

Course Description

The course deals with learning the basic concepts of Software Development through Software Development Life Cycle (SDLC), using major methodologies using Unified Modelling Language (UML). It presents the use of design patterns with software engineering concepts. Students will gain experience in various processes used in the Software industry for the development of a software product. They also learn about testing and maintenance of software products using design patterns to solve real-world problems.

Course Objectives

- To provide an idea of using various process models in the software industry according to given circumstances.
- To provide the idea of decomposing the given problem into Analysis, Designing,

Implementation, Testing and Maintenance phases.

- To gain knowledge of how to analyze the solution to the problems using design patterns.

Course Outcomes

After completing this course, students will be able to:

COs	Description
CO1	Gain knowledge of software engineering methods, practices and their appropriate applications which describe software engineering layered technology and satisfies its process framework in understanding of waterfall and evolutionary models.
CO2	Understand the concept of software requirements analysis and software design engineering methodologies.
CO3	Understand the concept of Design patterns and its importance in gaining behavioral knowledge of the problem and its solutions using Creational, Structural design patterns.
CO4	Understand and apply common design patterns to incremental/iterative development. To identify appropriate behavioral patterns for the design to propose solutions to the given problem.
CO5	Understand the need for programming by using basic design principles in solving real-life problems or case studies.

CO-PO Mapping

PO/PS O	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO												
CO1	-	2	3	1	1	1	3	3	3		3	1
CO2	-	2	3	2	2	1	3	3	3		3	2
CO3	3	2	3	2	3		3				1	1
CO4	3	3	3	3	3		3				1	1
CO5	3	2	2	2	3		3				1	1

Prerequisites

- Java

Unit I

Software Engineering – Introduction - Software Classification - Layered Technology - Software Process –Practice – SDLC - Generic Process Model, Process Assessment.

Unit II

Perspective Models - Agile Process Models – Scrum and Extreme Programming (XP) - Requirements Analysis - Unified Modelling Language – Design Engineering – Test Engineering.

Unit III

Introduction: What Is a Design Pattern? Describing Design Patterns, The Catalogue of Design Patterns, Organizing the Catalogue, How Design Patterns Solve Design Problems, How to Select a Design Pattern, How to Use a Design Pattern.

Unit IV

Creational Patterns: Abstract Factory, Factory Method, Singleton Structural Patterns: Adapter, Bridge, Composite, Decorator, Façade, and Proxy. Behavioral Patterns: Command, Iterator, Observer and Template Method. Case study on design patterns.

Textbooks / References:

1. Roger S. Pressman, “Software Engineering-A Practitioner’s Approach”, Seventh Edition, Tata McGraw-Hill, 2010.
2. Richard Fairley, “Software Engineering concepts”, Tata McGraw-Hill Publishing Company Pvt. Ltd., Ninth Edition
3. Patterns in JAVA Vol-I By Mark Grand, Wiley Dream Tech.
4. Head First Design Patterns by Eric Freeman-O’Reilly-spd