Prasad V. Potluri Siddhartha Institute of Technology, Kanuru, Vijayawada.

1/2 M.Tech. SECOND SEMESTER

CSCS2T2 **SOFTWARE ENGINEERING** **Credits: 4**

Lecture: 4 periods/week **Internal assessment: 30 marks**

Tutorial: 1 period /week **Semester end examination: 70 marks**

Objectives:

1. An understanding of different software processes and how to choose between them
2. How to elicit requirements from a client and specify them
3. Designing the large, including principled choice of software architecture, the use of modules and interfaces to enable separate development, and design patterns.
4. Understanding good coding practices, including documentation, contracts, regression tests and daily builds.
5. Various quality assurance techniques, including unit testing, functional testing, and automated analysis tools.
6. Understanding of software management including planning/scheduling
7. Understanding general case tools

Learning Outcomes:

1. They are expected to realize the software myths and the list the applications of software engineering.
2. They are expected to learn the various process models and their differentiation and choose a process model for mentioned application.
3. They are expected to learn various categories of requirements, identifies, differentiate and group the requirements for mentioned CASE study and prepare a SRS document.
4. They are expected to Implement the acquired knowledge in developing the strategic testing approach by writing the test cases for the given software code.
5. They are expected to Identify the various metrics used to estimate the quality at every stage of development and prepares the RMM plan that should be adopted in the process of development for a given simple case.
6. They are expected to understand the process and importance of technical reviews and also the ISO standards for maintaining the quality.

UNIT - I

Introduction to Software Engineering: Software Characteristics, Software Application Domains, Software myths.

A Generic view of process: Software engineering- A layered technology, a process framework, The Capability Maturity Model Integration (CMMI), Process patterns, process assessment,

UNIT - II

Process Models: The waterfall model, Incremental process models, Evolutionary process models, The Unified process, Agile process models

Software Requirements: Functional and non-functional requirements, User requirements, System requirements, Interface specification, the software requirements document.

UNIT – III

Requirements Engineering Process: Feasibility studies, Requirements elicitation and analysis, Requirements validation, Requirements management.

System models: Context Models, Behavioral models Data models, Object models, structured methods.

UNIT – IV

Design Engineering: Design process and Design quality, Design concepts

Creating an architectural design: Software architecture, Data design, Architectural styles and patterns,

User Interface Design: Characteristics of a good user interface, Types of interfaces, User Interface Design Methodology

UNIT – V

Object Modeling using UML: Overview of basic object-orientation concepts, Identification of objects, Boochs Object Identification Method, Unified Modeling Language, UML Diagrams, Use-case Model, Class Diagrams, Sequence Diagrams, Activity Diagrams, State Chart Diagram, Package & Deployment diagram

UNIT – VI

Testing Strategies: A strategic approach to software testing, test strategies for conventional software, Black-Box and White-Box testing, Validation testing, System testing, the art of Debugging

Product metrics: Software Quality, Metrics for Analysis Model, Metrics for Design Model, Metrics for source code, Metrics for testing, Metrics for maintenance.

UNIT – VII

Metrics for Process and Products: Software Measurement, Metrics for software quality

Risk management: Reactive vs. Proactive Risk strategies, software risks, Risk identification, Risk projection, Risk refinement, RMMM, RMMM Plan

UNIT - VIII

Quality Management : Quality concepts, Software quality assurance, Software Reviews, Formal technical reviews, Statistical Software quality Assurance, Software reliability, The ISO 9000 quality standards

Learning Resources

Text Books:

1. Software Engineering, 7/e , Roger S.Pressman , TMH
2. Software Engineering, 8/e, Sommerville, Pearson.
3. Fundamentals of Software Engineering ,3/e, Rajib Mall, PHI