

SOFTWARE ENGINEERING [As per Choice Based Credit System (CBCS) scheme] (Effective from the academic year 2016 -2017) SEMESTER – IV			
Subject Code	15CS42	IA Marks	20
Number of Lecture Hours/Week	04	Exam Marks	80
Total Number of Lecture Hours	50	Exam Hours	03
CREDITS – 04			
Course objectives: This course will enable students to <ul style="list-style-type: none"> • Outline software engineering principles and activities involved in building large software programs. • Identify ethical and professional issues and explain why they are of concern to software engineers. • Describe the process of requirements gathering, requirements classification, requirements specification and requirements validation. • Differentiate system models, use UML diagrams and apply design patterns. • Discuss the distinctions between validation testing and defect testing. • Recognize the importance of software maintenance and describe the intricacies involved in software evolution. • Apply estimation techniques, schedule project activities and compute pricing. • Identify software quality parameters and quantify software using measurements and metrics. • List software quality standards and outline the practices involved. • Recognize the need for agile software development, describe agile methods, apply agile practices and plan for agility. 			
Module 1			Teaching Hours
Introduction: Software Crisis, Need for Software Engineering. Professional Software Development, Software Engineering Ethics. Case Studies. Software Processes: Models: Waterfall Model (Sec 2.1.1), Incremental Model (Sec 2.1.2) and Spiral Model (Sec 2.1.3). Process activities. Requirements Engineering: Requirements Engineering Processes (Chap 4). Requirements Elicitation and Analysis (Sec 4.5). Functional and non-functional requirements (Sec 4.1). The software Requirements Document (Sec 4.2). Requirements Specification (Sec 4.3). Requirements validation (Sec 4.6). Requirements Management (Sec 4.7).			12 Hours
Module 2			
System Models: Context models (Sec 5.1). Interaction models (Sec 5.2). Structural models (Sec 5.3). Behavioral models (Sec 5.4). Model-driven engineering (Sec 5.5). Design and Implementation: Introduction to RUP (Sec 2.4), Design Principles (Chap 17). Object-Oriented design using the UML (Sec 7.1). Design patterns (Sec 7.2). Implementation issues (Sec 7.3). Open source development (Sec 7.4).			11 Hours
Module 3			
Software Testing: Development testing (Sec 8.1), Test-driven development (Sec 8.2), Release testing (Sec 8.3), User testing (Sec 8.4). Test Automation (Page no 42, 70,212, 231,444,695). Software Evolution: Evolution processes (Sec 9.1). Program evolution dynamics (Sec 9.2). Software maintenance (Sec 9.3). Legacy system management (Sec 9.4).			9 Hours

Module 4	
Project Planning: Software pricing (Sec 23.1). Plan-driven development (Sec 23.2). Project scheduling (Sec 23.3): Estimation techniques (Sec 23.5). Quality management: Software quality (Sec 24.1). Reviews and inspections (Sec 24.3). Software measurement and metrics (Sec 24.4). Software standards (Sec 24.2)	10 Hours
Module 5	
Agile Software Development: Coping with Change (Sec 2.3), The Agile Manifesto: Values and Principles. Agile methods: SCRUM (Ref “ The SCRUM Primer, Ver 2.0 ”) and Extreme Programming (Sec 3.3). Plan-driven and agile development (Sec 3.2). Agile project management (Sec 3.4), Scaling agile methods (Sec 3.5):	8 Hours
Course Outcomes: After studying this course, students will be able to:	
<ul style="list-style-type: none"> • Design a software system, component, or process to meet desired needs within realistic constraints. • Assess professional and ethical responsibility • Function on multi-disciplinary teams • Use the techniques, skills, and modern engineering tools necessary for engineering practice • Analyze, design, implement, verify, validate, implement, apply, and maintain software systems or parts of software systems. 	
Graduate Attributes	
<ul style="list-style-type: none"> • Project Management and Finance • Conduct Investigations of Complex Problems • Modern Tool Usage • Ethics 	
Question paper pattern:	
<p>The question paper will have ten questions. There will be 2 questions from each module. Each question will have questions covering all the topics under a module. The students will have to answer 5 full questions, selecting one full question from each module.</p>	
Text Books:	
<ol style="list-style-type: none"> 1. Ian Sommerville: Software Engineering, 9th Edition, Pearson Education, 2012. (Listed topics only from Chapters 1,2,3,4, 5, 7, 8, 9, 23, and 24) 2. The SCRUM Primer, Ver 2.0, http://www.goodagile.com/scrumprimer/scrumprimer20.pdf 	
Reference Books:	
<ol style="list-style-type: none"> 1. Roger S. Pressman: Software Engineering-A Practitioners approach, 7th Edition, Tata McGraw Hill. 2. Pankaj Jalote: An Integrated Approach to Software Engineering, Wiley India 	
Web Reference for eBooks on Agile:	
<ol style="list-style-type: none"> 1. http://agilemanifesto.org/ 2. http://www.jamesshore.com/Agile-Book/ 	