TEACHING PLAN

Course Code	Course Title	Year/Sem	Branch	Contact hr/week	Academic Year
24BC4T04	Software Engineering	II/IV	BCA(Honours)	6	2025-2026

Course Objectives:

The objectives of this course are to introduce

- Software life cycle models, Software requirements and SRS document
- Project Planning, quality control and ensuring good quality software.
- Software Testing strategies, use of CASE tools, Implementation issues, validation & verification procedures.

Course Outcome(co's) :At the end of the course, student will be able to:

CO No.	Course Outcome	Knowledge Level (K)
CO1	Understand the evolution of software engineering and compare various software development life cycle models.	K2
CO2	Apply project estimation techniques, risk management strategies, and formal requirement specification methods.	K3
CO3	Analyze and design function-oriented and agile-based software systems, including user interface design principles.	K4
CO4	Evaluate software through systematic coding practices, testing methods, and quality assurance techniques.	K4
CO5	Understand and analyze software maintenance models and reuse strategies in software development.	K3

Week No	Outcome	Blooms Level	Topic / Activity		Text Books	Contact Hours	Delivery Method
			UNIT-I				
1.2			1.1	Introduction: Evolution, Software development projects	T1	2	Chalk & Board, PPT, Interactive White boarding
			1.2	Exploratory style of software	T1	1	
			1.3	Emergence of software engineering	T1	1	
			1.4	Notable changes in software development practices.	T1	2	
			1.5	Computer system engineering.	T1	1	
			1.6	Software Life Cycle Models: Basic concepts	T1	2	
			1.7	Waterfall model and its extensions	T1	1	
			1.8	Rapid application development, Agile development model, Spiral model.	T1	3	
			UNIT-II				
3,4		K3	2.1	Software Project Management: Software project management complexities	T1	2	Chalk & Board, PPT , Interactive White boarding
			2.2	Responsibilities of a software project manager	T1	1	
			2.3	Metrics for project size estimation, Project estimation techniques,	T1	2	

				2.4	Empirical Estimation techniques, COCOMO, risk management.	TI	3	
				2.5	Requirements Analysis and Specification: Requirements gathering and analysis	TI	3	
				2.6	Software Requirements Specification (SRS) Formal system specification,	TI	2	
				2.7	Axiomatic specification Algebraic specification, Executable specification and 4GL.	TI	3	

Mid I Exam

UNIT-III

				3.1	Software Design: Overview of the design process, characterizing a good software design.	T1	2	
5, 6		K4		3.2	Layered arrangement of modules, Cohesion and Coupling. approaches to software design	T1	3	Chalk & Board, PPT , Interactive White boarding
				3.3	Agility: Agility and the Cost of Change, Agile Process,	T1	2	
				3.4	Extreme	T1	3	

					Programming (XP), Other Agile Process Models, Tool Set for the Agile Process		
				3.5	Function-Oriented Software Design: Overview of SA/SD methodology, Structured analysis, Developing the DFD model of a system, Structured design, Detailed design, and Design Review.	T1	4
				3.6	User Interface Design: Characteristics of a good user interface, Basic concepts, Types of user interfaces	T1	3
				3.7	Fundamentals of component-based GUI development, and user interface design methodology.	T1	3
UNIT-IV
7,8		K4		4.1	Coding and Testing: Coding, Code review, Software documentation, Testing, Black-box testing, White-Box Testing	T1	4
				4.2	Debugging, Program analysis tools, Integration testing	T1	2
				4.3	Testing object oriented programs, smoke testing and some general issues associated with	T1	3

Chalk & Board, PPT , Interactive White boarding

					testing			
				4.4	Software reliability and quality management: software Reliability, statistical testing	T1	3	
				4.5	Software Quality, Software quality management system, ISO 9000	T1	3	
				4.6	SEI Capability maturity model. Few other important quality standards, and Six Sigma	T1	3	

<div align="center">

UNIT-V

</div>

				5.1	**Software Maintenance:** Characteristics of software maintenance, Software reverse engineering	T1	3	
				5.2	Software maintenance process models and Estimation of maintenance cost.	T1	2	
9, 10				5.3	**Software Reuse:** reuse-definition, introduction, reason behind no reuse so far	T1	2	Chalk & Board, PPT , Interactive Whiteboarding
				5.4	Basic issues in any reuse program	T1	2	
				5.5	A reuse approach, and Reuse atorganization level.	T1	2	

Total No. of Classes	67

Recommended Text Books for Reading:

Text Book 1: Fundamentals of Software Engineering, Rajib Mall, 5th Edition, PHI

Text Book 2: Software Engineering: A Practitioner's Approach, Roger S. Pressman, 9th Edition, Mc Graw Hill International Edition.

Reference Books:

1. Software Engineering, Ian Sommerville,10th Edition, Pearson.

2. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.

Faculty

Head of the Department

Principal