SNDT Women's University, Mumbai

Master of Science (Computer Science)
(M.Sc.- CS.)

as per NEP-2020

Syllabus

w.e.f.

A.Y.: 2023-24

Programme	Master of Science (Computer Science) (M.Sc.- CS.)
Preamble	In unwavering commitment to the principles outlined herein, the Master of Science (Computer Science) program steadfastly aims to shape an educational experience that empowers its students to excel as adept scholars, critical thinkers, and responsible leaders within their respective fields. By articulating these guiding principles, we underscore our dedication to fostering a transformative learning environment that goes beyond the acquisition of technical skills to cultivate holistic and forward-thinking professionals.
Programme Outcomes (POs)	After completing this programme, Learner will be able to • **Advanced Knowledge-** Demonstrated proficiency in fundamental and specialized computer science concepts, encompassing algorithms, data structures, artificial intelligence, machine learning, and relevant domains. • **Critical Thinking and Problem Solving**- Exhibited ability to analyse intricate problems, synthesize information, and apply critical thinking skills for the creation of innovative and effective solutions within the computer science field • **Advanced Technical Skills-** Possession of advanced technical skills in programming languages, software development, system design, and other pertinent areas, enabling the creation of robust and efficient computing solutions • **Ethical Considerations-** Displayed a strong understanding of ethical considerations in computer science, encompassing privacy, security, intellectual property, and societal impact. Graduates are equipped to make informed and ethical decisions in their professional practice. • **Lifelong Learning-** Embraced a commitment to lifelong learning, showcasing the ability to stay current with emerging technologies, industry trends, and advancements in computer science through self-directed learning and ongoing professional development.
Programme Specific Outcomes (PSOs)	Programme Specific Outcomes (PSOs) for an MSc (Computer Science) specify the particular skills, knowledge and abilities that students are expected to gain upon completion of the program.

	• **Advanced Technical Proficiency-** Demonstrate mastery of advanced concepts in computer science, including algorithms, data structures, databases, and software engineering, to design and implement complex computing solutions. • **Specialized Knowledge in Focus Areas-** Develop expertise in specific focus areas within computer science such as artificial intelligence, machine learning, cyber security, or data science, showcasing advanced knowledge and skills in these specialized domains. • **Research and Innovation-** Conduct independent research, including formulating research questions, designing experiments, and analyzing results, contributing to the creation of new knowledge and advancements in the field. • **Advanced Programming Skills-** Exhibit proficiency in various programming languages and paradigms, enabling the development of efficient and scalable software solutions. • **System Design and Architecture-** Design and architect complex computing systems, demonstrating an understanding of system-level considerations, scalability, and performance optimization. • **Effective Communication of Technical Information-** Communicate complex technical information effectively to both technical and non-technical audiences through written reports, presentations, and documentation. • **Innovation and Entrepreneurship-** Foster innovation and entrepreneurial thinking, demonstrating the ability to identify opportunities, propose creative solutions, and potentially contribute to start-ups or innovative projects.
Eligibility Criteria for Programme	A woman Graduate in BSc. (Physics), BSc. (Maths.), BSc. (Elect.), BSc. (IT), B.Sc.(CS) or BCA or any engineering graduate in allied subject from the recognized university with an aggregate marks not less than 50% (Open Category) and 45%(Reserved category).
Intake	60
Duration	4 semesters (2 years)

Master of Science (Computer Science)(M.Sc.-CS.)

Year -I

Code	Subjects	Type of Course	Credits	Marks	Int.	Ext.
	Semester-I`					
115511	Operating Systems	Major (Core) Theory	4	100	50	50
115512	Data Communications and Networking	Major (Core) Theory	4	100	50	50
115513	Data Structures and Analysis of Algorithm	Major (Core) Theory	2	50	0	50
115524	Data Structures and Analysis of Algorithm- Lab	Major (Core) Practical	2	50	25	25
115525	Operating Systems-Lab	Major (Core) Practical	2	50	25	25
125511/ 125512/ 125513/ 125514	Elective-I	Major (Elective) Theory	4	100	50	50
135511	Research Methodology	Minor Stream (RM) Theory	4	100	50	50
			22	550	250	300
	Semester-II	.				
Code	**Subjects**	**Type of Course**	**Credit**	**Marks**	**Int.**	**Ext.**
215511	Data Warehousing and Data Mining	Major (Core) Theory	4	100	50	50
215512	Database Management Systems	Major (Core) Theory	4	100	50	50
215513	Web Technology	Major (Core) Theory	2	50	50	0
215524	Database Management Systems-Lab	Major (Core) Practical	2	50	25	25
215525	Web Technology-Lab	Major (Core) Practical	2	50	25	25
225511/ 225512/ 225513/ 225514	Elective-II	Major (Elective) Theory	4	100	50	50
245541	OJT	OJT	4	100	50	50
			22	550	300	250

Exit option (44 credits):Post Graduate Diploma in Computer Science

Year -II

Code	Subjects	Type of Course	Credit	Marks	Int.	Ext.
	Semester-III					
315511	Big Data Analytics	Major (Core) Theory	4	100	50	50
315512	Machine Learning	Major (Core) Theory	4	100	50	50
315513	Data Science	Major (Core) Theory	2	50	0	50
315524	Big Data Analytics-Lab	Major (Core) Practical	2	50	25	25
315525	Machine Learning-Lab	Major (Core) Practical	2	50	25	25
325511/ 325512/ 325513/ 325514	Elective-III	Major (Elective) Theory	4	100	50	50
355531	Research Project	RP	4	100	50	50
			22	550	250	300
	Semester-IV					
415511	Deep Learning	Major (Core) Theory	4	100	50	50
415512	Natural Language Processing	Major (Core) Theory	4	100	50	50
415513	Mobile Application Development using Android Programming	Major (Core) Practical	2	50	50	0
425511/ 425512/ 425513/ 425514	Elective-IV/(MOOC/SWAYAM)	Major (Elective) Theory	4	100	50	50
445541	Internship	OJT	8	200	100	100
			22	550	300	250

Code	Elective-I	Code	Elective-II
125511	1.Cyber Security	225511	1.Ethical Hacking
125512	2.Digital Image Processing	225512	2.Project Management
125513	3.Software Engineering	225513	3.Fuzzy Logic & Neural Network
125514	4.Artificial Intelligence	225514	4.IoT

Code	Elective-III	Code	Elective-IV
325511	1.Blockchain	425511	1.Information Security
325512	2.GIS and Remote Sensing	425512	2.Digital Forensics
325513	3.Software Testing	425513	3.Agile Methodology
325514	4.Robotic Process Automation	425514	4.Cloud Computing

SN	Courses, Modules and Outcomes	Course Contents	Cr
	Semester III		
315511	**BIG DATA ANALYTICS** **Major (Core) Theory**		**4**
	Course Outcomes: **Learners will be able to:** • Apply IBM's approach to big data and analytics, incorporating principles of data processing and analysis within a Hadoop ecosystem. • Analyse the flow of data in a Hadoop ecosystem to evaluate its efficiency and identify potential optimizations. • Design a comprehensive strategy for processing and analyzing data in Hadoop, considering steps involved and potential challenges. • Evaluate the functionality and effectiveness of Hive data warehousing and its SQL-like query language within Apache Hive's services and architecture.		
Module 1	**INTRODUCTION TO BIG DATA AND HADOOP, HDFS (Hadoop Distributed File System)**		1
	LOs: Learners will be able to • Apply Unix tools for data analysis to understand the various types of digital data, distinguishing between structured, semi-structured, and unstructured data sets. • Analyse the fundamental concepts of Big Data, exploring its three V's (Volume, Velocity, Variety) to evaluate its significance and implications. • Evaluate the challenges and opportunities presented by Big Data, considering its impact on various industries and sectors. • Design a learning pathway to develop skills in utilizing Unix tools for data analysis, integrating practical applications with theoretical understanding of Big Data concepts.	**Module Contents:** • Types of Digital Data, Introduction to Big Data, Bigdata Analytics, History of Hadoop, Apache Hadoop, Analysing Data with Unix tools, Analysing Data with Hadoop, Hadoop Streaming, Hadoop Echo System, IBM Big Data Strategy, Introduction to Info sphere Big Insights and BigSheets. • The Design of HDFS, HDFS Concepts, Command Line Interface, Hadoop file system interfaces, Data flow, Data Ingest with Flume and Scoop and Hadoop archives, Hadoop I/O: Compression, Serialization, Avro and File-Based Data structures	
Module 2	**MapReduce**		1
	LOs: Learners will be able to • Apply knowledge of key components and phases in a MapReduce job to execute tasks effectively. • Analyze the sequence of steps from job submission to	**Module Contents:** • Anatomy of a Map Reduce Job Run, Failures, Job Scheduling,ShuffleandSort,TaskExecution,MapReduceTypesandFormats,MapReduceFeatures	

	.completion to identify potential bottlenecks or optimizations. • Evaluate the job scheduling process in a MapReduce framework to ensure efficient resource utilization. • Design strategies for task scheduling and resource allocation in a distributed environment to optimize performance and scalability.		
Module 3	**Hadoop Eco-System**		1
	LOs: Learners will be able to • Apply knowledge of Pig's different execution modes, distinguishing between local and MapReduce modes to optimize data processing workflows. • Analyse the advantages and use cases of each execution mode in Pig, evaluating their suitability for various data processing requirements. • Evaluate the syntax and semantics of Pig Latin, the scripting language for Pig, to understand its structure and functionality in data processing. • Design Pig Latin scripts for data processing tasks, incorporating syntax rules and best practices to achieve efficient and effective data transformations.	**Module Contents:** • Pig: Introduction to PIG, Execution Modes of Pig, Comparison of Pig with Databases, Grunt, Pig Latin, User Defined Functions, Data Processing operators. Hive : Hive Shell, Hive Services, HiveMetastore,ComparisonwithTraditionalDatabases,HiveQL,Tables,QueryingDataandUserDefinedFunctions.Hbase:HBasics,Concepts,Clients,Example,HbaseVersus RDBMS. Big SQL :Introduction	
Module 4	**Data Analytics with R Machine Learning:**		1
	LOs: Learners will be able to • Apply insights into the significance of big data in the analytics landscape to inform strategic decision-making and resource allocation. • Analyse the challenges and opportunities presented by large-scale data, identifying potential solutions and	**Module Contents:** • Introduction, Supervised Learning, Unsupervised Learning, Collaborative Filtering. Big Data Analytics with Big R.	

	innovative approaches to maximize its value.		
Assignments/ Activities			
	These assignments aim to apply theoretical concepts to practical application and critical thinking. • **Module 1:** **Activity: Hadoop Installation and Setup:** Students will install Hadoop on their local machines or on a virtual environment. They will follow step-by-step instructions to set up Hadoop, explore the Hadoop Distributed File System (HDFS), and run basic commands to manage files. **Assignment: Analysis of HDFS Concepts:** Students will write a report analyzing the design principles and concepts of the Hadoop Distributed File System (HDFS). They should discuss the architecture, command line interface, data flow, and data ingestion techniques such as Flume and Scoop. • **Module 2:** **Activity: MapReduce Job Execution Simulation**: Students will simulate the execution of a MapReduce job using Hadoop MapReduce framework. They will design a simple MapReduce program, submit it to the Hadoop cluster, and monitor the job execution process. **Assignment: MapReduce Job Optimization:** Students will optimize a given MapReduce job to improve its performance and efficiency. They should identify bottlenecks, apply optimization techniques such as combiners and partitioners, and measure the impact on job execution time. • **Module 3:** **Activity: Pig Latin Scripting**: Students will write Pig Latin scripts to perform data processing tasks using Apache Pig. They will use the Grunt shell to interactively execute Pig scripts and explore different data processing operators. **Assignment: Comparative Analysis of Hadoop Ecosystem Tools:** Students will compare and contrast Apache Pig, Apache Hive, and HBase in terms of their architecture, features, and use cases. They should discuss how each tool addresses different data processing requirements and scenarios. • **Module 4:** **Activity: Introduction to R Programming:** Students will learn the basics of R programming language for data analysis and machine learning. They will write R scripts to perform simple data manipulation and visualization tasks. **Assignment: Implementation of Machine Learning Algorithms:** Students will implement supervised and unsupervised machine learning algorithms (e.g., decision trees, clustering) using R programming language. They should apply these algorithms to analyze a given dataset and interpret the results.	

References:

1. Acharya, S., & Chellappan, S. (2015). Big Data Analytics. Wiley.
2. Berthold, M., & Hand, D. J. (2007). Intelligent Data Analysis. Springer.

3. Franks, B. (2012). Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics. John Wiley & Sons.
4. Liebowitz, J. (2013). Big Data and Business Analytics. Auerbach Publications, CRC Press.
5. Minelli, M., Chambers, M., & Dhiraj, A. (2013). Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses. Wiley Publications.
6. Myat, G. J. (2007). Making Sense of Data. John Wiley & Sons.
7. Plunkett, T., & Hornick, M. (2013). Using R to Unlock the Value of Big Data: Big Data Analytics with Oracle R Enterprise and Oracle R Connector for Hadoop. McGraw-Hill/Osborne Media, Oracle Press.
8. Rajaraman, A., & Ullman, J. D. (2012). Mining of Massive Datasets. Cambridge University Press.
9. Sathi, A. (2012). Big Data Analytics: Disruptive Technologies for Changing the Game. MC Press.
10. Warden, P. (2011). Big Data Glossary. O'Reilly.
11. White, T. (2012). Hadoop: The Definitive Guide (3rd ed.). O'Reilly Media.
12. Zikopoulos, P., DeRoos, D., Parasuraman, K., Deutsch, T., Giles, J., & Corrigan, D. (2012). Harness the Power of Big Data: The IBM Big Data Platform. Tata McGraw Hill Publications.

SN	Courses, Modules and Outcomes	Course Contents	Cr
	Semester III		
315512	**Machine Learning** **Major (Core) Theory**		**4**
	Course Outcomes: **Learners will be able to:** • Apply knowledge of machine learning by recognizing real-world examples and applications to understand its practical impact and potential. • Analyse the principles of unsupervised learning to evaluate its methodologies and applications in various contexts. • Evaluate concepts of subset selection for dimensionality reduction, understanding its importance in enhancing model performance and efficiency. • Design strategies to handle multiclass classification using One vs One and One vs Rest approaches, applying these techniques to optimize classification tasks.		
Module 1	**Introduction**		1
	LOs: Learners will be able to • Apply the role of machine learning in automating tasks and making predictions to enhance efficiency and accuracy in various applications. • Analyse the importance of splitting data into training and testing sets, evaluating its impact on model evaluation and performance.	**Module Contents:** • Introduction**:** What is Machine Learning, Examples of Machine Learning applications, Training versus Testing, Positive and Negative Class, Cross- validation. • Types of Learning: Supervised, Unsupervised and Semi-Supervised Learning. • Dimensionality Reduction: Introduction to Dimensionality Reduction, Subset Selection, Introduction to Principal Component Analysis.	
Module 2	**Binary and Multiclass Classification:**		1
	LOs: Learners will be able to • Apply the concept of a confusion matrix to define and interpret classification results, understanding its role in performance evaluation. • Analyse how true positives, true negatives, false positives, and false negatives contribute to classification assessment, evaluating their impact on overall model accuracy	**Module Contents:** • Assessing Classification Performance, Handling more than two classes, Multiclass Classification-One vs One, One vs Rest Linear Models: Perceptron, Support Vector Machines (SVM), Soft Margin SVM, Kernel methods for non-linearity	

	and effectiveness.		
Module 3	**Regression**		1
	LOs: Learners will be able to • Apply knowledge of error metrics to interpret their implications in the context of regression problems, understanding their significance in model evaluation. • Analyse factors that contribute to overfitting in regression models, evaluating their impact on model performance and generalization.	**Module Contents:** • Assessing performance of Regression – Error measures, Overfitting and Underfitting, Catalysts for Overfitting, VC Dimensions • Linear Models: Least Square method, Univariate Regression, Multivariate Linear Regression, Regularized Regression - Ridge Regression and Lasso • Theory of Generalization: Bias and Variance Dilemma, Training and Testing Curves Case Study of Polynomial Curve Fitting.	
Module 4	**LOGIC BASED AND ALGEBRAIC MODELS, TRENDS IN MACHINE LEARNING**		1
	LOs: Learners will be able to • Apply decision-making processes based on instance proximity to enhance model accuracy in classification and regression tasks. • Analyze decision trees and their significance in both classification and regression tasks. • Evaluate the process of tree construction and decision-making to ensure the robustness of models. • Design strategies for building decision trees and making informed decisions based on their outcomes to improve model performance.	**Module Contents:** • Distance Based Models: Neighbors and Examples, Nearest Neighbor Classification, Distance based clustering algorithms - K-means and K-medoids, Hierarchical clustering. • Rule Based Models: Rule learning for subgroup discovery, Association rules mining – Apriori Algorithm, Confidence and Support parameters. • Tree Based Models: Decision Trees, Minority Class, Impurity Measures – Gini Index and Entropy, Best Split • Ensemble Learning: Combining Multiple Models, Bagging, Randomization, Boosting, Stacking	

		• Reinforcement Learning: Exploration, Exploitation, Rewards, Penalties • Deep Learning: The Neuron, Expressing Linear Perceptron as Neurons, Feed Forward Neural Networks, Linear Neurons and their Limitations, Sigmoid, Tanh and ReLU Neurons	
Assignments/ Activities towards CCE			
	• **Module 1:** **Activity: Machine Learning Application Exploration:** Students will research and present on various machine learning applications across different domains (e.g., healthcare, finance, autonomous vehicles). Each student or group will choose a specific application, describe its significance, and discuss how machine learning techniques are applied. **Assignment: Dimensionality Reduction Report**: Students will write a report explaining the concept of dimensionality reduction. They should include an overview of subset selection and principal component analysis (PCA), providing examples and discussing the importance of dimensionality reduction in machine learning. • **Module 2:** **Activity: Classification Performance Assessment:** Students will use a machine learning library (e.g., Scikit-learn) to implement and assess the performance of binary and multiclass classification models. They will work with a provided dataset to evaluate models using cross-validation, confusion matrices, and performance metrics such as accuracy, precision, and recall. **Assignment: SVM and Kernel Methods Implementation:** Students will implement Support Vector Machines (SVM) and explore kernel methods for handling non-linearity. They should write a report detailing their implementation process, experiments with different kernels, and the results obtained. • **Module 3:** **Activity: Regression Model Implementation:** Students will implement linear regression models using the least squares method. They will work on univariate and multivariate regression problems, applying regularization techniques such as ridge regression and lasso to prevent overfitting. **Assignment: Bias-Variance Analysis:** Students will conduct an experiment to analyze the bias-variance tradeoff. They will use polynomial curve fitting on a given dataset and generate training and testing curves. The assignment should include a detailed explanation of their findings and the impact of model complexity on generalization.	

	• **Module 4:** **Activity: Clustering Algorithm Exploration:** Students will implement distance-based clustering algorithms (e.g., K-means, hierarchical clustering) and visualize the results on a given dataset. They will compare the performance and behavior of different clustering methods. **Assignment: Decision Tree and Ensemble Learning Analysis:** Students will implement decision tree models and explore ensemble learning techniques such as bagging, boosting, and stacking. They should analyze the performance improvements achieved through ensemble methods and write a report discussing their findings and observations.	

References:

1. Murphy, K. P. (2012). Machine Learning: A Probabilistic Perspective. MIT Press.
2. Hastie, T., Tibshirani, R., & Friedman, J. (2009). The Elements of Statistical Learning. Springer.
3. Barber, D. (2012). Bayesian Reasoning and Machine Learning. Cambridge University Press. [Online version available]
4. Mitchell, T. (2017). Machine Learning (1st ed.). McGraw Hill.
5. Duda, R. O., Hart, P. E., & Stork, D. G. (2007). Pattern Classification. John Wiley & Sons.
6. Alpaydin, E. (2014). Introduction to Machine Learning (3rd ed.). MIT Press.

SN	Courses, Modules and Outcomes	Course Contents	Cr
	Semester III		
315513	**DATA SCIENCE** **Major (Core) Theory**		**2**
	Course Outcomes: **Learners will be able to:** • Apply understanding of Data Science concepts in real-world data analysis tasks. • Analyze data collection and handling techniques, including API usage, for efficient data processing. • Evaluate statistical concepts and terminologies to interpret data accurately. • Design implementations of Naive Bayes classification algorithms for effective data classification and prediction.		
Module 1	**Introduction to core concepts and technologies, Data collection and management, Data analysis:**		1
	LOs: Learners will be able to • Apply fundamental concepts and terminologies of data science in problem-solving scenarios. • Analyze practical applications of data science across industries to understand its relevance in real-world contexts. • Evaluate different methods of collecting data, emphasizing the use of APIs for efficient data acquisition. • Design implementations of the Naive Bayes algorithm for probabilistic classification tasks, considering its practical applications and limitations.	**Module Contents:** • Introduction, Terminology, data science process, data science toolkit, Types of data, Example applications. • Introduction, Sources of data, Data collection and APIs, Exploring and fixing data, Data storage and management, Using Multiple data sources. • Introduction, Terminology and concepts, Introduction to statistics, Central tendencies and distributions, Variance, Distribution properties and arithmetic, Samples/CLT. • Basic machine learning algorithms, Line a regression, SVM, Naive Bayes.	
Module 2	**Data visualization and applications:**		1
	LOs: Learners will be able to • Apply knowledge to identify and categorize various types of data visualizations for effective communication. • Analyze technologies and tools used in data visualization to select appropriate ones for specific tasks. • Evaluate methods and tools used in developing applications for data science to ensure efficient and accurate data processing. • Design data visualization techniques and application development strategies	**Module Contents:** • Introduction, Types of data visualization, Data for visualization: Data types, Data encodings, Retinal variables, Mapping variables to encodings, Visual encodings. • Applications of Data Science, Technologies for visualization. • Recent trends in various data collection and analysis techniques, various visualization techniques, application development methods of used in data science.	

	to enhance data analysis and interpretation in diverse contexts.		
Assignments/ Activities			
	These assignments aim to apply theoretical concepts to practical application and critical thinking. **Module 1:** Activity: Statistical Analysis Exercise • Select a dataset (e.g., from Kaggle, UCI Machine Learning Repository). • Perform statistical analysis to compute central tendencies (mean, median, mode), variance, and standard deviation. • Visualize the distribution of data and demonstrate understanding of distribution properties. • Conduct a simple experiment to demonstrate the Central Limit Theorem (e.g., sampling from a non-normal distribution and showing the sampling distribution of the mean). **Module 2:** Activity: Data Visualization Project • Select a dataset and identify key variables for visualization. • Create different types of visualizations (e.g., bar charts, scatter plots, heatmaps) using a visualization tool (e.g., Matplotlib, Seaborn, Tableau). • Experiment with various data encodings and retinal variables to effectively communicate the data insights. • Document the visualization choices and the rationale behind them.		

References:

1. O'Neil, C., & Schutt, R. (2013). Doing data science: Straight talk from the frontline. O'Reilly Media.
2. Leskovec, J., Rajaraman, A., & Ullman, J. (2014). Mining of massive datasets (2nd ed.). Cambridge University Press.

SN	Courses, Modules and Outcomes	Course Contents	Cr
	Semester III		
315524	**Big Data Analytics Lab: Practical** **Major (Core)**		**2**
.	**Course Outcomes:** **Learners will be able to:** • Apply Big Data analytics to optimize business decisions and create competitive advantage. • Analyze the business benefits derived from unstructured data. • Evaluate the architectural concepts of Hadoop and the MapReduce paradigm. • Design Big Data applications for streaming data using Apache Spark, incorporating tools like PIG and HIVE in the Hadoop ecosystem.		
Module 1	**Exploring Big Data with Hadoop**		1
	LOs: Learners will be able to • Apply big data tools and platforms, such as Hadoop, for data storage, retrieval, and processing. • Analyze and implement distributed computing techniques to efficiently handle and analyze massive datasets. • Evaluate proficiency in managing an Apache Hadoop cluster and using MapReduce. • Design scalable solutions for big data challenges using advanced big data technologies.	**Module Contents:** • Perform setting up and Installing Hadoop in its two operating modes: 1)Pseudo distributed, 2)Fully distributed. • Use web based tools to monitor your Hadoop setup. Implement the following file management tasks in Hadoop: 1)Adding files and directories 2)Retrieving files 3)Deleting files • Benchmark and stress test an Apache Hadoop cluster Run a basic Word Count Map Reduce program to understand Map Reduce Paradigm. 1)Find the number of occurrences of each word appearing in the input file(s) 2)Performing a MapReduce Job for word search count (look for specific keywords in a file) • Stop word elimination problem: 1)Input: A large textual file containing one sentence per line. A small file containing a set of stop words (One stop word per line) 2)Output: A textual file containing the same sentences of the large input file without the words appearing in the small file.	

Module 2	**MapReduce Implementation**		1
	LOs: Learners will be able to • Develop MapReduce programs to process and analyze semi-structured, record-oriented data. • Implement algorithms to calculate average, maximum, and minimum temperatures for each year in a large dataset using MapReduce. • Utilize Pig Latin and Hive to perform various data operations, including sorting, grouping, joining, projecting, and filtering on diverse datasets. • Create, alter, and drop databases, tables, views, functions, and indexes in Hive, demonstrating proficiency in managing data structures in a distributed environment.	**Module Contents:** • Write a Map Reduce program that mines weather data. Weather sensors collecting data every hour at many locations across the globe gather large volume of log data, which is a good candidate for analysis with MapReduce, since it is semi structured and record-oriented. Data available at:https://github.com/tomwhite/hadoop-book/tree/master/input/ncdc/all. 1) Find average, max and min temperature for each year in NCDC data set? 2) Filter the readings of a set based on value of the measurement, Output the line of input files associated with a temperature value greater than 30.0 and store it in a separate file. • Purchases.txt Dataset 1)Instead of breaking the sales down by store, give us a sales breakdown by product category across all of our stores What is the value of total sales for the following categories? Toys, Consumer Electronics 2) Find the monetary value for the highest individual sale for each separate store What are the values for the following stores? Reno, Toledo, Chandler 3)Find the total sales value Across all the stores, and the total number of sales. • Install and Run Pig then write Pig Latin scripts to sort, group, join, project, and filter your data. • Write a Pig Latin scripts for finding TF-IDF value for book dataset (A corpus of eBooks available at: Project Gutenberg) • Install and Run Hive then use Hive to create, alter, and drop databases, tables, views, functions,	

		and indexes. • Install, Deploy & configure Apache Spark Cluster. Run apache spark applications using Scala. • Data analytics using Apache Spark on Amazon food dataset, find all the pairs of items frequently reviewed together. • Write a single Spark application that: ▪ Transposes the original Amazon food dataset, obtaining a PairRDD of the type: ▪ <user_id> → <list of the product_ids reviewed by user_id> ▪ Counts the frequencies of all the pairs of products reviewed together; • Writes on the output folder all the pairs of products that appear more than once and their frequencies. The pairs of products must be sorted by frequency.	
Assignments/ Activities towards Comprehensive Continuous Evaluation			
	Module 1: Activity: Setting Up and Running a Word Count MapReduce Program • Hadoop Setup: • Pseudo Distributed Mode: Follow instructions to set up Hadoop in pseudo-distributed mode on a single node. • Fully Distributed Mode: Set up a fully distributed Hadoop cluster using multiple nodes. • Monitoring: Use web-based tools (like Hadoop's ResourceManager and HDFS NameNode web UIs) to monitor your Hadoop setup. • File Management in Hadoop: • Adding Files and Directories: Use Hadoop HDFS commands to add files and directories. • Retrieving Files: Retrieve files from HDFS. • Deleting Files: Delete files from HDFS. • Benchmark and Stress Testing: • Benchmark the performance of your Hadoop cluster using tools like TestDFSIO or TeraSort. • Perform stress testing to evaluate cluster stability and performance under heavy load. • Word Count MapReduce Program: • Basic Word Count: Write and run a MapReduce program to count the number of occurrences of each word in an input file. • Keyword Search Count: Modify the Word Count program to search for specific keywords and count their occurrences. • Stop Word Elimination: • Input Files: Use a large text file with one sentence per line and a small	

	file containing stop words. • Output File: Create a MapReduce program to output sentences from the large file without the stop words. **Module 2:** Weather Data Analysis Using MapReduce • Weather Data MapReduce Program: • Data Source: Use the NCDC weather dataset available at: NCDC Dataset. • Average, Max, and Min Temperature: Write a MapReduce program to find the average, maximum, and minimum temperatures for each year in the dataset. • Filter Readings: Filter the temperature readings to output lines with temperatures greater than 30.0 and store them in a separate file. • Sales Data Analysis Using MapReduce: • Product Category Breakdown: Use the Purchases.txt dataset to provide a sales breakdown by product category across all stores. Find the total sales value for categories like Toys and Consumer Electronics. • Highest Individual Sale: Determine the highest individual sale for each store (e.g., Reno, Toledo, Chandler). • Total Sales Value: Calculate the total sales value across all stores and the total number of sales.	

References:

1. Marz, N., & Warren, J. (2015). Big Data: Principles and Best Practices of Scalable Realtime Data Systems. Manning Publications.
2. White, T. (2015). Hadoop: The Definitive Guide. O'Reilly Media.
3. Guller, M. (2015). Big Data Analytics with Spark: A Practitioner's Guide to Using Spark for Large Scale Data Analysis. Apress.
4. Srinivasan, S. (2018). Big Data Analytics: Methods and Applications. CRC Press.
5. Gates, A., Thusoo, A., & et al. (2015). Hive: The Definitive Guide. O'Reilly Media.

SN	Courses, Modules and Outcomes	Course Contents	Cr
	Semester III		
325511	**BLOCK CHAIN** **Major (Elective) Theory**		**4**
	Course Outcomes: **Learners will be able to:** • Apply understanding of blockchain concepts, including its decentralized nature, distributed ledger, and cryptographic security features. • Analyze the historical context and evolution of blockchain technology, including the development of the first blockchain. • Evaluate different consensus algorithms, including the Nakamoto consensus, to understand their role in blockchain technology. • Design solutions incorporating concepts of interoperability, portability in Hyperledger Fabric, and sharding to enhance blockchain efficiency and scalability.		
Module 1	**Fundamentals of Blockchain**		1
	LOs: Learners will be able to • Apply principles of distributed databases to design and manage data across multiple network nodes, considering architecture, advantages, and challenges. • Analyze the complexities of consensus in distributed systems and the significance of Byzantine fault tolerance. • Evaluate the concept of ASIC resistance in cryptocurrencies and its implications for mining centralization. • Design secure systems using cryptography principles, incorporating hash functions, digital signatures (ECDSA), memory-hard algorithms, and zero-knowledge proofs to ensure confidentiality, integrity, and authenticity.	**Module Contents:** • Distributed Database, Two General Problem, Byzantine General problem and Fault Tolerance, Hadoop Distributed File System, Distributed Hash Table, ASIC resistance, Turing Complete. • Cryptography: Hash function, Digital Signature - ECDSA, Memory Hard Algorithm, Zero Knowledge Proof.	
Module 2	**Blockchain , Distributed Consensus:**		1

	LOs: Learners will be able to • Apply fundamental concepts of blockchain, including its decentralized nature, distributed ledger, and cryptographic security features. • Analyze the structure and operation of a blockchain network, focusing on nodes, peers, and the peer-to-peer communication model. • Evaluate the differences between private and public blockchains, considering their use cases, access control, and levels of decentralization. • Design blockchain solutions by exploring the Nakamoto consensus and various consensus algorithms, such as Proof of Work, Proof of Stake, and Proof of Burn, to understand their strengths and weaknesses.	**Module Contents:** • Introduction, Advantage over conventional distributed database, Blockchain Network, Mining Mechanism, Distributed Consensus, Merkle Patricia Tree, Gas Limit, Transactions and Fee, Anonymity, Reward, Chain Policy, Life of Blockchain application, Soft & Hard Fork, Private and Public blockchain. • Nakamoto consensus, Proof of Work, Proof of Stake, Proof of Burn, Difficulty Level, Sybil Attack, Energy utilization and alternate.	
Module 3	**Cryptocurrency , Cryptocurrency Regulation:**		1
	LOs: Learners will be able to • Apply knowledge of the historical context and evolution of blockchain technology, including the development of the first blockchain with Bitcoin. • Analyze the construction of the Ethereum blockchain, focusing on its architecture and functionalities. • Evaluate the concept and implementation of smart contracts and their role in Decentralized Autonomous Organizations (DAOs). • Design blockchain applications utilizing Ethereum's smart contracts and DAO principles to create decentralized solutions.	**Module Contents:** • History, Distributed Ledger, Bitcoin protocols - Mining strategy and rewards, Ethereum - Construction, DAO, Smart Contract, GHOST, Vulnerability, Attacks, Sidechain, Namecoin. • Stakeholders, Roots of Bit coin, Legal Aspects-Crypto currency Exchange, Black Market and Global Economy. Applications: Internet of Things, Medical Record Management System, Domain Name Service and future of Blockchain.	
Module 4	**Hyperledger , Scalability and other challenges :**		1
	LOs: Learners will be able to • Apply the features of Hyperledger Fabric, such as	**Module Contents:** • Hyperledger as a protocol :The reference architecture	

	modular architecture and identity management, to blockchain solutions. • Analyze scalability challenges in blockchain networks and how Hyperledger Fabric addresses them. • Evaluate Hyperledger Fabric's privacy, confidentiality, and deterministic transaction mechanisms. • Design interoperable and portable applications using Hyperledger Fabric to ensure seamless integration with other systems.	Requirements and design goals of Hyperledger Fabric: The modular approach Privacy and confidentiality, Scalability, Deterministic transactions Identity, Auditability Interoperability Portability Rich data queries Fabric Hyperledger Fabric Membership services Blockchain services Consensus services Distributed ledger ,The peer to peer protocol Ledger storage Chaincode services ,Components of the fabric • Scalability and Other Challenges: Scalability Network plane ,Consensus plane, Storage plane View plane ,Block size increase ,Block interval reduction Invertible Bloom, Lookup Tables Sharding State channels Private blockchain, Proof of Stake Sidechains Subchains Tree chains (trees) Block propagation Bitcoin-NG, Plasma ,Privacy Indistinguishability Obfuscation Homomorphic encryption ,Zero-Knowledge Proofs State channels Secure multiparty computation Usage of hardware to provide confidentiality Coin Join Confidential transactions, Mimble Wimble Security Smart contract security Formal verification and analysis Oyente tool	
Assignments/ Activities			
	These assignments aim to apply theoretical concepts to practical application and critical thinking. **Module 1:** Activity: Exploring Cryptographic Concepts and Distributed Systems. • Distributed Database and Fault Tolerance: • Distributed Database: Write a short essay explaining the concept of distributed databases and how they differ from centralized databases. • Two General Problem & Byzantine General Problem: Create a diagram and a brief explanation of the Two General Problem and the Byzantine General Problem. • Fault Tolerance: Research and summarize fault tolerance mechanisms in distributed systems, specifically referencing Hadoop Distributed File System (HDFS) and Distributed Hash Table (DHT). • Cryptography: • Hash Function: Implement a simple hash function in Python and demonstrate its use with various inputs. • Digital Signature - ECDSA: Write a program to generate a digital signature using Elliptic Curve Digital Signature Algorithm (ECDSA).		

	• Memory Hard Algorithm: Explain the concept of memory-hard algorithms and their importance in blockchain. • Zero Knowledge Proof: Write a report on zero-knowledge proofs, including a simple example to illustrate the concept. **Module 2:** Activity: Blockchain Network and Consensus Mechanisms • Blockchain Network: • Write a detailed report on the advantages of blockchain over conventional distributed databases, focusing on network, mining mechanisms, and distributed consensus. • Implement a basic blockchain network in a programming language of your choice, simulating transactions, fees, and mining rewards. • Consensus Mechanisms: • Merkle Patricia Tree: Explain the Merkle Patricia Tree and its role in blockchain. • Proof of Work (PoW): Implement a simple PoW algorithm and simulate mining. • Proof of Stake (PoS): Write a brief report on PoS and its differences from PoW. • Sybil Attack: Research and present strategies to mitigate Sybil attacks in blockchain networks. **Module 3:** Activity: Cryptocurrency Analysis and Regulation • Cryptocurrency History and Protocols: • Write a timeline of the history of cryptocurrency, highlighting key events and developments. • Analyze Bitcoin protocols, focusing on mining strategy and rewards. • Investigate Ethereum's construction, DAO, smart contracts, and notable attacks (e.g., GHOST, sidechain attacks). • Regulation and Legal Aspects: • Write an essay on the legal aspects of cryptocurrency, including regulation, cryptocurrency exchanges, and the impact on the global economy. • Analyze case studies on the use of cryptocurrency in the black market and its implications for law enforcement. **Module 4:** Activity: Hyperledger Fabric Implementation and Scalability Solutions • Hyperledger Fabric: • Write a detailed report on the architecture and components of Hyperledger Fabric, including its modular approach, privacy, scalability, and identity management. • Install and configure a basic Hyperledger Fabric network, demonstrating its membership, blockchain, and consensus services. • Scalability and Challenges: • Research and present solutions to scalability challenges in blockchain, focusing on concepts like sharding, state channels, and block propagation. • Implement a simple example demonstrating the use of state channels or sharding in a blockchain network.	

References:

1. Antonopoulos, A. M. (2014). Mastering Bitcoin: Unlocking digital cryptocurrencies. O'Reilly Media.
2. Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system. Link to the Bitcoin Whitepaper
3. Wood, G. (2014). Ethereum: A secure decentralized transaction ledger (Yellow paper). Link to the Ethereum Yellow Paper
4. Atzei, N., Bartoletti, M., & Cimoli, T. (2017). A survey of attacks on Ethereum smart contracts.
5. Bashir, I. (2018). Mastering blockchain. Wiley.

SN	Courses, Modules and Outcomes	Course Contents	Cr
	Semester III		
325512	**GIS AND REMOTE SENSING** **Major (Elective) Theory**		**4**
	Course Outcomes: **Learners will be able to:** • Apply fundamental and technical skills in data acquisition and management. • Analyze spatial data through mapping, visualization, and remote sensing applications. • Evaluate the integration of various technologies for comprehensive spatial analysis. • Design solutions for problem-solving and enhance communication skills through effective data presentation.		
Module 1	**Fundamentals of GIS**		1
	LOs: Learners will be able to • Apply GIS concepts to understand its components and the characteristics of spatial data. • Analyze spatial data maps and attribute data management techniques, focusing on database data models. • Evaluate GIS applications and advancements in database technologies for GIS. • Design strategies for incorporating spatial data into GIS systems for effective decision-making.	**Module Contents:** • Defining GIS, components of GIS, spatial data, spatial data-maps, characteristics, spatial data modeling, attribute data management-database data model, GIS applications and developments in database.	
Module 2	**Input-Output and Data Analysis in GIS**		1
	LOs: Learners will be able to • Apply methods for data input, editing, and integration in GIS. • Analyze data through measurements, queries, reclassification, buffering, map overlay, interpolation, and spatial analysis techniques. • Evaluate analytical modeling in GIS for physical, environmental, and human processes, assessing the effectiveness of different approaches. • Design outputs from GIS, including maps, non-cartographic output, spatial multimedia, and decision support systems.	**Module Contents:** • Data input and editing– methods, editing, integration, Data analysis-measurements, queries, reclassification, buffering, map overlay, interpolation, analysis of surfaces, network analysis, spatial analysis, Analytical modeling in GIS-physical, environment and human processes, output from GIS –maps, non-cartographic output, spatial multimedia, decision support.	

Module 3	**Issues in GIS:**		1
	LOs: Learners will be able to • Apply computer methods for managing and processing spatial data in GIS projects. • Analyze issues in GIS related to data quality, errors, and human and organizational factors. • Evaluate GIS project design and management processes, including problem identification, data model design, implementation, and evaluation. • Design future-oriented GIS solutions by leveraging internet resources and emerging technologies.	**Module Contents:** • Development of computer methods for spatial data, Issues in GIS- data quality and errors, sources of errors, human and organizational issues, GIS project design and management– problem identification, designing a data model, project management, Implementation, evaluation, the future of GIS, Internet resources of GIS.	
Module 4	**Remote Sensing, Global Positioning Systems (GPS)**		1
	LOs: Learners will be able to • Apply principles of remote sensing and system classification to extract information from images. • Analyze imaging characteristics and integrate remote sensing with GIS for comprehensive spatial analysis. • Evaluate GPS accuracy, including differential GPS, and explore its various applications. • Design solutions that integrate GIS and GPS for enhanced geospatial data management and analysis.	**Module Contents:** • Principles of remote sensing, remote sensing system-classification, Imaging, characteristics, extraction of information from images–metric and thematic, Integration of RS and GIS. • Introduction to GPS, Accuracy of GPS, Differential GPS, Applications of GPS, Integration of GIS and GPS.	
Assignments/ Activities			
	These assignments aim to apply theoretical concepts to practical application and critical thinking. **Module 1:** Activity: GIS Components and Applications Exploration • Define GIS and its components, including spatial data and spatial data modeling. • Explore the characteristics of spatial data and its relevance in GIS applications. • Research and present developments in database technologies for GIS. **Module 2:** Activity: Data Input, Editing, and Spatial Analysis • Explore methods and techniques for data input and editing in GIS systems. • Perform data analysis tasks such as measurements, queries, and map		

	overlay. • Implement spatial analysis techniques including buffering, interpolation, and network analysis. **Module 3:** Activity: GIS Project Design and Management • Investigate computer methods for spatial data development and issues related to data quality and errors in GIS. • Analyze human and organizational issues in GIS project design and management. • Discuss the future of GIS technology and explore Internet resources for GIS professionals. **Module 4:** Activity: Integration of Remote Sensing, GPS, and GIS • Study the principles of remote sensing and the classification of remote sensing systems. • Explore methods for extracting information from remote sensing images and integrating them into GIS. • Investigate the principles of GPS, its accuracy, applications, and integration with GIS.	

References :

1. Heywood, I., Cornelius, S., & Carver, S. (2000). An Introduction to Geographical Information Systems. Pearson Education Asia.
2. Lo, C. P., & Yeung, A. (2016). Concepts and Techniques of Geographic Information Systems. PHI.
3. Demers, M. N. (1999). Fundamentals of Geographic Information Systems (2nd ed.). John Wiley & Sons (Asia) Pte Ltd.
4. Razvi, M. (2002). ArcGIS Developer's Guide for Visual Basic Applications. Onword Press.

SN	Courses, Modules and Outcomes	Course Contents	Cr
	Semester III		
325513	**Software Testing** **Major (Elective) Theory**		**4**
	Course Outcomes: **Learners will be able to:** • Apply fundamental principles and concepts of software testing to enhance the software development process. • Analyze various test design techniques, including equivalence partitioning, boundary value analysis, decision tables, and state transition testing, to create effective test cases. • Evaluate different testing methods such as functional, non-functional, unit, integration, system, regression, and acceptance testing for diverse software systems. • Design quality assurance strategies and best practices, emphasizing the importance of testing in the software development lifecycle while upholding ethical and professional responsibilities.		
Module 1	**Overview of Software Testing**		1
	LOs: Learners will be able to • Apply software testing terminologies, methodologies, and life cycles proficiently. • Analyze the economic aspects of testing and its impact on organizational structures. • Evaluate the advantages of structured testing processes and their cost implications. • Design effective test strategies, policies, and risk management plans to meet customer needs.	**Module Contents:** • Software Testing Terminology and Methodology Software Testing Terminology, Software Testing Life Cycle, Writing a Policy for Software Testing, Economics of Testing, Testing – An organizational Issue, Management Support for Software Testing, Fig. of Software Testing Methodology, Risk associated with not meeting customer needs, Developing Test Strategy • Overview of Software Testing Process Advantages of Following a Process, The Cost of Computer Testing, The Seven-Step Software Testing Process • Verification and Validation Verification and Validation (V&V) Activities, Verification, Verification of Requirements, Verification of High –level Design, Verification of Low – level Design, How to Verify Code? ,Validation • Static Testing Inspections, Structured Walkthroughs,	

		Technical Reviews.	
Module 2	**Validation and Regression Testing**		1
	LOs: Learners will be able to • Apply various validation activities, including unit, integration, function, system, and acceptance testing, to ensure software compliance with requirements. • Analyze the differences between progressive and regressive testing, understanding the importance of regression testing for maintaining software quality. • Evaluate regression testing techniques to identify issues from software changes, ensuring stability and reliability. • Design effective regression testing strategies, defining objectives and selecting appropriate types of regression tests throughout the software development life cycle.	**Module Contents:** • Validation Activities Unit Validation Testing, Integration Testing, Function Testing, System Testing , Acceptance Testing • Regression Testing Progressive vs. Regressive Testing, Regression Testing Produces Quality Software, Regression Testability, Objectives of Regression Testing, When is Regression Testing Done? , Regression Testing Types, Defining Regression Test Problem, Regression Testing Techniques.	
Module 3	**Testing Management and Metrics**		1
	LOs: • Apply test management structures to organize and compose effective testing groups for detailed test planning and design. • Analyze the need for software metrics, demonstrating the ability to define, classify, and apply them within the software development life cycle. • Evaluate entities to be measured, focusing on size metrics and their implications on software management. • Design measurement objectives specific to testing, identifying relevant attributes and metrics for monitoring and controlling the testing process.	**Module Contents:** • Test Management Test Organization, Structure of Testing Group, Test Planning, Detailed Test Design and Test Specifications • Software Metrics Need for Software Management, Definition of Software Metrics, Classification of Software Metrics, Entities to be Measured, Size Metrics • Testing Metrics for Monitoring and Controlling the Testing Process Measurement Objectives for Testing, Attributes and Corresponding Metrics in Software Testing, Attributes, Estimation Models for Estimating Testing Efforts (include only topic Halstead Metrics), Test Point Analysis (TPA) – introduction only.	

Module 4	Automation Testing Tool		1
	LOs: Learners will be able to • Evaluate the necessity and significance of test process maturity, demonstrating the ability to measure, assess, and improve test processes within an organization using established maturity models. • Identify the rationale behind automation in testing, categorize various testing tools, and apply criteria for selecting appropriate tools while considering associated costs. • Analyze guidelines for automated testing and gain an overview of commercial testing tools, fostering the skills required for implementing automated testing effectively. • Apply agile methodologies to enhance software testing, recognizing the importance of agility, overcoming inhibitors, and implementing solutions to improve testing processes within an agile framework.	**Module Contents:** • Testing Process Maturity Models Need for Test Process Maturity, Measurement and Improvement of a Test Process, Test Process Maturity Models • Automation and Testing Tools Need for Automation, Categorization of Testing Tools, Selection of Testing Tools, Cost Incurred in Testing Tools, Guidelines for Automated Testing, Overview of Some Commercial Testing Tools Testing Object Oriented Software Object-Oriented Testing • Using Agile Methods to Improve Software Testing The importance of Agility, Building an Agile Testing Process, Agility Inhibitors, Is Improvement Necessary, Compressing Time, Challenges, Solutions , Measuring Readiness , The Seven-Step Process 4.5 Test Plan.	
Assignments/ Activities towards CCE			
	Module 1: Activity: Develop a Comprehensive Test Strategy and Policy • Define and explain key software testing terminologies and methodologies. • Write a detailed policy for software testing, considering the economics of testing and the organizational implications. • Develop a test strategy addressing risk management and customer needs. **Module 2:** Activity: Conduct Validation and Regression Testing • Perform various validation activities, including unit testing, integration testing, function testing, system testing, and acceptance testing. • Differentiate between progressive and regressive testing and explain the significance of regression testing. • Develop and execute a regression testing plan, identifying objectives and appropriate instances for conducting regression tests. **Module 3:** Activity: Design a Test Management Plan and Define Metrics • Organize and structure a testing group, and create a detailed test plan		

	and test design specifications. • Define and classify software metrics, focusing on entities to be measured and size metrics. • Formulate measurement objectives for testing, identifying relevant attributes and corresponding metrics. **Module 4:** Activity: Evaluate and Implement Automation Testing Tools • Study the need for test process maturity and models for measurement and improvement. • Evaluate various automation testing tools, considering cost and guidelines for their selection and use. • Implement automation testing tools on a sample project, and assess their impact on testing efficiency and effectiveness.	

References:

1. Chauhan, N. (2016). Software testing principles and practices. Oxford University Press.
2. Perry, W. E. (2006). Effective methods of software testing (3rd ed.). Wiley, India.
3. Desikan, S., & Ramesh, G. (2005). Software testing principles and practices. Pearson Education.
4. Patton, R. (2005). Software testing (2nd ed.). Pearson Education.
5. Dustin, E. (2002). Effective software testing: 50 specific ways to improve your testing. Pearson Education.

SN	Courses, Modules and Outcomes	Course Contents	Cr
	Semester III		
325514	**Robotic Process Automation** **Major(Elective) Theory**		**4**
	Course Outcomes: **Learners will be able to:** • Apply fundamental concepts and principles of Robotic Process Automation. • Analyze and gain proficiency in using popular RPA tools like UiPath, Automation Anywhere, and Blue Prism. • Design and develop RPA bots to automate specific tasks and processes. • Evaluate and troubleshoot common issues during RPA implementation.		
Module 1	**Robotic Process Automation Foundations, UiPath, Automation Anywhere**		1
	LOs: Learners will be able to • Apply understanding of RPA fundamentals, capabilities, and components. • Analyze the benefits and downsides of RPA, and compare it with other business technologies. • Evaluate and compare RPA with other business technologies, focusing on Automation Anywhere. • Design automation solutions using RPA tools, particularly Automation Anywhere.	**Module Contents:** • What is RPA, Flavors of RPA, History of RPA, What can RPA do, Components of RPA, The Benefits of RPA, The Downsides of RPA, RPA Compared to BPO, BPM, BPA, What is the Difference Between AI and RPA, RPA Tools and Platforms, Consumer Willingness for Automation, The Workforce of the Future • What is UiPath, UiPath Studio, UiPath Robot, UiPath Orchestrator, UiPath – an integrated view • What is Automation Anywhere, Enterprise Control Room, IQ Bot.	
Module 2	**Downloading and Installing UiPath Studio and Data Manipulation**		1
	LOs: Learners will be able to • Apply UiPath Studio to create automation workflows using the task recorder and step-by-step examples. • Analyze and implement control flow activities, including loops and decision-making, using sequences and flowcharts. • Evaluate the use of variables, collections, and arguments to manage data within automation projects. • Design and execute data table operations, including file management and CSV/Excel integrations, to enhance workflow efficiency.	**Module Contents:** • Learning UiPath Studio, Task Recorder, Step by step examples using the recorder • Sequencing the workflow, Activities, Control flow, various types of loops, and decision making, Step by step example using Sequence, Flowchart and Control Flow, Log Message. • Variables and scope, Collections, Arguments – purpose and use, Data table usage with examples, Clipboard management, File operation with step-by-step example, CSV/Excel to data table and vice	

		versa.	
Module 3	**Taking Control of the Controls, Exception Handling and Debugging**		1
	LOs: Learners will be able to • Apply techniques for finding and interacting with UI controls, utilizing UiExplorer, screen scraping, and OCR to avoid failure points. • Analyze the use of various plugins and extensions for automating tasks across different platforms such as SAP, Java, Citrix, and web applications. • Evaluate the creation and monitoring of assistant bots, including system event and image triggers, for efficient task automation. • Design robust exception handling strategies, implement logging, debugging techniques, and error reporting to ensure reliable automation workflows.	**Module Contents:** • Finding and attaching windows, Finding the control, Techniques for waiting for a control, Act on controls – mouse and keyboard activities, Working with UiExplorer, Handling events, Revisit recorder, Screen Scraping, When to use OCR, Types of OCR available, Avoiding typical failure points • Terminal plugin, SAP automation, Java plugin, Citrix automation, Mail plugin, PDF plugin, Web integration, Excel and Word plugins, Credential management, Extensions – Java, Chrome, Firefox and Silverlight • What are assistant bots, Monitoring system event triggers, Monitoring image and element triggers, Launching an assistant bot on a keyboard event. • Exception handling, Common exceptions and ways to handle them, Logging and taking screenshots, Debugging techniques, Collecting crash dumps, Error reporting	
Module 4	**Managing and Maintaining the Code.**		1
	LOs: Learners will be able to • Apply project organization techniques and updates effectively. • Analyze and implement reusability of workflows using state machines. • Evaluate the use of configuration files and orchestration servers for controlling bots. • Design, publish, and manage automation projects efficiently.	**Module Contents:** • Updates Project organization, Nesting workflows, Reusability of workflows, Commenting techniques, State Machine, When to use Flowcharts, State Machines or Sequences, Using config files and examples of a config file, Integrating a TFS Server • Publishing using publish utility, Overview of Orchestration Server, Using Orchestration Server to Control bots, Using Orchestration server to deploy bots, License management, Publishing and managing	

Assignments/ Activities		
	These assignments aim to apply theoretical concepts to practical application and critical thinking. **Module 1:** Assignment: Compare and Contrast RPA Tools • Research the history, components, benefits, and downsides of RPA. • Compare RPA to BPO, BPM, BPA, and AI. • Identify the key features of UiPath and Automation Anywhere. • Prepare a report summarizing the comparison between UiPath and Automation Anywhere in terms of features, ease of use, and integration capabilities. • Downloading and Installing UiPath Studio and Data Manipulation **Module 2:** Assignment: Build a Simple Automation Workflow • Download and install UiPath Studio. • Create a simple automation workflow using the Task Recorder. • Use sequences and flowcharts to structure the workflow. • Incorporate control flows, loops, decision making, and log messages. • Demonstrate data manipulation using variables, collections, arguments, and data tables. • Perform file operations and demonstrate CSV/Excel data handling. • Taking Control of the Controls, Exception Handling and Debugging **Module 3:** Assignment: Develop a Comprehensive UI Automation • Utilize UiExplorer to find and attach windows and controls. • Implement techniques for waiting for a control, and perform mouse and keyboard activities. • Create a workflow that incorporates screen scraping and OCR. • Use plugins (e.g., Terminal, SAP, Java, Citrix, Mail, PDF, Web, Excel, and Word) to enhance automation. • Develop assistant bots triggered by system events or keyboard events. • Implement exception handling and debugging techniques. • Create a detailed report on the common exceptions and methods to handle them, including logging, taking screenshots, and error reporting. • Managing and Maintaining the Code **Module 4:** Assignment: Project Organization and Deployment • Organize a project using updates, nesting workflows, and commenting techniques. • Demonstrate the reusability of workflows using state machines, flowcharts, and sequences. • Create and use configuration files within a project. • Integrate a TFS server for version control. • Publish the project using the publish utility and manage it using the Orchestration Server. • Deploy bots via the Orchestration Server and manage licenses. • Prepare a documentation report detailing the project organization, deployment process, and best practices for maintaining the code.	

References:

1. Tripathi, A. M. (2018). Learning Robotic Process Automation. Packt Publishing.

2. Taulli, T. (2020). The Robotic Process Automation Handbook: A Guide to Implementing RPA Systems. Apress.
3. Sireci, J. (2020). The Practitioner's Guide to RPA. Farchair Solutions.
4. Bornet, P., Barkin, I., & Wirtz, J. (2021). Intelligent Automation: Welcome to the World of Hyperautomation.

SN	Courses, Modules and Outcomes	Course Contents	Cr
	Semester IV		
415511	**Deep Learning** **Major(Core) Theory**		**4**
	Course Outcomes: **Learners will be able to:** • Apply the implementation of a perceptron, comprehending input features, weights, bias, and activation functions. • Analyze deep learning concepts, including activation functions and forward propagation. • Evaluate convolution operations, padding, stride, and batch processing, and implement convolution and pooling layers in TensorFlow. • Design and implement various RNN architectures, such as one-to-one, one-to-many, many-to-one, and many-to-many, for training.		
Module 1	**Introduction to Deep Learning**		1
	LOs: Learners will be able to • Apply the knowledge of building a perceptron by defining input features, weights, bias, and activation functions. • Analyze the limitations of a single-layer perceptron, particularly its inability to learn non-linear relationships. • Evaluate the structure of artificial neural networks, including input, hidden, and output layers. • Design activation functions to introduce non-linearity, facilitating the learning of complex patterns by neural networks.	**Module Contents:** • Perceptron: What is a Perceptron? Implementing perceptron, Introducing & Implementing Weights & Bias, Multilayer Perceptron, Limitations of perceptron. • Introduction to Deep Learning: What is deep learning? Biological and artificial neurons, ANN and its layers, Input layer, Hidden layer, Output layer, exploring activation functions, the sigmoid function, the tanh function, The Rectified Linear Unit function, The leaky ReLU function, The Swish function, The softmax function, Forward propagation in ANN, How does ANN learn?	
Module 2	**Convolutional Neural Networks:**		1
	LOs: Learners will be able to • Apply TensorFlow's representation of computations as directed acyclic graphs (DAGs) to analyze and optimize neural network architectures. • Analyze the concept of sessions in TensorFlow for executing operations within a computational graph efficiently. • Evaluate the general architecture of Convolutional Neural Networks (CNNs), including convolutional	**Module Contents:** • Getting to Know TensorFlow • What is TensorFlow? Understanding computational graphs and sessions, Sessions, Variables, constants, and placeholders, Introducing TensorBoard, Creating a name scope. • Back propagation Algorithm, Neural Network Training, • Convolutional Neural Networks: • Overall Architecture, The	

	layers, pooling layers, and fully connected layers. • Design and implement convolutional and pooling layers within a CNN architecture to process and extract features from input data effectively.	Convolution Layer, Issues with the Fully Connected Layer, Convolution Operations, Padding, Stride, Batch Processing, The Pooling Layer, Implementing a Convolution Layer, Implementing a Pooling Layer, Implementing a CNN, Visualizing a CNN.	
Module 3	**Optimizers in DL**		1
	LOs: Learners will be able to • Apply gradient descent as the optimization algorithm to minimize loss functions during training of neural networks. • Analyze adaptive learning rates based on historical gradients to enhance training efficiency. • Evaluate the challenges associated with training Recurrent Neural Networks (RNNs) and strategies for managing sequential dependencies. • Design backpropagation through time as the algorithm to train RNNs by unfolding them into computational graphs over time.	**Module Contents:** • Optimizers in DL: Gradient Descent, Stochastic Gradient Descent, Mini-Batch Gradient Descent, SGD with Momentum, AdaGrad (Adaptive Gradient Descent), RMS-Prop (Root Mean Square Propagation), AdaDelta, Adam (Adaptive Moment Estimation). • Introducing RNNs: • RNN implementation and training, Backpropagation through time, Vanishing & exploding gradients, long short-term memory LSTM, • Different types of RNN architectures: • One-to-one architecture • One-to-many architecture • Many-to-one architecture • Many-to-many architecture.	
Module 4	**Deep Unsupervised Learning**		1
	LOs: Learners will be able to • Apply the concept of autoencoders for unsupervised learning, encoding, and decoding input data. • Analyze Generative Adversarial Networks (GANs) as frameworks for generative model training via adversarial training. • Evaluate the utility of different models across various scenarios. • Design neural network architectures tailored to specific unsupervised learning tasks.	**Module Contents:** • Deep Unsupervised Learning: Auto encoders (standard, sparse, denoising, contractive, etc), Variational Auto encoders, • Deep Generative Models GANS.	
Assignments/ Activities			
	These assignments aim to apply theoretical concepts to practical application and critical thinking. **Module 1:** • Implement a multilayer perceptron (MLP) using Python and NumPy. Start by defining a simple perceptron with one hidden layer and demonstrate		

	its limitations in solving non-linear problems. Explore different activation functions (sigmoid, tanh, ReLU, etc.) and analyze their impact on the model's performance. Experiment with forward propagation to understand how artificial neural networks (ANNs) learn from data. **Module 2:** • Develop a simple Convolutional Neural Network (CNN) using TensorFlow. Begin by understanding TensorFlow basics, such as computational graphs, sessions, and variables. Implement the convolution and pooling layers of the CNN architecture and visualize the learned features using TensorBoard. Experiment with different configurations of convolutional and pooling layers to observe their effects on model performance. **Module 3:** • Compare and evaluate different optimization algorithms in deep learning. Implement gradient descent, stochastic gradient descent (SGD), and variations like SGD with momentum, AdaGrad, RMSProp, AdaDelta, and Adam. Analyze their convergence rates and effects on training neural networks. Additionally, explore the challenges of training Recurrent Neural Networks (RNNs) such as vanishing/exploding gradients and implement long short-term memory (LSTM) units to address them. **Module 4:** • Experiment with different types of unsupervised learning algorithms. Implement autoencoders, including standard, sparse, denoising, and contractive autoencoders, using TensorFlow. Explore the concept of variational autoencoders (VAEs) and their applications in generating new data samples. Additionally, implement Generative Adversarial Networks (GANs) to generate synthetic data and evaluate their performance in comparison to traditional autoencoders.	

References:

1. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning (Adaptive Computation and Machine Learning series). The MIT Press.
2. Chollet, F. (2018). Deep Learning with Python. Manning.
3. Buduma, N., & Locascio, N. (2017). Fundamentals of Deep Learning: Designing Next-Generation Machine Intelligence Algorithms. O'Reilly Media.

SN	Courses, Modules and Outcomes	Course Contents	Cr
	Semester IV		
415512	**NATURAL LANGUAGE PROCESSING** **Major (Core) Theory**		**4**
	Course Outcomes: **Learners will be able to:** • Apply the knowledge of various building blocks of Natural Language Processing (NLP) to design algorithms for solving NLP problems effectively. • Analyze techniques used in machine translation to comprehend and implement translation models. • Evaluate different tools available for NLP and select the most suitable ones for specific tasks based on their functionalities and performance. • Design workflows incorporating these tools to address various NLP challenges efficiently.		
Module 1	**Introduction**		1
	LOs: Learners will be able to • Apply the understanding of NLP's significance to modern computing and communication systems to design solutions that leverage its capabilities effectively. • Analyze the distinctions between different levels of language processing and their respective roles in NLP to develop comprehensive NLP systems. • Evaluate the key issues and limitations in NLP to devise strategies for overcoming challenges and improving system performance. • Design NLP applications tailored to specific domains by examining diverse use cases and selecting appropriate techniques and tools for tasks such as machine translation, sentiment analysis, and information retrieval.	**Module Contents:** • Need for processing of natural languages, Language processing levels, Issues and challenges in NLP, History, Classical approaches to NLP with knowledge bases and linguistic rules. Introduction to formal languages, finite state automata and regular expressions. • Applications of NLP.	
Module 2	**Morphology and Phonology**		1
	LOs: Learners will be able to • Apply the knowledge of	**Module Contents:** • Morphology fundamentals,	

	inflectional and derivational morphology to analyze word formation processes across languages and their structural implications. • Analyze phonetic fundamentals, such as phonemes and phonological rules, to understand the sound structure of languages and its variations. • Evaluate the role of inflectional and derivational morphology in linguistic analysis and language processing tasks. • Design linguistic analysis tools and algorithms that leverage morphological and phonetic principles to enhance language understanding and processing capabilities.	Inflectional and Derivational morphology, Morphological parsing, Finite State transducers, N- gram language models, phonetics fundamentals, phoneme and phonological rules, machine learning of phonology, phonological aspects of prosody and speech synthesis.	
Module 3	**Part-of-Speech Tagging and Parsing:**		1
	LOs: Learners will be able to • Recognize the significance of word classes in linguistic analysis and natural language understanding, defining and categorizing them accordingly. • Analyze parsing strategies like top-down and bottom-up parsing to understand their advantages and limitations in syntactic analysis. • Apply finite state parsing methods to process sequential structures in language, demonstrating an understanding of their utility.	**Module Contents:** • Word Classes, Part of speech tagging, Tagsets, Rule based, Stochastic and Transformation based POS tagging. Basic parsing strategies, top-down parsing, bottom up parsing, parsing with context free grammars, a basic top down parser, Earley parser, CYK parser, Finite state parsing methods, Unification of feature structures.	
Module 4	**Semantic Analysis and Pragmatics:**		1
	LOs: Learners will be able to • Apply knowledge of lexemes to differentiate	**Module Contents:** • Lexical Semantics, Lexemes, Relations among lexemes and their senses, WordNet, Internal structure	

	between them and understand their internal structures and relationships among word senses. • Analyze word sense disambiguation techniques to proficiently determine word meanings within context using computational methods. • Evaluate lexical semantic analysis techniques and the use of WordNet in computational models for tasks like information retrieval, text summarization, and sentiment analysis. • Design computational models integrating lexical semantic analysis techniques and WordNet for improved performance in various natural language processing tasks.	of words, metaphor and metonymy & their computational approaches, Word Sense Disambiguation. • Discourse, Reference resolution, syntactic and semantic constraints on coreference, pronoun resolution reference, text coherence, discourse structure, Dialogue- Acts, structure, conversational agents, Introduction to language generation, architecture, discourse planning.	
Assignments/ Activities			
	These assignments aim to apply theoretical concepts to practical application and critical thinking: **Module 1:** Explore the fundamentals of morphology, distinguishing between inflectional and derivational morphology, and analyze their significance in word formation and structure. **Module 2:** Implement morphological parsing techniques using finite state transducers and N-gram language models, facilitating the understanding and processing of sequential structures in natural language. **Module 3:** Investigate the fundamentals of phonetics, including phonemes and phonological rules, and their application in machine learning for phonology and aspects of prosody and speech synthesis. **Module 4:** Design and develop machine learning models for phonological analysis, focusing on the computational aspects of phonology and its relevance in speech processing and synthesis.		

References:

1. Jurafsky, D., & Martin, J. H. (2009). Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. Pearson Education.

2. Allen, J. (1995). Natural Language Understanding. Addison Wesley.
3. Siddiqui, T., & Tiwary, U. S. (2019). Natural Language Processing and Information Retrieval. Oxford University Press.
4. Handke, J. (2009). The Structure of the Lexicon: Human Versus Machine (Natural Language Processing). Mouton de Gruyter.
5. Bharati, V., Chaitanya, R., & Sangal, R. (2010). Natural Language Processing: A Paninian Perspective. Prentice Hall of India.

SN	Courses, Modules and Outcomes	Course Contents	Cr
	Semester IV		
415513	**Mobile Application Development using Android Programming: LAB Major (Core) Practical**		**2**
	Course Outcomes: **Learners will be able to:** • Apply knowledge of the Android platform architecture to set up the development environment using Android Studio IDE. • Analyse Java code relevant to Android app development and integrate it within the platform architecture. • Evaluate object-oriented programming concepts in Android applications for code efficiency and maintainability. • Design responsive and adaptive layouts for diverse screen sizes and orientations, manage activities as fundamental building blocks, and utilize intents for effective communication between app components.		
Module 1	**Fundamentals of Android Development**		1
	LOs: Learners will be able to • Applying knowledge of Android's evolution, analyse system requirements for setting up the Android Development Environment. • Evaluate the directory structure of an Android project and assess XML's significance in app development. • Design a basic Android application using Android Studio IDE, understanding Android application components' roles. • Analyse Android Services for background task processing and evaluate Content Providers for efficient data management in Android.	**Module Contents:** • What is Android, Android versions and its feature set The various Android devices on the market, The Android Market application store, Android Development Environment-System Requirements, Creating Android Virtual Devices (AVDs) • Android Software Development Platform, The Directory Structure of an Android Project , Common Default Resources Folders, The Values Folder, Leveraging Android XML, Screen Sizes , Launching Your Application: The Android Manifest.xml File, Creating Your First Android Application • Android Application Components, Android Activities: Defining the UI, Android Services: Processing in the Background, Broadcast Receivers: Announcements and Notifications Content Providers: Data Management, Android Intent Objects: Messaging for Components.	
Module 2	**Android Manifest XML**		1
	LOs: Learners will be able to • Applying Android component declaration in the Manifest XML file, analyse UI design for	**Module Contents:** • Android Manifest XML: Declaring Your Components, Designing for	

	diverse devices. • Evaluate the use of Views, View Groups, and Layout Managers for effective UI. • Design mechanisms for user input and implement interactive elements like buttons and check boxes. • Explore multimedia features, including audio/video playback, and configure the emulator for location-based services.	Different Android Devices, Views and View Groups, Android Layout Managers, The View Hierarchy, Designing an Android User Interface using the Graphical Layout Tool • Displaying Text with Text View, Retrieving Data from Users, Using Buttons, Check Boxes and Radio Groups, Getting Dates and Times from Users, Using Indicators to Display Data to Users, Adjusting Progress with Seek Bar, Working with Menus usingviews, Gallery, Image Switcher, GridView, and ImageView views to displayimages, Creating Animation • Intent Overview, Implicit Intents, Creating the Implicit Intent Example Project, Explicit Intents, Creating the Explicit Intent Example Application, Intents with Activities, Intents with Broadcast Receivers, An Overview of Threads, The Application Main Thread, Thread Handlers, A Basic Threading Example, Creating a New Thread, Implementing a Thread Handler, Passing a Message to the Handler. • Sending SMS Messages Programmatically, Getting Feedback after Sending the Message Sending SMS Messages Using Intent Receiving, sending email, Introduction to location-based service, configuring the Android Emulator for Location-Based Services, Map-Based Activities • Playing Audio and Video, Recording Audio andVideo, Using the Camera to Take and Process Pictures.	
Assignments/ Activities towards Comprehensive Continuous Evaluation			
	These assignments aim to apply theoretical concepts to practical application and critical thinking: **Module 1:** • Understanding Android Platform: Research and document the evolution of the Android operating system, its various versions, and key features. • Setting Up Development Environment: Install Android Studio IDE on your system and create Android Virtual Devices (AVDs) for testing. **Module 2:**		

	• Exploring Directory Structure: Analyze the directory structure of an Android project and identify common default resources folders. • Creating Your First Android Application: Design and develop a simple Android application that demonstrates basic functionality using XML layouts and Java programming.	

References:

1. Phillips, B., Stewart, C., Hardy, B., & Marsicano, K. (2017). Android Programming: The Big Nerd Ranch Guide, 3rd Edition. Big Nerd Ranch LLC.
2. Keur, C., & Hillegass, A. (2015). iOS Programming: The Big Nerd Ranch Guide, 6th Edition. Big Nerd Ranch LLC.
3. Urma, R.-G., Fusco, M., & Mycroft, A. (2015). Java 8 in Action: Lambdas, Streams, and Functional-Style Programming. Manning Publications.
4. Evans, B. J., & Verburg, M. (2013). The Well-Grounded Java Developer: Vital Techniques of Java 7 and Polyglot Programming. Manning Publications.
5. Fling, B. (2009). Mobile Design and Development. O'Reilly Media.
6. Firtman, M. (2013). Programming the Mobile Web, 2nd Edition. O'Reilly Media.
7. Crumlish, C., & Malone, E. (2015). Designing Social Interfaces. O'Reilly Media.
8. Muschko, B. (2014). Gradle in Action. Manning Publications.
9. Larman, C. (2004). Applying UML and Patterns: A Guide to Object-Oriented Analysis and Design and Iterative Development, 3rd Edition. Prentice Hall.

SN	Courses, Modules and Outcomes	Course Contents	Cr
	Semester IV		
425511	**Information Security** **Major(Elective) Theory**		**4**
	Course Outcomes: **Learners will be able to:** • Apply knowledge of symmetric ciphers and their mechanisms. • Analyse the OSI Security Architecture's relevance to information security. • Evaluate classical encryption techniques within the symmetric cipher model. • Design principles of public key cryptography, digital signatures, authentication applications like Kerberos and X.500 Authentication Service, and countermeasures against malicious software threats.		
Module 1	**Symmetric Ciphers**		1
	LOs: Learners will be able to • Apply knowledge of fundamental services provided by symmetric ciphers. • Analyse common attacks on symmetric ciphers and defend against them. • Evaluate the OSI Security Architecture's role in network security. • Design secure encryption algorithms using block cipher design principles and explain different modes of operation for secure communication.	**Module Contents:** • **Overview** – Services, Mechanism and Attacks, The OSI Security. Architecture, A model for network security Classical Encryption techniques – Symmetric Cipher model, Substitution. Techniques, Transposition techniques, Rotor Machines, Steganography. Block Cipher and Data Encryption Standard – Simplified DES, Block. Chiper principles, The Data Encryption Standard, The strength of DES, Differential and Linear Cryptanalysis, Block Cipher design principles, Block Cipher mode of Operation	
Module 2	**Asymmetric Ciphers**		1
	LOs: Learners will be able to • Apply principles of public key cryptography and its applications. • Analyse the RSA algorithm, including key management practices. • Evaluate different public key cryptosystems, assessing their strengths and weaknesses. • Design authentication protocols and discuss their role in information security, including message authentication codes and secure hash functions.	**Module Contents:** • Public Key Cryptography and RSA – Principles of Public Key Cryptosystems, The RSA Algorithm Key management ; Other public key cryptosystemsKey Management, Diffe-Hellman Key Exchange, Elliptical Curve Arithmetic, Elliptical curve Cryptography Message Authentication and HASH Functions – Authentication requirements, Authentication Functions, Message Authentication Codes, Hash Functions, security of Hash Functions and MACS Digital Signatures and Authentication	

		Protocols – Digital Signatures, Authentication Protocols, Digital Signature Standard	
Module 3	**Network Security practice**		1
	LOs: Learners will be able to • Apply authentication protocols such as Kerberos and X.500. • Analyse secure email communication using PGP and S/MIME. • Evaluate IPSec architecture and components. • Design SSL/TLS protocols for securing web communication, considering Secure Electronic Transaction (SET) principles for e-commerce.	**Module Contents:** • **Network Security practice** : Authentication Applications – Kerberos, X.500 Authentication Service Electronic Mail Security – Pretty Good Privacy, S/MIME IP Security – IP Security Overview, IP Security Architecture, Authentication Header, Encapsulating security payload, Combining Security Associations, Key Management WEB Security – Web Security Considerations, Secure Socket Layer and Transport Layer Security, Secure Electronic Transaction	
Module 4	**System Security**		1
	LOs: Learners will be able to • Apply intrusion detection systems for threat identification. • Analyse password management policies. • Evaluate countermeasures against viruses. • Design and configure firewalls based on security needs.	**Module Contents:** • System Security : Intruders – Intruders, Intruder detection, Password Management, Malicious Software – Viruses and Related Threats, Virus Countermeasures, Firewall design principles, Trusted system.	
Assignments/ Activities			
	These assignments aim to apply theoretical concepts to practical application and critical thinking. **Module 1:** • Intruder Detection System Design: Design and configure an intrusion detection system (IDS) to detect and respond to unauthorized access attempts and suspicious activities on a network. **Module 2:** • Password Management Policy: Develop and document an effective password management policy that includes guidelines for creating strong passwords, regular password updates, and secure storage practices. **Module 3:** • Malicious Software Countermeasures: Research and propose countermeasures against viruses and related threats, considering techniques such as antivirus software deployment, malware scanning, and user education. **Module 4:** • Firewall Configuration: Design and configure firewall rules based on specific security requirements, considering factors such as network topology, traffic		

	patterns, and permitted services. • Trusted System Implementation: Identify and implement mechanisms to establish and maintain trust in computing environments, including integrity verification, secure boot processes, and software validation.	

References:

1. Stallings, W. (2016). Network Security Essentials. Pearson.
2. Anderson, R. J. (2020). Security Engineering: A Guide to Building Dependable Distributed Systems. Wiley.
3. Pfleeger, C. P., Pfleeger, S. L., & Margulies, J. (2015). Security in Computing. Pearson.
4. Schneier, B. (1995). Applied Cryptography: Protocols, Algorithms, and Source Code in C. Wiley.
5. Murdoch, D., & Lee, R. (2014). Blue Team Handbook: Incident Response Edition. CreateSpace Independent Publishing Platform.

SN	Courses, Modules and Outcomes	Course Contents	Cr
	Semester IV		
425512	**Digital Forensics** **Major (Elective) Theory**		**4**
	Course Outcomes: **Learners will be able to:** • Apply the significance of digital forensics in cybersecurity and its application. • Analyse various methodologies for incident response to mitigate cybersecurity threats effectively. • Evaluate the process of forensic duplication and its implementation for preserving digital evidence. • Design proficiency in forensic analysis of file systems, including fundamentals and techniques for investigating network attacks and live systems.		
Module 1	**Introduction to Digital Forensics**		1
	LOs: Learners will be able to • Apply fundamental concepts of cybercrime to understand its definition and scope. • Analyse different types of cybercrime to distinguish and categorize them effectively. • Evaluate the fundamental concepts of digital forensics to comprehend its role in investigating cybercrimes. • Design an incident response methodology to effectively address and mitigate cybersecurity incidents.	**Module Contents:** • Introduction of Cybercrime: Types, The Internet spawns crime, Worms versus viruses, Computers' roles in crimes, Introduction to digital forensics, Introduction to Incident - Incident Response Methodology – Steps - Activities in Initial Response, Phase after detection of an incident.	
Module 2	**Initial Response and forensic duplication**		1
	LOs: Learners will be able to • Apply the concept of initial response to efficiently address cybersecurity incidents. • Analyse core concepts of forensic duplication to ensure accurate preservation of digital evidence. • Evaluate tools for forensic duplication to choose the most suitable ones for the task. • Design and demonstrate the process of forensic duplication of a hard drive to maintain integrity and authenticity of evidence.	**Module Contents:** • Initial Response & Volatile Data Collection from Windows system - Initial Response & Volatile Data Collection from Unix system – Forensic Duplication: Forensic duplication: Forensic Duplicates as Admissible Evidence, Forensic Duplication Tool Requirements, Creating a Forensic Duplicate/Qualified Forensic Duplicate of a Hard Drive.	

<table>
<tr><td>Module 3</td><td colspan="2">Preserving ,Recovering Digital Evidence and Network forensic</td><td>1</td></tr>
<tr><td></td><td>LOs: Learners will be able to

• Apply the concept of file systems to conduct forensic analysis effectively.
• Analyse storage fundamentals to comprehend data management and retrieval.
• Evaluate evidence handling procedures to maintain integrity and admissibility.
• Design proficiency in intrusion detection and analyse various network attacks for proactive security measures.</td><td>Module Contents:

• File Systems: FAT, NTFS - Forensic Analysis of File Systems – Storage, Fundamentals: Storage Layer, Hard Drives Evidence Handling: Types of Evidence, Challenges in evidence handling, Overview of evidence handling procedure.
• Intrusion detection; Different Attacks in network, analysis Collecting Network Based Evidence - Investigating Routers - Network Protocols - Email Tracing- Internet Fraud.</td><td></td></tr>
<tr><td>Module 4</td><td colspan="2">System Investigation and Law</td><td>1</td></tr>
<tr><td></td><td>LOs: Learners will be able to

• Apply data analysis techniques for both Windows and Unix systems to uncover potential cyber threats.
• Analyse various hacker tools and ethical considerations concerning cybercrime investigations.
• Evaluate legal frameworks pertinent to digital forensics and their implications.
• Design a comprehensive understanding of legal hierarchies and their applications in computer-related laws.</td><td>Module Contents:

• Data Analysis Techniques - Investigating Live Systems (Windows & 08 Unix) Investigating
• Hacker Tools - Ethical Issues – Cybercrime.
• Bodies of law: Constitutional law, Criminal law, Civil law, Administrative regulations, Levels of law: Local laws, State laws, Federal laws, International laws , Levels of culpability: Intent, Knowledge, Recklessness, Negligence Level and burden of proof : Criminal versus civil cases ,Vicarious liability, Laws related to computers: CFAA, DMCA, CAN Spam, etc.</td><td></td></tr>
<tr><td colspan="3">Assignments/ Activities towards CCE</td><td></td></tr>
<tr><td></td><td colspan="2">Module 1:
Assignment: Cybercrime Overview and Incident Response Plan
• Students will research and compile a comprehensive overview of cybercrime types, emphasizing the role of computers in criminal activities. They will then design an incident response plan, outlining the steps and activities involved in initial response and incident detection. The assignment will require students to apply their understanding of digital forensics to propose effective strategies for handling cyber incidents.

Module 2:
Assignment: Volatile Data Collection and Forensic Duplication</td><td></td></tr>
</table>

	• Students will simulate initial response scenarios and practice collecting volatile data from both Windows and Unix systems. They will then demonstrate their understanding of forensic duplication by creating forensic duplicates of hard drives using appropriate tools and techniques. Through this assignment, students will apply their knowledge to ensure the preservation and admissibility of digital evidence. **Module 3:** Assignment: File Systems Analysis and Network Evidence Collection • Students will conduct an in-depth analysis of FAT and NTFS file systems, focusing on forensic techniques for recovering digital evidence. They will explore storage fundamentals and challenges in evidence handling, proposing procedures for preserving and recovering digital evidence. Additionally, students will delve into network forensic techniques, including intrusion detection and collecting network-based evidence such as email tracing and investigating routers. **Module 4:** Assignment: Hacker Tools Investigation and Legal Analysis • Students will investigate various hacker tools and their ethical implications in cybercrime investigations. They will analyze the legal framework surrounding digital forensics, including constitutional, criminal, civil, and administrative laws. Furthermore, students will examine the levels of culpability and burden of proof in criminal and civil cases, along with laws specific to computers such as the Computer Fraud and Abuse Act (CFAA) and the Digital Millennium Copyright Act (DMCA). Through this assignment, students will gain a comprehensive understanding of the legal and ethical considerations in digital forensic investigations.	

References:

1. Mandia, K., & Prosise, C. (2006). Incident Response and Computer Forensics. Tata McGraw-Hill.
2. Stephenson, P. (1999). Investigating Computer Crime: A Handbook for Corporate Investigations.
3. Casey, E. (2001). Handbook Computer Crime Investigation's Forensic Tools and Technology (1st ed.). Academic Press.
4. Skoudis, E., & Perlman, R. (2001). Counter Hack: A Step-by-Step Guide to Computer Attacks and Effective Defenses. Prentice Hall Professional Technical Reference.
5. Zaenglein, N. (2000). Disk Detective: Secrets You Must Know to Recover Information From a Computer. Paladin Press.
6. Nelson, B., Philips, A., & Steuart, C. (n.d.). Guide to Computer Forensics Investigation (4th ed.). Course Technology.

SN	Courses, Modules and Outcomes	Course Contents	Cr
	Semester IV		
425513	**Agile Methodology** **Major(Elective) Theory**		**4**
	Course Outcomes: **Learners will be able to:** • Apply Agile requirement techniques to streamline software development processes. • Analyse various Agile software methodologies to identify the most suitable approach for the project. • Evaluate different Agile estimation techniques to accurately plan project timelines and resource allocation. • Design an Agile testing approach to ensure the quality and functionality of software products throughout the development lifecycle.		
Module 1	**Introduction to Agile Methodologies**		1
	LOs: Learners will be able to • Apply traditional software development methodologies to understand their approach and processes. • Analyse the limitations and challenges of traditional software development methodologies to identify areas for improvement. • Evaluate the concept of Agile methodology as a more flexible and iterative approach to software development. • Design collaborative requirements analysis using the Class Responsibility Collaborator (CRC) method to enhance communication and understanding among stakeholders.	**Module Contents:** • Traditional approach of Software Development Methodology, Need of Agile software Development, Defining Agile,Agile Manifesto Principles of Agile , Values of Agile ,Business Benefits of Agile Software Development • Traditional Requirements Development , Principle of Agile Requirements Development ,Agile Requirements : Epics and User stories ,Difference between Epics and User stories ,Backlog Management, Class Responsibility Collaborator.	
Module 2	**Scrum and Kanban Methodologies**		1
	LOs: Learners will be able to • Apply Scrum framework concepts to understand its role and relevance in Agile software development. • Analyse the advantages and benefits of adopting the Scrum framework to determine its suitability for project needs. • Evaluate the underlying principles of the Scrum framework to ensure alignment with project objectives and values. • Design key artifacts in Scrum, such as the Product Backlog, Sprint Backlog, and Increments, to effectively manage	**Module Contents:** • Introduction to Scrum framework,Advantages of Scrum Framework.Phases of Scrum, Principles of Scrum,Roles: Product owner, team members and scrum master, Scrum Ceremonies :Sprint, sprint planning, daily scrum, sprint review, and sprint retrospective, Artifacts: Product backlog, sprint backlog and increments. • Introduction to Kanban framework, Workflow, Limit the amount of work in progress, pulling work from column to	

	project requirements and deliverables.	column, Kanban board, Adding policies to the board, Cards and their optimization.Kanban Practices , Kanban Flow practices.Work Item Age.Kanban vs Scrum.	
Module 3	**Extreme Programming and Agile Estimation Techniques**		1
	LOs: Learners will be able to • Apply the foundational values and principles of Extreme Programming (XP) to guide software development practices. • Analyse and implement the twelve practices of XP, such as pair programming and test-driven development, to enhance software quality and productivity. • Evaluate the life cycle stages of an XP project, from planning to release, to ensure effective project management and delivery. • Design Agile estimation techniques like Planning Poker and Shirt Sizes to optimize planning processes and enhance project estimation accuracy.	**Module Contents:** • Basic values and principles, Roles, Twelve practices of XP, Pair programming, XP team, Life cycle and tools for XP., Good practices need to be practiced in extreme programming, Advantages of Extreme Programming • Agile Maturity Model and Agile Estimation Techniques - Planning Poker-Shirt Sizes. Dot Voting, Bucket System.	
Module 4	**Agile Testing**		1
	LOs: Learners will be able to • Apply the Agile Testing Quadrants model to classify testing activities into distinct categories. • Analyse the iterative nature of the Agile Testing Life Cycle within Agile development to ensure continuous improvement. • Evaluate the principles and practices of Behavior Driven Development (BDD) as an effective Agile testing technique. • Design Agile test metrics to measure and improve the testing process, utilizing them effectively to assess project progress and identify areas for improvement.	**Module Contents:** • Agile Testing Life Cycle, Agile Testing Quadrants, Agile Testing Techniques: Behavior Driven Development, Test Driven Development Acceptance Test Driven Development Testing.Role of Agile Tester.User stories approach in Acceptance Test Driven Development Testing.Other Techniques - Exploratory Testing , Session Based testing. • Agile Test Metrics.	
Assignments/ Activities			

	These assignments aim to apply theoretical concepts to practical application and critical thinking. **Module 1:** Assignment: Agile Manifesto Analysis • Students will analyze the Agile Manifesto and its principles. They will identify the values and principles of Agile software development and discuss the need for Agile methodologies in contrast to traditional software development approaches. The assignment will require students to critically evaluate the business benefits of Agile software development based on the principles outlined in the manifesto. **Module 2:** Assignment: Scrum vs. Kanban Analysis • Students will compare and contrast the Scrum and Kanban frameworks. They will analyze the advantages of each framework, including their principles, roles, ceremonies, and artifacts. Through this assignment, students will design a comparative analysis highlighting the differences between Scrum and Kanban, including their workflows, work-in-progress limits, and practices. **Module 3:** Assignment: Agile Practices Implementation Plan • Students will design an implementation plan for adopting Extreme Programming (XP) practices within a hypothetical software development team. They will apply Agile estimation techniques such as Planning Poker, Shirt Sizes, Dot Voting, and the Bucket System to plan and execute the adoption of XP practices. The assignment will require students to evaluate the advantages of XP and assess its suitability for different project scenarios. **Module 4:** Assignment: Agile Testing Strategies Proposal • Students will propose Agile testing strategies based on the Agile Testing Quadrants and techniques such as Behavior Driven Development (BDD) and Test Driven Development (TDD). They will design a testing approach for a given software project, considering user stories and acceptance criteria. Additionally, students will evaluate the role of Agile testers and propose Agile test metrics for measuring and improving the testing process.	

References:

1. Stellman, A., & Hart, J. A. (2015). Learning Agile. O'Reilly.
2. Crispin, L., & Gregory, J. (2008). Agile Testing: A Practical Guide for Testers and Agile Teams. Addison Wesley.
3. Schwaber, K., & Beedle, M. (2002). Agile Software Development with Scrum. Pearson.
4. Martin, R. C. (2002). Agile Software Development, Principles, Patterns and Practices. Pearson.

SN	Courses, Modules and Outcomes	Course Contents	Cr
	Semester IV		
425514	**Cloud Computing** **Major(Elective) Theory**		**4**
	Course Outcomes: **Learners will be able to:** • Identify security aspects of each cloud model. • Develop a risk-management strategy for migrating to the Cloud. • Implement a public cloud instance with a public cloud service provider. • Apply a trust-based security model to different layers.		
Module 1	**Introduction to Cloud Computing:**		1
	LOs: Learners will be able to • Define cloud computing and its key characteristics, service models, and deployment models. • Provide an overview of the historical development of cloud computing and its evolution from traditional models. • Compare major cloud service providers like AWS, Azure, and GCP. • Identify and analyze potential security risks and challenges in cloud computing.	**Module Contents:** • Introduction to Cloud Computing • Online Social Networks and Applications • Cloud introduction and overview • Different clouds, Risks, Novel applications of cloud computing	
Module 2	**Cloud Computing Architecture, Cloud Deployment Models**		1
	LOs: Learners will be able to: • Define the requirements driving the emergence of cloud computing and explain CPU virtualization's role. • Provide an overview of basic cloud computing principles, discuss hypervisors, and explain the SPI framework. • Identify key drivers motivating cloud adoption and assess the impact on end-users and businesses. • Explore best practices for establishing effective governance structures in cloud environments.	**Module Contents:** • **Cloud Computing Architecture:** Requirements, Introduction Cloud computing architecture, On Demand Computing Virtualization at the infrastructure level, Security in Cloud computing environments, CPU Virtualization, A discussion on Hypervisors Storage Virtualization Cloud Computing Defined, The SPI Framework for Cloud Computing, The Traditional Software Model, The Cloud Services Delivery Model • **Cloud Deployment Models:** Key Drivers to Adopting the Cloud, The Impact of Cloud Computing on Users, Governance in the Cloud, Barriers to Cloud Computing Adoption in the Enterprise	
Module 3	**Security Issues in Cloud Computing and Access management**		1

	LOs: Learners will be able to • Apply knowledge of infrastructure security in cloud computing to identify key components involved. • Analyse network-level security measures and protocols relevant to cloud environments. • Evaluate application-level security practices and challenges specific to cloud-based applications. • Design strategies to ensure data security and storage in cloud computing environments.	**Module Contents:** • **Security Issues in Cloud Computing**: Infrastructure Security, Infrastructure Security: The Network Level, The Host Level, The Application Level, Data Security and Storage, Aspects of Data Security, Data Security Mitigation Provider Data and Its Security. • **Identity and Access Management:** Trust Boundaries and IAM, IAM Challenges, Relevant IAM Standards and Protocols for Cloud Services, IAM Practices in the Cloud, Cloud Authorization Management.	
Module 4	**Security Management in the Cloud, Privacy Issues**		1
	LOs: Learners will be able to • Apply security management standards relevant to cloud computing to ensure robust security measures. • Analyse availability management practices for SaaS, PaaS, and IaaS to optimize service availability. • Evaluate risk assessments specific to cloud security and propose effective mitigation strategies. • Design and implement tailored incident response plans for cloud computing scenarios to ensure swift and effective responses.	**Module Contents:** • **Security Management in the Cloud:** Security Management Standards, Security Management in the Cloud, Availability Management: SaaS, PaaS, IaaS. • **Privacy Issues:** Privacy Issues, Data Life Cycle, Key Privacy Concerns in the Cloud, Protecting Privacy, Changes to Privacy Risk Management and Compliance in Relation to Cloud Computing, Legal and Regulatory Implications, U.S. Laws and Regulations, International Laws and Regulations.	
Assignments/ Activities			
	These assignments aim to apply theoretical concepts to practical application and critical thinking. **Module 1:** • Research and analyze the evolution of cloud computing and its impact on modern technological landscapes. Identify and explore various online social networks and applications that leverage cloud computing technologies. Develop a comprehensive overview of cloud computing, highlighting different types of clouds, associated risks, and novel applications in diverse domains. **Module 2:**		

	• Investigate the architecture of cloud computing systems, focusing on requirements and essential components. Explore virtualization at the infrastructure level and its role in cloud computing. Evaluate security measures implemented in cloud environments. Examine various cloud deployment models and discuss key drivers influencing cloud adoption. **Module 3**: • Conduct a comprehensive assessment of security issues inherent in cloud computing, covering infrastructure, network, host, and application levels. Explore data security and storage considerations, including mitigation strategies. Investigate identity and access management (IAM) challenges and relevant standards and protocols for cloud services. **Module 4**: • Examine security management standards and practices applicable to cloud computing environments. Analyze availability management for different cloud service models (SaaS, PaaS, IaaS). Investigate privacy concerns related to cloud computing, including data life cycle, key privacy concerns, and compliance with legal and regulatory frameworks.	

References:

1. Erl, T., Mahmood, Z., & Puttini, R. (2013). Cloud Computing: Concepts, Technology & Architecture. Prentice Hall.
2. Reese, G. (2009). Cloud Application Architectures: Building Applications and Infrastructure in the Cloud. O'Reilly Media.
3. Mather, T., Kumaraswamy, S., & Latif, S. (2009). Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance. O'Reilly Media.
4. Bahga, A., & Madisetti, V. (2014). Cloud Computing: A Hands-On Approach. CreateSpace Independent Publishing Platform.