

## Software Engineering

<b>Course Code</b>	20CS5502	<b>Year</b>	III	<b>Semester</b>	I
<b>Course Category</b>	Minor	<b>Branch</b>	Other Branches	<b>Course Type</b>	Theory
<b>Credits</b>	4	<b>L-T-P</b>	4-0-0	<b>Prerequisites</b>	-
<b>Continuous Evaluation :</b>	30	<b>Semester End Evaluation:</b>	70	<b>Total Marks:</b>	100

### Course Outcomes

Upon successful completion of the course, the student will be able to		
<b>CO1</b>	Understand the fundamentals of software engineering	<b>L2</b>
<b>CO2</b>	Apply various lifecycle activities for project development	<b>L3</b>
<b>CO3</b>	Apply Risk and Quality Management Strategies for project development	<b>L3</b>
<b>CO4</b>	Analyze the various requirements, design and Testing Techniques to select the appropriate techniques for the software project development.	<b>L4</b>

### Contribution of Course Outcomes towards achievement of Program Outcomes & Strength of correlations (3:Substantial, 2: Moderate, 1:Slight)

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
<b>CO1</b>	2													
<b>CO2</b>	3								3	3			3	
<b>CO3</b>	3								3	3			3	
<b>CO4</b>		3							3	3			3	

Syllabus		Mapped CO
Unit No.	Contents	
I	Introduction to Software Engineering: The Nature of Software, The changing Nature of Software, Software Process, Software Engineering Practice, Software Development myths. Process models: Prescriptive process Models, Unified Process Model, <b>Agile process model</b>	CO1,CO2
II	Requirements engineering Requirements engineering, Establishing the Groundwork, Eliciting Requirements, Developing Use Cases, Building the Analysis Model, Negotiating requirements. Agile Requirements. Requirements Analysis: Scenario Based Modeling.	CO1,CO2, CO4
III	Design Engineering: Design process, Design concepts, The Design Model. Architectural design: Software architecture, Architectural Design, Agility and Architecture.	CO1,CO2, CO4
IV	Software Testing Strategies: A strategic approach to software testing, Test strategies for conventional software, Validation Testing, System Testing, The art of Debugging. Testing Conventional Applications: Software testing fundamentals, White-Box testing, Basis path testing, Control structure testing, Black-Box testing.	CO1,CO2,CO4
V	Risk management: Reactive vs. Proactive Risk strategies, software risks, Risk identification, Risk projection, Risk refinement, RMMM, RMMM Plan. Software Quality Assurance: Elements of SQA, SQA Tasks, Goals, and Metrics, Statistical Software Quality Assurance, Software Reliability, The ISO 9000 Quality Standards, The SQA Plan.	CO1,CO3,CO4

### Learning Resources

#### Text Books

1. Software Engineering: A Practitioner's Approach, Roger S. Pressman, Bruce R. Maxim, Eighth edition, 2015, McGraw Hill, International Edition.

#### References

1. Robert C. Martin ,Agile Software Development, Principles, Patterns, and Practices Alan Apt Series (2011)
2. Software Engineering, Ian Sommerville, Seventh edition, 2004, Pearson,India
3. Software Engineering, K.K. Agarwal&Yogesh Singh, 2007, New Age International Publishers.
4. Software Engineering Principles and Practice, Waman S Jawadekar, 2004, McGraw Hill.
5. Fundamentals of Software Engineering, Rajib Mall, Fourth edition, 2009, PHI.
6. Succeeding with Agile : Software Development Using Scrum, Pearson (2010)

#### e-Resources & other digital material

1. [https://onlinecourses.nptel.ac.in/noc20\\_cs68](https://onlinecourses.nptel.ac.in/noc20_cs68)
2. <https://thedigitalprojectmanager.com>