PROJECT MILESTONE TWO

Group Name

Software Engineering Slayers

Application Overview

The information specified in the user and system requirements in this document was obtained from two primary sources. The first source was a Google forms survey of which over fifty UWI students participated in. The survey was conducted on October 2, 2018. The students in the survey were from various faculties and were at various levels in their degrees. The survey consisted of mainly closed choices and a few open-ended questions. The open-ended questions helped the group to gain insight as to what the clients required in an application of this scope. There were many trends and patterns that were ascertained throughout the response data set that helped with the extraction of the user requirements. However, the second source was what solidified the clients' requirements for the application. The second source was interviews. The interviews were conducted on the evening of October 2, 2018. A total of fifteen participants were interviewed by group member Maya Bannis. Through these interviews, the group was able to ascertain five additional requirements, as well as clarify many of the requirements already captured from the survey.

A copy of the questionnaire form can be found [here](). A copy of the interview questions can be found [here]().

User Requirements

1. Users shall be able to log into the application.

2. Users shall be able to define a new grading schema.

3. Users shall be able to add previous and current courses to be calculated.

4. Users shall be able to edit courses.

5. Users shall be able to delete courses.

6. Users shall be able to save weightings and due date information for assignments, exams and projects for each of their courses

7. Users shall be able to set a desired target grade for each of their courses.

8. Users shall be able to view the minimum set of grades needed to achieve a passing grade or their pre-set target grade.

9. Users shall be able to get warnings when their predicted grades fall below passing or their target grade

10. Users shall be able to view their automatically calculated cumulative GPA

11. Users shall be able to view their automatically calculated degree GPA

12. Users shall be able to save grades attained on courses they have already completed.

13. Users shall be able to save assessment scores on courses they are currently undergoing.

14. Users shall be able to set a desired target GPA.

System Requirements

1. The system shall allow users to securely log into the application.

2. The system shall store GPA weighting information.

3. The system shall allow users to enter their results to be calculated.

4. The system shall store a user's examination results data.

5. The system shall store a user's assignment results data.

6. The system shall be able to allow users to add courses.

7. The system shall be able to allow users to remove courses.

8. The system shall work offline.

9. The system shall be able to connect to a database.

10. The system shall calculate cumulative GPA

11. The system shall calculate Degree GPA

12. The system shall automatically recalculate data when edits are made

13. The system shall update data displayed according to course/grade edits

14. The system shall calculate the minimum number of grades the user needs

15. The system shall calculate the minimum amount of course and exam marks the user needs

16. The system shall allow users to customize the behavior of the application's notifications.

Non-functional Requirements

1. The system shall be secure i.e. the system should protect student data.

2. Resilient and Capacity i.e. the system should be able to handle increase of users without failures or overload.

3. The system shall focus on performance i.e. meet targeted processing and response time.

4. The system shall have high availability i.e. the system should not have unexpected downtime.

5. The system shall focus on Usability i.e. have an Interface that is intuitive. The system shall behave according to a user's intuition.

6. The system shall be maintainable i.e. clean and clear documentation, easy to comprehend code and a reusable code schema.

Use Case Diagram

See next page…

Activity Diagram

TBA…

Sequence Diagram

TBA…