Bansilal Ramnath Agarwal Charitable Trust's

Vishwakarma Institute of Technology

(An Autonomous Institute Affiliated to Savitribai Phule Pune University)

Structure & Syllabus

B.Tech. Computer Engineering (Software Engineering)

With Effect from Academic Year 2026-27

Prepared By: Board of Studies in CE(SE)

Approved By: Academic Board, Vishwakarma Institute of Technology, Pune

Chairman – BoS **Chairman – Academic Board**

Index

College Vision and Mission

Vision:

To be a globally acclaimed Institute in Technical Education and Research for holistic socio economic development

Mission:

- To ensure that 100% of students are employable and employed in industry, higher studies, entrepreneurship, civil or defence services, government jobs, and other areas like sports and arts.
- To strengthen Academic Practices in Curriculum, Pedagogy, Assessment and Faculty Competence.
- To Promote Research Culture among Students and Faculty through Projects and Consultancy
- To make students Socially Responsible Citizens

Department Vision and Mission

Vision:

Empowering Industry through Comprehensive Software Engineering Services to Achieve Excellence.

Mission:

- To produce industry-ready software engineering graduate with a blend of technical expertise and ethical responsibility
- To provide software engineering students with the utmost quality in developing technical, social, innovative, and entrepreneurial skills
- To drive technological innovation and shape the future of software engineering

Board of Studies

Sr. No.	Name	Affiliation in Committee	Designation
1.	Prof. (Dr.) Sunil Sangve	Head of Department (Chairman)	Professor and Head
2.	Dr. (Ms.) Leena Deshpande	Entire Faculty of Each Specialization Representative	Associate Professor
3.	Dr. (Ms.) Varsha Jadhav	Entire Faculty of Each Specialization Representative	Assistant Professor
4.	Mr. Shailesh Thaware	Entire Faculty of Each Specialization Representative	Assistant Professor
5.	Mr. Vilas Ghonge	Entire Faculty of Each Specialization Representative	Assistant Professor
6.	Prof. (Dr.) Sunil Wankhede	Subject Expert from Outside University nominated by Academic Council	Professor, Rajiv Gandhi institute of Technology, Mumbai
7.	Prof. (Dr.) Shashank Joshi	Subject Expert from Outside University nominated by Academic Council	Professor, Bharati Vidyapeeth Deemed University, Pune
8.	Prof. (Dr.) Nilesh Uke	VC Nominee	Principal, Indira College of Engineering, Pune
9.	Mr. Dinesh Kothawade	Industry Representative	TCS, Delivery Partner
10.	Mr. Himanshu Tyagi	Meritorious Alumnus Nominated by Director	Sr. Software Developer LinkedIn

Programme Education Objectives (PEO)

PEO	PEO Focus	PEO Statement
1	Core Competence	Demonstrate strong foundational knowledge and technical skills in software engineering principles and practices to solve real-world problems.
2	Application & Innovation	Apply engineering principles, contemporary software development methodologies, and modern computational tools to design and develop efficient, innovative, and scalable software systems that address complex real-world problems and solutions
3	Professionalism & Ethics	Exhibit professionalism, ethical behavior, effective communication, and teamwork in diverse and multidisciplinary environments.
4	Lifelong Learning & Entrepreneurship	Engage in lifelong learning, pursue higher education, research, or entrepreneurship in emerging areas of software engineering and allied fields.

Complex Engineering Problems (WP)

Depth of knowledge Required	**WP1:** Cannot be resolved without in-depth engineering knowledge at the level of one or more of WK3, WK4, WK5, WK6 or WK8 which allows a fundamental- based, first principles analytical approach
Range of conflicting requirements	**WP2:** Involve wide-ranging and/or conflicting technical, non- technical issues (such as ethical, sustainability, legal, political, economic, societal) and consideration of future requirements
Depth of analysis required	**WP3:** Have no obvious solution and require abstract thinking, creativity and originality in analysis to formulate suitable models
Familiarity of issues	**WP4:** Involve infrequently encountered issues or novel problems
Extent of applicable codes	**WP5:** Address problems not encompassed by standards and codes of practice for professional engineering
Extent of stakeholder involvement and conflicting requirements	**WP6:** Involve collaboration across engineering disciplines, other fields, and/or diverse groups of stakeholders with widely varying needs
Interdependence	**WP 7:** Address high level problems with many components or sub- problems that may require a systems approach

Complex Engineering Activities (EA)

Range of resources	**EA1:** Involve the use of diverse resources including people, data and information, natural, financial and physical resources and appropriate technologies including analytical and/or design software
Level of interactions	**EA2:** Require optimal resolution of interactions between wide-ranging and/or conflicting technical, non-technical, and engineering issues
Innovation	**EA3:** Involve creative use of engineering principles, innovative solutions for a conscious purpose, and research-based knowledge
Consequences to society and the environment	**EA4:** Have significant consequences in a range of contexts, characterized by difficulty of prediction and mitigation
Familiarity	**EA5:** Can extend beyond previous experiences by applying principles- based approaches

Knowledge and Attitude Profiles (WK)

WK	WK Statements
WK1	A systematic, theory-based understanding of the **natural sciences** applicable to the discipline and awareness of relevant **social sciences**.
WK2	Conceptually-based **mathematics**, numerical analysis, data analysis, statistics and formal aspects of computer and information science to support detailed analysis and modelling applicable to the discipline.
WK3	A systematic, theory-based formulation of **engineering fundamentals** required in the engineering discipline.
WK4	Engineering **specialist knowledge** that provides theoretical frameworks and bodies of knowledge for the accepted practice areas in the engineering discipline; much is at the forefront of the discipline.
WK5	Knowledge, including efficient resource use, environmental impacts, whole-life cost, reuse of resources, net zero carbon, and similar concepts, that supports **engineering design and operations** in a practice area.
WK6	Knowledge of **engineering practice** (technology) in the practice areas in the engineering discipline.
WK7	**Knowledge of** the role of engineering in society and identified issues in engineering practice in the discipline, such as the professional responsibility of an engineer to public safety and sustainable development.
WK8	Engagement with selected knowledge in the current **research literature** of the discipli awareness of the power of critical thinking and creative approaches to evaluate emerging issues.
WK9	**Ethics, inclusive behavior and conduct**. Knowledge of professional ethics, responsibilities, and norms of engineering practice. Awareness of the need for diversity by reason of ethnicity, gender, age, physical ability etc. with mutual understanding and respect, and of inclusive attitudes.

Program Outcomes (POs)

PO	PO Statements
PO1	**Engineering Knowledge:** Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified inWK1 to WK4 respectively to develop to the solution of **complex engineering problems.**
PO2	**Problem Analysis:** Identify, formulate, review research literature and analyze **complex engineering problems** reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)
PO3	**Design/Development of Solutions:** Design creative solutions for **complex engineering problems** and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)
PO4	**Conduct Investigations of Complex Problems:** Conduct investigations of **complex engineering problems** using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).
PO5	**Engineering Tool Usage:** Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve **complex engineering problems**. (WK2 and WK6)
PO6	**The Engineer and The World:** Analyze and evaluate societal and environmental aspects while solving **complex engineering problems** for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5& WK7).
PO7	**Ethics:** Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)
PO8	**Individual and Collaborative Team work:** Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams. (WK9)
PO9	**Communication:** Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences
PO10	**Project Management and Finance:** Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

PO11	**Life-Long Learning:** Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change. (WK8).

Program Specific Outcomes (PSO)

- PSO 1 – Software Design & Development
  Analyze, design, and implement reliable, efficient, and scalable software solutions by applying software engineering principles, modern programming tools, and innovative methodologies to solve real-world problems.
- PSO 2 – Professional Growth & Innovation
  Demonstrate professionalism, ethics, effective communication, and teamwork, while engaging in lifelong learning, research, Industry and entrepreneurial activities to contribute to emerging areas.

Course Name Nomenclature as per NEP (For FY and SY)

BSC: Basic Science Course

ESC: Engineering Science Course

PCC: Program Core Course

PEC: Program Elective Course

ELC: Experiential Learning Course

MD: Multi-Disciplinary

MDOE: Multi-Disciplinary Open Elective

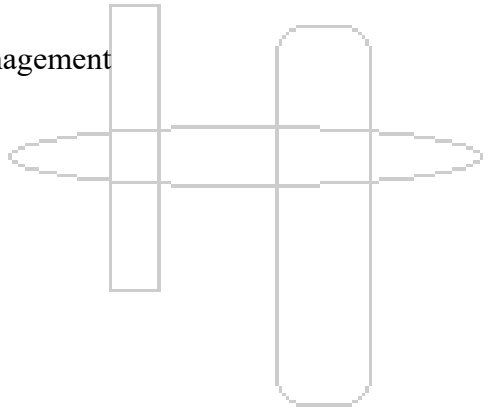
CC: Co-curricular Course

HSSM: Humanities Social Science and Management

IKS: Indian Knowledge System

FP: Field Project

INT: Internship



Nomenclature for Teaching and Examination Assessment Scheme

Sr No.	Category	Head of Teaching/ Assessment	Abbreviation used
1	Teaching	Theory	Th
2	Teaching	Laboratory	Lab
3	Teaching	Tutorial	Tut
4	Teaching	Open Elective	OE
5	Teaching	Multi-Disciplinary	MM
6	Teaching	Software Engineering	SE
7	Assessment	Laboratory Continuous Assessment	CA
8	Assessment	Mid Semester Assessment	MSA
9	Assessment	End Semester Assessment	ESA
10	Assessment	Home Assignment	HA
11	Assessment	Course Project	CP
12	Assessment	Group Discussion	GD
13	Assessment	PowerPoint Presentation	PPT
14	Assessment	Practical	PR
15	Assessment	Mid Semester Examination	MSE
16	Assessment	End Semester Examination	ESE
17	Assessment	Written Examination	W
18	Assessment	Online Examination	O
19	Assessment	Review	R
20	Assessment	Comprehensive Viva Voce	CVV
21	Assessment	Laboratory	LAB

Title: Course Structure **FF No. 653-A**

Branch: CESE **Year: SY** **Academic Year:** 2026-27 **Semester:** I **Module:** III **Pattern:** 2025

Level 5																						
Course Code	NEP Course Category	Course Name	Teaching Scheme (Hrs./week)			Examination Scheme and Marks													Credits			
			Theory	Tutorial	Practical	CVV	CP	LAB/ TUT	GD/PPT	HA	MSE (W)	MSE (R)	MSE (O)	ESE(R)	ESE (W)	ESE (O)	PRACT+ CVV	ESE-Direct Evaluation	Theory	Tutorial	Practical	Total
Semester I																						
SE2003	PCC	FUNDAMENTALS OF DATA STRUCTURES	3	-	2	-	20	-	-	-	-	-	-	-	40	-	40	-	3	-	1	4
SE2004	PCC	DATABASE MANAGEMENT SYSTEMS	2	-	2	20	30	10	-	-	-	-	-	-	40	-	-	-	2	-	1	3
SE2012	PCC	SOFTWARE ENGINEERING	2	-	2	20	30	10	-	-	-	-	-	-	40	-	-	-	2	-	1	3
SE2006	PCC	OBJECT ORIENTED PROGRAMMING	2	-	2	20	30	10	-	-	40	-	-	-	-	-	-	-	2	-	1	3
MM0805	MDM	PROBABILITY & STATISTICS	2	1	-	20	20	-	-	20	40	-	-	-	-	-	-	-	2	1	-	3
HS2002	HSSM	FROM CAMPUS TO CORPORATE - 1	2	-	-	-	-	-	-	-	-	-	50	-	-	50	-	-	2	-	-	2
HS2001	VSEC	REASONING AND APTITUDE DEVELOPMENT - 3	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	100	1	-	-	1
SE2001	AEC	DESIGN THINKING-1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	100	-	1	-	1
SE2002	FP	ENGINEERING DESIGN AND INNOVATION-I	-	-	2	-	-	-	-	-	-	30	-	70	-	-	-	-	-	-	2	2
Total			**14**	**2**	**10**	**80**	**130**	**30**	**0**	**20**	**80**	**30**	**50**	**70**	**120**	**50**	**40**	**200**	**14**	**2**	**5**	**22**

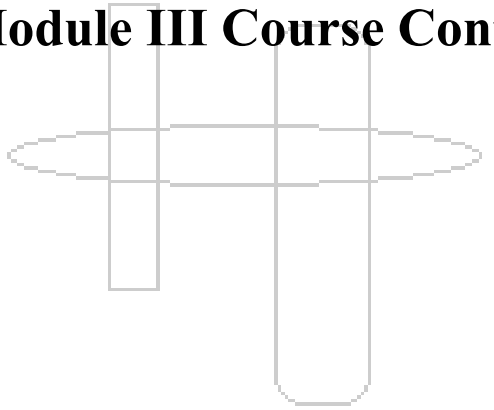
BOS Chairman **Dean Academics**

S.Y. B.Tech. Computer Engineering

(Software Engineering)

AY 2026-27

Module III Course Content



FF No.: 654-A

SE2003: Fundamentals of Data Structures

Teaching Scheme	**Examination Scheme**
Credits: 4	
Lectures: 3 Hrs/week	CP: 20 Marks
Practical: 2 Hrs/week	ESE:40 Marks
Tutorial: 0 Hrs/week	LAB: 40 Marks

Prerequisites:	
●	Knowledge of programming fundamentals
●	Understanding of control structures, functions, and arrays
●	Basic concepts of memory management and file handling
Course Relevance:	
	Data is the core of every software application, and the efficient storage, organization, and processing of data are essential for developing high-performance software, especially when handling large volumes of data. This course introduces students to a variety of data structures, including arrays, linked lists, stacks, queues, trees, and graphs, and enables them to select appropriate techniques for storing and processing data based on application requirements. By understanding the characteristics, advantages, and limitations of different data structures, students learn to design efficient algorithms that optimize time and memory usage. The knowledge gained in this course forms the foundation for developing robust, scalable, and maintainable software systems.
Course Objectives:	
●	To understand need for efficient data organization.
●	To understand need for efficient algorithms to process data effectively.
●	To understand and analyze the complexity of algorithms
●	To implement and apply various sorting and searching techniques
●	To develop efficient programs using stacks, queues, linked lists, trees, and graphs
●	To expose students to memory management, recursion, and performance trade-offs
Course Outcomes:	
	After completion of the course, student will be able to
1.	Analyze and compute time and space complexity of algorithms
2.	Apply searching and sorting algorithms to solve computational problems
3.	Implement and apply stack and queue operations in applications
4.	Develop and use various types of linked lists to manage dynamic data
5.	Apply tree data structures for hierarchical data representation and operations
6.	Select and apply appropriate data structures to design efficient solutions for real-world computing problems

Section1:	
Introduction to Data Structures and Complexity (6 Hours) Abstract Data Types, need of data structures, classification of data structures, complexity analysis of algorithms using Big-O, Big-Ω and Big-Θ notations, time-space trade-off, recursion basics and applications **Arrays and Searching & Sorting Techniques (6 Hours)** 1D and 2D arrays, sparse matrix representation, linear search, binary search, bubble sort, selection sort, insertion sort, merge sort, quick sort, time complexity of all techniques **Stacks and Queues (6 Hours)** Stack operations and applications like expression evaluation, infix to postfix conversion, recursion using stack; queue operations, circular queue, priority queue, deque, applications of queues in real-world problems	
Section2:	
Linked Lists (6 Hours) Singly linked list, doubly linked list, circular linked list, operations (insert, delete, traverse, search), applications such as polynomial operations, memory management using dynamic allocation **Trees (6 Hours)** Tree terminologies, binary trees, binary search trees (BST), tree traversals (inorder, preorder, postorder), AVL trees, expression trees, heap trees, applications in decision making and file systems **Graphs and Hashing (6 Hours)** Graph representation using adjacency list and matrix, BFS, DFS, applications of graphs in networking and social media, hash tables, collision resolution techniques: chaining and open addressing	
Text Books: ***(As per IEEE format)***	
1	E. Horowitz, S. Sahni, and D. Mehta, *Fundamentals of Data Structures in C*, 2nd ed. Hyderabad, India: Universities Press, 2008.
2	Y. Langsam, M. J. Augenstein, and A. M. Tenenbaum, *Data Structures Using C and C++*, 2nd ed. Upper Saddle River, NJ, USA: Pearson Education, 2006.
Reference Books: ***(As per IEEE format)***	
1	Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein; *Introduction to Algorithms*; 3rd Edition; MIT Press; https://mitpress.mit.edu/9780262033848/
2	Brad Miller, David Ranum; *Problem Solving with Algorithms and Data Structures using Python*; Open Book Project; Accessed: May 2025; https://runestone.academy/ns/books/published/pythonds/index.html
URL for Self-Study:	
1	Prof. Naveen Garg; Data Structures and Algorithms; NPTEL – IIT Delhi; https://nptel.ac.in/courses/106/102/106102064
2	UC San Diego, National Research University Higher School of Economics; Data Structures; Coursera; https://www.coursera.org/learn/data-structures

List of Practical's:

1. Write a program to manage person records stored in a file by implementing Create, Read, Update and Delete [CRUD] operations. Each record should contain the following fields: Name, Email, and Mobile Number.
2. Implement array operations including sparse matrix
3. a. Sort the data in ascending order using Bubble sort (Display pass by pass output) and search a particular data using Binary search.

   b. Implement Quick Sort to sort the given list of numbers. Display a corresponding list in each pass.
4. Implement polynomial using Doubly Linked List and perform Addition/ Multiplication of Polynomials
5. Perform implementation of STACK using array
   a. Push an element on to stack
   b. Pop an element
   c. Demonstrate overflow simulations on stack
   d. Display stack
6. Write a program to implement STACK using array and Linked List
   a. Push an element on to stack
   b. Pop an element
   c. report empty/full
   d. Display
7. Write a program to implement Queue using array and Linked List
   1.Enqueue
   2. Dequeue
   3. report empty/full
   4. display
8. Write program to implement CRUD operations on binary tree.
9. Write program to implement CRUD opertions on binary search tree.
10. Write program to implement CRUD opertions on AVL tree.
11. Write a program to represent a graph using an adjacency list. Implement the following graph traversal algorithms
    1. Breadth-First Search (BFS) using a queue.

2. Depth-First Search (DFS) using recursion or a stack.

12. Write a program to implement a hash table using a suitable hash function and resolve collisions using collision resolution techniques such as Linear Probing, Quadratic Probing, or Separate Chaining.

List of Project areas:

1. Finding Nearest Neighbours.
2. Calendar Application using File handling.
3. Path finder in Maze.
4. Word Completion Using Trie.
5. Bloom Filters.
6. Different Management Systems (Eg: Library, Banking with advanced features)
7. Scheduling Applications and Simulation.
8. Shortest Path Applications. (Kirchhoff's Circuit, TSP with Scenarios).
9. Efficient Storage and Data Retrieval Systems.
10. Different Gaming Application.
11. Cash flow minimizer
12. Students Grade Checker
13. Digital Manuscript Version Tracker – Industry project
14. Yantra Generator (Matrix Project) - Industry project
15. Path to Liberation Simulator– Industry project
16. Chakra Stack Simulator– Industry project

CO-PO Mapping

CO	**Program Outcomes (PO)**											**PSO**	
CO \ PO	PO1	PO 2	PO3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PSO 1	PSO 2
CO1	3	2	-	-	-	-	-	-	1	-	3	2	-
CO2	3	3	2	-	-	-	-	-	-	-	3	3	-
CO3	3	3	3	-	2	-	-	-	-	-	3	3	1
CO4	2	2	3	2	3	-	-	-	1	-	3	3	1
CO5	2	2	3	2	3	-	-	-	1	-	3	3	1
CO6	2	2	3	2	3	-	-	-	1	-	3	3	1
Average	2.5	2.33	2.8	2	2.75	-	-	-	1	-	3	2.833	1

CO – WK Mapping (Knowledge & Attitude Profiles) Sample Table

'✓' indicates the knowledge profile drawn upon by the CO.

CO	**WK1**	**WK2**	**WK3**	**WK4**	**WK5**	**WK6**	**WK7**	**WK8**	**WK9**
CO1	✓	✓	✓	✓	–	–	–	✓	–

CO	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO2	✓	✓	✓	✓	✓	–	–	✓	–
CO3	✓	✓	✓	✓	✓	✓	–	✓	–
CO4	✓	✓	✓	✓	✓	✓	–	✓	–
CO5	✓	✓	✓	✓	✓	✓	–	✓	–
CO6	✓	✓	✓	✓	✓	✓	–	✓	–

CO – WP Mapping (Complex Engineering Problems) Sample Table

CO	WP1	WP2	WP3	WP4	WP5	WP6	WP7
CO1	✓	✓	✓	–	–	–	–
CO2	✓	✓	✓	✓	–	–	–
CO3	✓	✓	✓	✓	✓	–	–
CO4	✓	✓	✓	✓	✓	✓	–
CO5	✓	✓	✓	✓	✓	✓	✓
CO6	✓	✓	✓	✓	✓	✓	✓

CO – EA Mapping (Complex Engineering Activities)

'✓' indicates the activity attribute addressed by the CO

CO	EA1	EA2	EA3	EA4	EA5
CO1	✓	✓	✓	–	–
CO2	✓	✓	✓	–	–
CO3	✓	✓	✓	✓	–
CO4	✓	✓	✓	✓	✓
CO5	✓	✓	✓	✓	✓
CO6	✓	✓	✓	✓	✓

CO – SDG Mapping (Sustainable Development Goals) Sample Table

CO	SDG(s) Addressed	Justification / Linkage
CO1	**SDG 4 – Quality Education**	Develops engineering fundamentals, analytical thinking and lifelong learning skills essential for quality education.
CO2	**SDG 4, SDG 9 – Industry, Innovation & Infrastructure**	Encourages engineering design, innovation and application of technical knowledge for sustainable technological development.

CO	SDG(s) Addressed	Justification / Linkage
CO3	**SDG 4, SDG 9**	Promotes innovative engineering solutions using modern software tools and technologies for industrial applications.
CO4	**SDG 4, SDG 9, SDG 12 – Responsible Consumption & Production**	Emphasizes efficient engineering design, optimized software solutions, sustainable use of computing resources and professional practices.
CO5	**SDG 4, SDG 9, SDG 12**	Focuses on designing sustainable engineering solutions with efficient resource utilization, software quality and responsible engineering practices.
CO6	**SDG 4, SDG 9, SDG 12**	Develops industry-ready engineers capable of delivering innovative, efficient and sustainable software engineering solutions aligned with responsible technology development.

FF No.: 654-A

SE2004: Database Management Systems

Teaching Scheme	**Examination Scheme**
Credits : 3	CP: 30 Marks
Lectures : 2 Hrs/week	CVV: 20 Marks
Practical : 2 Hrs/week	Lab : 10 Marks
Tutorial : - Hrs/week	ESE (W): 40 Marks

Prerequisites:	
●	Basic programming knowledge (C/Python)
●	Understanding of data structures and algorithms
●	Logical reasoning and problem solving skills
●	Basic knowledge of file systems
Course Objectives:	
●	To understand the fundamental concepts, architecture, data models, and design principles of database management systems.
●	To develop the ability to design, query, and manage relational databases using SQL and relational algebra.
●	To apply normalization techniques and database design principles for developing efficient, secure, and reliable database systems.
●	To analyze transaction management, concurrency control, recovery mechanisms, and modern NoSQL databases for scalable applications.
Course Outcomes:	
	After completion of the course, student will be able to
1.	Explain database concepts, data models, database architecture, and ER modeling for designing relational databases.
2.	Apply relational algebra and SQL to create, manipulate, and retrieve data while enforcing integrity constraints.
3.	Design normalized relational database schemas using functional dependencies, normalization, and basic database security principles.
4.	Analyze transaction processing, concurrency control, recovery techniques, and NoSQL databases to support reliable and scalable database applications.

Section1:	**Topics/Contents**
Introduction to Databases and Data Models Database concepts, characteristics of database systems, three-schema architecture, data independence, database users and DBA roles, data models: hierarchical, network, relational, object-oriented, entity types, entity sets, attributes, types of attributes, keys, relationship types, ER diagram, ER to relational mapping, Real-world case study: Railway Reservation **Relational Model and SQL**	

Relational model concepts, domains, attributes, tuples, relational algebra operations: select, project, union, set difference, Cartesian product, rename, joins, SQL: DDL, DML, DCL commands, integrity constraints, built-in functions, subqueries, views, joins, set operations, indexing, Modern SQL practices: JSON support in PostgreSQL
Section2: Topics/Contents
Relational Database Design and Normalization DBMS Utilities for Data Warehousing: schema design, problems in database design, functional dependencies, closure of attributes, canonical cover, decomposition, lossless join and dependency preservation, normal forms: 1NF, 2NF, 3NF, BCNF,4NF,5NF. Data Transformation and Database Privacy and Access Control (Basic), Case Study: Transaction management in E-Commerce portals **Transactions, Concurrency Control, and Recovery** Concept of transactions, properties of transactions (ACID), schedule and serializability, concurrency control techniques: locking protocols, timestamp ordering, deadlock prevention and detection, crash recovery, log-based recovery, shadow paging, checkpointing. Hands-on introduction to MongoDB, CAP theorem, use cases for NoSQL, Case Study: Firebase or Amazon DynamoDB for mobile apps

Text Books: *(As per IEEE format)*	
1	Silberschatz A., Korth H., Sudarshan S., "Database System Concepts", McGraw Hill Publishers, ISBN 0-07-120413-X, 6th edition
2	Connally T, Begg C., "Database Systems", Pearson Education, ISBN 81-7808-861-4
Reference Books: *(As per IEEE format)*	
1	C J Date, "An Introduction to Database Systems", Addison-Wesley, ISBN: 0201144719
2	S.K.Singh, "Database Systems: Concepts, Design and Application", Pearson Education, ISBN 978-81-317-6092-5
3	Ivan Bayross, "SQL, PL/SQL the Programming Language of Oracle", BPB Publications ISBN: 9788176569644, 9788176569644

List of Practical's:

1.Identify a problem statement and write a detailed description of the identified problem for any DBMS application.

2.Design an Entity Relationship Diagram by identifying entities, attributes, keys, and relationships.

3. Select a suitable real-world application domain and prepare an exhaustive glossary of data terms. Identify entities, attributes, keys, and relationships from the glossary list. Prepare the Conceptual Data Model, transform it into a Logical Data Model, and physicalize the model into a Physical Database Structure by defining tables, data types, primary keys, foreign keys, and integrity constraints.

4. Using the physical data model developed in Assignment 3, create the database and required tables using DDL commands. Define appropriate integrity constraints and populate the database using DML commands such as INSERT, UPDATE, and DELETE. Demonstrate modification and deletion of records while maintaining referential integrity.

5. Using the populated database, perform data extraction using DQL operations. Write SQL queries using SELECT, WHERE, ORDER BY, DISTINCT, aggregate functions such as COUNT(), SUM(), AVG(), MIN(), and MAX(), along with GROUP BY and HAVING clauses to generate meaningful information from the database.

6. Using the developed database, retrieve information from multiple related tables by implementing INNER JOIN, OUTER JOIN, and SELF JOIN operations. Develop nested queries, single-row and multi-row subqueries, and correlated subqueries to solve complex data retrieval requirements.

7. Using the developed database, create and manipulate views for simplified data access and implement suitable indexes to improve query performance. Apply DCL commands such as GRANT and REVOKE to manage user access and privileges on database objects.

8. Using a suitable real-world database application, demonstrate TCL operations such as COMMIT, ROLLBACK, and SAVEPOINT for managing database transactions. Perform a comprehensive set of DDL, DML, DCL, DQL, and TCL operations, including basic and complex SQL queries involving joins, aggregate functions, subqueries, views, and indexing. Each student shall independently generate and execute a minimum of 20 SQL queries covering the specified operations.

9.**Mini-project:** Design and implement a database management system for a suitable real-world application. The project shall involve creating a Conceptual, Logical, and Physical Data Model, identifying entities, attributes, relationships, and constraints, and implementing the database using SQL. The system should support DDL, DML, DQL, DCL, and TCL operations, data insertion, modification, deletion, and retrieval, aggregate functions, inner/outer/self joins, nested queries, subqueries, views, and indexing.
(Group Activity choose Database like Restaurant, Bank, Library, Air Line reservation etc.)

List of Project areas:

1. Education Management Systems
    - Examples: Student Information System, Library Management, Learning Management System, Online Examination System
2. Healthcare Management Systems
    - Examples: Hospital Management, Clinic Appointment System, Pharmacy Management, Blood Bank Management
3. Business and E-Commerce Systems
    - Examples: Inventory Management, Retail Billing, E-Commerce Order Management, Customer Relationship Management (CRM)
4. Banking and Financial Systems
    - Examples: Banking Management, Loan Processing, ATM Management, Digital Payment Management
5. Smart Applications and NoSQL Systems
    - Examples: Smart Parking System, IoT Data Management, MongoDB-Based Social Media Application, Firebase Mobile Application, Smart City Database System

CO – PO / PSO Articulation Matrix

Correlation: 1 = Low (Slight), 2 = Medium (Moderate), 3 = High (Substantial), '-' = no correlation.

CO	Program Outcome (PO)											PSO	
CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	1	1	1	–	–	–	1	–	2	3	1
CO2	3	3	2	2	3	–	–	–	1	–	2	3	1
CO3	3	3	3	2	2	–	–	1	2	2	2	3	2
CO4	3	3	3	2	3	1	1	1	2	2	2	3	2
Avg	3	2.75	2.25	1.75	2.25	1	1	1	1.5	2	2	3	1.5

CO – WK Mapping (Knowledge & Attitude Profiles)

'✓' indicates the knowledge profile drawn upon by the CO.

CO	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1			✓	✓					
CO2		✓	✓	✓					
CO3			✓	✓					
CO4			✓	✓		✓			

CO – WP Mapping (Complex Engineering Problems)

CO	WP1	WP2	WP3	WP4	WP5	WP6	WP7
CO1	✓						
CO2			✓				
CO3	✓		✓			✓	
CO4	✓				✓	✓	

CO – EA Mapping (Complex Engineering Activities)

'✓' indicates the activity attribute addressed by the CO

CO	EA1	EA2	EA3	EA4	EA5
CO1					
CO2					
CO3				✓	
CO4				✓	

CO – SDG Mapping (Sustainable Development Goals)

CO	SDG(s) Addressed	Justification / Linkage
CO1	4	SDG 4: Quality Education
CO2	4	SDG 4: Quality Education

CO	SDG(s) Addressed	Justification / Linkage
CO3	4	SDG 4: Quality Education
CO4	4,9	SDG 4: Quality Education, SDG 9: Industry, Innovation and Infrastructure

FF No.: 654-A

SE2012: SOFTWARE ENGINEERING

Teaching Scheme	**Examination Scheme**
Credits : 3	CP: 30 Marks
Lectures : 2 Hrs/week	CVV: 20 Marks
Practical : 2Hrs/week	Lab: 10Marks
	ESE (W): 40 Marks

Prerequisites:	
●	Basic Programming Skills, Problem Solving and Programming
Course Objectives:	
●	To introduce the fundamental concepts of Software Engineering and various Software Development Life Cycle (SDLC) models used in real-world applications.
●	To enable students to perform effective requirements engineering, including elicitation, analysis, specification, and validation for software systems.
●	To understand and apply software design principles and architectures using a specific Model.
●	To provide knowledge of software testing, maintenance, configuration management, and project management practices for delivering high-quality software products.
Course Outcomes:	
	After completion of the course, student will be able to
1.	Explain the fundamentals of Software Engineering and various SDLC models.
2.	Elicit, analyse , and document software requirements using standard techniques and SRS.
3.	Apply software design principles and architectural concepts to design a System.
4.	Apply software testing, version control, and project management concepts to develop quality software.

Section1:	**Topics/Contents**

Introduction (7 Hours)

Introduction to Software Engineering, Software crisis, evolution of software engineering, characteristics of good software, attributes of software quality, software engineering layers, software process framework, software development life cycle (SDLC), generic process model, Waterfall model, Incremental model, Prototype model, Spiral model, V-model, Agile overview, Scrum roles, Case Study: Online Shopping System.

Requirements Engineering and Analysis (7 Hours)

Software requirements, types of requirements (functional and non-functional), requirement elicitation techniques (interviews, questionnaires, observation, brainstorming, workshops), feasibility study (technical, operational, economic), requirement analysis, Software

Requirement Specification (SRS) document, components of analysis model, user stories, requirement validation and verification, risk management Case Study: Hospital Management System.
Section2: **Topics/Contents**
Software Design and Architecture (7 Hours) Design Process, Software design principles, Design Concepts, The Design Model, Software Architecture, architectural Genres, user interface design principles, Case Study: Library Management System. **Fundamentals of Testing (7 Hours)** Software testing objectives, testing principles, verification vs validation, types of testing: Unit, Integration, System, Acceptance testing, Alpha and Beta testing, White-box testing, Black-box testing, test case design techniques, debugging, Introduction to software maintenance (corrective, adaptive, perfective, preventive), software configuration management.

Text Books: ***(As per IEEE format)***	
1	*Pressman, R. S., Software Engineering: A Practitioner's Approach, 8th Edition, McGraw-Hill, 2015*
2	*Rajib Mall, Fundamentals of Software Engineering, PHI Learning*
Reference Books: ***(As per IEEE format)***	
1	*Ian Sommerville, Software Engineering, 10th Edition, Pearson, 2016*
2	*Pankaj Jalote, An Integrated Approach to Software Engineering, Wiley*

Software Engineering Laboratory – Practical Assignments

Lab Assignment 1:
Study a real-world software project and analyze the causes and consequences of the Software Crisis. Prepare a report explaining how Software Engineering principles address these challenges.

Lab Assignment 2:
Select an appropriate System as a case study, compare different SDLC models, and justify the selection of the most appropriate development model.

Lab Assignment 3:
Conduct requirement elicitation for your System through a given case study or stakeholder interview, and prepare a Functional and Non-Functional Requirement Specification.

Lab Assignment 4:
Develop a Software Requirement Specification (SRS) document and write user stories with acceptance criteria for a selected software application.

Lab Assignment 5:
Model the behavior of your System by designing Use Case, Activity, Sequence, and State Machine diagrams using any UML modeling tool.

Lab Assignment 6:
Develop and execute test cases for a Login or Registration module using Black-box (Boundary Value Analysis, Equivalence Partitioning) and White-box (Basis Path Testing) techniques. Document the test results.

Lab Assignment 7:
Analyze an existing software application's maintenance history, identify the maintenance type (Corrective, Adaptive, Perfective, or Preventive), and recommend an appropriate maintenance plan.

Lab Assignment 8:
Prepare a Risk Register for a software development project, perform risk analysis, propose mitigation strategies, and maintain project versions using a version control tool such as Git.

Lab Assignment 9:
Perform Verification and Validation activities on the SRS and design documents prepared in previous assignments. Record defects identified during reviews and prepare a validation report.

Lab Assignment 10:
Design comprehensive test cases for a Login Module, execute them using a testing tool or manually, and prepare a report demonstrating the role of software maintenance, configuration management, and risk management in ensuring software quality.

List of Project areas:
- E-Commerce and Online Retail Systems
- Healthcare and Hospital Management Systems
- Education and E-Learning Systems
- Banking and Financial Management Systems
- Transportation and Reservation Systems
- Inventory and Supply Chain Management Systems
- Human Resource and Payroll Management Systems
- Smart Agriculture and Farming Solutions
- IoT-Based Smart Monitoring Systems
- Cybersecurity and Authentication Systems

CO – PO / PSO Articulation Matrix

Correlation: 1 = Low (Slight), 2 = Medium (Moderate), 3 = High (Substantial), '-' = no correlation.

CO	Program Outcome (PO)											PSO	
CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 1	PSO 2
CO1	3	2	1	-	1	-	-	-	1	-	2	3	3

CO	Program Outcome (PO)											PSO	
CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 1	PSO 2
CO2	2	3	2	2	2	-	-	1	1	-	1	2	2
CO3	2	2	3	1	2	-	-	1	1	1	1	3	-
CO4	2	2	2	3	3	1	1	2	2	3	2	3	1
Avg	2.25	2.25	2	1.5	2	0.25	0.25	1	1.25	1	1.5	2.75	1.5

CO – WK Mapping (Knowledge & Attitude Profiles) Sample Table

'✓' indicates the knowledge profile drawn upon by the CO.

CO	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1	✓	✓	✓	✓	✓	✓		✓	
CO2	✓	✓	✓	✓	✓	✓	✓	✓	
CO3	✓	✓	✓	✓	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓	✓	✓	✓

CO – WP Mapping (Complex Engineering Problems) Sample Table

CO	WP1	WP2	WP3	WP4	WP5	WP6	WP7
CO1	✓	✓	✓				
CO2	✓	✓	✓	✓			
CO3	✓	✓	✓	✓	✓	✓	
CO4	✓	✓	✓	✓	✓	✓	✓

CO – EA Mapping (Complex Engineering Activities)

'✓' indicates the activity attribute addressed by the CO

CO	EA1	EA2	EA3	EA4	EA5
CO1	✓				
CO2	✓	✓	✓		
CO3	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓

CO – SDG Mapping (Sustainable Development Goals) Sample Table

CO	SDG(s) Addressed	Justification / Linkage
CO1	4,9	Provides students with fundamental software engineering knowledge and professional competencies essential for lifelong learning and quality technical education (SDG 4), Introduces systematic software development practices that support innovation and the development of reliable digital infrastructure. (SDG 9).
CO2	4	Enhances analytical, communication, and problem-solving skills through practical requirements engineering activities. (SDG 4)
CO3	9	Develops scalable, maintainable, and innovative software architectures that strengthen modern digital infrastructure. (SDG 9)
CO4	9	Ensures the delivery of reliable, maintainable, and high-quality software systems supporting industrial applications. (SDG 9)

FF No.: 654-A

SE2006: OBJECT ORIENTED PROGRAMMING

Teaching Scheme	**Examination Scheme**
Credits: 3	CP: 30 Marks
Lectures: 2 Hrs/week	CVV: 20 Marks
Practical: 2 Hrs/week	Lab CA: 10 Marks
Tutorial: 0 Hrs/week	MSE (W): 40Marks

Prerequisites:	
•	Basic programming knowledge, Understanding of logic building and problem solving
Course Relevance:	
	Object-Oriented Programming (OOP) forms the backbone of modern software development. Java is one of the most widely used programming languages in industry for building enterprise applications, Android apps, and web-based systems. This course enables students to develop modular, reusable, and scalable software using OOP principles. It also prepares learners for advanced topics like frameworks, software engineering, and distributed systems.
Course Objectives:	
•	To introduce the fundamental concepts of Object-Oriented Programming and Java programming structure
•	To develop understanding of classes, objects, inheritance, polymorphism, abstraction, and relationships
•	To enable students to design modular and reusable programs using packages and interfaces
•	To build robust Java applications using exception handling and file handling mechanisms
Course Outcomes:	
	After completion of the course, student will be able to
1.	Explain Java programming fundamentals, program structure, and core OOP concepts such as class, object, encapsulation, and constructors
2.	Apply inheritance, polymorphism, abstraction, and object relationships to design and implement object-oriented solutions
3.	Develop modular applications using packages, interfaces, and access control mechanisms
4.	Implement exception handling and file handling to build reliable and efficient Java applications
Section1:	**Topics/Contents**
Introduction to Java & OOP Concepts: Introduction to OOP: Principles and benefits, Features of Java (platform independent, robust, secure, etc.), Java program structure, JVM, JRE, JDK architecture, Data types, variables,	

operators, Control structures (if, switch, loops), Class and object concepts, Constructors and destructor concept (finalize), Method overloading, Static members and this keyword, Encapsulation and access specifiers

Inheritance and Polymorphism
Types of inheritance (single, multilevel, hierarchical), Method overriding, Dynamic method dispatch, Use of super keyword, Using final with Inheritance, The Object Class, Polymorphism: Compile time and runtime polymorphism, Abstract classes, Association, Aggregation and Composition.

Section2:	**Topics/Contents**

Packages and Interfaces
Packages, Creating and Using Packages, Access Protection, Importing Packages, Java Standard Packages (java.lang, java.util, java.io, java.net), Constructor chaining (with and without packages), Interfaces, Features of Interfaces, Types of Interfaces, Interface Inheritance

Exception Handling and File Handling
Exception-Handling Fundamentals, Exception Types, Uncaught Exceptions, Using try and catch, Multiple catch Clauses, Nested try Statements, throw, throws, finally, Java‟s Built-in Exceptions, Creating Your Own Exception Subclasses, Chained Exceptions, Using Exceptions, File handling (byte stream and character stream), Reading and writing files

Text Books: ***(As per IEEE format)***	
1	H. Schildt, *Java: The Complete Reference*, 11th ed. New York, NY, USA: McGraw-Hill, 2019, ISBN: 978-1260440232.
2	P. Deitel and H. Deitel, *Java: How to Program*, 11th ed. Pearson, 2018, ISBN: 978-0134743356.
Reference Books: ***(As per IEEE format)***	
1	E. Balagurusamy, *Programming with Java*, 6th ed. McGraw-Hill, 2019.
2	C. S. Horstmann, *Core Java Volume I*, 11th ed. Pearson, 2018.
3	K. Sierra and B. Bates, *Head First Java*, 2nd ed. O'Reilly, 2005.
URL for Self-Study:	
1	NPTEL Java Programming Course https://nptel.ac.in/courses/106105191
2	Oracle Java Tutorials https://docs.oracle.com/javase/tutorial/
3	Coursera – Java Programming https://www.coursera.org/specializations/java-programming

List of Practical's:

1. Basic Java & OOP Foundations
    a. Develop a Java program for a simple billing system in a retail shop that demonstrates use of data types, operators, and control structures. The program should accept item details (name, quantity, price), calculate total

bill, apply discount using conditional statements, and display the final amount.

b. Develop a program to create a Student class with attributes such as roll number, name, class, and division. Include member functions to accept and display student information, and create multiple objects to represent different students.

c. Design a program for a Bank Account system where constructors are used to initialize account details. Implement both default and parameterized constructors and demonstrate constructor overloading to initialize accounts with or without initial balance.

d. Develop a program for an Employee Management System demonstrating the use of the static keyword (e.g., company name shared by all employees) and this reference to differentiate between instance variables and parameters.

2. Encapsulation & Access Control
   a. Develop a program for a Student Result Management System implementing encapsulation. Use private data members (marks, grade) and provide public getter/setter methods to access and update them securely.
   b. Create a program demonstrating access specifiers by designing a Library Management module. Show how public, private, and protected members behave within the same class, subclass, and different packages.
3. Inheritance & Polymorphism
   a. Develop a program for a Vehicle Management System demonstrating single and multilevel inheritance. For example, create a base class Vehicle, derived class Car, and further derived class ElectricCar with relevant attributes and methods.
   b. Implement a Banking Application to demonstrate method overriding and dynamic method dispatch. Create a base class Bank and derived classes like SBI, HDFC, etc., overriding methods such as interest calculation.
   c. Design a program for a University System to demonstrate the use of super keyword (calling parent constructor/method) and final keyword (restricting method overriding or inheritance).
   d. Develop a program for a Shape Calculator to demonstrate compile-time polymorphism (method overloading) for calculating area and runtime polymorphism (method overriding) for different shapes like circle, rectangle, etc.
4. Packages and Interfaces
   a. Create user-defined packages for a College Management System (e.g., student, faculty, admin) and implement classes within them.

b. Develop a program to demonstrate importing packages and access protection by accessing classes from different packages in the College Management System.
c. Design a program for an Online Payment System implementing multiple interfaces (e.g., UPI, CreditCard, NetBanking) and demonstrate how a class implements multiple payment methods.
d. Develop a program to demonstrate constructor chaining across classes/packages using a hierarchical employee structure.

5. Exception Handling
   a. Develop a program for a Login Authentication System using:
      i. try-catch-finally to handle invalid login attempts
      ii. Multiple catch blocks to handle different exceptions like invalid input, null values, etc.
   b. Implement a Bank Withdrawal System with user-defined exceptions, such as InsufficientBalanceException.
   c. Create a program to demonstrate use of throw and throws in a Student Admission System where invalid data (e.g., age < required limit) triggers exceptions.
   d. Develop a program for a File Processing System demonstrating nested try blocks and exception propagation while reading multiple files.
6. File Handling
   a. Create a program for a text editor application using character streams (FileReader/FileWriter) to read and write textual data.
   b. Develop a utility program for file backup that copies content from one file to another.

List of Project areas:

1. Library Management System (using file handling + OOP + exception handling)
2. Banking System Simulation (using interfaces + inheritance + exception handling)
3. Employee Payroll System (using inheritance + polymorphism + file handling)
4. Online Shopping System (using interfaces + packages + file handling)
5. Hospital Management System (using aggregation + file handling + exception handling)
6. ATM Simulation System (using abstraction + interfaces + exception handling)
7. Quiz Management System (using file handling + multithreading + OOP)
8. Contact Management System (using file handling + encapsulation)
9. Ticket Reservation System (using inheritance + polymorphism + file handling)
10. Vehicle Rental System (using interfaces + inheritance + exception handling)
11. E-Learning Management System (using packages + interfaces + file handling)
12. Inventory Management System (using OOP + file handling + exception handling)

13. Hotel Booking System (using abstraction + file handling + packages)
14. Online Food Ordering System (using interfaces + OOP + file handling)
15. Student Attendance System (using file handling + encapsulation + exception handling)
16. Chat Application (using multithreading + file handling + OOP)
17. Course Registration System (using packages + interfaces + exception handling)
18. Examination Result System (using polymorphism + file handling)
19. Complaint Management System (using OOP + file handling + exception handling)
20. Bus Reservation System (using inheritance + file handling + exception handling)

CO-PO Mapping

CO	**Program Outcomes (PO)**											**PSO**	
CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	-	-	-	-	-	-	1	-	3	2	-
CO2	3	3	2	-	-	-	-	-	-	-	3	3	-
CO3	3	3	3	-	2	-	-	-	-	-	3	3	1
CO4	2	2	3	2	3	-	-	-	1	-	3	3	1
Average	2.75	2.5	2.67	2	2.5	-	-	-	1	-	3	2.75	1

CO – WK Mapping (Knowledge & Attitude Profiles) Sample Table

'✓' indicates the knowledge profile drawn upon by the CO.

CO	**WK1**	**WK2**	**WK3**	**WK4**	**WK5**	**WK6**	**WK7**	**WK8**	**WK9**
CO1	✓	✓	✓	✓	–	–	–	✓	–
CO2	✓	✓	✓	✓	✓	–	–	✓	–
CO3	✓	✓	✓	✓	✓	✓	–	✓	–
CO4	✓	✓	✓	✓	✓	✓	–	✓	–

CO – WP Mapping (Complex Engineering Problems) Sample Table

CO	**WP1**	**WP2**	**WP3**	**WP4**	**WP5**	**WP6**	**WP7**
CO1	✓	✓	✓	–	–	–	–
CO2	✓	✓	✓	✓	–	–	–
CO3	✓	✓	✓	✓	✓	–	–
CO4	✓	✓	✓	✓	✓	✓	–

CO – EA Mapping (Complex Engineering Activities)

'✓' indicates the activity attribute addressed by the CO

CO	EA1	EA2	EA3	EA4	EA5
CO1	✓	✓	✓	–	–
CO2	✓	✓	✓	–	–
CO3	✓	✓	✓	✓	–
CO4	✓	✓	✓	✓	✓

CO – SDG Mapping (Sustainable Development Goals) Sample Table

CO	SDG(s) Addressed	Justification / Linkage
CO1	**SDG 4**	Develops engineering knowledge and lifelong learning skills through fundamental concepts and problem-solving.
CO2	**SDG 4, SDG 9**	Enhances analytical skills and modern engineering practices supporting innovation and quality education.
CO3	**SDG 4, SDG 9**	Focuses on system design, innovation, engineering solutions, and digital technologies for sustainable industrial development.
CO4	**SDG 4, SDG 9, SDG 12**	Promotes efficient engineering solutions, sustainable software/system development, responsible use of resources, and continuous learning.

FF No.: 654-A

MM0805: PROBABILITY & STATISTICS

Teaching Scheme	Examination Scheme
Credits: 3	Course Project: 30 Marks
Lectures: 2 Hrs/week	GD/PPT: 20 Marks
Practical: -- Hrs/week	Home Assignment: 10 Marks
Tutorial: 1 Hrs/week	ESE (W): 40 Marks

Prerequisites:	
●	Basic knowledge of algebra and calculus
●	Familiarity with Python programming fundamentals
●	Understanding of Data Analytics
Course Objectives:	
●	Understand the fundamental concepts of statistics, data exploration, probability, and sampling techniques for analyzing real-world data.
●	Apply descriptive statistics, probability distributions, and inferential statistical methods to solve engineering and data-driven problems.
●	Analyze datasets using exploratory data analysis, correlation, hypothesis testing, and statistical inference to identify patterns and relationships.
●	Develop and evaluate regression-based predictive models using real-world datasets and effectively communicate analytical findings through technical reports and presentations.
Course Outcomes:	
	After completion of the course, student will be able to
1.	Classify, summarize, and visualize different types of data using appropriate descriptive statistical techniques and exploratory data analysis.
2.	Apply probability concepts, probability distributions, sampling techniques, and statistical inference methods to solve real-world engineering problems.
3.	Analyze relationships among variables using correlation, hypothesis testing, confidence intervals, and ANOVA to support data-driven decision making.
4.	Develop and evaluate simple regression models for predictive analytics using real-world datasets, and communicate the results through technical reports and presentations.

Section1:	**Topics/Contents**
DATA EXPLORATION AND STATISTICAL FOUNDATIONS Introduction to Data Science and Statistics, Types of Data and Variables, Data C by collection Methods, Observational vs. Experimental Studies, Sampling Techniques, Data Cleaning and Preprocessing, Exploratory Data Analysis (EDA), Measures of Central Tendency, Measures of Dispersion, Covariance and Correlation (Pearson & Spearman), Introduction to Predictive Analytics	

PROBABILITY AND PROBABILITY DISTRIBUTIONS

Probability Concepts, Conditional Probability, Bayes' Theorem (Introduction), Random Variables, Discrete and Continuous Probability Distributions:
Binomial Distribution
Poisson Distribution
Normal Distribution
Standard Normal (Z) Distribution
Normal Approximation to Binomial Distribution
Applications of Probability in Data Analytics

Section2:	**Topics/Contents**

SAMPLING AND STATISTICAL INFERENCE

Sampling Distribution, Central Limit Theorem, Estimation, Confidence Intervals, Hypothesis Formulation, One-sample Z-test, One-sample t-test, Independent t-test, Paired t-test, One-way ANOVA

CORRELATION, REGRESSION AND PREDICTIVE ANALYTICS

Review of Correlation, Simple Linear Regression, Introduction to Multiple Linear Regression, Regression Assumptions
Model Evaluation Metrics:

- Mean Absolute Error (MAE)
- Mean Squared Error (MSE)
- Root Mean Squared Error (RMSE)
- R^2 Score

Interpretation of Regression Results,
End-to-End Data Analytics Workflow:

- Dataset Selection
- EDA
- Statistical Analysis
- Regression Modeling
- Result Interpretation

Text Books: ***(As per IEEE format)***

1	R. E. Walpole, R. H. Myers, S. L. Myers, and K. Ye, *Probability and Statistics for Engineers and Scientists*, 9th ed. Upper Saddle River, NJ, USA: Pearson/Prentice Hall, 2012, **ISBN-10: 0321629116**, **ISBN-13: 978-0321629111**
2	D. M. Diez and M. Çetinkaya-Rundel, *OpenIntro Statistics*, 3rd ed. OpenIntro Inc., 2015, **ISBN-10: 194345003X**, **ISBN-13: 978-1943450039**
3	S. D. H. Evergreen, *Effective Data Visualization: The Right Chart for the Right Data*, 3rd ed. SAGE Publications, 2019, **ISBN-10: 1506303056**, **ISBN-13: 978-1506303055**

Reference Books: ***(As per IEEE format)***

1	M. Gardener, *Beginning R: The Statistical Programming Language*. Wiley India Pvt. Ltd., 2013, **ISBN-13: 978-8126541201**.
2	D. Navarro, *Learning Statistics with R: A Tutorial for Psychology Students and Other Beginners*, Version 0.6. Lulu Press / independent publication, 2013, **ISBN-10: 1326189727**, **ISBN-13: 978-1326189723**

3	G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning with Applications in R*. New York, NY, USA: Springer, 2013, **ISBN-10: 1461471370**, **ISBN-13: 978-1461471370** (1st ed.)
4	T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed. New York, NY, USA: Springer, 2009, **ISBN-10: 0387848576**, **ISBN-13: 978-0387848570**.
5	B. Illowsky and S. Dean, *Introductory Statistics*. Houston, TX, USA: OpenStax, 2018, **ISBN-10: 1938168205**, **ISBN-13: 978-1-938168-20-8**
URL for Self-Study:	
1	https://statisticswithrk.blogspot.com/
2	https://nptel.ac.in/courses/111105090
3	https://www.coursera.org/learn/the-power-of-statistics.
4	https://www.coursera.org/projects/probability-distributions-real-world-problems-r

Tutorial:

1. Using Python (without built-in statistical functions), write programs to calculate Mean, Median, Mode, Variance, and Standard Deviation from a given numerical dataset. Compare the manually computed results with the corresponding functions available in NumPy/Pandas and comment on the accuracy of the implementation.
2. Using Python, simulate random experiments such as dice rolling, coin tossing, or card selection to estimate simple probability and conditional probability. Compare the simulated probabilities with the corresponding theoretical probabilities and interpret the results.
3. Develop a Python function to manually calculate Binomial Probability using the Binomial Probability Formula without using built-in probability distribution functions. Validate the results using SciPy and visualize the probability distribution using Matplotlib.
4. Using Python, plot the Normal Distribution Curve for different values of mean (μ) and standard deviation (σ). Compute the probability (area under the curve) for specified intervals using the Standard Normal Distribution available in SciPy and interpret the results.
5. Using Python, repeatedly generate random samples from a given dataset to create the Sampling Distribution of Sample Means, demonstrate the Central Limit Theorem, and construct a 95% Confidence Interval for the population mean. Visualize the sampling distribution and interpret the confidence interval.
6. Using Python, perform a One-Sample Z-Test on a real-world dataset to test a population claim. Calculate the test statistic and p-value, interpret the hypothesis testing results, and conclude whether the null hypothesis should be accepted or rejected.
7. Using Python and SciPy, perform Independent Two-Sample t-Test and Paired t-Test on suitable datasets. Compare the statistical significance of the two tests using p-values, interpret the results, and summarize the findings in a short report.

Dataset:

https://docs.google.com/spreadsheets/d/1NDegFt9UFWRG8JwFOezaAe6VrVmYPRft/edit?usp=sharing&ouid=109552846483903277540&rtpof=true&sd=true

Home Assignment

1. Collect academic data of students, including attendance, study hours, assignment marks, internal assessment scores, and final examination marks, to understand the factors influencing academic performance. Using Python and appropriate libraries such as Pandas, NumPy, and Matplotlib, import the dataset, perform data cleaning and preprocessing, classify the variables into numerical and categorical types, compute descriptive statistical measures, explore the dataset using suitable visualizations, analyze the correlation among variables, and prepare a concise analytical report summarizing the important patterns and insights obtained from the data.
2. A manufacturing company records the number of defective products produced each day and wants to evaluate the reliability of its production process. Using Python, analyze the given dataset by applying probability concepts and appropriate probability distributions. Estimate the probability of defective products, model the data using Binomial, Poisson, and Normal distributions wherever applicable, visualize the distributions, compare theoretical and observed results, and prepare a brief report recommending the most suitable probability model for quality-control decision making.
3. A healthcare organization has conducted a survey to study patient satisfaction across different hospitals and age groups. Using Python, analyze the survey data by selecting appropriate samples, estimating population parameters using confidence intervals, performing suitable hypothesis tests and one-way ANOVA wherever applicable, interpreting the statistical significance of the results using p-values, and preparing a report that justifies whether the observed differences are statistically significant and supports the conclusions with appropriate statistical evidence.
4. A university intends to predict students' final examination performance using attributes such as attendance, study hours, assignment scores, and internal assessment marks. Using Python and suitable machine learning libraries, preprocess the dataset, perform exploratory data analysis, identify relationships among variables through correlation analysis, develop a simple linear regression model for predicting final examination marks, evaluate the model using appropriate regression performance metrics, visualize the prediction results, interpret the findings, and present the complete analysis through a concise technical report.

Dataset:

https://docs.google.com/spreadsheets/d/1oMURnssT0t1gp412MgCEfj0Rraar7pHc/edit?usp=sharing&ouid=109552846483903277540&rtpof=true&sd=true

List of Project areas:

Sr. No.	Project Title	Brief Problem Statement
1	**Student Academic Performance**	Analyze student attendance, assignments, internal assessment, and study-hour data to identify factors affecting academic performance. Perform EDA, descriptive statistics, hypothesis testing, correlation

	Analytics and Prediction	analysis, and build a regression model to predict final examination marks.
2	**House Price Analysis and Prediction System**	Analyze residential property data to understand how area, number of bedrooms, age of property, and location affect house prices. Develop a statistical model using correlation and linear regression to predict house prices and evaluate prediction accuracy.
3	**Employee Salary Analysis and Prediction**	Study employee data containing education, experience, department, and skill ratings to analyze salary trends. Apply descriptive statistics, hypothesis testing, and regression analysis to predict employee salaries and identify the most influential factors.
4	**Healthcare Patient Satisfaction Analysis**	Analyze hospital survey data to evaluate patient satisfaction across hospitals, age groups, and treatment categories. Use confidence intervals, hypothesis testing, ANOVA, correlation, and regression techniques to identify key factors influencing patient satisfaction.
5	**Retail Sales Performance Analysis and Forecasting**	Analyze retail transaction data containing sales, discounts, customer ratings, and seasonal information. Perform statistical analysis and regression modeling to predict future sales and provide recommendations for business improvement.
6	**Air Quality Index (AQI) Prediction**	Analyze environmental parameters such as PM2.5, PM10, CO, NO_2, SO_2, humidity, and temperature. Explore relationships among variables and develop a regression model to predict the Air Quality Index using statistical techniques.
7	**Traffic Volume Analysis and Congestion Prediction**	Analyze traffic sensor data including vehicle count, weather conditions, time of day, and road type. Perform statistical analysis and regression to predict traffic congestion and recommend traffic management strategies.
8	**E-Commerce Customer Purchase Analysis**	Analyze customer demographic information, browsing behavior, product categories, and purchase history. Apply descriptive statistics, probability, hypothesis testing, correlation, and regression to identify factors affecting customer spending and purchase decisions.
9	**Electric Vehicle Energy Consumption Prediction**	Analyze electric vehicle datasets containing battery capacity, vehicle speed, ambient temperature, and driving distance. Use statistical analysis and regression modeling to predict energy consumption and driving range under different operating conditions.
10	**Campus Placement Analytics and Salary Prediction**	Analyze placement records including CGPA, aptitude scores, programming skills, internships, communication skills, and technical interviews. Perform exploratory data analysis, hypothesis testing, correlation analysis, and regression to predict placement package and identify key success factors.

CO – PO / PSO Articulation Matrix

Correlation: 1 = Low (Slight), 2 = Medium (Moderate), 3 = High (Substantial), '-' = no correlation.

CO	Program Outcome (PO)											PSO	
CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	–	2	2	–	–	–	1	–	2	2	2
CO2	3	3	1	2	3	–	–	–	1	–	2	3	2
CO3	2	3	2	3	3	1	–	1	2	–	2	3	2
CO4	2	2	3	3	3	1	–	2	3	2	3	3	3
Avg	**2.5**	**2.5**	**2**	**2.5**	**2.75**	**1**	**–**	**1.5**	**1.75**	**2**	**2.25**	**2.75**	**2.25**

CO – WK Mapping (Knowledge & Attitude Profiles)

'✓' indicates the knowledge profile drawn upon by the CO.

CO	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1	✓	✓	✓	–	–	✓	–	–	–
CO2	✓	✓	✓	✓	–	✓	–	–	–
CO3	–	✓	✓	✓	–	✓	✓	✓	–
CO4	–	✓	✓	✓	–	✓	✓	✓	✓

CO – WP Mapping (Complex Engineering Problems)

CO	WP1	WP2	WP3	WP4	WP5	WP6	WP7
CO1	✓	–	–	–	–	–	–
CO2	✓	✓	✓	–	–	–	✓
CO3	✓	✓	✓	✓	–	✓	✓
CO4	✓	✓	✓	✓	–	✓	✓

CO – EA Mapping (Complex Engineering Activities)

'✓' indicates the activity attribute addressed by the CO

CO	EA1	EA2	EA3	EA4	EA5
CO1	✓	–	–	–	✓
CO2	✓	✓	✓	–	✓
CO3	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓

CO – SDG Mapping (Sustainable Development Goals)

CO	SDG(s) Addressed	Justification / Linkage
CO1	**SDG 4**	Students develop skills in data exploration, statistical analysis, and visualization to support evidence-based learning and educational decision-making using real-world datasets.
CO2	**SDG 9**	Students apply probability, sampling, and statistical inference techniques to solve engineering and industrial problems, supporting innovation and data-driven decision making in manufacturing and technology.
CO3	**SDG 3,9**	Students analyze healthcare, manufacturing, and engineering datasets using statistical inference to support informed decision-making, quality improvement, and efficient engineering practices.
CO4	**SDG 4, 8, 9**	Students develop regression-based predictive models for real-world applications such as education, business, manufacturing, and engineering, promoting data-driven innovation, workforce readiness, and sustainable technological development through project-based learning.

FF No.: 654-A

HS2001: REASONING & APTITUDE DEVELOPMENT-3

Teaching Scheme	Examination Scheme
Credits : 1	CP: -- Marks
Lectures : 1 Hrs/week	GD/PPT/HA: -- Marks
Practical : -- Hrs/week	MSE (W/O): -- Marks
Tutorial : -- Hrs/week	ESE (W/O): 100 Marks

Prerequisites:	
●	Basic knowledge of English grammar and vocabulary.
●	Fundamental understanding of sentence construction and communication skills.
●	Basic arithmetic and mathematical concepts (Class XII level).
●	Elementary logical and analytical reasoning skills.
●	Familiarity with problem-solving techniques.
●	Basic computer literacy and programming concepts.
●	Exposure to at least one programming language (C/C++/Java/Python) is desirable.
Course Objectives:	
●	Develop proficiency in English grammar, vocabulary, reading comprehension, and effective written communication.
●	Strengthen logical, analytical, critical, and reasoning abilities to solve diverse aptitude-based problems.
●	Apply quantitative and mathematical concepts to solve real-world numerical and analytical problems efficiently.
●	Interpret, analyze, and synthesize information from textual, graphical, tabular, and numerical data for informed decision-making.
●	Develop computational thinking and programming aptitude through logic building, pseudocode, algorithmic problem solving, and debugging techniques.
●	Enhance coding proficiency, optimization skills, and problem-solving abilities to perform effectively in programming assessments, campus recruitment tests, and technical interviews.
Course Outcomes:	
	After completion of the course, student will be able to
1.	**Apply** appropriate English language skills, including grammar, vocabulary, reading comprehension, and effective communication techniques to improve verbal and written proficiency.
2.	**Analyze** logical reasoning problems, critical thinking scenarios, and data interpretation tasks to arrive at accurate and evidence-based conclusions.
3.	**Solve** quantitative aptitude problems using mathematical concepts, algebraic reasoning, probability, geometry, and analytical techniques in real-life contexts.

4.	**Develop** efficient algorithmic solutions by applying computational thinking, programming fundamentals, debugging, optimization techniques, and time complexity analysis to solve coding problems.

Section1:	**Topics/Contents**
English Language and Communication Skills: English tenses (Present, Past, Future), active and passive voice, sentence correction, parts of speech, syntax, sentence structure, subject–verb agreement, reading comprehension, skimming and scanning, inference, tone, theme, intent identification, contextual vocabulary. **Logical Reasoning and Analytical Thinking:** Deductive reasoning (syllogisms, coding–decoding, statement–conclusion, data sufficiency, directional sense, logical sequencing), inductive reasoning (analogies, classification, number series, sequence completion, trend identification), abductive and critical reasoning (assumptions, arguments, cause–effect, logical evaluation, decision making), information gathering, interpretation and synthesis using graphs, charts, tables and multi-source data.	
Section2:	**Topics/Contents**
Quantitative Aptitude and Mathematical Reasoning: Number systems, integers, fractions, decimals, divisibility, HCF and LCM, prime and rational numbers, algebraic expressions, linear equations and inequalities, proportional reasoning, speed–time–distance, profit and loss, percentages, averages, ages, mixtures, permutations and combinations, probability, geometry, spatial reasoning, pattern recognition and quantitative problem solving. **Programming Aptitude and Automata Skills:** Logic building, pseudocode, programming fundamentals, basic coding problems, program correctness, execution efficiency, intermediate coding challenges, data structures, algorithms, debugging, optimization techniques, time complexity analysis, industry best coding practices, Automata Pro IT, Automata Pro Max and comprehensive mock assessments.	
Text Books: ***(As per IEEE format)***	
1	R. S. Aggarwal, *Quantitative Aptitude for Competitive Examinations*, Revised and Enlarged Edition. New Delhi, India: S. Chand and Company Ltd., 2017.
2	R. S. Aggarwal, *A Modern Approach to Verbal & Non-Verbal Reasoning*. New Delhi, India: S. Chand Publishing.
3	Dr. M. B. Lal and A. Gupta, *CSAT Logical Reasoning & Analytical Ability*. Agra, India: Upkar Prakashan.
4	G. Laakmann McDowell, *Cracking the Coding Interview: 189 Programming Questions and Solutions*, 6th ed. Palo Alto, CA, USA: CareerCup, 2015.
Reference Books: ***(As per IEEE format)***	
1	B. H. Dowden, *Logical Reasoning*. California State University, Sacramento, CA, USA, 2011.
2	*501 Challenging Logic and Reasoning Problems*, 2nd ed. New York, NY, USA: LearningExpress, LLC, 2005.
3	*Logic and Reasoning Problems*. LearningExpress Practice Series.

Link:
https://drive.google.com/drive/folders/1fXLzuofNhMFq7skonrC0TpCdaCoT7_i0?usp=sharing

CO – PO / PSO Articulation Matrix

Correlation: 1 = Low (Slight), 2 = Medium (Moderate), 3 = High (Substantial), '-' = no correlation.

CO	Program Outcome (PO)											PSO	
CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 1	PSO2
CO1	–	–	–	–	–	–	–	–	3	–	2		
CO2	2	2	–	–	–	–	–	–	1	–	2		
CO3	2	2	–	–	–	–	–	–	1	1	2		
CO4	2	2	3	3	3	–	1	–	1	1	3		
Avg	**2**	**2**	**3**	**3**	**3**	**–**	**1**	**–**	**1.5**	**1**	**2.25**		

CO – WK Mapping (Knowledge & Attitude Profiles)

'✓' indicates the knowledge profile drawn upon by the CO.

CO	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1	–	✓	–	–	–	–	–	✓	✓
CO2	–	✓	–	–	–	–	–	✓	✓
CO3	–	✓	–	–	–	–	–	✓	✓
CO4	–	✓	–	–	–	–	–	✓	✓

CO – WP Mapping (Complex Engineering Problems)

CO	WP1	WP2	WP3	WP4	WP5	WP6	WP7
CO1	–	–	–	–	–	–	–
CO2	✓	–	✓	✓	–	–	–
CO3	✓	–	✓	✓	–	–	–
CO4	✓	–	✓	✓	–	–	✓

CO – EA Mapping (Complex Engineering Activities)

'✓' indicates the activity attribute addressed by the CO

CO	EA1	EA2	EA3	EA4	EA5
CO1	–	–	–	–	–
CO2	–	–	✓	–	✓

CO	EA1	EA2	EA3	EA4	EA5
CO3	–	–	✓	–	✓
CO4	✓	–	✓	–	✓

CO – SDG Mapping (Sustainable Development Goals)

CO	SDG(s) Addressed	Justification / Linkage
CO1	SDG 4	Develops proficiency in English communication, reading comprehension, and language skills, promoting inclusive and quality education and lifelong learning.
CO2	SDG 4	Enhances logical reasoning, critical thinking, and analytical decision-making skills, enabling responsible problem-solving and informed decision-making.
CO3	SDG 4, 8	Strengthens quantitative aptitude and analytical skills essential for employability, professional competence, and workplace productivity.
CO4	SDG 4, 8, 9	Develops computational thinking, programming, and algorithmic problem-solving skills required for innovation, digital transformation, and industry-ready engineering professionals.

FF No.: 654-A

SE2001: Design Thinking 1 and 2

Teaching Scheme	**Examination Scheme**
Credits : 1	CP: Marks
Lectures : Hrs/week	GD/PPT/HA: Marks
Practical : Hrs/week	MSE (W/O):30 Marks
Tutorial : 1 Hrs/week	ESE (W/O):70 Marks

Prerequisites:	
●	Course Prerequisites: Problem Based Learning, Project Centric Learning Course Relevance:
Course Objectives:	
●	To provide ecosystem for students and faculty for paper publication and patent filing
Course Outcomes:	**[Publication of paper or patent]**
	After completion of the course, student will be able to
1.	Understand the importance of doing Research
2.	Interpret and distinguish different fundamental terms related to Research
3.	Apply the methodology of doing research and mode of its publication
4.	Write a Research Paper based on project work
5.	Understand Intellectual property rights
6.	Use the concepts of Ethics in Research, Understand Entrepreneurship and Business Planning

Section1:	**Topics/Contents**
What is research? Importance of Paper Publication and Patents Importance of Paper Publication and Patents Structure of Paper Journal Publication Publication in conference Literature Review Research Paper Writing Journal Ratings and Evaluation How to rate a Journal? Intellectual property (IP) Research Ethics Entrepreneurship	
Section2:	**Topics/Contents**
Structure of the paper Journal List (Top 50 Journals) Selection of the journal Use of various online journal selection tools Plagiarism checking Improving contents of the paper Patent drafting Patent search Filing of patent Writing answers to reviewer questions Modification in manuscript Checking of publication draft	

CO – PO / PSO Articulation Matrix

Correlation: 1 = Low (Slight), 2 = Medium (Moderate), 3 = High (Substantial), '-' = no correlation.

CO	Program Outcome (PO)											PSO	
CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 1	PSO 2
CO1	1	1	1	1	1							1	2
CO2	1	1	1	1	1							1	2
CO3	2	2	3	3	2	2	1	2	2	3		1	3
CO4	3	3	3	3	3	2	1	2	2	3	1	1	3
CO5	1	1	1	1	1							1	2
CO6	2	2	2	2	2	2	1	3	2	3		1	3
Avg	1.7	1.7	1.8	1.8	1.7	2.0	1.0	2.3	2.0	3.0	1.0	1.0	2.5

CO – WK Mapping (Knowledge & Attitude Profiles) Sample Table

'✓' indicates the knowledge profile drawn upon by the CO.

CO	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1	✓	✓		✓					✓
CO2		✓	✓	✓	✓				
CO3		✓	✓	✓	✓	✓			
CO4			✓	✓	✓	✓	✓		
CO5				✓	✓	✓	✓	✓	
CO6	✓			✓		✓		✓	✓

CO – WP Mapping (Complex Engineering Problems) Sample Table

CO	WP1	WP2	WP3	WP4	WP5	WP6	WP7
CO1	✓	✓	✓		✓		✓
CO2	✓	✓	✓	✓	✓		
CO3	✓	✓	✓	✓	✓	✓	
CO4	✓	✓	✓	✓	✓	✓	✓
CO5	✓	✓	✓	✓	✓	✓	✓
CO6			✓		✓	✓	✓

CO – EA Mapping (Complex Engineering Activities)

'✓' indicates the activity attribute addressed by the CO

CO	EA1	EA2	EA3	EA4	EA5
CO1	✓	✓			
CO2	✓	✓	✓		
CO3	✓	✓	✓	✓	
CO4		✓	✓	✓	✓
CO5			✓	✓	✓
CO6				✓	✓

CO – SDG Mapping (Sustainable Development Goals) Sample Table

CO	SDG(s) Addressed	Justification / Linkage
CO1	4, 9	Understanding the importance of research promotes quality education, scientific inquiry, innovation, and technological advancement for solving real-world problems.
CO2	4, 9	Knowledge of research terminology and concepts strengthens analytical thinking, research competency, and innovation capabilities.
CO3	9, 17	Applying research methodology and publishing research fosters innovation, knowledge dissemination, and collaboration among researchers and institutions.
CO4	4, 9, 17	Writing research papers develops scientific communication, promotes lifelong learning, and contributes to global knowledge sharing through collaborative research.
CO5	9,12	Understanding Intellectual Property Rights encourages innovation, protection of inventions, responsible use of knowledge, and sustainable technological development.
CO6	4, 16	Applying research ethics promotes academic integrity, responsible conduct of research, transparency, fairness, and ethical decision-making.
	8, 9	Entrepreneurship and business planning encourage innovation, job creation, sustainable economic growth, and technology-based enterprises.

FF No.: 654-A

SE2002: Engineering Design & Innovation

Teaching Scheme	Examination Scheme
Credits : 2	CP: Marks
Lectures : Hrs/week	GD/PPT/HA: Marks
Practical : 2 Hrs/week	MSE (W/O):30 Marks
Tutorial : Hrs/week	ESE (W/O):70 Marks

Prerequisites:	
●	Course Prerequisites: Problem Based Learning Course Relevance: Project Centric Learning (PCL) is a powerful tool for students to work in areas of their choice and strengths. Along with course based projects, curriculum can be enriched with semester long Engineering Design and Development courses, in which students can solve socially relevant problems using various technologies from relevant disciplines. The various socially relevant domains can be like Health care, Agriculture, Defense, Education, Smart City, Smart Energy and Swaccha Bharat Abhiyan. To gain the necessary skills to tackle such projects, students can select relevant online courses and acquire skills from numerous sources under guidance of facultyand enrich their knowledge in the project domain, thereby achieving project centric learning. Modern world sustained and advanced through the successful completion of projects. In short, if students are prepared for success in life, we need to prepare them for a project-based world. It is a style of active learning and inquiry-based learning. Project centric learning will also redefine the role of teacher as mentor in the learning process. The PCL model focuses the student on a big openended question, challenge, or problem to research and respond to and/or solve. It brings students not only to know, understand and remember rather it takes them to analyze, design and apply categories of Bloom's Taxonomy.
Course Objectives:	
●	To develop critical thinking and problem solving ability by exploring and proposing solutions to realistic/social problems.
●	To Evaluate alternative approaches, and justify the use of selected tools and methods,
●	To emphasize learning activities those are long-term, inter-disciplinary and student-centric.
●	To engage students in rich and authentic learning experiences.
●	To provide every student the opportunity to get involved either individually or as a group so as to develop team skills and learn professionalism.
●	To develop an ecosystem to promote entrepreneurship and research culture among the students
Course Outcomes:	
	After completion of the course, student will be able to
1.	Identify the real life problem from societal need point of view
2.	Choose and compare alternative approaches to select most feasible one

3.	Analyze and synthesize the identified problem from technological perspective.
4.	Design the reliable and scalable solution to meet challenges.
5.	Evaluate the solution based on the criteria specified
6.	Inculcate long life learning attitude towards the societal problems

Section1:	**Topics/Contents**

Preamble - The content and process mentioned below is the guideline document for the faculties and students to start with. It is not to limit the flexibility of faculty and students; rather they are free to explore their creativity beyond the guideline mentioned herewith. For all courses of ED, laboratory course contents of "Engineering Design" are designed as a ladder to extend connectivity of software technologies to solve real word problem using interdisciplinary approach. The ladder in the form of gradual steps can be seen as below:

Industry Communication Standards, Single Board Computers and IoT, Computational Biology (Biomedical and Bioinformatics), Robotics and Drone, Industry 4.0 (Artificial Intelligence, Human Computer Interfacing, 5G and IoT, Cloud Computing, Big Data and Cyber Security etc).

Group Structure:

There should be a team/group of 4-5 students. A supervisor/mentor teacher assigned to individual groups. It is useful to group students of different abilities and nationalities together.

Selection of Project/Problem:

● Students must focus on initiating the task/idea. The idea of inception and consideration shall be from following areas as a real world problem:

-Health Care, Agriculture, Defense, Education, Smart City, Smart Energy, Swaccha Bharat Abhiyan, Environment, Women Safety.

● This is the sample list to start with. Faculty and students are free to include other ar eas which meet society's requirements at large.

● The model begins with the identifying of a problem, often growing out of a question or "wondering". This formulated problem then stands as the starting point for learning. Students design and analyze the problem/project within an articulated disciplinary subject frame/domain.

● A problem can be theoretical, practical, social, technical, symbolic, cultural, and/or scientific and grows out of students' wondering within different disciplines and professional environments. A chosen problem has to be exemplary. The problem may involve an interdisciplinary approach in both the analysis and solving phases.

● By exemplarity, a problem needs to refer to a particular practical, scientific, social and/or technical domain. The problem should stand as one specific example or manifestation of more general learning outcomes related to knowledge and/or modes of inquiry.

Teacher's Role in PCL:

● Teacher is not the source of solutions rather he will act as the facilitator and mentor.

● To utilize the principles of problem solving, critical thinking and metacognitive skills of the students.

● To make the group aware about time management.

● Commitment to devote the time to solving student's technical problems and interested in helping students to empower them better.

Student's Role in PCL:

- Students must have ability to initiate the task/idea, they should not be mere imitators.
- They must learn to think.
- Students working in PCL must be responsible for their own learning.
- Students must quickly learn how to manage their own learning, instead of passively receiving instruction.
- Students in PCL are actively constructing their knowledge and understanding of the situation in groups.
- Students in PCL are expected to work in groups.
- They have to develop interpersonal and group process skills, such as effective listening or coping creatively with conflicts.

Developing Inquiry Skills:

- Students in PCL are expected to develop critical thinking abilities by constantly relating: What they read to do? What do they want to do with that information?
- They need to analyze information presented within the context of finding answers.
- Modeling is required so that the students can observe and build a conceptual model of the required processes.
- Use the following mechanism to maintain the track of moving towards the solution.
  - How effective is …….?
  - How strong is the evidence for ……?
  - How clear is ……?
  - What are the justifications for thinking? Why is the method chosen?
  - What is the evidence given to justify the solution?

Literature Survey – To avoid reinvention of wheel:

- It is an integral part of self- directed learning
- Identify the information needed to solve a given problem or issue
- Be able to locate the needed information
- Use the information to solve the given problem effectively.
- Skills required by students in information literacy include:
- · How to prepare for the search? How to carry out research
- Sorting and assessing of information in general

Use of Research Methodology: - investigation, collaboration, comprehension, application, analysis, synthesize and evaluation

Focus on the following skills while working in a team to reach to solution:

- Collaborative learning
- Interpersonal Skills
- Resources Evaluation
- Metacognitive Skills
- Reflection Skills

Section2:	**Topics/Contents**

EDD Sample Case Studies: -

With the adaptation of industry communication standards, Raspberry Pi and Sensors, the following projects can be taken up: 1) Design a deployable product for soil moisture detection 2) Design a deployable product for temperature detection 3) Design a deployable product for pressure detection 4) Design a deployable product smoke detection 5) Design a deployable product for motion detection 6) Design a deployable product for collision detection 7) Design a deployable product for sound detection ...**not limited to.....Faculty and students are free to include other areas which meet society's requirements at large.**	
Text Books: ***(As per IEEE format)***	
1	*A new model of problem based learning. By Terry Barrett. All Ireland Society for higher education (AISHE). ISBN:978-0-9935254-6-9; 2017*
2	*Problem Based Learning. By Mahnazmoallem, woei hung and Nada Dabbagh, Wiley Publishers. 2019.*
3	*Stem Project based learning and integrated science, Technology, Engineering and mathematics approach. By Robert Robart Capraro, Mary Margaret Capraro*
4	*A new model of problem based learning. By Terry Barrett. All Ireland Society for higher education (AISHE). ISBN:978-0-9935254-6-9; 2017*
Reference Books: ***(As per IEEE format)***	
1	De Graaff E, Kolmos A., red.: Management of change: Implementation of problembased and project-based learning in engineering. Rotterdam: Sense Publishers. 2007.
2	Project management core textbook, second edition, Indian Edition , by Gopalan.
3	The Art of Agile Development. By James Shore & Shane Warden

CO – PO / PSO Articulation Matrix

Correlation: 1 = Low (Slight), 2 = Medium (Moderate), 3 = High (Substantial), '-' = no correlation.

CO	Program Outcome (PO)											PSO	
CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 1	PSO2
CO1	2	2	2	2					3		2	3	3
CO2	2	2	3	2	2		2		3		2	3	3
CO3	2	2	3	2	3		2		3		2	3	3
CO4	2	2	3	2	3	3		2	3		2	3	3
CO5	2	2	3	2	3	2			3		2	3	3
CO6	2	2	3	3	2				3		3	3	3
Avg	2.0	2.0	2.83	2.83	2.6	2.5	2.0	2.0	3.0	1.0	2.16	3.0	3.0

CO – WK Mapping (Knowledge & Attitude Profiles) Sample Table

'✓' indicates the knowledge profile drawn upon by the CO.

CO	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1		✓	✓	✓	✓			✓	
CO2		✓	✓	✓	✓	✓		✓	
CO3		✓	✓	✓	✓	✓		✓	
CO4		✓	✓	✓	✓	✓	✓	✓	✓
CO5		✓	✓	✓	✓	✓	✓	✓	
CO6		✓	✓	✓	✓	✓		✓	

CO – WP Mapping (Complex Engineering Problems) Sample Table

CO	WP1	WP2	WP3	WP4	WP5	WP6	WP7
CO1	✓			✓		✓	✓
CO2	✓	✓				✓	
CO3	✓		✓		✓		
CO4	✓				✓	✓	
CO5	✓	✓	✓				
CO6	✓					✓	✓

CO – EA Mapping (Complex Engineering Activities)

'✓' indicates the activity attribute addressed by the CO

CO	EA1	EA2	EA3	EA4	EA5
CO1	✓	✓			
CO2	✓	✓	✓	✓	
CO3	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓
CO5	✓	✓	✓	✓	✓
CO6	✓	✓	✓		✓

CO – SDG Mapping (Sustainable Development Goals) Sample Table

CO	SDG(s) Addressed	Justification / Linkage
CO1	4,9	Identifies real-life societal problem and develops engineering knowledge for innovative solutions
CO2	4,9,12	Evaluates and compares alternative solutions to select sustainable and resource efficient approaches
CO3	4,9,11	Analyze engineering problems and synthesizes technological solutions for societal needs

CO	SDG(s) Addressed	Justification / Linkage
CO4	4,9,11,13	Designs reliable,scalable and sustainable engineering solutions addressing environmental and societal challenges
CO5	4,9,16	Evaluates engineering solutions based on defined technical, ethical,and quality criteria to ensure reliability and accountability
CO6	4,17	Promotes lifelong learning,professional development,collaboration and commitment to solving societal problems