Software Engineering

EG2105CT

Year: II **Total: 5 hours /week**

Part: I **Lecture: 3 hours/week**

Tutorial: hours/week

Practical: hours/week

Lab: 2 hours/week

Course Description:

This course aims to guide the students in both the theoretical and practical aspects of developing computer solutions for real-world problems. One will study the tools and techniques used in analysis and design of software systems, and apply those tools within a recognized software.

Course Objectives:

After completing this course, the students will be able to:

1. Introduce the theory and foundations of software engineering
2. Explain Software Project Management
3. Describe some key aspects of a software engineering process
4. Apply fact-finding and problem-solving skills
5. Determine the requirements for a software system
6. Enlist/Explain key aspects of models and processes for design of a software system
7. Apply current trends in the area of software engineering

Course Contents:

Theory

Unit 1. Introduction **[4 Hrs.]**

1.1. Introduction to software
1.2. Program Vs software
1.3. Software components
1.4. Characteristics of software
1.5. Types of software
1.6. Generic view of software engineering

Unit 2. Software Development Life Cycle Models **[7 Hrs.]**

2.1. Build and fix model
2.2. The waterfall model
2.3. Prototyping model
2.4. Iterative enhancement model
2.5. Spiral model
2.6. Rapid application development model (RAD)
2.7. Selection criteria of a lifecycle model

Unit 3. Software Project Management **[7 Hrs.]**

3.1. Activities in project management
3.2. Software project planning
3.3. Software project management plan
3.4. Software project scheduling and Time Line Charts
3.5. Software project team management and organization
3.6. Software Project estimation

3.6.1. LOC Based Estimation
3.6.2. FP Based Estimation
3.6.3. COCOMO model
3.7. Risk analysis and management
3.8. Risk management process
3.9. Software configuration management

Unit 4. Software Requirement Analysis & Specification [6 Hrs.]
4.1. Requirement engineering
4.2. Requirement elicitation
4.2.1. Interviews
4.2.2. Brainstorming series
4.2.3. Use case approach
4.3. Requirement analysis
4.3.1. Data flow diagram
4.3.2. Data dictionary
4.3.3. Entity-Relationship diagram
4.4. Requirement documentation
4.4.1. Nature of SRS
4.4.2. Characteristics of a good SRS
4.4.3. Organization of SRS

Unit 5. Software Design [6 Hrs.]
5.1. Objectives of design
5.2. Design framework
5.3. Software design models
5.4. Design process
5.5. Architecture design
5.6. Low level design
5.7. Software design strategies
5.8. Function oriented design Vs Object oriented design

Unit 6. Software Metrics [3 Hrs.]
6.1. Software metrics
6.2. Token count
6.3. Data structure metrics
6.4. Information flow metrics
6.5. Metrics analysis

Unit 7. Software Reliability [2 Hrs.]
7.1. Basic Concepts
7.2. Software quality
7.3. Software reliability model

Unit 8. Quality Management and Testing [7 Hrs.]
8.1. Software quality attributes
8.2. Quality factors
8.3. Quality control
8.4. Quality assurance
8.5. Verification and validation

8.6. Testing and debugging
8.7. Testing process
8.8. Unit testing
8.9. Integration testing
8.10. System testing
8.11. Regression testing
8.12. White Box testing and Black Box testing

Unit 9. Software Maintenance **[3 Hrs.]**

9.1. Need for software maintenance
9.2. Types of software maintenance
9.3. Software maintenance process model
9.4. Software maintenance cost

Practical: **[30 Hrs.]**

The practical should contain all features mentioned above.

Final written exam evaluation scheme			
Unit	**Title**	**Hours**	**Marks Distribution***
1	Introduction	4	7
2	Software Development Life Cycle Models	7	12
3	Software Project management	7	12
4	Software Requirement Analysis & Specification	6	11
5	Software Design	6	11
6	Software Metrics	3	5
7	Software Reliability	2	5
8	Quality Management and Testing	7	12
9	Software Maintenance	3	5
	Total	**45**	**80**

* There may be minor deviation in marks distribution.

Reference:

1. Agarwal, K. and Singh, Y., 2007. *Software Engineering*. (3rd ed). New Delhi: New Age International Publisher.
2. Ghezzi, Jayazeri and Mandrioli(2002). *Fundamentals of Software engineering* (2nd ed).
3. Mall, Rajib(2006).*Fundamentals of Software Engineering* (2nd ed). India: Prentice-Hall of India
4. Sommerville, I. (2010). *Software engineering* (10th ed). Harlow, England: Addison-Wesley.