**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 application where different users of the university can interact with it.

* 1. 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.
  2. 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.
  3. The instructor can view all the students that are registered to their course/class, and they can assign them their final grades.

# 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 screenshot of a computer

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 screen shot 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

AI-generated content may be incorrect.

# Implemented statistics use case

# User Interface

A screenshot of a course statistics

AI-generated content may be incorrect.

# Implemented queries

# Data used in the statics

# Conducted tests

# Implemented queries

# Discussion of the project contribution of each team member

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