**CMPS 350 Project Phase 2 – 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** | **%** | **Functionality**\* | **Quality of the implementation** | **Grade** |
| Design and implement the Data Model. | 10% | 1**0** |  |  |
| Init DB: populate the database with the data from the json files in seed.js | 5% | 5 |  |  |
| Server actions, APIs and Repository Implementation to read/write data from the database | 25% | 25 |  |  |
| Statistics use-case with NextJS | 40% | 40 |  |  |
| **Documentation**  - Data Model diagram.  - UI Design with screenshots and description.  - Database queries.  - Conducted tests and evidence.  - **Contribution** of each team member [-10pts if not done] | 20% | 20 |  |  |
| **Total** | 100% | 100 |  |  |
| Copying and/or plagiarism or not being able to explain or answer questions about the implementation. | -100 |  |  |  |

**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 platform designed for university users to interact with course-related functionalities. The system supports three primary user roles: Students, Administrators, and Instructors, each with distinct capabilities:

* 1. **Students**
     + Browse all available courses.
     + View their registered and completed courses.
     + Register for new courses.
     + View GPA based on their final grades.
  2. **Administrators**
* Manage course availability (approve or cancel courses).
* Create new courses and assign them to classes.
* Configure class details such as assigned instructors, available seats, and schedules.
  1. **Instructors**

 View all students registered in their courses.

 Assign final grades to students.

# Data Model

A screenshot of a computer

AI-generated content may be incorrect.

# Web API, Server Actions and repository

**User Methods**

addUser(user)

getUserByEmail(email)

getUsers(type)

**Instructor Methods**

addInstructor(instructor)

getInstructors()

getInstructorWithCourses(id)

**Student Methods**

addStudent(student)

enrollStudent(studentId, courseOfferingId, status = 'enrolled')

getAllStudents()

getStudents()

getStudentCourses(studentId)

getStudentsWithGPAUnder(threshold = 2.5)

getTopPerformingStudents(limit = 5)

getFailingStudents()

getStudentsEnrolledInCourse(CRN)

getCoursesCompletedByStudent(studentId)

**CourseDefinition Methods**

addCourseDefinition(course)

getAllCourses()

**CourseOffering Methods**

addCourseOffering(offering)

updateCourseOffering(CRN, data)

getCourseOffering(CRN)

**Prerequisite Methods**

addPrerequisite(courseId, prerequisiteId)

getCourseWithPrerequisites(id)

**Reporting / Analytics Methods**

getEnrollmentReport()

getCourseEnrollmentCount(CRN)

getAverageGradePerCourse()

getStudentCountPerInstructor()

getMostPopularCourses(limit = 5)

getCourseFailRate(CRN)

A white background with black text

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.A screen shot of a computer code

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

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

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

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

# Implemented statistics use case

# User InterfaceA screenshot of a computer AI-generated content may be incorrect.

A screenshot of a course statistics

AI-generated content may be incorrect.

# Implemented queries

We have implemented all the data by sending them into the Prisma

A screenshot of a computer program

AI-generated content may be incorrect.

A screen shot of a computer code

AI-generated content may be incorrect.

# Data used in the statics

This is the data that we are using. It is all in the **prisma**

A screenshot of a computer

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

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

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

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

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

AI-generated content may be incorrect.

# Conducted tests

To ensure the reliability and functionality of the application, several tests were conducted during development:

**1. Manual Testing**

* Each user role (Student, Instructor, Administrator) was tested manually to verify the correct functioning of key features such as:
  + Student course registration and GPA calculation.
  + Instructor grade submission.
  + Administrator course creation and validation.
* Form inputs were tested for validation, including required fields and invalid input handling.

2. **Functional Testing:**

* Verified that the routing between pages works as expected.
* Ensured that authentication and role-based access control were functioning properly (e.g., only admins can create courses).

3. **API Testing**

* Backend API routes (e.g., for course registration, grade submission) were tested using tools like **Postman**.
* Ensured that correct responses were returned and edge cases (e.g., duplicate registrations) were handled.

# Implemented queries

A screenshot of a computer program

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% |