J.C. BOSE UNIVERSITY OF SCIENCE AND TECHNOLOGY, YMCA, FARIDABAD, HARYANA, (INDIA)
A State Government University (Accredited 'A⁺' Grade by NAAC)
(Established by Haryana State Legislative Act No. 21 of 2009, Recognized by U.G.C. u/s 2 (f) and 12(B) of U.G.C. Act 1956)
SECTOR-6, MATHURA ROAD, FARIDABAD-121006, HARYANA, (INDIA)

Community College of Skill Development

Lesson Plan: Software Engineering (WDD-207-V)

Program: B.Voc. Web Development   **Semester: III**   **Credits: 03**

Course Objectives:

To provide students with a solid foundation in software engineering principles, life cycle models, design methodologies, and project management, with emphasis on skills for requirement analysis, software design, testing, and maintenance of reliable software systems.

Course Outcomes: After the successful completion of the course, students will be able to:

CO1: Understand and explain the nature of software crises, various software life cycle models, and requirements engineering processes including elicitation techniques, DFDs, ER diagrams, and Software Requirement Specifications (SRS).

CO2: Apply software project management concepts such as estimation techniques, cost models like COCOMO, and risk management strategies for effective software planning and execution.

CO3: Demonstrate the ability to design software systems using function-oriented and object-oriented methodologies, evaluate cohesion and coupling, and apply software metrics for quality measurement and implementation practices.

CO4: Design and execute test cases for various levels of testing (unit, integration, system), debug software, and understand software maintenance processes, including reengineering and configuration management.

Equipment required in Classroom

      i. LCD/Projector
      ii. Whiteboard/ Black Marker

Assessment Scheme

S.No.	Criteria	Marks
1	End Term Examination	75
2	Internal Evaluation Scheme	25
2a	Class Tests	15
2a (i)	Class Test-I	7.5

2a (ii)	Class Test-II	7.5	
2(b)	Teacher Assessment (Continuous Evaluation)	10	
2b (i)	Attendance	5	
2b (ii)	Assignment / Presentation	5	

Lesson Plan

Lecture No.	Content to be Covered	Pedagogy	Date of Implementation	Course Outcomes Covered	Faculty Sign
Unit – 1: Introduction & Requirement Analysis (9 Lectures)					
1	Introduction to Software Engineering and Software Crisis	Lecture	05 Aug 2025 (Tuesday)	CO1	
2	Software Process and Characteristics	Lecture	06 Aug 2025(Wednesday)	CO1	
3	Software Life Cycle Models – Overview	Lecture	07 Aug 2025 (Thursday)	CO1	
4	Waterfall, Prototype, and Evolutionary Models	Lecture and Chart	12 Aug 2025(Tuesday)	CO1	
5	Spiral Model – Concept and Application	Lecture	13 Aug 2025(Wednesday)	CO1	
6	Requirement Engineering – Process and Activities	Lecture	14 Aug 2025(Tuesday)	CO1	
7	Requirement Elicitation Techniques – FAST, QFD	Lecture	19 Aug 2025 (Tuesday)	CO1	
8	Requirement Analysis using DFD and Data Dictionary	Diagram Explanation	21 Aug 2025(Thursday)	CO1	
9	ER Diagrams, SRS Documentation, and Characteristics	Lecture	26 Aug 2025 (Tuesday)	CO1	
Unit – 2: Software Project Management (7 Lectures)					
10	The Management Spectrum – People, Problem, Process, Project	Lecture	27 Aug 2025 (Wednesday)	CO2	
11	Project Planning Concepts	Lecture	28 Aug 2025(Thursday)	CO2	
12	Size Estimation – Lines of Code, Function Count	Lecture	02 Sep 2025(Tuesday)	CO2	
13	Cost Estimation Models – Basic and COCOMO	Lecture	03 Sep 2025(Wednesday)	CO2	

14	Risk Management – Identification and Analysis	Lecture	04 Sep 2025(Thursday)	CO2	
15	Risk Control and Monitoring	Lecture	09 Sep 2025(Tuesday)	CO2	
16	Software Project Planning and Risk Estimation	Lecture	11 Sep 2025 (Thursday)	CO2	
Unit – 3: Software Design and Implementation (12 Lectures)					
17	Software Design Concepts	Lecture	16 Sep 2025 (Tuesday)	CO3	
18	Cohesion and Coupling	Lecture	17 Sep 2025(Wednesday)	CO3	
19	Classification of Cohesiveness and Coupling	Lecture	18 Sep 2025(Thursday)	CO3	
20	Function-Oriented Design – Structure Chart	Diagram and Explanation	23 Sep 2025 (Tuesday)	CO3	
21	Object-Oriented Design – Concepts and Principles	Lecture	24 Sep 2025 (Wednesday)	CO3	
22	Software Metrics – Definition and Types	Lecture	25 Sep 2025 (Thursday)	CO3	
23	Software Measurement – Token Count and Halstead's Metrics	Lecture	30 Sep 2025(Tuesday)	CO3	
24	Design Metrics and Data Structure Metrics	Lecture	01 Oct 2025 (Wednesday)	CO3	
25	Relationship between Design and Implementation	Lecture	02 Oct 2025 (Thursday)	CO3	
26	Implementation Issues and Coding Standards	Lecture	07 Oct 2025 (Wednesday)	CO3	
27	Coding the Procedural Design	Lecture	08 Oct 2025(Wednesday)	CO3	
28	Good Coding Style and Programming Practices	Lecture	09 Oct 2025 (Thursday)	CO3	
Unit – 4: Testing and Maintenance (12 Lectures)					
29	Introduction to Software Testing	Lecture	14 Oct 2025 (Tuesday)	CO4	
30	Testing Process and Test Case Design	Lecture	15 Oct 2025 (Wednesday)	CO4	
31	Types of Testing – Overview	Lecture	16 Oct 2025(Thursday)	CO4	
32	Functional Testing Techniques	Lecture		CO4	
33	Structural Testing Techniques	Lecture		CO4	
34	Test Activities and Unit Testing	Lecture		CO4	

35	Integration Testing	Lecture		CO4	
36	System Testing	Lecture		CO4	
37	Debugging Activities and Tools	Lecture		CO4	
38	Software Maintenance and its Management	Lecture		CO4	
39	Reverse Engineering and Reengineering	Lecture		CO4	
40	Configuration Management and Documentation	Lecture		CO4	

Text Books

1. Pressman, Roger S. Software Engineering: A Practitioner's Approach. McGraw Hill Education, 9th Edition, 2020.
2. Sommerville, Ian. Software Engineering. Pearson Education, 10th Edition, 2020.

Reference Books

1. Pfleeger, Shari Lawrence, and Atlee, Joanne M. Software Engineering: Theory and Practice. Pearson Education, 4th Edition, 2018.
2. Rajaraman, V. and Mall, Rajib. Fundamentals of Software Engineering. PHI Learning, 5th Edition, 2018.