

Software Engineering

EC803A

Credit: 4

1. Introduction to Software Engineering

- 1.1 Concept of Software and Software Engineering
- 1.2 Difference between Software and Systems Engineering
- 1.3 Introduction to Software Process and Software process model
- 1.4 Concept of Software Products

2 The Process

- 2.1 The Software Process Model
- 2.2 Data Flow Diagrams
- 2.3 Petri Net Models
- 2.4 Object Models
- 2.5 Use Case Diagrams
- 2.6 Sequence Diagrams
- 2.7 Hierarchy Diagrams
- 2.8 Control Flow Graphs
- 2.9 State Diagrams
- 2.10 Lattice Models
- 2.11 Concept of UML

3. Software Project Planning

- 3.1 Project Scheduling and Tracking
- 3.2 Software Configuration Management
- 3.3 Project Planning
- 3.4 WBS–Work Breakdown Structure
- 3.5 PERT–Program Evaluation and Review Technique
- 3.6 Software Cost Estimation
 - 3.6.1 Different models like Cocomo model,

4. Software Project Management

- 4.1 Project Management Concepts & Team Approaches
- 4.2 Critical Practices
- 4.3 Personal Software Process
- 4.4 Earned Value Analysis
- 4.5 Error Tracking
- 4.6 Postmortem Reviews
- 4.7 Software Process and Project Metrics

5. Risk Analysis and Management

- 5.1 Introduction
- 5.2 Risk Identification
- 5.3 Risk Estimation
- 5.4 Risk Exposure
- 5.5 Risk Mitigation
- 5.6 Risk Management Plans

6. Requirement Analysis and Software requirement specifications

- 6.1 Feasibility study
- 6.2 Introduction to requirements analysis
- 6.3 Steps for requirements Analysis
- 6.4 Defining the new System
- 6.5 Software requirement specifications
- 6.6 Documentation

6.7 Case studies

7. Software Design & Development

- 7.1 Introduction to Software design & methodology
- 7.2 Phases of the Design Process
- 7.3 Design Concepts & approaches
- 7.4 Representing algorithm for Software development
- 7.5 Top down programming
- 7.6 Architectural Design
- 7.7 Deciding the operating Systems
- 7.8 Design of Data bases
- 7.9 User Interface Design
- 7.10 Component-Level Design
- 7.11 Design of forms and reports
- 7.12 Design of web applications
- 7.13 Case studies

8. Software Coding

- 8.1 Coding Languages-A review
- 8.2 Coding tools
- 8.3 Selection of Coding Languages
- 8.4 Coding Errors
- 8.5 Software Coding Standards & good Coding practices
- 8.6 Coding verification and validation

9. Software Testing

- 9.1 Introduction to software testing and its importance.
- 9.2 Software Testing Process & methods
- 9.3 Source code testing
- 9.4 Testing Software Usability, Reliability and Performance
- 9.5 Automated testing
- 9.6 Data Flow Testing
- 9.7 Random Testing
- 9.8 Boundary Testing
- 9.9 Software testing tools
- 9.10 Case studies

10. Software Quality Assurance

- 10.1 Software Measurement Theory
- 10.2 Introduction to Software Metrics
- 10.3 Product Metrics
- 10.4 Process Metrics
- 10.5 The QM Approach
- 10.6 Formal Inspections and Technical Reviews
- 10.7 Software Reliability
- 10.8 Statistical Quality Assurance

11. Object Oriented Software Engineering

- 11.1 Object-Oriented Concepts and Principles
- 11.2 Object-Oriented Analysis
- 11.3 Object-Oriented Design
- 11.4 Object-Oriented Testing
- 11.5 Software re-usability

12. Software Security

12.1 Introduction to Software Security

12.2 Security attacks and Security threats

12.3 Ensuring security in business environments.

12.4 Security standards

12.5 Forming security policy and security audit

13. Software Maintenance

13.1 Software maintenance concept in general

13.2 Importance of Software maintenance

13.3 Code Transformation

13.4 Upgrading Software

13.5 Software maintenance model

13.6 Maintenance of Legacy applications

13.7 Case studies