**Part 1 – UML Class Diagram (Software Structure Blueprint)**

For this assignment, I am choosing a small system, such as the student registration system

I’ll start by defining the following classes:

1. **Student**
2. **Course**
3. **Registration**

**Student Registration System Explanation**

This system models a student course registration process where students can enroll in courses, view their enrolled courses, and drop courses if needed. The system consists of three main classes:

1. **Student –** Represents students with personal details and methods to manage course registration.
2. **Course** – Represents an academic course with attributes like capacity and enrollment count.
3. **Registration –** Acts as a bridge between Student and Course, tracking enrollment status.

**UML Class Diagram**

**A diagram of a computer

AI-generated content may be incorrect.**

**Explanation:**

* **Associations:**
* **Student** and **Registration** have a many-to-many relationship via **Course** (aggregation denoted by **<>**).
* **Aggregation:**
* **Registration** aggregates **Course**
* **<>** Diamond indicates aggregation.
* **Attributes:**
* Public methods: **+**, Private attributes**: -**
* **Student**: Private student\_id, name, email (integers and strings).
* **Registration**: Private reg\_id, reg\_date, submit (integers, strings, and booleans).
* **Course**: Private course\_id, title, credits, (integers and strings).
* **Methods:**
* **register(course): void** – Allows a student to initiate and submit a registration for a course.
* **submit(): bool** – Finalizes the registration and returns confirmation of success.
* **get\_info(): str** – Returns a formatted string containing course details like title, ID, and credits.

**Part 2 – Sequence Diagram**

**Scenario: A student registers for a course**

The sequence diagram illustrates how a Student object calls the **register() method**, which creates a **Registration object** that interacts with a **Course object** using **get\_info()** and finalizes the process by calling **submit(),** showing the flow of messages and the order of interactions during course registration**.**

**A diagram of a computer program

AI-generated content may be incorrect.**

**Explanation:**

1. The Student initiates registration.
2. Registration fetches course info.
3. Registration submits the registration.
4. A confirmation message is returned.