**CMPS 350 Project Phase 1 – Report**

**Education Platform**

**(10% of the course grade)**

**The report must be submitted in Word format only**

|  |  |
| --- | --- |
| **Group Members** | Ghanim Mubarak Alkuwari (202208523)  Fahrel Azki Hidayat (202206836)  Mohammad Alam (202211429)  AbdulWasay Saqib (202211598)  **Emails:**  ga2208523@qu.edu.qa  fh2206836@qu.edu.qa  ma2211429@qu.edu.qa  aw2211598@qu.edu.qa |
| **GitHub link** | <https://github.com/fahrel-fh2206836/student-management-app> |

**Grades :**

**The student fills only the “Implementation Percentage”: Please specify either: *Working (completed x%)*, *Not Working (completed x%)* or *Not done*.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Criteria** | **Points** | **Implementation Percentage** | **Implementation Quality** | **Your Grade** |
| Design and implement the app Web UI and navigation using HTML, CSS and JavaScript. Including designing the App Web UI and navigation. | 50 |  |  |  |
| Design and implement the Web API and data access repositories to read/write the app data JSON files. | 30 |  |  |  |
| Application modeling (e.g. UML diagrams) to explain the data entities and the functionalities | 5 |  |  |  |
| Testing documentation using screen shots illustrating the testing results. | 5 |  |  |  |
| Team work quality. Contributions of team members - All members should collaborate and contribute equally to the project. | 5 |  |  |  |
| Project report – description of the implemented app, what is implemented, what is missed .. | 5 |  |  |  |
| **Total** | 100 |  |  |  |
| **Plagiarism, outsourcing, free riders** | -100 |  |  |  |
| **Delivery behind the deadline** | -5 |  |  |  |

**Important remark: In case of copying and/or plagiarism or not being able to explain or answer questions about the implementation, you lose the whole grade.**

**\* Criteria for grading the functionality:**

- The functionality is working: you get 70% of the assigned grade.

- The functionality is not working: you lose 40% of assigned grade.

- The functionality is not implemented: you get 0.

- The remaining grade in all cases from above **is assigned to the quality of the implementation**,

- The grades are distributed on the various use cases, when the design/implementation is partial, you get only the grades of designed/implemented use cases.

Code quality criteria, include:

- Use of meaningful identifiers for variables and functions (e.g. using JavaScript naming conventions)

- Pages are responsive

- Clean code: simple and concise code, no redundancy

- Clean implementation without unnecessary files/code

- Use of comments where necessary

- Proper code formatting and indentation.

**You lose marks** for code duplication, poor/inefficient coding practices, poor naming of identifiers, unclean/untidy submission, and unnecessary complex/poor user interface design.

**Important Remark**:

**[Grades: 100-85]:** Will be given only to **fully functional application** with **all the quality criteria cited above met** and the project has excellent **design for the various functionalities**. **The report is professional**.

**[Grades: 85-80]:** Will be given only **to functional application** **with most of all the quality criteria cited above met** and the project has good design for the various functionalities. **The report is professional**.

**[Grades: 80-75]:** 80% of the application functionalities are functional. The project respects partially the quality criteria. **The report is professional** but misses some information.

The grades are not negotiable. We expect that only a small portion (around 15%) of the class will be able to meet the criteria for the grades **[100-85]. You should work hard to and demonstrate the merits of your application to earn those grades.+**

# Description of your proposed platform

A Platform for course management with three different types of users that can login into the system: Admin, Student, Instructors.

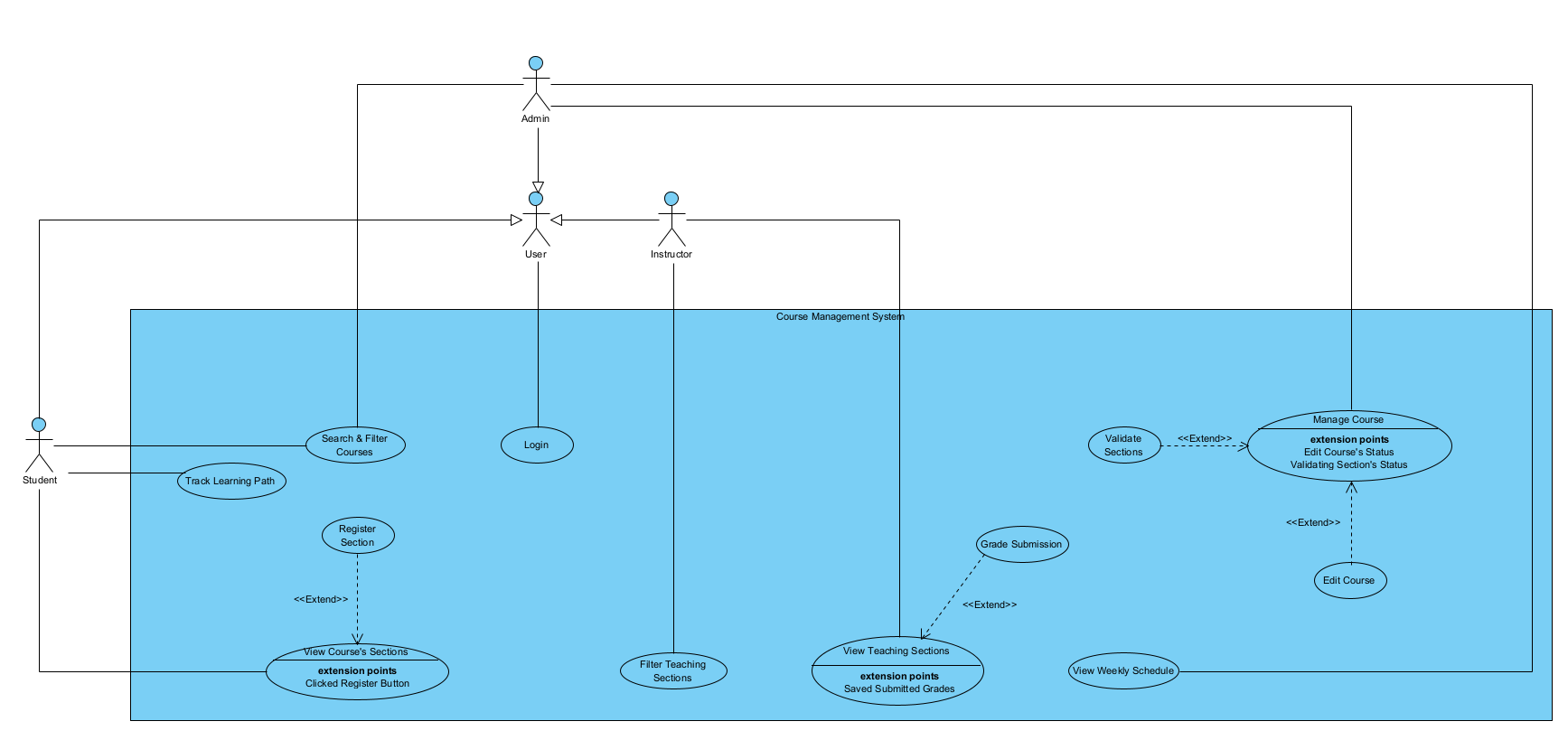
Admins can View All Courses and Sections, Add Courses, Add sections of a course, View Weekly Schedules of all Courses, and Manage courses which include editing courses and section’s status and validating sections to be approved.

Students can View their Ongoing Courses, Browse and Filter other courses, View and Register sections of a course, and View their learning path which includes their progress statistics (Credit hour completed etc.) and list of courses they have completed, In-progress, and pending.

Instructors can View their previous, future, and ongoing teaching sections and Submit student grades of a section.

# Application Design

# Use case diagram



# Entities class diagram

A diagram of a computer program

AI-generated content may be incorrect.

# Web API class

**Web API was not Implemented.**

# Webpage UI Skecthes

**UI sketches can be found in the repo under Diagrams/UI SKECTH.pdf**

# Implementation

# Implemented use-cases

**All Use cases (1,2,3,4,5,6,8) were implemented and functional.**

# Unimplemented use-cases and not functioning parts

**All Use cases (1,2,3,4,5,6,8) were implemented and functional.**

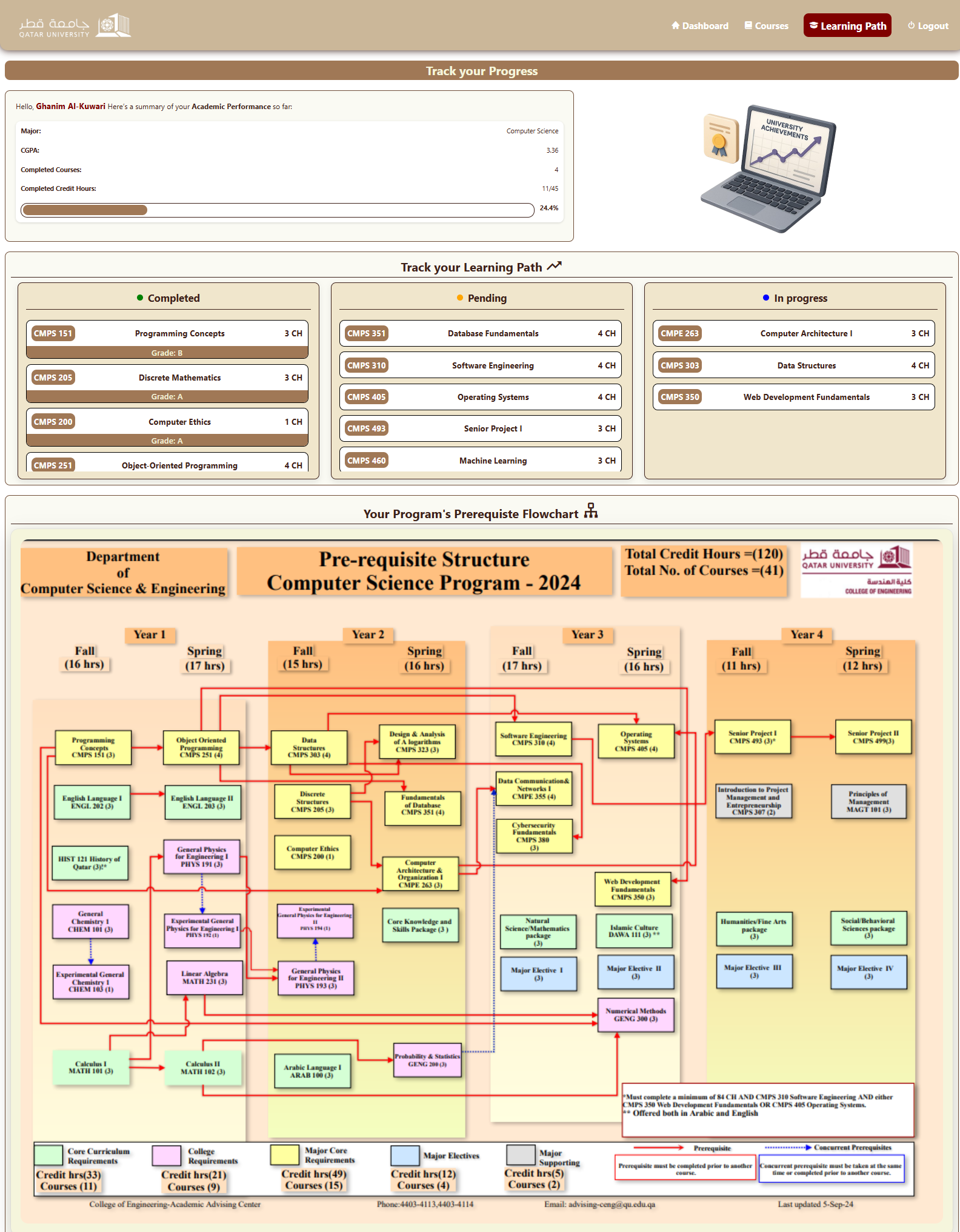
# Testing

# Use case 1

# Use case 2

# Use case 3

# Use case 4



# Use case 5

# Discussion of the project contribution of each team member

|  |  |
| --- | --- |
| **Student name** | **Student contributions** |
| Ghanim Mubarak Alkuwari | Completed use case 4 (learning paths), coded the javascript for use case 1 (login) and Performed Testing of Use Cases |
| Fahrel Hidayat | Completed Use Cases 5 and 8. Led the Styling and set the system’s theme styles. UI sketching and Created Entity Relationship and Use Case Diagram. Performed Testing of Use Cases. |
|  |  |
|  |  |
|  |  |