**Test Plan For the GPA Calculator Application**

**Ziyad Alkhenin**

**7-31-2015**

**Final**

**“On my honor, I have neither given nor received aid on this assignment”**

Table of Contents

Description: 1

Test Scope: 2

Testing strategy: 2

Level of effort: 4

Qualification Requirements for the Tester: 4

Entry criteria: 4

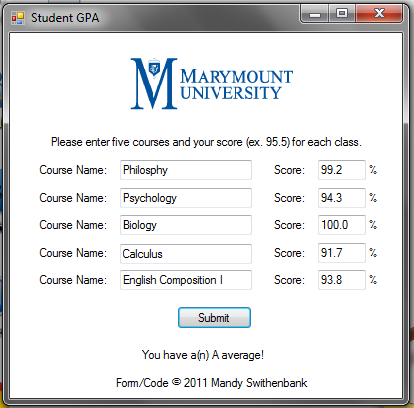
Exit criteria: 5

### Description:

We are going to test an application that calculates the overall GPA for students in a specific semester. Students should be able to enter one to five courses with their midterm grades. The midterm grades are entered as a number between 0 to 100 and the application is supposed to convert the grades into a GPA. Moreover, the application must display the student’s name and the grade is expressed as a letter where A= 91- 100, B= 81-90, C= 71- 80, any other way, the grade is F. Furthermore, the application should contain the Marymount University’s logo at the top of the page.

The intended audience of this application are Marymount University students. The application would be very helpful for the students since it will give them the opportunity to see how they are doing in the courses after completing their midterms. Therefore, students will find out weather they have to put more effort to raise their GPA or the amount of effort they have put is good enough. The application would be a great tool to make students attain their GPA goals.

**How the application will be used?**



Even though the application is a great tool for students to achieve their GPA goal, it is very simple to use. Students only have to put the courses names in the “Course name” fields and enter the midterm grade for each course. The application in return will display their overall GPA as shown in the image.

### Test Scope:

We will test the GPA calculator application that is being developed by Mandy Swithenbank. The application is very sensitive because it interacts with the official students records and grades, therefore, it will not be available on the website until the testing process in done. We will use different test techniques to make sure that the application works perfectly fine and 100% accurate. The techniques that we are going to use are: Functional Testing, Integration Testing, Performance Testing and finally User Acceptance Testing.

### Testing strategy:

**Functional Testing:**

Functional testing is performed basically to verify that the application functions exactly according to the design specifications. During this test technique, testers focus on the core application functions, such as test input and menu functions. We will perform this technique to make sure that when grades are entered, the application should display the student’s name with the midterm grade expressed as a letter. Also, the system should calculate the overall GPA and displays it to the user. If some fields of the application are missing, the system should indicate which fields are missing, so the student can fill them out.

**Integration Testing:**

This technique usually goes after unit testing and before system testing. Integration testing is basically a level of software testing where units are combined together and tested as a group. The main purpose of this technique is to expose faults and bugs in the interaction between integrated units. We can use this testing technique to see if the different modules of the system interact with each other properly and so not expose any faults. For example, when a student enters his/her grades and hits “ Submit”, the output should be the student’s name with the overall GPA not the “Register for sections” page.

**Performance Testing:**

This testing technique is a non-functional testing method. This testing technique determines how fast a system is under some particular load. It demonstrates that the system meets the performance criteria. This test technique is very important in our test plan to make sure that when a large number of students log into the application at the same time and enter their grades to calculate their overall GPA, the application does not slow down or stops responding.

**Acceptance Testing:**

Acceptance test is basically a test to determine the requirements are met. Acceptance test takes place at the end of testing process after all parts have been tested. The goal of this test is to establish confidence in the new system or application. For instance, the application calculates the grades accurately and displays them to the users.

### Level of effort:

This is a small simple application; therefore, testing it should not take a long time. We need 22 days to finish testing this application and make the necessary changes. I made my assumption because I think each test strategy will take maximum 5 days, and I put two extra days in case that we have to put extra effort on one of the testing strategies.

### Qualification Requirements for the Tester:

The tester of this application should have sharp analytical skills because it will help to break down the complex of the application functions into units and fix the bugs if there are any. Also, it is important that the tester has a very good verbal and written communication skills since the tester must write test cases, test strategies and bug reports. Finally it is essential that the tester is experienced at software testing and has tested some applications in the past.

### Entry criteria:

* Necessary documentation and requirements information should be available before the testing process, so the tester can judge the correct behavior of the system.
* Some automated testing tools and Microsoft Word should be installed before the testing process.
* Proper test data should be available before the testing process.

### Exit criteria:

* Users and managers should identify what parts of the system must work correctly 100% to pay more attention to these parts and make sure there is absolutely no single bug in these parts.
* There should be specified indicators or measures of the system that you are looking for.
* All test cases have been run.
* All high-risk areas have been fully tested; only minor residual risks are left outstanding.
* All severe bugs were fixed.