**Test Plan**

**Brendan College, Alejandro Fernandez, Charles Karlson, Samantha Maddox**

**Project 1: Group Matching App**

**COP 4331, Fall 2021**

**Contents of this Document**

Overall Objective for Software Test Activity

Description of Test Environment

Overall Stopping Criteria

Description of Individual Test Cases

Appendices

**Overall Objective for Software Test Activity**

* A few sentences stating what you expect the software test effort to accomplish
* Test cases and testing should attempt to accomplish at least one of the following criteria:
  + Verification of general operation and behavior
  + Correcting behavior of the program under unexpected conditions or edge cases
  + Testing security measures through replication of malicious or careless behavior

**Description of Test Environment**

< What is the hardware and software in the environment in which you will run the test? Who will be the testers (actual users? developers? ...) Will this test environment be the same environment in which the software will operate? If not, how does it differ?>

Tests should be performed in vitro isolated to different modules of the program. This shall be performed by individual developers on developers’ systems before committing any modules that have been developed on or under maintenance. Multiple test cases shall be used.

Once in vitro tests have been passed, tests should be performed on the whole program in vivo. These tests may be performed by developers on application systems. Bug reports submitted by users may also be evaluated by developers and attempts may be made to reproduce the issue. Multiple test cases may be used.

**Stopping Criteria**

<How will you determine when to stop testing the software and either deliver it or send it "back" to development? Things to consider:

* If you find errors during testing: Will you stop testing each time you find a problem and immediately fix that problem? Will you continue testing and recording errors until you find a fatal error that won't allow you to continue? Will you test for 2 hours and then fix whatever errors have been found? Will you test for 2 hours and then hold a group meeting to decide whether to continue to test? ...
* If you find no errors during testing: Does that mean that your software is error-free? How many test cases will you run before you declare the software to be "good enough to deliver"?
* How do you define "good enough to deliver"? Does it require that there are no known errors? Or no known errors other than cosmetic errors? Or no known errors other than cosmetic errors and errors for which there is a well-defined workaround? .... >

Please clearly state your stopping criteria.

Tests will conclude on a specific issue once the specific issue has been fixed and the related modules perform as intended. These conclusions should be verified by other developers and shall be demonstrated to management upon request. “Fixed” shall mean that the concerned operations perform as intended under multiple sets of test cases. “As intended” should mean that the operation behaves normally, repeatably, and safely.

Time spent on testing should proportionally match the severity of the issue. The issue should be relegated to further maintenance if higher priority issues arise or if a different module lacks functionality considered more important. In summary, work performed on modules (whether it be development, testing, or maintenance) should be prioritized by the developer according to their own intuition and expertise.

Modules should be in a complete-enough state before performing work on other modules. “Complete-enough” can be defined as retaining most of the module’s functionality at that point (i.e. if the module relies on another undeveloped module, that module can be complete up to the point where it will need to rely on the undeveloped module).

The program will be considered for launch when almost all of the functionality and cosmetic appearance are demonstrated successfully.

**Description of Individual Test Cases**

Describe EACH individual test to be run. For instance, if you plan to run 30 test cases, you would answer the following questions for each of them. A table format or a bulleted format is acceptable.

* Test Objective: <exactly what does this specific test demonstrate?>
* Test Description: <exactly what will you test? What test data will be used (specifically -- what data values, what data files? This data must be determined in advance. So if you plan to use a test file, include that file in an Appendix of this document. If you plan to key in specific data, include the data here. Make sure the data you choose will allow you to achieve your stated objective for this test.>
* Test Conditions: <Under what conditions will you run this test? This is relevant for software for which there are multiple "modes". For some systems, the test conditions are totally described in the above test environment section -- in this case, the response to this would be "See Test Environment".
* Expected Results: <If the test executes correctly, what will be the result -- i.e., exactly what will the output look like; what will be the resulting data in the database, etc...>

**Trace of Individual Test Cases to Requirements**

< In this section, provide a trace of individual test cases to requirement(s): for each test case, indicate which requirement(s) are validated by that test case? Please make sure each requirement appears at least in one test case and some requirements may appear in multiple test cases >

Please fill the cells of this table for Requirements Traceability Matrix

|  |  |  |  |
| --- | --- | --- | --- |
| Requirement ID | Requirement Description | Test Case Reference | Status |
| Requirement ID that is stated in the SRS document | The system shall… | Write the test case IDs that are responsible to test this feature. There may be more than one tests responsible for this requirement | Status of the tests, failed, succeeded or in progress |
| .. | .. | … | … |

**Appendices:**

Include any test files used in your test cases. If you do not use any test files, leave this section blank.