Software Engineering

Course Objectives:

The students will have a broad understanding of the discipline of software engineering and its application to the development of and management of software systems.

Course Outcomes:

- 1. knowledge of basic SW engineering methods and practices, and their appropriate application;
- 2. general understanding of software process models such as the waterfall and evolutionary models.
- 3. understanding of the role of project management including planning, scheduling, risk management, etc.
- 4. understanding of software requirements and the SRS document□
- 5. understanding of different software architectural styles.
- 6. understanding of implementation issues such as modularity and coding standards.
- 7. understanding of approaches to verification and validation including static analysis, and reviews.
- 8. understanding of software testing approachs such as unit testing and integration testing
- 9. understanding of software evolution and related issues such as version management.
- 10. understanding on quality control and how to ensure good quality software.
- 11. understanding of some ethical and professional issues that are important for software engineers
- 12. development of significant teamwork and project based experience

Syllabus:

UNIT I:

Introduction to Software Engineering: Software, Software Crisis, Software Engineering definition, Evolution of Software Engineering Methodologies, Software Engineering Challenges.

Software Processes: Software Process, Process Classification, Phased development life cycle, Software Development Process Models- Process, use, applicability and Advantages/limitations

UNIT II:

Requirements Engineering: Software Requirements, Requirements engineering Process, Requirements elicitation, Requirements Analysis, Structured Analysis, Data Oriented Analysis, Object oriented Analysis, Prototyping Analysis, Requirements Specification, Requirements Validation, requirement Management.

UNIT III:

Software Design: Software Design Process, Characteristics of Good Software Design, Design Principles, Modular Design, Design Methodologies, Structured Design, Structured Design Methodology, Transform Vs Transaction Analysis.

Object-Oriented Design: Object oriented Analysis and Design Principles

UNIT IV:

Implementation: Coding Principles, Coding Process, Code verification, Code documentation **Software Testing:** Testing Fundamentals, Test Planning, Black Box Testing, White Box Testing, Levels of Testing, Usability Testing, Regression testing, Debugging approaches

UNIT V:

Software Project Management: Project Management Essentials, What is Project management, Software Configuration Management.

Project Planning and Estimation: Project Planning activities, Software Metrics and measurements, Project Size Estimation, Effort Estimation Techniques.

UNIT VI:

Software Quality: Software Quality Factors, Verification & Validation, Software Quality Assurance, The Capability Maturity Model

Software Maintenance: Software maintenance, Maintenance Process Models, Maintenance Cost, Reengineering, Reengineering activities, Software Reuse.

P C 0 3

Т

4

TEXT BOOKS:

- 1. Software Engineering, concepts and practices, Ugrasen Suman, Cengage learning
- 2. Software Engineering, 8/e, Sommerville, Pearson.
- 3. Software Engineering, 7/e, Roger S.Pressman, TMH

REFERENCE BOOKS:

- 1. Software Engineering, A Precise approach, Pankaj Jalote, Wiley
- 2. Software Engineering principles and practice, W S Jawadekar, TMH
- 3. Software Engineering concepts, R Fairley, TMH