**CMPS 350 Project Phase 1 – Report**

**Education Platform**

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

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

|  |  |
| --- | --- |
| **Group Members** | Ahmed Alamoudi 202104223  Faisal Taleb 202205767  Omar Aboelrous 202008853  **Emails:**  Aa2104223@qu.edu.qa  ft2205767@qu.edu.qa  oa2008853@qu.edu.qa |
| **GitHub link** | Give a public link to you code :  <https://github>.com/Ahmed-aa2104223/Web-Project.git |

**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 | 50 |  |  |
| Design and implement the Web API and data access repositories to read/write the app data JSON files. | 30 | 30 |  |  |
| Application modeling (e.g. UML diagrams) to explain the data entities and the functionalities | 5 | 5 |  |  |
| Testing documentation using screen shots illustrating the testing results. | 5 | 5 |  |  |
| Team work quality. Contributions of team members – All members should collaborate and contribute equally to the project. | 5 | 5 |  |  |
| Project report – description of the implemented app, what is implemented, what is missed .. | 5 | 5 |  |  |
| **Total** | 100 | 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

Our application is a web-based application where different users of the university can interact with it.

* The students can view all the courses available and the courses they have registered/finished, they also can register for new courses and view their GPA depending on their grades.
* The administrator can view all the courses that are available to register and either validate or cancel the course, they also can create a new course or a new class that has an instructor/seat etc.
* The instructor can view all the students that are registered to their course/class, and they can assign them their final grades.

# Application Design

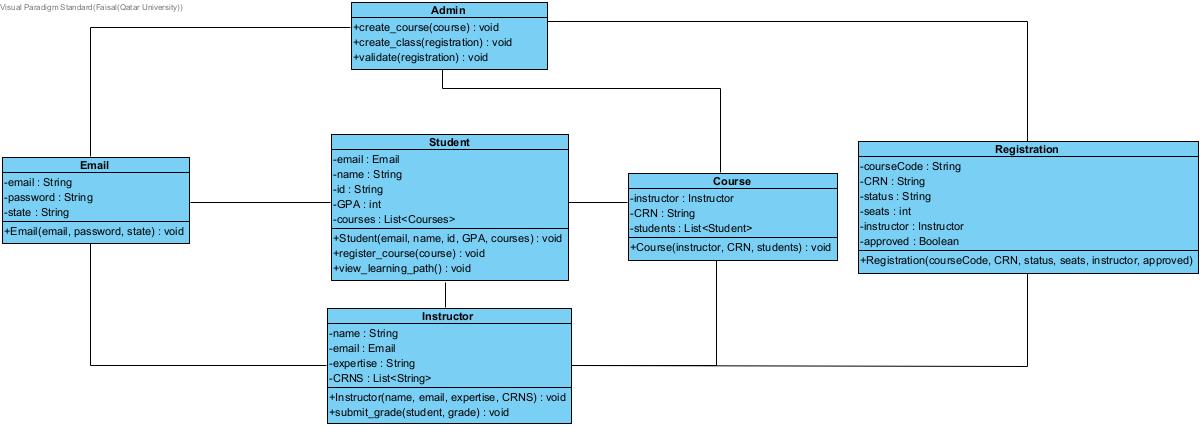
# Use case diagram

A diagram of a student management system

AI-generated content may be incorrect.

# Entities class diagram

Describe your data as a class diagram or Entity Association diagram



# Web API class

Fetching from a JSON file and storing all the data to a Local Storage

# Implementation

# Implemented use-cases

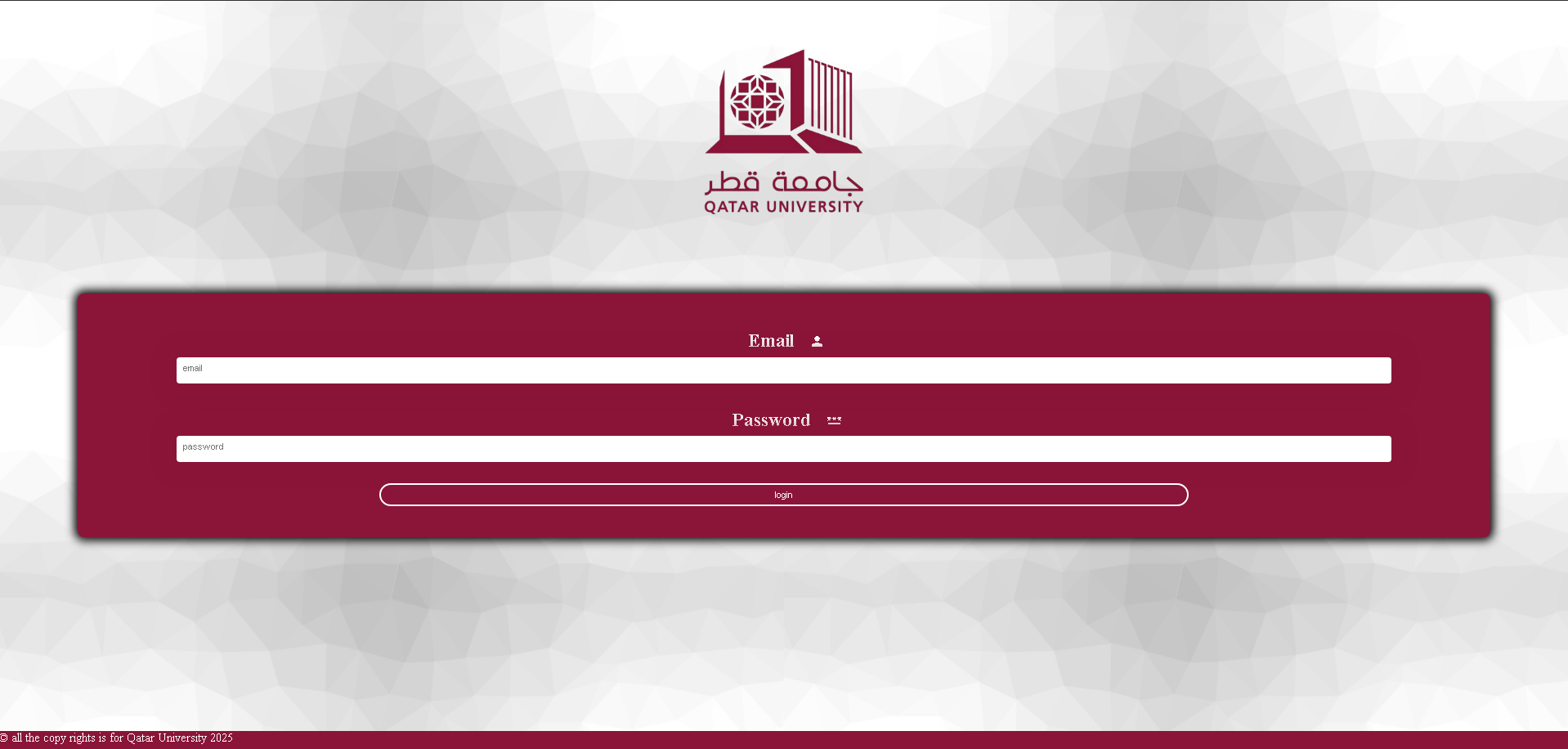
We have implemented every use case and every one of them works perfectly

# Unimplemented use-cases and not functioning parts

None of the use cases are non-functional

# Testing

# Use case 1



# Use case 2

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

# Use case 3

A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.A white background with black text

AI-generated content may be incorrect.A white rectangular sign with black text

AI-generated content may be incorrect.

# Use case 4

A screenshot of a computer

AI-generated content may be incorrect.

# Use case 5

A screenshot of a computer

AI-generated content may be incorrect.A white screen with black text

AI-generated content may be incorrect.

A white and black lines on a white background

AI-generated content may be incorrect.A white screen with black text

AI-generated content may be incorrect.

# Discussion of the project contribution of each team member

|  |  |
| --- | --- |
| **Student name** | **Student contributions** |
| Ahmed Alamoudi | 33% |
| Faisal Taleb | 33% |
| Omar Aboelrous | 33% |