SNDT Women's University, Mumbai

Undergraduate Degree / UG Programme (Syllabus as Per NEP) - Faculty of Science & Technology

(AEDP)

Bachelor of Science (Information Technology)

B.Sc. (IT)

As Per NEP – 2020

Semester – I to VI

Syllabus
(W.E.F. Academic Year 2025-26)

Terminologies

Vertical	Full-Form/Definition	Remarks	Related To Major and Minor Courses
Major (Core)	Subject Comprising Mandatory and Elective Courses, Major Specific IKS, Vocational Skill Courses, Internship/ Apprenticeship, Field Projects, Research Projects Connected to Major	Minimum 50% Of Total Credits Corresponding to Three/Four - Year UG Degree- Mandatory Courses	Related To the Major
Minor Course	Course From Same or Different Faculty	Minimum 18-20 Credits to Be Completed in The First Three Years of UG Programme	Related To the Minor
OEC	Open Elective Courses/ Generic Courses	10-12 Credits to Be Offered in I And/Or II Year. Faculty-Wise Baskets of OEC To Be Prepared	OEC Is to Be Chosen Compulsorily from Faculty Other Than That of the Major
VSC	Vocational Skill Courses, Including Hands on Training Corresponding to The Major And/Or Minor Subject	8-10 Credits, To Be Offered in First Three Years, Wherever Applicable Vocational Courses Will Include Skills Based on Advanced Laboratory Practical's of Major	Related To the Major or Minor
SEC	Skill Enhancement Courses	06 Credits, To Be Offered in I And II Year, To Be Selected from The Basket of Skill Courses Approved by University	Related To the Major or Minor Any Relevant Skill
AEC	Ability Enhancement Courses	08 Credits, To Be Offered in I And II Year, English: 04 Credits to Be Earned in Sem - I, Modern Indian Language Of 04 Credits to Be Offered in II Year	NA
VEC	Value Education Courses	Understanding India, Environmental Science/Education, Digital	NA

		and Technological Solutions, Health & Wellness, Yoga Education, Sports, And Fitness	
IKS	Indian Knowledge System	Generic IKS Course: Basic Knowledge of The IKS To Be Offered at First Year Level	Major-Specific IKS Courses: Advanced Information About the Major, Part of the Major Credit to Be Offered at Second- Or Third-Year Level
OJT	On-Job Training (Internship / Apprenticeship)	Corresponding To the Major Subject	Related To the Major
FP	Field Projects	Corresponding To the Major Subject	Related To the Major
CC	Co-Curricular Courses	Health And Wellness, Yoga Education Sports, And Fitness, Cultural Activities, NSS/NCC And Fine/ Applied/Visual/ Performing Arts	NA
CE	Community Engagement and Service		Related To Major
RP	Research Project	Corresponding To the Major Subject	Related To Major

Programme Template

Faculty		**Science and Technology**
Programme		**B.Sc.**
Specialization		**Information Technology (IT)**
Programme Outcomes (POs)		**After completing this programme, Learner will**
Major	1.	Describes a strong foundation in computer application, including knowledge of Programming languages, Database, Mathematics, Operating system, Cyber Security and Networking.
	2.	Analyze the ethical and professional responsibilities related knowledge acquisition, mastery over the implications of adhering to professional standards and practices.
	3.	Applying programming knowledge to develop a software application to solve specific problems.
	4.	Evaluate software designs and architectures for efficiency, security and user experience.
	5.	Design a software application to meet the requirements of the Industrial Standards.
Minor	6.	To Understand AI/ML working and its application.
	7.	To apply the AI/ML in real time use case.
Eligibility Criteria for Programme		**B. Sc.IT – I Year:** A women candidate must have passed the higher secondary school certificate (std. XII) examination conducted by the different divisional boards of the Maharashtra state board of secondary and higher secondary education in any stream with 45% marks (40% for candidates belonging to reserved category). Or must have passed an examination of any other recognized board or body recognized as equivalent. Students who have not done mathematics at 12th std. Are needed to take a bridge course in mathematics and pass in university conducted test before semester 1 examination. Or must have passed any three-year government recognized diploma programme in second class. **B.Sc. (IT) - II Year:** For a student from our university - should have cleared or has ATKT as admissible in first year in the same subject. For a student from another university- studied at least two courses of four credits each in the I year in the same subject. Passed I year without ATKT.

		Passed three-year full-time Diploma in Engineering with an aggregate of not less than 60% (Open Category) and 55% (Reserved Category) OR must have passed any three-year Diploma in Engineering Course. **B.Sc. IT - III Year:** For a student from our university- should have cleared second year in the same subject or has passed with admissible ATKT. For a student from another university– should have completed at least six courses of 4 credits each or 24 credits in the I and II year in the same subject. Passed in I and II year without ATKT.
Intake **(For SNDTWU Departments and Conducted Colleges)**		As Per University Affiliation Letter

Structure with Course Titles

AEDP - B. Sc. In Information Technology

Semester - I (Starting from 2026-27)

SN	Courses	Type of Course	Credits	Marks	Int	Ext
	Semester I					
10135213	Problem solving with C Programming (Th+Pr) (2+2)	Major (Core)	4	100	50	50
10135204	Introduction to Artificial Intelligence (Th)	Major (Core) **(Related to Minor)**	2	50	50	0
10435211	Introduction to Web Technology (Th)	OEC **(Any One)**	4	100	50	50
10435212	Fundamentals of Statistics using Spreadsheets (Th)					
10435213	Introduction to Cyber Security (Th)					
10635202	Introduction to operating system (Th)	VSC S1 **(Related to Major)**	2	50	50	0
10735221	Web Technology – I (Pr)	SEC **(Any One)**	2	50	0	50
10735222	UI / UXI design (Pr)					
	Ability Enhancement Course (AEC) Link: **https://sndt.ac.in/pdf/academics/syllabus-as-per-nep/aec-syllabus/ug-degree/ability-enhancement-course.pdf** (Available on Website)	AEC (English) **(Any One)**	2	50	0	50

10810111	English For Academic Writing-Paper I (For Students of English Medium)					
10810112	English Language and Literature-I (For Students of Non-English medium)					
11051111	Inception of India Knowledge System **IKS Link:** **https://www.sndt.ac.in/pdf/academics/syllabus-as-per-nep/iks-syllabus/ug-degree/inception-of-indian-knowledge-system.pdf** (Available on Website)	IKS (Generic)	2	50	0	50
10952111	Introduction to Indian Constitution **Link:** **https://www.sndt.ac.in/pdf/academics/syllabus-as-per-nep/vec-syllabus/ug-degree/introduction-to-indian-constitution.pdf** (Available on Website)	VEC	2	50	0	50
	Co-Curricular Course (CC) Link: **https://www.sndt.ac.in/pdf/academics/syllabus-as-per-nep/cc-syllabus/ug-degree/co-curricular-course-as-per-nep-2020-semester-i-syllabus.pdf** (Available on Website)	CC **(Any One)**	2	50	50	0
11450121	Basics of National Service Scheme					

11450221	National Cadets Corps. (NCC) Studies - I					
11450322	Health and Wellness					
11450421	Performing Arts Exploration					
			22	**550**	**250**	**300**

Semester – II (for NON-AEDP for the batch 2025-26)

SN	Courses	Type of Course	Credits	Marks	Int	Ext
	Semester II					
20135211	Data Structure & Algorithms (Th+Pr) (2+2)	Major (Core)	4	100	50	50
20135212	Computer Organization and Architecture (Th)	Major (Core)	2	50	0	50
20635201	Object Oriented Programming using C++ (Pr)	VSC S2 **(Related to Major)**	2	50	50	0
20635202	Introduction to Data Science (Pr)	VSC S3 **(Related to Minor)**	2	50	50	0
20435211	Multimedia System (Th)	OEC **(Any One)**	4	100	50	50
20435212	Introduction to Artificial Intelligence (Th)					
20435213	Cyber forensic (Th)					
20735202	Mathematics – I (Th)	SEC	2	50	50	0
	Ability Enhancement Course (AEC) Link: **https://www.sndt.ac.in/pdf/academics/syllabus-as-per-nep/aec-syllabus/ug-degree/ability-enhancement-course.pdf** (Available on Website)	AEC (English) **(Any One)**	2	50	0	50

20810111	English For Academic Writing - Paper II (For Students of English Medium)					
20810112	English Language and Literature - II (For Students of Non-English medium)					
20952111	Environment Awareness **Link:** **https://www.sndt.ac.in/pdf/academics/syllabus-as-per-nep/vec-syllabus/ug-degree/environment-awareness.pdf** (Available on Website)	VEC	2	50	0	50
	Co-Curricular Course (CC) Link: **https://www.sndt.ac.in/pdf/academics/syllabus-as-per-nep/cc-syllabus/ug-degree/co-curricular-course-as-per-nep-2020-semester-ii-syllabus.pdf** (Available on Website)	CC **(Any One)**	2	50	50	0
21450121	Volunteerism and National Service Scheme					
21450221	National Cadets Corps. (NCC) Studies - II					
21450323	Yoga Education					
21450421	Fine Art					
			22	**550**	**300**	**250**

Exit with UG Certificate with 4 extra credits (44 + 4 credits)

Semester – III (for Semester 3 of 2026-27 batch)

SN	Courses	Type of Course	Credits	Marks	Int	Ext
30135211	Java Programming (Th+Pr) (2+2)	Major (Core)	4	100	50	50
30135212	Database Management System (Th+Pr) (2+2)	Major (Core)	4	100	50	50
30135214	Cyber security (Th)	Major (Core)	2	50	0	50
30335211	Mathematics – II (Th)	Minor Stream	4	100	50	50
30435211	Intellectual Property Rights and Cyber Law (Th)	OEC **(Any One)**	2	50	0	50
30435212	Digital Marketing (Th)					
30435213	E-Commerce (Th)					
	Modern Indian Language **Ability Enhancement Course (AEC) Link:** **https://sndt.ac.in/pdf/academics/syllabus-as-per-nep/aec-syllabus/ug-degree/aec-semester-iii.pdf** (Available on Website)	AEC **(Any One)**	2	50	50	0
30810201	**શીખો ગુજરાતી – પ્રાથમિક ભાગ ૧: પરિચય, લેખન અને વાચન કૌશલ્ય (ગુજરાતી)** Shikho Gujarati – Prathmik Bhag 1: Parichay, Lekhan ane Vachan Kaushalya (Gujarati)					
30810301	रचनात्मक लेखन (हिन्दी) Rachnatmak Lekhan (Hindi)					
30810401	मराठी भाषेचा परिचय भाग १ - (मराठी) Marathi Bhashecha Parichay- Bhaag 1 (Marathi)					

30810501	लौकिक संस्कृत न्यायः (संस्कृत) Contemporary Sanskrit Nyaya (Sanskrit)					
30135205	Linux Operating System (Pr)	Major (Core)	2	50	50	0
	Co-Curricular Course (CC) Link: **https://sndt.ac.in/pdf/academics/syllabus-as-per-nep/cc-syllabus/ug-degree/co-curricular-course-as-per-nep-2020-semester-iii-syllabus.pdf** (Available on Website)	CC **(Any One)**	2	50	50	0
31450121	Social issues Advocacy and Action					
31450221	National Cadets Corps. (NCC) Studies – III					
31450321	Traditional Sports and Fitness					
31450421	Unfolding The Beauty of Indian Music					
			22	**550**	**300**	**250**

Semester – IV (current 4th semester for 2024-25 batch onwards)

SN	Courses	Type of Course	Credits	Marks	Int	Ext
40135211	Advanced Java (Th+Pr) (2+2)	Major (Core)	4	100	50	50
40135212	Web Technology (Th+Pr) (2+2)	Major (Core)	4	100	50	50
40335211	Machine learning (Th+Pr) (2+2)	Minor Stream	4	100	50	50
40435211	Introduction to Canva (Th)	OEC **(Any One)**	2	50	0	50
40435212	Management Information System (Th)					
40435213	Introduction to Entrepreneurship (Th)					
40735211	Introduction to Microprocessor 8086 (Th)	SEC	2	50	0	50
	Modern Indian Language **Ability Enhancement Course (AEC) Link:** **https://sndt.ac.in/pdf/academics/syllabus-as-per-nep/aec-syllabus/ug-degree/aec-semester-iv.pdf** (Available on Website)	AEC **(Any One)**	2	50	0	50
40810211	**શીખો ગુજરાતી – પ્રાથમિક** (ગુજરાતી) Shikho Gujarati – Prathmik (Gujarati)					
40810311	सूचना प्रौद्योगिकी और हिन्दी भाषा (हिन्दी) Suchana Praudyogiki aur Hindi Bhasha (Hindi)					
40810411	मराठी भाषेचा परिचय भाग -२ (मराठी)					

	Marathi Bhashecha Parichay- Bhaag 2 (Marathi)					
40810511	वाल्मीकीयकृत रामायणम् – अयोध्याकाण्डः (संस्कृत) Valmikiyakrut Ramayanam – Ayodhyakanda (Sanskrit)					
40335202	Python for Machine Learning (Pr)	Minor Stream	2	50	50	0
	Co-Curricular Course (CC) Link: https://sndt.ac.in/pdf/academics/syllabus-as-per-nep/cc-syllabus/ug-degree/co-curricular-course-as-per-nep-2020-semester-iv-syllabus.pdf (Available on Website)	CC **(Any One)**	2	50	50	0
41450121	NSS Volunteers under National service scheme special camp					
41450122	Personality and Leadership Development through National Service Scheme					
41450221	National Cadets Corps. (NCC) Studies – IV					
41450321	Basics of Self Defence and Fitness Training					
41450421	Theatre & Dance					
			22	**550**	**250**	**300**

Exit with UG Diploma with 4 extra credits (44 + 4 credits)

Semester – V (2024-25 batch and onwards) - AY 2026-27

SN	Courses	Type of Course	Credits	Marks	Int	Ext
50135211	Computer Networks (Th+Pr) (2+2)	Major (Core)	4	100	50	50
50135212	Software Engineering (Th)	Major (Core)	4	100	50	50
51035211	Indian Knowledge Systems and Ethical Practices in IT (Th)	IKS (Major Specific)	2	50	0	50
50335211	Deep learning (Th+Pr) (2+2)	Minor Stream	4	100	50	50
50335212	AI / ML Use Case (Th+Pr) (2+2)	Minor Stream	4	100	50	50
50635201	Mobile Application Development using Android (Pr)	VSC-4	2	50	50	0
50135203	Full Stack Development (Pr)	Major (Core)	2	50	50	0
			22	**550**	**300**	**250**

Semester – VI

SN	Courses	Type of Course	Credits	Marks	Int	Ext
61235221	Internship (AEDP-Apprenticeship Embedded Degree Programmes) (Pr)	Major (Core)	20	500	250	250
61535221	Digital Literacy and Community Outreach (Pr)	CE	2	50	0	50
			22	**550**	**250**	**300**

Exit with Degree (3-year)

Couser Syllabus

Semester - I

.1.1 Major (Core)

Course Title	**Problem solving with C Programming**
Course Credits	**4 Credits**
Course Outcomes	1. Apply logic to create programs in C.
	2. Analyze and understand computer programming language concepts.
	3. Evaluate and interpret the use of pointers, including their declarations, initialization, and operations.
	4. Translate an algorithm into a C computer program
	5. Design and develop applications using basic programming constructs, facilitating the transition to other languages.
	6. Apply algorithms by writing C code.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Learn steps in problem solving using C.
	2. Understand structure, keywords, operators, and functions of C programming.
	3. Evaluate the concept of I/O functions, header files, and pre- processor directives.
	4. Design and apply concepts of the C language.

Content Outline	• **Introduction to problem solving**: Concept: Steps in problem solving - (Define Problem, Analyze Problem, Explore Solution), Problem solving techniques - (Trial & Error, Brainstorming, Divide & Conquer), Algorithms and Flowcharts (Definitions, Characteristics, Advantage & Disadvantages, Symbols, Examples), Pseudo-code (Definition, Conditional statements, Loops) etc. • **Overview of programming languages**: Definition of the program, Concept- Source code, Object code, Compilation, Interpretation, Execution, Input and Output, Debugging etc., Expressions, control structures; sub routines, Storage management; scoping rules; bindings for names, Storage types: Automatic, external, register and static variables • **Introduction to 'C' Language:** History of C Programming, Structures of C Programming, Simple example, Basic Input/ Output, Function as building blocks. Language Fundamentals: Character Set, C Tokens, Keywords, Identifiers, Variables, Constant, Data Types, Comments • **Operators**: Types of operators, Precedence and Associativity, Expression. Statement and types of statements, Built-in Operators and functions. Console based I/O and related built in I/O Function: printf(), scanf(), getch(), getchar(), putchar(), etc; Concept of header files, Pre-process or directives: #include, #define, Conditional statements and Loops
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Gain proficiency in writing C programs to solve various Problems.
	2. Learn the syntax and semantics of the C language, including its specific features such as pointers and memory management.
	3. Analyze the difference between structure and union.
	4. Design and handle operations of the files.

Content Outline	• **Control structures** Decision making structures: If, If-else, Nested If –else, Switch, Loop Control structures, While, Do-while, For, Nested for, while, do- while loop, jumping statements: break, continue, go to, and exit. • **Functions:** Definition, Basic types of function, Declaration and definition, Function call, Types of function, Parameter passing, Call by value, Call by reference, Scope of variables, Recursion, String: Declaration, string Functions, String Manipulations • **Pointers:** Introduction to pointers, Pointer notation, Pointer arithmetic, Null Pointer • **Arrays:** Definition, Declaration, Initialization, Bounds checking, One-Dimensional Array, Two-Dimensional Array, Passing array to a function, pointer to Array • **Structure and Union:** Introduction to Structure, Definition, Declaration of Structure Variables, Dot Operator, Nested Structure, Array of Structure, pointer to structure, Introduction to Union, Difference between Structure and Union • **File Handling:** Concept of File, Definition, File operations (create, open, read, move, write, close), File opening Mode, Closing a file, Input / output operations, Creating and reading a file, Command Line Argument
Module 3 (Credit 1)	**After learning the module, learners will be able to**
Learning Outcomes	1. Apply operators to write simple programs.
	2. Analyze and use control, iterative, and jumping statements.
	3. Evaluate the use of break and continue statements.

	4. Design programs with header files and pre-processor directives.
Content Outline	• **Simple Program Implementation of Operators:** Built in Operators and function, Arithmetic, Logical, Relational, bitwise, Precedence And Associativity, composite statements. Unary, binary and ternary operators. • **Concept of header files**, Pre-processor directives: #include, #define. And macros implementations, Implementation of Storage types: Automatic external, register and static variables • **Console based I/O and related built in I/O function**: printf(), scanf(), getch(),getchar(), putchar(); • **Control Statement:** Decision Making Statements, if, Nested if, if- else, Nested if-else, if-else-if, switch, etc. The Conditional Expression; • **Iterative Statements**- The for loop, The while loop, The do-while loop; Jumping Statements- The goto & label, The break & continue, The exit() function.
Module 4 (Credit 1)	**After learning the module, learners will be able to**
Learning Outcomes	1. Apply functions in programs.
	2. Analyze the declaration, initialization of pointers, and passing arrays to functions.
	3. Evaluate the definitions and declarations of structure variables in programs.
	4. Design programs effectively using functions, pointers, and structures.
Content Outline	• **Implementation of Functions:** Defining and accessing, passing arguments, Function prototypes, function calling mechanism, call by value, call by reference, recursive function; String Manipulations Pointer Declaration and Initialization of Pointer variables, pointer Arithmetic, Pointers and Character Strings Implementation of 1-D and multidimensional Array, One Dimensional Array, Two- Dimensional Array, Passing array to a function, pointer to Array.

	• **Programs Using Structure and Union:** Defining and Declaring Structure Variables, .Dot Operator, Nested Structure, Array of Structure, pointer to structure, Examples of Union. • **Programs using I/O Operations File Handling:** File Operations (Create, open, read, move, write, close) Input/output operations on file Character by –(fgetc, fputc), Reading and writing files

Reference Books

1. Schildt, H. (2000). C: The Complete Reference (4th ed.). Tata McGraw-Hill Education Pvt.Ltd.

2. Ramkumar, & Agrawal. (1996). Programming in ANSI C. Tata McGraw-Hill. 3. Kanetkar,
   Y. P. (2008). Let Us C. Infinity Science Press.

3. The C Programming Language, Authors**:** Brian W. Kernighan & Dennis M. Ritchie

4. C: The Complete Reference, Author**:** Herbert Schildt

5. The C Programming Language, Kernighan & Ritchie

Internal Assessment 50 Marks Evaluation Scheme:

1. **UT -25**
2. **Project - 25**

The 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 C program, correct logic, syntax	5
Problem Understanding	Clarity of problem interpretation and correct approach	5
Output Correctness	Program compiles without errors and gives correct output	5
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	3
Lab Record/Journal	Completeness, correctness, formatting, and timely submission	2

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

Project 1: Using C Language creates a system that analyzes and evaluates the performance of students based on their scores, using computational problem-solving techniques.

Project 2: Using C Language, build a project that performs various matrix operations and demonstrates their real-world applications.

External Assessment: (Marks 50)

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

Semester - I

.1.2 Major (Core) (Related to Minor)

Course Title	**Introduction to Artificial Intelligence**
Course Credits	**2 Credits**
Course Outcomes	1. Understand the basic concepts and applications of Artificial Intelligence.
	2. Describe various types of intelligent agents and analyze the characteristics of their operational environments.
	3. Formulate and solve Constraint Satisfaction Problems, including real- world examples like map colouring and scheduling.
	4. Integrate knowledge from previous modules to design and demonstrate simple AI applications.
	5. Critically evaluate ethical dimensions of AI systems, including bias, fairness, transparency, privacy, and explainability.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Explain the significance and application of data in real-world contexts
	2. Explain fundamental AI concepts, including definitions, objectives, and major historical milestones
	3. Classify the different types of agents
	4. Analyze and classify task environments (fully vs. partially observable, deterministic vs. stochastic, episodic vs. sequential, static vs. dynamic, single- agent vs. multi-agent) and assess how these affect agent design.

Content Outline	• Definitions and applications of AI: problem-solving, learning, perception, reasoning • Historical milestones: Turing Test, Dartmouth workshop, expert systems, emergence of ML and deep learning • Agent architectures: reflex, model-based, goal-oriented, utility-based agents • Task environments: fully/partially observable, deterministic vs. stochastic, episodic vs. sequential, static vs. dynamic, single vs. multi-agent. **Applications and Ethical AI** • Overview of AI applications: NLP, CV, robotics, expert systems • Basic overview of fuzzy logic, genetic algorithms Discussion on AI ethics, fairness, and societal impact
Learning Outcomes	1. Explain and differentiate between uninformed and informed search strategies, including BFS, DFS, Uniform Cost Search, Greedy Best-First Search, and A, with respect to their applications and performance.
	2. Apply uninformed and informed search algorithms to solve problems such as path finding and puzzle-solving, demonstrating an understanding of their strengths and limitations.
	3. Formulate real-world problems as Constraint Satisfaction Problems (CSPs), identify variables, domains, and constraints, and apply appropriate techniques to find valid solutions.
	4. Analyze adversarial game scenarios using the Minimax algorithm and enhance efficiency using alpha-beta pruning, understanding how these methods optimize decision-making in competitive environments.
	5. Design and interpret flowcharts representing different search strategies, highlighting the algorithmic flow and decision points within each approach.

Content Outline	**Search, Planning & Problem-Solving** ● Uninformed search: BFS, DFS, Uniform Cost Search ● Informed search: Greedy Best-First, A* algorithm ● Constraint satisfaction problems (CSPs): definitions, examples (e.g., scheduling, map coloring) ● Adversarial games: Minimax, alpha-beta pruning **Activities:** Puzzle-solving lab, planning exercises, flowcharts of search strategies

Text Book:

1. Artificial Intelligence (Third Edition) McGraw-Hill Elaine Rich, Kevin Knight.

Reference Books:

1. A First course in Artificial Intelligence (McGraw-Hill) Deepak Khemani.
2. Artificial Intelligence A modern approach (Second Edition) Pearson, Stuart Russell, and Peter
3. Norvig.
4. Fuzzy Logic with Engineering application (Third edition) Timothy J. Rose.
5. Artificial Intelligence and Intelligence system: N. P. Padhy
6. Artificial Intelligence: Patrick Henry Winston.
7. Artificial Intelligence (Structure & Strategies for Complex Problem solving): George F. Luger

Assessment:

Internal Assessment: (50 marks)

Evaluation Scheme:

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.pert 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**
Conceptual Understanding	Understanding of AI concepts and problem formulation	10
Technical Implementation & Accuracy	Produces flawless code and algorithms (search, CSP, ML, demo) with well-documented, efficient, logically structured implementation.	10
Critical Reasoning & Problem-Solving	Chooses and justifies appropriate AI techniques; compares alternatives; analyzes algorithmic trade-offs and optimization opportunities.	10
Ethical Awareness & Reflection	Critically evaluates fairness, bias, transparency, and privacy in AI; proposes concrete mitigation strategies supported by frameworks.	10
Communication & Presentation	Presents ideas clearly and professionally; visuals are polished; responds effectively to questions.	10

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

Project 1: Intelligent Vacuum Cleaner Agent Simulation

Description:
Design and simulate a simple intelligent agent for a vacuum cleaner operating in a 2-room or 3- room environment.

Project 2: AI Agent Analysis Using PEAS Framework

Description:
Analyze real-world AI systems such as chatbots, self-driving cars, smart thermostats, or recommender systems using the PEAS model**.**

Semester - I

.1.3 A. Open Elective Courses/ Generic (OEC any one)

Course Title	**Introduction to Web Technology**
Course Credits	**4 Credits**
Course Outcomes	1. Apply HTML5 and CSS to create and style web pages effectively.
	2. Analyze and implement JavaScript to develop interactive web pages
	3. Evaluate and develop dynamic, database-driven web pages using PHP.
	4. Design and implement comprehensive client-side and server-side scripting programs.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Apply HTTP concepts and security measures to manage client server interactions, including persistent connections and cookie
	2. Analyze and utilize HTML5 elements and tags to create structured and semantic web pages.
	3. Evaluate the effectiveness of HTML5 web forms in capturing various types of user input, ensuring usability and accessibility
	4. Design and implement multimedia content and interactive form elements in web pages to enhance user experience.
Content Outline	• **Introduction to web and Security Concepts** HTTP: Overview – HTTP Basics, Client request, Server response; HTTP Headers; Session Management – Persistent connections, Cookies. General concepts on web server: virtual hosting, General concepts of caching proxy server, Web security, Digital signatures, Digital Certificates, Encryption, and Authentication • **HTML5** Basics of HTML elements and Tags. Introduction of HTML5 (evolutions, limitation of HTML4, advantages of HTML5, Overview of HTML5)

Content Outline	• **Page Layout of Semantic Elements** Header, Navigation, Section & Articles, Footer Organizing Text in HTML, Links and URLs in HTML, Tables in HTML, Images on a Web Page, Image Formats, Image Maps, Colors, FORMs in HTML, Frames in HTML. Working with Multimedia -Inserting Audio and Video on a web page, audio video file format. • **HTML5 Web Forms** HTML 5 Global Attributes Displaying a Search Input. Field, Contact Information Input Fields, Utilizing Date and Time Input Fields, Number Inputs, selecting from a Range of Numbers, Selecting Colors, Creating an Editable Drop-Down, requiring a Form Field, autofocusing a Form Field, Displaying Placeholder Text.
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. CSS syntax, selectors, and properties effectively to style web pages, including background settings, fonts, text styles, and element positioning
	2. Analyze and implement JavaScript programming fundamentals such as variables, operators, control flow statements, and core JavaScript objects (Array, Boolean, Date, Function, etc.), along with handling events and using browser objects.
	3. Evaluate the integration of PHP with HTML for server-side scripting, understanding syntax, variables, and passing information between web pages.
	4. Design and implement advanced JavaScript functionalities including the Document Object Model (DOM) manipulation, form validation techniques, and utilizing cookies for enhanced web application interactivity and functionality.
Content Outline	• **CSS**: Understanding the Syntax of CSS, CSS Selectors, Inserting CSS in an HTML Document, CSS properties to work with- background of a Page CSS Fonts and Text Styles, positioning an element. • **JavaScript:** Using JavaScript in an HTML Document, Programming Fundamentals of JavaScript - Variables, Operators, Control Flow

	Statements, Popup Boxes, Core JavaScript (Properties and Methods of Each) : Array, Boolean, Date, Function, Math, Number, Object, String, regExp, Events and Event Handlers, Browser Objects - Window, Navigator, History, Location, Document, Cookies, Document Object Model, Form Validation using JavaScript. • **INTRODUCTION TO PHP AND SQL**: - Server-side web scripting, Installing PHP, SQL, Adding PHP to HTML, Syntax and Variables, Passing information between pages.
Module 3(Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Apply basic HTML tags to design web pages with organized text, links, tables, images, forms, and multimedia elements.
	2. Analyze and utilize HTML5 semantic elements (navigation, section, articles, footer, etc.) to structure web page content effectively.
	3. Evaluate and implement CSS syntax and properties to enhance web page aesthetics, including backgrounds, fonts, text styles, and element positioning
	4. Design interactive and multimedia-rich web pages by integrating HTML and CSS, ensuring proper use of image maps, audio, and video file formats.
Content Outline	**Use of Basic Tags**, **Image maps, Tables, Forms and Media Design** webpages using the given tools in HTML Navigation, Section & Articles, Footer, aside and more. Organizing Text in HTML, Links and URLs in HTML, Tables in HTML, Images on a Web Page, Image Formats, Image Maps, Colors, FORMs in HTML, Frames in HTML Interactive Elements, Working with Multimedia - Audio and Video File Formats, HTML elements for inserting Audio / Video on a web page **CSS** Syntax of CSS, CSS Selectors, Inserting CSS in an HTML Document, CSS properties to work with background of a Page, CSS properties to work with Fonts and Text Styles, CSS properties for positioning an element
Module 4(Credit 1)	

Learning Outcomes	**After learning the module, learners will be able to**
	1. Apply control flow and looping statements in JavaScript to create dynamic and interactive web pages
	2. Analyze and utilize core JavaScript properties and methods (Array, Boolean, Date, Function, Math, Number, Object, String, RegExp) along with events and event handlers to enhance web page functionality
	3. Evaluate and implement PHP in web development, including syntax, variables, and the integration of PHP with HTML for passing information between pages.
	4. Design and develop database-driven web applications using PHP and SQL, demonstrating the installation and configuration of PHP and the creation of dynamic content
Content Outline	**Java Script:** Control and looping statements and Java Script reference, Using JavaScript design, a web page; Control Flow, Statements, Design a web page demonstrating different conditional statements. Design a web page demonstrating different looping statements; Popup Boxes, Core JavaScript (Properties and Methods of Each): Array, Boolean, Date, Function, Math, Number, Object, String, regExp, Events and Event Handlers. PHP & SQL Demonstrate program in PHP, Installing PHP, SQL, Adding PHP to HTML, Syntax and Variables, Passing information between pages

Reference Books: -

1. Powell, T. (2000). Web design: The complete reference. Tata McGraw-Hill.
2. Powell, T. (2008). HTML and XHTML: The complete reference. Tata McGraw-Hill.
3. Powell, T., & Schneider, F. (2004). JavaScript 2.0:
   The complete reference (2nd ed.). Tata McGraw-Hill.
4. Holzner, S. (2017). PHP: The complete reference. Tata McGraw-Hill.
5. Lecky-Thompson, G. W. (2009). Web programming. Cengage Learning.

Assessment:

Internal Assessment 50 Marks Evaluation Scheme:

1. UT -25

2. Project - 25

The 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**
Implementation of Web Pages with Report	Proper use of HTML, CSS, JavaScript for layout, styling, and interactivity	5
Backend/Server-Side Logic with Report	Use of server-side technologies (PHP, Node.js, Python, etc.) and database integration	5
Functionality and Output with Report	Correct working of features (form submission, validations, dynamic content, etc.)	5
UI/UX Design Principles with Report	Aesthetics, layout consistency, responsiveness, accessibility	5
Viva Voce	Conceptual understanding of HTML, CSS, JS, HTTP, client-server model, etc.	5

Given below is a sample projects but it is expected to work on similar sort of projects

Project 1: Using HTML/CSS design a single product

HTML/CSS design a single-page resume builder using HTML forms.

External Assessment: (Marks 50)

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

Semester - I

.1.3 B. Open Elective Courses/ Generic (OEC any one)

Course Title	**Fundamentals of Statistics using Spreadsheets**
Course Credits	**4 Credits**
Course Outcomes	1. Understand the basic concepts and applications of statistics.
	2. Use spreadsheets effectively for data handling and statistical analysis.
	3. Visualize data and interpret results using appropriate charts and graphs.
	4. Perform descriptive and inferential statistical analyses.
	5. Develop data-based reasoning and critical thinking.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Explain the significance and application of data in real-world contexts
	2. Differentiate between qualitative and quantitative data. Identify appropriate methods of data collection and assess their merits and limitations.
	3. Classify data based on types (primary vs secondary) and levels of measurement.
	4. Use basic spreadsheet tools to enter, format, and organize raw data.

Content Outline	**Introduction to Statistics and Spreadsheets** • Types of data: qualitative vs quantitative. • Levels of measurement. • Introduction to Excel/Google Sheets interface. • Basic spreadsheet operations and formulas. **Introduction to different types of data** • Relevance of data in the current scenario • Types of data: primary and secondary data • Methods of data collection: Questionnaire method, Interview method, case study method in brief • Sources of secondary data collection with merits and Limitations. • Classification of data
Module 2 (Credit 1)	
Learning Outcomes	1. Define and calculate measures of central tendency and dispersion.
	2. Interpret and compare datasets using summary statistics.
	3. Use spreadsheet functions to compute descriptive statistics from raw data.
	4. Calculate and interpret the correlation between two variables.
	5. Use spreadsheet functions to derive regression coefficients and trendlines.

Content Outline	• **Descriptive Statistics** Measures of central tendency (mean, median, mode). Measures of dispersion (range, variance, standard deviation). Percentiles, quartiles Using built-in spreadsheet functions: AVERAGE(), MEDIAN(), STDEV(), etc. • **Correlation and Regression** Correlation: Pearson coefficient Simple linear regression. Using CORREL(), SLOPE(), INTERCEPT(), Interpreting R^2 and trendlines.
Module 3(Credit 1)	
Learning Outcomes	1. Understand various chart types and their appropriate use cases for different data sets.
	2. Apply bar, pie, histogram, scatter, and line charts using spreadsheet tools.
	3. Apply Customize charts by adding titles, labels, legends, and colors for improved clarity.
	4. Understand key concepts in probability, including theoretical and empirical probability
	5. Apply simple probabilistic events using spreadsheet functions.

Content Outline	• **Data Visualization** Types of charts: bar, pie, histogram, scatter, line. Creating and customizing charts in spreadsheets. Interpreting graphical summaries. Chart design best practices • **Probability and Distributions** Basic probability concepts (theoretical & empirical). Random variables, probability rules Common distributions: Binomial, Normal, Poisson Spreadsheet functions: RAND(), BINOM.DIST(), NORM.DIST(), etc.
Module 4(Credit 1)	
Learning Outcomes	1. Understand the role of sampling and the importance of sample size in statistical analysis.
	2. Identify and apply appropriate sampling methods for different scenarios.
	3. Construct and interpret confidence intervals using spreadsheet tools.
	4. Formulate hypotheses and perform basic hypothesis tests (t-test, z- test, chi-square).
	5. Draw conclusions from test results and evaluate statistical significance.
Content Outline	**Inferential Statistics** • Sampling methods. • Confidence intervals. • Hypothesis testing: t-test, z-test, chi-square (basic). • Spreadsheet functions: T.TEST(), Z.TEST(), CHISQ.TEST().

Reference Books:

1. Statistics for People Who (Think They) Hate Statistics: Using Microsoft Excel"
Author: Neil J. Salkind

2. Business Statistics: A First Course" (with Excel Guide)

3. Authors: David M. Levine, Timothy C. Krehbiel, Mark L. Berenson

4. Statistics for Managers Using Microsoft Excel"

5. Authors: David M. Levine, David Stephan, Kathryn A. Szabat

Internal Assessment 50 Marks Evaluation Scheme:

1. **UT -25**
2. **Project - 25**

The 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**
Data Collection, Classification & Spreadsheet Basics, Descriptive Statistics & Interpretation	Correct classification of data, explanation of data collection methods, accurate use of basic spreadsheet functions, proper data organization, Accurate calculations using AVERAGE, MEDIAN, STDEV, VAR, etc., correct interpretation of results and summary measures	**5**
Correlation, Regression & Trend Analysis	Correct calculation of correlation and regression values, use of trendlines, clear explanation of relationships and results	**5**
Data Visualization & Presentation Skills	Proper chart selection, chart creation and customization, interpretation of visuals, design best practices followed	**5**
Probability, Distributions, Inferential Stats	Simulations, probability calculations, matching distributions, use of T.TEST, Z.TEST, CHISQ.TEST with hypothesis interpretation, sampling justification, regularity, discipline, and active participation	**5**
Vice viva	Presentation and question Answer	**5**

External Assessment: (Marks 50)

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

Semester - I

.1.3 C. Open Elective Courses/ Generic (OEC Any one)

Course Title	**Introduction to Cyber Security**
Course Credits	**4**
Course Outcomes	**After Completion of this Course, students will be able to**
	1. Understand basic concepts of Cyber security
	2. Analyze different types of cyber attacks
	3. Implement Basic network Security Measures.
	4. Understand the concept of cryptography and secure communication over the Internet.
	5. Understand the concept of Malicious software
Module1 (Credit1): Fundamentals of Cyber Security	
Learning Outcomes	**After learning this module, learners will be able to**
	1. Understand the basic security principals.
	2. Understand the concepts of the CIA Triad.
	3. Understand all System and Network Security Concept.
	4. Aware about all the malwares.

Content Outline	• **Introduction to Cyber Security:** Introduction to Cyber Security, History, Goals, Need of Security, Principles, Type of all Cyber Attacks, Security Policies(IT Act 2000, cyber Law), Security Techniques, Steps for Better Security, Basics Security Terminology (Cryptography, Hacking, Encryption, Decryption, Cryptanalysis, Cryptanalyst, Steganography), Threats and Vulnerability, Difference between Security and Privacy. • **Malicious Software:** Trojans, Backdoors, Viruses, and Worms, Trojans and Backdoors, Types of Trojans, Types of Viruses. • **Network & System Security:** Public Key Infrastructure (PKI), Firewall, IDS(Intrusion Detection System),IPS(Intrusion Prevention System), Kerberos, Digital Signatures, Digital Certificates, Virtual Private Network(VPN). • **Internet Security Protocol:** Basic concepts, SSL, SHTTP, TSP, SET, SSL versus SET, 3D secure protocol, Electronic Money, Email security, WAP security.
Module 2 (Credit1): Cryptography Basics	
Learning Outcomes	**After learning this module, learners will be able to**
	1. Understand Cryptographic Fundamentals.
	2. Apply Symmetric and Asymmetric Cryptography.
	3. Evaluate Cryptographic Algorithms.
	4. Explore Cryptographic Attacks.

Content Outline	• **Cryptographic techniques:** Plaintext and Ciphertext, Substitution Techniques, Transposition Techniques, Encryption, decryption, Symmetric, Asymmetric Key Cryptography. • **Substitution Ciphers & Transposition Ciphers:** Caesar, Monoalphabetic, Playfair, Hill, Polyalphabetic, One-time Pad or Vernam, Single Columnar, Rail Fence Cipher. • **Symmetric & Asymmetric Cryptographic Algorithms:** Data Encryption Standard (DES), Advanced Encryption Standard (AES), Blowfish, RC5. RSA (Rivest-Shamir-Adleman), Diffie-Hellman Key Exchange, DSA (Digital Signature Algorithm). • **Authentication of People:** Introduction, Authentication Methods(Password-based, Multi-Factor, Biometric, Verification techniques, passwords, length of passwords, smart cards).
Module 3 (Credit 1)	**Introduction to Public Key Cryptography and Authentication Mechanisms**
Learning Outcomes	After learning this module learners will be able to understand
	1. Understand the fundamentals of public key cryptosystems
	2. Describe and analyze the RSA algorithm
	3. Compare Kerberos with other security protocols
	4. Comprehend various authentication methods

Content Outline	• **Public Key Cryptosystems:** Introduction, Public Key Cryptography, RSA Algorithm (Working of RSA, Key length and Security) • **Authentication:** Introduction, Authentications Methods(Password-based, Two-factor, Biometric, Extensible). • **Kerberos:** Basics, Ticket Granting Approach, Public Key Cryptography, Advantges, Weakness and attacks, Applications and Limitations, Comparison of Kerberos with SSL, Authentication Servers.
Module 4 **(Credit 1)**	**Digital Signatures, Email Security, IP Security, and Web Security Protocols**
Learning Outcomes	1. Learners will be able to understand
	2. Analyze algorithms used for digital signatures
	3. Understand electronic mail security mechanisms
	4. Describe IP security concepts and architecture
	5. Understand web security protocols and their components
Content Outline	• **Digital Signatures:** Introduction, Implementation, Association of Digital Signatures and Encryption, Using Different Key pairs for Signing and Encryption. Algorithms for Digital Signature: DSA(Digital Signature Algorithm, ECDSA(Elliptic Curve Digital Signature Algorithm), DSS, Attacks on Digital Signature. • **Electronic Mail Security:** Introduction, Pretty Good Privacy (PGP), MIME, S/MIME, Comparison of PGP and S/MIME. • **IP Security:** Introduction, IP Security Architecture, IPv6, IPsec, IPv4 and IPv6, IPsec Protocols and Operations • **Web Security:** Introduction, SSL, SSL Session and Connection, SSL Record Protocol, Secure Electronic Transaction.

Reference Books:

1. Atul Kahate, Cryptography and Network Security, McGraw Hill.
2. William Stallings' "Cryptography and Network Security: Principles and Practice" is the 8th edition, published by Pearson Education Limited on June 6, 2022.
3. "Network Security: Private Communication in a Public World" is the 3rd edition, published on August 31, 2022.
4. "Cryptography and Information Security" by V.K. Pachghare, published in 2019.
5. "Cybersecurity Fundamentals: A Real-World Perspective" by Kutub Thakur and Al-Sakib Khan Pathan,published on June 30, 2020.
6. ”Computer Security: Principles and Practice (4th Edition)” by William Stallings and Lawrie Brown
7. Cybersecurity and Cyberwar: What Everyone Needs to Know by P.W. Singer and Allan Friedman

Assessment:

Internal Assessment 50 Marks Evaluation Scheme:

1. **UT -25**
2. **Project – 25**

The 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**
Conceptual Understanding	Clear definitions and explanations of cyber security concepts (e.g., RSA, Kerberos, Digital Signatures, IPsec). Demonstrates strong subject clarity and grasp of protocols and algorithms.	5
Accuracy and Completeness	All questions/activities are attempted. Answers are factually correct. Diagrams (e.g., flow of Kerberos tickets), algorithms, or cryptographic steps are accurately represented.	5

Application and Analysis	Correct application of concepts in practical scenarios (e.g., RSA key generation, attack analysis, email security protocols). Includes real-life relevance and security implications.	5
Innovation and Effort	Attempts original examples or case studies. Uses practical demonstrations like protocol simulation, screenshots of Software tools, or explains cybersecurity cases practically.	5
Presentation and Neatness	Well-organized layout with proper use of headings, indentation, bullet points, tables, and clearly labelled diagrams or screenshots of tools (e.g., encryption outputs).	5

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

1. Identify and classify five real-world malware attacks (e.g., WannaCry, Stuxnet). Explain their impact and how they could have been prevented.
2. Create five strong passwords using best practices and explain why they are secure.
3. Write a step-by-step guide on configuring an SSL certificate for a website.

External Assessment: (Marks 50)

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

Semester - I

.1.4 Vocational Skill Courses (VSC-S1)

Course Title	**Introduction to Operating System**
Course Credits	**2**
Course Outcomes	**After going through the course, learners will be able to**
	1. Understand the role, functions, and importance of an Operating System.
	2. Explain core OS concepts such as process management, memory management, and file systems.
	3. Describe CPU scheduling and deadlock concepts with real-world examples.
	4. Analyze how operating systems manage resources efficiently and securely.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Explain the purpose and functions of an Operating System.
	2. Differentiate between various types of Operating Systems.
	3. Understand process concepts and CPU scheduling mechanisms.
	4. Describe basic inter-process communication concepts.

Content Outline	**Introduction to Operating Systems**
	• Definition and objectives of Operating System
	• Functions of an Operating System
	• Evolution of Operating Systems
	Types of Operating Systems
	• Batch Operating System
	• Multiprogramming and Multitasking OS
	• Time-Sharing OS
	• Real-Time Operating System
	• Distributed and Network Operating Systems
	Process Management
	• Process concept and states
	• Process Control Block (PCB)
	• CPU Scheduling concepts
	• Scheduling algorithms:
	  ○ FCFS
	  ○ Shortest Job First (SJF)
	  ○ Priority Scheduling
	  ○ Round Robin

Module 2 (Credit 1)	**Memory, File Systems & Security**
Learning Outcomes	**After learning the module, learners will be able to**
	1. Explain memory management techniques used by Operating Systems.
	2. Describe file system organization and access methods.
	3. Understand deadlocks and resource management.
	4. Identify basic OS security and protection mechanisms.
	5. Explain memory management techniques used by Operating Systems.

Content Outline	**Memory Management** • Logical vs Physical memory • Contiguous and non-contiguous memory allocation • Paging and Segmentation (conceptual) • Virtual memory overview **File System Management** • File concept and file attributes • File access methods • Directory structures • File allocation methods **Deadlocks** • Deadlock concept and conditions • Deadlock prevention and avoidance (overview) **Security & Protection** • User authentication • Access control • Overview of OS security threats

Reference books

1. Silberschatz, Galvin, Gagne – *Operating System Concepts*, Wiley
2. Andrew S. Tanenbaum – *Modern Operating Systems*, Pearson
3. William Stallings – *Operating Systems: Internals and Design Principles*
4. D. M. Dhamdhere – *Operating Systems: A Concept-Based Approach*

<u>Assessment</u>

Internal Assessment – 50 Marks

Evaluation Scheme:

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.pert 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**
Core Functionality & Feature Coverage	Successful implementation of core Operating System concepts such as process management, memory management, file system concepts, scheduling, synchronization, or system utilities with correct and complete functionality.	10
Code Structure & Best Practices	Programs, simulations, or models are well-structured, modular, properly commented, follow naming conventions, and demonstrate correct application of Operating System logic	10
User Interface & Design Consistency	Interaction (menu-driven programs, simulations, or demonstrations) is user-friendly, clear, intuitive, and maintains consistent input/output formatting.	10
Input Validation & Exception Handling	Validates all user inputs properly and handles errors or exceptional cases gracefully using appropriate control structures or logical checks.	10
Technical Documentation	Project report is professionally prepared with a clear structure including introduction, problem definition, system design/flow, implementation details, outputs/screenshots, and explanation of results.	10

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

Project 1: CPU Scheduling Algorithms

1. Study and simulate CPU scheduling algorithms: FCFS, SJF, Priority, Round Robin
2. Compare Waiting Time and Turnaround Time
3. Draw simple Gantt charts
4. Explain which algorithm is better and why

Project 2: Memory Management Techniques

1. Study memory management in OS
2. Explain Fixed & Variable Partitioning
3. Conceptual study of Paging and Segmentation
4. Show memory allocation using diagrams/tables
5. Discuss fragmentation issues

Semester - I

.1.5 A. Skill Enhancement Courses (SEC)

Course Title	**Web Technology-I**
Course Credits	**2 Credits**
Course Outcomes	1. Familiarize with the basics of web technologies and the Internet
	2. Develop skills in designing static and dynamic web pages using HTML and CSS.
	3. Develop proficiency in creating interactive forms and Multimedia content for web applications.
	4. Gain proficiency in crafting responsive, user- friendly, and visually appealing web designs.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Describe the Internet and Web Technologies
	2. Apply Web Standards and Best Practices

Content Outline	**Introduction to Web Technologies Overview of the Internet** History and evolution of the web. Understanding web protocols (HTTP, HTTPS, FTP). The role of web browsers and servers. **Web Standards** W3C guidelines and web accessibility (WCAG). Importance of semantic web and SEO. **HTML5** Structure of an HTML document. Key HTML elements: headings, paragraphs, lists, links, images, tables, and forms. Semantic elements (<header>, <nav>, <section>, <article>, <footer>, etc.). Multimedia elements: audio and video tags. HTML5 form enhancements: new input types and attributes (e.g., date, email, required, etc.)
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Apply the Basics and styles of CSS
	2. Differentiate Between Types of CSS and Their Uses
	3. Utilize CSS Selectors, properties for Styling

Content Outline	**Cascading Style Sheets (CSS) Introduction to CSS (5 Hours) Overview of CSS** History and evolution of CSS. Importance of CSS in web development. CSS syntax: selectors, properties, and values. **Types of CSS** Internal CSS: defining styles within <style> tags. External CSS: linking external stylesheets using <link> tags. **CSS Selectors** Basic selectors: element, class, and ID selectors. Grouping and combining selectors: descendant, child, adjacent sibling, general sibling selectors. Pseudo-classes and pseudo-elements: hover, focus, before, after Text **Properties** Font properties: font-family, font-size, font-weight, font-style. Text alignment, spacing, and decoration: text-align, line-height, letter-spacing, text-decoration. **Colour and Backgrounds** Colour models: HEX, RGB, RGBA, HSL, HSLA. Background properties: background- colour, background-image, background-size, background-repeat, background-position.

<u>Reference Books:</u>

1. Kumar, R. (2012). *Web Technologies: Theory and Practice*. Dreamtech Press.
2. Patel, K. (2015). *Fundamentals of Web Development*. PHI Learning.
3. Mohan, P. K. (2016). *Web Technologies: An Overview*. New Age International.
4. Bansal, N., & Kumar, S. (2018). *HTML, CSS, and JavaScript: A Comprehensive Guide*. Laxmi Publications.
5. Patel, A. (2017). *Responsive Web Design: Techniques and Best Practices*. Oxford University Press.
6. Sharma, R. K., & Kumar, N. (2021). Fundamentals of Web Programming. Tata McGraw-Hill Education.

Assessment:

External Assessment: (Marks 50)

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

Semester - I

.1.5 B. Skill Enhancement Courses (SEC)

Course Title	**UI/UXI Design**
Course Credits	**2**
Course Outcomes	**After Completion of this Course, students will be able to**
	1. Understand the principles of User Interface (UI) and User Experience (UX) design
	2. Create wireframes, prototypes, and user personas based on research and usability.
	3. Apply design thinking methodologies to solve user-centric problems.
	4. Use industry-standard UI/UX tools for creating interactive designs.
Module1 (Credit1)	
Learning Outcomes	**After learning this module, learners will be able to**
	1. Explain the fundamentals of UI/UX and the difference between them.
	2. Conduct user research and define user personas.
	3. Design user flows, wireframes, and low-fidelity prototypes.
	4. Apply visual design principles in interface creation.
Content Outline	**Introduction to UI/UX Design** • Difference between UI and UX • Importance of UI/UX in product development • Design Thinking and User-Cantered Design • UX Design Process Overview User Research & Analysis • Methods of User Research (Interviews, Surveys, Observations) • Creating User Personas • Empathy Mapping. • User Journey Mapping Wire framing & Prototyping

	• Sketching Techniques & Tools • Wireframe Design Principles. • Low-Fidelity vs High-Fidelity Prototypes. • Prototyping Tools (Figma, Adobe XD, Balsamiq) **Visual Design Principles** • Color Theory & Typography. • Layout & Composition. • Iconography & Imagery. • Accessibility & Inclusive Design
Module2 (Credit1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Develop interactive and responsive prototypes using UI tools.
	2. Apply usability testing methods to improve user experience.
	3. Understand the basics of responsive and mobile-first design.
	4. Evaluate and refine designs based on feedback and analytics
Content Outline	**UI Design Tools & Interactive Prototyping** • Introduction to Figma / Adobe XD. • Components and Design Systems. • Interactive Elements and Transitions • Designing for Multiple Devices **Usability Testing & Feedback** • Heuristic Evaluation. • Usability Testing Methods. • Gathering & Analyzing User Feedback. • Iterative Design Process **Responsive and Mobile-First Design** • Responsive Design Principles. • Grid Systems & Breakpoints • Designing for iOS and Android. • Mobile Usability Best Practices **Design Evaluation & Handoff** • Preparing Design Specifications. • Collaboration with Developers. • Design Handoff using Zeplin/Figma Inspect. • Design Documentation & Versioning

Reference Books

1. "Don't Make Me Think" – Steve Krug
2. "The Design of Everyday Things" – Don Norman
3. "About Face: The Essentials of Interaction Design" – Alan Cooper
4. "Lean UX: Designing Great Products with Agile Teams" – Jeff Gothelf
5. "UI is Communication" – Everett McKay

Assessment:

External Assessment: (Marks 50)

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

Course Syllabus

Semester - II

.2.1 Major (Core)

Course Title	**Data Structure & Algorithms**
Course Credits	**4 credits**
Course Outcomes	1. Apply abstract data structures using arrays and linked lists.
	2. Analyze the efficiency and performance of algorithms.
	3. Evaluate data structures like arrays, linked lists, stacks, queues, binary trees, and graphs for different applications.
	4. Design and apply appropriate searching and sorting techniques.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Classify data structures into linear and non-linear categories with suitable examples.
	2. Identify and explain various operations performed on data structures such as insertion, deletion, and traversal.
	3. Implement basic array operations like insertion and deletion in one-dimensional arrays.
	4. Perform traversal of a one-dimensional array using appropriate logic.
Content Outline	• **Introduction:** Introduction to data structure, Classification of data structure, Difference between data types and data structure. Operations performed on data structures, Linear data structure, Non Linear Data Structure. • **Arrays:** Implementations of Array and Operations- Insertion, deletion of an element from one dimensional array, Traversing of array.

Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Implement basic operations on linked lists including insertion and deletion of nodes.
	2. Perform stack operations such as push and pop.
	3. Perform queue operations such as insertion (enqueue) and deletion (dequeue).
	4. Apply queue concepts in real-life and computing applications.
Content Outline	• **Linked List:** Introduction, Key terms, Advantages & disadvantages, Linear linked lists - Types (Singly, Doubly, Circular) Operations (Inserting, Deleting nodes) • **Stack:** Stack Implementation, Operations on stack (Push Pop). Implementation of stack, Applications of stack, Infix prefix, postfix notations • **Queue:** Implementation of Queue Implementation, Operations on queue (Insertion and deletion), Application of Queue and types of queues.

Module 3 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Implement tree structures using array and linked list representations.
	2. Discussing the concept of binary trees and classify types of binary trees.
	3. Perform graph traversal techniques including Breadth First Search (BFS) and Depth First Search (DFS).
	4. Perform Shortest Path and construct spanning trees and minimum spanning trees.

Content Outline	• **Trees** Implementation of tree as Array and Linked lists and Traversal (Inorder, Preorder, Postorder), Binary Trees & types of Binary Trees • **Graphs** Introduction, terminology, Graph representation, Applications of graph, Graph traversal (BFS, DFS, Shortest path), Spanning tree, Minimum spanning tree
Module 4 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Analyze different sorting algorithms such as Bubble Sort, Quick Sort, and Heap Sort.
	2. Implement sequential (linear) search and binary search techniques.
	3. Compare searching algorithms based on efficiency and applicability.
	4. Analyze algorithm characteristics, including space and time complexity, using asymptotic notation.
Content Outline	• **Sorting and Searching** Implementation of searching (Sequential, Binary search) Sorting sums on (Quick Sort, Bubble sort, Selection sort, Insertion Sort.) • **Algorithm Analysis:** Algorithm Characteristics, Space complexity, Time complexity. Asymptotic notation (Big O, 0, Omega and Theta)

Reference Books

1. Trembley, P., & Sorrenson, R. (2005). Data Structure. McGraw Hill Publication. Lipschuiz, S. (2014). Data structures (Schaum's Outline Series). McGraw Hill Publication. Horowitz, E., & Sawhney, S. (2008). Fundamentals of Computer Algorithms. Computer Sci.P.
2. Aho, A. V., Hopcroft, J. E., & Ullman, J. D. (1982). Data Structures and Algorithms. Pearson.
3. Abhyankar, A. (2016) Data Structures and Files. Nirali Prakashan, Educational Publishers Baluja, G. S. (2022). Data Structures Through C. VISIONIAS.
4. Loomis, M. E. S. (1989). Data Management and File Structures (2nd ed.). Prentice Hall. Samanta, D. (2009). Classical Data Structures. PHI, New Delhi.

Assessment:

Internal Assessment 50 Marks Evaluation Scheme:

1. UT -25

2. Project - 25

The 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.

Marks Evaluation Scheme:

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.pert 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**
Conceptual Understanding	Ability to define, describe, and explain core data structures concepts clearly and accurately.	5
Practical Implementation	Competence in coding data structure operations in C, with focus on correctness, memory handling, and efficiency.	5
Problem Solving & Analysis	Ability to choose appropriate data structures, analyze complexity, and solve given problems effectively.	5
Design & Application	Skill in applying data structures to real-world problems (e.g., graphs, trees, queues), and implementing custom solutions.	5
Timely Submission	Submitted within the deadline. Late submission deducts up to 5 marks unless exempted.	5

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

Project 1:

Using C language create a library management system that handles book borrowing, user registrations, book searches, and reservations
(Use: Hash Tables for fast lookups of books, Linked Lists for

maintaining queues of borrowed books, Trees for maintaining ordered catalogues)

Project 2:

Using C language create a flight reservation system where users can book, cancel, and search for flights. (Use: Arrays or Lists for flight schedules, Trees for efficient searching of available flights, Queues for handling waiting lists)

External Assessment: (Marks 50)

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

Semester – II

.2.2 Major (Core)

Course Title	**Computer Organization and Architecture**
Course Credits	**2 Credits**
Course Outcomes	1. Apply basic concepts to understand the structure of computers.
	2. Analyze the working of different interrupts and mapping techniques, and study register organization.
	3. Evaluate different addressing modes and their functionalities.
	4. Design and demonstrate the working of the central processing
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Apply knowledge to understand the basic structure of a computer.
	2. Analyze memory organization and its components.
	3. Evaluate different memory types and their roles within a computer system.
	4. Design and implement memory organization techniques in computing scenarios.
Content Outline	• **Basic Structure of computers:** Basic organization of computer, Intel 8086 Architecture, Basic Measures of Computer Performance, CPU: Registers, Computer Function: Instruction Cycle, Interrupts, Interconnection Structures, Bus Interconnection, Peripheral Component Interconnection (PCI). • **Memory Organization:** Classifications of primary and secondary memories. Types of RAM (SRAM, DRAM, SDRAM, DDR, SSD) and ROM, Characteristics of memory, Memory hierarchy: cost and performance measurement. • **Cache Memory:** Principles, Elements of cache design (Size, Mapping, Replacement, Write policies, Block size) Virtual Memory Concept.

Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Apply various I/O techniques, including programmed I/O, interrupt-driven I/O, and direct memory access, to manage external devices and I/O modules.
	2. Analyze the instruction sets, operand types, operations on operands, and addressing modes of the 8086 processor.
	3. Evaluate the characteristics and performance differences between RISC and CISC architectures, including their pipelining capabilities.
	4. Design processor and register organizations based on the principles of RISC and CISC instruction execution.
Content Outline	• **Input/Output:** External devices, I/O Modules, Programmed I/O, Interrupted-Driven I/O, Direct Memory Access. • **Central Processing Unit:** Instruction sets: Instruction characteristics, Types of operands, Types of operations on operands, addressing modes of 8086 processor, Processor Organization, Register organization. • **RISC:** Instruction Execution, RISC Characteristics, and RISC Pipelining, RISC Vs. CISC, Reduced Instruction Set Computers (RISCs), Introduction to CISC. CISC Characteristics

Reference Books

1. Stallings, W. (2013). Computer organization and architecture: Designing for performance (10th ed.). Pearson.
2. Hayes, J. P. (1988). Computer architecture and organization. McGraw-Hill.
3. Hall, D. V. (2005). Microprocessor and interfacing (2nd ed.). Tata McGraw-Hill.
4. Brey, B. B. (2009). The Intel microprocessors 8086/8088… (4th ed.). PHI.
5. Tanenbaum, A. S. (2016). Structured computer organization (6th ed.). Pearson.
6. Mano, M. (2007). Computer system architecture (3rd ed.). Pearson.
7. Hwang, K., & Briggs, F. A. (1986). Computer architecture and parallel processing. McGraw Hill.
8. Chaudhuri, P. P. (2004). Computer organization and design. Prentice Hall India.
9. Usha, M., & Shrikant, T. S. (2014). Computer system architecture and organization. Wiley India.

External Assessment: (Marks 50)

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

Semester – II

.2.3 Vocational Skill Course (VSC-S2)

Course Title	**Object Oriented Programming Using C++**
Course Credits	**2 Credits**
Course Outcomes	1. Apply object-oriented programming concepts in C++.
	2. Analyze problems and develop C++ programs to solve them.
	3. Evaluate the use of file input/output in C++.
	4. Design solutions using object-oriented programming principles in C++.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Apply fundamental programming concepts including variables, data types, control structures, functions, arrays, and objects.
	2. Analyze and implement object-oriented programming concepts like objects, classes, and defining functions and variables.
	3. Evaluate the use of object-oriented programming in solving programming problems.
	4. Design solutions using a combination of fundamental programming and object-oriented concepts.
Content Outline	• **Evolution of OOP:** Advantages and disadvantages of OOP over its predecessor paradigms. • **Characteristics of Object-oriented Programming:** Abstraction, Encapsulation, Data hiding, Inheritance, Polymorphism, Code Extensibility and Reusability, User defined Data Types. • **C++Program** Structure, Simple Input/ Output Program, Program Comments, Identifiers, Literals, String, Character, Integer, Floating Point, Constants, Keywords, Data Types, Operators in C++, Control Structures in C++.

	• **Object and Classes:** Core object concepts, Encapsulation, Abstraction, Polymorphism, Classes, Messages Association, Interfaces, Implementation of class in C++, C++ Objects as physical object, C++ object as data types constructor Object as function arguments. • **Functions and Variables:** Functions: Declaration and Definition, Variables: Definition Declaration, and Scope, Dynamic Creation and Derived Data, Arrays and Strings in C++.
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Apply concepts of constructors, inheritance (including its types), and polymorphism in C++.
	2. Analyze file input/output handling in C++ and class templates
	3. Evaluate the effectiveness of constructors, inheritance, and polymorphism in solving programming tasks.
	4. Design solutions involving file input/output operations and the utilization of class templates in C++.
Content Outline	• **Inheritance:** Concept of Inheritance, Derived class and base class, Types of Inheritance, Functions and Friend Functions. • **Constructors:** Multiple Constructors and Initialization, Using estructors to Destroy Instances. • **Polymorphism:** Syntax for Operator overloading, overloading of unary and binary operators. • **File input and output:** Reading a File, Managing I/O Streams, opening a File – Different Methods, Checking for Failure with File Commands • **Class templates:** Implementing a class template, implementing class template member functions, Using a class template, Function template.

Reference books:

1. Balaguruswamy, E. (2008). Object Oriented Programming with C++. Tata McGraw– Hill
2. Venugopal, K. R. (1997). Mastering C++. Tata McGraw-Hill Education.
3. Stroustrup, B. (1997). C++ Programming Language (3rd ed.). Addison Wesley.
4. Chandra, B. (1998). A Treatise On Object Oriented Programming using C++. Narosa Publications.
5. Schildt, H. (2001). The Complete Reference CN. Tata McGraw-Hill.

Internal Assessment – 50 Marks

Evaluation Scheme:

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.pert 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**
Evolution & Concepts of OOP	Understanding of OOP evolution, advantages/disadvantages vs procedural programming. Clear explanation of abstraction, encapsulation, inheritance, polymorphism, and reusability.	10
C++ Program Structure	Demonstrates correct program structure with comments, proper use of identifiers, literals, keywords, data types, and operators.	10
Input/Output and Control Structures	Correct use of input/output (cin, cout), conditional statements (if, switch), and loops (for, while, do-while).	10
Classes and Objects	Correct implementation of classes, constructors, destructors, objects, member functions, and passing objects as arguments. Applies encapsulation and abstraction.	10
Inheritance	Accurate implementation of inheritance (single, multiple, multilevel, hierarchical), correct base and derived class usage, and understanding of access specifies.	10

Semester – II

.2.4 Vocational Skill Course (VSC- S3) (With Respect to Minor)

Course Title	**Introduction to Data Science**
Course Credit	**2 Credits**
Course Outcomes	**After successful completion of the course, learners will be able to:** 1. Understand fundamental concepts of Data Science, data types, and the data science lifecycle. 2. Identify different data sources and real-world applications of Data Science. 3. Analyze datasets using data preprocessing and basic statistical techniques. 4. Interpret data insights to support data-driven decision-making.
Module 1 (Credit 1) Introduction to Data Science	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Understand the core concepts and scope of Data Science.
	2. Identify different types of data and their sources.
	3. Explain the Data Science lifecycle and its importance.
	4. Analyze real-world applications of Data Science across domains.

Content Outline	**Introduction to Data Science** • Definition and scope of Data Science • Difference between Data Science, Data Analytics, and Big Data • Data Science lifecycle: data collection, data preparation, analysis, interpretation, and decision-making • Types of data: structured, semi-structured, and unstructured • Sources of data: databases, web data, sensors, surveys • Role of Data Science in organizations **Applications of Data Science** • Data Science in Healthcare • Data Science in Banking and Finance • Data Science in Retail and E-commerce • Data Science in Education • Data Science in Government and Smart Systems
Module 2 (Credit 1) Data Preparation and Statistical Analysis	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Understand basic data preparation and preprocessing techniques.
	2. Apply data cleaning methods to improve data quality.
	3. Use basic statistical measures to analyze and interpret data.

Content Outline	**Data Preparation and Preprocessing** • Understanding datasets and data attributes • Data cleaning concepts • Handling missing values • Removing duplicates and inconsistent data • Data transformation and normalization (conceptual**)** **Basic Statistical Analysis** • Measures of central tendency: mean, median, mode • Measures of dispersion: range, variance, standard deviation • Introduction to correlation analysis • Interpreting statistical results for decision-making

Reference: -

- McKinney, W. (2017). *Python for Data Analysis*. O'Reilly Media.
- Grus, J. (2019). *Data Science from Scratch*. O'Reilly Media.
- VanderPlas, J. (2016). *Python Data Science Handbook*. O'Reilly Media.
- Matthes, E. (2019). *Python Crash Course*. No Starch Press.
- Official Python Documentation – https://docs.python.org

Assessment:

Internal Assessment- 50

Evaluation Scheme:

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. pert 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**
Conceptual Understanding	Demonstrates grasp of empathy, innovation, and problem-solving.	10
Creativity and Original Thinking	Demonstrates originality in ideation exercises (e.g., creative uses of objects). Proposes novel or useful ideas.	10
Practical Application	Develops functional empathy maps, prototypes, and testing results.	10

Team Collaboration & Participation	Actively participates in group tasks and discussions. Demonstrates collaborative effort	5
Communication & Presentation	Effectively communicates ideas through diagrams, posters, or presentations. Clear, neat, and organized submission.	5
Reflection & Feedback Integration	Reflects thoughtfully on user feedback and applies changes. Shows growth in understanding data analytics	5
Timeliness & Completeness	Submits all assignments on time. Completes all tasks as per guidelines (including peer feedback, testing, etc.) prior approval.	5

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

Project 1: Employee Attendance and Productivity Analysis using Python

Description:

In this project, Python is used to analyze employee attendance and productivity data stored in CSV files. The dataset includes employee ID, department, attendance records, working hours, leaves, and task completion status. Using Python libraries such as NumPy and Pandas, data cleaning is performed to handle missing and inconsistent records. Basic statistical analysis is applied to calculate attendance percentage, average working hours, absenteeism rate, and department-wise productivity levels. The analysis helps management understand workforce engagement, identify absenteeism trends, and improve productivity planning without relying on visualization tools.

Project 2: Sales Data Analysis for a Retail Store using Python Description:

This project focuses on analyzing sales transaction data for a retail store using Python. The dataset contains information such as product category, quantity sold, selling price, discount, and sales date. Using Pandas, the data is cleaned, filtered, and grouped to calculate total sales, average revenue per product, and category-wise performance. Basic statistical measures are applied to identify top-selling and low-performing products. The outcome of this project supports data-driven decision-making for inventory management and sales strategy without using dashboards or charts.

Semester – II

.2.5 A. Open Elective Courses/ Generic (OEC any one)

Course Title	**Multimedia System**
Course Credits	**4**
Course Outcomes	**After going through the course, learners will be able to**
	1. Apply creativity, organization, and communication effectively in project stages.
	2. Analyse and optimize text, images, sound, and video for different mediums.
	3. Evaluate mastery of multimedia software tools for image editing, sound editing, animation, and video production.
	4. Design efficient file management strategies using compression techniques like CODECs, GIF, JPEG, MPEG to control file sizes.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Apply understanding of multimedia significance and project development stages.
	2. Analyse font types, utilize text editing tools, and design web specific multimedia content.
	3. Evaluate production of still images using bitmap and vector drawing techniques, and understanding of color theory and image file formats.
	4. Design application of digital audio concepts, video technology workings, and animation principles in multimedia projects effectively.

Content Outline	• **INTRODUCING MULTIMEDIA:** Multimedia- Definitions, Use of Multimedia, Introduction To Making Multimedia: The Stages of a Multimedia Project, Need, Creativity, Organization, Communication. • **Text** About Fonts and Faces, Cases, Serif Versus Sans Serif, Using Text in Multimedia, Computers and Text, Font editing and design tools, Hypermedia and Hypertext. Designing for the World Wide Web- Developing for the Web, Text for the Web, Images for the Web, Sound for the Web, Animation for the Web. • **IMAGES:** Making Still Images, Bitmaps, Vector Drawing, 3-D Drawing and Rendering, Color, Understanding Natural Light and Color, Computerized Color, Color Palettes, Image File Formats.
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Apply multimedia system sound principles, including digital audio, MIDI, and various audio file formats.
	2. Analyse differences between MIDI and digital audio technologies in multimedia contexts.
	3. Evaluate methods for audio CD playback, recording, and voice recognition systems.
	4. Design and implement video technologies, including broadcast standards, digital video formats, and techniques for optimizing video files and animations for multimedia applications.
Content Outline	• **Sound** Multimedia System Sounds, Digital Audio, MIDI Audio, Audio File Formats, MIDI vs Digital Audio, Audio CD Playback. Audio Recording. Voice Recognition & Response. • **Video** How Video Works, Broadcast Video Standards: NTSC, PAL, SECAM, ATSC DTV, Analog Video, Digital Video, Digital Video Standards – ATSC, DVB, ISDB, Video recording & Shooting Videos,

	Video Editing, Optimizing Video files for CD ROM, Digital display standards. • **Animation** Principle of Animations. Animation Techniques, Animation File Formats
Module 3 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Apply image editing techniques using selection tools, layers, masks, and channels in software applications.
	2. Analyse the differences between Macintosh and Windows hardware platforms, including connectivity options like IDE, SCSI, ATA, USB, and Firewire.
	3. Evaluate storage devices and multimedia input/output devices suitable for multimedia projects.
	4. Design strategies for producing web-ready files and integrating typographic design and vector drawing into multimedia content.
Content Outline	• **IMAGE EDITING** Image Editing software: selection tools, working with layers, masks and channels, correcting and enhancing photographs, typographic design and vector drawing, working with 3D images, and producing files for the web. • **Hardware** Macintosh versus Windows, Connections: IDE, SCSI, UIDE, ATA, USB, Firewire etc. Storage devices, Input , Output devices for Multimedia Projects
Module 4 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Apply mastery of image editing software for diverse content creation.
	2. Analyse hardware differences to select appropriate devices.

	3. Evaluate proficiency in various multimedia software tools.
	4. Design effective application of compression principles for web development.
Content Outline	• **Multimedia Software Tools** Text Editing & Word processing tools, OCR S/W, Painting &Drawing Tools, 3D Modelling & Animation Tools, Image editing tools, Sound Editing tools, Animation, Video & Digital movie tools, Overview of various types of Multimedia Authoring tools. • **Compression** CODEC, Types of Compression & redundancies, GIF, JPEG & MPEG Standards Overview, Fractals • **Multimedia tools for WWW & Designing for WWW:** Plug Ins, Text, Images, Sound & Animation for the Web

Reference Books:

1. Buford, J. F. K. (2002). Introduction to Multimedia Systems.Pearson.
2. Vaughan, T. (1999). Introduction to Multimedia. McGraw-Hill Osborne Media
3. Gonzalez, R. C., & Woods, R. E. (2018). Digital Image Processing.Pearson.
4. Pratt, W. K. (1991). Digital Image Processing. John Wiley & Sons
5. Adobe Creative Team. (2021). Adobe Photoshop Classroom in a Book. Pearson.

<u>Assessment</u>

Internal Assessment 50 Marks Evaluation Scheme:

1. UT -25

2. Project - 25

The 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**
Core Functionality & Feature Coverage	Implements all multimedia features (e.g., image/audio/video processing, compression, codecs, GUI) correctly and completely	5
Code Structure & Best Practices	Modular, reusable code with good documentation, follows design patterns and clean coding practices	5
User Interface & Design Consistency	Attractive and intuitive UI/UX; consistent layout, design aesthetics, and responsive behaviour.	5
Input Validation & Exception Handling	Validates multimedia input types (image/audio/video), prevents crashes, and handles exceptions gracefully.	5
Technical Documentation	Well-organized report: introduction, methods, screenshots, technical details, user manual	5

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

Project 1: Mini Multimedia Player with Custom Codec Support

Create a basic multimedia player that supports multiple audio/video formats and demonstrates codec implementation.

Project 2: Image & Video Compression Toolkit

Build a tool that demonstrates compression techniques and shows quality-size trade-offs visually.

External Assessment: (Marks 50)

End Semester examination of 50 marks for 2 hours duration will be conducted by the college**.**

Semester – II

.2.5 B. Open Elective Courses/ Generic (OEC any one)

Course Title	**Introduction to Artificial Intelligence**
Course Credits	**4 Credits**
Course Outcomes	1. Understand the basic concepts and applications of Artificial Intelligence.
	2. Describe various types of intelligent agents and analyze the characteristics of their operational environments.
	3. Formulate and solve Constraint Satisfaction Problems, including real-world examples like map colouring and scheduling.
	4. Integrate knowledge from previous modules to design and demonstrate simple AI applications.
	5. Critically evaluate ethical dimensions of AI systems, including bias, fairness, transparency, privacy, and explainability.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Explain the significance and application of data in real-world contexts
	2. Explain fundamental AI concepts, including definitions, objectives, and major historical milestones
	3. Classify the different types of agents
	4. Analyze and classify task environments (fully vs. partially observable, deterministic vs. stochastic, episodic vs. sequential, static vs. dynamic, single-agent vs. multi-agent) and assess how these affect agent design.

Content Outline	• **Definitions and applications of AI:** problem-solving, learning, perception, reasoning • **Historical milestones:** Turing Test, Dartmouth workshop, expert systems, emergence of ML and deep learning • **Agent architectures:** reflex, model-based, goal-oriented, utility-based agents • **Task environments:** fully/partially observable, deterministic vs. stochastic, episodic vs. sequential, static vs. dynamic, single vs. multi-agent.
Module 2 (Credit 1)	
Learning Outcomes	1. Explain and differentiate between uninformed and informed search strategies, including BFS, DFS, Uniform Cost Search, Greedy Best-First Search, and A, with respect to their applications and performance.
	2. Apply uninformed and informed search algorithms to solve problems such as path finding and puzzle-solving, demonstrating an understanding of their strengths and limitations.
	3. Formulate real-world problems as Constraint Satisfaction Problems (CSPs), identify variables, domains, and constraints, and apply appropriate techniques to find valid solutions.
	4. Analyze adversarial game scenarios using the Minimax algorithm and enhance efficiency using alpha-beta pruning, understanding how these methods optimize decision-making in competitive environments.
	5. Design and interpret flowcharts representing different search strategies, highlighting the algorithmic flow and decision points within each approach.

Content Outline	• **Search, Planning & Problem-Solving** • **Uninformed search:** BFS, DFS, Uniform Cost Search • **Informed search:** Greedy Best-First, A* algorithm • **Constraint satisfaction problems (CSPs):** definitions, examples (e.g., scheduling, map coloring) • **Adversarial games:** Minimax, alpha-beta pruning • **Activities:** Puzzle-solving lab, planning exercises, flowcharts of search strategies
Module 3(Credit 1)	
Learning Outcomes	1. Understand the basic concepts of machine learning.
	2. Understand and distinguish between supervised, unsupervised, and reinforcement learning, identifying when each paradigm is appropriate.
	3. Compare machine learning algorithms in terms of their assumptions, strengths, limitations, and performance across different problem types
	4. Explain the different application and ethical aspects.

Content Outline	• **Machine Learning Foundations** • **Overview:** supervised vs unsupervised learning, classification vs regression. Decision trees, naïve Bayes, k-means clustering, PCA basics • **Neural networks:** architecture, perceptron, backpropagation. Real-world applications and ethical aspects (fairness, bias, transparency)
Module 4(Credit 1)	
Learning Outcomes	1. Identify and describe key AI application areas—including Natural Language Processing (NLP), Computer Vision (CV), Robotics, and Expert Systems—and explain their real-world significance and use cases.
	2. Assess the ethical dimensions of AI, including fairness, bias, transparency, accountability, and privacy, particularly in sensitive applications like education and hiring.
	3. Apply ethical reasoning to case studies in AI, such as grading systems and recruitment tools, to identify risks and propose mitigation strategies.
	4. Understand the responsible AI practices, incorporating human-centered and transparency-driven methods into AI design, deployment, and governance
Content Outline	**Applications and Ethical AI** • Overview of AI applications: NLP, CV, robotics, expert systems • Basic overview of fuzzy logic, genetic algorithms • Discussion on AI ethics, fairness, and societal impact

Text Book:

1. Artificial Intelligence (Third Edition) McGraw-Hill Elaine Rich, Kevin Knight.

Reference Books:

1. A First course in Artificial Intelligence (McGraw-Hill) Deepak Khemani.
2. Artificial Intelligence A modern approach (Second Edition) Pearson, Stuart Russell, and Peter Norvig.

3. Fuzzy Logic with Engineering application (Third edition) Timothy J. Rose.
4. Artificial Intelligence and Intelligence system: N. P. Padhy
5. Artificial Intelligence: Patrick Henry Winston
6. Artificial Intelligence (Structure & Strategies for Complex Problem solving): George F. Luger

Assessment:

Internal Assessment 50 Marks Evaluation Scheme:

1. **UT -25**
2. **Project - 25**

The 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**
Conceptual Understanding	Demonstrates understanding of core Machine Learning concepts including problem definition, data preprocessing, learning algorithms, model training, and evaluation techniques.	5
Technical Implementation & Accuracy	Produces flawless code and algorithms (search, CSP, ML, demo) with well-documented, efficient, logically structured implementation.	5
Critical Reasoning & Problem-Solving	Chooses and justifies appropriate AI techniques; compares alternatives; analyzes algorithmic trade-offs and optimization opportunities.	5
Ethical Awareness & Reflection	Critically evaluates fairness, bias, transparency, and privacy in AI; proposes concrete mitigation strategies supported by frameworks.	5
Communication & Presentation	Presents ideas clearly and professionally; visuals are polished; responds effectively to questions.	5

External Assessment: (Marks 50)

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

Semester – II

.2.5 C. Open Elective Courses/ Generic (OEC any one)

Course Title	**Cyber Forensic**
Course Credits	**4 Credits**
Course Outcomes	**After Completion of this Course, students will be able to**
	1. Understand the Fundamentals of Cyber Forensics
	2. Analyze Digital Evidence and Investigative Techniques
	3. Investigate Cybercrimes and Incident Response
	4. Apply Forensic Tools and Report Findings
	5. Understand the concept of legal considerations.
Module 1 (Credit 1)	
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.
	3. **Analyze suspect behavior and digital footprints**, aiding in understanding user activities during forensic investigations.
	4. **Interpret cyber laws and ethical principles** governing the conduct of forensic investigators.

Content Outline	• **Introduction to Digital Forensic:** Introduction to cybercrimes & 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)	
Learning Outcomes	**After learning this module, learners will be able to**
	1. Identify different types of digital evidence and legal considerations.
	2. **Analysis** various sources of digital evidence, including computers, mobile devices, and network environments.
	3. Learn forensic methodologies and the role of forensic experts in investigations.
	4. **Assess ethical issues in cyber forensics** and apply professional ethical standards during digital investigations.
Content 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 Forensics
Module 3 (Credit 1)	
Learning Outcomes	**After learning this module, learners will be able to**
	1. Gain knowledge of forensic tools used in digital investigations.
	2. **Capture and analyze network traffic** using tools such as Wireshark and TCPDump.

	3. Apply forensic techniques to real-world cybercrime cases.
	4. **Analyze digital evidence from mobile devices**, including Android and iOS platforms.
Content 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 Investigations Mobile Device Forensics (Android, iOS), Why do we need mobile forensics? and Challenges in mobile forensics.
Module 4 (Credit 1)	
Learning Outcomes	**After learning this module, learners will be able to**
	1. **Analyze financial and banking cyber frauds**, including ransomware attacks and cryptocurrency-related crimes.
	2. **Evaluate evidence patterns and attacker behavior** in different categories of cybercrimes.
	3. **Evaluate the role of Artificial Intelligence and Machine Learning** in enhancing digital forensic analysis.
	4. **Assess the future scope of cyber forensics** in addressing evolving cyber threats.

Content 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

References:-

1. "Guide to Computer Forensics and Investigations" by Bill Nelson, Amelia Phillips, and Christopher Steuart is the 7th Edition, published by Cengage Learning.
2. "The Basics of Digital Forensics: The Primer for Getting Started in Digital Forensics" by John Sammons is the Second Edition, published by Syngress,
3. Nilakshi Jain, Dhananjay Kalbande, ”Digital Forensic : The fascinating world of Digital Evidences ” Wiley India Pvt Ltd 2017.
4. Clint P Garrison “Digital Forensics for Network, Internet, and Cloud Computing A forensic evidence guide for moving targets and data , Syngress Publishing, Inc. 2010.
5. Cory Altheide, Harlan Carvey ”Digital forensics with open source tools “Syngress Publishing,Inc. 2011.

Assessment:

Internal Assessment 50 Marks Evaluation Scheme:

1. **UT -25**
2. **Project - 25**

The 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**
Conceptual Understanding	Taught in class	5

Technical Implementation & Accuracy	Taught in class	5
Critical Reasoning & Problem-Solving	Taught in class	5
Ethical Awareness & Reflection	Taught in class	5
Communication & Presentation	Taught in class	5

External Assessment: (Marks 50)

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

Semester – II

.2.6 Skill Enhancement Courses (SEC)

Course Title	**Mathematics -I**
Course Credits	**2 Credits**
	1. Identify and explain the fundamental concepts of sets.
	2. Understand and illustrate concepts of permutations, combinations, and graph theory.
	3. Apply logical reasoning to construct simple mathematical proofs and solve related problems.
	4. Demonstrate various graph traversal techniques and explain their use in different scenarios.
	5. Analyze and apply discrete mathematical structures such as relations, functions, and graph theory to solve computational problems.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Students will demonstrate the ability to perform and apply operations on sets, such as union, intersection, complement, set difference, and Cartesian product, and will understand how these operations are visualized using Venn diagrams. 2. Students will be able to differentiate between permutations and combinations and apply the appropriate formulae to solve simple problems involving arrangements and selections of objects, demonstrating the practical applications of these concepts.
Course Outline	• **Set Theory:** Definition of Sets, Subsets, Cardinality of Sets, types of sets: Equal Sets, Universal Sets, Finite and Infinite • **Sets, proper set, power sets, Operations on Sets:** Union, Intersection, Complement of Sets, set difference, Cartesian Product, Venn Diagrams, and Algebra of sets • **Permutations and Combinations:** Definitions: Permutation, Combination and simple problems

Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Students will be able to define matrices, perform various operations (addition, multiplication, and finding the inverse of square matrices), and compute determinants of matrices, applying properties of determinants to solve problems. 2. Students will be able to define and classify different types of graphs, apply concepts like the Handshaking Lemma, understand graph isomorphism, and work with subgraphs and the complement of graphs to solve related problems.
Course Outline	• **Matrices and Determinants** Definition of a matrix; Operations on matrices; Square Matrix and its inverse; determinants; properties of determinants; • **Graph theory** Graphs, types of graphs, Handshaking Lemma, Isomorphism of graphs, Subgraphs, Complement of graph.

Reference Books:

1. Kolman, Busby and Ross, "Discrete mathematical Structures and graph theory"
2. Alan Doerr, K. Levasseur, "Applied discrete structure for computer science", Galgotia publications, 1988
3. Trembley & Manohar, "Discrete mathematical Structures with application to computer science", McGraw Hill, 1987.
4. Swapan Kumar Chakraborty, BikashKantiSarkar, Discrete Mathematics, Oxford Higher Education, 2011
5. C. L. Liu, D. P. Mohapatra, Elements of Discrete Mathematics A Computer Oriented Approach, Tata Mcgraw-Hill, 3 rd Edition, 2008.S.

<u>Assessment:</u>

Internal Assessment – 50

Case study based on the concept thaught in class and its application to computer science.

Marks Evaluation Scheme:

The Rubric will have the following Evaluation Parameters:		
Evaluation Parameters	**Description / Evaluation Points**	**Marks**
Conceptual Understanding & Practical Implementation	Ability to define, describe, and explain core data structures concepts clearly and accurately. Competence in coding data structure operations in C, with focus on correctness, memory handling, and efficiency.	15
Problem Solving & Analysis	Ability to choose appropriate data structures, analyze complexity, and solve given problems effectively.	15
Design & Application	Skill in applying data structures to real-world problems (e.g., graphs, trees, queues), and implementing custom solutions.	10
Presentation & viva	Submitted within the deadline. Late submission deducts up to 5 marks unless exempted.	10

Course Syllabus

Semester - III

.3.1 Major (Core)

Course Title	**Java Programming**
Course Credits	**1 Credits**
Course Outcomes	1. Apply logic to develop programs in Java
	2. Evaluate the History and Features of Java.
	3. Apply and implement the various concepts of OOP in Java
	4. Analyze and implement the relationships of classes through inheritance and polymorphism.
	5. Implement Java FX programming in Java.
Module 1 (Credit 1)	**Basics of Java**
Learning Outcomes	**After learning the module, learners will be able to**
	1. Apply basic Java syntax to write simple programs using variables, data types, operators, and control structures.
	2. Implement object-oriented programming principles by creating classes and objects
	3. Demonstrate method overloading to enhance program modularity and readability.
	4. Write, compile, and debug Java programs using industry- standard tools and best practices.

Course Outline	• **Evolution of Java:** Java History, Java Features,Comparison of Java with C and C++,Java Support Systems, Java Environment. • **Overview of Java: Introduction**, Java Statements, Java Virtual Machine, main (), public, static, void, string [] args, statements, Command Line Arguments in Java. • **Arrays, Strings and Vectors:** One and two dimensional arrays, Strings, Vectors, Wrapper Classes. • **Classes, Objects and Methods in Java:** Defining a Class, Fields Declaration, Creating Objects, Accessing Class Members, Constructors, Method Overloading, Static Members and Methods.
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Demonstrate method overriding to achieve runtime polymorphism.
	2. Develop programs using abstract methods and abstract classes.
	3. Apply visibility control using access specifiers (public, private, protected, default) to ensure encapsulation.
	4. Create threads by extending the Thread class and implementing the Runnable interface.
Course Outline	• **Inheritance:** Extending a Class, Overriding methods, Final variables and methods, Final classes, Finalizer Methods, Abstract Methods and Classes, Visibility Control. • **Interfaces:** Defining Interfaces, Extending Interfaces, Implementing Interfaces, Accessing Interface Variables. • **Packages:** Java API Packages, Using System Packages, Naming Conventions, Creating Pakages, Accessing a Package. • **Multithreading in Java**: Introduction, Creating Thread, Extending the thread class, Life Cycle of a Thread, using Thread methods, Thread Exceptions, Thread Priority, Thread Synchronization, Implementing the Runnable interface.

Module 3 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Learn Basics of Statistics, Measure Central Tendency and Dispersion.
Course Outline	• **Introduction to collection framework.** Interfaces of collection, Describe the collection framework and its importance in Java Programming, Implement and use various collection classes, Apply generics in classes, methods and collections to eliminate runtime type errors. Implement Exception handling using Try, catch, Throw and finally block.
Module 4 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Create a basic JavaFX application using the standard application lifecycle.
	2. Implement standard JavaFX controls.
	3. Apply the concepts of event-driven programming in JavaFX.
	4. Implement user input validation techniques within JavaFX applications.
Course Outline	• **JavaFX architecture** (Stage, Scene, Nodes, Controls) Creating a basic JavaFX application. • **JavaFX UI controls:** (Button, Label, TextField, TextArea,CheckBox, RadioButton, ProgressBar). Working with containers (Panes, AnchorPane, BorderPane, HBox, VBox, GridPane,StackPane) layout management. Handling form inputs. • **Event-driven programming in JavaFX:** Handling mouse and keyboard events Using EventHandler and EventFilter. • **Handling user input validations:** Handling form inputs. Handling user input validations. Using CSS with JavaFX (stylesheets, selectors, properties). • **Customizing UI components with CSS:**. Working with Font and Color API.

Reference Books:

1. Balaguruswamy, E. (2023). Programming with Java: A Primer (7th ed.). McGraw Hill Education.

2. Schildt, H. (2022). Java: The Complete Reference (12th ed.). McGraw-Hill Education.

3. Goyal, A. (2012). The Essentials of JAVA. Khanna Book Publishing Company Private Limited.

4. Alam, T. (2015). Core JAVA. Khanna Book Publishing Company Private Limited.

5. Liang, Y. D. (2008). Introduction to Java Programming (7th ed.). Pearson.

6. Malhotra, S., & Choudhary, S. (2014). Programming in Java (2nd ed.). Oxford University Press.

Assessment:

Internal Assessment 50 Marks Evaluation Scheme:

1. **UT -25**

2. **Project - 25**

The 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**
Core Functionality & Feature Coverage	Implementation of all core features: OOP principles, control structures, arrays, strings, inheritance, interfaces, exception handling, and packages.	5
Code Structure & Best Practices	Clean, well-structured, modular code following Java naming conventions, and DRY (Don't Repeat Yourself) principles.	5
User Interface & Design Consistency	Intuitive, consistent, user-friendly (even if CLI) UI. Attention to layout, color, and message clarity if GUI is implemented.	5
Input Validation & Exception Handling	Proper validation for user input Exception handling gracefully using try-catch blocks and custom exceptions where appropriate.	5

Project Report & Technical Documentation	Detailed report including objective, methodology, code explanation, screenshots, challenges faced, and conclusion. Documentation of all classes and methods.	5

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

Project 1: Student Management System (Console-based) Project Description: This project involves creating a simple student management system where users can add, delete, update, and display information about students. The system will store student details (e.g., name, roll number, course, and marks) in memory (using collections or arrays). The application will be text-based and will provide a simple user interface for interaction.

Project 2: Online Quiz Application (Java Console-Based) Project Description: Develop a console-based application where users can take multiple-choice quizzes on various topics (e.g., general knowledge, programming, history, etc.). The system should allow administrators to add, edit, and delete questions. Users will take the quiz, and at the end, they will receive their score.

External Assessment: (50 Marks)

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

Semester - III

.3.2 Major (Core)

Course Title	**Database Management System**
Course Credit	**4 Credits**
Course Outcomes	1. Apply the fundamental architecture, data models, and user roles of database management systems to real-world scenarios.
	2. Analyze entity relationships and map conceptual models into relational schemas for structured database design.
	3. Evaluate complex SQL and relational algebra queries fo r accuracy, performance, and business logic implementation.
	4. Apply normalization techniques and transaction management principles to maintain database consistency and reduce anomalies.
	5. Create optimized database solutions with appropriate concurrency control and indexing mechanisms based on application needs.
Module 1 (Credit 1) Introduction to Database Management system and Database Systems Architecture	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Apply the core concepts of data, information, and knowledge to distinguish database use-cases.
	2. Analyze the advantages of DBMS over file systems.
	3. Describe real-world DBMS applications across domains.
	4. Understand 2-tier and 3-tier architectures and abstraction levels.
	5. Identify different database models and users.

Content Outline	**Introduction to Database Systems** • **Basic Concepts**:Data, Information, and Knowledge,Database and Database Management System (DBMS),Characteristics & Advantages of DBMS,Applications of DBMS in Real-World Scenarios • **Database System Architecture**: 2-Tier Architecture & 3-Tier Architecture, Data Abstraction Levels (Physical, Logical, View),Database Users: DBA, Developers, End Users • **Types of Database Models** : Hierarchical Model,Network Model,Relational Model (RDBMS),Object-Oriented Model
Module 2 (Credit 1) Introduction to Data Models, RDBMS and Database Design	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Apply ER modeling to visualize data systems.
	2. Analyze attributes, relationships, and keys to construct ER diagrams.
	3. Convert ER models into relational schemas.
	4. Evaluate integrity constraints in relational designs.
Content Outline	**Entity-Relationship (ER) Model & Relational Model** • **Entity-Relationship (ER) Model** : Entities, Attributes, Relationships,Types of Attributes (Simple, Composite, Derived, Multi-Valued),Relationship Types (One-to-One, One-to-Many, Many-to-Many),Keys: Primary, Foreign, Super, Candidate,ER Diagram Notations,Mapping ER Model to Relational Model • **Relational Model Concepts** : Tables, Tuples, Attributes, AttributesRelational Schema & Constraints,Domain, Entity, Referential Integrity Constraints

Module 3 (Credit 1) Introduction to Functional dependencies, Normalization and Structured Query Language	
Learning Outcomes	**After learning this module learners will be able to understand**
	1. Apply relational algebra operations to retrieve and manipulate datasets.
	2. Create and execute SQL queries for DDL, DML, DCL, and TCL operations.
	3. Evaluate query outcomes using aggregate functions, grouping, and joins.
	4. Analyze indexing and views to improve performance.
Content Outline	• **Structured Query Language (SQL) & Relational Algebra** • **Relational Algebra** : Basic Operations: Selection (σ), Projection (π), Union (∪), Intersection, Difference, Cartesian Product (×),Joins (Inner, Outer, Natural, Equi-Join),Division and Assignment Operations • **SQL (Structured Query Language)** • **Data Definition Language (DDL):** CREATE, ALTER, DROP • **Data Manipulation Language (DML):** INSERT, UPDATE, DELETE • **Data Control Language (DCL):** GRANT, REVOKE • **Transaction Control Language (TCL):** COMMIT, ROLLBACK, SAVEPOINT • **Aggregate Functions:** COUNT, SUM, AVG, MIN, MAX GROUP BY, HAVING, ORDER BY • **Joins:** INNER, LEFT, RIGHT, FULL Subqueries & Nested Queries Views and Indexing
Module 4 (Credit 1) Transaction Processing System, Concurrency Control Techniques and Database Recovery System	
Learning	**Learners will be able to understand**

Outcomes	1. Apply functional dependency rules to detect anomalies.
	2. Analyze relations to achieve appropriate normal forms.
	3. Evaluate transaction properties and implement basic control mechanisms.
	4. Create database systems with concurrency solutions and deadlock strategies.
Content Outline	• **Normalization, Database Design, Transaction Management & Concurrency Control** • **Concept of Normalization:** Functional Dependencies, Anomalies in Unnormalized Relations. Normal Forms: First Normal Form (1NF),Second Normal Form (2NF),Third Normal Form (3NF), Boyce-Codd Normal Form (BCNF),Fourth (4NF) & Fifth Normal Form (5NF),Lossless Join Decomposition & Dependency Preservation • **Transactions and ACID Properties** Atomicity, Consistency, Isolation, Durability (ACID),Transaction States: Active, Partially Committed, Failed, Aborted, Committed • **Concurrency Control Techniques** Problems in Concurrency: Dirty Read, Lost Update, Phantom Read, Locking Mechanisms (Shared & Exclusive Locks),Two-Phase Locking (2PL),Timestamp-Based Protocols, Deadlock Handling: Prevention, Detection, Recovery

Reference Books:

1. Korth, H. F., & Silberschatz, A. (2010). Database system concepts (6th ed.). McGraw- Hill.
2. Elmasri, R., & Navathe, S. B. (2010). Fundamentals of database systems (6th ed.). McGraw-Hill.
3. Bayross, P. (n.d.). Oracle: The complete reference. BPB Publications.
4. Datapro InfoWorld Ltd. (n.d.). Upgrade to Oracle 8.
5. Widom, J., & Wiederhold, G. (1995). Database design. McGraw-Hill.

Assessment:

1. **UT -25**

2. Project - 25

Internal Assessment: (Marks 50) Evaluation Scheme:

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. Pert 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**
Conceptual Understanding	Clear definitions of terms (e.g., DBMS, normalization, transaction). Demonstrates subject clarity.	5
Accuracy and Completeness	All questions are attempted. Answers are factually correct. Diagrams/SQL code are properly done.	5
Application and Analysis	Applies concepts correctly (e.g., ER diagrams, joins, normalization). Includes real-life relevance.	5
Presentation and Neaness	Organized layout. Proper use of headings, indentation, bullet points, tables, and labeled diagrams.	5
Innovation and Effort	Attempting original examples. Uses SQL screenshots, ER tools, or explains business cases practically.	5

Given below are two sample projects but it is expected to work on similar sorts of projects. (ER Diagrams, Tables and SQL Queries)

Project 1: Inventory Management System (Console-based) Project Description:

This project is about creating a console-based Inventory Management System for a small store or warehouse. The system will help manage product details, stock levels, and sales. Users can add new products, update stock quantities, and generate reports on current inventory and sold items. The application will store data using collections like arrays or lists, and ensure data consistency through input validation.

Project 2: Library Management System (Console-based)

Project Description:

This project involves building a console-based Library Management System to manage books, borrowers, and transactions. The system will allow the librarian (admin) to add, delete, update, and search for books. It will also manage borrowing and returning books, tracking due dates, and checking the availability status of each book.
The data will be stored in memory using collections such as arrays, lists, or dictionaries.

External Assessment: (Marks 50)

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

Semester - III

.3.3 Major Core

Course Title	**Cyber Security**
Course Credits	**2 Credits**
Course Outcomes	After Completion of this Course, students will be able to
	1. Understand basic concepts of Cyber security
	2. Analyze different types of cyber attacks
	3. Implement Basic network Security Measures.
	4. Understand the concept of cryptography and secure communication over the Internet.
	5. Understand the concept of Malicious software
Module1 (Credit1): Fundamentals of Cyber Security & Cryptography Basics	
Learning Outcomes	**After learning this module, learners will be able to**
	1. Understand the basic security principals.
	2. Understand the concepts of the CIA Triad.
	3. Understand all System and Network Security Concept.
	4. Aware about all the malwares.

Content Outline	• **Introduction to Cyber Security:** Introduction to Cyber Security, History, Goals, Need of Security, Principles, Type of all Cyber Attacks, Security Policies(IT Act 2000,cyber Law),Security Techniques, Steps for Better Security, Basics Security Terminology (Cryptography, Hacking, Encryption, Decryption, Cryptanalysis, Cryptanalyst, Steganography), Threats and Vulnerability, Difference between Security and Privacy. • **Malicious Software:** Trojans, Backdoors, Viruses, and Worms, Trojans and Backdoors, Types of Trojans, Types of Viruses. • **Network & System Security:** Public Key Infrastructure (PKI),Firewall, IDS(Intrusion Detection System),IPS(Intrusion Prevention System),Kerberos, Digital Signatures, Digital Certificates, Virtual Private Network(VPN). • **Internet Security Protocol:** Basic concepts, SSL, SHTTP, TSP, SET, SSL versus SET, 3D secure protocol, Electronic Money, Email security, WAP security. Vernam, Single Columnar, Rail Fence Cipher. Symmetric & Asymmetric Cryptographic Algorithms: Data Encryption Standard (DES),Advanced Encryption Standard (AES),Blowfish,RC5. RSA (Rivest-Shamir-Adleman),Diffie-Hellman Key Exchange,DSA (Digital Signature Algorithm). • **Authentication of People:** Introduction, Authentication Methods(Password-based, Multi-Factor, Biometric,Verification techniques, passwords, length of passwords, smart cards).

Module 2 (Credit 1) **Introduction to Public Key Cryptography and Authentication Mechanisms** **Digital Signatures, Email Security, IP Security, and Web Security Protocols**	
Learning Outcomes	**After learning this module learners will be able to understand**
	1. Understand the fundamentals of public key cryptosystems
	2. Describe and analyze the RSA algorithm
	3. Compare Kerberos with other security protocols
	4. Comprehend various authentication methods
Content Outline	• **Public Key Cryptosystems:** Introduction, Public Key Cryptography, RSA Algorithm (Working of RSA, Key length and Security) • **Authentication:** Introduction, Authentications Methods(Password- based, Two-factor, Biometric, Extensible). • **Kerberos:** Basics, Ticket Granting Approach, Public Key Cryptography, Advantges, Weakness and attacks, Applications and Limitations, Comparison of Kerberos with SSL, Authentication Servers • **Digital Signatures:** Introduction, Implementation, Association of Digital Signatures and Encryption, Using Different Key pairs for Signing and Encryption. Algorithms for Digital Signature: DSA(Digital Signature Algorithm, ECDSA(Elliptic Curve Digital Signature Algorithm), DSS, Attacks on Digital Signature. • **Electronic Mail Security:** Introduction, Pretty Good Privacy (PGP), MIME, S/MIME, Comparison of PGP and S/MIME. • **IP Security:** Introduction, IP Security Architecture, IPv6, IPsec, IPv4 and IPv6, IPsec Protocols and Operations • **Web Security:** Introduction, SSL, SSL Session and Connection, SSL Record Protocol, Secure Electronic Transaction.

Reference Books:

1. Atul Kahate, Cryptography and Network Security, McGraw Hill.

2. William Stallings' "Cryptography and Network Security: Principles and Practice" is the 8th edition, published by Pearson Education Limited on June 6, 2022.

3. "Network Security: Private Communication in a Public World" is the 3rd edition, published on August 31, 2022.

4. "Cryptography and Information Security" by V.K. Pachghare, published in 2019.

5. "Cybersecurity Fundamentals: A Real-World Perspective" by Kutub Thakur and Al-Sakib Khan Pathan,published on June 30, 2020.

6. ”Computer Security: Principles and Practice (4th Edition)” by William Stallings and Lawrie Brown

7. Cybersecurity and Cyberwar: What Everyone Needs to Know by P.W. Singer and Allan Friedman

Assessment:

External Assessment: (Marks 50)

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

Semester - III

.3.4 Minor Stream

Course Title	**Mathematics – II (Foundation Mathematics for AI)**
Course Credit	**4 Credits**
Course Outcomes	1. Understand numerical representation, floating point arithmetic, and error estimation in computations. 2. Apply interpolation and numerical methods to solve real-world mathematical problems. 3. Use linear algebra concepts such as vectors, matrices, and matrix operations in AI-related contexts. 4. Apply probability and statistical concepts to analyze uncertainty and data distributions. 5. Integrate mathematical foundations with computational thinking required for Artificial Intelligence and Machine Learning.
Module 1 (Credit 1) Mathematical Foundations – Linear Algebra	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Understand floating point representation and different sources of computational errors.
	2. Analyze propagated errors in numerical computations.
	3. Apply interpolation techniques to estimate functional values.

Content Outline	**Floating Point Arithmetic and Errors** • Floating point representation • Sources of errors • Propagated errors **Interpolation** • Operators: Δ, ∇, E and their relations • Newton's forward and backward interpolation formula • Lagrange's interpolation formula • Divided differences • Newton's divided difference formula

Module 2 (Credit 1) Numerical Methods for Linear & Non-Linear Equations	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Solve systems of linear equations using numerical techniques.
	2. Apply numerical methods to find roots of non-linear equations.
	3. Analyze convergence and accuracy of numerical solutions.
Content Outline	• Matrix inversion method • Gauss elimination method **Solution of Non-Linear Equations** • Bisection method • Newton–Raphson method • Regula-Falsi method
Module 3 (Credit 1) Linear Algebra Foundations for AI	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Understand vectors, matrices, and matrix operations used in AI systems.
	2. Apply linear algebra concepts to represent data and transformations.
	3. Perform numerical computations relevant to AI models.
Content Outline	• Scalars, vectors, and matrices • Vector operations • Matrix addition, subtraction, and multiplication • Transpose of a matrix • Identity and inverse matrices • Applications of linear algebra in AI (conceptual)

Module 4 (Credit 1) Probability & Statistics Foundations for AI	
Learning Outcomes	1. Understand probability concepts used in AI and data-driven systems.
	2. Apply statistical measures to analyze datasets.
	3. Interpret uncertainty and variability in data.
Content Outline	• Introduction to probability • Sample space and events • Probability rules • Random variables (conceptual) • Measures of central tendency: mean, median, mode • Measures of dispersion: variance and standard deviation • Introduction to probability distributions (conceptual) • Role of statistics in AI decision-making

Reference: -

1. Strang, G. (2016). *Introduction to Linear Algebra*. Wellesley-Cambridge Press.
2. Deisenroth, M. P., Faisal, A. A., & Ong, C. S. (2020). *Mathematics for Machine Learning*. Cambridge University Press.
3. McKinney, W. (2017). *Python for Data Analysis*. O'Reilly Media.
4. VanderPlas, J. (2016). *Python Data Science Handbook*. O'Reilly Media.
5. Official Python & NumPy Documentation

Assessment:

1. **UT -25**
2. **Project - 25**

The 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**
Conceptual Understanding	Demonstrates grasp of empathy, innovation, and problem-solving.	5

Creativity and Original Thinking	Demonstrates originality in ideation exercises (e.g., creative uses of objects). Proposes novel or useful ideas.	5
Practical Application	Develops functional empathy maps, prototypes, and testing results.	5
Team Collaboration, Participation, Communication & Presentation	Actively participates in group tasks and discussions. Demonstrates collaborative effort & Effectively communicates ideas through diagrams, posters, or presentations. Clear, neat, and organized submission.	5
Reflection & Feedback Integration	Reflects thoughtfully on user feedback and applies changes. Shows growth in understanding data analytics	5

Assessment:

External Assessment: (Marks 50)

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

Semester – III

.3.5 A. Open Elective Courses/ Generic (OEC any one)

Course Title	**Intellectual Property Rights and Cyber Law**
Course Credits	**2 Credits**
Course Outcomes	**After Completion of this Course, students will be able**
	1. Evaluate the importance of Intellectual property.
	2. Apply knowledge of Intellectual property to protect creative work.
	3. Apply Patent Registration Procedure.
	4. Evaluate how to protect intellectual property.
	5. Evaluate the concept of cyber law and IT Act.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Identify the use of Intellectual Property. 2. Ability to use Intellectual property to protect their work.
Content Outline	• **Intellectual Property Rights:** Intellectual Property Rights and its types. Basic principles, objectives of Copyrights, Trademark, Design Rights, Patent & Patent Registration Procedure. International Background of Intellectual Property. • **Ownership and Enforcement:** Copyrights, Patent, Trademark, Design Right. • **Practical Aspects of Licensing**: Benefits, important clauses, licensing clauses.
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to** 1. Identify Cyber Law and Cyber Crime in detail. 2. Identify Objective of IT Act 2000

Content Outline	• **Cyber Law:** Introduction, Basic Concepts of Cyber Law, Scope of Cyber Laws, Cyber Jurisprudence. Cyber Security Regulations, Role of International Law, Cyber Security Standards, Indian Cyber Space, National Cyber Security Policies. • **Law of Digital Contracts:** The Essence of Digital Contracts, The System of Digital Signatures, The Role and Function of Certifying Authorities, • **Information Technology Act 2000:** Objectives of IT Act 2000.Following sections to be explained in detail Sections 43(Penalty for damage to the computer), Section 66(Hacking of computer systems), Section 67(publishing of obscene information), Section 72(penalty for breach of privacy), Section 73(penalty for publishing false digital signature certificates). • **Cyber Crimes:** Cyber Crimes and Types of Cyber Crimes • **Ethical hacking:** Introduction to Ethical Hacking.

Reference Books

1. A Complete Manual on Intellectual Property Rights & Cyber Laws in India by by Dr. Anusuya Yadav
2. Essential Reading: Title: "Cyber Crimes and Laws: An Introduction" by Dr. Pavan Duggal, Publisher: Universal Law Publishing, 2022 edition.
3. Supplementary Reading: Title: "Cyber Crime and Cyber Laws" by: V. K. Ahuja , Publisher: Taxmann Publications 2021 edition
4. Title: "Cyber Laws: A Comprehensive Guide" by Rohas Nagpal, Publisher: McGraw Hill Education , 2020 edition.
5. Title: "Cyber Crime and the Law: Challenges, Issues, and Response" by Yogesh K. Dwivedi, Himanshu Gupta, and Matthew K. O. Lee, Publisher: Springer India, 2019 edition.

Assessment:

External Assessment: (Marks 50)

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

Semester - III

.3.5 B. Open Elective Courses/ Generic (OEC)(any one)

Course Title	**Digital Marketing**
Course Credits	**2 Credits**
Course Outcomes	**After Completion of this Course, students will be able**
	1. Understand the fundamentals of digital marketing and its role in the business environment
	2. Develop digital strategies including SEO, SEM, email, and content Marketing
	3. Design and execute performance-based digital ad campaigns
	4. Analyze campaign metrics using web and social analytics.
	5. Use social media tools and analytics platforms to plan and measure marketing campaigns
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Describe the evolution and scope of digital marketing 2. Identify key components of a digital marketing strategy 3. Apply the basics of search engine optimization (SEO) and search engine marketing (SEM). 4. Use tools like Google Ads and Google Analytics
Content Outline	• **Introduction to Digital Marketing**: Definition, Importance, Traditional vs. Digital Marketing, Latest Trends • **Digital Marketing Channels**: Owned, Earned, and Paid Media • **Search Engine Optimization (SEO)**: On-page & Off-page SEO, Keywords, Backlinks, SEO tools • **Search Engine Marketing (SEM)**: Google Ads, PPC, Keyword Planning, Ad Creation, Bidding Strategies • **Digital Marketing Tools**: Introduction to Google Analytics, Google Search Console, SEMrush

Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to** 1. Create effective content and email marketing campaigns 2. Develop marketing strategies using customer segmentation 3. Automate digital communication using CRM tools 4. Comply with ethical and legal standards in digital marketing
Content Outline	• **Content Marketing**: Blogging, Video Marketing, Influencer Marketing, Content Strategy • **Email Marketing**: Campaign Planning, Tools (Mailchimp), A/B Testing, GDPR Compliance • **Customer Relationship Management (CRM)**: Email Automation, Segmentation, Lead Scoring • **Digital Ethics and Laws**: Data Privacy, Intellectual Property, Digital Advertising Laws

<u>Reference Books</u>

1. Digital Marketing: Strategy, Implementation, and Practice – Dave Chaffey & Fiona Ellis-Chadwick
2. Marketing 5.0: Technology for Humanity – Philip Kotler, Hermawan Kartajaya & Iwan Setiawan
3. Contagious: Why Things Catch On – Jonah Berger
4. Jab, Jab, Jab, Right Hook – Gary Vaynerchuk
5. SEO 2024 – Adam Clarke
6. Digital Marketing for Dummies – Ryan Deiss & Russ Henneberry
7. Google Ads (AdWords) Workbook 2024 – Jason McDonald
8. Social Media Marketing Workbook 2024 – Jason McDonald

Assessment:

External Assessment: (Marks 50)

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

Semester – III

.3.5 C. Open Elective Courses/ Generic (OEC)(any one)

Course Title	**E-Commerce**
Course Credits	**2 Credits**
	1. Identify and explain an analytical framework to understand the emerging world of e-commerce
	2. To make the learners familiar with current challenges and issues in e- commerce.
	3. To develop the understanding of the learners towards various business models.
	4. Analyze the impact of e-payment, legal, and security issues in E- Commerce.
	5. Evaluate different digital marketing strategies.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Analyze the historical evolution, scope, and business models of E- Commerce (B2B, B2C, B2G), and explain the roadmap of E- Commerce development in India by identifying its main activities, functions, and benefits and challenges; and illustrate how different models apply to real-world marketing, sales, and promotional strategies. • Apply E-Commerce concepts to design a basic business model by evaluating the E-Commerce Sales Life Cycle (ESLC), comparing alternative B2B/B2C processes, and recommending strategies for effective online operations aligned with Indian market trends and technological advancements.

Course Outline	• **Introduction to Electronic Commerce–Evolution and Models:** • Evolution of E-Commerce-Introduction, History/ Evolution of Electronic Commerce, Roadmap of E-Commerce in India, Main activities, Functions and Scope of E-Commerce. • Benefits and Challenges of E-Commerce, E-Commerce Business Strategies for Marketing. Business Models of E-Commerce- Characteristics of Business to Business(B2B), Business to Consumers (B2C), Business to Government(B2G) Concepts of other models of E-commerce Business to Consumer E-Commerce process, Business to Business E-Commerce- Need and Importance, alternative Models of B2B E-Commerce. E- Commerce Sales Product Life Cycle (ESLC)Model
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Explain the scope and techniques of E-Marketing—including traditional web promotion methods, web counters, advertisements, and social media—and evaluate customer strategies, support activities, and digital planning approaches; further, justify the feasibility of launching an Internet business by weighing the pros and cons of online shopping. 2. Explain the scope and techniques of E-Marketing—including traditional web promotion methods, web counters, advertisements, social media—and evaluate customer strategies, support activities, and digital planning approaches; further, justify the feasibility of launching an Internet business by weighing the pros and cons of online shopping.

Course Outline	**E-marketing and Electronic Payment** • System E-Marketing-Scope and Techniques of E-Marketing, Traditional web promotion; Web counters; Web advertisements, Role of Social media. • E-Commerce Customer Strategies for Purchasing and Support activities, Planning for Electronic Commerce and its initiates, The pros and cons of online shopping, Justify an Internet business. • Electronic Payment System-Characteristics of E-payment system, SET Protocol for credit card payment, prepaid epayment service, post-paid E-payment system, Types of payment systems. Operational, credit and legal risks of E-payment system, Risk management options for E-payment systems, Set standards/principles for E-payment.

Reference Books:

1. K Laudon, Kenneth C. and Carol Guercio Traver E-commerce: business, technology, society. (New Delhi : Pearson Educatin).
2. Awad, Elias M., Electronic Commerce: From Vision to Fulfillment (NewDelhi : Pearson Education).
3. Kalakota, Ravi and Marcia Robinson . Business 2.0: Roadmap for Success (newDelhi : Pearson Education).
4. Smith, P.R. and Dave Chaffey, eMarketing eXcellence; The Heart of eBusiness (UK : Elsevier Ltd.)
5. VivekSood Cyber Laws Simplified-TMH
6. VakulSharma Handbook of cyber Laws-Macmillan
7. Sundeep Oberol e Security and you-TMH
8. Greenstein & Feinman Electronic Commerce-Security, Risk Mgt and Control-TMH

External Assessment: (Marks 50)

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

Semester - III

.3.7 Major (Core)

Course Title	**Linux Operating System**
Course Credits	**2**
Course Outcomes	**After going through the course, learners will be able to**
	1. Apply Linux environment skills effectively for various tasks and operations.
	2. Analyse and utilize basic Linux commands and shell scripting techniques.
	3. Evaluate file system creation, directory management, and related operations using Linux programs.
	4. Design and implement solutions using a diverse set of standard Linux programming and development tools.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Apply basic Linux commands proficiently for various tasks and operations.
	2. Analyse the significance of Linux architecture and its key features.
	3. Evaluate Linux utilities for creating and managing simple file processing operations effectively.
	4. Design and demonstrate memory management techniques in file handling, including file and region locking mechanisms.

Content Outline	• **Introduction to Linux** Introduction to Unix architecture, General-purpose utilities, All basic commands introduction, Usage of commands. • **Files and Directories:** File Concept, File types, File System Structure, File metadata- Inodes, kernel support for files, file System calls for file I/O operations- open, create, read, write, close. • **Directories-** mkdir, rmdir, chdir, obtaining current working directorygetcwd, directory contents, scanning directories opendir, readdir, closedir, rewind dir functions.
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Apply the concept of filters in Linux to manipulate data effectively.
	2. Analyse different editors available in Linux and their functionalities.
	3. Design and implement shell scripts capable of performing complex tasks in a shell programming environment.
	4. Evaluate the role and functionalities of the Linux kernel in system operations.
Content Outline	• **The vi Editor** Introduction to the vi editor of Linux. all commands. • **Introduction to filters** Simple Filters, Filters using regular expressions - use of grep command • **Introduction to shell concept and writing shell script-** What is kernel, What is Shell, what is terminal, Advantage and disadvantage, first simple Shell Programme.

Reference books:

1. Card, R., Dumas, E., & Mevel, F. (2003). The Linux kernel book. Wiley.
2. Das, S. (2017). Unix concepts and applications (4th ed.). TMH.
3. Suehring, S. (2002). MySQL Bible. John Wiley & Sons.
4. Lerdorf, R., & Tatroe, K. (2002). Programming PHP. O'Reilly Publications. 5. Collings, T., & Wall, K. (Wiley). (2003). Red Hat Linux network and system administration (3rd ed.).
5. Mathews, N. (2007). Beginning Linux programming (4th ed.). Wrox Press.
6. Koparkar, P. (2001). Unix for you. Tata McGraw-Hill.
7. Kanetkar, Y. P. (2013). Unix shell programming (1st ed.). BPB Publications.

Assessment:

Internal Assessment: (50 marks) Evaluation Scheme:

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.pert 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**
Core Functionality & Feature Coverage	All required Linux features and functions are implemented (file ops, filters, memory mgmt, shell scripts, vi editor, etc.) with full functionality	10
Code Structure & Best Practices	Code is modular, well-commented, follows naming conventions, and uses appropriate Linux commands and logic	10
User Interface & Design Consistency	Command-line interaction is user-friendly, intuitive, with helpful prompts and consistent format	10
Input Validation & Exception Handling	Validates all user inputs (e.g., file names, paths, user options); handles all exceptions gracefully using built-in checks or control structures	10
Technical Documentation	Professional report with clear structure: introduction, architecture, code snippets, output screenshots, and detailed explanation	10

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

Project 1: Linux Automation & Memory Safety Toolkit

A multi-functional script that automates system tasks (file handling, process monitoring, logs) and implements memory/file locking mechanisms.

Project 2: Linux Command Trainer with vi & Filters

Build an interactive command-line learning environment for beginners. Create interactive shell script tutorials that teach use of:
grep, awk, sed, cut, sort, etc. Regular expressions and filters
Build an in-terminal vi editor tutorial, explaining modes, commands, macros
Write a formal report, use screenshots, and prepare a demo presentation with voice-over

Course Syllabus

Semester – IV

.4.1 Major (Core)

Course Title	**Advanced Java**
Course Credits	**4 Credits**
Course Outcomes	**After Completion of this Course, students will be able**
	1. Develop advanced Java applications using object- oriented programming principles.
	2. Integrate Java applications with databases using JDBC for efficient data management.
	3. Design and implement dynamic web applications using Servlets and JSP.
	4. Implement Java frameworks such as Spring and Hibernate for enterprise-level development.
	5. Implement secure and scalable Java applications by applying best practices in exception handling, multithreading, and deployment.
Module 1 (Credit 1) Understanding Java EE	
Learning Outcomes	**After learning the module, learners will be able to** 1. Design and develop Java EE web applications using Servlets, JSP. 2. Implement Session Management & Request Handling in a multi-user web application. 3. Integrate Java EE applications with databases using JDBC 4. Deploy Java EE applications on Tomcat Server and understand enterprise-level web application.

Content Outline	• **Understanding Java EE:** What is an Enterprise Application? What is java enterprise edition? Java EE Technologies, Java EE evolution, Tomcat server. • **Java EE Architecture, Server and Containers:** Types of System Architecture, Java EE Server, Java EE Containers. • **Introduction to Java Servlets:** The Need for Dynamic Content, Java Servlet Technology, Why Servlets? What can Servlets do? Servlet API and Lifecycle: Java Servlet API, The Servlet Skeleton, The Servlet Life Cycle, A Simple Welcome Servlet Working with Servlets: Getting Started, Using Annotations Instead of Deployment Descriptor. • **Request Dispatcher:** Resquestdispatcher Interface, Methods of Requestdispatcher, Requestdispatcher Application. COOKIES: Kinds of Cookies, Where Cookies Are Used? Creating Cookies Using Servlet, Dynamically Changing the Colors of A Page SESSION: What Are Sessions? Lifecycle of Http Session, Session Tracking With Servlet API, A Servlet Session Example. • **Working with Databases:** What Is JDBC? JDBC Architecture,Accessing Database, JDBC-ODBC Bridge Driver, Native Driver, Network Protocol Driver, and Thin Driver.JDBC Environment should set-up along with database creation. The Servlet GUI and **Database** Example.Connectivity with SQL, JDBC ResultSet, Execute read select query, update records and execute query.
Module 2 (Credit 1) JSP	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Analyze the lifecycle of a JSP page and describe how it functions within a web container.
	2. Demonstrate the use of various JSP elements, including declarations, expressions, directives, and action tags, to build dynamic web pages.
	3. Design and create JavaBeans to encapsulate application data and business logic using setter and getter methods.

	4. Compare and contrast JSP and Servlets in terms of lifecycle, functionality, and usage in web application development.
Content Outline	1. **Java Server Pages Standard Tag Libraries:** What is wrong in using JSP Scriptlet Tags? How JSTL Fixes JSP Scriptlet's Shortcomings? Disadvantages of JSTL, Tag Libraries. 2. **Introduction To Java Server Pages:** Why use Java Server Pages? Disadvantages Of JSP, JSP v\s Servlets, Life Cycle of a JSP Page, How does a JSP function? How does JSP execute?
Module 3 (credit 1) Introduction to Hibernate	
Learning Outcomes	1. Design Hibernate Application
	2. Design JPA application.
Content Outline	• **Writing JPA Application:** Application Requirement Specifications, Software Requirements, The Application Development Approach, Creating Database and Tables in Mysql, creating a Web Application, Adding the Required Library Files, creating a Javabean Class, Creating Persistence Unit [Persistence.Xml] , Creating JSPS, The JPA Application Structure, Running the JPA application. • **Introduction to Hibernate:** What is Hibernate? Why Hibernate? Hibernate, Database and The Application, Components of Hibernate, Architecture of Hibernate, How Hibernate Works? • **Writing Hibernate Application:** Application Requirement Specifications, Software Requirements, The Application Development Approach, Creating Database and Tables in Mysql, creating a Web Application, Adding the Required Library Files, Creating Hibernate Configuration File, Adding a Mapping Class. Creating JSPS, Running The Hibernate Application.

Module 4 (Credit 1) Introduction to SpringBoot	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Evaluate the core concepts of the Spring Framework and Spring Boot. 2. Develop web applications using Spring MVC, implementing controllers, views, and request mappings. 3. Create RESTful web services & microservices with Spring Boot. 4. Create and expose RESTful web services using Spring Boot, and explain the difference between REST and SOAP.
Content Outline	• **Spring Framework (Spring Boot):** Introduction to Spring Core & Spring Boot. Environment Setup. • Spring MVC for web applications RESTful Web Services using Spring Boot Spring Boot with JPA & Hibernate RESTful Web Services & Microservices • **Introduction to Web Services (REST vs. SOAP):** Creating RESTful APIs with Spring Boot JSON& XML Data Exchange using Jackson Microservices Architecture in Java Building Microservices with Spring Boot API Gateway & Load Balancing with Spring Cloud

Reference Books

1. **"Pro JPA 2: Mastering the Java Persistence API"** - **Author:** Mike Keith & Merrick Schincariol
2. **"Java Persistence with Hibernate"** - **Author:** Christian Bauer, Gavin King, Gary Gregory
3. **"Spring Data: Modern Data Access for Enterprise Java"** - **Author:** Mark Pollack, Oliver Gierk.
4. **"Spring Boot in Action"** -**Author:** Craig Walls
5. **"Spring Boot 3 and Spring Framework 6: Learn to Develop, Test, and Deploy Your Spring Applications"** - **Author:** Claudio Eduardo de Oliveira

<u>Assessment</u>

Internal Assessment 50 Marks

Evaluation Scheme:

1. **UT -25**
2. **Project - 25**

The 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.

Evaluation Criteria	**Description**	**Marks**
1. Core Functional Implementation	Functionality (Core Features)	5
2. User Interface Design & Mobile Responsiveness	Code Quality (Structure, Readability)	5
3. Code Quality & Structure	GUI Design & User Experience	5
4. Input Validation & Error Handling	Input Handling & Error Management	5
5. App Documentation / Project Report	Report writing or Documentation	5

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

Project 1: Build a mini e-commerce website where products are dynamically fetched and displayed using Servlets, JSP, and a database**.**

Project 2: Simple CRUD Application using Spring Boot + JPA

Objective: Build a Spring Boot CRUD app that manages a specific entity (e.g., Books, Students, Employees).

External Assessment: (Marks 50)

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

Semester - IV

.4.2 Major (Core)

Course Title	**Web Technology**
Course Credit	**4 Credit**
Course Outcome	1. Demonstrate a comprehensive understanding of fundamental Web technologies and their applications in modern web development.
	2. Develop interactive, responsive, and visually appealing web pages using HTML, CSS, and JavaScript.
	3. Implement client-side scripting techniques with JavaScript to enhance user experience and interactivity.
	4. Design and construct dynamic, component-based web applications using ReactJS, adhering to industry best practices.
	5. Integrate APIs and manage state effectively to build scalable and data-driven web applications.
Module 1(Credit 1) Introduction to Web Technologies	
Learning Outcome	**After learning the modules, learners will be able to**
	1. Understand the basics of web development and the role of front-end technologies.
	2. Analyze and implement HTML and CSS for structuring and styling web pages.
	3. **Develop** structured web pages using **HTML5**, incorporating forms, semantic tags, and multimedia elements.
	4. Develop structured and responsive web pages using CSS frameworks.

Content Outline	• **Introduction to Web Technologies** Basics of the Internet, Web Servers, and Web Browsers Introduction to HTML, CSS, and JavaScript Role of Front-end and Back-end Technologies HHTML & CSS • **HTML Basics:** Structure of an HTML document, Elements, Attributes. Forms and Form Handling, HTML5 Features CSS Basics: Selectors, Box Model, Flexbox, Grid Responsive Design: Media Queries, Bootstrap Overview.
Module2 (Credit 1) JavaScript Basics	
Learning Outcome	**After learning the modules, learners will be able to**
	1. Evaluate and implement JavaScript for web interactivity.
	2. Develop basic HTML5 forms and semantic layouts.
	3. Use ES6 features such as let and const for variable declaration, arrow functions for more concise function definitions, and template literals for efficient string handling.
	4. Describe the structure of HTML documents, tags, attributes, and elements.

Content Outline	• **Introduction to JavaScript** Role of JavaScript in web development, client-side scripting, and integration with HTML and CSS for dynamic web pages. • **JavaScript Basics and ES6 Variables** JavaScript syntax, statements, comments, and variable declaration using letand const; overview of data types. • **Functions and ES6 Arrow Functions** Function declaration and invocation, parameters and return values, arrow functions, and their advantages. • **DOM Manipulation** Understanding the Document Object Model; selecting, modifying, creating, and removing HTML elements dynamically. • **Event Handling** Event-driven programming; handling user events such as click, submit, change using event listeners. • **JavaScript Form Handling and Validation** Accessing form data, client-side validation, handling form submission, and displaying dynamic feedback.
Module3 (Credit 1)	**Advanced Java Script and Introduction to React JS**
Learning Outcome	**After learning the modules, learners will be able to**
	1. Evaluate concepts of callbacks, and demonstrate their usage in handling a synchronous operation in JavaScript.
	2. Implement Promises in JavaScript to handle asynchronous tasks, and chain multiple promises to create more efficient asynchronous workflows.
	3. Define key React concepts such as components, JSX, props, and state, and explain their role in building dynamic web applications.
	4. Implement React Hooks such as useState, useEffect, and others, in functional components.

Content Outline	• API **Advanced JavaScript** Callbacks, Promises, and Async/Await Error Handling and Debugging Techniques Working with APIs: Fetch and JSON **Introduction to ReactJS** • React Basics: Components, JSX, Props, and State Functional vs. Class Components React Lifecycle Methods and Hooks
Module 4(Credit 1)	**Building and deploying react applications**
Learning Outcome	**After learning the modules, learners will be able to**
	1. Evaluate the concept of state in React and demonstrate how to manage state within functional and class component
	2. Develop React applications from scratch, including component creation, state management, and event handling.
	3. Create forms in React and manage form input data using controlled components with state.
	4. Implement form validation and handle form submission to process user inputs in a React application.
Content Outline	• **React State Management** Managing Component State React Context API Introduction to Redux (Optional) • **Building and Deploying React Applications** Handling Forms in React Routing in React (React Router) • **Styling React Components:**CSS Modules, Styled Components Deployment of React Applications (Notlify, Vercel, GitHub Pages)

Reference Books

1. "JavaScript and JQuery: Interactive Front-End Web Development" – Jon Duckett
2. "Eloquent JavaScript" – Marijn Haverbeke
3. "Learning React" – Alex Banks & Eve Porcello
4. "You Don't Know JS" series – Kyle Simpson
5. "React Up and Running" – Stoyan Stefanov

Assessment:

Internal Assessment 50 Marks Evaluation Scheme:

1. **UT -25**
2. **Project - 25**

The 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.

Evaluation Criteria	**Description**	**Marks**
Implementation of Web Pages	Functionality (Core Features)	5
Backend/Server-Side Logic	Code Quality (Structure, Readability)	5
UI/UX Design Principles	GUI Design & User Experience	5
Input Handling	Input Handling & Error Management	5
Code Quality and Documentation	Report writing or Documentation	5

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

1. **Basic E-Commerce Site (React + Shopping Cart)**

Objective: Develop a simple e-commerce website with product listings, cart management, and checkout.

2. **Build a quiz app where users answer questions, get instant feedback, and see their score at the end.**

External Assessment: (Marks 50)

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

Semester - IV

.4.3 Minor (Stream)

Course Title	**Machine Learning**
Course Credits	**4 Credits**
Course Outcomes	1. Understand fundamental concepts and principles of Machine Learning.
	2. Apply Python programming and libraries to implement Machine Learning algorithms.
	3. Analyze and preprocess real-world datasets using Python tools.
	4. Build, evaluate, and compare Machine Learning models.
	5. Solve real-world problems using supervised and unsupervised learning techniques.
	6. Understand challenges, limitations, and ethical considerations in Machine Learning.
Module 1 (Credit 1) Introduction to Machine Learning	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Explain the fundamentals, need, and applications of Machine Learning.
	2. Understand the general architecture and workflow of Machine Learning systems.
	3. Analyze core Machine Learning concepts such as bias, variance, and overfitting.
	4. Classify different types of Machine Learning approaches.

Content Outline	**Introduction to Machine Learning** • Definition and need of Machine Learning • Knowledge-driven vs data-driven approach • Applications of Machine Learning • Advantages, disadvantages, and challenges • General architecture of Machine Learning systems **Core Concepts in Machine Learning** • Inductive learning and generalization • Bias and variance • Overfitting and underfitting • Parametric vs non-parametric algorithms **Types of Machine Learning** • Supervised learning • Unsupervised learning • Semi-supervised learning • Reinforcement learning • Introduction to Deep Learning and Neural Networks
Module 2 (Credit 1)	
Learning Outcomes	**Classification Techniques**
	After learning the module, learners will be able to
	1. Understand classification problems and their types.
	2. Implement classification algorithms using Python.
	3. Analyze decision-making techniques in classification models.
	4. Apply Artificial Neural Networks for classification tasks.
	5. Understand classification problems and their types.
Content Outline	**Introduction to Classification**

	• Concept of classification
	• Binary classification
	• Multi-class classification
	• Multi-label classification
	
	K-Nearest Neighbour (KNN) Algorithm
	
	• Need and working of KNN
	• Distance computation methods
	• Advantages and disadvantages
	• Implementation of KNN using Python (Scikit-learn)
	
	Decision Tree Classification
	
	• Terminology and assumptions
	• Working of Decision Trees
	• ID3 Algorithm
	• Attribute selection methods:
	  ○ Entropy
	  ○ Information Gain
	  ○ Gini Impurity
	• Decision Tree implementation using Python
	
	Artificial Neural Networks (ANN)
	
	• Basic structure of ANN
	• Neuron model and activation functions
	• ANN for classification (conceptual + Python-based overview)

Module 3 (Credit 1) Regression, SVM and Model Evaluation using Python	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Explain regression techniques and their applications. 2. Implement Support Vector Machines using Python. 3. Understand probabilistic learning approaches. 4. Evaluate Machine Learning models using standard metrics.

Content Outline	**Regression Techniques** • Linear Regression • Logistic Regression • Differences between linear and logistic regression • Regression model implementation using Python **Support Vector Machines (SVM)** • Concept and working of SVM • Support vectors • Hard margin and soft margin • Kernel functions • Advantages and disadvantages • SVM implementation using Python **Probabilistic Learning** • Introduction to Bayesian learning • Bayes' theorem • Naïve Bayes classifier • Characteristics and applications • Naïve Bayes implementation using Python **Model Performance Evaluation** • Confusion matrix • Accuracy • Precision • Recall (Sensitivity) • F1-score • ROC and AUC curve • Cross-validation using Python
Module 4 (Credit 1)	**Unsupervised Learning and Dimensionality Reduction**
Learning	**After learning the module, learners will be able to**
	1. Understand the importance of unsupervised learning.
	2. Apply clustering algorithms to real-world datasets.

Outcomes	3. Analyze high-dimensional data using dimensionality reduction techniques.
	4. Implement unsupervised learning techniques using Python.
Content Outline	**Introduction to Unsupervised Learning** • Concept, importance, and challenges • Applications of unsupervised learning **Clustering Techniques** • Concept of clustering • Applications of clustering **Hierarchical Clustering** • Introduction to hierarchical clustering • Types of hierarchical clustering • Issues with hierarchical clustering • Implementation using Python **Partition-Based Clustering** • K-means clustering • Steps involved in K-means algorithm • Strengths and weaknesses of K-means • Issues in K-means clustering • Implementation using Python **Dimensionality Reduction** • Curse of dimensionality • Principal Component Analysis (PCA) • Importance of dimensionality reduction • PCA implementation using Python

Reference Books

1. Machine Learning: Concepts, Techniques and Applications, T V Geetha S, Sendhil kumar CRC Press, Taylor and Francis, 1 st Edition, 2023

2. Machine Learning for Decision Sciences with Case Studies in Python, S. Sumathi, Suresh V. Rajappa, CRC Press, Taylor and Francis, 1 st Edition, 2022

3. Introduction to Machine Learning with Python, Andreas C. Müller, Sarah Guido O'Reilly Media, Inc. 1 st Edition, 2016

4. Machine Learning for Beginners, Harsh Bhasin, BPB, 1st Edition, 2020

5. Machine Learning S Sridhar, Oxford University, Press, 1 st Edition, 2021

6. Machine Learning Ruchi Doshi, Kamal Kant Hiran, BPB, 1 St Edition, 2021

Assessment:

Internal Assessment 50 Marks Evaluation Scheme:

1. **UT -25**
2. **Project - 25**

The 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 C program, correct logic, syntax	5
Problem Understanding	Clarity of problem interpretation and correct approach	5
Output Correctness	Program compiles without errors and gives correct output	5
Code Quality, Structure & Debugging Skills	Proper use of indentation, comments, meaningful variable names, modularity. 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: Customer Churn Prediction Using Classification Algorithms

Description:
This project focuses on predicting whether a customer is likely to churn using classification algorithms such as K-Nearest Neighbour, Decision Tree, Support Vector Machine, and Naïve Bayes. The model is evaluated using performance metrics like accuracy, precision, recall, and F1-score.

Techniques Used:
KNN, Decision Tree (ID3), SVM, Naïve Bayes, Performance Evaluation Metrics

Project 2: Spam Email Detection System Description:
Students build a machine learning model to classify emails as spam or non-spam using supervised learning algorithms. Feature extraction and model evaluation are performed to analyze the effectiveness of different classifiers.

Techniques Used:
Naïve Bayes, SVM, Classification Metrics, Cross Validation

External Assessment: (Marks 50)

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

Semester - IV

.4.4 A. Open Elective Courses/ Generic (OEC)

Course Title	**Introduction to Canva**
Course Credit	**2 Credits**
Course Outcomes	1. **Apply** design principles such as color, typography, and composition to create visually compelling reports, presentations, and marketing materials.
	2. **Analyze** the effectiveness of design elements, such as layout, fonts, and branding consistency, in communicating messages across various formats.
	3. **Evaluate** the quality and impact of design work based on industry standards, visual communication strategies, and user engagement metrics.
	4. **Create** professional documents, reports, and marketing materials using Canva's templates, customization tools, and collaboration features.
Module 1 (Credit 1) Introduction to canva and design basics	
Learning Outcomes	**After learning the module, learners will be able to**
	1. **Apply** design principles such as color, typography, and composition to create visually appealing and effective designs using Canva's tools and features. 2. **Analyze** Canva's templates and customization options to select and modify designs that align with specific business needs and branding guidelines. 3. **Evaluate** the effectiveness of reports, presentations, and other design projects based on their visual appeal, clarity, and communication of key information. 4. **Create** professional business reports, including charts, tables, and infographics, using Canva's features and tools for data visualization and effective communication. 5. **Design** engaging and dynamic presentations by incorporating multimedia elements, animations, and transitions to enhance audience interaction and message delivery.

Content Outline	**Introduction to Canva & Design Basics** • Navigating the Canva interface • Understanding design principles (color, typography, and composition) • Exploring Canva's templates and customization options **Designing Reports** • Creating professional business reports • Formatting charts, tables, and infographics • Exporting and sharing reports effectively **Creating Engaging Presentations** • Designing visually compelling slides • Using animations and transitions effectively • Enhancing presentations with multimedia elements.
Module 2 (Credit 1) Marketing Material and Branding	
Learning Outcomes	**After learning the module, learners will be able to**
	1. **Apply** Canva's design tools to create marketing materials such as posters, brochures, and social media graphics that adhere to branding guidelines. 2. **Analyze** the integration of branding elements (logos, color schemes, fonts) in marketing materials to ensure visual consistency and alignment with brand identity. 3. **Evaluate** the effectiveness of digital and print marketing materials by applying best practices to assess their visual appeal, communication, and target audience engagement. 4. **Create** professional documents such as letterheads, resumes, and official materials, utilizing Canva's design and collaboration features to produce polished, high-quality outputs suitable for business and personal use.

Content Outline	**Marketing Materials & Branding** • Designing posters, brochures, and social media graphics • Incorporating branding elements (logos, color schemes, fonts) • Best practices for digital and print marketing **Document Design & Collaboration** • Creating letterheads, resumes, and official documents • Utilizing Canva's real-time collaboration features • Sharing, exporting, and printing document

External Evaluation for 50 Marks will be conducted by university.

Textbooks: -

1. "The Non-Designer's Design Book" by Robin Williams.
2. "Canva for Work: How to Design Like a Pro" by Melanie H. K.
3. "Branding: In Five and a Half Steps" by Michael Johnson
4. "Creating Documents with Canva: A Practical Guide for Beginners" by Olivia Davis

Additional Resources:

1. **Canva's own design tutorials**: Canva provides free, comprehensive tutorials on its platform, which are highly beneficial for students.
2. **Online articles and video tutorials** on YouTube or platforms like Skillshare and Udemy, focusing on using Canva for specific projects.

Required Tools:

1. **Canva Free or Pro Account** (depending on access)
2. Computer with internet connection

External Assessment: (Marks 50)

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

Semester - IV

.4.4 B. Open Elective Courses/ Generic (OEC)

Course Title	**Management Information Systems (MIS)**
Course Credits	**2 Credits**
Course outcomes	1. Evaluate the fundamental concepts and components of MIS.
	2. Analyze the role of MIS in decision-making and business strategy
	3. Evaluate database management techniques for effective information processing
	4. Apply business intelligence and data analytics for strategic insights
Module 1(Credit 1)	**Introduction to MIS**
Learning Outcomes	**After learning the module, learners will be able to**
	1. Evaluate Define MIS and its role in organizations
	2. Apply various types of information systems
	3. Evaluate database management techniques.
	4. Analyze the ethical and security challenges in MIS.
Content Outline	• **Introduction to MIS** Definition, Purpose, and Scope of MIS Role of MIS in Business and Management Components of MIS: Hardware, Software, Data, People, Processes • **Types of Information Systems** Transaction Processing Systems (TPS) Decision Support Systems (DSS) Enterprise Resource Planning (ERP) Customer Relationship Management (CRM) Business Intelligence Systems

	• **Database Management & Data Processing** Introduction to Database Management Systems (DBMS) SQL Basics and Database Queries Data Warehousing & Big Data in MIS Cloud Computing in Data Management • **Decision Support Systems (DSS) & Business Analytics** Role of DSS in Managerial Decision-Making Business Analytics and Predictive Analytics Data Visualization Techniques (Dashboards, BI Tools) • **MIS Security & Ethical Concerns** Cybersecurity Threats and MIS Security Strategies Ethical Issues in Information Management Data Privacy and Compliance (GDPR, HIPAA)
Module 2(Credit 1)	**Cloud Computing and Emerging Technologies**
Learning Outcomes	1. Create enterprise information systems using MIS principles.
	2. Apply data analytics for business decision-making
	3. Evaluate IT governance and risk management in MIS
	4. Create cloud computing and emerging technologies in MIS
Content Outline	• **Enterprise Systems & IT Governance** Introduction to Enterprise Information Systems (EIS) IT Governance Frameworks (COBIT, ITIL) Risk Management in Information Systems • **Cloud Computing & Emerging Technologies** Cloud-Based MIS Solutions AI, Machine Learning, and Automation in MIS Blockchain in Information Systems • **Business Intelligence & Data Analytics** Business Intelligence Tools (Power BI, Tableau) Data Mining and Machine Learning in MIS Case Studies on Data-Driven Decision Making, • **Strategic Role of MIS in Organizations** MIS for Competitive Advantage Digital Transformation and Business Innovation Future Trends in MIS

External Evaluation for 50 Marks will be conducted by university.

References:

1. **"Management Information Systems"** – Kenneth C. Laudon & Jane P. Laudon
2. **"Business Intelligence: A Managerial Perspective on Analytics"** – Ramesh Sharda, Dursun Delen
3. **"Database Management Systems"** – Raghu Ramakrishnan
4. **"IT Strategy for Business"** – Bernard Marr
5. **"Information Systems for Managers"** – Gabriele Piccol

External Assessment: (Marks 50)

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

Semester - IV

.4.4 C. Open Elective Courses/ Generic (OEC)

Course Title	**Introduction to Entrepreneurship**
Course Credits	**2 Credits**
Course Outcomes	1. Understand the Concept and Role of Entrepreneurship
	2. Identify and Develop Entrepreneurial Traits and Business Ideas
	3. Apply Knowledge to Launch and Manage a Startup
	4. Analyze Institutional Support and Contemporary Trends
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Explain Key Concepts of Entrepreneurship
	2. Demonstrate Entrepreneurial Traits and Skills
	3. Develop a Framework to Start and Manage a Business
Content Outline	• **Introduction to Entrepreneurship** • **Concept and Meaning of Entrepreneurship** Definition and evolution Characteristics and need for entrepreneurship • **Types and Classifications of Entrepreneurs** Innovative, Imitative, Drone, Fabian Social, Serial, Women, and Corporate Entrepreneurs • **Entrepreneur vs Manager vs Intrapreneur** Roles, differences, and similarities • **Importance of Entrepreneurship** Economic and social contributions Role in job creation, innovation, and self-reliance • **Entrepreneurship in Indian Context** Historical background Present trends and future scope

Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Explain key motivational theories relevant to
	2. Identify and assess entrepreneurial traits and competencies
	3. Use creative thinking techniques
	4. Analyze the business environment
Content Outline	• **Entrepreneurial Motivation** McClelland's Theory of Need for Achievement (nAch) Maslow's Hierarchy of Needs • **Entrepreneurial Traits and Competencies** Creativity, innovation, leadership, risk-taking Competency mapping • **Idea Generation Techniques** Brainstorming, mind mapping, design thinking Opportunity identification and assessment • **Feasibility Study and Business Plan Development** Elements of a business plan Market research and demand analysis Project appraisal (technical, financial, social)

Reference Books:

1. Entrepreneurship Development Author: S. S. Khanka Publisher: S. Chand Publishing
2. Entrepreneurship and Small Business Management Author: C.B. Gupta and N.P. Srinivasan Publisher: Sultan Chand & Sons
3. Entrepreneurship Development,Author: Vasant Desai Publisher: Himalaya Publishing House.
4. Dynamics of Entrepreneurial Development and Management,Author: Vasant Desai Publisher: Himalaya Publishing House

Assessment:

External Assessment: (Marks 50)

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

Semester - IV

.4.5 Skill Enhancement Courses (SEC)

Course Title	**Introduction to Microprocessor 8086**
Course Credits	**2 Credits**
Course Outcomes	**After Completion of this Course, students will be able**
	1. Evaluate key features of microprocessors, including their architecture, processing capabilities, speed, and integration level.
	2. Apply various types of instructions with diagrams.
	3. Create various assembly language programs by using different types of instructions.
	4. Evaluate Advanced microprocessor Features, functions.
	5. Evaluate interrupts, DMA controllers etc features, functions.
Module 1 (Credit 1) Introduction to Microprocessor	
Learning Outcomes	1. Evaluate features, functions and Operations of Microprocessor.
	2. Evaluate the significance of the 8086 microprocessors.
Content Outline	• **Introduction to Microprocessor:** What is Microprocessor, Features, Application of Microprocessor. Functions and Operations performed by Microprocessor. • **8086 Microprocessor:** Introduction, Pin diagram, Architecture, Memory Segmentation, Flag registers, Registers used in 8086.Signals Description of 8086, Interrupts of 8086. • **Programming of 8086:** Addressing Mode of 8086, Instruction Set of 8086, Assembly language Programming of 8086, Simple programs.
Module 2 (Credit 1) Introduction to Advance Microprocessor	
Learning Outcomes	1. Evaluate purpose of Interrupts and USART Functionality & DMA controller
	2. Evaluate Advance Microprocessor for the 8088.

Content Outline	• **Interrupts:** Purpose of interrupts, Hardware and Software Interrupts, Interrupt vectors. 8259-Interrupt & its organization. • **Introduction to Programmable Peripheral Interface & USART:** 8255 PPL & USART 8251. • **Direct Memory Access:** DMA controller 8257 & its organization**.** • **Introduction to Advance Microprocessor:** 8088, 80386, 80486, Pentium (80586, II, III, VI) Microprocessor & features. Difference between 8085, 8086 & 8088.

References:

1. Microprocessor and Digital Systems by D.V. Hall.
2. 16 bit Microprocessor by Triebel and A. Singh.
3. Advanced Microprocessors and Peripherals **by** A.K. Ray & K.M. Bhurchandi.
4. The 8086 Microprocessor: Programming & Interfacing the PC by Kenneth J. Ayala.
5. Microprocessors and Interfacing: Programming and Hardware by Douglas V. Hall
6. Microprocessor Architecture Programming ~ Application, with 8080/8085 by Ramesh S. Gaonkar.
7. The Intel Microprocessors: 8086/8088, 80186/80286, 80386, 80486, Pentium by Barry B. Brey.

Assessment:

External Assessment: (Marks 50)

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

Semester - IV

.4.7 Minor Stream

Course Title	**Python for Machine Learning**
Course Credit	**2 Credits**
Course Outcomes	1. Develop logical solutions using Python programming.
	2. Apply fundamental Python concepts to solve real-world problems.
	3. Implement control structures, functions, and data structures.
	4. Analyze, debug, and optimize Python programs.
	5. Develop basic applications using Python and its standard libraries.
Module 1 (Credit 1) Python Fundamentals & Control Structures	
Learning Outcomes	**After learning the module, learners will be able to:**
	1. Understand Python syntax and dynamic typing.
	2. Implement control structures and loops.
	3. Use functions and modules for structured programming.
	4. Apply basic file handling concepts.
Course Outline	**Python Basics** • Introduction to Python • Variables and dynamic typing • Data types and type conversion • Operators and expressions **Control Flow** • Conditional statements: if, if-else, nested if • Looping statements: for, while • break, continue, pass

	Functions & Modules • User-defined functions • Functions with arguments and return values • Default and keyword arguments • Lambda functions • Importing and using modules (math, random – conceptual) **Basic File Handling** • Reading and writing text files • File operations using read(), readline(), readlines() • Introduction to CSV file handling for datasets
Module 2 (Credit 2) Data Structures, OOP & & ML-Oriented Applications	
Learning Outcomes	1. Use Python data structures effectively. 2. Implement object-oriented programming concepts. 3. Handle runtime errors using exception handling. 4. Understand basic web, automation, and API concepts.

Course Outline	**Python Data Structures** • Lists, Tuples • Sets and Dictionaries • Basic operations and use cases **Object-Oriented Programming** • Classes and Objects • Inheritance • Polymorphism (method overriding – conceptual) **Exception Handling** • try–except blocks • Handling multiple exceptions • Error handling in file operations **Introduction to Applications (Conceptual)** • JSON and CSV file handling • Role of Python in Machine Learning • Introduction to ML libraries (NumPy, Pandas, Scikit-learn – overview only) • Basic Machine Learning workflow (data → model → prediction – conceptual) • Python for automation, testing, and debugging • Version control basics: Git & GitHub (conceptual)

Internal Assessment – 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.pert 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**
Core Functionality & Feature Coverage	Functionality (Core Features)	10

Code Structure & Best Practices	Code Quality (Structure, Readability)	10
User Interface & Design Consistency	GUI Design & User Experience	10
Input Validation & Exception Handling	Input Handling & Error Management	10
Project Report & Technical Documentation	Report writing or Documentation	10

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

1. **Simple Console-Based To-Do List (Basic)**

Uses **Python lists** to store tasks.

Users can **add, remove, and mark tasks as completed** via command-line input.

Saves tasks to a **text file** for persistence.

Ideal for beginners.

Tech : Python (with file handling)

Libraries: None (pure Python)

2. **GUI-Based ATM Simulation (Intermediate)**

Uses **Tkinter or PyQt** for a user-friendly **ATM interface**. Features **buttons for**

deposits,

withdrawals, and balance checks.

Stores user data in **SQLite or JSON files** for persistence.

Tech: Python + Tkinter/PyQt

Libraries: Tkinter/PyQt, SQLite3, JSON

Course Syllabus

Semester – V

.5.1 Major (Core)

Course Title	**Computer Networks**
Course Credits	**4 Credits**
Course Outcomes	1. Explain fundamental networking concepts and architecture.
	2. Demonstrate knowledge of data communication techniques and protocols.
	3. Configure and implement basic network addressing and subnetting schemes.
	4. Analyze network performance and identify bottlenecks and issues.
	5. Evaluate network security mechanisms and propose mitigation strategies.
Module 1 (Credit 1) Networking Fundamentals & Models *(Builds strong conceptual foundation)*	
Learning Outcomes	1. Students understand how networks are structured and how data flows layer-by-layer.

Course Outline	• Introduction to Computer Networks
	  ○ Definition, goals, and applications
	  ○ Network components: hosts, links, switches, routers
	
	• Types of Networks
	  ○ LAN, MAN, WAN, PAN
	  ○ Client–Server vs Peer-to-Peer
	
	• Network Topologies
	  ○ Bus, Star, Ring, Mesh, Hybrid
	
	• Network Models
	  ○ OSI Model – all 7 layers with functions
	  ○ TCP/IP Model – layers and comparison with OSI
	
	• Transmission Media
	  ○ Guided: Twisted Pair, Coaxial, Fiber Optic
	  ○ Unguided: Radio, Microwave, Satellite
	
	• Switching Techniques
	  ○ Circuit Switching, Packet Switching, Message Switching

Module 2 (Credit 1) Data Link & Network Layer Protocols *(How data moves reliably and logically across networks)*	
Learning Outcomes	1. Students learn IP addressing, routing basics, and error handling, which are critical for networking jobs and certifications.

Course Outline	• Data Link Layer
	  ○ Framing techniques
	  ○ Error detection and correction
	  ▪ Parity, CRC, Hamming Code
	  ○ Flow control
	  ▪ Stop-and-Wait, Sliding Window
	  ○ Multiple Access Protocols
	  ▪ CSMA/CD, CSMA/CA
	
	• Network Layer
	  ○ Logical addressing
	  ▪ IPv4 addressing, subnetting, CIDR
	  ▪ Introduction to IPv6
	  ○ Routing concepts
	  ▪ Static vs Dynamic routing
	  ○ Routing Algorithms
	  ▪ Distance Vector (RIP)
	  ▪ Link State (OSPF – concept)
	
	• Internetworking devices
	  ○ Hub, Switch, Router, Gateway

Module 3 (Credit 1) Transport, Application Layer & Network Services *(Real-world protocols students actually use)*	
Learning Outcomes	1. Students understand how web, email, and file transfer systems work behind the scenes.

Course Outline	• Transport Layer
	  ○ TCP vs UDP
	  ○ TCP features: flow control, congestion control
	  ○ Ports and sockets
	
	• Application Layer Protocols
	  ○ HTTP / HTTPS
	  ○ FTP, SFTP
	  ○ SMTP, POP3, IMAP
	  ○ DNS – working and resolution process
	
	• Network Services
	  ○ DHCP
	  ○ NAT
	  ○ Proxy Servers
	
	• Performance Metrics
	  ○ Bandwidth, Throughput, Latency, Jitter

Module 4 (Credit 1) **Modern Networking, Security & Emerging Trends** *(Latest topics + industry relevance)*	
Learning Outcomes	1. Students get exposure to current technologies used in cloud, wireless, and secure networks, preparing them for industry & higher studies.

Course Outline	• Network Security Basics
	  ○ Threats: sniffing, spoofing, DoS/DDoS
	  ○ Firewalls: packet filtering, stateful inspection
	  ○ VPN concepts
	  ○ Basics of encryption (SSL/TLS overview)
	
	• Wireless & Mobile Networks
	  ○ Wi-Fi standards (802.11 a/b/g/n/ac/ax)
	  ○ Bluetooth, NFC
	  ○ Cellular networks overview: 4G & 5G basics
	
	• Modern Networking Concepts
	  ○ Software Defined Networking (SDN) – concept
	  ○ Network Function Virtualization (NFV)
	  ○ Cloud Networking basics (VPC, virtual networks)
	
	• Emerging Trends
	  ○ IoT networking
	  ○ Edge Computing
	
	• Introduction to AI in network management

Reference Books

1. **Computer Networking: A Top-Down Approach**
   James F. Kurose, Keith W. Ross – Pearson Education
   (Highly recommended for application-layer to physical-layer understanding)

2. **Data Communications and Networking**
   Behrouz A. Forouzan – McGraw-Hill
   (Excellent for OSI, TCP/IP models, protocols, and fundamentals)

3. **Computer Networks** – Andrew S. Tanenbaum, David J. Wetherall, Pearson

4. **Internetworking with TCP/IP (Vol. 1)** – Douglas E. Comer, Pearson

5. **Computer Networking: Principles, Protocols and Practice** – Olivier Bonaventure

6. **Network Security Essentials** – William Stallings, Pearson

7. **Wireless Communications & Networks** – William Stallings

8. **Cloud Computing Concepts, Technology & Architecture** – Thomas Erl

Internal Assessment 50 Marks Evaluation Scheme:

1. **UT -25**
2. **Project - 25**

The 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	Clear understanding of networking concepts, protocols, models, and architectures used in the project	5
Problem Understanding	Correct use of networking concepts such as IP addressing, routing, protocols, security mechanisms, or simulations	5
Output Correctness	Ability to analyze networking problems, identify bottlenecks, and propose logical solutions	5
Code Quality & Structure	Understanding of basic network security threats and suitable mitigation strategies	5
Debugging Skills	Understanding of basic network security threats and suitable mitigation strategies	5

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

Project 1: Small Office Computer Network Design

Design a basic network for a small office using appropriate topology, devices, and IP addressing. Explain data flow using OSI and TCP/IP models.

Project 2: Study of Network Protocols and Security

Study common network protocols (TCP, UDP, HTTP, DNS) and basic network security threats with simple diagrams and explanations.

External Assessment: (Marks 50)

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

Semester – V

.5.2 Major (Core)

Course Title	**Software Engineering**
Course Credits	**4 Credits**
Course Outcomes	1. Explain basic principles, characteristics, and the need for Software Engineering.
	2. Understand and compare different Software Development Life Cycle (SDLC) models.
	3. Analyze software requirements and apply appropriate design concepts
	4. Use UML diagrams to model software systems
	5. Describe software testing strategies, maintenance activities, and quality assurance practices.
	6. Understand software project management fundamentals and emerging trends in software engineering.
Module 1 (Credit1) Introduction to Software Engineering	
Learning Outcomes	Students will be able to: 1. Explain the principles, characteristics, and need of Software Engineering. 2. Understand ethical issues in software development. 3. Describe SDLC and its importance. 4. Compare different SDLC models. 5. Identify functional and non-functional requirements. 6. Understand the purpose and structure of SRS.
Course Outline	1. Introduction to Software Engineering 2. Characteristics of Software 3. Software Engineering Ethics 4. Software Development Life Cycle (SDLC) 5. SDLC Models: • Waterfall Model • Incremental Model • Spiral Model (Overview) • Agile Model (Introduction) 6. Software Requirements: • Functional and Non-functional Requirements • Software Requirement Specification (SRS)

Module 2 (Credit 1) Software Design, Testing and Maintenance	
Learning Outcomes	Students will be able to: 1. Explain software design concepts. 2. Differentiate architectural and detailed design. 3. Understand basic UML diagrams. 4. Describe software testing levels and approaches. 5. Explain software maintenance activities. 6. Understand Software Quality Assurance.
Course Outline	1. Software Design Concepts 2. Types of Design: • Architectural Design • Detailed Design 3. UML Diagrams (Introduction): • Use Case Diagram • Class Diagram (Basic) 4. Software Testing: • Levels of Testing • Black Box and White Box Testing 5. Software Maintenance 6. Software Quality Assurance (SQA)
Module 3 (Credit1) Advanced Software Engineering & Project Management	
Learning Outcomes	**Students will be able to:** 1. Understand software project planning and estimation. 2. Apply basic project management concepts. 3. Identify software risks and mitigation strategies. 4. Understand configuration and version management.

Course Outline	1. Software Project Management • Software project characteristics • Project planning and scheduling 2. Software Cost Estimation • LOC and Function Point (Conceptual) 3. Risk Management • Types of risks • Risk identification and mitigation • 4. Software ConfigurationManagement • Version control concepts • Change management 5. Introduction to CASE Tools
Module 4 (Credit 1) Modern Software Engineering Practices & Emerging Trends	
Learning Outcomes	**Students will be able to:** 1. Understand modern development practices. 2. Describe DevOps and continuous integration concepts. 3. Analyze quality and reliability of software systems. 4. Understand current trends and future directions in Software Engineering.
Course Outline	1. Agile Software Development (Scrum overview) 2. DevOps Concepts • Continuous Integration • Continuous Deployment (CI/CD – overview) 3. Software Reliability and Metrics 4. Software Reusability and Component-Based Development 5. Emerging Trends: • Cloud-based software development • Microservices architecture (conceptual) • AI-assisted software development 6. Ethical, Legal, and Social Issues in Software Engineering

Reference Books

1. **Ian Sommerville** — *Software Engineering*, Pearson Education.
   (Widely used, clear explanation of fundamentals and practices.)

2. **Roger S. Pressman & Bruce R. Maxim** — *Software Engineering: A Practitioner's Approach*, McGraw-Hill.
   (Classic comprehensive text with examples and case studies.)

3. **Pankaj Jalote** — *An Integrated Approach to Software Engineering*, Springer.

4. **K.K. Aggarwal & Yogesh Singh** — *Software Engineering*, New Age International.

5. **Ali Behforooz & Frederick J. Hudson** — *Software Engineering Fundamentals*, Oxford University Press.

6. **Bernd Bruegge & Allen H. Dutoit** — *Object-Oriented Software Engineering*, Pearson.

7. **R. Mall** — *Fundamentals of Software Engineering*, PHI Learning.

Internal Assessment 50 Marks Evaluation Scheme:

1. **UT -25**
2. **Project - 25**

The 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**
Conceptual Understanding	Clear understanding of Software Engineering principles, SDLC, design concepts, testing, and maintenance.	5
Technical Implementation & Accuracy	Correct application of concepts (e.g., SRS, UML, test plan, design diagrams) and logical structure of deliverables.	5
Problem Analysis & Solution Design	Ability to analyze requirements, propose suitable design and methodologies, identify constraints and justify decisions.	5
Software Quality Awareness	Demonstrates understanding of testing strategies, maintenance implications, and quality assurance considerations.	5
Communication & Documentation	Report and presentation clarity; including diagrams, tables, proper formatting, screenshots (if any), and professional delivery.	5

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

Project 1: Library Management System (SRS + Design + Test Plan)

Description:
Create a Software Requirement Specification (SRS), UML diagrams, and a basic test plan for a Library Management System. This project focuses on requirement analysis, design,

and testing documentation rather than coding.

Project 2: Online Examination System – Design & Quality Plan Description:
Develop a design documentation and quality assurance plan for an online examination application. Focus is on design decisions and quality/testing strategies.

External Assessment: (Marks 50)

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

Semester – V

.5.3 Indian Knowledge Systems (IKS)

Course Title	**Indian Knowledge Systems and Ethical Practices in IT**
Course Credits	**2 Credits**
Course Outcomes	1. Fundamental principles of Indian Knowledge Systems (IKS), including Dharma, Karma, Ahimsa, and Lokasangraha, and relate them to ethical practices in IT and computing.
	2. Ancient Indian contributions to mathematics, logic, astronomy, linguistics, and traditional technology, and relate their significance to modern computing concepts and responsible technological development.
Module 1 (Credit 1) Foundations of Indian Knowledge Systems (IKS) *(Philosophical and cultural roots of Indian knowledge)*	
Learning Outcomes	1. Students understand Indian philosophical values and their relevance to ethical thinking in IT.
Course Outline	1. Explain the foundational principles of Indian Knowledge Systems (IKS) **such as *Dharma* (duty), *Karma* (action), and *Pramana* (means of valid knowledge), and relate these to ethical reasoning in computing and technology.** 2. Analyze ethical challenges in IT and computing (e.g., data privacy, algorithmic bias, and digital governance) using traditional Indian ethical concepts such as ***Lokasangraha*** (welfare of society) and ***Ahimsa*** (non-violence). 3. Apply Indian moral values and ethical frameworks to formulate responsible technology practices that promote societal well-being, fairness, and accountability in IT projects. 4. Evaluate contemporary technology issues (e.g., AI ethics, cybersecurity policies, and user privacy) in light of constitutional, cultural, and moral values emphasized by Indian ethical traditions. 5. Design guidelines and strategies for ethical decision-making in IT organizations that reflect both global standards (e.g., fairness, transparency) and Indian philosophical insights about duty and societal harmony.
Module 2 (Credit 1) Indian Contributions to Science, Mathematics & Computing	

(India's legacy in scientific and logical thinking)	
Learning Outcomes	1. Students appreciate India's intellectual heritage and its influence on modern computing concepts.

	• **Ancient Indian Mathematics**
	  ○ Zero, decimal system, place value
	  ○ Contributions of Aryabhata, Brahmagupta, and Bhaskaracharya
	• **Indian Logic and Algorithms**
	  ○ Nyaya and Vaisheshika schools
	  ○ Algorithmic thinking in ancient texts
	• **Astronomy & Timekeeping**
	  ○ Indian calendars, eclipses, planetary motion
	• **Linguistics & Computing**
	  ○ Panini's grammar as a precursor to formal languages
	  ○ Sanskrit and structured programming logic
	• **Traditional Engineering & Technology**
	  ○ Metallurgy (Iron Pillar of Delhi)
	  ○ Water management systems

Suggested Pedagogy

- Case-study based learning
- Group discussions & debates
- Reflection essays
- Guest lectures from ethics / IKS experts

External Assessment: (Marks 50)

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

Semester – V

.5.4 Minor (Stream)

Course Title	**Deep Learning**
Course Credits	**4 Credits**
Course Outcomes	1. Understand the core principles of deep learning and artificial neural networks.
	2. Build and optimize deep learning models for various applications.
	3. Implement convolutional and recurrent architectures for structured and unstructured data.
	4. Train and fine-tune deep generative models for feature extraction and synthesis.
	5. Apply deep learning techniques to solve real-world problems in vision, NLP, and generative AI.
	6. Evaluate model performance using metrics such as accuracy, precision, recall, and loss functions.
	7. Deploy deep learning models efficiently on cloud or edge devices.
	8. Understand the ethical implications of AI and deep learning applications.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Explain the evolution, significance, and key applications of deep learning across various industries.
	2. Apply fundamental mathematical concepts from linear algebra, including vectors, matrices, tensors, eigenvalues, norms, and matrix operations, in deep learning models.
	3. Understand numerical computation issues, such as overflow, underflow, conditioning, and algorithmic stability.
	4. Describe the structure and working of neural networks, including neurons, activation functions, and loss functions.

	5. Explain the training process of neural networks, including forward and backward propagation and gradient descent methods.
	6. Analyze overfitting and underfitting and apply regularization techniques such as L1, L2, and dropout to improve model performance.
Content Outline	• **Foundations of Deep Learning Introduction:** Evolution and significance of deep learning, Key applications across industries • **Mathematical Foundations: Linear Algebra:** Scalars, Vectors, Matrices, and Tensors, Matrix operations: Multiplication, Identity, and Inverses, Eigenvalues and Eigenvectors, Norms and special matrices • **Numerical Computation:** Issues of overflow and underflow, Conditioning and stability in algorithms • **Neural Networks Basics:** Structure and function of neurons, Activation functions: Sigmoid, Tanh, ReLU, and variants, Loss functions: MSE, Cross- Entropy • **Training Neural Networks:** Forward and backward propagation, Gradient Descent and its variants, Overfitting and underfitting, Regularization techniques: L1, L2, Dropout
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Explain deep feedforward network architectures, training challenges, and weight initialization strategies.
	2. Apply regularization techniques such as batch normalization and dropout to improve deep network training and generalization.
	3. Understand optimization techniques for deep networks, including advanced optimizers (RMSprop, Adam), learning rate scheduling, and methods to handle vanishing and exploding gradients.

	4. Describe the working principles of Convolutional Neural Networks (CNNs)**,** including convolution, feature maps, pooling, and popular architectures like LeNet, AlexNet, VGG, and ResNet.
	5. Explain sequence modeling using Recurrent Neural Networks (RNNs) and advanced variants such as LSTM, GRU, and bidirectional RNNs.
	6. Understand attention mechanisms and their role in improving performance for sequence-based deep learning applications.
Content Outline	• **Advanced Neural Network Architectures** • **Deep Feedforward Networks:** Architecture and training challenges, Weight initialization strategies • **Regularization Techniques:** Batch Normalization, Dropout and its impact on training • **Optimization for Deep Networks**: Advanced optimizers: RMSprop, Adam, Learning rate scheduling, Handling vanishing and exploding gradients • **Convolutional Neural Networks (CNNs):** Convolution operations and feature maps Pooling mechanisms, Popular CNN architectures: LeNet, AlexNet, VGG, ResNet • **Recurrent Neural Networks (RNNs):** Sequence modeling and applications, Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRUs), Bidirectional RNNs and attention mechanisms
Module 3 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Explain autoencoders and representation learning techniques**,** including undercomplete, sparse, denoising, and variational autoencoders.

	2. Understand generative modeling concepts, including Generative Adversarial Networks (GANs), conditional GANs, and their practical applications.
	3. Describe advanced sequence modeling techniques, including advanced RNN architectures, transformer models, and attention mechanisms.
	4. Explain the fundamentals of deep reinforcement learning, including Deep Q-Networks and policy gradient methods.
	5. Apply deep learning models to real-world applications in natural language processing, computer vision, and speech processing.
	6. Analyze the suitability of specialized deep learning models for different problem domains and applications.
Content Outline	• **Specialized Deep Learning Models and Applications** • **Autoencoders and Representation Learning:** Undercomplete and sparse autoencoders, Denoising and variational autoencoders • **Generative Models:** Generative Adversarial Networks (GANs), Conditional GANs and applications • **Sequence Modeling**: Advanced RNN architectures, Transformer models and attention mechanisms • **Deep Reinforcement Learning:** Fundamentals of reinforcement learning, Deep Q-Networks and policy gradients • **Applications of Deep Learning:** Natural Language Processing: Text generation, translation, Computer Vision: Image classification, object detection, Speech Recognition and synthesis
Module 4 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	1. Explain ethical issues in deep learning, including bias, fairness, interpretability, and explainability of AI models.

	2. Analyze the societal and technical impact of ethical challenges in the design and deployment of deep learning systems..
	3. Understand scalability and deployment considerations, including model optimization and hardware requirements such as GPUs and TPUs.
	4. Evaluate real-world deep learning case studies to identify best practices and lessons learned from industry applications.
	5. Discuss emerging trends and future directions in deep learning, including new architectures, quantum computing, and neuromorphic computing
Content Outline	• **Ethics and Practical Considerations in Deep Learning** • **Ethical Considerations:** Bias and fairness in AI models, Interpretability and explainability • **Scalability and Deployment:** Modeloptimization for deployment, Hardware considerations: GPUs, TPUs • **Case Studies:** Analysis of real-world deep learning applications, Lessons learned from industry implementations • **Future Directions:** Emerging Architectures, Quantum computing and deep learning, Neuromorphic computing

Reference Books

1. Goodfellow, Ian, et al. Deep Learning. MIT Press, 2016.

2. Chollet, François. Deep Learning with Python. Manning Publications, 2018.

3. Géron, Aurélien. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow. O'Reilly Media, 2019.

4. Nielsen, Michael A. Neural Networks and Deep Learning. Determination Press, 2015.

5. Aggarwal, Charu C. Neural Networks and Deep Learning: A Textbook. Springer, 2018

Internal Assessment 50 Marks Evaluation Scheme:

1. **UT -25**
2. **Project - 25**

The 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**
Problem Understanding	Clarity in understanding the problem statement, correct identification of objectives, dataset selection, and suitability of deep learning approach	5
Model Implementation	Correct implementation of deep learning model (CNN / RNN / Autoencoder / GAN / Transformer), appropriate architecture, and correct use of libraries	5
Training & Output Correctness	Proper training of the model, correct execution without errors, valid outputs, and meaningful results	5
Model Evaluation & Analysis	Use of appropriate evaluation metrics (accuracy, loss, precision, recall, etc.), interpretation of results, and comparison of outcomes	5
Code Quality & Documentation	Clean, well-structured code with proper indentation, comments, modularity, and reproducibility	3
Project Report / Lab Record	Completeness of report, clarity of explanation, formatting, screenshots/outputs, and timely submission	2

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

Project 1: Image Classification Using Convolutional Neural Networks

Description:
This project focuses on building and training a Convolutional Neural Network (CNN) for image classification tasks using standard datasets such as CIFAR-10 or MNIST. Students analyse the impact of different CNN architectures and regularization techniques on model performance.

Techniques Used:
CNN, Convolution and Pooling Layers, Dropout, Batch Normalization, Performance Metrics.

Project 2: Sentiment Analysis Using Recurrent Neural Networks Description:
Students develop a deep learning model to perform sentiment analysis on text data such as movie reviews or social media posts. The project uses Recurrent Neural Networks (RNNs) and LSTM/GRU architectures to capture sequential patterns in text.

Techniques Used:
RNN, LSTM / GRU, Text Preprocessing, Accuracy and Loss Evaluation

External Assessment: (Marks 50)

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

Semester – V

.5.5 Minor (Stream)

Course Title	**AI / ML Use Cases**
Course Credits	**4 Credits**
Course Outcomes	**After Completion of this Course, students will be able**
	1. Understand real-world applications of Artificial Intelligence and Machine Learning.
	2. Identify appropriate ML techniques for different problem domains.
	3. Analyze datasets and problem statements to select suitable AI/ML models.
	4. Build, evaluate, and interpret ML solutions for practical use cases.
	5. Apply AI/ML concepts in domains such as healthcare, finance, education, retail, and smart systems.
	6. Understand ethical, social, and deployment challenges in AI/ML systems.
Module 1(Credit 1) Introduction to AI & ML Applications	
Learning Outcomes	**After learning the module, learners will be able to:**
	1. Explain the scope and evolution of AI and ML. 2. Differentiate between AI, ML, and Deep Learning. 3. Identify real-world AI/ML use cases across industries.

Content Outline	**Introduction to Artificial Intelligence** • Definition and scope of AI • Strong AI vs Weak AI **Machine Learning Overview** • Supervised, Unsupervised, Reinforcement Learning • ML workflow (data → model → evaluation → deployment) **AI/ML vs Traditional Programming** **Applications of AI/ML** • Healthcare • Finance • Retail & E-commerce • Education • Smart Cities **Challenges in AI/ML** • Data quality • Bias and fairness • Interpretability
Module 2 (Credit 1) Machine Learning Use Cases – Supervised Learning	
Learning Outcomes	**After learning the module, learners will be able to:**
	1. Analyze supervised learning problems. 2. Apply classification and regression techniques to real-world use cases. 3. Interpret model outputs for decision-making.

Content Outline	**Classification Use Cases** • Spam detection • Credit risk assessment • Disease diagnosis • Customer churn prediction **Regression Use Cases** • House price prediction • Sales forecasting • Demand prediction **Common Algorithms (Conceptual)** • Linear & Logistic Regression • Decision Trees • k-Nearest Neighbors • Naïve Bayes **Model Evaluation** • Accuracy, Precision, Recall, F1-Score • Confusion Matrix
Module 3 (credit 1) Machine Learning Use Cases – Unsupervised & Intelligent Systems	
Learning Outcomes	**After learning the module, learners will be able to:**
	1. Apply unsupervised learning techniques for pattern discovery. 2. Understand recommendation and anomaly detection systems. 3. Analyze intelligent system behavior.

Content Outline	**Clustering Use Cases** • Customer segmentation • Market basket analysis • Student performance grouping **Anomaly Detection** • Fraud detection • Network intrusion detection **Recommendation Systems** • Content-based recommendation • Collaborative filtering (conceptual) **Intelligent Systems** • Chatbots (conceptual) • Virtual assistants • Smart recommendation engines
Module 4 (Credit 1)Advanced AI/ML Use Cases, Ethics & Deployment	
Learning Outcomes	1. Understand advanced AI/ML applications. 2. Evaluate ethical and societal impacts of AI. 3. Analyze challenges in deploying AI/ML systems.
Content Outline	1. **Advanced AI/ML Use Cases** • Autonomous vehicles (overview) • Facial recognition systems • Predictive maintenance • Smart healthcare systems 2. **Model Deployment Concepts** • ML lifecycle • Model monitoring and updates 3. **Ethical and Social Issues** • Bias and fairness • Privacy and data security • Responsible AI 4. **Future Trends** • Explainable AI (XAI) • Generative AI (overview) • AI governance

Text Books/Reference Books

1. **Tom M. Mitchell** – *Machine Learning* McGraw-Hill Education
   (A classic foundational text covering ML concepts and algorithms.)

2. **Christopher M. Bishop** – *Pattern Recognition and Machine Learning* Springer
   (Statistical view of machine learning algorithms with real use case discussions.)

3. **Aurélien Géron** – *Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow* O'Reilly Media
   (Practical approach to ML with Python — excellent for case studies.)

4. **Charu C. Aggarwal** – *Machine Learning for Text* Springer
   (Covers ML techniques with applications; useful for real-world use cases.)

5. **Russell & Norvig** – *Artificial Intelligence: A Modern Approach* Pearson Education
   (AI concepts and applications across domains.)

Assessment

Internal Assessment 50 Marks Evaluation Scheme:

1. **UT -25**
2. **Project - 25**

The 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 Criteria	**Description**	**Marks**
Core Functional Implementation	Correct identification and implementation of AI/ML use case	5
Problem Understanding & Model Selection	Appropriate ML technique selection	5
Solution Design & Interpretation	Result analysis and interpretation	5
Input Handling & Error Management	Data validation and robustness	5
Project Report & Documentation	Proper documentation and presentation	5

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

Project 1: Disease Prediction System

- Predict disease based on patient data
- Techniques: Classification models

Project 2: Customer Churn Prediction

- Identify customers likely to leave a service
- Techniques: Logistic Regression, Decision Tree

External Assessment: (Marks 50)

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

Semester – V

.5.6 (VSC-S4)

Course Title	**Mobile Application Development using Android**
Course Credits	**2 Credits**
Course Outcomes	1. Understand the fundamentals of Dart programming and Flutter framework for mobile app development.
	2. Design and develop responsive and interactive mobile applications using Flutter widgets.
	3. Integrate APIs, databases, and external services into mobile applications.
	4. Debug, test, and deploy mobile applications for Android platforms efficiently.
	5. Apply best practices in UI/UX design, state management, and performance optimization for mobile apps.
Module 1 (Credit 1) Introduction to Mobile App Development & Dart Basics ***(Strong foundation for Flutter development)***	
Learning Outcomes	1. Students understand Flutter fundamentals and can write basic Dart programs.

Course Outline	• Introduction to Mobile Application Development
	  ○ Native vs Hybrid vs Cross-platform apps
	  ○ Overview of Android ecosystem
	• Introduction to Flutter
	  ○ Flutter architecture
	  ○ Advantages of Flutter
	• Development Environment Setup
	  ○ Flutter SDK
	  ○ Android Studio / VS Code
	• Dart Programming Fundamentals
	  ○ Variables, data types
	  ○ Functions & arrow functions
	  ○ Control statements
	  ○ OOP concepts in Dart
	  ○ Null safety

Module 2 (Credit 1) Flutter UI Design & Navigation ***(Building beautiful and responsive interfaces)***	
Learning Outcomes	1. Students can design interactive, responsive Android UIs using Flutter.

Course Outline	• Flutter Widgets • Stateless vs Stateful widgets • Widget tree & build method • Layouts & UI Design • Row, Column, Stack • Container, Padding, Expanded • Material Design in Flutter • Scaffold, AppBar, Themes • User Interaction • Buttons, forms, input fields • Gesture detection • Navigation • Named routes • Bottom navigation & drawer

Reference Books

1. Eric Windmill – *Flutter in Action* Manning Publications
(Excellent practical guide for Flutter fundamentals and UI development)

2. Alessandro Biessek – *Beginning Flutter: A Hands-On Guide to App Development* Apress
(Good for beginners with Dart + Flutter concepts)

3. Marco L. Napoli – *Flutter Cookbook* O'Reilly Media
(Widget-based solutions and UI patterns)

Assessment:

Internal Assessment – 50 Marks

Internal assessment will be conducted through Mini Project Development, Project Presentation, and Report Submission.

The Rubric will have the following Evaluation Parameters:		
Evaluation Criteria	**Description**	**Marks**
Core Functionality	Implementation of main features, logic correctness	20
UI Design & Responsiveness	Proper widget usage, layout design, navigation	10
Code Quality & Structure	Readability, modularity, naming conventions	10
Debugging & Error Handling	Handling invalid inputs, crash-free execution	5
Project Report & Documentation	Proper documentation and presentation	5

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

Project 1: Student Notes App

Description:
Develop a simple notes application to add, edit, and delete notes. Features:

- Text input
- List view of notes
- Navigation between screens Concepts Used:

Flutter widgets, Stateful widgets, Navigation, Forms

Project 2: To-Do List Application

Description:
Create a task management app to track daily activities. Features:

- Add / delete tasks
- Mark tasks as completed
- Responsive UI Concepts Used:

ListView, Checkbox, State management, UI design

Semester – V

.5.7 Major (Core)

Course Title	**Full Stack Development**
Course Credits	**2 Credits**
Course Outcomes	1. Understand core web and mobile application fundamentals including client-server architecture, HTTP/HTTPS, and responsive design.
	2. Develop interactive and responsive user interfaces using modern front-end technologies.
	3. Implement server-side logic and database operations to support dynamic application behavior.
	4. Analyze and resolve full stack development challenges, including state management, asynchronous communication, and security considerations.
	5. Evaluate and apply best practices for deployment, testing, performance optimization, and security in full stack development projects.
Module 1 (Credit 1) Frontend Fundamentals & Modern UI Development & Backend Development & APIs	
Learning Outcomes	1. Students can build responsive, interactive user interfaces using modern frontend tools

Course Outline	• Introduction to Full Stack Development
	  ○ Frontend vs Backend vs Database
	  ○ Web & App development workflow
	• HTML5 & CSS3
	  ○ Semantic HTML
	  ○ Flexbox & Grid
	  ○ Responsive design & mobile-first approach
	• JavaScript (ES6+)
	  ○ Variables, functions, arrays, objects
	  ○ Arrow functions, promises, async/await
	• Modern Frontend Framework – React
	  ○ Component-based architecture
	  ○ JSX
	  ○ Props & state
	  ○ Hooks (useState, useEffect)
	• UI Libraries
	  ○ Tailwind CSS / Material UI (overview)

	• Backend basics
	  ○ Client–Server architecture
	  ○ RESTful API concepts
	• Node.js
	  ○ Event-driven architecture
	  ○ npm & package management
	• Express.js
	  ○ Routing
	  ○ Middleware
	  ○ Request & response handling
	• API Development
	  ○ CRUD operations
	  ○ JSON data handling
	• Authentication Basics
	  ○ JWT authentication
	  ○ Password hashing (bcrypt – concept)

Module 2 (Credit 1) Database, State Management & Full Stack Integration & App Development, Cloud Deployment & Latest Trends	
Learning Outcomes	1. Students can build complete full-stack applications with real databases.

Course Outline	• Database Concepts
	  ○ SQL vs NoSQL
	• MongoDB
	  ○ Collections & documents
	  ○ CRUD operations
	  ○ Mongoose ORM
	• Full Stack Integration
	  ○ Connecting React frontend with Node backend
	  ○ API consumption using Axios / Fetch
	• State Management
	  ○ Lifting state
	  ○ Context API
	  ○ Introduction to Redux
	• File Uploads & Media Handling
	  ○ Image upload concepts
	  ○ Cloud storage overview
	
	• App Development using React Native
	  ○ Cross-platform mobile apps (Android & iOS)
	  ○ Components, navigation, APIs
	• TypeScript
	  ○ Benefits over JavaScript

	○ Typing props, functions, and APIs • Deployment & DevOps Basics ○ Hosting frontend (Vercel / Netlify) ○ Hosting backend (Render / Railway) ○ Environment variables • Security & Performance ○ OWASP Top 10 (overview) ○ CORS ○ Rate limiting • Latest Trends ○ Serverless architecture ○ JAMstack ○ AI integration in web apps (chatbots, APIs)

Assessment:

Internal Assessment: (50 marks)

Evaluation Scheme:

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.pert 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
Requirement Analysis & Planning	Clear understanding of problem statement, features, and user requirements	10
Frontend Implementation	UI design, responsiveness, and usability using frontend technologies	10
Backend & Database Integration	Server-side logic, database connectivity, CRUD operations	10
Functionality & Error Handling	Proper working of features, validation, and exception handling	10
Documentation & Presentation	Project report, architecture explanation, and presentation clarity	10

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

Project 1: Student Management System Description:

Develop a web application to manage student records.

- Backend CRUD operations

Project 2: Online Feedback / Complaint System Description:
A web-based system where users can submit feedback or complaints.

Course Syllabus

Semester – VI

6.2 Community Engagement (CE)

Course Title	**Digital Literacy and Community Outreach**
Course Credits	**2 Credits**
Course Outcomes	1. Understand basic digital literacy concepts and the importance of digital skills in daily life. 2. Use digital tools safely and responsibly for communication, information access, and services. 3. Create awareness about digital platforms and e-governance services among the community. 4. Apply digital knowledge to support community outreach and social inclusion initiatives.
Module 1 (Credit 1) Digital Literacy Fundamentals	
Learning Outcomes	1. Understand fundamental digital literacy concepts and digital devices. 2. Use basic digital tools for communication and information access. 3. Practice safe, ethical, and responsible use ofdigital technologies.
Course Outline	**Introduction to Digital Literacy** ● Meaning and importance of digital literacy ● Digital divide and digital inclusion **Digital Devices and Platforms** ● Computers, smartphones, tablets ● Operating systems and basic applications (conceptual) **Internet Basics** ● Internet services and applications ● Email, search engines, and social media (overview) **Digital Safety and Ethics** ● Cyber safety and security ● Password management and online privacy ● Cyberbullying and responsible digital behaviour

	Digital Payments and Services (Overview) • UPI, online banking, and digital wallets 4. Introduction to e-governance services
Module 2 (Credit 1) Community Outreach and Digital Empowerment	
Learning Outcomes	• Apply digital literacy skills to community-based activities. • Promote awareness of digital services and government initiatives. • Plan and participate in community outreach programs using digital tools.
Course Outline	**Digital Empowerment and Society** • Role of digital literacy in social development • Digital inclusion for rural and underserved communities **E-Governance and Public Digital Services** • Digital India initiatives (overview) • Online public services: education, healthcare, banking, and welfare schemes **Community Outreach Programs** • Planning digital awareness campaigns • Conducting workshops and training sessions • Use of digital media for community engagement **Social Responsibility and Ethics** • Ethical use of digital platforms • Accessibility and inclusivity in digital systems **Case Studies** • Successful digital literacy and community outreach initiatives

Module 1

- Identify common digital tools used in daily life and demonstrate their usage.
- Create a short presentation on cyber safety and responsible internet use.
- Conduct a quiz or worksheet on digital literacy concepts.

Module 2

- Plan a digital awareness program for a local community (conceptual).
- Prepare posters or digital content promoting digital payments or e-governance services.
- Write a brief report on the impact of digital literacy in community development.

Assessment

Internal Assessment: (50 Marks) Evaluation Scheme:

The Internal Assessment shall be conducted based on **Project Development, Presentation, and Report Writing**, evaluated using a well-defined rubric.
Students are required to **select and develop one appropriate project** related to the course, demonstrating practical understanding and application of concepts learned.
Two sample project topics are provided for reference; however, students are encouraged to choose **a relevant and sufficiently advanced project topic** aligned with their interests and capabilities, subject to faculty approval.

The project evaluation will be carried out through **project demonstration and viva/presentation**, based on the following rubric:

Evaluation Criteria	Description	Marks
1. Core Functional Implementation	Correct implementation of core features, completeness of functionality, and logical flow of the application	**20**
2. User Interface Design & Responsiveness	Quality of user interface design, usability, layout consistency, and responsiveness (where applicable)	**10**
3. Code Quality & Structure	Code organization, readability, use of meaningful names, modularity, and adherence to coding standards	**10**
4. Input Validation & Error Handling	Proper handling of user input, validation techniques, and error/exception management	**5**
5. Project Documentation / Report	Quality of project report/documentation including problem statement, methodology, screenshots, results, and conclusions	**5**