

III B.Tech I Semester

15ACS21-SOFTWARE ENGINEERING

L T P C

3 1 0 3

Course Objectives:

- To understand the software life cycle models.
- To understand the software requirements and SRS document.
- To understand the importance of modeling and modeling languages.
- To design and develop correct and robust software products.
- To understand the quality control and how to ensure good quality software.
- To understand the planning and estimation of software projects.
- To understand the implementation issues, validation and verification procedures.
- To understand the maintenance of software

UNIT I

Software and Software Engineering: The Nature of Software, The Unique Nature of Web Apps, Software Engineering, Software Process, Software Engineering Practice, Software Myths.

Process Models: A Generic Process Model, Process Assessment and Improvement, Prescriptive Process Models, Specialized Process Models, The Unified Process, Personal and Team Process Models, Process Terminology, Product and Process, Agile Process model, Extreme Programming, A comparison of different process models.

UNIT II

Software Project Planning and Management: Responsibilities of a Software Project Manager, Project Planning, Metrics for Project Size Estimation, Project Estimation Techniques, Empirical Estimation Techniques, COCOMO-A Heuristic Estimation Technique, Halstead's Software Science-An Analytical Technique, Staffing Level Estimation, Scheduling, Organization and Team Structures, Staffing, Risk Management, Software Configuration Management.

UNIT III

Understanding Requirements: Requirements Engineering, Establishing the Groundwork, Eliciting Requirements, Requirements Analysis, Structured Analysis, Data Oriented Analysis, Object oriented Analysis Developing Use Cases, Building the Requirements Model, Negotiating Requirements, and Validating Requirements.

Requirements Modeling: Requirements Analysis, Scenario-Based Modeling, UML Models That Supplement the Use Case, Data Modeling Concepts, Class-Based Modeling.

UNIT IV

Design Concepts: Design within the Context of Software Engineering, Design Process, Design Concepts, The Design Model, Function oriented software design, Object oriented software development.

Architectural Design: Software Architecture, Architectural Genres, Architectural Styles, Architectural Design.

Component-Level Design: What is a Component, Designing Class-Based Components, Conducting Component-Level Design, Component-Level Design for WebApps

UNIT V

User Interface Design: The Golden Rules, User Interface Analysis and Design, Interface Analysis, Interface Design Steps, Design Evaluation.

Coding and Testing: Coding, Code Review, Software Documentation, Testing, Testing in the Large versus Testing in the Small, Unit Testing, Black-Box Testing, White-Box Testing, Debugging, Program Analysis Tools, Integration Testing, Testing Object-Oriented Programs, System Testing, Some General Issues Associated with Testing.

Software Maintenance: Characteristics of Software Maintenance, Software Reverse Engineering, Software Maintenance Process Models, Estimation of Maintenance cost.

Course Outcomes:

- Define and develop a software project from requirement gathering to implementation.
- Obtain knowledge about principles and practices of software engineering.
- Focus on the fundamentals of modeling a software project.
- Obtain knowledge about estimation and maintenance of software systems

TEXT BOOKS :

1. Software Engineering A practitioner's Approach, Roger S. Pressman, Seventh Edition, 2009, McGrawHill International Edition.
2. Fundamentals of Software Engineering, Rajib Mall, Fourth Edition, 2014, PHI.
3. Software Engineering: Concepts & Practices- Ugrasen Suman, Cengage Learning publications

REFERENCE BOOKS:

1. Software Engineering, Ian Sommerville, Ninth edition, Pearson education.
2. Software Engineering : A Primer, Waman S Jawadekar, Tata McGraw-Hill, 2008
3. Software Engineering, A Precise Approach, Pankaj Jalote, Wiley India, 2010.
4. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.
5. Software Engineering 1: Abstraction and modeling, Diner Bjorner, Springer International Edition, 2006.
6. Software Engineering 2: Specification of systems and languages, Diner Bjorner, Springer International edition, 2006.
7. Software Engineering Foundations, Yingxu Wang, Auerbach Publications, 2008.
8. Software Engineering Principles and Practice, Hans Van Vliet, 3rd edition, John Wiley & Sons Ltd.
9. Software Engineering 3: Domains, Requirements, and Software Design, D. Bjorner, Springer International Edition.
10. Introduction to Software Engineering, R.J. Leach, CRC Press.