Objective

- Apply software design principles to break down a system into modules.
- Create UML diagrams to visually represent software design.
- Develop a basic design document incorporating architecture and patterns.

Setup

- Tools: Use drawing tools like draw.io, Lucidchart, or pen & paper for UML diagrams.
- Word processor or Google Docs for documentation.

Tasks

Task 1: Select a Project

Choose a simple system to design, for example:

- Online Food Ordering App
- Library Management System
- Event Ticket Booking System

Task 2: Identify Modules and Components

- Break down the system into at least 4 major modules (e.g., User Management, Order Processing).
- For each module, write a brief description of its responsibility.

Task 3: Create UML Diagrams

- **Class Diagram:** Identify key classes and their relationships. Include attributes and methods.
- **Sequence Diagram:** Illustrate the flow of operations for a key feature (e.g., placing an order).
- **Use Case Diagram:** Show actors and main interactions with the system.

Task 4: Apply Design Principles

- Explain how your design applies principles like modularity, encapsulation, and low coupling.
- Identify if you used any design patterns and justify their use.

Task 5: Document the Design

- Create a simple design document including:
    - Project Overview

  - Module Descriptions
  - UML Diagrams
  - Design Principles and Patterns Applied

Deliverables

- UML diagrams (images or PDFs).
- Design document (.docx, .pdf, or Google Docs).

Evaluation Criteria

- Clarity and completeness of module decomposition.
- Accuracy and clarity of UML diagrams.
- Application of design principles and patterns.
- Organization and detail in the design document.

Bonus Task (Optional)

- Create a UI wireframe or mockup for one main screen.
- Include UX considerations in your design document.