**Day 26 - 22nd Aug 2025 docs**

**Task – 1:**

### **What is Design Workflow?**

Design Workflow is the step-by-step process or plan to create, or to organize, and implement a design — whether it’s for a software or for a product, or for a system. It defines how tasks flow from start to finish so that the final design is efficient, and meets requirements.

Think of it like a recipe:

* 1. Gather ingredients (requirements)
  2. Follow steps in order (planning, designing, testing)
  3. Produce a final dish (completed design/product)

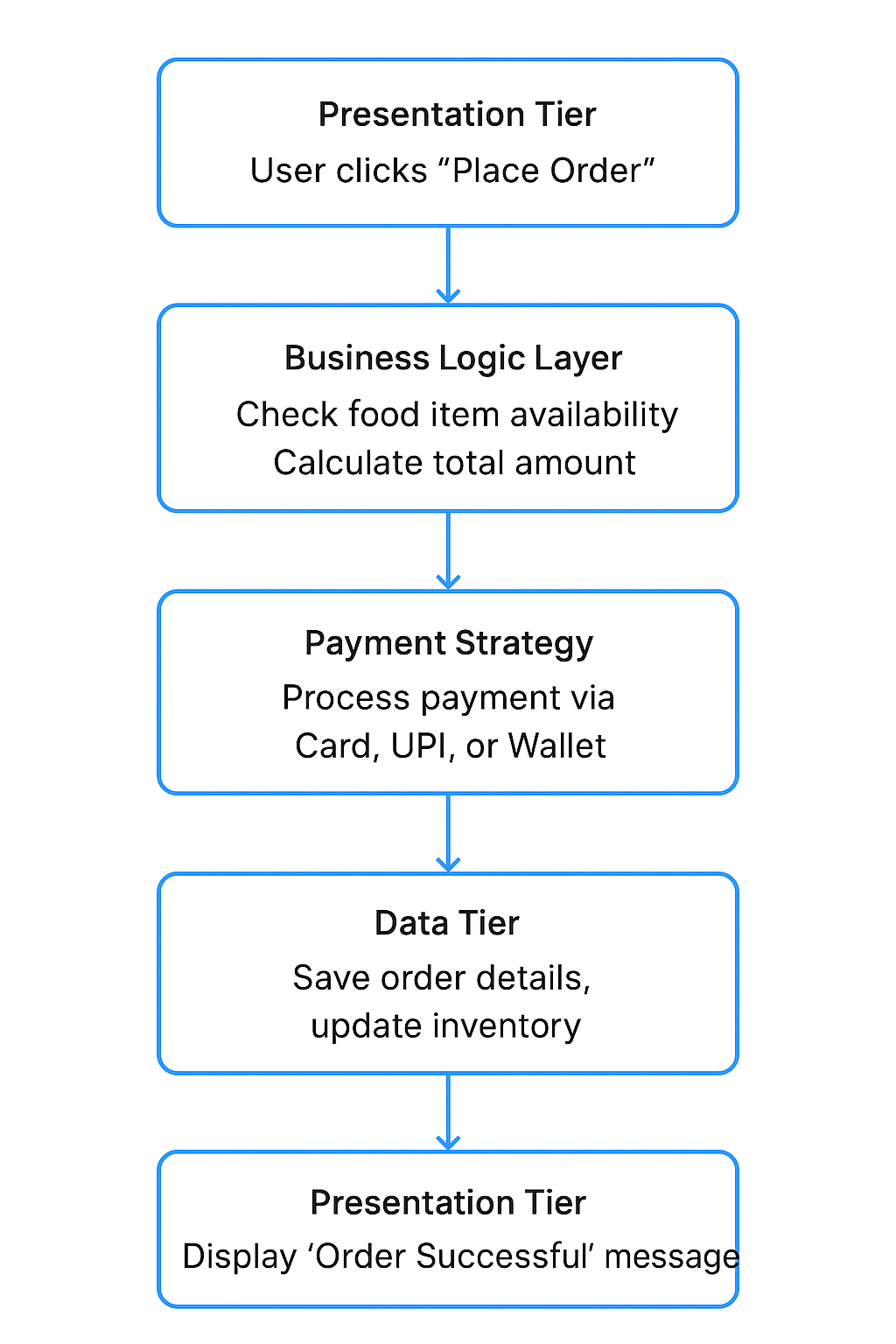
Design Workflow is equal to a structured sequence of steps followed to create a design efficiently, from understanding requirements to final implementation and feedback.

### Example: Typical Steps in a Design Workflow:

1. Requirement Analysis  
   * Understand what the system or product should do.
2. Planning & Architecture  
   * Decide how components will interact (e.g., 3-tier architecture).
3. Design  
   * Create diagrams, UI layouts, data models, workflows.
4. Implementation / Development  
   * Convert designs into working code or prototype.
5. Testing & Validation  
   * Check if the design works as expected.
6. Deployment & Feedback  
   * Release the product and gather feedback to improve.

Project Idea:

An application where users can browse restaurants, place orders, and pay online.



**Flow Diagram of Design Work Flow**

### 

**Task - 2:**

## **What are Persistent Objects?**

Persistent objects are objects in a program whose state (data) can be saved permanently so that it can be used even after the program stops running.

* Normally, when a program stops, all objects in memory are lost.
* A persistent object is saved in a database, file, or storage, so it can be retrieved and reused later.

**Examples**

1. **User Object in a Web App**
   * User signs up → details stored in database
   * Later, user logs in → persistent object is retrieved from database
2. **Order Object in E-Commerce App**
   * Order is created and saved in database
   * Even if the server restarts, order details are still available
3. **File Storage**
   * An object representing a document can be saved to disk and reloaded later

In short:

Persistent Objects is equal to objects whose data is saved permanently in storage, allowing the program to retrieve and reuse them even after it stops.

**Task 4:**

Which of the following components is not typically part of the Command pattern?

a) Invoker

b) Receiver

c) Abstract Factory

d) Command (interface/abstract class)

**Task 5:**

What role does the Invoker play in the Command pattern?

a) It knows how to perform the operations associated with a request.

b) It encapsulates the request as an object.

c) It asks the command to carry out the request.

d) It defines the interface for executing an operation.

**Task 6:**

A key benefit of using the Command pattern is its ability to support:

a) Lazy initialization

b) Undo/Redo functionality

c) Singleton instance creation

d) Compile-time polymorphism

**Task 7:**

In the Strategy pattern, what role does the "Context" play?

A. It defines the interface for the algorithms.

B. It implements a specific algorithm.

C. It maintains a reference to a Strategy object and delegates the task to it.

D. It creates the Concrete Strategy objects.

**Task 8**

1. In which of the following mechanisms, types of all variables and expressions are fixed at compilation time.

a) Strong Typing

b) Weak Typing

c) Static Binding/ early binding

d) Dynamic Binding/ late binding

**Task 9:**

In which pattern does a class represent the functionality of another class, providing a simplified interface to a complex subsystem?

a) Decorator Pattern

b) Facade Pattern

c) Proxy Pattern

d) Composite Pattern

**Task 10:**

Which of the following statements about Persistence is correct?

a) It is the enforcement of the class of an object, such that objects of different types may not be interchanged, or at the most they may be interchanged only in very restricted ways.

b) It is the property of an object through which its existence transcends time and/or space.

c) It is the property that distinguishes an active object from one that is not active.

d) All of the mentioned

**Task 11:**

What is that concept in type theory in which a single name may denote objects of many different classes that are related by some common super class referred to \_\_\_\_\_\_

a) Monomorphism

b) Type Checking

c) Polymorphism

d) Generalization

**Task 12:**

Which of the following patterns is used to create a single instance of a class and provide a global point of access to it?

a) Factory Pattern

b) Singleton Pattern

c) Builder Pattern

d) Prototype Pattern

**Task 13:**

The Adapter pattern is a type of \_\_\_\_\_\_ pattern.

a) Creational

b) Structural

c) Behavioral

d) Concurrency

**Task 14:**

Which design pattern defines a one-to-many dependency between objects so that when one object changes state, all its dependents are notified and updated automatically?

a) Strategy Pattern

b) Command Pattern

c) Observer Pattern

d) Mediator Pattern

**Task 15:**

The Model-View-Controller (MVC) is an example of a \_\_\_\_\_\_ pattern.

a) Creational

b) Structural

c) Behavioral

d) Architectural