

# **ACADEMIC CURRICULA FORMAT**

## **UNDERGRADUATE DEGREE PROGRAMS**

**Bachelor of Technology**  
**B. Tech Four Years**  
**(Choice Based Credit System)**

**Regulations (2023)**



**SRM University – AP, Andhra Pradesh**

**Software Engineering and Project Management**

Course Code	CSE 305	Course Category	Professional Core (C)	L-T-P-C	3	0	2	4
Pre-Requisite Course(s)	CSE 101, CSE 236	Co-Requisite Course(s)		Progressive Course(s)				
Course Offering Department	CSE	Professional / Licensing Standards	IEEE					
Board of Studies Approval Date	-- Board of Studies -- , 2018	Academic Council Approval Date						

**Course Objectives / Course Learning Rationales (CLRs)**

**Objective 1:** To comprehend software development life cycle.

**Objective 2:** To gain knowledge of requirement engineering and SRS documents.

**Objective 3:** To understand software architecture styles.

**Objective 4:** To learn various software testing techniques and their applicability.

**Objective 5:** To apply and analyze project management life cycle.

**Course Outcomes / Course Learning Outcomes (CLOs)**

	At the end of the course the learner will be able to	Bloom's Level	Expected Proficiency Percentage	Expected Attainment Percentage
<b>Outcome 1</b>	Describe the principles of software engineering, life cycle models	2	75%	70%
<b>Outcome 2</b>	Analyze the computing requirements to solve a given problem	3	75%	70%
<b>Outcome 3</b>	Demonstrate the importance of software modeling and modeling languages	3	70%	65%
<b>Outcome 4</b>	Illustrate the necessity of software testing and design test cases for a software	3	75%	70%
<b>Outcome 5</b>	Interpret Software maintenance and state the concepts of project management.	3	75%	70%

## Course Articulation Matrix (CLO) to Program Learning Outcomes (PLO)

CLOs	Program Learning Outcomes (PLO)														
	Engineering Knowledge	Problem Analysis	Design and Development	Analysis, Design and Research	Modern Tool and ICT Usage	Society and Multicultural Skills	Environment and Sustainability	Moral, and Ethical Awareness	Individual and Teamwork Skills	Communication Skills	Project Management and Finance	Self-Directed and Life Long Learning	PSO 1	PSO 2	PSO 3
Outcome 1	2	1	2	2	2						2	3	2	3	1
Outcome 2	2	3	2	3	3							3	3	3	2
Outcome 3	2	3	3	2	3							3	3	3	2
Outcome 4	2	3	3	2	3						3	3	3	3	2
Outcome 5	2	3	2	2	3						3	3	3	3	2
<b>Course Average</b>	<b>2</b>	<b>3</b>	<b>2</b>	<b>2</b>	<b>3</b>						<b>1</b>	<b>3</b>	<b>2</b>	<b>3</b>	<b>2</b>

## Course Unitization Plan

Unit No.	Unit Name	Required Contact Hours	CLOs Addressed	References Used
<b>Unit 1</b>	<b>Software Product and Software Process</b>	<b>7</b>		
	Software Product and Process Characteristics	1	1	1
	Software Process Models	1	1	1
	Perspective and Specialized Process Models	2	1	1
	Introduction to Agility	1	1	1
	Agile process	1	1	1,2
	Software Process customization and improvement	1	1	1
<b>Unit 2</b>	<b>Requirements Analysis and Specification</b>	<b>18</b>		
	Software Requirements: Functional and Non-Functional	1	2	1,2
	Requirement Sources and Elicitation Techniques	1	2	1,2
	Software Requirements Document	1	2	1,3
	Requirement Engineering Process: Feasibility Studies	1	2	1,3
	Requirements elicitation and analysis	1	2	1,2
	requirements validation, requirements management	1	2	1,2
	Classical analysis: Structured system Analysis	1	2	1,2
	Petri Nets- Data Dictionary.	1	2	1,3
	Lab Experiment: Develop requirements specification for a given problem	2	2	1,2,3
	Lab Experiment: Develop DFD Model (Level 0, Level 1 DFD and data dictionary) of the sample problem	2	2	1,2,3
	Lab Experiment: To perform the function oriented diagram : DFD and Structured chart	2	2	1,2,4
	Lab Experiment: To perform the user's view analysis : Use case diagram	2	2	1,2,4

	Lab Experiment: To perform the user's view analysis : Use case diagram Scenario's	2	2	1,2,4
<b>Unit 3</b>	<b>Software Design</b>	<b>27</b>		
	Design process and Design Concepts	1	3	1,4
	Design Model– Design Heuristic	1	3	2,3
	Architectural Design - Architectural styles,	1	3	1,5
	Architectural Design, Architectural Mapping using Data Flow- User Interface	2	3	1,2
	Design: Interface analysis, Interface Design	1	3	1,3
	Component level Design: Designing Class based components, traditional Components	2	3	1,4
	Lab Experiment: To draw the structural view diagram : Class diagram	2	3	1,4,5
	Lab Experiment: To draw the structural view diagram : Object diagram	2	3	1,4,5
	Lab Experiment: To draw the structural view diagram : Package diagram	2	3	1,4,5
	Lab Experiment: To draw the behavioral view diagram: Sequence diagram	2	3	1,4,5
	Lab Experiment: To draw the behavioral view diagram: Collaboration diagram	2	3	1,4,5
	Lab Experiment: To draw the behavioral view diagram: State-chart diagram	2	3	1,4,5
	Lab Experiment: To draw the behavioral view diagram: Activity diagram	2	3	1,4,5
	Lab Experiment: To draw the implementation view diagram: Component diagram	2	3	1,4,5
	Lab Experiment: To draw the environmental view diagram : Deployment diagram	2	3	1,4,5
<b>Unit 4</b>	<b>Testing and Maintenance</b>	<b>13</b>		
	Software testing fundamentals	1	4	1,2
	Internal and external views of Testing	1	4	1,3,4
	white box testing : Basis path testing-control structure testing	2	4	1,4
	black box testing- Regression Testing	2	4	1,5
	Unit Testing – Integration Testing – Validation Testing	1	4	1,3
	System Testing And Debugging	1	4	1,2
	Software Implementation Techniques: Coding practices- Refactoring	1	4	1,5
	Maintenance and Reengineering-BPR model	1	4	1,3
	Reengineering process model-Reverse and Forward Engineering.	1	4	1,2
	Lab Experiment: To perform various testing using the testing tool unit testing, integration testing	2	4	1,4
<b>Unit 5</b>	<b>Software Maintenance &amp; Software Project Measurement</b>	<b>10</b>		
	Software Configuration Management (SCM)	2	5	2,3
	Software Change Management	2	5	2,5

	Version Control, Change control and Reporting	2	5	1,3
	Re-engineering, Reverse Engineering	1	5	1,4
	Project Management Concepts	1	5	1,5
	Project Scheduling and Tracking	1	5	3
	Software Quality Assurance (SQA)	1	5	1

### Recommended Resources

1. Roger S. Pressman, Software Engineering – A Practitioner’s Approach, Seventh Edition, Mc Graw-Hill International Edition, 2010.
2. Ian Sommerville, Software Engineering, 9th Edition, Pearson Education Asia, 2011.
3. Rajib Mall, Fundamentals of Software Engineering, Third Edition, PHI Learning Private Limited, 2009.
4. Pankaj Jalote, Software Engineering, A Precise Approach, Wiley India, 2010. Kelkar S.A., Software Engineering, Prentice Hall of India Pvt Ltd, 2007.
5. Pankaj Jalote, “An Integrated Approach to Software Engineering”, Narosa Pub, 2005
6. Bob Hughes, M. Cotterell, Rajib Mall “Software Project Management”, McGraw Hill.

### Learning Assessment

Bloom’s Level of Cognitive Task		Continuous Learning Assessments (50%)				End Semester Exam (50%)	
		Theory (30%)		Practical (20%)			
		CLA-1 (5%)	Mid-1 (10%)	CLA-2 (5%)	Mid-2 (10%)	Internal	Th    Prac
Level 1	Remember	70%	60%	50%	40%	50%	30%    30%
	Understand						
Level 2	Apply	30%	40%	50%	60%	50%	70%    70%
	Analyse						
Level 3	Evaluate						
	Create						
<b>Total</b>		100%	100%	100%	100%	100%	100%

### Course Designers

- a. Dr. Sobin C C, Associate Professor, Computer Science Engineering, SRM University AP.