**1. Use Case Diagram:**

A use case diagram illustrates how different types of users (actors) interact with the Smart study system. In the case of the Smart Study Hub, the key actors include:

* **Student:** Access study materials, attempt quizzes, view performance.
* **Teacher:** Upload content, create quizzes, view student progress.
* **Administrator:** Manage users, update system, monitor performance.

**Key use cases for the system include:**

* Access Study Material: Students access subject-specific study materials (like notes and videos).
* Attempt Quizzes: Students can take subject-wise quizzes.
* Track Performance: Students and teachers can track learning progress and quiz results.
* Upload Materials: Teachers upload notes, videos, and quizzes.
* Manage Users: Administrators manage user roles, registration, and system updates.

**Advantages:**

* Clear visual of system functionality.
* Identifies roles and tasks efficiently.
* Helps align stakeholders on system scope.

**Disadvantages:**

* High-level; lacks detailed workflows.
* Complex use cases may not be fully represented.

**2. Sequence Diagram:**

A sequence diagram shows the interaction between objects over time, step by step, focusing on how a process or task is performed.

**Steps in the Sequence Diagram:**

1. **Student requests study material** from the system.
2. **System verifies access** and retrieves the material.
3. **Student accesses the material** and **attempts a quiz**.
4. The **system evaluates the quiz** and stores the result.
5. The **student views their performance**.

**Advantages:**

* Provides clear process flow for specific actions.
* Shows system responses, roles, and data flow.

**Disadvantages:**

* Focuses on a specific scenario (may need several diagrams).
* Can become complex if too many interactions are represented.

**Sequence Diagram (Accessing Study Material and Attempting Quiz)**

**Objects:**

* Student
* Study Hub System
* Quiz Module

**Class Descriptions and Relationships:**

The **Smart Study Hub** is an e-learning platform involves multiple classes such as students, lecturers, admins, subjects, quizzes, individual projects, and feedback. Each class has its own attributes and methods that define its role in the system

**Student**:

* + **Attributes**:
    - student ID: Unique identifier for the student.
    - name: Name of the student.
    - email: Contact information.
  + **Methods**:
    - register (): Registers a student into the system.
    - login (): Authenticates a student.
    - view Subject (): Allows the student to view available subjects.
    - attempt Quiz (): Allows the student to take a quiz.
    - submit Project (): Allows the student to submit a project.
    - view Feedback (): Allows the student to check feedback on assessments.

1. **Lecturer**:
   * **Attributes**:
     + lecturer ID: Unique identifier for the lecturer.
     + name: Name of the lecturer.
     + email: Contact information.
   * **Methods**:
     + login (): Authenticates the lecturer.
     + create Subject (): Allows the lecturer to create subjects.
     + create Quiz (): Allows the lecturer to create a quiz.
     + grade Project (): Allows the lecturer to grade submitted projects.
     + provide Feedback (): Provides feedback to students.
2. **Admin**:
   * **Attributes**:
     + adminID: Unique identifier for the admin.
     + name: Name of the admin.
   * **Methods**:
     + manage Users (): Manages students and lecturers.
     + manage Subjects (): Manages the subjects and platform settings.
3. **Subject**:
   * **Attributes**:
     + subject ID: Unique identifier for the subject.
     + title: Name of the subject.
     + description: Overview of the subject.
   * **Methods**:
     + add Quiz (): Adds quizzes to the subject.
     + add Project (): Assigns individual projects to students.
4. **Quiz**:
   * **Attributes**:
     + quiz ID: Unique identifier for the quiz.
     + questions: A list of questions in the quiz.
     + maxScore: Maximum score for the quiz.
   * **Methods**:
     + create Quiz (): Lecturer creates a quiz.
     + attempt Quiz (): Students attempt the quiz.
5. **Individual Project**:
   * **Attributes**:
     + Project Number: Unique identifier for the project.
     + description: Detailed explanation of the project.
     + submission Date: The date by which the project is due.
   * **Methods**:
     + submit Project (): Allows the student to submit the project.
     + grade Project (): Allows the lecturer to grade the project.
6. **Feedback**:
   * **Attributes**:
     + comments: Lecturer’s comments on quizzes/tests/projects.
     + grade: The score or result given to the student.
   * **Methods**:
     + provide Feedback (): Lecturer provides feedback.
     + view Feedback (): Student views the feedback.