

**Objective:** To understand and apply various project management techniques based on Software Engineering guidelines and Principles.

**Prerequisite:** Object Oriented Programming Fundamentals

**Credits Earned:** 3 Credits

**Course Outcomes:** After completion of this course, student will be able to

- Knowledge of various software engineering principles.
- Understand and implement Various Agile principles of software development
- Apply various models such as data, behavioural, object, interaction and context.
- Understand and apply different metrics and risk assessment strategies.
- Apply various project management techniques (Umbrella Activities) such as planning, scheduling, tracking testing etc.
- Understanding of software quality using quality control techniques.
- Understanding different Testing Methods and using Various Testing Tools.

### Teaching and Examination Scheme

Teaching Scheme (Hours)			Credits	Theory Marks			Tutorial/ Practical Marks		Total Marks
Theory	Tutorial	Practical		ESE (E)	Mid Sem (M)	Internal (I)	Viva (V)	Term work (TW)	
2	0	2	3	50	30	20	25	25	150

**Contents:**

<b>Unit</b>	<b>Topics</b>	<b>Contact Hours</b>
1	<b>Introduction:</b> Software engineering, Dual role of software, Various Myths Associated with Software, Different Software Process Models, The Linear Sequential Model, The Prototyping Model, The RAD Model, Evolutionary Process Models, Component-Based Development.	5
2	<b>Project Management Concepts, Requirement Engineering &amp; Metrics:</b> The Management Spectrum 4P's (People, Project, Product and Process), The W5HH Principle. Basic concept of Requirement (Functional & Non Functional), Metrics in the Process and Project Domains, Software Measurement, Metrics for Software Quality	7
3	<b>Project Planning Scheduling &amp; Tracking:</b> Software Scope, Feasibility Analysis, Empirical Estimation Models, Gantt Charts and Tools used.	4
4	<b>Risk Analysis And Management:</b> Reactive versus Proactive Risk Strategies, Risk Management Process, Risk Identification, Risk Projection, Risk Refinement, RMMM Plans	5
5	<b>Design and Modeling using UML</b> Outcome of Design Process, Effective Modular Design (Functional Independence, Cohesion, and Coupling), Modeling design Technique, Three models - Class Model, State model and Interaction model. <b>Class Modeling</b> Object and class concepts, link and association, Inheritance, Advanced class modeling- aggregation, constraints. <b>State Modeling</b> Event, state, Transition and Interaction Modeling, State diagram <b>Use case Models, Sequence Models, Activity Models</b>	9
6	<b>Software Coding &amp; Testing:</b> Coding standards & Coding Guidelines, Code Review, Software Documentation: Internal and External Documentation, Software Testing Fundamentals, Software	6

**Computer Engineering**

	Testing Techniques, White Box Testing Techniques, Black Box Testing Techniques.	
7	<b>Software Quality Management:</b> Quality Concepts and Software Quality Assurance, Quality principles and Attributes, Software Reviews, Formal Technical Reviews, The SQA Plan, Software Reliability, The Quality Standards: ISO 9000, Six Sigma.	<b>6</b>
	<b>Total Hours</b>	<b>42</b>

**Reference Books:**

1. Roger S.Pressman, Software engineering- A practitioner's Approach, McGraw-Hill International Editions
2. Ian Sommerville, Software engineering, Pearson education Asia
3. Pankaj Jalote, Software Engineering – A Precise Approach Wiley
4. Software Engineering Fundamentals by Ali Behhforoz & Frederick Hudson OXFORD
5. Rajib Mall, Fundamentals of software Engineering, Prentice Hall of India.
6. Engineering Software as a Service and Agile Software Approach, Armando Fox and David Patterson
7. John M Nicolas, Project Management for Business, Engineering and Technology, Elsevier.
8. Nageswara Rao Pusuluri, Software Testing Concepts and Tools, DreamTech
9. Sanjay Mohapatra, Software Project Management, Cengage Learning

**Suggested Theory distribution:**

The suggested theory distribution as per Bloom's taxonomy is as per follows. This distribution serves as guidelines for teachers and students to achieve effective teaching-learning process.

Distribution of Theory for course delivery and evaluation					
Remember	Understand	Apply	Analyze	Evaluate	Create
35%	35%	30%	0%	0%	0%

**Suggested List of Experiments:**

1. Prepare a use case diagram for hospital management system.
2. Prepare a sequence diagram for school management system.
3. Prepare a use case diagram for an online airline reservation system.
4. Draw a sequence diagram for issuing a book and renewing a book in online library management system.
5. Prepare an activity diagram for computing a restaurant bill. There should be a charge for each delivered item. The total amount should be subject to tax and service charge of 18%. Any coupons or gift certificates submitted by the customer should be subtracted.
6. Prepare an activity diagram that elaborates the details of logging into an email system.
7. Draw the use-case diagram for Hotel Information System. There are two types of customers: Tour-group customers and Individual customers. Both can book, cancel, check-in and check-out of a room by Phone or via the Internet. There are booking process, clerk and reception staff who manages it. A customer can pay his bill by credit card or pay utility bill.
8. A soft drink vending machine accepts coins for a variety of products. When the amount of money deposited into the machine is equal to or greater than the price of any of its available products, the respective product selection buttons will be enabled for the user to make the selection. After the user has made a valid selection, the machine will dispense the soft drink, together with the change (if applicable). Draw the Activity Diagram for this vending machine.
9. Consider you are interacting with an online travel agent and encounter the following use cases. Prepare a use case diagram, using the generalization and include relationships.
  - **Purchase a flight:** - Reserve a flight and provide payment and address information.
  - **Provide payment information:** - Provide a credit card to pay for the incurred changes
  - **Provide address:** - Provide mailing and residence address.
  - **Purchase car rental:** - Reserve a rental car and provide payment and address information.
  - **Purchase a hotel stay:** - Reserve a hotel room and provide payment and address information.
  - **Make a Purchas:** - Make a travel purchase and provide payment and address information.
10. Prepare SRS document considering any specific Social Project according to the below guidelines.

**Detail of Report/Chapterwise:****1.0 Introduction**



**Computer Engineering**

**1.1 Project Summary** – Key to a good summary is the FIRST sentence, which MUST contain the most essential information that you wish to convey.

**1.2 Purpose:** Goals & Objectives

**1.3 Scope** (Scope – what it can do and can't do)

**1.4 Technology and Literature Review** of Past Work/System

**2.0 Project Management**

**2.1 Project Planning and scheduling**

**2.1.1 Project Development Approach** (Process Paradigm) and Justification

**2.1.2 Project Plan** including Milestones, Deliverables, Roles, Responsibilities and Dependencies

**2.1.3 Schedule Representation**

**2.2 Risk Management**

**2.2.1 Risk Identification** (it is concerned with discovering possible risk to the project)

**2.2.2 Risk Analysis** (each identified risk is considered in turn and a judgment made about the probability and the seriousness of the risk)

**2.2.3 Risk Planning** (Identify strategies to manage the risk)

**Note: Discuss the risks associated with your project or system only.**

**2.3 Estimation**

**2.3.1 Effort Estimation**

**2.3.2 Cost Analysis** (Total cost of the project including resources and labors. Labor cost should be broken down into the areas of design, analysis, prototype construction, software development, hardware, software integration, testing, design modifications and documentation. A cost analysis is NOT a tabulation of your expenditure)

**3.0 System Requirements Study**

**3.1 User Characteristics** (Type of users who is dealing with the system)

**3.2 Hardware and Software Requirements** (minimum requirements to run your system)

**3.3 Constraints**

(It includes Regulatory Policies, Hardware Limitations, Interfaces to Other Applications, Parallel Operations, Higher Order Language Requirements, Reliability Requirements, Criticality of the Application, Safety and Security Consideration, Assumptions and Dependencies or any other constraints related to your system)

**4.0 System Analysis**

**4.1 Study of Current System**

**4.2 Problem and Weaknesses of Current System**



**Computer Engineering**

**4.3 Requirements of New System**

(Mention all functional and non- functional including user and system requirements)

**4.4 Feasibility Study**

(In this section, does feasibility analysis by finding answers of the questions like Does the system contribute to the overall objectives of the organization?

Can the system be implemented using the current technology and within the given cost and schedule constraints? Can the system be integrated with other systems which are already in place? etc)

**4.5 Requirements Validation** (is concerned with showing that the requirements actually define the system which the customer wants)

**4.6 Functions of System**

4.6.1 Use Cases, event trace or scenario

**4.7 Data Modeling**

4.7.1 Class Diagram/ E-R diagrams

4.7.2 System Activity or Object interaction Diagram

4.7.3 Data Dictionary

**4.8 Functional and Behavioral Modeling**

4.8.1 Context Diagram

4.8.2 Data Flow Diagram (0 and 1 level)

4.8.3 Process Specification and Decision Table

4.8.4 Control flow diagram

**4.9 Main Modules of New System**

**4.10 Selection of Hardware and Software and Justification**

**Note: Prepare System Requirement Specification (SRS) after analysis phase. Choose appropriate guideline for your system.**

**5.0 System Design**

**5.1 Database Design/Data Structure Design**

5.1.1 Mapping objects/classes to tables (if non OO languages)

5.1.2 Tables and Relationship

5.1.3 Logical Description of Data

**5.2 System Procedural Design**

5.2.1 Designing Pseudo code or algorithm for Method or operations

5.2.2 Flow chart or activity design

**5.3 Input / Output and Interface Design**



**Computer Engineering**

5.3.1 Samples of Forms, Reports and Interface

5.3.2 Access Control and Security

5.3.3 State-Transition Diagram

**5.4 System Architecture Design**

(Transformation of DFD into structural chart/Hierarchical Charts which shows control hierarchy of modules or sub-systems)

**6.0 Implementation Planning and details**

6.1 Implementation Environment (Single vs Multiuser, GUI vs Non GUI)

6.2 Program/Modules Specification

6.3 Security Features

6.4 Coding Standards

6.5 Sample Coding

**7.0 Testing (choose appropriate testing strategy or techniques suitable to your system)**

7.1 Testing Plan

7.2 Testing Strategy

7.3 Testing Methods

7.4 Test Cases (Purpose, Required output, Expected Result)

**8.0 Screen shots and User manual**

**9.0 Limitation and Future Enhancement**

**10.0 Conclusion and Discussion**

**Instructional Method:**

- a. The course delivery method will depend upon the requirement of content and need of students. The teacher in may be using following teaching approaches: black board, or use of any of tools such as demonstration, role play, Quiz, brainstorming, MOOCs etc.
- b. The internal evaluation will be done on the basis of continuous evaluation of students in the laboratory and class-room.
- c. Practical examination/Viva will be conducted at the end of semester for evaluation of performance of students in laboratory.
- d. Students will use supplementary resources such as online videos, NPTEL videos, e-courses, Virtual Laboratory

**Supplementary Resources:**

**Computer Engineering**

1. <http://nptel.ac.in/courses/106101061/>
2. <https://www.joelonsoftware.com/>
3. <http://www.codesimplicity.com/>
4. <http://www.sparxsystems.com/products/ea/index.html>
5. URL:<http://www.smartdraw.com>
6. URL:<http://viu.eng.rpi.edu>
7. [www.en.wikipedia.org/wiki/Software\\_engineering](http://www.en.wikipedia.org/wiki/Software_engineering)
8. [www.win.tue.nl](http://www.win.tue.nl)
9. [www.rspa.com/spi](http://www.rspa.com/spi)
10. [www.onesmartclick.com/engsineering/software-engineering.html](http://www.onesmartclick.com/engsineering/software-engineering.html)
11. [www.sei.cmu.edu](http://www.sei.cmu.edu)
12. <https://www.edx.org/school/uc-berkeleyx>