**Requirements Modeling**

**Scenario:**  
You are tasked with developing an online food delivery application. The app should allow users to browse restaurants, place orders, and track delivery. Additionally, restaurant owners should be able to manage their menus and track order status.

**Questions:**

1. Identify **functional requirements** for the application.
2. Identify **non-functional requirements** for the application.

****

**Scenario 2:**  
A fitness app is being developed to allow users to track workouts, monitor calories, and connect with wearable fitness devices. The app will also have social features for sharing achievements with friends.

* + Identify at least five **functional requirements** for the app.
  + Identify at least four **non-functional requirements** that are critical for this app.

**Scenario 3:**  
A library management system needs to be developed for a public library. It should support book borrowing, returning, searching for books, and generating overdue notifications.

* + Create a categorized list of functional and non-functional requirements for the library management system.

**Advanced Scenario: Requirements Modeling and Use Case Documentation**

**Scenario:**  
You are hired to design and document a **smart home automation system**. The system should allow homeowners to control and monitor devices such as lights, thermostats, and security cameras through a mobile app. Key features include scheduling automation, remote access, and notifications for unusual activities (e.g., motion detection). The system must integrate with popular voice assistants (e.g., Alexa, Google Assistant).

**Tasks:**

1. Identify **functional requirements** for the smart home automation system.
2. Identify **non-functional requirements** for the system.
3. List and document at least **two use cases** for the system.

**Use Case Documentation**

1. **Scenario 1:**  
   For the fitness app described above, write a detailed use case for the "Track Workout" functionality. Ensure you include:
   * Use Case Name
   * Actors
   * Preconditions
   * Postconditions
   * Main Flow
   * Alternate Flows
2. **Scenario 2:**  
   For the library management system:
   * Write a use case for "Borrow a Book."

**Architectural Design**

1. **Scenario 1:**  
   For the fitness app, design a high-level architecture. Identify the key components and their roles in the system.
   * What type of database(s) would you choose for this app? Justify your choice.
   * How would you ensure scalability for real-time synchronization with wearable devices?
2. **Scenario 2:**  
   For the library management system:
   * Propose an architecture to support thousands of simultaneous users searching and borrowing books.
   * How would you ensure data consistency when multiple users try to borrow the same book simultaneously?

**SOLID Principles**

**Short Answer Questions**

1. Explain the **Single Responsibility Principle** and give a real-world example of its application in software.
2. How does the **Open/Closed Principle** support extensibility in software design?
3. What is the **Liskov Substitution Principle**, and how can its violation lead to bugs in object-oriented systems?
4. Why is the **Interface Segregation Principle** important for API design?
5. Provide an example of how the **Dependency Inversion Principle** can decouple high-level and low-level modules.

**Compare and Contrast Questions**

1. Compare and contrast **Single Responsibility Principle** and **Interface Segregation Principle.**
2. Explain how the **Open/Closed Principle** and **Dependency Inversion Principle** complement each other in designing flexible software systems.
3. Contrast the role of **Liskov Substitution Principle** and **Open/Closed Principle** in maintaining software correctness.