



# GUJARAT TECHNOLOGICAL UNIVERSITY

Program Name: Master of Engineering

Level: PG

Branch: Information Technology

Subject Code : ME02023051

Subject Name : Software Design Patterns

w. e. f. Academic Year:	A.Y. 2024-25
Semester:	2
Category of the Course:	Professional Elective Course

<b>Prerequisite:</b>	Concept of Software Engineering and Object Oriented Analysis and Design
<b>Rationale:</b>	A Software Design Pattern course is essential to build a strong foundation in software architecture, enhance practical design skills, and prepare students for advanced research and industry challenges.

### Course Outcome:

After Completion of the Course, Student will able to:

No	Course Outcomes	RBT Level
1	Understand the role and significance of software design patterns in software development	U
2	Select and implement appropriate design patterns for specific software problems	A
3	Analyze and refactor software systems to improve quality using design patterns	E
4	Design and develop complex software systems by integrating multiple design patterns	C
5	Adapt and apply design patterns in emerging areas like distributed systems, cloud computing, and IoT	A

*\*Revised Bloom's Taxonomy (RBT)*

### Teaching and Examination Scheme:

Teaching Scheme (in Hours)			Total Credits L+T+ (PR/2)	Assessment Pattern and Marks				Total Marks
L	T	PR	C	Theory		Tutorial / Practical		
				ESE (E)	PA / CA (M)	PA/CA (I)	ESE (V)	
3	0	2	4	70	30	20	30	150



# GUJARAT TECHNOLOGICAL UNIVERSITY

Program Name: Master of Engineering

Level: PG

Branch: Information Technology

Subject Code : ME02023051

Subject Name : Software Design Patterns

## Course Content:

Unit No.	Content	No. of Hours	% of Weightage
1.	<b>Introduction to Software Design Pattern</b> Definition and significance of design patterns in software development, Design Patterns in Smalltalk MVC, The Catalog of Design Patterns, Organizing the Catalog, How Design Patterns Solve Design Problems, How to Select a Design Pattern, How to Use a Design Pattern.	04	10
2.	<b>Creational Design Patterns</b> Factory Method, Abstract Factory, Builder, Prototype, Singleton	08	20
3	<b>Structural Design Patterns</b> Adapter, Bridge, Composite, Decorator, Façade, Flyweight, Proxy	09	20
4	<b>Behavioral Design Patterns</b> Chain of Responsibility, Command, Interpreter, Iterator, Mediator, Memento, Observer, State, Strategy, Template Method, Visitor	10	25
5.	<b>Patterns in Modern Software Development</b> Patterns in distributed systems: Microservices, Event-driven architecture, Patterns in cloud computing: Circuit Breaker, Service Registry, Load Balancer, Anti-patterns and their consequences, Tools for modelling design patterns: UML, design tools	07	15
6.	<b>Interactive systems and the MVC architecture</b> Introduction, The MVC architectural pattern, Analyzing a simple drawing program, Designing the system, Designing of the subsystems	07	10
<b>TOTAL</b>		<b>45</b>	<b>100</b>

## Suggested Specification Table with Marks (Theory):

Distribution of Theory Marks (in %)					
R Level	U Level	A Level	N Level	E Level	C Level
10	30	30	20	10	---

Where R: Remember; U: Understanding; A: Application, N: Analyze and E: Evaluate C: Create (as per Revised Bloom's Taxonomy)



# GUJARAT TECHNOLOGICAL UNIVERSITY

**Program Name: Master of Engineering**

**Level: PG**

**Branch: Information Technology**

**Subject Code : ME02023051**

**Subject Name : Software Design Patterns**

## References/Suggested Learning Resources:

### (a) Books:

1. Design Patterns: Elements of Reusable Object-Oriented Software by Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides (GoF).
2. Head First Design Patterns by Eric Freeman and Elisabeth Robson
3. Refactoring: Improving the Design of Existing Code by Martin Fowler.
4. Patterns of Enterprise Application Architecture by Martin Fowler.
5. Design Patterns Explained: A New Perspective on Object-Oriented Design by Alan Shalloway and James Trott.
6. Brahma Dathan, Sarnath Rammath, Object-oriented analysis, design and implementation, Universities Press,2013

### (b) Open source software and website:

Web resource

- [https://www.tutorialspoint.com/design\\_pattern/index.htm](https://www.tutorialspoint.com/design_pattern/index.htm)
- <https://www.edx.org/course/design-patterns>
- <https://github.com/iluwatar/java-design-patterns>
- <https://www.coursera.org/learn/design-patterns>

## Suggested Course Practical List: (List can be change according to Latest Development)

1. Implement a configuration manager using the Singleton pattern.
2. Develop a product generation module using Factory and Builder patterns.
3. Implement a file hierarchy using Composite and Decorator patterns.
4. Develop a caching system using Proxy and Flyweight patterns.
5. Develop a notification system using Observer and Strategy patterns.
6. Implement a transaction management system using State and Command patterns.
7. Analyze and refactor an existing system to resolve anti-pattern issues.
8. Develop a service-oriented system using distributed design patterns.

**List of Laboratory/Learning Resources Required:** Programming Languages – Python / JAVA etc.

\* \* \* \* \*