SNDT Women's University, Mumbai

Master of Science in Computer Science

(M.Sc.-CS.)

as per NEP-2020

Syllabus

(2025-26)

Programme	Master of Science in Computer Science (M.Sc.-CS.)
Preamble	In unwavering commitment to the principles outlined herein, the Master of Science in 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 analyze 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 in 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 startups 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 in Computer Science(M.Sc.-CS.)

Year I

Code	Subjects	Type of Course	Credits	Marks	Int.	Ext.
	Semester-I					
115561	Mathematics for Computer Science	Major (Core)	4	100	50	50
115562	Advance Data Structure and Algorithms	Major (Core)	4	100	50	50
115563	Data Communication and Networking	Major (Core)	4	100	50	50
115504	Object-Oriented Programming with Java	Major (Core)	2	50	50	0
	Elective -I	Major (Elective)	4	100	50	50
135581	Research Methodology	Minor stream	4	100	50	50
			22	550	300	250
	Semester-II					
Code	**Subjects**	**Type of Course**	**Credit**	**Marks**	**Int.**	**Ext.**
215561	Artificial Intelligence	Major (Core)	4	100	50	50
215562	Advance DBMS / Next generation databases	Major (Core)	4	100	50	50
215563	Data Science Using Pythion	Major (Core)	4	100	50	50
215564	Full Stack Development	Major (Core)	4	100	50	50
	Elective - II	Major (Elective)	4	100	50	50
245581	OJT	OJT	2	50	0	50
			22	550	250	300

Exit option(44 credits):

Post Graduate Diploma in Computer Science

Year II

Code	Subjects	Type of Course	Credit	Marks	Int.	Ext.
	Semester-III					
315561	Cloud Computing (Web Services)	Major (Core)	4	100	50	50
315562	Machine Learning	Major (Core)	4	100	50	50
315563	Mobile App Development Technologies	Major (Core)	4	100	50	50
315504	Machine Learning- LAB	Major (Core)	2	50	50	0
	Elective - III	Major (Elective)	4	100	50	50
355581	Dissertation - Research Project	RP	4	100	50	50
			22	550	300	250
	Semester-IV					
415561	Interactive & Advanced Data Visualization (with Python)	Major (Core)	4	100	50	50
425561, 425562	Swayam / MOOC	Major (Elective)	4	100	50	50
415562	Deep learning	Major (Core)	4	100	50	50
415563	Digital Forensics Security	Major (Core)	4	100	50	50
455581	Internship (IT) (Full Time Industrial Training (IT))	Internship	6	150	50	100
			22	550	250	300

Code	Elective-I
125561	1. Cyber Security
125562	2. Software Engineering and Project Management
125563	3. Theory of Computation and Applications
125564	4. UI/UX Design

Code	Elective-II
225561	1. Software Testing and Quality Assurance
225562	2. Fuzzy Logic & Neural networks
225563	1. Ethical Hacking
225564	4. Natural Language Processing

Code	Elective-III
325561	1. Digital Forensics Security
325562	2. Big data Analytics
325563	3. IOT
325564	4. GIS and Remote Sensing

MSc (Computer Science) Semester I(NEP pattern)

Major core

Course Title	**Mathematics for Computer Science**
Course Credits	**4 Credits**
Course Outcomes	**CO1 :** To Study Set and Operations .
	CO2: To study logic and boolean Algebra
	CO3: To enable the students to use nonlinear and linear methods to analyze and understand problems in the same
	CO4: To demonstrate how the mathematics help to develop thinking ability
	CO5: To demonstrate how the graph and logic helps to develop thinking ability
Module 1 (Credit 1) Set Theory	
Learning Outcomes	**After learning the module, learners will be able to**
	To Understand the set theory and operations To facilitate applications of set
Course Outline	Set definition ,set builder form .Operations on set ,Venn diagram ,DeMorgan's Law
Module 2 (Credit 1) Logic and Boolean Algebra	
Learning Outcomes	**After learning the module, learners will be able to**
	Logic operations
	Boolean operations
Course Outline	Logic gates ,truth tables ,Boolean expression ,Laws
Module 3 (Credit 1) Solution to nonlinear equation	
Learning Outcomes	**After learning the module, learners will be able to**
	To Understand the methods of non linear equation
Course Outline	**Solution to nonlinear equation** : Bisection method , Newton Raphson Method, Regula Falsi method
Module 4 (Credit 1) Graph	
Learning Outcomes	**After learning the module, learners will be able to**
	Understand Basics of Graph
Course Outline	Graph Definition basic terminology ,Matrix representation of graphs , Types of graphs , Hand Shaking Lemma

Reference Books:

(1) S.S. Sastry; Introductory Methods of Numerical Analysis, 3rd edition, Prentice Hall of India,
(2) H.C. Saxena; Finite differences and Numerical Analysis, S. Chand and Company.
(3) Kolman and Ross - Discrete Mathematical Structures
(4) B.S. Vatssa:-Discrete Mathematics
(5) Set Theory: A First Course by Daniel W. Cunningham:

Assessment :

Internal Assessment – 50 Marks

The Rubric will have the following Evaluation Parameters:		
Evaluation Parameters	**Description / Evaluation Points**	**Marks**
Interpretation	Accuracy in interpreting the example	15
Representation	Translating problems	15
Calculation	To perform correct numerical calculations	15
Logic Development	To enhance logical thinking	5

Sample Projects

Project 1: Find real world applications of graphs
Project 2 :Find real world the applications of logic
Project 3: Prepare the model for Set operations
Project 4: Write program for non linear equations

External Assessment: (Marks 50)

End Semester examination of 50 marks for 2 hours duration will be conducted.

Major (Core)

Course Title	**Advance Data Structure and Algorithms** **Major (Core) Theory**
Course Credits	**4 Credits**
Course Outcomes	**CO1 :**Understand and apply linear data structures-List
	CO2: Understand and apply non linear data structures graphs and trees.
	CO3: Analyze the time and space complexities of basic data structure operations.
	CO4: Apply advanced data structures to solve specific computational problems.
Module 1 (Credit 1)	**Analysis of Algorithm**
Learning Outcomes	**After learning the module, learners will be able to**
	Analyze the time and space complexities of algorithms. ·
	Apply Big-O notation to express the upper bounds of algorithmic performance.
	Evaluate the efficiency and suitability of data structures in different scenarios.
Course Outline	**Introduction:** ADT- stack and queue, data structure: Definition & classification **Space and time complexity**

	Graphical understanding of the relation between different functions of n, examples of linear loop, logarithmic,quadratic loop etc. Analysis of algorithms with emphasis on best case, average case and worst case. Best, Worst, Average case analysis, Asymptotic notations (Big O, Omega Ω, Theta Θ)
Module 2 (Credit 1)	**Linear Data structures**
Learning Outcomes	**After learning the module, learners will be able to**
	Identify and classify linear data structures.
	Analyze the advantages and limitations of each linear data structure.
Course Outline	**Stack and queue** : Introduction, implementation using array. **Linked Lists-** insert, delete and update Singly Linked List- insert, delete and update Circularly Linked List- insert, delete and update Doubly Linked lists- insert, delete and update Linked list - Polynomial Manipulation.
Module 3	**Non Linear Data Structures -Tree**
Learning Outcomes	Understand tree structures, including binary trees and n-ary trees.
	Representation of binary Tree
	Implement tree traversal algorithms (e.g., in-order, pre-order, post-order).
	Working of a balanced tree.
	Types of Binary trees Binary tree, skewed tree, strictly binary tree, full binary tree, complete binary tree, expression tree, binary search tree. **Representation of binary trees.** Implementation and Operations on Binary Search Tree - Create, Insert, Delete, Search **Tree traversals**– preorder, inorder, postorder (recursive implementation), Level-order traversal using queue Max-Heap, Min-Heap Huffman Algorithm Binary search tree **Balanced Trees** AVL Tree- concept and rotations Red Black trees -, insertion and deletion(concept) Multi-way search tree - B and B+ tree - Insertion, Deletion (concept)

Module 4	Hashing
Learning Outcomes	Explain the concept of hashing and hash functions.
	Implement and analyse hash tables for efficient data retrieval
	Hash Table Concept of hashing: Terminologies Hash table,Hash function, Bucket, Hash address, collision,overflow etc. Properties of good hash function **Collision resolution techniques** Open Addressing - Linear probing, quadratic probing, rehashing Chaining - Coalesced , separate chaining

References:

1. Weiss, M. A. (2003). Data Structures and Algorithm Analysis in C (2nd ed.). Pearson Education.
2. Pai, G. A. V. (2008). Data Structures and Algorithms: Concepts, Techniques, and Applications (1st ed.).
3. Horowitz, E., Sahni, S., & Anderson-Freed, S. (2007). Fundamentals of Data Structures in C (2nd ed.). University Press.
4. Tremblay, J.-P., & Sorenson, P. G. (2007). An Introduction to Data Structures with Applications (2nd ed.). Tata McGraw-Hill.
5. SNDTWU Faculty of Science and Technology: M.Sc.-CS. Syllabus 2023-24
6. Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2003). Introduction to Algorithms (2nd ed.). PHI.
7. Dave, P., & Dave, H. (2008).Design and Analysis of algorithms. Pearson Education.
8. Tanenbaum, A. S. (PHI). Data Structures Using C & C++.
9. Goodrich, M., &Tamassia, R. (Wiley). Algorithm Design: Foundation, Analysis & Internet Examples.
10. Aho, A. V., Hopcroft, J. E., & Ullman, J. D. (1983).Data Structures & Algorithms. Addison-Wesley Publishing.
11. Lipschutz, S. (2017).Data Structures with C. McGraw-Hill.
12. Kanetkar, Y. (BPB publications). Data Structures Through C.

Assignments/Activities towards Comprehensive Continuous Evaluation (CCE)

Module 1

Comparative Complexity Study

- Compare two algorithms for the same problem
- Analyze based on:Time Complexity,Space Complexity
- Practical performance for different input sizes.

Module 2

Stack and Queue Using Arrays:

Description: Implement stack and queue data structures using arrays.

Objective: Understand LIFO (stack) and FIFO (queue) behavior.

Evaluation: Functional implementation, correct use of push/pop or enqueue/dequeue.

Group Activity: Linked List Concept Mapping

Description: In small groups, draw and explain how different types of linked lists work (singly, doubly, circular).

Objective: Develop conceptual understanding through peer learning.

Evaluation: Diagram accuracy, explanation clarity, teamwork.

Module 3

Assignment: AVL Tree Rotations

Description: Write code or explain the different types of AVL rotations (LL, RR, LR, RL).

Objective: Understand self-balancing trees and the impact of imbalance.

Evaluation Focus: Rotation logic, balance factor calculation, height updates.

Module 4

Assignment: Hash Function Design

Description: Design simple hash functions for different key types (integers, strings) and analyze their effectiveness.

Objective: Understand the role of a good hash function and collision minimization.

Evaluation Focus: Creativity, correctness, and justification of design.

Assessment

Internal Assessment – 50 Marks

Evaluation Scheme: - Depending on the activities mentioned above a project should be developed for 50 marks. The internal assessment, which is a project evaluation, will be done by conducting a project presentation at the College level, where an External Examiner (Industry Expert or Subject Expert) appointed by the College will be evaluating the project depending on evaluation rubrics given below.

S. No.	Component	Criteria	Marks
1	**Assignments / Lab Work**	Correctness and completeness of implementation	5
		Use of appropriate advanced data structures	3

		Code efficiency and readability	2
		Subtotal	**10**
2	**Internal Test / Viva**	Understanding of concepts and logic	5
		Problem-solving approach	3
		Communication during viva or written clarity	2
		Subtotal	**10**
3	**Mini Project / Lab Task**	Relevance of problem and solution design	5
		Use of appropriate advanced data structures (trees,list, graphs)	5
		Functionality and correctness	3
		Subtotal	**13**
4	**Seminar / Presentation**	Clarity of explanation and content organization	4
		Depth of technical understanding	3
		Presentation skills (PPT/diagrams/Q&A)	3
		Subtotal	**10**
5	**Attendance & Participation**	Attendance, punctuality, and class engagement	2
		Subtotal	**2**
		Total Internal Marks	**50**

External Assessment: (Marks 50)

End Semester examination of 50 marks for 2 hours duration will be conducted.

Major (Core)

Course Title	**Data Communication and Computer Networks**
Course Credits	**4 Credits**
Course Outcomes	**CO1:** Define and explain key concepts in data communications.
	CO2: Understand the OSI (Open Systems Interconnection) model and TCP/IP protocol suite.
	CO3: Describe and differentiate between various networking protocols.
	CO4: Analyze the functions and characteristics of key protocols such as TCP, UDP, IP, and HTTP.
	CO5: Design and implement basic computer networks based on specific requirements**.**
	CO6: Evaluate and select appropriate networking topologies and configurations.
Module 1 (Credit 1)	**Introduction to Computer Networks**
Learning Outcomes	**After learning the module, learners will be able to**
	Understand the concept of computer networks and their significance in modern computing.
	Explain the purposes and advantages of connecting computers in a networked environment.
	Identify and differentiate between common network topologies.
	Analyze the strengths and weaknesses of various topologies in different scenarios.
	Explore various applications of computer networks in different domains (e.g., business, education, healthcare).
	Analyze case studies to understand how networks are utilized to meet specific organizational needs.

Content Outline	**Introduction:** Computer Networks and its uses, Network categorization and Hardware: Broadcast and point-to-point networks, Local Area Network (LAN), Metropolitan Area Network (MAN), Wide Area Networks (WAN), Inter networks, Topologies, Wireless Networks, Network Software: Protocols, Services, network architecture, design issues, OSI Reference model, TCP/IP Reference model, Comparison of OSI and TCP/IP Models. Introduction to Example Networks: Internet, Connection-Oriented Networks–X.25, Frame Relay, ATM **Data Communication Model:** Digital and Analog data and signals, bit rate, baud, bandwidth, Nyquist bit rate, Guided Transmission Media – Twisted Pair, Coaxial cable, Optical fiber; wireless transmission–Radio waves, microwaves, infrared waves; Satellite Communication.
Module 2 (Credit 1)	**Switching, Error Detection and Correction**
Learning Outcomes	**After learning the module, learners will be able to**
	Explain the concept of circuit switching in telecommunication networks.
	Differentiate circuit switching from other switching techniques.
	Describe the process of establishing a circuit in a circuit-switched network.
	Describe how packets are routed and forwarded in a packet-switched network.
	Explore routing algorithms used in packet switching.
Content Outline	**Switching:** Circuit Switching, Packet switching; Multiplexing: Frequency Division Multiplexing, Time Division Multiplexing, Synchronous and Asynchronous TDM, Modems, Transmission impairments, Manchester and differential Manchester encoding **Error Detection and Correction:** Types of errors Redundancy, Detection Versus Correction, Error Detection, Error Correction, Hamming Code, Cyclic Redundancy Check, Check sum and Its idea.
Module 3 (Credit 1)	**Data Link Layer Design issues**

Learning Outcomes	**After learning the module, learners will be able to**
	Explain the role of the Data Link Layer in the OSI model.
	Differentiate between the functions of the Physical Layer and the Data Link Layer.
	Discuss various framing techniques used in the Data Link Layer.
	Implement framing algorithms for efficient data encapsulation and transmission
Content Outline	Data Link Layer Design issues: Framing, error control, Flow Control, Error Detection and correction; Elementary Data Link Protocols, Sliding Windows Protocols; Medium Access Control: Aloha, CSMA protocols, Collision free protocols, Limited Contention Protocols; Wave length division Multiple access protocol, Wireless LAN Protocol: MACA; IEEE 802.3Ethernet, IEEE 802.4 Token Bus; IEEE 802.5 Token ring, Binary Exponential Back off algorithm, Digital Cellular, Radio : Global System for Mobile, Communication (GSM), Code Division Multiple Access (CDMA)
Module 4 (Credit 1)	**Network Layer, Design issues**
Learning Outcomes	**After learning the module, learners will be able to**
	Explain the purpose and functions of the Network Layer in the OSI model.
	Differentiate between the responsibilities of the Network Layer and other layers.
	Understand the concepts of addressing and routing at the Network Layer.
	Design and implement addressing schemes for efficient packet routing.
	Define virtual circuit switching and its advantages.
	Compare and contrast virtual circuit switching with other switching techniques.
Content Outline	Network Layer, Design issues Virtual circuit and Datagram Subnet, Routing Algorithms, Optimality principle, shortest path routing, Flooding, Distance Vector Routing,Link State Routing, Hierarchical

	Routing, Broadcast and Multi Cast Routing, Routing for Mobile hosts, Routing in Ad hoc Networks, congestion Control Algorithm, General Principals Traffic Shaping, Leaky Bucket, Token Bucket, choke packets, Load Shedding

Reference Books

1. Forouzan, B. A. (2007). Data Communications and Networking (4th ed.). McGraw Hill. ISBN: 0-07-296775-7.
2. Stallings, W. (2013). Data and Computer Communications (10th ed.). Pearson.
3. Tanenbaum, A. S. (2010). Computer Networks (5th ed.). Pearson.

Classroom Activities

Activities towards Comprehensive Continuous Evaluation(CCE)

Module 1:

1. Define a computer network and explain the applications of computer networks.
2. Explain LAN, MAN, WAN.
3. What is the OSI model? Explain the functions of each layer in the OSI model.
4. Explain the network topologies.
5. Describe the guided and unguided transmission media.

Module 2:

1. Differentiate between circuit switching and packet switching.
2. What is multiplexing? Explain various types of multiplexing.
3. Explain Manchester and differential Manchester encoding.
4. Describe Hamming code and CRC.

Module 3:

1. What are the design issues of the data link layer?
2. Explain the sliding window protocol.
3. Describe the IEEE token ring and token bus protocols.
4. Explain the concept of GSM.

Module 4:

1. What are the design issues of the network layer?
2. Explain the link state routing protocol.
3. Describe the distance vector routing protocol.
4. Explain the various congestion control techniques.

Assessment

Internal Assessment – 50 Marks

Evaluation Scheme:

Component	Weight	Typical Criteria
Mid-term Exam (CIA I)	~25%	Coverage of core theory, routing, protocols, addressing
Assignments & Quizzes (CIA II & III)	~10–20%	Case studies, seminars, problem-solving
Lab Reports	~10%	Clarity, accuracy, visuals, completeness, timely submission
Group Project / Final Report	~20%	Design, topology, addressing, cost, analysis, presentation
Attendance & Participation	~5%	Active engagement throughout the course

External Assessment: (Marks 50)

End Semester examination of 50 marks for 2 hours duration will be conducted.

Major (Core)

Course Title	**Object Oriented Programming with Java**
Course Credits	**2 Credits**
	CO1 : To introduce the object-oriented programming system concepts
	CO2: To develop modular programs using Java
	CO3: To setup JDK environment to create, debug and run Java programs
	CO4: To introduce syntax and semantics of Java programming language as well as JDBC
Module 1 (Credit 1) Introduction to Java Programming	
Learning Outcomes	**After learning the module, learners will be able to**
	Apply Object-Oriented Programming Concepts
	Demonstrate Proficiency with Java Syntax and Language Fundamentals
	Implement Control Flow and Decision-Making in Java Programs
Course Outline	**Introduction to Java :** Basics of Java programming, Datatypes, Variables, Operators, Control structures including selection, Looping, Java methods, Overloading, Math class,Arrays in java. **Inheritance in java:** Super and sub class, Overriding, Object class, Polymorphism, Dynamic binding, Generic programming, Casting objects,Instance of operator, Abstract class, Interface in java, Package in java, UTIL package,Multithreading,Exception Handling Introduction to JDBC: What is JDBC. Database connectivity, JDBC Architecture, JDBC drivers, Using JDBC API – Loading a Driver, connecting and executing JDBC statement, Handling SQL Exceptions. Accessing Result Sets, method of Result Set interface, Methods of Prepared Statement interface, retrieving row, inserting row, Managing Database Transactions, creating and calling stored procedures in JDBC, using Metadata in JDBC.
Module 2 (Credit 1) JSP,Servlet,Hibernet,Spring MVC	
Learning Outcomes	**After learning the module, learners will be able to**

	1)Understanding of advanced web concepts associated with JAVA.
Course Outline	**Introduction to servlets:** Servlet vs CGI, Servlet API overview, Servlet Life cycle, Generic servlet, HTTP Servlet, ServletConfig, Servlet Context, Handling HTTP Request and response –GET / POST method, request dispatching, Using cookies, Session tracking. **Web development using JSP:** Introduction to JSP, JSP Architecture, JSP Directives, JSP scripting elements, Default objects in JSP, JSP Actions, JSP with beans and JSP with Database, Error handling in JSP, tracking techniques in JSP, Introduction to custom tags, JSTL tags in detail. Introduction to jQuery, JS, JS JSON, jQuery vs JS. **Introduction to Spring Framework:** Spring Architecture, Spring Aspect of Object-Oriented Concepts – Join Point and Point Cuts. Spring web applications with Spring MVC. Features of the Spring Boot. Use of Spring Boot to create and configure a Spring application. Customize Spring Boot features. REST web services with Spring. Spring Data Secure with Spring Security. JMS- Introduction, requirement, JMS Programming model. JMS support of Spring. **Introduction to Hibernate:** Problem with JDBC - paradigm mismatch, ORM. Different components of Hibernate, How to persist objects using Hibernate, mapping files in hibernate, configuration files and Session object, Instance states, Implementing Inheritance in Hibernate, Transactions in Hibernate, Querying with HQL (Hibernate Query Language)

Reference Books

1. The Complete Reference, Third Edition, by Patrick Naughton and Herbert Schildt, Tata McGraw Hill Edition 1999.
2. Java Enterprise in a Nutshell: A Desktop Quick Reference (Nutshell Handbook) or any other

book with similar contents.

3. Mastering Java2 J2SE1.4 by John Zukouski PBP Publication
4. JavaTM How to Program Sixth Edition by H.M Deitel, P.J. Deitel
5. Core Servlets & JavaServer Pages by Marty Hall, Larry Brown
6. Spring Boot in Action 1st Edition by Craig Walls
7. Beginning Hibernate Second Edition by Jeff Linwood, Dave Minter is the third book for Hibernate beginners.

Assessment

Internal-50 Marks

Internal Assessment to be done on the basis of Project (20M) ,Unit test (20M) and assignment

Project 1

Using jdk design a Bank Account System which allows users to
create accounts, deposit money, withdraw money, and check balances,Examples of JDBC

Project 2

Assignments of-Servlet,JSP,JMS Support of Spring,Introduction to Hibernet and Querying with HQL.

The Rubric will have the following Evaluation Parameters:		
Evaluation Parameters	**Description / Evaluation Points**	**Marks**
Code Implementation	● Accuracy in writing and executing the Java program, correct logic, syntax	15
Problem Understanding	● Clarity of problem interpretation and correct approach	10
Output Correctness	● Program compiles without errors and gives correct output	10
Code Quality & Structure	● Proper use of indentation, comments, meaningful variable names, modularity	5
Debugging Skills	● Ability to identify and fix logical/syntax errors during execution	5
Lab Record/Journal	● Completeness, correctness, formatting, and timely submission	5

Major (Elective - I)

Course Title	**Cyber Security (Elective-I) Theory**
Course Credits	**4 Credits**
	CO1 : Provide an overview of the field of Cyber Security, including its challenges, constraints, and the role of Internet governance.
	CO2: Understand the imperative for a comprehensive Cyber Security policy, the establishment of a nodal authority, and the importance of an international convention on Cyberspace.
	CO3: Identify vulnerabilities in software, system administration, network architectures, data access, authentication, broadband communications, and poor awareness.
	CO4: Identify intrusion types, such as physical theft, privilege abuse, unauthorized access, malware infection, and implement techniques including anti-malware software, network-based intrusion detection/prevention systems with its types.
Module 1 (Credit 1) Introduction to Cyber Security	
Learning Outcomes	**After learning the module, learners will be able to**
	Demonstrate the ability to conduct security audits, identifying potential,weaknesses in systems and networks.
	Explain the role of cryptography in Cyber Security and apply cryptographic techniques to secure data communication.
	Understand the concept of ethical hacking and its role in proactively identifying and addressing vulnerabilities.
Course Outline	**Introduction to Cyber Security** Overview of Cyber Security, Internet Governance–Challenges and Constraints, Cyber Threats:- Cyber Warfare-Cyber Crime-Cyber, terrorism-Cyber Espionage, Need for a Comprehensive Cyber Security Policy, Need for a Nodal Authority, Need for an International convention on Cyberspace **Cyber Security Vulnerabilities and Safeguards** Cyber Security Vulnerabilities Overview, vulnerabilities in software, System administration, Complex Network Architectures, Open Access to Organizational Data, Weak Authentication, Unprotected Broadband

	communications, Poor Cyber Security Awareness. Cyber Security Safeguards Overview,Accesscontrol,Audit, Authentication,Biometrics,Cryptography, Deception, Denial of Service Filters, Ethical Hacking, Firewalls, Intrusion Detection Systems, Response, Scanning, Security policy, Threat Management.
Module 2 (Credit 1) Securing Web Application	
Learning Outcomes	**After learning the module, learners will be able to**
	Define and explain the roles of services and servers in the context of web applications.
	Understand identity management principles and implement secure identity practices within web services.
	Apply security measures to prevent and detect unauthorized access attempts by external entities.
Course Outline	**Securing Web Application** Services and Servers Introduction, Basic security for HTTP Applications and Services, Basic Security for SOAP Services, Identity Management and Web Services, Authorization Patterns, Security Considerations, Challenges. **Intrusion Detection and Prevention** Intrusion, Physical Theft, Abuse of Privileges, Unauthorized Access by Outsider, Malware infection, Intrusion detection and Prevention Techniques, Anti-Malware software, Network based Intrusion detection Systems, Network based Intrusion Prevention Systems, Host based Intrusion prevention Systems, Security Information Management, Network Session Analysis, System Integrity Validation
Module 3 (Credit 1) Cryptography and Network Security	
Learning Outcomes	**After learning the module, learners will be able to**
	Define cryptography and explain its role in securing information and communication.
	Differentiate between symmetric and asymmetric key cryptography, and understand their applications in securing data.
	Understand and apply cryptography in various applications, demonstrating proficiency in securing data in different contexts.
Course Outline	**Cryptography and Network Security:**

	Introduction to Cryptography, Symmetric key Cryptography, Asymmetric key Cryptography, Message Authentication, Digital Signatures, Applications of Cryptography. Overview of Firewalls- Types of Firewalls, User Management, VPN Security, Security Protocols:-security at the Application Layer-PGP and S/MIME, Security at Transport Layer-SSL And TLS, Security at Network Layer-IPSec.
Module 4 (Credit 1) Cyber jurisprudence & Forensic	
Learning Outcomes	Analyze the specificities of the Indian cyber space, including its regulatory framework, challenges, and initiatives.
	Define Cyber Forensics and understand its significance in investigating cybercrimes and digital incidents.
	Demonstrate the ability to initiate and conduct preliminary investigations in response to suspected cyber incidents, ensuring the preservation of digital evidence.
Course Outline	**Cyberspace and the Law:** Introduction, Cyber Security Regulations, Roles of International Law, the state and Private Sector in Cyber space, Cyber Security Standards. The INDIAN Cyber space, National Cyber Security Policy 2013. **Cyber Forensics** Introduction to Cyber Forensics, Handling Preliminary Investigations, Controlling an Investigation, Conducting disk-based analysis, Investigating Information-hiding, Scrutinizing E-mail, Validating E-mail header information, Tracing Internet access, Tracing memory in real-time.
Activities to be done in the class towards Comprehensive Continuous Evaluation	
1. These assignments and activities are designed to engage students in practical applications of cyber security concepts, fostering critical thinking and skill development throughout the course. 2. Divide students into groups and assign each group a specific cyber security vulnerability (e.g., software vulnerabilities, weak authentication). Have them research, analyze, and present strategies to mitigate the assigned vulnerability. 3. Organize a workshop where students present and demonstrate various cyber security safeguards. This can include access control, encryption, firewalls, and intrusion detection systems. Encourage hands-on activities and practical demonstrations.	

4. Assign students a case study involving a web application. They should conduct a security assessment, identify vulnerabilities, and propose safeguards. Emphasize securing HTTP and SOAP services, identity management, and authorization patterns.

5. Provide case studies related to intrusion incidents. Students should analyze each case, identify the type of intrusion, and propose effective prevention and detection techniques. Encourage discussion on ethical hacking and security policy enforcement.

6. Provide legal cases related to cyber space and cyber security. Students should analyze the legal implications, court decisions, and the role of international law and regulations.

7. Assign students to review and critique the National Cyber Security Policy of 2013. They should assess its effectiveness, identify areas for improvement, and propose updated recommendations.

8. Encourage students to write research papers on cyber security and spreading awareness among society.

Bibliography:

1.Cybersecurity Essentials
Charles J. Brooks et al.,Publisher: Wiley, 2022.
2.Cryptography and Network Security
Atul Kahate ,Publisher: McGraw Hill Education.
3.Cybersecurity in Context: Technology, Policy, and Law
Chris Jay Hoofnagle & Golden G. Richard III ,Publisher: Wiley, 2024 1st Edition.
4. Digital Forensics and Incident Response
Gerard Johansen,Publisher: Packt Publishing, 3rd Edition, 2023.
5. Digital Privacy and Security Using Windows: A Practical Guide
Nihad Hassan,Rami Hijazi, Apress.
6. Securing the Future: The Role of AI in Cybersecurity
Nimeshkumar Patel.

Assessment

Internal Assessment – 50 Marks

Evaluation Scheme:

Depending on the activities mentioned above a project should be developed for 50 marks. The internal assessment, which is a project evaluation, will be done by conducting a project presentation at the College level, where an External Examiner (Industry Expert or Subject Expert) appointed by the College will be evaluating the project depending on evaluation rubrics given below.

The Rubric will have the following Evaluation Parameters:		
Evaluation Parameters	**Description / Evaluation Points**	**Marks**

Understanding of Cyber Security Fundamentals	Demonstrates in-depth understanding of cyber threats, governance, policies, and global conventions.	10
Securing Web Applications	Implements identity/access management principles; prevents unauthorized access efficiently.	10
Problem Solving, Ethical Hacking, IDS/IPS Implementation	Applies ethical hacking tools & techniques proficiently; accurately implements IDS/IPS with justifications.	10
Application of Cryptographic Techniques	Effectively applies cryptographic algorithms and differentiates between symmetric/asymmetric cryptography.	10
Cyber Forensics and Legal Awareness	Demonstrates ability to analyze digital evidence and explain Indian and global cyber laws precisely.	10

Given below are two sample case study, but it is expected to work on similar sort of Case study

Case study 1:-The Data Breach at Any Bank,Identify the cyber vulnerabilities that led to the breach.Discuss what kind of Intrusion Detection/Prevention Systems should have been used. Propose a Cyber Security policy and technical safeguards to prevent such attacks.

Case study 2:-E-commerce Website Under DDoS AttackExplain the working of a DDoS attack and how it affects server availability.Suggest network-level and application-level safeguards to mitigate DDoS risks.Recommend changes in server architecture to prevent similar issues in the future.

External Assessment: (Marks 50)

End Semester examination of 50 marks for 2 hours duration will be conducted by University

Major (Elective)

Course Title	**Software Engineering and Project Management**
Course Credits	**4 Credits**
	CO1- Understand structured development methodologies and various models like agile or waterfall.
	CO2- Recognize the pivotal role of Software Requirements Specification (SRS) in documenting software needs.
	CO3 -Estimate costs, create timelines, allocate resources efficiently, implement quality assurance, and manage risks.

	CO4- Adhere to effective coding, thorough verification, and engage in testing methodologies.
	CO5 - Demonstrate knowledge beyond development, covering maintenance, risk management, and project management concepts.
Module 1 (Credit 1) **Software Processes, Software Requirement Analysis and Specification**	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Understand the concepts of software processes, projects, and products.
	2. Examine component software processes and their roles in the development lifecycle.
	3. Investigate the software configuration management process and its importance.
	4. Identify the characteristics of a software process and how they influence project outcomes.
	5. Understand other modeling approaches, such as prototyping, and their relevance to requirement analysis.
	6. Define software requirements and recognize the need for Software Requirement Specification (SRS).
Content Outline	● **Software Processes:** Processes projects and products,Component software processes, characteristics of a software process, software Development Process, project management process, software configuration management process, software configuration management process, and process management process ● **Software requirement Analysis and Specification:** Software requirement, need for SRS, requirement process, problem analysis, analysis issues. Informal approach, structured analysis, object-oriented modelling, other modelling approaches, prototyping,
Module 2 (Credit 1) **Planning Software Project and Coding**	
Learning Outcomes	**After learning the module, learners will be able to**
	Explore verification techniques in coding and Identify size measures in the context of coding and programming.
	Understand the fundamentals of software testing.
	Explore code walk-throughs and inspections in the testing process.

	Understand different testing strategies and the associated issues.
Content Outline	**Module Contents:** **Planning Software Project:** Cost estimation, uncertainties in cost estimation, building cost estimation models, on size estimation, COCOMO model, project scheduling, average duration estimation, project scheduling and milestones, staffing and personnel planning, ray leigh curve, personnel plan, team structure, software configuration management plans, quality assurance plans, verification and validation, project monitoring plans, risk management. **Coding:** Programming practice, verification, size measures, complexity analysis, coding standards. Testing– fundamentals, white box testing, control structure testing, black box testing, basis path testing, code walk-throughs and inspection, testing strategies-Issues, Unit testing, integration testing, Validation testing, System testing
Module 3 (Credit 1) **Maintenance**	
Learning Outcomes	**After learning the module, learners will be able to** Understand the overall process of software maintenance and its significance in the software development lifecycle.
	Identify and differentiate between various types of maintenance activities.
	Define software risks and recognize their impact on software projects.
	Understand the fundamental concepts of project management
Content Outline	**Maintenance:** Overview of maintenance process, types of maintenance. **Risk management:** software risks-risk identification-risk monitoring and management. Project Management concept: People–Product-Process-Project.
Module 4 (Credit 1) **Protection and Security**	
Learning Outcomes	**After learning the module, learners will be able to** Understand the basic concepts of project scheduling and tracking in software development.
	Understand the basics of software configuration management (SCM)

	Understand the basics of CASE tools and their role in the software development lifecycle.
	Learn about the building blocks of CASE tools and how they contribute to the development process
Content Outline	**Project scheduling and tracking:** Basic concepts-relation between people and effort-defining task set for the software project-selecting, software engineering task **Software configuration management:** Basics and standards User interface design-rules. Computer aided software engineering tools-CASE building blocks, taxonomy of CASE tools, integrated CASE environment.

Assignments towards Comprehensive Continuous Evaluation

Module 1:

1. What are some common software process models?
2. What is the difference between traditional and agile processes in handling uncertainty?
3. How can you customize the development of a software process?
4. What are functional and non-functional requirements?
5. Choose a software project scenario and develop a project schedule,considering task dependencies and the allocation of resources. Discuss the challenges and benefits of the chosen scheduling approach.

Module 2:

1. What is the project's goal and purpose?
2. How will communication be managed between stakeholders and the development team?
3. What is the overall architecture of the software?
4. What coding standards and best practices will be followed?
5. What is the process for handling bug fixes and updates after the initial release?
6. How will the team handle changes to requirements during development?

Module 3:

1. What is the difference between preventive and predictive maintenance?
2. How is data analysis used to improve maintenance effectiveness and predict potential issues?
3. What is the role of technology in supporting maintenance and risk management?
4. How are risks assessed and prioritized?
5. What is the process for reviewing and updating risk assessments?

Module 4:

1. What are the project milestones?
2. What are the project dependencies?
3. What is a Work Breakdown Structure (WBS)?
4. What is critical path analysis?
5. What is Software Configuration Management (SCM)?
6. What is the difference between SCM and Version Control?
7. Why is SCM important in software development?

Reference Books :

1. Pressman, Roger. (Year). "Software Engineering: A Practitioner's Approach." Tata McGrawHill, New Delhi.
2. Jalote, Pankaj. (Year)."An Integrated Approach to Software Engineering."Narosa, New Delhi.
3. Fairley, R. E. (1985). "Software Engineering Concepts." McGraw-Hill, Inc.

Poyce.(Year)."Software Project Management."Addison-Wesley.

4. Sommerville.(Year)."Software Engineering."Addison-Wesley.

Assessment:
Internal Assessment – 50 Marks

The Rubric will have the following Evaluation Parameters:

Criteria	Description	Marks	Assessment Type
1. Understanding of Concepts	Clarity in core software engineering principles and project management fundamentals (SDLC, models, tools, planning, risk management)	10	Written exam/quiz/assignment

2. Application of Methodologies	Application of software development methodologies (Agile, Waterfall, DevOps) and project management techniques (Gantt charts, CPM, etc.)	8	Case study/assignment/practical
3. Practical Tool Usage	Proficiency in using software engineering and project management tools (JIRA, MS Project, Git, etc.)	6	Lab/practical/project
4. Project Planning and Execution	Ability to plan, schedule, and execute a software project, including resource allocation and risk mitigation	8	Project report/presentation
5. Teamwork and Communication	Collaboration skills and communication effectiveness in project scenarios	6	Peer evaluation/presentation
6. Innovation and Problem-Solving	Creative solutions and handling project challenges effectively	6	Project outcome and report
7. Report and Documentation	Quality and clarity of project documentation and software engineering artifacts	6	Project report/documentation

External Assessment: (Marks 50)
End Semester examination of 50 marks for 2 hours duration will be conducted.

Major (Elective)

Course Title	**Theory of Computation and Application**
Course Credit	**4 Credits**
Course Outcomes	To give an overview of the theoretical foundations of computer science from the perspective of formal languages
	CO1. To give an overview of the theoretical foundations of computer science from the perspective of formal languages.
	CO2. To illustrate finite state machines to solve problems in computing

	CO3. Prove properties of languages and automata, including their limitations.
	CO4. To familiarize Regular grammars, context frees grammar.
	CO5. Understand the theoretical foundations of computation and its limitations.
Module 1 (Credit 1)	**Foundations of Theory of Computation**
Learning Outcomes	**After learning the module, learners will be able to**
	1. Understanding fundamental concepts in automata theory, formal languages, and computability
Content Outlines	1. Define alphabet, string, language 2. Operations on languages. Introduce the concept of regular expressions and their relationship to regular languages. 3. Discuss deterministic finite automata (DFA), non-deterministic finite automata (NFA), and their equivalence. Explore regular languages and their representation using FA.
Module 2 (Credit 1)	Regular Languages and Regular Grammars
Learning Outcomes	**After learning the module, learners will be able to**
	Understand the fundamental concepts of formal languages and their relationship to automata
Content Outlines	1. Introduce regular grammars and their relationship to regular languages and finite automata. 2. Cover pumping lemma for regular languages.
Module 3 (Credit 1)	Context-Free Languages (CFL) and Pushdown Automata (PDA)
Learning Outcomes	**After learning the module, learners will be able to**
	Understand the fundamental concepts of formal language theory, including the differences between regular and context-free

	languages, and the power of using a stack for recognizing patterns in languages that finite automata cannot handle
Content Outlines	1. Introduce context-free grammars and their relationship to pushdown automata. Cover pumping lemma for CFLs. 2. Turing Machines 3. Concepts of Decidability and Undecidability,Computational Complexity,NP-Completeness
Module 4 (Credit 1)	Applications of Theory of Computation
Learning Outcomes	**After learning the module, learners will be able to**
	Design and analyze different types of automata, understand formal languages, and grasp the limitations of computation
Content Outlines	**1. Compiler Design**-Explain how concepts like regular expressions, context-free grammars, and pushdown automata are used in lexical analysis, parsing, and syntax checking. **2. Short explanation about Database Theory**,Formal Verification,Bioinformatics,Natural Language Processing,Cryptography

Evaluation Scheme:

Internal Assessment: (Marks 50)

This involves a combination of tests, assignments, quizzes, and attendance. where an External Examiner (Industry Expert or Subject Expert) appointed by the College will be evaluating the internal assessment depending on evaluation rubrics given below
Tests:-One or more theory tests are conducted to assess the students' understanding of the course material.
Assignments:-Students are assigned problems and exercises related to the course content, which are evaluated at regular intervals.
Attendance:-Regular attendance in lectures is often a component of the internal assessment.

The Rubric will have the following Evaluation Parameters:

Component	Marks	Evaluation Criteria

1. Unit Tests / Internal Exams (2 Tests)	**20**	**Problem solving (DFA/NFA/PDA/CFG/RE/TM) – 10 ,Theoretical questions & proofs – 10**
2. Assignments	**10**	**DFA/PDA/CFG design tasks – 5.Presentation, accuracy, and submission – 5**
3. Practical	**10**	**Correct implementation of automata models – 5 Output explanation – 5**
4. Viva / Oral Assessment	**5**	**Concept clarity (definitions, conversions) 3,Communication – 2**
5. Attendance & Class Participation	**5**	**Regularity and engagement**

Classroom Activities

Activities towards Comprehensive Continuous Evaluation(CCE)

Module 1:

1. **Prove or disprove:** The relation R={(a,b)∣a≤b}R = \{(a, b) \mid a \leq b\}R={(a,b)∣a≤b} on the set of integers is reflexive, antisymmetric, and transitive.
2. **Construct a truth table** for the logical expression: (P→Q)∧(¬Q→¬P)(P \rightarrow Q) \land (\neg Q \rightarrow \neg P)(P→Q)∧(¬Q→¬P) and determine whether it is a tautology.
3. **Use proof by contradiction** to show that 2\sqrt{2}2 is an irrational number.
   Given the sets
   A={1,2,3},B={2,3,4}A = \{1, 2, 3\}, B = \{2, 3, 4\}A={1,2,3},B={2,3,4}, compute A∪B,A∩B,A−B,B−AA \cup B, A \cap B, A - B, B - AA∪B,A∩B,A−B,B−A.
4. **Determine whether the function** f:Z→Zf: \mathbb{Z} \rightarrow \mathbb{Z}f:Z→Z, defined by f(x)=2x+1f(x) = 2x + 1f(x)=2x+1, is one-to-one and/or onto. Justify your answer.

Module 2:

Design DFA for a given language:- Design a Deterministic Finite Automaton (DFA) over the alphabet Σ={0,1}\Sigma = \{0,1\}Σ={0,1} that accepts all strings ending with "01".

1. **Convert Regular Expression to NFA and then to DFA:-**Given the regular expression R=(a+b)∗abR = (a + b)^*abR=(a+b)∗ab, construct an NFA using Thompson's construction and then convert it to an equivalent DFA.
2. **Prove a language is not regular using Pumping Lemma:-** Use the Pumping Lemma to prove that the language L={anbn∣n≥0}L = \{ a^n b^n \mid n \geq 0 \}L={anbn∣n≥0} is not regular.

3. **Minimize a given DFA:-** Consider any DFA with redundant states, apply the state minimization algorithm to find an equivalent minimized DFA.
4. **Construct a Regular Grammar:-** Construct a right-linear grammar (regular grammar) that generates the language L={w∈{0,1}∗|w ends with 11}L = \{ w \in \{0,1\}^* \mid w \text{ ends with } 11 \}L={w∈{0,1}∗|w ends with 11}.

Module 3:

1. Construct a Context-Free Grammar (CFG) :- Design a context-free grammar (CFG) for the language L={anbn|n≥1}L = \{ a^n b^n \mid n \geq 1 \}L={anbn|n≥1}. Also, derive the string "aaabbb" using your grammar.

2. Convert CFG to Chomsky Normal Form (CNF):- Convert the following CFG to Chomsky Normal Form:S→aSb|abS \rightarrow aSb \mid abS→aSb|ab

3. Design a PDA for a Given Language:- Construct a Pushdown Automaton (PDA) that accepts the language L={anbn|n≥0}L = \{ a^n b^n \mid n \geq 0 \}L={anbn|n≥0} by empty stack or final state (specify your choice).

4. Prove a Language is Not Context-Free:- Use the Pumping Lemma for CFLs to show that the language L={anbncn|n≥1}L = \{ a^n b^n c^n \mid n \geq 1 \}L={anbncn|n≥1} is not context-free.

5. Parse Tree Construction and Ambiguity Check:- Given the CFG:S→SS|aS \rightarrow SS \mid aS→SS|a Draw two different parse trees for the string "aaa" and explain whether the grammar is ambiguous.

Module 4:1. Essay/Report: Application of Automata in Compiler Design :- Write a short essay or report explaining how Finite Automata are used in lexical analysis during compiler design. Include an example of token recognition using a DFA.

2. Case Study: TOC in Natural Language Processing (NLP):- Choose one real-world application of TOC in NLP (e.g., syntactic parsing, grammar checking). Describe how context-free grammars (CFGs) are used to model sentence structure in natural languages.

3. Assignment: TOC in Artificial Intelligence / Machine Learning :- Explain the role of formal languages and automata in the field of AI/ML. Give examples where automata or grammar-based systems are used in intelligent systems.

4. Research Activity: TOC in Network Protocols / Cybersecurity :- Investigate how regular expressions and automata are applied in validating network protocols or filtering malicious traffic. Submit a 1–2 page research summary with examples.

5. Presentation: Real-Life Use Cases of Turing Machines:- Prepare a group presentation on the theoretical concept of the Turing Machine and its relevance to modern-day computing devices or algorithmic problem solving. Use at least one real-life analogy or simulation.

TEXT BOOKS:1. John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman (2007), Introduction to Automata Theory , 2. Languages andComputation, 3rd edition, Pearson Education, India.

REFERENCE BOOKS:

1. K. L. P Mishra, N. Chandrashekaran (2003), Theory of Computer Science-AutomataLanguages and Computation, 2nd edition, Prentice Hall of India, India

Minor Stream

Course Title	**Research Methodology**
Course Credit	**4 Credits**
Course Outcome	**CO1**: Demonstrate an understanding of the key steps involved in the research process, including problem formulation, literature review, data collection, analysis, and interpretation.
	CO2: Formulate clear and focused research questions and hypotheses based on a thorough review of existing literature and identification of research gaps.
	CO3: Conduct comprehensive literature reviews to identify relevant studies, theories, and methodologies within a specific research domain.
	CO4: Apply basic statistical techniques for data analysis, interpretation, and drawing meaningful conclusions from research findings
Module 1(Credit 1)	**Introduction to Research methodology**
Learning Outcome	**After learning the modules,learners will be able to**
	Understand the fundamentals of research – Explain the meaning, objectives, and significance of research in various disciplines.
	Identify and define research problems – Formulate clear and researchable problems and hypotheses based on real-world issues.
	Conduct literature reviews – Search, analyze, and synthesize existing research to support new investigations.
	Apply appropriate data collection methods – Use surveys, interviews, observations, and experiments effectively in gathering primary data.
	Differentiate between qualitative and quantitative research – Understand the applications, strengths, and limitations of both approaches.
Content Outline	● **Research methodology** An Introduction Objectives of Research, Types of Research, Research Methods and Methodology, Defining a Research Problem, Techniques Involved in Defining a Problem. ● **Research Design**

	Need for Research Design, Features of Good Design, Different Research Designs, Basic Principles of Experimental Designs, Sampling Design, Steps in Sampling Design, Types of Sampling Design, Sampling Fundamentals, Estimation, Sample size Determination, Random sampling.
Module2 (Credit 1)	**Measurement and Scaling Techniques**
Learning Outcome	**After learning the modules,learners will be able to**
	Understand the concepts of measurement in research – Explain the meaning, importance, and levels of measurement (nominal, ordinal, interval, and ratio).
	Understand the principles of scaling – Explain the purpose and use of different scaling techniques in behavioral and social science research.
	Interpret scaled data effectively – Analyze and draw conclusions from data collected through various scaling methods.
Content Outline	● **Measurement in Research** Measurement Scales, Sources in Error, Techniques of Developing Measurement Tools, Scaling, Meaning of Scale, Scale Construction Techniques ● **Methods of Data Collection and Analysis** Collection of Primary and Secondary Data, Selection of appropriate method Data Processing Operations, Elements of Analysis, Statistics in Research, Measures of Dispersion, Measures of Skewness, Regression Analysis, Correlation.
Module3 (Credit 1)	**Techniques of Hypotheses**
Learning Outcome	**After learning the modules,learners will be able to**
	Understand the concept of a hypothesis – Define hypothesis, its role, and importance in the research process.
	Formulate testable hypotheses – Develop clear, specific, and researchable hypotheses based on the research problem and literature review.
	Select appropriate statistical tests – Choose suitable parametric or non-parametric tests based on the type of data and research design.

Content Outline	• **Techniques of Hypotheses** Parametric or Standard Tests Basic concepts, Tests for Hypotheses I and II, Important parameters limitations of the tests of Hypotheses, Chi-square Test, Comparing Variance, as a non-parametric Test, Conversion of ChitoPhi, Caution in using Chi-square test.
Module 4(Credit 1)	**Analysis of Variance and Co-variance**
Learning Outcome	**After learning the modules,learners will be able to**
	Understand the concepts of ANOVA and ANCOVA – Explain the purpose, assumptions, and applications of variance and covariance analysis in research.
	Apply One-way and Two-way ANOVA – Analyze data involving one or more independent variables to test for significant differences between group means.
	Interpret the results of ANOVA and ANCOVA – Understand F-statistics, significance values, and post-hoc test results to make valid inferences.
Content Outline	• **Analysis of Variance and Co-variance ANOVA** One way ANOVA, Two Way ANOVA, ANOCOVA Assumptions in ANOCOVA, Multivariate Analysis Technique Classification of Multivariate Analysis, factor Analysis, R-type Q Type, factor Analysis, Path Analysis

References:

1. Kothari, C. R., & Garg, G. (2019). Research methodology: Methods and techniques (4th ed.). New Age International Publishers.
2. Kumar, R. (2019). Research methodology: A step-by-step guide for beginners (5th ed.). SAGE Publications.
3. Cooper, D. R., & Schindler, P. S. (2014). Business research methods (12th ed.). McGraw-Hill Education.
4. Creswell, J. W., & Creswell, J. D. (2018). Research design: Qualitative, quantitative, and mixed methods approaches (5th ed.). SAGE Publications.
5. Zikmund, W. G., Babin, B. J., Carr, J. C., & Griffin, M. (2012). Business research methods (9th ed.). Cengage Learning.
6. Krishnaswamy, K. N., Sivakumar, A. I., & Mathirajan, M. (2009). Management research methodology: Integration of principles, methods and techniques. Pearson Education India.
7. Dawson, C. (2002). Practical research methods: A user-friendly guide to mastering research techniques and projects. How to Books.

Classroom Activities

Activities towards Comprehensive Continuous Evaluation(CCE)

Module 1:

Research Proposal Development:

Assignment: Ask students to develop a research proposal for a hypothetical research study. The proposal should include a clear research problem statement, objectives, literature review, research questions/hypotheses, methodology, and potential challenges.

Critical Analysis of Research Articles:

Activity: Provide students with a set of research articles from different disciplines. Ask them to critically analyse the methodology section, identifying strengths and weaknesses, and discussing how they would improve the research design.

Module 2:

Survey Design and Implementation:

Assignment: Have students design a survey on a topic of their choice.They should consider question wording, response options, and survey structure. After designing the survey, ask them to administer it to a small sample and analyse the results.

Module 3:

Activity: Assign students a qualitative research design task. They can choose a research question and develop a plan for data collection (e.g.,interviews, focus groups, observation).

Module 4:

Peer Review Simulation:

Activity: Have students conduct a peer review of a research proposal or a manuscript. This can include evaluating the clarity of the research question, appropriateness of methodology, and overall rigor of the study.

Assessment

Internal Assessment – 50 Marks

Evaluation Scheme:

Depending on the activities mentioned above a project should be developed for 50 marks. The internal assessment, which is a project evaluation, will be done by conducting a project presentation at the College level, where an External Examiner (Industry Expert or Subject Expert) appointed by the College will be evaluating the project depending on evaluation rubrics given below.

The Rubric will have the following Evaluation Parameters:		
Evaluation Parameters	**Description / Evaluation Points**	**Marks**
Understanding of Research Concepts	Demonstrates understanding of RM concepts, terms, and techniques	10
Problem Formulation & Hypothesis Framing	Ability to identify and define a research problem and objectives	10
Research Design & Method Selection	Selection and justification of research design, tools, sampling techniques	10

Data Collection & Analysis	Quality and relevance of collected data; proper statistical treatment	10
Report Writing & Documentation	Structure, clarity, referencing, and academic tone of report	5
Presentation & Communication Skills	Ability to present findings clearly and confidently	5

(Given below are two sample projects but it is expected to work on similar sort of projects)

Project 1:

Research Presentation:

Assignment: Ask students to create a presentation summarizing a research paper. They should highlight the key elements of the study, discuss the methodology, and present the findings. Encourage a focus on effective communication.

Project 2:

Case Study Analysis:

Activity: Provide students with a research-related case study involving methodological challenges. Ask them to analyze the case, identify issues, and propose solutions based on their understanding of research methodology.

External Assessment: (Marks 50)

End Semester examination of 50 marks for 2 hours duration will be conducted.

MSc (Computer Science) Semester II(NEP pattern)
Major Core

Course Title	**Artificial Intelligence**
Course Credit	**4 Credits**
Course Outcome	**CO1**: Understand the fundamental concepts, history, and evolution of Artificial Intelligence, including key domains and applications.
	CO2: Apply problem-solving techniques such as search algorithms (uninformed and informed searches) and game playing strategies for developing intelligent solutions.
	CO3: Demonstrate the ability to implement knowledge representation methods like logic-based models, semantic networks, and ontologies for AI problem domains.
Module 1(Credit 1)	**Introduction to Artificial Intelligence**
Learning Outcome	**After learning the modules,learners will be able to**
	Define Artificial Intelligence and explain its scope, history, and evolution.
	Identify key areas and applications of AI across various industries like healthcare, education, finance, and automation.
	Illustrate real-world examples of AI systems and discuss how they improve efficiency and decision-making.
Content Outline	**Definition, History & Evolution of AI** Types of AI: Narrow AI, General AI, Super AI Applications of AI in various sectors **Intelligent Agents:** Types, Structures, and Environments Solving by searching: Travelling salesman Problem, MonkeyBanana Problem, Tower of Hanoi and Block World.
Module2 (Credit 1)	**Problem solving by Searching Methods**
Learning Outcome	**After learning the modules,learners will be able to**
	Explain the role of search in problem-solving within Artificial Intelligence.
	Differentiate between various search strategies, including uninformed (blind) and informed (heuristic) search techniques.

	Apply informed search techniques like Greedy Best-First Search and A* Search for efficient problem-solving.
Content Outline	**Uninformed Search Techniques** ● Breadth-First Search (BFS) ● Depth-First Search (DFS) ● Iterative Deepening Search **Informed Search Techniques** ● Best-First Search ● A* Search ● Hill Climbing **Adversarial Search** ● Game Playing: Minimax Algorithm ● Alpha-Beta Pruning
Module3 (Credit 1)	**Knowledge Representation and Reasoning**
Learning Outcome	**After learning the modules,learners will be able to**
	Describe the importance of knowledge representation in building intelligent systems.
	Apply logic-based methods (Propositional and Predicate Logic) to represent facts, rules, and relationships in problem domains.
	Implement reasoning techniques such as forward chaining, backward chaining, and resolution to infer new knowledge from existing information.
Content Outline	**Knowledge Representation Techniques** ● Propositional Logic ● Predicate Logic ● Semantic Networks ● Frames and Ontologies **Reasoning Methods** ● Forward Chaining ● Backward Chaining ● Resolution **Reasoning under Uncertainty** ● Bayesian Networks ● Fuzzy Logic and Reasoning
Module 4(Credit 1)	**Knowledge Representation**
Learning Outcome	**After learning the modules,learners will be able to**
	Explain various knowledge representation techniques such as logic-based models, semantic networks, frames, and ontologies for structuring information in AI systems.

	Describe the architecture and components of Expert Systems, including knowledge base, inference engine, and user interface.
	Identify challenges and limitations in representing knowledge, reasoning under uncertainty, and processing natural language.
Content Outline	**Structured Knowledge Representation:** Associative Networks, Semantic Nets, Frames Structures, Conceptual, Dependencies & Scripts, Learning – Concept of Learning – Learning Automata, Learning by induction. **Natural Language Processing:** Overview of Linguistics, Grammars and Languages, basic Parsing techniques, semantic analysis, and representation structures. Natural Language generation and Natural Language Systems. **Expert System:** Knowledge acquisition methods, knowledge engineering process, goals in knowledge system development, basic architecture of expert systems, problem domain versus knowledge domain, Development of ES and life cycle of ES. Advantages of expert system, structure of Rule based expert system, characteristics of conventional system and expert system.

References:

1. Introduction to AI and Expert Systems - Patterson.
2. Artificial Intelligence A Modern Approach- Stuart Rushell.
3. Artificial Intelligence - Rich E and Knight K.
4. Principles of Artificial Intelligence - Nilsson.
5. Artificial Intelligence – An Engineering Approach – Schalkoff R J.
6. Introduction to Expert System - Peter Jackson.
7. Artificial Intelligence – Janakiraman.

Evaluation:
Internal Evaluation: 50 Marks

External Assessment: (Marks 50)

End Semester examination of 50 marks for 2 hours duration will be conducted.

Major (Core)

Course Title	**Advance DBMS**
Course Credits	**4 Credits**
Course Outcomes	**CO1**:-Students will grasp the theoretical foundations of advanced database systems, including object-oriented databases, parallel databases, and distributed databases.
	CO2:-Students will be able to understand the basic concepts of transaction management, including various concurrency control protocols and recovery mechanisms.
	CO3:- Students will be able to analyze the architecture and design of distributed and parallel databases, and understand data distribution, query processing, and system coordination.
	CO4:-Students will be able to differentiate NoSQL databases from traditional RDBMS, and explain the need for NoSQL in handling big data, horizontal scalability, and schema flexibility.
Module 1 (Credit 1) Transaction Management and Concurrency Control	
Learning Outcomes	**After learning the modules,Learners will be able to**
	1. Understand the concept of Transactions.
	2. Explain the need for concurrency control in multiuser database environments. Understand and apply locking protocol.
	3.Determine the serializability of transaction schedules using conflict and view serializability.
Content Outline	**Introduction to Transaction**: - Define the purpose of Transaction, ACID property, Tra state. **Transaction Processing**: - Transaction life cycle, System and user-define transaction, s e vs concurrent execution **Concurrency Control**: - Need for concurrency in DBMS, Problems in concurrency, S and their type. **Serializability**: - Conflict serializability, view serializability. **Concurrency Control Protocol**: - Lock-based protocol Binary locks Shared and exclusive locks, Two-phase locking (2PL) protocol, Strict 2PL and rigorous 2PL,Timestamp-based protocol.
Module 2 (Credit 1) Introduction to Advance Database Model	
Learning Outcomes	**After learning the modules,Learners will be able to**
	1.Compare and contrast advanced data models with traditional relational models.

	2. Develop solutions involving object-oriented, temporal, or spatial data using relevant tools or extensions.
	3. Analyze use-cases and select appropriate advanced database models for different application domains
Content Outline	**Introduction to Object-Oriented Databases**: - Features of OODB, Object Oriented Data Models (e.g. ODMG, ORDBMS extensions **Introduction to Temporial Databases**: - Difference between conventional databases and temporal databases. Types of temporal data: valid time, transaction time, bitemporal, temporal Data Model – time-stamped tuples, attribute-version, Event-based models. **Introduction to Spatial Databases: -** Definition and importance of spatial databases, Applications: GIS, urban planning, environmental monitoring, LBS (location-based services), Spatial data types: Points, Lines, Polygons,Raster vs. Vector data models **Introduction to Deductive Databases:** - Definition and motivation, Differences between relational and deductive databases ,Applications of deductive databases (rule-based systems, expert systems, knowledge bases)
Module 3 (Credit 1) Distributed and Parallel Databases	
Learning Outcomes	**After learning the modules,Learners will be able to** 1.Understand the need for distributed and parallel database systems.
	2.Differentiate between centralized, distributed, and parallel database systems.
	3.Describe the architectures of distributed and parallel database systems and identify the challenges and trade-offs associated with distributed and parallel database systems.
Content Outline	**Introduction to Distributed Databases**:- Centralized vs. Distributed vs. Parallel Systems. Need for distributed and parallel databases. Performance metrics (e.g., speedup, scale-up). Basic concepts: fragmentation, replication, and distribution. **Distributed Database Architectures**:- Shared-nothing, shared-disk, and shared-memory architectures. Examples of distributed database systems. **Introduction to Distributed Database Design**: - Data fragmentation (horizontal, vertical, derived). Data allocation strategies.
Module 4 (Credit 1) NoSQL Databases	
Learning Outcomes	**After learning the module, learners will be able to** 1 .Explain the differences between NoSQL and traditional relational databases.
	2. Describe the need for NoSQL in handling big data, scalability, and flexible schemas.
	3.Understand BASE properties and how they differ from ACID properties

Content Outline	**Introduction to NoSQL** :- What is NoSQL? , Comparison with relational databases, **Characteristics of NoSQL databases**:- Schema-free design, Horizontal scalability, BASE vs. ACID properties **Types of NoSQL Data Models**:- Classification of NoSQL databases: Key-Value Stores (e.g., Redis, Riak); Document Stores (e.g., MongoDB, CouchDB); Column-Family Stores (e.g., Cassandra, HBase) ;Graph Databases (e.g., Neo4j)

References:-

1. "Database System Concepts" Authors: Abraham Silberschatz, Henry F. Korth, and S. Sudarshan, Publisher: McGraw-Hill
2. "Fundamentals of Database Systems" Authors: Ramez Elmasri and Shamkant B. Navathe, Publisher: Pearson
3. "Advanced Database Management System" Author: Dr. Rini Chakravarty Publisher: Dreamtech Press
4. NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence" Authors: Pramod J. Sadalage and Martin Fowler, Publisher: Addison-Wesley
5. Principles of Distributed Database Systems Authors: M. Tamer Özsu and Patrick Valduriez Publisher: Springer

Assignments/Activities towards Comprehensive Continuous Evaluation (CCE)

<u>Classroom Activities:</u>

Activities towards Comprehensive Continuous Evaluation (CCE)

Module 1:

1. Use of SQL commands for managing transactions: BEGIN, COMMIT, ROLLBACK, SAVEPOINT.

Module 2:

1. Case study on design the OOAD schema.
2. Short quiz or viva on spatial data types , deductive database, temporal database.

Module 3:

1.Design a simple distributed database schema for a fictional organization with multiple branch offices (e.g., a university or retail chain). Include: Schema design with tables, Distribution strategy (horizontal/vertical fragmentation or replication) , Justification for your design choices.

Module 4:

1. Set up and interact with at least one NoSQL database tool (e.g., MongoDB, Redis, or Neo4j).
2. Introduced and explored any one database tools for handling any advance database e.g. explore MongoDB .

Assessment

Internal Assessment – 50 Marks

Evaluation Scheme: -

Depending on the activities mentioned above a project should be developed for 50 marks. The internal assessment, which is a project evaluation, will be done by conducting a project presentation at the College level, where an External Examiner (Industry Expert or Subject Expert) appointed by the College will be evaluating the project depending on evaluation rubrics given below.

The Rubric will have the following Evaluation Parameters:		
Evaluation Parameters	**Description / Evaluation Points**	**Marks**
Remembering	The basic definition of the transaction analysis, ACID concept, other advance concept about the advance databases	10
Understanding	Understand the schema design of the different type of databases	10
Applying	The student can design the database schema for different type of databases through case study	10
Creating	Install the MongoDB and operate it.	15
Submission	Completeness, correctness, formatting, and timely submission	5
	Total	50

(Given below are two sample projects but it is expected to work on similar sort of projects)

Project 1:

1. Using MongoDB create the Database schema and execute queries on it.

Project 2:

2. Design the case study for creation of database Schema for OORDBMs, Spital database.

External Assessment: (Marks 50)

End Semester examination of 50 marks for 2 hours duration will be conducted.

Major (Core)

Course Title	**Data Science Using Python**
Course Credits	**4 Credits**
Course Outcomes	**CO1**: Students will be able to define the concept, scope, and evolution of Data Science and differentiate between Data Science, Machine Learning, and Artificial Intelligence.
	CO2: Students will be able to explain the phases of the Data Science lifecycle, including data collection, preprocessing, modeling, evaluation, and deployment.
	CO3: Students will be able to apply descriptive statistics and visualization techniques to summarize and interpret datasets, and identify patterns, trends, and anomalies in data.
	CO4 : Students will be able to describe basic machine learning algorithms (e.g., regression, classification, clustering) and explain the principles of model training, testing, and evaluation.
	CO5: Students will be able to discuss issues related to data privacy, bias, and fairness, and evaluate the importance of ethical data handling and responsible AI.
Module 1 (Credit 1) Introduction to Data Science and Python	
Learning Outcomes	**After learning the modules**,**Learners will be able to**
	1. Understand the basic concepts, scope, and components of Data Science.
	2. Familiarize with data types, sources, and preprocessing techniques.
	3.Learn the foundational tools and applications of data science in real-world problems.
Content Outline	**Overview of Data Science**: What is data science, its applications, Difference between Data Science, Data Analytics, and Business Intelligence, The Data Science lifecycle (CRISP-DM or OSEMN model) ,Roles in data science: Data Scientist, Analyst, Engineer, etc. Applications across industries (healthcare, finance, education, etc.) **Data Science Components** :- Structured, Semi-structured, and Unstructured data , Introduction to Big Data ,Data Collection: Sources and Methods, Data Warehousing and Data Lakes , Data preprocessing: Cleaning, Transformation, Normalization
Module 2 (Credit 1) Data Manipulation , Analysis and Visualization with Python	
Learning Outcomes	**After learning the modules**,**Learners will be able to**
	1. Understand the structure and advantages of NumPy
	2. Create and manipulate Pandas Series and DataFrames for structured data.

	3. Create basic visualizations such as line plots, bar charts, and scatter plots using Matplotlib
Content Outline	**NumPy**: Working with arrays, numerical operations, linear algebra, and array manipulation. **Pandas**: Data structures like Series and DataFrames, data cleaning, filtering, and transformation. **Data Input/Output**: Reading and writing data from various formats (CSV, Excel, etc.). **Matplotlib**: Creating basic plots, charts, and graphs to visualize data. **Seaborn**: Advanced visualization techniques, statistical plots, and creating visually appealing charts. Exploratory Data Analysis (EDA) :-Descriptive statistics (mean, median, mode, std. dev.), Data visualization techniques: Histograms, Box plots, Scatter plots ,Correlation and Causation Outlier detection and missing value handling.
Module 3 (Credit 1)	**Statistical Concepts**
Learning Outcomes	**After learning the modules,Learners will be able to**
	. 1. Define and compute measures of central tendency (mean, median, mode) to summarize data. Calculate measures of dispersion (range, variance, standard deviation) to describe data spread.
	2.Interpret data distribution using graphical techniques (histograms, boxplots) and numerical summaries.
	3. Explain the concept of population vs. sample and the need for inferential statistics.
	4.Understand fundamental probability concepts including independent, dependent events, and conditional probability.
Content Outline	**Descriptive Statistics**: Measures of central tendency, dispersion, and distribution. **Inferential Statistics**: Hypothesis testing, confidence intervals, and statistical significance.**Probability:** Basic probability concepts, distributions (normal, binomial, etc.).
Module 4 (Credit 1)	**Machine Learning model Development**
Learning Outcomes	**After learning the modules,Learners will be able to**
	1. Use Scikit-learn to build machine learning pipelines including data preprocessing, model training, and evaluation.
	2. Build and evaluate linear regression and logistic regression models for predictive tasks. Apply classification algorithms like Decision Trees and Support Vector Machines (SVM) to labeled datasets.
	3. Implement K-Means and Hierarchical clustering using Scikit-learn. Interpret clustering results and visualize groupings effectively.

Content Outline	**Introduction to Scikit-learn**:- Using the Scikit-learn library for building and evaluating machine learning models. **Supervised Learning**:- Linear regression, logistic regression, classification algorithms (e.g., decision trees, SVM). **Unsupervised Learning**:-Clustering (k-means, hierarchical clustering). **Model Evaluation**:-Metrics for evaluating model performance (e.g., accuracy, precision, recall).

References:-

1. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, Author: Aurélien Géron, Publisher: O'Reilly Media
2. Data Science from Scratch: First Principles with Python , Author: Joel Grus, Publisher: O'Reilly Media
3. Think Stats: Probability and Statistics for Programmers ,Author: Allen B. Downey,Publisher: O'Reilly / Green Tea Press
4. Practical Statistics for Data Scientists , Authors: Peter Bruce, Andrew Bruce, and Peter Gedeck Publisher: O'Reilly Media
5. Introduction to Machine Learning with Python Authors: Andreas C. Müller & Sarah Guido Publisher: O'Reilly Media

Assignments/Activities towards Comprehensive Continuous Evaluation (CCE)

Classroom Activities

Module 1: Python Basics for Data Science

Activity 1. Assignment on using lists, tuples, dictionaries, and string manipulations in a dataset-like context.

Activity 2: Implement NumPy operations (array creation, slicing, reshaping, and broadcasting).

Activity 3: Use Pandas to load a CSV dataset, clean missing values, and perform filtering and aggregation.

Module 2: Data Visualization

Activity 1: Create plots using Matplotlib – line chart, bar graph, scatter plot with proper labels and titles.

Activity 2: Use Seaborn to generate correlation heatmaps, boxplots, and pairplots for a given dataset.

Module 3: Exploratory Data Analysis (EDA)

Activity 1: Perform EDA on a real-world dataset (e.g., Titanic, Iris, or custom). Summary statistics, distribution, missing values, correlation matrix.

Module 4 : Machine Learning with Scikit-learn

Activity 1: Build and evaluate a linear regression model using Scikit-learn.

Activity 2: Classification using Decision Tree or SVM; evaluate using accuracy, precision, recall.

Activity 3: Apply K-means clustering on a dataset and visualize the results.

Evaluation: Algorithm implementation, model accuracy, justification of choice of algorithm.

Assessment

Internal Assessment – 50 Marks

Evaluation Scheme: - Depending on the activities mentioned above a project should be developed for 50 marks. The internal assessment, which is a project evaluation, will be done by conducting a project presentation at the College level, where an External Examiner (Industry Expert or Subject Expert) appointed by the College will be evaluating the project depending on evaluation rubrics given below.

The Rubric will have the following Evaluation Parameters:		
Evaluation Parameters	**Description / Evaluation Points**	**Marks**
Remember- Syntax Knowledge (NumPy/Pandas)	· Students recall and use correct syntax of NumPy and Pandas. · Ability to write syntactically correct Python code using NumPy and Pandas.	10
Understand Statistical Concept	· Students understand core statistical concepts and know when to apply them to a given problem. · Correct identification of suitable statistical techniques (mean, std, correlation). · Justification for method choice in context.	10
Apply	· Students apply Python libraries and statistical techniques on real-world case studies or datasets. · Implement end-to-end data preprocessing, EDA, and insight extraction.	10
Analyze / Evaluate	· Students develop machine learning models and evaluate their performance using appropriate metrics. · Model selection, training, and testing. · Evaluation using accuracy, precision, recall, etc.	15
	· Completeness, correctness, formatting, and timely submission	5
	Total	50

(Given below are two sample projects but it is expected to work on similar sort of projects)

Project Work (Summative)

- Select a dataset (Kaggle/UCI).
- Perform EDA, build ML models, visualize results.
- Submit report + present findings.
- Evaluation Criteria:
- Problem statement clarity
- Code and data analysis
- Interpretation and insights
- Oral presentation & report

Major (Elective II)

Course Title	**Software Testing and Quality assurance**
Course Credits	**4 Credits**
	CO1- Understand the fundamental principles and concepts of software testing, including its purpose, objectives, and the role it plays in software development.
	CO2- Apply various testing methods such as functional, non-functional, unit, integration, system, regression, and acceptance testing to diverse software systems.
	CO3- Explain software quality concepts and quality assurance practices, models, and standards like ISO and CMMI.
	CO4- Enable students to evaluate software quality using metrics, process audits, and quality improvement strategies.
	CO 5- Understand quality assurance principles and best practices,emphasizing the importance of testing in the software development lifecycle.
Module 1 (Credit 1) **Fundamentals of Software Testing**	
Learning Outcomes	**After learning the module, learners will be able to**
	Gain proficiency in software testing terminologies, methodologies, and life cycles.
	Describe key testing concepts, including error, fault, failure, defect, and bug.
	Differentiate between software testing types such as manual vs automated testing, static vs dynamic testing.
	Recognize the principles and objectives of testing, including early testing, defect clustering

	Understand and illustrate the phases of the Software Testing Life Cycle (STLC)
Content Outline	**Module Contents:** **Introduction to Software Testing** ● Goals and Principles of Testing ● Software Development Life Cycle (SDLC) vs Software Testing Life Cycle (STLC) ● V-Model and Agile Testing Overview ● Types of Testing: Manual vs Automated, Static vs Dynamic ● Defect Lifecycle, Severity & Priority
Module 2 (Credit 1) **Testing Techniques**	
Learning Outcomes	**After learning the module, learners will be able to**
	Differentiate between black-box and white-box testing techniques and identify when to apply each in real-world scenarios.
	How to apply black-box testing techniques and white-box testing techniques
	Analyze code and specifications to design appropriate test cases using both black-box and white-box methods.
	Explain the significance of static techniques, including code reviews and walkthroughs, in improving software quality.
	Select suitable testing techniques based on software requirements and risk analysis
Content Outline	**Module Contents:** **Black Box Testing Techniques** ● Equivalence Partitioning ● Boundary Value Analysis ● Cause-Effect Graphing ● Decision Table Testing **White Box Testing Techniques** ● Statement, Branch, Condition, and Path Coverage ● Cyclomatic Complexity ● Code Reviews and Walkthroughs
Module 3 (Credit 1)	

Test Management	
Learning Outcomes	**After learning the module, learners will be able to** Understand the components of test management, including test planning, scheduling, monitoring, and control.
	Create a comprehensive test plan, including scope, objectives, resources, schedules, and deliverables. Design and document effective test cases and test scenarios based on software requirements.
	Plan and perform test closure activities, including final reporting, test summary creation, and lessons learned
	Understand configuration and version control and their importance in maintaining test artifacts.
	Coordinate testing efforts within teams, ensuring collaboration, communication, and adherence to deadlines
Content Outline	● Test Planning and Documentation ● Test Strategy, Test Plan Components ● Test Case Design and Test Data Preparation ● Test Environment Setup ● Test Execution and Defect Reporting ● Test Closure Activities ● Configuration Management and Version Control
Module 4 (Credit 1) **Software Quality Assurance (SQA) and Automation and Tools**	
Learning Outcomes	**After learning the module, learners will be able to**
	Identify and apply key software quality attributes, including functionality, reliability, usability, efficiency, maintainability, and portability.
	Apply SQA practices such as reviews, audits, and inspections to improve software quality and reduce defects.
	Differentiate between process assurance and product assurance, and explain their relevance in a quality framework
	Analyze software process improvement models, such as CMMI, Six Sigma, and Total Quality Management (TQM), and their impact on software projects.
	Differentiate between various automation frameworks, such as linear, modular, data-driven, keyword-driven, and hybrid frameworks.
	Use popular automation tools (e.g., Selenium, JUnit/TestNG) to write, execute, and manage automated test scripts.Apply bug-tracking tools (e.g., Bugzilla, Mantis) to report, manage, and track defects throughout the test lifecycle.

	Critically evaluate organizational quality assurance strategies, and recommend improvements based on industry best practices.
Content Outline	**Concepts of Software Quality** ● Quality Factors (McCall's and ISO 9126) ● SQA Activities and Audits ● Reviews: Formal Technical Reviews, Inspections ● Software Process Improvement Models: 1. Capability Maturity Model Integration (CMMI) 2. ISO 9001:2015 3. Six Sigma in Software Engineering **Automation and Tools** Basics of Test Automation Benefits and Challenges of Automation Frameworks: Linear, Modular, Data-Driven, Keyword-Driven, Hybrid Overview of Tools: ● Selenium ● JUnit/TestNG ● Bug Tracking: Bugzilla, Mantis ● CI Tools: Jenkins Introduction to Performance Testing (LoadRunner, JMeter)

Assignments towards Comprehensive Continuous Evaluation

Module 1:

1. Explain the software testing life cycle (STLC) with diagrams.
2. Describe the role of a test plan. What are its key components?
3. Explain the significance of early testing in SDLC.
4. Provide a case study of an organization's testing process and have students analyze and propose improvements based on maturity model principles.
5. Case study -A software product is nearing completion. As a test engineer, what approach will you take to ensure it meets quality standards?

Module 2:

1. Differentiate between black-box and white-box testing techniques.
2. What is boundary value analysis? Why is it important?
3. State any two advantages of using decision table-based testing
4. Explain the following black-box testing techniques with examples.
   a) Equivalence Class Partitioning
   b) Boundary Value Analysis
5. Describe the following white-box testing techniques with examples:
   a) Statement Coverage

b) Branch Coverage

6. Design test cases using equivalence partitioning and boundary value analysis for a function that accepts age input between 18 and 60.
7. Given a simple program to calculate grades based on percentage, create a decision table and generate test cases.

Module 3:

1. What is test management? Define the role of a test plan in test management.
2. Differentiate between test case and test scenario.
3. Describe the test management activities throughout the Software Testing Life Cycle (STLC).
4. What are the types of risks in testing? How are they managed?
5. Discuss the importance of traceability in test management. How is a traceability matrix prepared and used?
6. Describe the challenges in test management for large-scale enterprise software and how they can be addressed.
7. Evaluate the role of a test manager in Agile and DevOps environments.
8. You are the test lead for an e-commerce project. Outline how you will manage:
   - Test planning
   - Test team coordination
   - Defect reporting
   - Test metrics tracking
9. Given a project with limited time and resources, how would you prioritize and manage test activities effectively?

Module 4:

1. Define Software Quality Assurance. How does it differ from software testing?
2. What are the different types of software quality standards? Give examples and explain their importance.
3. Discuss the role of metrics in SQA. What types of metrics are commonly used to measure software quality?
4. Describe the difference between defect prevention and defect detection in the context of SQA.
5. What is test automation? Explain its benefits and limitations in software testing.
6. Explain the difference between automated testing and manual testing. When should automation be preferred?
7. What is Continuous Integration (CI), and how do automated tests fit into CI pipelines?
8. Describe the process of selecting an automation tool for a software project. What factors should be considered?

Reference Books :

1. Software Testing Principles and Practices – Srinivasan Desikan, Gopalaswamy Ramesh
2. Foundations of Software Testing – Dorothy Graham, Erik van Veenendaal, Isabel Evans
3. Software Engineering – Ian Sommerville (Relevant QA chapters)
4. Effective Software Testing – Elfriede Dustin

5. Software Quality Assurance – Daniel Galin
6. Testing Computer Software – Cem Kaner

Assessment:
Internal Assessment – 50 Marks

The Rubric will have the following Evaluation Parameters:

Criteria	Excellent (9-10 marks)	Good (7-8 marks)	Satisfactory (5-6 marks)	Needs Improvement (0-4 marks)	Marks
1. Understanding of Concepts	Deep understanding of testing fundamentals, QA principles, methodologies; clear explanation with examples.	Good understanding with minor gaps; explains most concepts well.	Basic understanding; some misconceptions or incomplete explanations.	Poor understanding; many misconceptions or unclear explanations.	/10
2. Application of Testing Techniques	Correctly applies unit, integration, system, acceptance testing techniques in practical scenarios.	Applies most techniques correctly with minor errors or omissions.	Applies basic techniques but misses advanced or appropriate application.	Fails to apply correct testing techniques or misunderstands them.	/10
3. Quality Assurance Processes	Clearly explains and applies QA processes including planning, defect management, metrics, standards.	Explains QA processes with minor inaccuracies or omissions.	Basic description of QA processes but lacks depth or practical understanding.	Limited or incorrect understanding of QA processes.	/10
4. Use of Tools & Automation	Proficient use of testing tools (Selenium, JIRA, JUnit) and automation frameworks; explains	Uses tools adequately but with limited depth or some errors.	Basic use of tools; mostly manual testing or unclear on automation.	No or incorrect use of testing tools or automation.	/10

	advantages.				
5. Critical Analysis & Reporting	Produces clear, structured test reports; critically analyses results, defects, suggests,improvements.	Produces acceptable reports; some analysis present but lacks depth.	Reports are basic with limited analysis or unclear presentation.	Poor or no reports; lacks analysis and critical thinking.	/10

External Assessment: (Marks 50)

End Semester examination of 50 marks for 2 hours duration will be conducted.

Major (Elective)

Course Title	**Fuzzy Logic and Neural Networks**
Course Credits	**4 Credits**
Course Outcomes	**CO1**: Understand the fundamental concepts of fuzzy logic systems
	CO2: Understand the structure and functioning of biological and artificial neural networks
	CO3: Implement learning algorithms for single-layer and multi-layer neural networks
	CO4 : Analyze and compare the performance of neural and fuzzy systems for various applications

Module 1 (Credit 1) Fundamentals of Fuzzy Logic	
Learning Outcomes	**After learning the modules,Learners will be able to**
	1. **Differentiate** between classical (crisp) sets and fuzzy sets in terms of uncerta handling.
	2. **Explain** the concepts of membership functions and perform fuzzy set operations like union, intersection, and complement.
	3.**Construct** fuzzy relations and apply fuzzy composition techniques.
Content Outline	Crisp sets vs Fuzzy sets, Fuzzy set operations (union, intersection, complement), Properties of fuzzy sets, Fuzzy relations and fuzzy composition, Membership functions: types and properties, Fuzzification and defuzzification techniques, Applications of fuzzy logic in control systems
Module 2 (Credit 1) Fuzzy Inference and Reasoning Systems	
Learning Outcomes	**After learning the modules,Learners will be able to**
	1. **Explain** the structure and components of a fuzzy rule-based system including linguistic variables and fuzzy rules.
	2. **Implement** fuzzy rule evaluation and approximate reasoning to derive conclusions from imprecise inputs.
	3. **Analyze** and compare the behavior of different inference mechanisms.
Content Outline	Linguistic variables and fuzzy rules, Mamdani and Sugeno fuzzy models, Inference mechanism and rule evaluation, Approximate reasoning, Design of fuzzy logic controllers, Real-world applications: Industrial control, smart appliances
Module 3 (Credit 1) Fundamentals of Neural Networks	
Learning Outcomes	**After learning the modules,Learners will be able to**
	1.**Explain** the structure and functioning of biological and artificial neurons.
	2.**Differentiate** between various neural network architectures such as feedforward and feedback networks.

	3.**Describe** the working of basic neural network models like Perceptron, ADALINE, and MADALINE.
	4.**Solve** simple classification and pattern recognition problems using basic neural models.
Content Outline	Biological neuron vs artificial neuron, Neural network models: Perceptron, ADALINE, MADALINE, Activation functions: Sigmoid, Tanh, ReLU, Learning rules: Hebbian, Delta, Perceptron rule, Multi-layer perceptrons (MLP), Backpropagation algorithm: derivation and working
Module 4 (Credit 1)	**Advanced Neural Networks and Neuro-Fuzzy Systems**
Learning Outcomes	**After learning the modules**,**Learners will be able to**
	1. **Describe** the structure and working of advanced neural network models such as Self-Organizing Maps (SOM), Radial Basis Function (RBF) Networks, and Hopfield Networks.
	2. **Explain** the concept and architecture of Neuro-Fuzzy systems like ANFIS (Adaptive Neuro-Fuzzy Inference System).
	3. **Apply** advanced neural and neuro-fuzzy models for solving complex problems in classification, control systems, and prediction.
Content Outline	Self-organizing maps (SOM), Radial basis function networks, Hopfield networks (brief overview), Introduction to Neuro-Fuzzy systems (ANFIS), Hybrid systems combining fuzzy logic and neural networks, Case studies and real-world applications (classification, prediction)

References:

1. Ross, T. J. (2010). *Fuzzy logic with engineering applications* (3rd ed.). Wiley.
2. Haykin, S. (2009). *Neural networks and learning machines* (3rd ed.). Pearson Education.
3. Rajasekaran, S., & Vijayalakshmi Pai, G. A. (2003). *Neural networks, fuzzy logic and genetic algorithms: Synthesis and applications*. PHI Learning.
4. Kosko, B. (1992). *Neural networks and fuzzy systems: A dynamical systems approach to machine intelligence*. Prentice Hall.
5. Jang, J. S. R., Sun, C. T., & Mizutani, E. (1997). *Neuro-fuzzy and soft computing: A computational approach to learning and machine intelligence*. Pearson Education.

6. Fausett, L. (1994). *Fundamentals of neural networks: Architectures, algorithms and applications*. Prentice Hall.

Assignments towards Comprehensive Continuous Evaluation
Module 1: 1. Define fuzzy sets and explain how they differ from classical sets with examples. 2. Draw and describe different types of membership functions (triangular, trapezoidal, Gaussian). 3. Perform fuzzy set operations (union, intersection, complement) on given data. 4. Design a fuzzification and defuzzification process for a washing machine temperature control system.
Module 2: 1. Construct a Mamdani fuzzy inference system for a traffic control problem using at least three linguistic variables. 2. Implement a simple Sugeno model using Python/Matlab/Excel. 3. Create a fuzzy rule base for an air conditioner control system. 4. Discuss the role of approximate reasoning with a real-world scenario (e.g., medical diagnosis or flood prediction).
Module 3: 1. Explain the difference between biological and artificial neurons with diagrams. 2. Implement a single-layer perceptron model in Python to solve the AND/OR/XOR problem. 3. Use Python to simulate the working of an ADALINE model with a small dataset. 4. Analyze the effect of different activation functions on neural output with graphical representation.
Module 4: 1. Develop a simple Self-Organizing Map (SOM) using a given 2D dataset and visualize the clustering. 2. Implement an Adaptive Neuro-Fuzzy Inference System (ANFIS) using any tool (Python, MATLAB, etc.) for a classification problem. 3. Write a comparative analysis of neural networks vs fuzzy logic vs neuro-fuzzy systems. 4. Case Study: Design and document a hybrid model for medical diagnosis, weather prediction, or stock trend prediction.

Assessment

Internal Assessment – 50 Marks

Evaluation Scheme: - Depending on the activities mentioned above a project should be developed for 50 marks. The internal assessment, which is a project evaluation, will be done by

conducting a project presentation at the College level, where an External Examiner (Industry Expert or Subject Expert) appointed by the College will be evaluating the project depending on evaluation rubrics given below..

The Rubric will have the following Evaluation Parameters:		
Evaluation Parameters	**Description / Evaluation Points**	**Marks**
Understanding of Concepts	Ability to clearly explain theoretical concepts related to fuzzy logic or neural networks.	10
Problem Solving / System Design	Design and structuring of fuzzy inference systems or neural models as per problem statements.	10
Code / Implementation Accuracy	Functionality and correctness of the developed model, code execution, and adherence to logic.	10
Report and Documentation	Clarity of report writing, logical structure, diagrams, input-output explanation, formatting.	10
Presentation & Timely Submission	Neatness, formatting, adherence to submission deadline, and overall presentation of work.	10

Sample Projects

1. XOR Problem Using Single Layer and Multi-Layer Perceptron

- Objective: Train and compare a Perceptron and a Backpropagation Neural Network to solve the XOR logic problem.
- Tools: Python (TensorFlow, Keras, NumPy)
- Learning Outcome: Explore the limitations of single-layer networks and the significance of non-linearity and hidden layers.

2. Handwritten Digit Recognition using Neural Networks

- Objective: Build a neural network to classify handwritten digits (e.g., using the MNIST dataset).
- Tools: Python (Keras, TensorFlow, or PyTorch)
- Learning Outcome: Apply concepts of MLP, activation functions, backpropagation, and accuracy metrics.

External Assessment: (Marks 50)

End Semester examination of 50 marks for 2 hours duration will be conducted.

Major (Elective)

Course Title	**Ethical Hacking**
Course Credits	**4 Credits**
	CO1- To introduce key concepts of ethical hacking and cybersecurity.
	CO2- To identify and analyze various security threats and vulnerabilities.

	CO3-To develop hands-on skills with ethical hacking tools and techniques.
	CO4- To understand legal frameworks and ethical responsibilities of security professionals.
Module 1 (Credit 1) **Introduction to Ethical Hacking and Cyber Laws**	
Learning Outcomes	**After learning the module, learners will be able to**
	Grasp the fundamentals of ethical hacking, cyber threats, attack vectors, and hacker methodologies.
	Use tools and techniques for reconnaissance, vulnerability assessment, and basic penetration testing in a legal environment.
	Identify and implement appropriate security measures and countermeasures such as firewalls, IDS/IPS, and encryption.
	Understand key cyber laws, ethical issues, and the legal responsibilities of ethical hackers in national and global contexts.
Content Outline	● Fundamentals of hacking: types of hackers, attack vectors ● Phases of ethical hacking: reconnaissance to covering tracks ● Cybersecurity principles: CIA triad, threat modeling ● Legal, professional, and ethical issues ● Overview of IT Act 2000 and amendments ● Cybercrime case studies (India and global)
Module 2 (Credit 1) **Reconnaissance, Scanning & Vulnerability Assessment**	
Learning Outcomes	**After learning the module, learners will be able to**
	Distinguish between active and passive reconnaissance and explain their role in the ethical hacking lifecycle.
	Utilize tools and methods for port scanning, network mapping, and service detection to gather target system information.
	Identify, analyze, and classify system vulnerabilities using automated and manual assessment tools.
	Evaluate scan and assessment results to determine security risks and prepare actionable reports for remediation.

Content Outline	**Module Contents:** ● Footprinting techniques: passive & active reconnaissance ● Tools: Whois, Nslookup, Recon-ng, Maltego, Google Dorking ● Network scanning: types of scans (TCP, SYN, UDP), ping sweep ● Banner grabbing, OS fingerprinting ● Enumeration techniques: SNMP, NetBIOS, LDAP ● Vulnerability assessment using Nessus, OpenVAS
Module 3 (Credit 1) **System Exploitation and Post-Exploitation Techniques**	
Learning Outcomes	**After learning the module, learners will be able to**
	Explain common exploitation techniques and vulnerabilities used to gain unauthorized access to systems.
	Use tools like Metasploit and other frameworks to exploit known vulnerabilities in a simulated environment.
	Demonstrate methods for privilege escalation, maintaining access, and data exfiltration after initial compromise.
	Evaluate the impact of successful exploitation and generate professional reports outlining findings and recommendations.
Content Outline	● Exploiting system vulnerabilities: buffer overflow, password attacks ● Privilege escalation techniques ● Malware: trojans, viruses, rootkits ● Backdoors and maintaining access ● Covering tracks: log deletion, anti-forensics ● Tools: Metasploit, Hydra, John the Ripper
Module 4 (Credit 1) **Web Application & Wireless Network Hacking**	
Learning Outcomes	**After learning the module, learners will be able to**
	Identify typical security flaws such as SQL injection, XSS, weak encryption, and insecure Wi-Fi protocols.
	Use appropriate tools and techniques to test web apps and wireless networks for vulnerabilities ethically.

	Recommend and apply security controls like input validation, secure configurations, and strong wireless encryption.
	Evaluate attack outcomes and prepare clear reports with mitigation strategies for stakeholders.
Content Outline	● Web application vulnerabilities: SQLi, XSS, CSRF, file inclusion ● Secure coding practices ● Web vulnerability scanning tools: OWASP ZAP, Burp Suite ● Wireless security: WEP/WPA/WPA2 vulnerabilities ● Wireless hacking tools: Aircrack-ng, Kismet ● Network sniffing with Wireshark ● Introduction to penetration testing reports and documentation

Reference Books :

1. Matt Walker, All-In-One-CEH-Certified-Ethical-Hacker-ExamGuide.
2. Tutorials Point Professionals, Ethical Hacking by Tutorials Point.
3. BIBLIOGRAPHY • Kimberly Graves(26th-April-2010), "CEH Certified Ethical Hacker Study Guide" 1st Edition, ISBN-13: 978-0470525203, ISBN-10: 0470525207, Sybex-Wiley Publishing. •
4. Sean-Philip Oriyano, Sybex, Certified Ethical Hacker Study Guide v9, Study Guide Edition, 2016.

Assignments towards Comprehensive Continuous Evaluation
Module 1: 5. Define ethical hacking. How does it differ from malicious hacking? 6. List and explain the phases of ethical hacking. 7. Discuss the legal and ethical considerations an ethical hacker must keep in mind before conducting a security test. 8. Discuss some real-world examples where ethical hacking helped prevent cyber attacks. 9. Explain the key provisions of the Information Technology Act, 2000 10. What is data protection law? How does it impact individuals and organizations? 11. What are the responsibilities of internet service providers (ISPs) under cyber law? 12. Explain the legal challenges in prosecuting cybercrimes across international borders.

Module 2:

1. What is reconnaissance in ethical hacking, and why is it considered the first step in the hacking process?
2. Differentiate between active and passive reconnaissance with suitable examples.
3. Explain the purpose of port scanning and name three tools used for this task.
4. What are the different types of scanning techniques used to identify vulnerabilities in a target system?
5. Define vulnerability assessment and explain how it helps in improving cybersecurity.
6. Describe the difference between vulnerability assessment and penetration testing.
7. What is the role of Common Vulnerabilities and Exposures (CVE) in vulnerability management?

Module 3:

1. What is system exploitation in ethical hacking? Explain its significance in penetration testing.
2. Describe the common types of system exploits used by ethical hackers.
3. What are buffer overflow attacks? How can they be exploited to gain system access?
4. Explain the concept of privilege escalation and its importance in post-exploitation.
5. What are backdoors and rootkits? How do they aid attackers after gaining system access?
6. Discuss the techniques used to maintain access to a compromised system during post-exploitation.
7. What is data exfiltration, and what methods do attackers use to extract sensitive information during post-exploitation?

Module 4:

1. What are the common vulnerabilities found in web applications? Explain with examples.
2. Describe the SQL Injection attack and how it can be prevented in web applications.
3. Explain Cross-Site Scripting (XSS) and its impact on web security.
4. What tools and techniques are commonly used for hacking wireless networks?
5. Discuss the differences between WEP, WPA, and WPA2 encryption in wireless security.
6. What is a man-in-the-middle (MITM) attack in wireless networks and how can it be mitigated?
7. Explain how session hijacking works in web applications and the measures to prevent it.

Assessment:
Internal Assessment – 50 Marks

The Rubric will have the following Evaluation Parameters:

Criteria	Excellent (9-10 marks)	Good (7-8 marks)	Satisfactory (5-6 marks)	Needs Improvement (0-4 marks)	Marks
1. Understanding of Ethical Hacking Concepts	Clear, in-depth understanding of ethical hacking principles, methodologies, legal and ethical considerations	Good understanding with minor gaps; some awareness of ethical/legal issues	Basic understanding, some misconceptions about ethical/legal aspects	Poor or incorrect understanding of ethical hacking concepts	10
2. Practical Skills and Tools Usage	Proficient use of multiple hacking tools (e.g., Nmap, Metasploit), demonstrating advanced techniques	Effective use of common tools with some advanced techniques	Basic usage of standard tools; limited application of techniques	Inadequate or incorrect use of tools; unable to demonstrate skills	10
3. Vulnerability Analysis & Reporting	Thorough and detailed vulnerability assessment with clear, actionable, and well-documented reports	Good vulnerability analysis with acceptable documentation	Basic identification of vulnerabilities; report lacks depth	Poor or incomplete vulnerability assessment and reporting	
4. Problem Solving & Critical Thinking	Demonstrates excellent analytical skills; proposes innovative and effective mitigation	Good problem-solving ability; mitigation strategies mostly effective	Identifies problems but solutions lack depth or feasibility	Unable to analyze or suggest practical solutions	10

	strategies				
5. Ethical & Legal Compliance Awareness	Demonstrates strong awareness of ethical standards and legal frameworks; adheres strictly to guidelines	Adequate awareness and compliance with ethical/legal standards	Some understanding but inconsistent application	Little or no awareness of ethical/legal considerations	10
6. Communication & Presentation	Presents work clearly and professionally with excellent organization and technical language	Clear presentation with minor clarity or organization issues	Basic communicatio n; some ambiguity or lack of detail	Poor communication; unclear or disorganized presentation	10

External Assessment: (Marks 50)

End Semester examination of 50 marks for 2 hours duration will be conducted.

Major (Elective)

Course Title	**Natural Language Processing**
Course Credits	**4 Credits**
Course Outcomes	**CO1**:- Understand the fundamentals of natural language processing and its role in artificial intelligence and machine learning.
	CO2:-Explain the structure of language using linguistic concepts such as morphology, syntax, semantics, and pragmatics.

	CO3:- Apply basic NLP techniques such as tokenization, stemming, lemmatization, POS tagging, and parsing to process raw text data.
	CO4:- Implement and evaluate classical NLP models including n-gram models, TF-IDF, and word embeddings (e.g., Word2Vec, GloVe).
	CO5:- Build machine learning models for NLP tasks such as text classification, sentiment analysis, and named entity recognition using libraries like NLTK, SpaCy, and Scikit-learn.
Module 1 (Credit 1)	**Introduction and Foundational Concepts**
Learning Outcomes	**After learning the module, learners will be able to**
	•Define the significance and necessity of natural language processing (NLP) in modern computing and communication systems. • How the text preprocessing is perform • Understand the construction of regular expression • Understand corpus and utilization of corpus in NLP
Content Outline	**What is NLP?** - Definition, applications, and the role of NLP in the broader field of artificial intelligence and computational linguistics. **Text Processing**: -Tokenization, stemming, lemmatization, and stop word removal. **Regular Expressions**:-Understanding and using regular expressions for text manipulation. **Corpora and Resources**:- Introduction to commonly used text corpora and NLP resources. **Python Libraries**:- Overview of Python libraries like NLTK, spaCy, and others commonly used in NLP.
Module 2 (Credit 1) Core NLP Techniques	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Differentiate between various levels of language processing (phonetics, syntax, and semantics) and comprehend their role in NLP. 2. Identify and analyze the key issues, challenges, and limitations in natural language processing. 3. Analyze and discuss diverse applications of NLP across various domains such as machine translation, sentiment analysis, information retrieval, catboats, and more.

Content Outline	**Language Modeling**: - Statistical language modeling, n-grams, smoothing techniques, and perplexity. **Part-of-Speech (POS) Tagging**: - Algorithms for assigning grammatical tags (noun, verb, etc.) to words. Type of POS Tagging **Parsing**:-Understanding syntactic structures, constituency parsing (phrase structure trees), and dependency parsing. **Types of parsing**: **Constituency and Dependency Parsing.** **Constituency Parsing**:- Top-down parsing (e.g., recursive descent, predictive parsing), Bottom-up parsing (e.g., shift-reduce parsing). **Dependency Parsing:-**Dependency relations and their representation, Transition-based dependency parsing (e.g., Arc-Standard, Arc-Eager). Graph-based dependency parsing. **Semantic Analysis**:- Lexical semantics, word sense disambiguation, distributional semantics (word embedding), and semantic role labeling.
Module 3 (Credit 1)	**Morphology and Techniques of POS Tagging**
Learning Outcomes	**After learning the module, learners will be able to**
	1. Define the morphology, the different types of morphology, and its process and differentiate between inflectional and derivational morphology, recognizing their significance in understanding word formation and structure. 2. Define and categorize word classes, recognizing their significance in linguistic analysis and natural language understanding

Content Outline	**Introduction to Morphology**: Definition of morphology, its role in NLP, and its relationship to other linguistic levels. **Morphemes and Allomorphs**: Understanding the concept of morphemes (the smallest meaningful units of language) and allomorphs (different forms of the same morpheme). **Morphological Processes:** **Inflectional Morphology**: Changes in word form to express grammatical features (e.g., tense, number) **Derivational Morphology**: Creation of new words with different meanings or grammatical categories **Compositional Morphology**: Combining words to form new words (e.g., compound nouns) **Techniques for POS Tagging:** o Rule-based POS tagging, Statistical POS tagging:-Using Hidden Markov Models (HMMs). o Maximum Entropy Markov Models (MEMMs) and Conditional Random Fields (CRFs).
Module 4 (Credit 1)	**Applications and Advanced Topics (Examples)**
Learning Outcomes	**After learning the module, learners will be able to**
	Implement the NLP knowledge
Content Outline	**Information Retrieval**: Building search engines and understanding information retrieval techniques. **Text Classification**: Classifying text into different categories (e.g., spam detection, sentiment analysis).

References:-

1. Foundations of Statistical Natural Language Processing Authors: Christopher D. Manning and Hinrich Schütze, Publisher: MIT Press
2. Natural Language Processing with Python (a.k.a. the NLTK Book), Authors: Steven Bird, Ewan Klein, and Edward Loper, Publisher: O'Reilly Media

3. Natural Language Processing: A Practical Guide for Beginners, Author: Rajesh Arumugam, Rajalingappaa Shanmugamani, Publisher: Packt Publishing
4. Practical Natural Language Processing ,Authors: Sowmya Vajjala, Bodhisattwa Majumder, Anuj Gupta, Harshit Surana ,Publisher: O'Reilly Media
5. Transformers for Natural Language Processing , Author: Denis Rothman,Publisher: Packt Publishing

Assignments/Activities towards Comprehensive Continuous Evaluation (CCE)

Classroom Activities

Module 1: Introduction to NLP

Activity 1**:** Short quiz or MCQs on NLP basics, applications, and language models.

Activity 2: Group presentation: "Applications of NLP in real-world domains (health, finance, etc.)"

Activity 3: **Crossword** or flashcards for terminology (tokens, corpus, lexicon, etc.)

Module 2: Text Preprocessing and Linguistic

Activity 1: Write a Python program using NLTK or SpaCy to tokenize, clean, and preprocess raw text

Activity 2: Annotate a paragraph using POS tagging and lemmatization using SpaCy

Activity 3: Compare outputs of stemming vs lemmatization on a given dataset

Module 3: Language Modeling and Feature Extraction

Activity 1: Implement unigram, bigram, and trigram models for a corpus and compute TF-IDF values.

Activity 2: Visualize and interpret word frequency and TF-IDF scores for a news or review dataset

Module 4 : Text Classification & Sentiment Analysis

Activity 1**:** Build a text classification model (e.g., spam vs ham, positive vs negative) using Scikit-learn

Activity 2: Compare Naive Bayes vs Logistic Regression for the same dataset.

Assessment

Internal Assessment – 50 Marks

External Assessment-50 Marks

Evaluation Scheme: - Depending on the activities mentioned above a project should be developed for 50 marks. The internal assessment, which is a project evaluation, will be done by conducting a project presentation at the College level, where an External Examiner (Industry Expert or Subject Expert) appointed by the College will be evaluating the project depending on evaluation rubrics given below.

The Rubric will have the following Evaluation Parameters:		
Evaluation Parameters	**Description / Evaluation Points**	**Marks**

Apply	Code Implementation & Execution. Writing functional, modular code for NLP tasks	10
Understand And Apply	Text Preprocessing and NLP Techniques. Using tokenization, lemmatization, POS tagging, etc. appropriately	10
Apply	Use of NLP Libraries (NLTK, SpaCy, etc.) Applying functions and tools from standard NLP libraries	10
Analyze / Evaluate	Model Development (ML/NLP Models). Selecting and building appropriate models based on problem Evaluation Metrics & Interpretation - Using accuracy, precision, recall, F1-score and interpreting results	15
	Completeness, correctness, formatting, and timely submission	5
	Total	50

(Given below are two sample projects but it is expected to work on similar sort of projects)

Project Work (Summative)

- Select a dataset (Kaggle/UCI).
- Perform text cleaning , build ML models, visualize results.
- Submit report + present findings.
- Evaluation Criteria:
- Problem statement clarity
- Code and data analysis
- Interpretation and insights
- Oral presentation & report

Course Syllabus Semester III

3.1 Major(Core)

Course Title	**Cloud Computing(web services)**
Course Credits	**4 Credits**
	CO1 : To introduce the cloud technology concepts through virtualization
	CO2: To Identify security aspects of each cloud model
	CO3: To Understand different industrial platforms of cloud
	CO4: Develop a risk-management strategy for moving to the Cloud
Module 1 :-Introduction to Cloud Computing and virtualization	
Learning Outcomes	**After learning the module, learners will be able to**
	Need of cloud computing
	Understand Eras of computing
	Define and explain the concept of cloud computing as well as virtualization
Course Outline	**Introduction to Cloud Computing**: Introduction, Historical developments, Need of Cloud Computing Environments, **Principles of Parallel and Distributed Computing**: Eras of Computing, Parallel v/s distributed computing, Elements of Parallel Computing, Elements of distributed computing, Technologies for distributed computing. **Virtualization:** Introduction, Characteristics of virtualized environments, Taxonomy of virtualization techniques, Virtualization and cloud computing, Pros and cons of virtualization, Technology examples. Virtual Server, Cloud Storage Device, Cloud usage monitor, Ready-made environment.
Module 2 Cloud Computing Architecture, Cloud models and its security	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Identify the key characteristics, service models (IaaS, PaaS, SaaS), and deployment models (public, private, hybrid) of cloud computing. 2. Fundamentals of cloud security 3. Study major cloud service providers such as Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform (GCP).
Course Outline	**Cloud Computing Architecture:** Introduction, Fundamental concepts and models, Cloud Characteristics, Cloud Delivery models, Cloud Deployment models.

	Fundamental Cloud Security: Basics, Cloud security threats, security considerations. **Industrial Platforms and New Developments:** Amazon Web Services, Google App Engine, Microsoft Azure.
Module 3:Cloud mechanisms, Cloud management and security mechanism	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Understand knowledge of cloud mechanisms 2. Cloud management mechanisms 3. Cloud security mechanisms
Course Outline	**Specialized Cloud Mechanisms:** Automated Scaling listener, Load Balancer, SLA monitor, Pay-per-use monitor, Audit monitor, fail over system, Hypervisor. **Cloud Management Mechanisms**: Remote administration system, Resource Management System, SLA Management System, Billing Management System. **Cloud Security Mechanisms**:Digital Signature, Public Key Infrastructure (PKI), Identity and Access Management (IAM), IAM Standards and Protocols for Cloud Services, IAM Practices in the Cloud, Single Sign-On (SSO), Cloud-Based Security Groups.
Module 4: Security Management in the Cloud, Privacy Issues	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Understand and apply security management standards relevant to cloud computing. 2. Differentiate availability management practices for Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS). 3. .Propose and evaluate measures for protecting privacy in the cloud, including encryption and access controls.
Course Outline	**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

Internal Evaluation =50marks

BOOKS :

1. ***Mastering Cloud Computing Foundations and Applications Programming*** by Rajkumar Buyya, Christian Vecchiola, S. Thamarai Selvi. publisher:-Elsevier , Edition:-First 2013

2. ***Cloud Computing Concepts, Technology & Architecture*** Thomas Erl, Zaigham Mahmood, and Ricardo Puttini. Publisher:-PrenticeHall , Year:-2013

Project 1:-Cloud-based E-Learning Platform
developing an online learning management system (LMS) that leverages cloud resources for hosting course content, managing user data, and enabling collaborative learning.

Project 2:-IoT Data Processing with Cloud Services
In this project, we need to combine the power of the cloud with the Internet of Things (IoT) by creating a project that involves collecting, processing, and analyzing data from IoT devices.

Case Study *(5 Marks)*

1. **Create a free account on a cloud platform (e.g., AWS Free Tier, Google Cloud, or Microsoft Azure) and:**
    - **Launch a virtual machine (VM)**
    - **Document the steps and screenshots**
    - **Mention specifications (OS, storage, pricing model)**

2. Innovation / Research Insight *(5 Marks)*

Write a short essay (200–300 words) on one emerging trend in cloud computing:

- **Edge Computing**
- **Serverless Computing**
- **Multi-cloud Strategy**
- **Green Cloud Computing**
- **AI and Cloud Integration**

Cloud Computing Evaluation Rubric – M.Sc. Level (Total: 50 Marks)

S. No.	Evaluation Component	Description	Max Marks
1	Understanding of Cloud Concepts	Assessment of core cloud principles like virtualization, service models (IaaS, PaaS, SaaS), and deployment models.	10
2	Application & Case Study Analysis	Analyze real-world use cases or case studies related to cloud platforms, scalability, or cost models.	10
3	Hands-on Implementation	Practical exercises using AWS, Azure, or GCP: setting up VMs, storage, and cloud services.	10
4	Mini Project / Innovation	Cloud-based project implementation such as hosting a website, serverless architecture, or cloud security demo.	10
5	Presentation / Viva Voce	Evaluation through oral presentation or viva on project work and concept clarity.	5
6	Continuous Internal Evaluation	Based on assignments, attendance, participation, and periodic tests or quizzes.	5

External Assessment: (Marks 50)

End Semester examination of 50 marks for 2 hours duration will be conducted.

3.2 Major (Core)

Course Title	**Machine Learning**
Course Credits	**4 Credits**
	CO1: Learners will be able to: Apply knowledge of machine learning by recognizing realworld examples and applications to understand its practical impact and potential.
	CO2: Analyse the principles of unsupervised learning to evaluate its methodologies and applications in various contexts.
	CO3: Evaluate concepts of subset selection for dimensionality reduction, understanding its importance in enhancing model performance and efficiency.
	CO4: 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 to Machine Learning**
Learning	**After learning the module, learners will be able to**

Outcomes	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.
Content Outline	History of ML Examples of Machine Learning Applications, Learning Types, ML Life cycle, AI & ML, dataset for ML, Data Pre-processing, Training versus Testing, Positive and Negative Class, Cross-validation. Machine learning Models: Geometric Models, Logical Models, Probabilistic Models. Features: Feature types, Feature Construction and Transformation, Feature Selection.
Module 2 (Credit 1)	**Learning**
Learning Outcomes	**After learning the module, 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.
Content Outline	Supervised, Unsupervised and Semi-Supervised Learning. Supervised: Learning a Class from Examples, Types of supervised Machine learning Algorithms, Unsupervised: Types of Unsupervised Learning Algorithm, Dimensionality Reduction: Introduction to Dimensionality Reduction, Subset Selection, and Introduction to Principal Component Analysis. Support Vector Machines (SVM), Soft Margin SVM, Kernel methods for nonlinearity
Module 3	**Classification and Regression**
Learning Outcomes	**After learning the module, 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.
Content Outline	Classification: Binary and Multiclass Classification: , Assessing Classification Performance, Handling more than two classes, Multiclass Classification-One vs One, One vs Rest. Regression: Assessing performance of Regression – Error measures, Overfitting and Underfitting, Catalysts for Overfitting, VC Dimensions.
Module 4 (Credit 1)	**Logic Based And Algebraic Models, Trends In Machine Learning**

Learning Outcomes	**After learning the module, learners will be able to**
	Understand the decisionmaking process based on the proximity of instances.
	Define decision trees and their role in classification and regression. • Understand the process of tree construction and decision-making.
Content Outline	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

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.

Classroom Activities

Activities towards Comprehensive Continuous Evaluation(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., Kmeans, 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.

Assessment

Internal Assessment – 50 Marks
Evaluation Scheme:

Depending on the activities mentioned above a project should be developed for 50 marks. The internal assessment, which is a project evaluation, will be done by conducting a project presentation at the College level, where an External Examiner (Industry Expert or Subject Expert) appointed by the College will be evaluating the project depending on evaluation rubrics given below.

The Rubric will have the following Evaluation Parameters:		
Evaluation Parameters	**Description / Evaluation Points**	**Marks**
Code Implementation	• Accuracy in writing and executing the ML Problems by using Python Programming correct logic, syntax	15
Problem Understanding	• Clarity of problem interpretation and correct approach	10
Output Correctness	● Program compiles without errors and gives correct output	10
Code Quality & Structure	● Proper use of indentation, comments, meaningful variable names, modularity	5
Debugging Skills	● Ability to identify and fix logical/syntax errors during execution	5
Lab Record/Journal	● Completeness, correctness, formatting, and timely submission	5

(Given below are two sample projects but it is expected to work on similar sort of projects)

Project 1:

House Price Prediction

Predict housing prices based on features like area, bedrooms, location, etc.
Dataset: Kaggle: House Prices
Linear Regression, Feature Scaling, MAE/MSE
Tools: pandas, scikit-learn, matplotlib

Project 2:

Iris Flower Classification

Classify iris flowers into 3 species based on sepal/petal size.

Dataset: sklearn.datasets.load_iris()
KNN, Logistic Regression, Decision Trees
Tools: scikit-learn, matplotlib, seaborn

External Assessment: (Marks 50)

End Semester examination of 50 marks for 2 hours duration will be conducted.

3.4 Machine Learning LAB (2 credits)

Installing Python, Jupyter Notebook, Anaconda
Introduction to libraries: NumPy, Pandas, Matplotlib, Scikit-learn, Seaborn
Git/GitHub basics (version control)

Assignment 1	Clean a raw dataset of your choice Apply transformations and output ready-to-model data Handling missing data, Encoding categorical variables, Train-test-split, cross validation
Assignment 2	Implement Linear regression on a dataset of your own choice by using Python Program
Assignment 3	Implement Logistic Regression for data classification , choose dataset of your own choice by using Python Program
Assignment 4	Implement Naïve Bayes Algorithm for data classification , choose dataset of your own choice by using Python Program
Assignment 5	Implement K-Means clustering algorithm by using Python Program
Assignment 6	Implement Decision Tree by using Python Program

Mobile Application Development Technologies

3.3 Major (Core)

Course Title	**Mobile Application Development Technologies**
Course Credits	**4 Credits**
	CO1: To provide students with a solid understanding of the mobile app development, Android operating system, its architecture, components, and the software development kit (SDK).
	CO2: To teach students how to build Android applications from scratch, including UI design, handling user interactions, and integrating various features.
	CO3: To learn about Android's UI components, layouts, and design principles to create visually appealing and user-friendly interfaces.
	CO4: To know various methods of data storage in Android applications, such as using SQLite databases, shared preferences, and cloud-based solutions.
Module 1	**Introduction Mobile Technologies & Fundamentals of Android Programming**
Learning Outcomes	**After learning the module, learners will be able to**
	Understand the evolution and scope of mobile technologies & Identify and compare various mobile operating systems. Describe Android architecture and its key components & Basic User Interface with UI Widgets
Content Outline	Introduction to Mobile Computing- Features, Advantages, Disadvantages and Applications , Factors in Developing Mobile Applications , Mobile Apps and Types of Mobile Apps , Mobile Apps Design & Development Process, Mobile Operating System: IOS, BlackBerry, Android, Windows Phone, PalmOS, SymbianOS. Introduction to Android - Overview and Evolution of Android , Features of Android, Android Architecture, Android Environment Setup Android-SDK, Emulators /Android AVD , First Android Application., Introduction to

	Components of an Android Application , Resources and Manifest File , Android App / Project Folder Structure
Module 2 (Credit 1)	**Android Activity, Intents, and Services, UI Layouts and GUI Design**
Learning Outcomes	**After learning the module, learners will be able to**
Content Outline	Android Activity and Android Activity life Cycle ,Toast in , Intents & Types, Android Services, Fragments, View, View Groups- Linear Layout, Relative Layout, Table Layout, Frame Layout, Web View, List View, Grid View ,Android UI Controls – TextView, EditText, AutoCompleteTextView, Button, ImageButton, ToggleButton, CheckBox, RadioButton, RadioGroup, ProgressBar, Spinner, TimePicker, DatePicker, SeekBar, AlertDialog, Switch, RatingBar , Event-driven Programming in Android, List and Adaptors
Module 3	**Android Menus, Threads, Notification and Alarms**
Learning Outcomes	**After learning the module, 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.
Content Outline	Creating a splash screen, Threads in Android, Threads running on UI thread (runOnUiThread), Worker thread, Handlers & Runnable, AsynTask (in detail) Android Menus - Options, Context, Popup 5.5. Android Notification- Progress and Push
Module 4 (Credit 1)	**ContentProviders, Broadcast Receivers &Advanced Programming**
Learning Outcomes	**After learning the module, learners will be able to**
	Understand the decisionmaking process based on the proximity of instances.

	Define decision trees and their role in classification and regression. • Understand the process of tree construction and decision-making.
Content Outline	Basic operation of SQLite Database, Android Application Priorities,. Android Content Providers – SQLite Programming : Open Helper and create the database, open and close a database, and insert, update, and delete operation in database , Android BroadcastReceiver, Accessing Phone Service (Call, SMS, MMS), Email, Location-based services, Storage in Android-Shared Preferences, Internal and External Storage,. Multimedia in Android – Android Camera, Audio Player. Video player , Android Bluetooth, Android WiFi, Android Sensors

References:

Reference Books
1. Professional Android Application Development by Reto Meier, Wiley India Pvt Ltd publication.
2. Android Cookbook by Ian F. Darwin O‟Reilly Media, Inc.
3. Beginning Android by Mark L. Murphy, Wiley India Pvt Ltd publication.
4. Professional Android by Sayed Y Hashimi and Satya Komatineni, Wiley India Pvt Ltd publication.
5. Building Android Apps by in easy Steps, McGraw-Hill Education publication. 6. 20 Recipes for Programming PhoneGap: Cross-Platform Mobile Development for Android and iPhone by Jamie Munro O'Reilly Media
References:
1. https://developer.android.com/guide
2. https://www.openxcell.com/mobile-app-development/
3. https://magora-systems.com/mobile-software-development-for-newbies/
4. https://www.apogaeis.com/blog/mobile-application-development-top-10-factors-to-consider/
5. https://www.ibm.com/topics/mobile-application-development
6. https://www.tutorialspoint.com/phonegap/index.htm

Classroom Activities

Activities towards Comprehensive Continuous Evaluation(CCE)

Module 1:

Assignment:

Module 2:

Activity:
Assignment:
Module 3:

Activity:
Assignment:
Module 4:

Activity:
Assignment:
Assessment

Internal Assessment – 50 Marks
Evaluation Scheme:

Depending on the activities mentioned above a project should be developed for 50 marks. The internal assessment, which is a project evaluation, will be done by conducting a project presentation at the College level, where an External Examiner (Industry Expert or Subject Expert) appointed by the College will be evaluating the project depending on evaluation rubrics given below.

The Rubric will have the following Evaluation Parameters:		
Evaluation Parameters	**Description / Evaluation Points**	**Marks**
Code Implementation	• Accuracy in writing and executing the ML Problems by using Python Programming correct logic, syntax	15
Problem Understanding	• Clarity of problem interpretation and correct approach	10
Output Correctness	● Program compiles without errors and gives correct output	10
Code Quality & Structure	● Proper use of indentation, comments, meaningful variable names, modularity	5
Debugging Skills	● Ability to identify and fix logical/syntax errors during execution	5
Lab Record/Journal	● Completeness, correctness, formatting, and timely submission	5

(Given below are two sample projects but it is expected to work on similar sort of

projects)

Project 1:

House Price Prediction

Predict housing prices based on features like area, bedrooms, location, etc.
Dataset: Kaggle: House Prices
Linear Regression, Feature Scaling, MAE/MSE
Tools: pandas, scikit-learn, matplotlib

Project 2:

Iris Flower Classification

Classify iris flowers into 3 species based on sepal/petal size.
Dataset: sklearn.datasets.load_iris()
KNN, Logistic Regression, Decision Trees
Tools: scikit-learn, matplotlib, seaborn

External Assessment: (Marks 50)

End Semester examination of 50 marks for 2 hours duration will be conducted.

3.5 Major (Elective III)

Course Title	**Internet Of Things (IoT)**
Course Credits	**4 Credits**
	CO1: Understand IoT Fundamentals: Explain the basic concepts, architecture, and components of IoT systems including sensors, actuators, communication protocols, and cloud integration.
	CO2: Understand the Impact of IoT on Society and Industry: Assess the potential impacts, challenges, and ethical considerations of IoT in different sectors such as healthcare, agriculture, smart cities, and manufacturing.
	CO3: Design and Develop IoT Solutions: Develop simple IoT applications using microcontrollers (e.g., Arduino, Raspberry Pi) and relevant sensors/actuators to solve real-world problems.

	CO4: Implement Data Collection and Processing: Collect, process, and visualize IoT sensor data using edge computing, cloud platforms, and databases.
Module 1 (Credit 1)	**Introduction to IoT & M2M**
Learning Outcomes	**After learning the module, learners will be able to**
	Understand the fundamental of IoT
	Understand the components used in IoT design
	Understand the fundamental of M2M
Content Outline	**Introduction to IoT :** Definition of IoT, Characteristics of IoT, Physical design of IoT, Logical design of IoT, Functional blocks of IoT, Communication models, Communication APIs, Communication Protocols, Challenges in IoT Design challenges **Introduction to M2M :**IoT & M2M Machine to Machine, Difference between IoT and M2M, Software define Network
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	Understand the concept of IoTWF
	Understand the various communication protocols and its use
	Gain knowledge about network & communication concepts
Content Outline	**IoT Network Architecture and Design**: The IoT World Forum (IoTWF) Standardized Architecture, A Simplified IoT Architecture, IoT protocol stack, The Core IoT Functional Stack **IoT Data Management and Compute Stack:** Fog Computing, Edge Computing, MAC protocol survey, Survey routing protocols, Sensor deployment & Node discovery, Data aggregation & dissemination
Module 3 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	Understand the concepts related to IoT Platforms
	Gain knowledge about various IoT platforms
	Understand the basics of python
Content Outline	**IoT Platforms:** IPv6 Tunneling, IPsec in IPv6, Header Compression Schemes, Raspberry Pi, Raspberry Pi Interfaces **Other IoT Devices**: pcDuino, Beagle Bone Black, CubieBoard, ARDUINO **Developing IoTs :** Introduction to Python, Introduction to different IoT tools, Developing applications through IoT tools,

Module 4 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	Identify the applications of IoT in various fields
	Elaborate case studies
Content Outline	**IoT case studies:** Domain specific applications of IoT Home automation, Industry applications, Surveillance applications, Other IoT applications

Reference Books

1. Vijay Madisetti, Arshdeep Bahga, "Internet of Things: A Hands-On Approach"
2. Smart Internet of things projects Agus Kurniawan Packt - Sep 2016 978-1- 78646- 651-8.
3. The Internet of Things Key Olivier Willy Publication 2nd Edition 978
4. Waltenegus Dargie,Christian Poellabauer, "Fundamentals of Wireless Sensor Networks: Theory and Practice"
5. IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things David Hanes, Gonzalo Salgueiro, Patrick Grossetete Cisco Press – Paperback – 16 Aug 2017 978-1- 58714-456- 1 599.

Classroom Activities

Activities towards Comprehensive Continuous Evaluation(CCE)

Module 1:

Assignment:

- Write a brief report on IoT system architecture for your selected application.
- Diagram the physical and logical design with functional blocks.

Module 2:

- Design a communication model and data flow architecture.
- Mention the protocols used (MQTT, CoAP, etc.) and justify their selection.

Module 3:

- Implement a prototype using Raspberry Pi/Arduino with Python.
- Demonstrate sensor data collection and basic processing.

Module 4:

- Present a case study related to the chosen problem.
- Discuss the societal impact and challenges.

Assessment

Internal Assessment – 50 Marks

Evaluation Scheme:

Depending on the activities mentioned above a project should be developed for 50 marks. The internal assessment, which is a project evaluation, will be done by conducting a project presentation at the College level, where an External Examiner (Industry Expert or Subject Expert) appointed by the College will be evaluating the project depending on evaluation rubrics given below.

The Rubric will have the following Evaluation Parameters:		
Evaluation Parameters	**Description / Evaluation Points**	**Marks**
Code Implementation	• Functional code in Python, Arduino, etc.	15
Problem Understanding	• Clear identification of problem and IoT relevance	10
Output Correctness	● System gives correct/expected output	10
Code Quality & Structure	● Readability, structure, comments, modularity	5
Debugging Skills	● Ability to identify and fix errors	5
Lab Record/Journal	● Neatly documented process with screenshots and results	5

(Given below are two sample projects but it is expected to work on similar sort of projects)

Project 1:

IoT-Based Health Monitoring System

Use sensors to monitor body temperature, heart rate, or oxygen level, and send data to a cloud dashboard.

Project 2:

Smart Parking System

Design a system to detect available parking spots and send data to a mobile app or dashboard.

External Assessment: (Marks 50)

End Semester examination of 50 marks for 2 hours duration will be conducted.

3.5 Major(Elective III)

Course Title	**Big Data Analytics**
Course Credits	**4 Credits**
	CO1: Understand the Big Data Platform and its Use cases
	CO2:Provide HDFS Concepts and Interfacing with HDFS
	CO3:Design a comprehensive strategy for processing and analyzing data in Hadoop, considering steps involved and potential challenges.
	CO4:Evaluate the functionality and effectiveness of Hive data warehousing and its SQL-like query language within Apache Hive's services and architecture.
Module 1 (Credit 1) INTRODUCTION TO BIG DATA AND HADOOP, HDFS (Hadoop Distributed File System)	
Learning Outcomes	**After learning the module, 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.
Content Outline	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 (Credit 1) Mapreduce	
Learning	**After learning the module, learners will be able to**

Outcomes	Apply knowledge of key components and phases in aMapReduce job to execute tasks effectively.
	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.
Content Outline	Anatomy of a Map Reduce Job Run, Failures, Job Scheduling, Shuffle and Sort, Task Execution, Map Reduce Types and Formats, Map Reduce Features.
Module 3 (Credit 1) Hadoop Eco-System	
Learning Outcomes	**After learning the module, learners will be able to**
	Apply knowledge of Pig's different execution modes,distinguishing between local and MapReduce modes to optimize data processing workflows.
	Evaluate the syntax and semantics of Pig Latin, the scripting language for Pig, to understand its structure and functionality in data processing.
	Analyse the advantages and use cases of each execution mode in Pig,evaluating their suitability for various data processing requirements.
Content Outline	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,HiveServices,HiveMetastore,ComparisonwithTraditi onalDatabases,HiveQL,Tables,QueryingDataandUserDefinedFunctions. Hbase:HBasics,Concepts,Clients,Example,HbaseVersus RDBMS. BigSQL :Introduction
Module 4 (Credit 1) Data Analytics with R Machine Learning	
Learning Outcomes	**After learning the module, 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 innovative approaches to maximize its value.
Content Outline	Introduction, Supervised Learning,Unsupervised Learning,CollaborativeFiltering. Big Data Analytics with Big R.

Reference Books

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

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 in 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.

Assessment

Internal Assessment – 50 Marks

Evaluation Scheme:

Depending on the activities mentioned above a project should be developed for 50 marks. The internal assessment, which is a project evaluation, will be done by conducting a project presentation at the College level, where an External Examiner (Industry Expert or Subject Expert) appointed by the College will be evaluating the project depending on evaluation rubrics given below.

The Rubric will have the following Evaluation Parameters:		
Evaluation Parameters	**Description / Evaluation Points**	**Marks**
Understanding of Concepts	Demonstrates deep understanding of Big Data fundamentals, frameworks (Hadoop, Spark), and analytics concepts. Clear and precise explanations.	10

Application & Problem Solving	Effectively applies concepts to real-world Big Data problems, using appropriate tools and techniques.	10
Data Handling & Tools Usage	Proficient use of Big Data tools (Hadoop, Spark, NoSQL DBs) and demonstrates advanced data handling skills.	10
Analytical & Critical Thinking	Insightful analysis of data results; identifies trends, anomalies, and draws meaningful conclusions.	10
Communication & Presentation	Clear, concise report/presentation; well-structured, free of errors; visuals and explanations enhance understanding.	10

3.5 Major(Elective III)

Course Title	**GIS and Remote Sensing**
Course Credits	**4 Credits**
	CO1:Understanding Fundamentals, Technical Skills, Data Acquisition and Management.
	CO2:Spatial Analysis, Mapping and Visualization, Remote Sensing Applications, Integration of Technologies.
	CO3:**Problem Solving, Communication Skills**.
Module 1 :-Fundamentals of GIS	
Learning Outcomes	**After learning the module, learners will be able to**
	1.Define GIS (Geographic Information System), Understand Components of GIS, Comprehend Spatial Data, Understand Characteristics of Spatial Data.
	2.Analyze Spatial Data Maps, Attribute Data Management - Database Data Model, GIS Applications, Developments in the Database for GIS.
Course Outline	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	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Data Input and Editing, Data Analysis 2. Analytical Modeling in GIS, Output from GIS.
Course Outline	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:	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Development of Computer Methods, Human and Organizational Issues in GIS, GIS Data Quality and Error Analysis, GIS Project Design and Management. 2. Project Implementation and Evaluation, Understanding the Future of GIS, Internet

	Resources for GIS, Communication Skills
Course Outline	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)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Principles of Remote Sensing, Remote Sensing System Classification, Imaging Characteristics, Extraction of Information from Images, Integration of Remote Sensing and GIS. 2. Introduction to GPS, Accuracy of GPS, Differential GPS, Applications of GPS, Integration of GIS and GPS.
Course Outline	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 1. These assignments aim to apply theoretical concepts to practical application and critical thinking. 2. Discuss the importance of metadata in data warehousing, to create a metadata management plan for a given data warehouse, outlining how metadata will be collected, stored, and utilized. 3. From a dataset, extract relevant information, transform it according to a predefined business rule, and load it into a data warehouse. 4. Create a set of business queries related to a hypothetical business problem Find a dataset suitable for clustering analysis. then use clustering algorithms to identify natural groupings within the data and interpret the results. 5. web mining project (perform web scraping, and apply web mining techniques)	

Internal Assessment=50marks

Books:

1. Heywood, I., Cornelius, S., & Carver, S. (2000). An Introduction to Geographical Information Systems. Pearson Education Asia.

2. Lo, C. P., & Yeung, A. (n.d.). Concepts and Techniques of Geographic Information Systems. PHI, New Delhi.

3. Demers, M. N. (n.d.). Fundamentals of Geographic Information Systems, 2nd Edition. John Wiley & Sons (Asia) Pte Ltd.

Rubric for Evaluation: GIS and Remote Sensing (M.Sc. Level – 50 Marks)

Assessment Component	Learning Outcome Addressed	Marks	Evaluation Criteria
Theory Assignment / Written Test	**Understanding core concepts of GIS & Remote Sensing**	**10**	**Conceptual clarity, completeness, use of diagrams/maps, and accuracy**
Practical Lab Work	**Ability to use GIS/RS software like QGIS, ERDAS, or ArcGIS; data analysis skills**	**10**	**Accuracy of map creation, correct data processing, application of techniques**
Field Survey / Data Collection Task	**Real-world application, data collection using GPS or remote sensing sources**	**5**	**Use of appropriate tools, data accuracy, documentation of observations**
Mini Project / Case Study	**Problem-solving using GIS/RS techniques in real-world scenarios**	**10**	**Innovation, data usage, spatial analysis, conclusions, and presentation**
Presentation / Viva Voce	**Communication and interpretation of GIS/RS findings**	**5**	**Clarity, confidence, use of terminology, and understanding of concepts**
Research Paper / Report Review	**Exposure to current trends and academic literature**	**5**	**Depth of review, critical analysis, and synthesis of information**

Peer Learning / Group Task	**Collaborative and experiential learning**	**5**	**Active contribution, team coordination, and outcomes achieved in the group**

Course Syllabus Semester IV

4.1 Major (Core)

Course Title	**Interactive & Advanced Data Visualization (with Python)**
Course Credits	**4 Credits**
	CO1: Understand Visualization Foundations
	CO2: Design and Construct Visual Solutions
	CO3: Develop Interactive Dashboards and Applications
	CO4: Implement Visualization with Clean Code and Usability
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	Understand the architecture and components of a plot.
	Create a wide range of static and custom plots using Matplotlib.
	Integrate Matplotlib with Pandas/NumPy for analytical storytelling.
Content Outline	Anatomy of plots: Figure, Axes, Axis, Ticks, Labels Plot types: Line, Scatter, Bar, Pie, Histogram, Boxplot Styling: Colors, Fonts, Gridlines, Colormaps, Themes (rcParams) Subplots & Layouts: subplot(), subplots(), tight_layout() Plotting with Pandas: Time series, grouped summaries Annotations, legends, saving in multiple formats (PNG, PDF, SVG).
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	Generate informative and visually rich statistical plots.
	Use Seaborn's high-level functions for rapid exploration.
	Apply theming, faceting, and palette customization for clarity.
Content Outline	Univariate plots: histplot, displot, kdeplot Bivariate plots: scatterplot, regplot, lineplot Categorical plots: barplot, boxplot, violinplot, swarmplot Grouping data using hue, col, row Multi-panel plots: FacetGrid, catplot, Pairplot Styling: Themes (whitegrid, darkgrid), context (poster, talk) Heatmaps for correlation and pivot analysis
Module 3 (Credit 1)	

Learning Outcomes	**After learning the module, learners will be able to**
	Create interactive charts with tooltips, sliders, filters, and callbacks
	Use Plotly Express and Graph Objects for rich chart customization.
	Build lightweight dashboards and export them to HTML.
Content Outline	Plotly Express: px.line, px.bar, px.scatter, px.pie Customizing layout: Legends, axes, background, annotations Interactivity: Hover templates, sliders, dropdown menus Graph Objects: Layering, subplots, animations Exporting: HTML, JSON, static images Plot types: Heatmaps, sunbursts, indicators, bubble charts Comparison with Matplotlib and Seaborn use-cases
Module 4 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	Create real-time interactive plots using ipywidgets and bqplot.
	Link widget controls to data changes and UI elements.
	Build Jupyter-native dashboards with full interactivity.
Content Outline	ipywidgets: Dropdown, Slider, Checkbox, Text, interactive() Widget layout: VBox, HBox, GridBox, styling with Layout Event handling: .observe(), .on_click() bqplot architecture: Marks, Scales, Axes, Figures Visualization types: Line, Bar, HeatMap, Scatter, Pie Linking visuals with widgets dynamically Dashboard composition and deployment inside Jupyter

Reference Books:

- Rahman, K. (2021). Python Data Visualization Essentials Guide. BPB Publications. Covers Matplotlib, Seaborn, Plotly, Pandas, and Bokeh with 50+ examples and case studies.
- Plotly Team. (2023). Plotly Dash User Guide. Plotly Official Documentation. Comprehensive guide to building dashboards using Plotly and Dash.
- Jupyter Team. (2023). Interactive Widgets for Jupyter Notebooks. Jupyter Documentation.Covers ipywidgets, layout design, and event handling for interactivity.
- Ashford, D. (2024). Python for Data Visualization: Explore Matplotlib, Seaborn, Plotly, and Bokeh. Amazon Kindle.Hands-on guide with real-world projects and interactive dashboard design.
- Milovanovic, I. (2015). Python Data Visualization Cookbook. Packt Publishing. Practical recipes for Matplotlib and Seaborn with customization techniques.

Classroom Activities

Activities towards Comprehensive Continuous Evaluation(CCE)

Module 1 – Static Plotting Foundations with Matplotlib

Assignment 1: Create line plots and bar charts using student performance data. Add axis labels, titles, legends, and grid styling.

Assignment 2: Use multiple subplots to compare weather metrics (temperature, rainfall, humidity) across seasons.

Assignment 3: Plot multiple mathematical functions (e.g. sine, cosine, tangent) and style each uniquely.

Module 2 – Statistical & Categorical Plotting with Seaborn

Assignment 1: Use `histplot`, `kdeplot`, and `boxplot` to explore internal assessment scores.

Assignment 2: Create `FacetGrid` or `plot` to examine trends across gender, course stream, and year.

Assignment 3: Compare count of social media users by region using `countplot`, adding custom palettes and themes.

Module 3 – Interactive Charts with Plotly

Assignment 1: Build an interactive line chart showing attendance trends with hover tooltips and custom annotations.

Assignment 2: Create an animated time-series chart to track COVID-19 cases across months using slider controls.

Assignment 3: Design a responsive dashboard using subplots with `Plotly.graph_objects`—include dropdown filters for department performance.

Assignment 4 : Embed a bar chart inside an HTML export and personalize the styling using layout attributes.

Module 4 – Reactive Visualizations with ipywidgets & bqplot

Assignment 1: Design a real-time KPI dashboard using ipywidgets sliders, dropdowns, and checkboxes.

Assignment 2: Use `bqplot` to visualize student scores with reactive feedback—change data source using widgets.

Assignment 3: Create a mini dashboard inside Jupyter Notebook with scatter, bar, and pie charts linked to widgets.

Assessment
Internal Assessment – 50 Marks
Evaluation Scheme:

Depending on the activities mentioned above a project should be developed for 50 marks. The internal assessment, which is a project evaluation, will be done by conducting a project presentation at the College level, where an External Examiner (Industry Expert or Subject Expert) appointed by the College will be evaluating the project depending on evaluation rubrics given below.

The Rubric will have the following Evaluation Parameters:		
Evaluation Parameters	**Description / Evaluation Points**	**Marks**
Visualization Implementation	Accurate use of the plotting library (Matplotlib / Seaborn / Plotly / bqplot) and its core functions. Correct syntax, logical structure, and output quality.	10
Problem Understanding & Analysis	Clarity of dataset selection, contextual relevance, and correct interpretation. Justification for plot type and design choices	10
Interactivity & User Control	Use of dynamic features — sliders, dropdowns, tooltips, filters, widget linking, callbacks — with effective user experience design. (Applicable in Plotly, ipywidgets, bqplot)	10
Design & Styling Quality	Aesthetics of visual output: color palette, axis labels, titles, legends, annotations, layout structure. Use of themes and customization for clarity and readability.	10
Code Quality & Modularity	Proper use of indentation, meaningful variable names, reusable code blocks/functions, adherence to best practices in notebook/code organization.	5
Lab Record / Documentation	Completeness of submission, clarity of insights and markdown annotations, formatting, and timely submission of lab notebooks or project reports.	5

(Given below are two sample projects but it is expected to work on similar sort of projects)

Project 1:

Using Python and Matplotlib, create a system that visualizes internal assessment scores of students across subjects using bar charts, histograms, and pie charts. The system should allow selection of student ID and subject to compare performance trends and statistical summaries.

Project 2:

Using Python, ipywidgets, and bqplot, create a student feedback analysis tool where users can adjust satisfaction scores with sliders and view real-time updates in visual form (bar, pie, and scatter charts). Integrate widgets to filter feedback based on department or course type.

External Assessment: (Marks 50)

End Semester examination of 50 marks for 2 hours duration will be conducted.

4.3 Major (Core)

Course Title	**Deep Learning**
Course Credits	**4 Credits**
	CO1 : Demonstrate the ability to implement a perceptron, understanding the input features, weights, bias, and the activation function.
	CO2: Understanding Deep Learning, Activation Functions, Forward Propagation.
	CO3: Explore convolution operations, including padding, stride, and batch processing. Implement a convolution layer and a pooling layer in TensorFlow.
Module 1 :-Introduction to Deep Learning	
Learning Outcomes	**After learning the module, learners will be able to**
	1.Understanding how to build a perceptron involves defining the input features, weights, bias, and the activation function.
	2.Understand backpropagation algorithm
	3.Description of the structure of artificial neural networks, including input layer, hidden layers, and output layer.
	4.Overview of activation functions that introduce non-linearity, enabling neural networks to learn complex patterns.
Course Outline	**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 **The Back-Propagation Algorithm**: Heuristics for Making the Back- Propagation Algorithm Perform Better, Back Propagation and Differentiation, Neural Network Training
Module 2: Convolutional Neural Network and Tensorflow	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Understand convolution layer and its working 2. Implementing a Convolution Layer, Pooling Layer 3. Understand Tensorflow
Course Outline	**Convolutional Neural Networks:** Overall Architecture, The 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. **What is TensorFlow?** Understanding computational graphs and sessions, Sessions, Variables, constants, and placeholders, Introducing TensorBoard, Creating a name scope.
Module 3:Optimizers in DL	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Understanding the basic concept of gradient descent as an optimization algorithm for minimizing the loss function during training. 2. Introduction to adaptive learning rates based on the historical gradients of parameters.
Course Outline	**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).
Module 4: RNN and Unsupervised Deep Learning	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Understanding the challenges of training RNNs and the need for handling sequential dependencies. 2. Explanation of backpropagation through time, the

	algorithm used to train RNNs by unfolding them into a computational graph over time. Different type of RNNarchitectures 3.Understanding autoencoders as neural network architectures designed for unsupervised learning by encoding and decoding input data.
Course Outline	**Introducing RNN**:-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. **Deep Unsupervised Learning:** Auto encoders (standard, sparse, denoising, contractive, etc), Variational Auto encoders

Internal Assessment=50marks

BOOKS :

1. Goodfellow, Ian, Bengio, Yoshua, Courville, Aaron. Deep Learning (Adaptive Computation and Machine Learning series). The MIT Press, 2016.

2. Buduma, Nikhil, Locascio, Nicholas. Fundamentals of Deep Learning: Designing Next-Generation Machine Intelligence Algorithms. O'Reilly Media, 2017.

Project 1:Simple Chatbot:
Build a chatbot that can respond to basic user queries using techniques like rule-based systems or simple neural networks.

Project 2:Color Detection System:
Develop a system that can identify and classify colors in images or video.

Mini Project *(5 Marks)*

1. Study an open-source deep learning project (e.g., YOLO, BERT, or StyleGAN) and provide:

- **A summary of its purpose and how it works**
- **A real-world application of the model**

2. Research/Innovation Component *(5 Marks)*

Write a short essay (200–300 words) on one emerging or interdisciplinary topic in deep learning:

- **Deep Learning in Healthcare / Finance / Agriculture**
- **Transformers and their role in NLP**
- **Explainable AI (XAI)**
- **Ethical issues in Deep Learning**
- **Generative AI (GANs, Diffusion Models)**

Deep Learning Evaluation Rubric (M.Sc. Level – 50 Marks)

S. No.	Evaluation Component	Description	Max Marks
1	Conceptual Understanding	Assessment of core concepts such as neural networks, activation functions, loss functions, optimizers, etc.	10
2	Application & Problem Solving	Applying DL models (e.g., CNNs, RNNs, LSTMs) to real-world problems, case-based questions	10
3	Programming & Implementation Skills	Practical implementation using tools like TensorFlow, PyTorch; model building, training, testing	10
4	Innovation / Mini Project	A DL-based project (e.g., image classification, NLP task, time series) evaluated on novelty and relevance	10

5	Presentation / Viva Voce	Oral defense and demonstration of project or concept clarity through a seminar or viva	5
6	Continuous Internal Evaluation	Assignments, attendance, quizzes, and classroom participation	5

External Assessment: (Marks 50)

End Semester examination of 50 marks for 2 hours duration will be conducted.

4.4 Major (Core)

Course Title	**Digital Forensics Security**
Course Credits	**4 Credits**
	CO1 : Understand the Fundamentals of Digital Forensics, Understand the concept of legal considerations.
	CO2: Analyze Digital Evidence and Investigative Techniques
	CO3:Investigate Cybercrimes and Incident Response
	CO4: Apply Forensic Tools and Report Findings
Module 1 (Credit 1) Introduction to Digital Forensic	
Learning Outcomes	**After learning the module, learners will be able to**
	1.Understand the fundamental concepts of Digital forensics.
	2.Understand the fundamental concepts of cyber forensics.
Course Outline	**Introduction to Digital Forensic:** Introduction to cyber crimes & Digital Forensic, Types of Digital Forensics, Digital Forensics Process, Areas of Application of computer forensics, Understanding the Suspects, Examples of Computer Forensics, Free space and Slack Space. **Introduction to Cyber Forensics**

	Definition, Scope, Importance of Cyber Forensics, Digital Crime Investigation Process, Cyber Laws and Ethics in Forensics
Module 2 (Credit 1): Types of Digital Evidence & Crime Scenes	
Learning Outcomes	**After learning the module, learners will be able to**
	1.Identify different types of digital evidence and legal considerations.
	2.Learn forensic methodologies and the role of forensic experts in investigations.
Course Outline	**Types of Digital Evidence & Crime Scenes** Sources of Digital Evidence (Computers, Mobile Devices,Networks) ,Volatile vs. Non-Volatile Evidence, Evidence Collection and Chain of Custody. **Legal and Ethical Considerations in Cyber Forensics** Admissibility of Digital Evidence in Court,Role of Law Enforcement Agencies,Ethical Issues in Cyber ForensicsDevice Forensics (Android, iOS),Why do we need mobile forensics?, Challenges in mobile forensics.
Module 3 (Credit 1)Forensic Tools and Techniques	
Learning Outcomes	**After learning the module, learners will be able to**
	Gain knowledge of forensic tools used in digital investigations.
	Analyze forensic methodologies for network and mobile forensics.
Course Outline	**Forensic Tools and Techniques:** Popular Forensic Tools: Autopsy, EnCase, FTK, Wireshark File System Forensics (FAT, NTFS) Recovering Deleted Files & Metadata Analysis **Email,Social Media,Network & Mobile Forensics:** Network Traffic Capture & Analysis (Wireshark, TCPDump) Email & Social Media Forensics,Exploring the role of email investigation,Applying Digital Forensics Methods to Social Media Communications, Social Media Forensics on Mobile Devices Forensics Tools for Social Media InvestigationsMobile Device Forensics (Android, iOS),Why do we need mobile forensics?, Challenges in mobile forensics.
Module 4 (Credit 1) Cybercrime Investigation & Case Studies	
Learning Outcomes	**After learning the module, learners will be able to**
	Apply forensic techniques to real-world cybercrime cases.

	Understand the concept of mobile & network forensic.
Course Outline	**Cybercrime Investigation & Case Studies** Investigating Hacking, Identity Theft, Phishing Attacks Financial & Banking Frauds (Ransomware, Cryptocurrency Fraud) Case Studies on Real-World Cybercrimes **Emerging Trends in Cyber Forensics** Cloud & IoT Forensics,AI & Machine Learning in Digital Forensics,Challenges in Dark Web & Deep Web Investigations.

Internal Assessment – 50 Marks

Evaluation Scheme:
Depending on the activities mentioned above a project should be developed for 50 marks. The internal assessment, which is a project evaluation, will be done by conducting a project presentation at the College level, where an External Examiner (Industry Expert or Subject Expert) appointed by the College will be evaluating the project depending on evaluation rubrics given below.

The Rubric will have the following Evaluation Parameters:		
Evaluation Parameters	**Description / Evaluation Points**	**Marks**
Understanding of Digital Forensics Fundamentals	Demonstrates thorough understanding of concepts, types, and application areas of digital forensics.	10
Identification and Handling of Digital Evidence	Accurately identifies types of digital evidence, explains chain of custody and legal implications clearly.	10
Cybercrime Investigation Skills	Effectively applies forensic procedures to investigate complex cybercrimes with logical analysis.	10
Use of Forensic Tools and Techniques	Proficient in using forensic tools (Autopsy, FTK, EnCase, Wireshark), and accurately analyzes output.	10
Report Writing and Case Presentation	Well-organized, clear, technical report with logical conclusions and citations.	10

Given below are two sample case study, but it is expected to work on similar sort of Case study

Case study 1:-Financial Fraud via Phishing – Investigation and Evidence Recovery Identify potential digital evidence in this case,Outline the steps for evidence collection and preservation,Suggest forensic tools to analyze email, browser activity, and file system,Draft

a mini-report with findings, IP traceback, and potential prosecution points.

Case study 2:-Social Media Harassment – A Mobile Forensics Investigation -List the digital evidence sources from mobile and cloud.Choose forensic tools to recover WhatsApp and social media data.Explain how to preserve evidence admissible in court.Suggest ethical steps to protect victim privacy during investigation.

External Assessment: (Marks 50)
End Semester examination of 50 marks for 2 hours duration will be conducted by University