Retrieval, Cambridge University Press

Additional Reference(s):

1. Ricci, F, Rokach, L. Shapira, B. Kantor, ―Recommender Systems Handbook‖, First Edition.
2. Bruce Croft, Donald Metzler, and Trevor Strohman, Search Engines: Information Retrieval in Practice, Pearson Education.
3. Stefan Buttcher, Charlie Clarke, Gordon Cormack, Information Retrieval: Implementing and Evaluating Search Engines, MIT Press.

Course Code	Course Title	Credits	Lectures /Week
USCSP6032	**Information Retrieval – Practical**	**1**	**3**
1	Document Indexing and Retrieval • Implement an inverted index construction algorithm. • Build a simple document retrieval system using the constructed index.		
2	Retrieval Models • Implement the Boolean retrieval model and process queries. • Implement the vector space model with TF-IDF weighting and cosine similarity.		
3	Spelling Correction in IR Systems • Develop a spelling correction module using edit distance algorithms. • Integrate the spelling correction module into an information retrieval system.		
4	Evaluation Metrics for IR Systems • Calculate precision, recall, and F-measure for a given set of retrieval results. • Use an evaluation toolkit to measure average precision and other evaluation metrics.		
5	Text Categorization • Implement a text classification algorithm (e.g., Naive Bayes or Support Vector Machines). • Train the classifier on a labelled dataset and evaluate its performance.		
6	Clustering for Information Retrieval • Implement a clustering algorithm (e.g., K-means or hierarchical clustering). • Apply the clustering algorithm to a set of documents and evaluate the clustering results.		
7	Web Crawling and Indexing • Develop a web crawler to fetch and index web pages. • Handle challenges such as robots.txt, dynamic content, and crawling delays.		
8	Link Analysis and PageRank • Implement the PageRank algorithm to rank web pages based on link analysis. • Apply the PageRank algorithm to a small web graph and analyze the results.		
9	Learning to Rank • Implement a learning to rank algorithm (e.g., RankSVM or RankBoost). • Train the ranking model using labelled data and evaluate its effectiveness.		
10	Advanced Topics in Information Retrieval • Implement a text summarization algorithm (e.g., extractive or abstractive). • Build a question-answering system using techniques such as information extraction		

Course Code	Course Title	Credits	Lectures /Week
USCS6041	**Data Mining & Warehousing**	**2**	**3**

About the Course: The course covers data warehousing, data mining, association rule mining, classification and prediction, and clustering. Students will learn about OLAP and OLTP, multi-dimensional data models, measures, concept of hierarchy, and data warehouse architecture. They will also explore different data mining functionalities. The course covers classification methods, prediction techniques, and classifier accuracy assessment. Students will learn various clustering methods and their applications to different data types, such as time-series data, text databases, and web mining.

Course Objectives:

- Understand the concept and framework of data warehousing and differentiate between OLAP and OLTP.
- Gain knowledge of data mining techniques and their applications in knowledge discovery.
- Acquire skills in data preprocessing, including handling missing data, cleaning, integration, and transformation.
- Apply association rules mining algorithms such as APRIORI and FP-Growth to discover frequent item sets.

Learning Outcomes:

After successful completion of this course, students would be able to

- Explain the purpose and components of a data warehouse and differentiate it from transactional databases.
- Perform OLAP operations on a multidimensional data model to analyze and query data.
- Implement data preprocessing techniques to address missing data and prepare the data for mining.
- Apply association rules mining algorithms to discover patterns and relationships in large datasets.

Unit	Topics	No of Lectures
I	**Introduction to Data Warehouse:** Introduction, Necessity, Framework of the data warehouse, options, developing data warehouses, Differences between OLAP and OLTP, OLAP Operations in the Multidimensional Data Model, Back-End Tools and Utilities, Metadata Repository, Types of OLAP servers. **DW Design Consideration And Dimensional Modeling:** Defining Dimensional Model, Granularity of Facts, Additivity of Facts, Functional dependency of the Data, Helper Tables, Implementation many-to-many relationships between fact and dimensional modeling. **Data Warehouse Models:** Enterprise Data Warehouse (EDW), Data Mart, Virtual Data Warehouse, Hybrid Data Warehouse.	**15**
II	**Data Mining:** Introduction to Data Mining, Definition, Knowledge Discovery in Data (KDD), Kinds of databases, Data to be mined, Basic mining techniques, Data Mining Issues, Data Mining Metrics, Social Implications of Data Mining, Overview of Applications of Data Mining.	**15**

	Data Preprocessing: Data Processing prerequisites, Attributes and Data types, Statistical descriptions of data, Distance and similarity measures, Need for Preprocessing, Handling Missing data, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization. **Association Rules Mining:** Problem Definition, Frequent item set generation, The APRIORI Principle, Support and confidence measures, Association rule generation: APRIORI algorithm, FP-Growth Algorithms, Compact Representation of Frequent item Set: Maximal Frequent item set, closed frequent item set.	
III	**Classification And Prediction:** Definition of classification, Model construction, Model Usage, Choosing algorithm, Decision tree Induction, Information gain, gain ratio, gini index, Bayesian Classification, Bayes Theorem, Naïve Bayes classifier, Linear Regression, Non-linear Regression, Logistic Regression. **Validating Model:** Measuring performance of classifiers, Precision, Recall, F-measure, confusion matrix, cross-validation, Bootstrap**.** **Clustering:** Types of data, Categorization of major clustering methods, K-means partitioning methods, Hierarchical methods, Density-based methods, Grid-based methods, Model-based clustering methods, Outlier analysis, Mining Time-Series and Sequence Data, Mining Text Databases, Mining the World Wide Web.	**15**

Textbook(s):

1. Data Warehousing : Design, Development And Best Practices by Soumendra Mohanty (Author), Tata McGraw Hill Education (Publisher).
2. Jiawei Han, Michelin Kamber, “Data Mining-Concepts and techniques”, Morgan Kaufmann Publishers, Elsevier, 3nd Edition.
3. Alex Berson, Stephen J.Smith, “Data warehousing Data mining and OLAP”, Tata McGraw- Hill, 2nd Edition.

Additional Reference(s):

1. Arum K Pujari, “Data Mining Techniques”, 3rd Edition, Universities Press, 2005
2. PualrajPonnaiah, Wiley, “Data Warehousing Fundamentals”, Student Edition, 2004.
3. Ralph Kimball, Wiley, “The Data warehouse Life Cycle Toolkit”, Student Edition, 2006.

Course Code	Course Title	Credits	Lectures /Week
USCSP6041	**Data Mining & Warehousing – Practical**	**1**	**3**
1	Perform different operations of extraction, transformation, and loading (ETL) processes on a sample dataset using PowerBI.		
2	Integrate data from multiple sources by merging and transforming datasets using Python's pandas library and data manipulation techniques.		
3	Apply feature selection techniques like variance thresholding and correlation analysis using Python's **scikit-learn library** to reduce dimensionality in a dataset.		
4	**Discretize** continuous variables and create concept hierarchies for categorical variables in a market basket dataset using Python's pandas library.		
5	Implement the **Apriori algorithm** in Python to mine frequent itemsets from a retail transaction dataset and extract association rules.		
6	Build a **decision tree classifier** using Python's scikit-learn library to predict customer churn based on historical data.		
7	Implement a **Naive Bayes classifier in** Python using scikit-learn to classify emails as spam or non-spam based on their content.		
8	Implement a **linear regression** method to make predictions based on the sample data set using Python.		
9	Implement a **logistic regression** method to make predictions based on the sample data set using Python.		
10	Implement **K-means clustering** algorithm in Python using scikit-learn to group customers based on their purchasing behavior.		

Course Code	Course Title	Credits	Lectures /Week
USCS6042	**Ethical Hacking**	**2**	**3**

About the Course: This course provides an in-depth exploration of ethical hacking and penetration testing methodologies. Students will learn about hacking technology types, the phases of ethical hacking, footprinting, social engineering, system hacking, web server and application vulnerabilities, wireless hacking, and more. The course emphasizes hands-on lab exercises and real-world scenarios to develop practical skills in identifying and mitigating security vulnerabilities.

Course Objectives:

- Understand the terminology and concepts related to ethical hacking and penetration testing.
- Explore various hacking technologies and the skills required to become an ethical hacker.
- Learn the different phases involved in ethical hacking and the methodologies used in penetration testing.
- Gain knowledge of common hacking techniques, such as footprinting, scanning, enumeration, and session hijacking.
- Develop proficiency in identifying and exploiting vulnerabilities in web servers, web applications, and wireless networks.

Learning Outcomes:

After successful completion of this course, students would be able to

- Apply ethical hacking methodologies to conduct comprehensive security assessments and penetration tests.
- Perform effective footprinting and reconnaissance techniques to gather critical information about target systems.
- Identify and exploit vulnerabilities in various network and system components using appropriate tools and techniques.
- Evaluate the security posture of web servers, web applications, and wireless networks, and recommend appropriate countermeasures.
- Demonstrate an understanding of ethical and legal considerations in conducting ethical hacking activities and adhere to professional codes of conduct.

Unit	Topics	No of Lectures
I	**Introduction:** Terminology, Hacking Technology Types, Ethical Hacking Phases, Hacktivism, Hacker Classes, Skills Required for an Ethical Hacker, Vulnerability Research, Ways to Conduct Ethical Hacking **Footprinting:** Definition, Information Gathering Methodology, Competitive Intelligence, DNS Enumeration, Whois and ARIN Lookups, Types of DNS Records, Traceroute in Footprinting, E-Mail Tracking **Social Engineering:** Common Types Of Attacks **Scanning and Enumeration:** Port Scanning, Network Scanning, Vulnerability Scanning, CEH Scanning Methodology, Ping Sweep Techniques, Nmap Command Switches, SYN, Stealth, XMAS, NULL,	**15**

	IDLE, FIN Scans, Anonymizers, HTTP Tunneling Techniques, IP Spoofing Techniques, SNMP Enumeration, Steps Involved in Enumeration	
II	**System Hacking:** Password-Cracking Techniques, Types of Passwords, Keyloggers and Other Spyware Technologies, Escalating Privileges, Rootkits **Sniffers:** Protocols Susceptible to Sniffing, Active and Passive Sniffing, ARP Poisoning, MAC Flooding, DNS Spoofing Techniques, Sniffing Countermeasures **Denial of Service:** Types of DoS Attacks, Working of DoS Attacks, BOTs/BOTNETs, "Smurf" Attack, "SYN" Flooding, DoS/DDoS Countermeasures **Session Hijacking:** Spoofing vs. Hijacking, Types, Sequence Prediction, Steps, Prevention **Hacking Web Servers:** Web Server Vulnerabilities, Attacks against Web Servers, Patch Management Techniques, Web Server Hardening	**15**
III	**Web Application Vulnerabilities:** Web Application Hacking, Web Application Threats, Google Hacking, Countermeasures **Web-Based Password Cracking Techniques:** Authentication Types, Password Crackers, Countermeasures **SQL Injection:** Steps, SQL Server Vulnerabilities, Countermeasures **Buffer Overflows:** Types, Stack-Based Buffer Overflows, Mutation Techniques **Wireless Hacking:** WEP, WPA Authentication Mechanisms, and Cracking Techniques, Wireless Sniffers, Rogue Access Points, Wireless Hacking Techniques, Securing Wireless Networks **Penetration Testing Methodologies:** Methodologies, Steps, Automated Tools, Pen-Test Deliverables	**15**

Textbook(s):

1. CEH official Certfied Ethical Hacking Review Guide, Wiley India Edition

Additional Reference(s):

1. Certified Ethical Hacker: Michael Gregg, Pearson Education
2. Certified Ethical Hacker: Matt Walker, TMH.

Course Code	Course Title	Credits	Lectures /Week
USCSP6042	**Ethical Hacking - Practical**	**1**	**3**
1	Google and Whois Reconnaissance • Use Google search techniques to gather information about a specific target or organization. • Utilize advanced search operators to refine search results and access hidden information. • Perform Whois lookups to retrieve domain registration information and gather details about the target's infrastructure.		
2	Password Encryption and Cracking with CrypTool and Cain and Abel • Password Encryption and Decryption: o Use CrypTool to encrypt passwords using the RC4 algorithm. o Decrypt the encrypted passwords and verify the original values. • Password Cracking and Wireless Network Password Decoding: o Use Cain and Abel to perform a dictionary attack on Windows account passwords. o Decode wireless network passwords using Cain and Abel's capabilities.		
3	Linux Network Analysis and ARP Poisoning • Linux Network Analysis: o Execute the ifconfig command to retrieve network interface information. o Use the ping command to test network connectivity and analyze the output. o Analyze the netstat command output to view active network connections. o Perform a traceroute to trace the route packets take to reach a target host. • ARP Poisoning: o Use ARP poisoning techniques to redirect network traffic on a Windows system. o Analyze the effects of ARP poisoning on network communication and security.		
4	Port Scanning with NMap • Use NMap to perform an ACK scan to determine if a port is filtered, unfiltered, or open. • Perform SYN, FIN, NULL, and XMAS scans to identify open ports and their characteristics. • Analyze the scan results to gather information about the target system's network services.		
5	Network Traffic Capture and DoS Attack with Wireshark and Nemesy • Network Traffic Capture: o Use Wireshark to capture network traffic on a specific network interface. o Analyze the captured packets to extract relevant information and identify potential security issues.		

	• Denial of Service (DoS) Attack:
	  ○ Use Nemesy to launch a DoS attack against a target system or network.
	  ○ Observe the impact of the attack on the target's availability and performance.

6	Persistent Cross-Site Scripting Attack • Set up a vulnerable web application that is susceptible to persistent XSS attacks. • Craft a malicious script to exploit the XSS vulnerability and execute arbitrary code. • Observe the consequences of the attack and understand the potential risks associated with XSS vulnerabilities.
7	Session Impersonation with Firefox and Tamper Data • Install and configure the Tamper Data add-on in Firefox. • Intercept and modify HTTP requests to impersonate a user's session. • Understand the impact of session impersonation and the importance of session management.
8	SQL Injection Attack • Identify a web application vulnerable to SQL injection. • Craft and execute SQL injection queries to exploit the vulnerability. • Extract sensitive information or manipulate the database through the SQL injection attack.
9	Creating a Keylogger with Python • Write a Python script that captures and logs keystrokes from a target system. • Execute the keylogger script and observe the logged keystrokes. • Understand the potential security risks associated with keyloggers and the importance of protecting against them.
10	Exploiting with Metasploit (Kali Linux) • Identify a vulnerable system and exploit it using Metasploit modules. • Gain unauthorized access to the target system and execute commands or extract information. • Understand the ethical considerations and legal implications of using Metasploit for penetration testing.

Course Code	Course Title	Credits	Lectures /Week
USCS6051	**Customer Relationship Management**	**2**	**3**

About the Course: This course on Customer Relationship Management (CRM) provides an in-depth understanding of the principles, strategies, and tools necessary for managing customer relationships effectively. Students will explore the various forms of CRM and its impact on business performance. Additionally, the course covers customer acquisition, retention, and the measurement of customer-perceived value. Students will also gain insights into strategic and operational CRM, including customer portfolio management, marketing automation, and service automation. The course concludes with an examination of analytical CRM and the implementation of CRM strategies through real-life case studies.

Course Objectives:

- To provide students with a comprehensive understanding of CRM concepts, theories, and models.
- To equip students with the knowledge and skills to manage the customer journey, including customer acquisition and retention.
- To explore the factors that contribute to customer-perceived value and the role of CRM in enhancing the customer experience.
- To familiarize students with strategic and operational CRM approaches, including customer portfolio management and marketing automation.
- To introduce students to the analytical aspects of CRM, including data management, analytics for strategy and tactics, and the implementation of CRM systems.

Learning Outcomes:

After successful completion of this course, students would be able to

- Students will be able to define and explain the various forms of CRM and their relevance to business contexts.
- Students will acquire the skills to manage the customer journey effectively, including implementing customer acquisition and retention programs.
- Students will understand the importance of customer-perceived value and its impact on customer satisfaction, loyalty, and business performance.
- Students will be able to apply strategic and operational CRM approaches, such as customer portfolio management and marketing automation, to enhance organizational effectiveness.
- Students will develop proficiency in analytical CRM techniques, including data management, analytics for strategy and tactics, and the successful implementation of CRM systems. They will also be able to analyze and draw insights from real-life case studies and success stories related to CRM.

Unit	Topics	No of Lectures
I	**Understanding Customer Relationships** **Introduction to CRM:** Three forms of CRM, The changing face of CRM, Misunderstandings about CRM, Defining CRM, CRM constituencies, Commercial contexts of CRM, Models of CRM, Understanding relationships, Relationship Quality, Customer life-time value, Customer	**15**

	satisfaction, loyalty and business performance, Relationship management theories, Benefits of CRM **Managing the customer journey:** customer acquisition, Portfolio purchasing, Prospecting, Key performance indicators of customer acquisition programs, Operational CRM tools that help customer acquisition, Customer retention, Economics and Strategies of customer retention, Key performance indicators of customer retention programs. **Managing customer-experienced value:** Understanding value, modeling customer-perceived value, its sources, Customization, Value through the marketing mix, Customer Experience concepts, Service marketing, Total quality management, relationship management, CRM's influence on CX, How CRM software applications influence customer experience	
II	**Strategic and Operational CRM** **Customer portfolio management:** Portfolio, customer, Basic disciplines for CPM, CPM in the business-to-business context, customer portfolio management tools, strategically significant customers, The seven core customer management strategies **Marketing automation:** Introduction to marketing automation, Benefits, Software applications for marketing, Sales force automation **Service automation:** Introduction, customer service, Modeling service quality, Customer service standards, service automation, Benefits, Software applications for service	**15**
III	**Analytical CRM** **Developing and managing customer-related databases:** Corporate customer-related data, Structured and unstructured data, Developing a customer-related database, Data integration, Data warehousing, Data marts, Analytics for CRM strategy and tactics, Big data analytics, Analytics for structured data, ways to generate analytical insight, Data-mining procedures, Artificial intelligence (Al), machine learning (ML) and deep learning (DL) **Implementing CRM:** Introduction, develop the CRM strategy, build CRM project foundations, needs specification and partner selection, project implementation, performance evaluation **Case studies and success stories related to CRM**	**15**

Textbook(s):

1. Customer Relationship Management Concepts and Technologies, Francis Buttle, Stan Maklan, Routledge Taylor and Francis Group, 2019
2. Jagdish N Sheth, Parvatiyar Atul, G Shainesh, Customer Relationship Management- Emerging Concepts, Tools and Applications, 2017
3. Anderrson Kristin , Carol Kerr, Customer Relationship Management, Tata McGraw-Hill, 2017

Additional Reference(s):

1. V. Kumar & Werner J., CUSTOMER RELATIONSHIP MANAGEMENT, Willey India, 2008
2. S. Shanmugasundaram, CUSTOMER RELATIONSHIP MANAGEMENT, Prentice Hall of India Private Limited, New Delhi, 2008

Course Code	Course Title	Credits	Lectures /Week
USCS6052	**Cyber Laws and IPR**	**2**	**3**

About the Course: The course on Cyber Laws and IPR provides a comprehensive understanding of the legal aspects and regulations related to cyberspace and information technology. The course covers a wide range of topics, including basic concepts, internet technology, network security, cyber law, e-commerce, electronic signatures, cyber crimes, privacy, intellectual property rights, and more. Students will explore the legal framework governing cyberspace and develop an understanding of the legal and ethical issues associated with information technology.

Course Objectives:

- Understand fundamental concepts of cyber laws and their relevance in the digital age.
- Examine legal frameworks and regulations in cyber laws, including the Information Technology Act 2000 in India.
- Explore key issues in cyber laws such as e-commerce, e-governance, and electronic records and contracts.
- Gain knowledge of cybercrimes, enforcement mechanisms, and the role of the Cyber Appellate Tribunal.
- Analyze emerging issues in cyber laws, including liability of ISPs, privacy concerns, and jurisdictional aspects.
- Understand intellectual property rights and online regulations, including copyrights, patents, and domain name disputes.

Learning Outcomes:

After successful completion of this course, students would be able to

- Demonstrate a comprehensive understanding of cyber laws and their application in the digital age.
- Evaluate legal frameworks and regulations governing cyber laws.
- Identify and assess key issues in cyber laws, such as e-commerce, e-governance, and electronic records and contracts.
- Understand cyber crimes, enforcement mechanisms, and the role of the Cyber Appellate Tribunal.
- Analyze emerging issues in cyber laws, including liability of ISPs, privacy concerns, and jurisdictional complexities.
- Recognize intellectual property rights and online regulations, including copyrights, patents, and domain name disputes.

Unit	Topics	No of Lectures
I	**Introduction to Cyber Laws and Technology:** Basic Concepts, Internet and Advantages and Disadvantages of Internet Technology, Network and Network Security **Legal Framework and Regulations:** Cyber Law & Components of Cyber Law, Cyber Law in India: An Overview of Information Technology Act 2000, Cryptography, Encryption Technique & Algorithm and Digital	**15**

	Signature & Electronic Signature **Key Issues in Cyber Laws:** E-Commerce, E-Governance, E-Record & E-Contract, Regulator, Certifying Authority, Electronic Signature Certificates	
II	**Cyber Crimes and Enforcement:** Cyber Appellate Tribunal, Cyber Crimes-Cyber Contraventions, Cyber Offences, Power of Investigation & Search, E-Evidence and Computer Forensic **Emerging Issues and Legal Considerations:** ISP & Intermediary Not to be Liable in Certain Cases, Consequential Amendments in Various Conventional Laws in India, Grey Areas of Information Technology Act, 2000, **Jurisdiction and Privacy:** Cyber Jurisdiction, E-Consumers, Privacy of Online Data and Information	**15**
III	**Intellectual Property Rights and Online Regulations:** Free Speech Online or Online Freedom of Speech and Expression and Liability of Intermediary **Intellectual Property Rights (IPRs), Copyrights & Patents:** International and Indian Scenario, Copyright Issues and Digital Medium, Patent Issues in Digital Medium **Disputes and Resolution:** Domain Name Dispute & Resolution and Trademark Issues in Digital Medium, Spamming and Phishing	**15**

Textbook(s):

1. Cyber Laws & Information Technology (For LL.B.) Paperback – 1 January 2020
2. Cyber Law in India, Satish Chandra, ABS Books, 2017
3. Cyber Security and Cyber Laws, Nilakshi Jain, Wiley India, October 2020

Additional Reference(s):

1. Cyber Laws, Justice Yatindra Singh, Universal Law Publishing, Universal Publishing, 2016
2. Cyber laws, Dr. Gupta & Agrawal, PREMIER PUBLISHING COMPANY, 2022
3. Cyber Law - An Exhaustive Section Wise Commentary On The Information Technology, Pavan Duggal, Universal Publishing (LexisNexis), 2nd Edition, 2017

Course Code	Course Title	Credits	Lectures /Week
USCSP605	**Project Work – II**	**2**	**3**
Refer to the Project Guidelines mentioned at the end			

Evaluation Scheme

I. Internal Evaluation for Theory Courses – 25 Marks

(i) Mid-Term Class Test – 15 Marks

- It should be conducted using any **learning management system** such as **Moodle** (Modular object-oriented dynamic learning environment)
- The test should have **15 MCQ's** which should be solved in a time duration of **30 minutes.**

(ii) Assignment/ Case study/ Presentations – 10 Marks

- Assignment / Case Study Report / Presentation can be uploaded on any **learning management system**.

II. External Examination for Theory Courses – 75 Marks

- Duration: **2.5 Hours**
- Theory question paper pattern:

	All questions are compulsory.		
Question	**Based on**	**Options**	**Marks**
Q.1	Unit I	*Any 4 out of 6*	20
Q.2	Unit II	*Any 4 out of 6*	20
Q.3	Unit III	*Any 4 out of 6*	20
Q.4	Unit I, II and III	*Any 5 out of 6*	15

- All questions shall be compulsory with internal choice within the questions.
- Each Question may be sub-divided into sub questions as a, b, c, d, etc. & the allocation of Marks depends on the weightage of the topic.

III. Practical Examination

- Each subject carries 50 Marks

  40 marks + 05 marks (journal) + 05 marks (viva)
- Duration: **2 Hours** for each practical course.
- Minimum **80% practical** from each core subjects are required to be completed.
- **Certified Journal is compulsory for appearing at the time of Practical Exam**
- The final submission and evaluation of **journal in electronic form** using a Learning Management System / Platform can be promoted by college.

IV. Project

The evaluation of the project will include a viva voce, which will assess the project based on the following parameters:

- **Documentation – 30 Marks:** The completeness, accuracy, and professionalism of the project documentation, including the project report and supporting materials, will be considered.
- **Quality of the Project – 15 Marks:** The overall quality of the project, including its design, implementation, and user experience, will be evaluated.
- Working of the Project – 20 Marks: The functionality and performance of the project will be assessed to determine how well it meets the specified requirements and objectives.
- **Project Presentation – 15 Marks:** The clarity, organization, and effectiveness of the project presentation will be evaluated.
- **Viva – 20 Marks:** The viva voce session will provide an opportunity for the student to demonstrate their knowledge and understanding of the project, as well as to answer questions and engage in a discussion with the evaluators.

Project Guidelines *(for USCSP505 and USCSP605)*

Aim:

The Project Work as part of B.Sc. Computer Science program provides students with practical experience in applying their knowledge and skills to real-world projects, emphasizing hands-on experience in industry-standard project practices. It focuses on project development, implementation, and deployment using computer science principles and techniques. Students will work individually or in teams to design, develop, and present a substantial software project, gaining exposure to real-life project scenarios. It also covers project planning, requirements gathering, software design, coding, testing, debugging, documentation, and project management, following industry best practices. Through these projects, students will enhance their problem-solving abilities, gain proficiency in software development methodologies, and strengthen their practical skills in computer science.

Objectives:

- Apply interdisciplinary knowledge to effectively solve real-life problems using acquired skills and concepts.
- Gain hands-on experience in the software development life cycle, encompassing requirements analysis, design, implementation, testing, and deployment.
- Familiarize with global IT industry standards, ethics, and professional practices to thrive in a professional environment.
- Develop teamwork and project management skills through structured collaboration, effective communication, and task delegation.
- Produce professional technical documentation aligning with industry practices, ensuring clarity, accuracy, and usability.
- Acquire time management, resource allocation, and personnel coordination skills for efficient project execution.

Project Types:

a) Developing a solution for a real-life problem: In this case, the project focuses on addressing an existing requirement for a computer-based solution that has practical applications. The project should successfully implement the different stages of the system development life cycle. Examples: Secure Online Banking System, Machine Learning-based Disease Diagnosis System, Cloud-based Document Management System.

b) **Innovative Product Development:** These projects involve exploring and developing a computer-based solution with a unique and innovative utility. Examples: Cybersecurity Monitoring and Threat Detection System, Machine Learning-powered Predictive Maintenance System for Industrial Equipment, IoT-based Smart Energy Management System.

c) **Research-Level Project:** These projects involve conducting research and development to explore advanced technologies and solve complex problems. Examples: Deep Learning-based Image Recognition System for Medical Imaging, Cloud Computing Infrastructure Optimization for Big Data Processing, Data Science-driven Predictive Analytics for Sales Forecasting. The methodology and reporting of such projects may vary based on the project supervisor's guidance.

Tools & Technologies:

In the project work, students are granted complete freedom to select platforms, tools, and programming languages without any imposed restrictions. This approach encourages creativity, flexibility, and exploration of various technologies. By prioritizing open-source technologies, students can leverage a vast array of resources and community support. Commonly employed tools include IDEs, version control systems (e.g., Git), programming languages (e.g., Python, Java), databases (e.g., MySQL), and web frameworks (e.g., Django, Ruby on Rails). The evaluation process focuses on the project's content and implementation rather than the specific tools chosen, ensuring a fair assessment of the students' skills and problem-solving abilities.

Project Guide:

Assigning a project guide to each project or group is a mandatory requirement to ensure the successful completion of the project work. The guide plays a crucial role as a mentor and technical expert, providing invaluable support and guidance to students. They are expected to facilitate effective communication and teamwork, review project proposals, assign schedules, and monitor progress on a regular basis. Additionally, guides are expected to offer timely feedback, provide guidance on project planning and implementation strategies, evaluate the quality of work, and promote professionalism and ethical conduct. Their expertise and involvement are essential in helping students navigate challenges, make informed decisions, and achieve their project goals effectively.

Project Team Size: 1 – 2 members

Project Proposal: The project proposal is a mandatory document that serves as a foundation for the project. It helps students define their project idea, receive early evaluation and feedback, establish clear communication with the project guide, and take ownership of the project's successful execution. A formal proposal ensures systematic and professional project planning, fostering critical thinking, effective communication, and project management skills. The proposal provides a roadmap and increases the chances of a successful outcome. Before initiating a project, it is mandatory to submit a project proposal for approval. **The original duly approved project proposal should be attached to the final project report.** The project proposal for UG computer science projects should include the following contents:

- Title
- Introduction
- Objectives: Clearly state the objectives of the project. What specific goals do you aim to achieve?
- Scope
- Methodology
- Tools and Technologies
- Timeline
- Resources
- Expected Outcomes
- References

Project Report:

The Certified Copy of Hard Bound Project Report must adhere to the following guidelines:

- No of Copies: Team Size + 1 (College / Department Copy)
- The project report should include the following
    - Title Page *(Sample attached in Appendix)*
    - Certificate *(Sample attached in Appendix)*
    - Declaration *(Sample attached in Appendix)*
    - Acknowledgement
    - Table of Contents
    - Original Copy of approved Project Proposal
    - Self-attested copy of Plagiarism Report from any open source tool.
    - Chapters / Sections depending upon the type of project
    - List of Tables and/or List of Figures
    - References (IEEE / Springer format)
    - Glossary
    - Appendices (Survey datasheets / Questionnaires, ect)
- Use of LaTeX for documentation purposes should be preferred.
- The text of the report should be set in 12 pt, Times New Roman font, and single-spaced.
- Chapter headings should be centered, written in 20 pt, Times New Roman font, bold, and in all caps.
- These guidelines ensure a standardized format for the project report, promoting clarity and readability.

SAMPLE TITLE PAGE FORMAT

A PROJECT REPORT

on

<PROJECT NAME>

Submitted by

Mr. XYZ

in partial fulfillment for the award of the degree

of

BACHELOR OF SCIENCE

in

COMPUTER SCIENCE

under the guidance of

<Guide Name>

Department of Computer Science

<<College Logo>>

<<College Name>>

(Sem V / VI)

(202- – 202-)

SAMPLE CERTIFICATE FORMAT

<<College Logo>>

<<College Name>>,

<<College Address>>

Department of Computer Science

CERTIFICATE

This is to certify that Mr./Ms. _____ of **T.Y.B.Sc. (Sem V/VI)** class has satisfactorily completed the Project _____, to be submitted in the partial fulfillment for the award of **Bachelor of Science** in **Computer Science** during the academic year **202- – 202-**.

Date of Submission:

Project Guide

Head / Incharge,
Department Computer Science

College Seal

Signature of Examiner

SAMPLE DECLARATION FORMAT

DECLARATION

I, _____, hereby declare that the project entitled "_____" submitted in the partial fulfillment for the award of **Bachelor of Science** in **Computer Science** during the academic year **202- – 202-** is my original work and the project has not formed the basis for the award of any degree, associateship, fellowship or any other similar titles.

Signature of the Student:

Place:

Date: