

<b>Course Code:</b>	<b>Course Title</b>	<b>Credit</b>
<b>CSC502</b>	<b>Software Engineering</b>	<b>3</b>

<b>Prerequisite:</b> Object Oriented Programming with Java , Python Programming	
<b>Course Objectives:</b>	
1	To provide the knowledge of software engineering discipline.
2	To apply analysis, design and testing principles to software project development.
3	To demonstrate and evaluate real world software projects.
<b>Course Outcomes:</b> On successful completion of course, learners will be able to:	
1	Identify requirements & assess the process models.
2	Plan, schedule and track the progress of the projects.
3	Design the software projects.
4	Do testing of software project.
5	Identify risks, manage the change to assure quality in software projects.

<b>Module</b>	<b>Content</b>	<b>Hrs</b>
<b>1</b>	<b>Introduction To Software Engineering and Process Models</b>	<b>7</b>
	1.1 Software Engineering-process framework, the Capability Maturity Model (CMM), Advanced Trends in Software Engineering	
	1.2 Prescriptive Process Models: The Waterfall, Incremental Process Models, Evolutionary Process Models: RAD & Spiral	
	1.3 Agile process model: Extreme Programming (XP), Scrum, Kanban	
<b>2</b>	<b>Software Requirements Analysis and Modeling</b>	<b>4</b>
	2.1 Requirement Engineering, Requirement Modeling, Data flow diagram, Scenario based model	
	2.2 Software Requirement Specification document format(IEEE)	
<b>3</b>	<b>Software Estimation Metrics</b>	<b>7</b>
	3.1 Software Metrics, Software Project Estimation (LOC, FP, COCOMO II )	
	3.2 Project Scheduling & Tracking	
<b>4</b>	<b>Software Design</b>	<b>7</b>
	4.1 Design Principles & Concepts	
	4.2 Effective Modular Design, Cohesion and Coupling, Architectural design	
<b>5</b>	<b>Software Testing</b>	<b>7</b>
	5.1 Unit testing, Integration testing, Validation testing, System testing	
	5.2 Testing Techniques, white-box testing: Basis path, Control structure testing black-box testing: Graph based, Equivalence, Boundary Value	
	5.3 Types of Software Maintenance, Re-Engineering, Reverse Engineering	
<b>6</b>	<b>Software Configuration Management, Quality Assurance and Maintenance</b>	<b>7</b>
	6.1 Risk Analysis & Management: Risk Mitigation, Monitoring and Management Plan (RMMM).	
	6.2 Quality Concepts and Software Quality assurance Metrics, Formal Technical Reviews, Software Reliability	
	6.3 The Software Configuration Management (SCM) ,Version Control and Change Control	
		<b>39</b>

<b>Textbooks:</b>	
1	Roger Pressman, " <i>Software Engineering: A Practitioner's Approach</i> ", 9 <sup>th</sup> edition, McGraw-Hill Publications, 2019
2	Ian Sommerville, " <i>Software Engineering</i> ", 9 <sup>th</sup> edition, Pearson Education, 2011
3	Ali Behfroz and Fredeick J. Hudson, " <i>Software Engineering Fundamentals</i> ", Oxford University Press, 1997
4	Grady Booch, James Rumbaugh, Ivar Jacobson, " <i>The unified modeling language user guide</i> ", 2 <sup>nd</sup> edition, Pearson Education, 2005
<b>References:</b>	
1	Pankaj Jalote, " <i>An integrated approach to Software Engineering</i> ", 3 <sup>rd</sup> edition, Springer, 2005
2	Rajib Mall, " <i>Fundamentals of Software Engineering</i> ", 5 <sup>th</sup> edition, Prentice Hall India, 2014
3	Jibitesh Mishra and Ashok Mohanty, " <i>Software Engineering</i> ", Pearson, 2011
4	Ugrasen Suman, " <i>Software Engineering – Concepts and Practices</i> ", Cengage Learning, 2013
5	Waman S Jawadekar, " <i>Software Engineering principles and practice</i> ", McGraw Hill Education, 2004

<b>Assessment:</b>	
<b>Internal Assessment:</b>	
Assessment consists of two class tests of 20 marks each. The first-class test is to be conducted when approx. 40% syllabus is completed and the second-class test when an additional 40% syllabus is completed. Duration of each test shall be one hour.	
<b>End Semester Theory Examination:</b>	
1	Question paper will comprise a total of six questions.
2	All question carries equal marks
3	Only Four questions need to be solved.
4	In question paper weightage of each module will be proportional to number of respective lecture hours as mentioned in the syllabus.

<b>Useful Links</b>	
1	<a href="https://nptel.ac.in/courses/106/105/106105182/">https://nptel.ac.in/courses/106/105/106105182/</a>
2	<a href="https://onlinecourses.nptel.ac.in/noc19_cs69/preview">https://onlinecourses.nptel.ac.in/noc19_cs69/preview</a>
3	<a href="https://www.mooc-list.com/course/software-engineering-introduction-edx">https://www.mooc-list.com/course/software-engineering-introduction-edx</a>

Lab Code	Lab Name	Credit
CSL501	Software Engineering Lab	1

**Prerequisite:** Object Oriented Programming with Java , Python Programming

**Lab Objectives:**

- |   |   |
|---|---|
| 1 | To solve real life problems by applying software engineering principles |
| 2 | To impart state-of-the-art knowledge on Software Engineering            |

**Lab Outcomes:** On successful completion of laboratory experiments, learners will be able to :

- |   |  |
|---|--|
| 1 | Identify requirements and apply software process model to selected case study. |
| 2 | Develop architectural models for the selected case study.                      |
| 3 | Use computer-aided software engineering (CASE) tools.                          |

**Suggested List of Experiments** - Assign the case study/project as detail statement of problem to a group of two/three students. Laboratory work will be based on course syllabus with minimum 10 experiments. Open source computer-aided software engineering (CASE) tools can be used for performing the experiment.

Sr. No.	Title of Experiment
1	Application of at least two traditional process models.
2	Application of the Agile process models.
3	Preparation of software requirement specification (SRS) document in IEEE format.
4	Structured data flow analysis.
5	Use of metrics to estimate the cost.
6	Scheduling & tracking of the project.
7	Write test cases for black box testing.
8	Write test cases for white box testing.
9	Preparation of Risk Mitigation, Monitoring and Management Plan (RMMM).
10	Version controlling of the project.

**Term Work:**

- |   |  |
|---|--|
| 1 | Term work should consist of 10 experiments.  |
| 2 | Journal must include at least 2 assignments on content of theory and practical of “Software Engineering”   |
| 3 | The final certification and acceptance of term work ensures that satisfactory performance of laboratory work and minimum passing marks in term work. |
| 4 | Total 25 Marks (Experiments: 15-marks, Attendance Theory & Practical: 05-marks, Assignments: 05-marks)   |

**Oral & Practical exam**

Based on the entire syllabus of CSC502 and CSL501 syllabus
--