

## COURSE HANDOUT

### B.Tech- 4<sup>th</sup> Semester

Course Title : **Software Engineering**

Dated: 28-6-2017

Course Code : **16CS404**

Academic Year 2017-18

Course Structure : 3-0-2-4

**Course coordinator : Dr.A.V.Ramana**

**Instructor(s) : Dr.A.V.Ramana, Dr.B.D.Deebak, Mr.AjitKumar Routh and Mr.A.K.Sahu**

#### **Course Description:**

This Course is a comprehensive Study of Software engineering. In this course students will gain a broad understanding of the discipline of software engineering and its application to the development of and management of software systems. This course also describes knowledge of basic SW engineering methods and practices, and their appropriate application, a general understanding of software process models such as the waterfall and evolutionary models, an understanding of implementation issues such as modularity and coding standards. It also describes the understanding of software testing approaches such as unit testing and integration testing.

#### **Scope and Objective:**

The main objectives of the course are to:

1. Understand the software engineering principles and Software life cycle models
2. Understanding of software requirements and the SRS documents.
3. Describe data models, object models, context models and behavioral models.
4. Understanding of different software architectural styles.
5. Understanding of implementation issues such as modularity and coding standards
6. Understanding of software testing approaches such as unit testing and integration testing.

#### **Text Books:**

1. Rajib Mal, Fundamentals of software Engineering, 3rdEdition, Eastern Economy Edition, 2009
2. Roger S. Pressman, Software Engineering, A practitioner"s Approach, 6thEdition, McGraw-Hill International Edition, 2005

#### **Reference(s)**

1. I. Sommerville, Software Engineering, 7thEdition, Pearson education, 2004
2. K K Aggarwal and Yogesh singh, Software engineering,3rd Edition, New age international publication,2008

## **SYLLABUS:**

### **Unit I**

**12+6 Hours**

#### **Introduction to Software Engineering and Life Cycle Models**

Software engineering definitions, software development projects, the evolving role of software, Changing Nature of Software, Software myths. Perspective, evolutionary, specialized (agile), Unified software process model

*Business Process Engineering Overview - Product Engineering Overview*

#### **Practical Components**

1. Suggest the suitable life cycle model for credit card validation system
2. Perform Agile Testing

### **Unit II**

**11+12 Hours**

#### **Software Requirements Engineering Process and System Models**

Requirement classification, Feasibility studies, elicitation and analysis, validation and change management, Context Models, Behavioral model1, Data models, Object models

*Software Prototyping – Prototyping in the Software Process*

#### **Practical Components**

1. Prepare the requirement analysis and SRS document
2. Estimate the effort using COCOMO Model
3. Develop DFD and Context model for given problem

### **Unit III**

**10+4 Hours**

#### **Design Engineering**

Design concepts, the design model, Creating an architectural design - Software architecture, Architectural styles and patterns, Performing User interface design - Golden rules, User interface analysis and design and steps

*Data Acquisition System - Monitoring and Control System*

#### **Practical Components**

1. Suggest architecture style for internet banking system

### **Unit IV**

**11+9 Hours**

#### **Testing and Risk Management**

Unit testing, integration testing, system testing, object oriented programs testing, black box and white box testing, debugging, Risk management - Risk types, strategies, estimation and Planning. Software Quality - Quality assurance and its techniques

*Software Implementation Techniques - Testing as an Engineering Activity*

#### **Practical Components**

1. Prepare Test suite
2. Perform code generation using automated tools

**Total:45+30=75 Hours**

## **Course Outcomes:**

After undergoing the course students will be able to:

1. Illustrate the need of Software Life Cycle Models
2. Demonstrate the Requirements of the Software Systems process
3. Summarize the system models of software engineering
4. Choose appropriate software architecture style for real-time software projects
5. Analyze Risk management and Software quality of the software products
6. Identify the Illustrate and Find quality assurance techniques.

## Course Plan:

No. Lecture	Learning objectives	Topic(s) to be covered	Chapter in the textbook/reference
<b>UNIT – I</b>			
<b>Introduction to Software Engineering and Life Cycle Models</b>			
1	To study the basic terminologies of software engineering	Software Engineering Definitions	T1,Chapter 1
2	To broadly categories of computer software present continuing challenges for software engineers	Software Development Projects: Product Based and Service Based	T1,Chapter 1
3	To helps the architect focus on the right goals, such as understandability, reliance to future changes, and reuse”	Evolving Role of Software	T1,Chapter 1
4&5	To assist in the definition, execution, and management of prescriptive process models.		
6	To learn how to delivers the computing potential embodied by computer hardware or more broadly, by a network of computers that are accessible by local hardware	Changing Nature of Software	T1,Chapter 1
7	To learn how to maintain budgets, keep schedules from slipping, and improve quality	Software Myths	T2,Chapter 2
8	To learn how to make consumer products for home and commercial use	How a Project Starts	T2,Chapter 2
9&10	Umbrella activities occur throughout the software process. Do you think they are applied evenly across the process, or are some concentrated in one or more framework activities?		
11	To help in the application of a meaningful software process and the execution of effective software engineering methods.	Perspective	T2,Chapter 7
12	To study on how the activity encompassed into three actions: delivery, support, and feedback	Evolutionary Models	T1,Chapter 7
13	To provide provides a framework for software engineering practice	Software Process Model	T2,Chapter 3
13	To learn how to characterize the traditional model when a specialized or narrowly defined software engineering approach is chosen	Specialized Process Models	T2,Chapter 4
14&15	Provide three examples of software projects that would be amenable to the prototyping model. B specific. As you move outward along the spiral process flow, what can you say about the software that is being developed or maintained?		
16	To learn how to differentiate Working Software over Comprehensive Documentation	Agile Development	T2,Chapter 5
17	To learn how computers become more powerful every year, leading users to expect	Unified Software Process Model	T2,Chapter 4

	more from them. This trend		
18	Business Process Engineering Overview Product Engineering Overview		
<b>UNIT – II</b> <b>Software Requirements Engineering Process and System Models</b>			
19	To introduce the concepts of user and system requirements & To explain how software requirements may be organized in a requirements document	Requirement Classification	R1,Chapter 5
20	To learn how to discover, analyse and validate system requirements	Feasibility Studies, Management	R1,Chapter 6
21 & 22	A feasibility study is a short, focused study that should take place early in the RE process. It should answer three key questions: a) does a system contributes to the overall objectives of the organization? b) Can the system be implemented within schedule and budget using current technology? and c) can the system be integrated with other systems that are used?		
23	To find out about the application domain, what services the system should provide, the required performance of the system, hardware constraints	Requirements Elicitation	R1,Chapter 4
24&25	Discover ambiguities or omissions in the following statement of requirements for part of a ticket-issuing system: An automated ticket-issuing system sells rail tickets. Users select their destination and input a credit card and a personal identification number. The rail ticket is issued and their credit card account charged. When the user presses the start button, a menu display of potential destinations is activated, along with a message to the user to select a destination. Once a destination has been selected, users are requested to input their credit card. Its validity is checked and the user is then requested to input a personal identifier. When the credit transaction has been validated, the ticket is issued.		
26	To learn how to extensively rework in cost when the problems are discovered during development or after the system is in service	Requirements Validation	R1,Chapter 4
27	To learn how to decide what functionality should be included in the system and what is provided by the system's environment	Context Models Object Model Data Model	R1,Chapter 5
28	To learn what happens or what is supposed to happen when a system responds to a stimulus from its environment	Behavioral Model	R1,Chapter 5
29&30	You have been asked to develop a system that will help with planning large-scale events and parties such as weddings, graduation celebrations, birthday parties, etc. Using an activity diagram, model the process context for such a system that shows the activities involved in planning a party (booking a venue, organizing invitations, etc.) and the system elements that may be used at each stage.		
31	to demonstrate concepts, try out design options, and find out more about the problem and its possible solutions	Software Prototyping & Prototyping in the Software Process	R1,Chapter 2
<b>UNIT – III</b>			

	Design Engineering		
32	To learn the set of principles, concepts, and practices that lead to the development of a high-quality system or product	Design concepts	T2, Chapter 12
33	To learn how to design at technical kernel of software engineering and is applied regardless of the software process model that is used	Design Model	T2, Chapter 12
34&35	Assume Jamie’s cubicle, as the team prepares to translate requirements into design. Prepare Design Versus Coding statements		
36	To learn how Methods required to create “coherent, well-planned representations” of the data and architectural layers of the design model	Creating an architectural design	T2, Chapter 13
37		Software Architecture	
38		Architectural Styles	
39		Architectural Patterns	
40	To learn how consumer electronics, industrial equipment, automobiles, corporate systems, military systems, personal computer software, mobile apps, and WebApps—require human interaction	Performing User Interface Design	T2, Chapter 13
41		Golden Rules	
42		User Interface Analysis	
43		Interface Design Steps	
44	To learn how to use in scientific experiments and process control systems where physical processes, such as chemical reactions, are very rapid	Data Acquisition System	R1, Chapter 20
45	To learn how to design step occurs a number of times, elaborating and refining information developed in the preceding step	Monitoring and Control System	R1, Chapter 15
46	Develop (1) an elaborated design class, (2) interface descriptions, (3) an activity diagram for one of the operations within the class, and (4) a detailed state chart diagram for one of the Safe Home classes		
	UNIT – IV Testing and Risk Management		
47	To suggest a set of testing principles	Testing Principles	T2, Chapter 2
48	Do some research on “negotiation” for the communication activity and prepare a set of guidelines that focus solely on negotiation		
49	To learn how to plan a set of activities that can be planned in advance and conducted systematically	Strategic Approach To Software Testing	T2, Chapter 22
50	To learn an inherent conflict of interest that occurs as testing begins	Organizing for Software Testing	T2, Chapter 22

51	To learn how to define the role of software and leads to software requirements analysis, where the information domain, function, behavior, performance, constraints, and validation criteria for software	Software Testing Strategy	T2, Chapter 22
52&53	Develop a complete test strategy for the Safe Home system discussed along with its Test Specification.		
54	To present a systematic strategy for software testing	Strategic Issues	T2, Chapter 22
55	To uncover errors within the boundary of the module.	Unit Testing	T2, Chapter 22
56	To take unit-tested components and build a program structure that has been dictated by design	Integration Testing	T2, Chapter 22
57	To learn how to improve the probability of successful software integration in the larger system	System Testing	T2, Chapter 22
58	Is unit testing possible or even desirable in all circumstances? Provide examples to justify your answer.		
59	To understand the object-oriented point of view	Object Oriented Programs Testing	T2, Chapter 22
60	To enable how to partition a system hierarchically, moving from essential representation at the top to implementation-specific detail at the bottom	Black Box And White Box Testing	T2, Chapter 28
61	To learn how to begin with the execution of a testcase	Debugging	T2, Chapter 22
61&62	Add three additional questions or topics to each of the risk item checklists presented at the SEPA website.		
63	To eliminate risk, and questionable to try to minimize it, it is essential that the risks taken be the right risks.”	Risk Management	T2, Chapter 35
64	To specify threats to the project plan (estimates, schedule, resource loading, etc.). To refine the risk into a set of more detailed risks, each somewhat easier to mitigate, monitor, and manage.	Risk Types, Strategies, Estimation and Planning	T2, Chapter 35
65&66	You’re the project manager for a major software company. You’ve been asked to lead a team that’s developing “next generation” word-processing software. Create a risk table for the project.		
67	Describe the difference between risk components and risk drivers		
68	To introduce the software quality management and software measurement.	Software Quality	T2, Chapter 24
69	To be aware of the importance of standards in the quality management process and know how standards are used in quality Assurance	Quality assurance and its Techniques	T2, Chapter 24
70	To understand how measurement may be helpful in assessing some software quality	Software standards	T2, Chapter 24



	attributes and the current limitations of software measurement		
71&72	Design an electronic form that may be used to record review comments and which could be used to electronically mail comments to reviewers.		
71	To reuse and proposes a new approach, based on aspects, to organizing and structuring software systems	Software Implementation Techniques -	T2, Chapter 21
72&73	To describe a series of tasks required to accomplish a software engineering activity.	Testing as an Engineering Activity	T2, Chapter 25
74&75	Assume you work for an organization that develops database products for individuals and small businesses. This organization is interested in quantifying its software development. Write a report suggesting appropriate metrics and suggest how these can be collected.		

## Evaluation scheme:

Component	Duration (minutes)	Marks	% of weightage	Date & Time	Venue
Sessional Test – 1	90	20	20 Marks (80 % of marks secured in 1 <sup>st</sup> best internal tests and 20% marks secured in 2 <sup>nd</sup> best internal test)	29-01-2018 to 03-02-2018 (3:15P.M. to 04:45P.M.)	Block-5
Sessional Test – 2	90	20		26-03-2018 to 31-03-2018 (3:15P.M. to 04:45P.M.)	Block-5
Laboratory					
20 marks are awarded for continuous assessment and following is the pattern for the award of 20 marks					
Preparation, Observation & Result: 10 Marks					
Record : 05 Marks					
Internal Test : 05 Marks					
External Exam(Theory)	180	60			
External Exam(Lab)	180	30			

**Chamber Consultation Hour:** 4 PM to 5PM

**Venue:** Ground Floor staff room (Block-6)

**Notices:** Main notice board

**Signature of the Instructor**

**Signature of the course-coordinator**