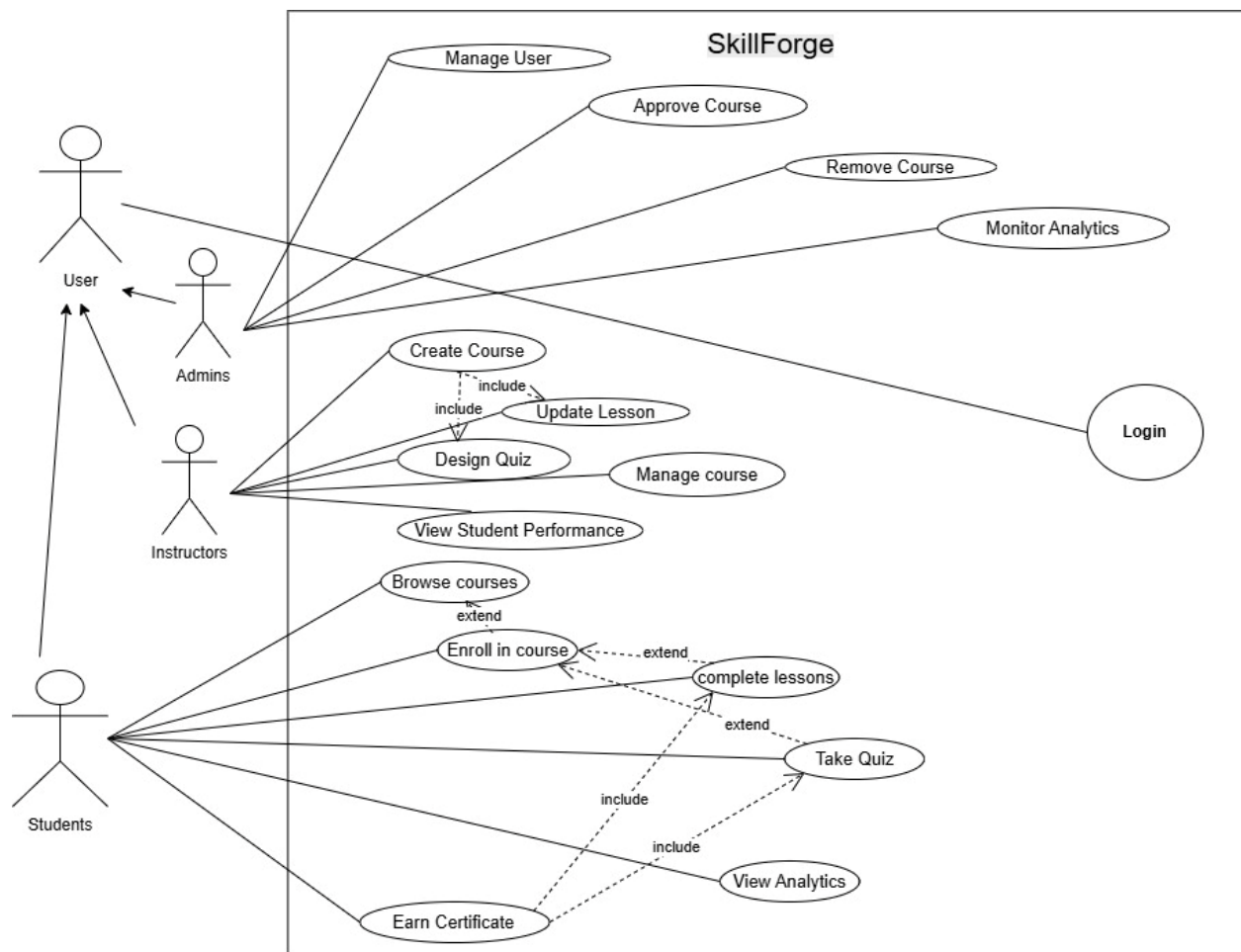


Programming 2
Lab 6 Report
SkillForge UML Diagram

Yasmin Haytham Mahmoud	9324
Farah Ali El-Said	9281
Yahya Ali El-Sayed	9351
Habiba Salah Alansary	9269

1. Use Case Diagram

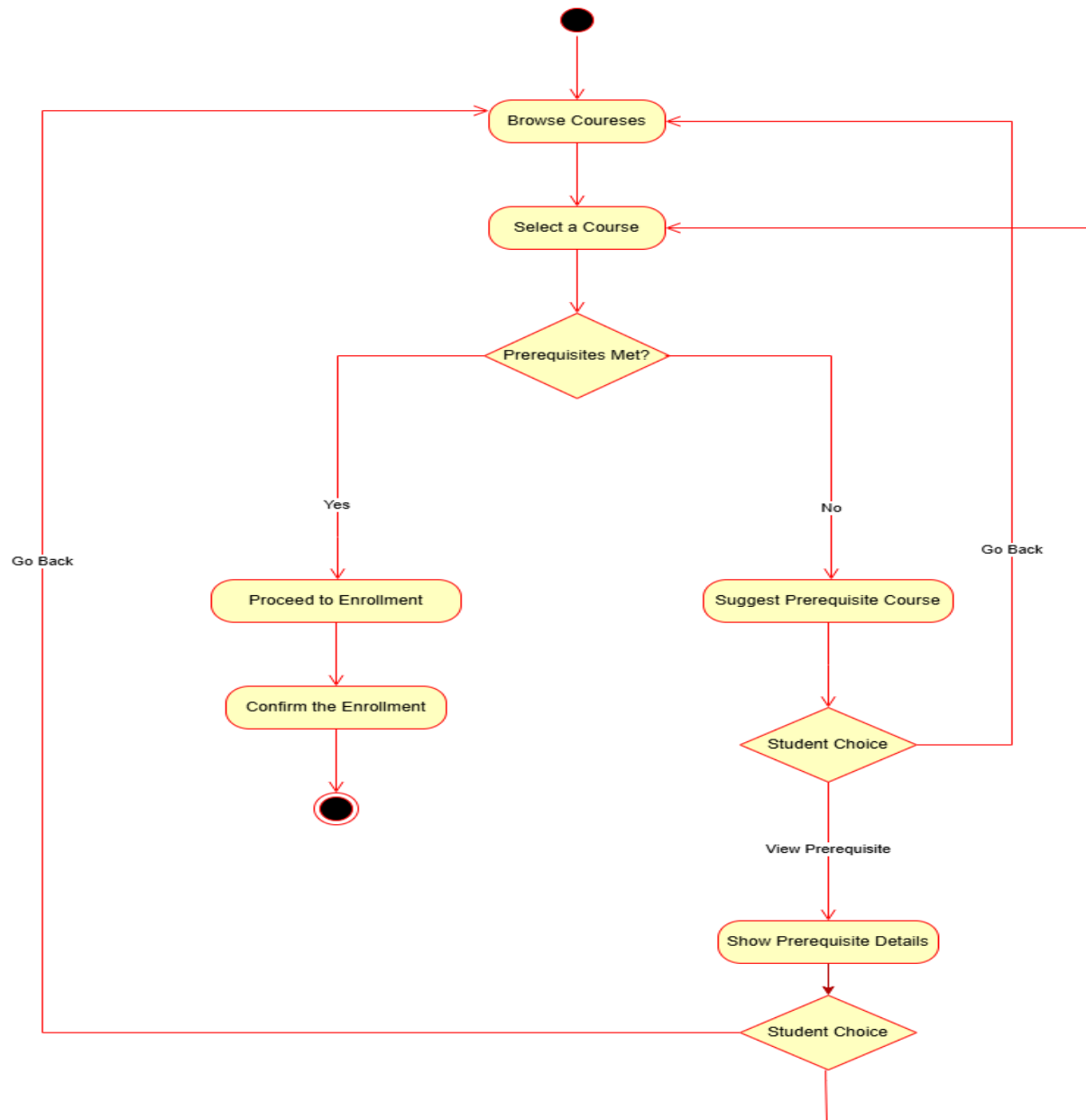


The Use Case Diagram represents the main actors (Student, Instructor, and Admin) and their interactions with the SkillForge system.

It shows the essential functions each user can perform, such as enrolling in courses, creating lessons, managing users, and earning certificates.

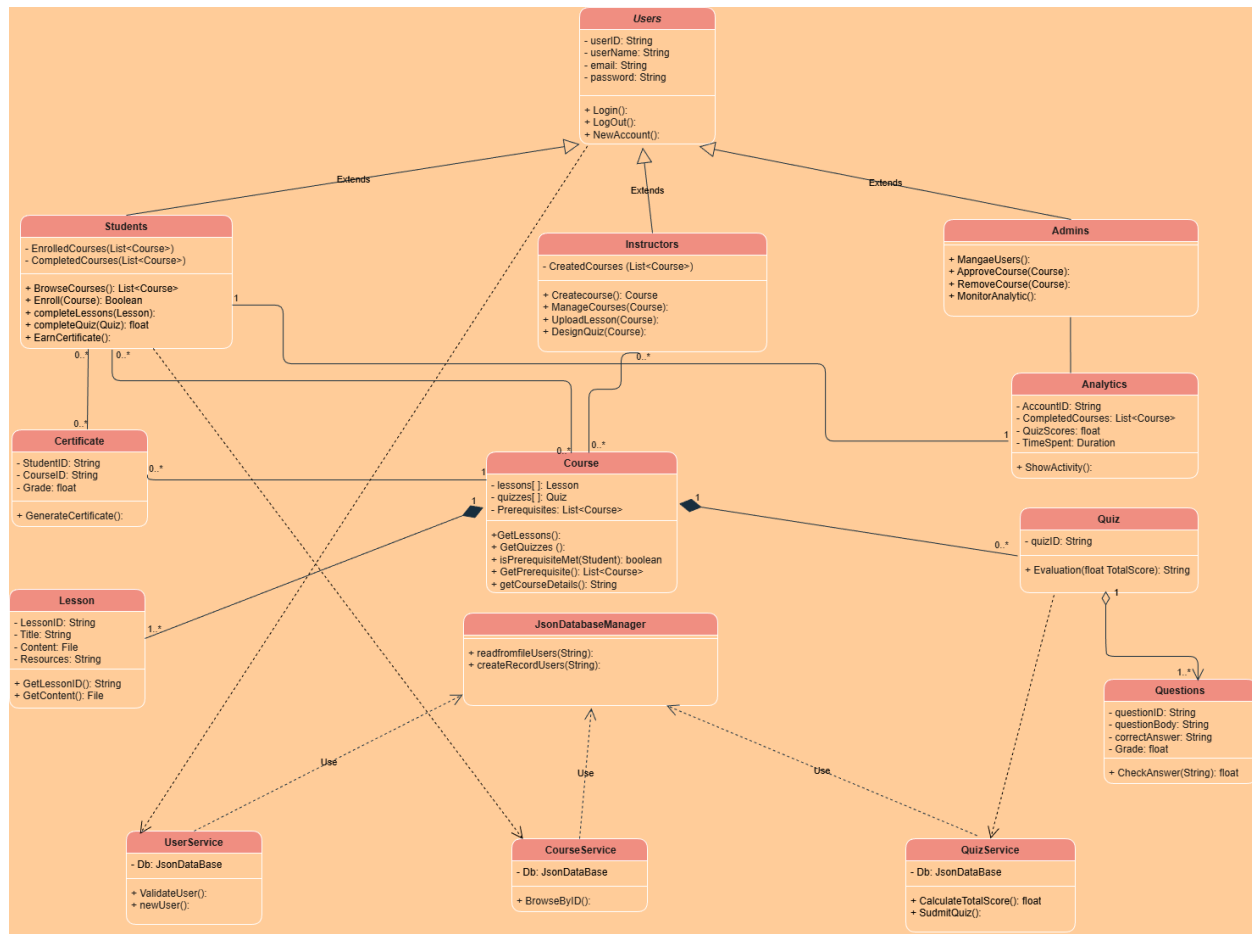
This diagram is important because it provides a high-level overview of system functionality and helps identify user requirements before detailed design and implementation.

2. Activity Diagram



Activity diagram showing the flow of enroll method where the student need to have the prerequisite courses that one course might need ,if not the system will show the prerequisite courses for the student to enroll in if they would like

3. Class Diagram



Class diagram showing the overall system of SkillForge , it shows all classes needed to implement the system, main methods of each class and the relationship between them.

```
sequenceDiagram
    participant User
    participant System
    participant student as student : Students
    participant course as course : Course

    User->>System: BrowseCourses()
    activate System
    System->>student: BrowseCourses()
    activate student
    student-->>System: return list <Course>
    deactivate student
    System-->>User: Display Courses
    deactivate System
    User->>System: View Course
    activate System
    System->>course: getCourseDetails()
    activate course
    course-->>System: return course details
    deactivate course
    System-->>User: Display Course Details
    deactivate System
    User->>System: Enroll (Course)
    activate System
    System->>student: Enroll(course)
    activate student
    student->>course: isPrerequisiteMet(student)
    activate course
    course-->>student: return boolean
    deactivate course
    student-->>System: return boolean
    deactivate student
    System-->>User: "Enrollment Successful"
    deactivate System
    System->>System: {set Flag = false}
    alt [else]
        System->>course: GetPrerequisite()
        activate course
        course-->>System: return list <Course>
        deactivate course
        System-->>User: "Prerequisites Required: "
        System-->>User: Display Prerequisites
    else [Student Choice]
        alt [View Prerequisite]
            User->>System: View Prerequisite
            activate System
            System->>course: getCourseDetails()
            activate course
            course-->>System: return course details
            deactivate course
            System-->>User: Display Course Details
            deactivate System
        else [Enroll in Prerequisite]
            User->>System: Enroll(prerequisite)
            activate System
            System->>System: [Go Back]
            deactivate System
            System->>System: Go Back
            activate System
            System->>System: {set Flag = false}
            deactivate System
        end
    end
    alt [Go Back]
        User->>System: Go Back
        activate System
        System->>System: {set Flag = false}
        deactivate System
    end
    System->>student: BrowseCourses()
    activate student
    student-->>System: return list <Course>
    deactivate student
    System-->>User: Display Courses
    deactivate System
```

This sequence diagram represents the flow of the system when a user wants to enroll in a course.

It is important as it provides common understanding for clients, developers, testers, ...etc.

Also, it visualizes the system's behavior over time as it displays the order of operations and timing and shows how objects interact in a specific scenario.

(**Note:** As the Sequence Diagram Picture is not clear, please check the following link for GitHub where you can find the picture under the name **Sequence Diagram Final.drawio.png**)

<https://github.com/YasminHaytham/UML.git>