

**SEMESTER-III**  
**COURSE 8: SOFTWARE ENGINEERING**

Theory

Credits: 3

3 hrs/week

**Course Objectives:** The Objective of the course is to assist the student in understanding the basic theory of software engineering, and to apply these basic theoretical principles to a group software development project.

**Course Outcomes**

1. Ability to gather and specify requirements of the software projects.
2. Ability to analyze software requirements with existing tools
3. Able to differentiate different testing methodologies
4. Able to understand and apply the basic project management practices in real life projects
5. Ability to work in a team as well as independently on software projects

**Syllabus**

**UNIT I**

Introduction to Software Engineering: Definitions - Size Factors - Quality and Productivity Factors – Managerial Issues.

Planning a software project: Defining the problem - Developing a Solution Strategy - Planning the Development Process - Planning an Organization structure - Other Planning Activities.

**UNIT – II**

Software Cost Estimation: Software cost factors - Software Cost.

Estimation Techniques – Staffing level Estimation- Estimating Software Maintenance Costs - The Software Requirements, Specification - Formal Specification Techniques - Languages and Processors for Requirements Specification.

**UNIT – III**

Software design: Fundamental Design Concepts - Modules and Modularization Criteria – Design Notations -Design Techniques - Detailed Design Considerations.

Real-Time and Distributed System Design - Test Plans - Milestones, walkthroughs, and Inspections.

**UNIT IV**

User interface design and real time systems: User interface design - Human factors - Human computer interaction - Human - Computer Interface design - Interface design - Interface standards.

**UNIT V**

Software quality and testing: Software Quality Assurance - Quality metrics - Software Reliability - Software testing - Path testing – Control Structures testing - Black Box testing - Integration, Validation and system testing - Reverse Engineering and Reengineering.

CASE Tools: Projects management, tools - analysis and design tools – programming tools - integration and testing tool - Case studies.

**REFERENCE BOOKS:**

1. R.Fairley, Software Engineering Concepts, Tata McGraw-Hill, 1997.
2. R.S. Pressman, Software Engineering, Fourth Ed., McGraw Hill, 1997.
3. Software Engineering, H. Sommerville Ian , Addison Wesley Pub. Co.
4. Software Engineering: An object Oriented Perspective by Braude, E.J., Willey, 2001

**Student Activity:**

1. Visit any financial organization nearby and prepare requirement analysis report
2. Visit any industrial organization and prepare risk chart

**SEMESTER-III**  
**COURSE 8: SOFTWARE ENGINEERING**

Practical

Credits: 1

2 hrs/week

---

**(Using Object Oriented Analysis and Design (OOAD))**

**Case Studies:**

1. Student Marks Analysis System
2. E-Commerce Management System
3. Inventory Control System
4. Food Delivery Management system
5. Logistics Management System

**Choose any two of above case studies and do the following exercises for that Case Study**

1. Write the complete problem statement
2. Write the software requirements specification document
3. Draw the entity relationship diagram
4. Draw the data flow diagrams
5. Draw use case diagrams
6. Draw activity diagrams for all use cases
7. Draw sequence diagrams for all use cases
8. Draw collaboration diagram
9. Assign objects in sequence diagrams to classes and make class diagram.

- Note:**
1. To draw dataflow diagrams using Microsoft Visio Software, SmartDraw, etc...
  2. To draw UML diagrams using Rational Rose Software, Star UML, etc.