



SOFTWARE ENGINEERING

1. Course Code	2321101122
2. Course Category	Core
3. Semester	II

4. Prerequisite:

Basic understanding of computer systems, databases, and operating systems. Familiarity with programming fundamentals and logical reasoning skills for analytical problem-solving.

5. Course Outcome:

After Completion of the Course, Student will be able to:

CO No.	CO	Bloom's Taxonomy Level*
1	Understand the principles of software engineering and distinguish between various software development models.	2
2	Develop and document software requirements using structured techniques like SRS and information modeling.	3
3	Design modular, user-friendly, and maintainable software systems using design principles and UML diagrams.	6
4	Evaluate agile frameworks, including Scrum and DevOps, to optimize software development processes.	5
5	Analyze emerging trends like AI and DevOps and their implications in modern software development projects.	4

**1: Remember, 2: Understand, 3: Apply, 4: Analyze, 5: Evaluate, 6: Create*



6. Teaching and Examination Scheme:

Teaching Scheme (in Hours)			Total Credits L+T+ (P/2)	Assessment Pattern and Marks				Total Marks
L	T	P	C	Theory		Practical		
				Internal	External	Internal	External	
4	0	0	4	50	50	-	-	100

7. CO- PO Mapping:

CO	PO												PSO		
	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3
1	3	2	2	-	-	2	1	2	2	-	-	-	2	1	-
2	2	3	3	2	2	2	2	2	2	-	-	-	2	2	-
3	1	2	3	2	2	2	2	2	1	1	-	-	2	2	2
4	3	3	3	3	2	2	3	2	2	2	2	2	2	2	2
5	2	1	2	3	2	2	2	2	3	3	2	2	2	3	3
Avg.	2.4	2.2	2.6	2.25	2	2	2	2	2.2	1.5	1.25	1	2	2	1.75

'3' for high, '2' for medium, '1' for low and '-' for no correlation

8. Course Content:

Unit No.	Content	No. of Hours	Weightage
1.	Fundamentals of Software Engineering: Software Characteristics and Applications, Software Myths and Software Crisis, Software Engineering vs. Traditional Engineering, Software Engineering a Layered Technology, Software Development Life Cycle, SDLC Models: Water Fall Model, Prototype Model, Spiral Model, Evolutionary Development Model,	7	20%



	Iterative Enhancement Model, Agile Model, Software Process Framework		
2.	Requirement Engineering: Functional and Non-Functional Requirements, Requirement Engineering Process - Elicitation, Analysis, Documentation, Review and Management of User Needs, Feasibility Study, Information Modeling, SRS Document	7	20%
3.	Design Engineering: Design Concepts: Abstraction, Modularity, Cohesion, Coupling, Golden Rules of User Interface Design, User Interface Analysis and Design, Object-Oriented Analysis and Design using UML Diagrams (Structure Diagram, Behavior Diagrams: Use case, Activity, Sequence Diagram), Design Patterns (Singleton, Factory, Observer)	8	22%
4.	Agile Development: What is agility? Agility and the cost of change, Agile process, Principles of Agile Manifesto, Scrum framework (roles, ceremonies, artifacts), Sprint planning, sprint execution, Agile Software Engineering Process Models, Extreme Programming, Agile Software Development, Site Reliability Engineering (SRE), Kanban, A toolset for Agile process	8	22%
5.	Emerging Trends in Software Engineering: Incorporating security practices in the SDLC, Measuring User Satisfaction, Software Engineering Ethics, Artificial Intelligence in Software Development, Low-Code/No-Code, DevOps and Continuous Delivery, Emerging Process Models	6	16%
Total		36	

9. References/Suggested Learning Resources:

a. Reference Books:

- Ian Sommerville - Software Engineering, Latest Edition, Pearson Education



- Roger S. Pressman - Software Engineering: A Practitioner's Approach, Latest Edition, McGraw Hill Education
- Andrew Stellman, Greene Jennifer - Beginning Agile, Latest Edition, O'Reilly
- K.K. Aggarwal & Yogesh Singh - Software Engineering, Latest Edition, New Age International Publishers
- Rods Stephen - Beginning Software Engineering, Latest Edition, WROX
- Pankaj Jalote - Software Engineering - A Precise Approach, Latest Edition, Wiley India

b. Online Resources:

- SAP Build: What is Low-Code No-Code?
<https://www.sap.com/india/products/technology-platform/build/what-is-low-code-no-code.html>
- The Agile Modeling Mission: Agile Modeling Mission
<https://agilemodeling.com/>
- Agile Alliance Resources: Agile Alliance Resources
<https://www.agilealliance.org/resources/>
- Scrum Guides: Scrum Guides
<https://www.scrumguides.org/>
- UML Tools, IDEs, Application Servers, Data Sources: Comparison of UML Tools
<https://pejava.tripod.com/comparison.html>
- Unified Modeling Language (UML) Resources: UML Diagrams
<https://www.uml-diagrams.org/>
- UML Design Tutor
<http://groups.umd.umich.edu/cis/tinytools/cis375/f21/team8/UMLTutorMainMenu.html>
- ISO/IEC/IEEE International Standard: Systems and Software Engineering Standards
<https://standards.ieee.org/ieee/29148/6937/>
- Tiny Software Engineering Tools
<http://groups.umd.umich.edu/cis/tinytools/index.html>
- NPTEL Courses:
<http://nptel.ac.in/courses/106105087/3>