GROUP-5

Project Test Plan

**1. Introduction**

* 1. Test Plan Objectives

The project involves dealing with a very practical approach to make a program for local delivery company that has three different trucks on three different routes. The

map of the city have been divided into square grids (25\*25). All the trucks are of

same size and can hold upto 1000 kg of cargo and 36 cubic meters of boxes.

The program asks us to do the following tasks: When a shipment comes in, find a truck which is big enough to hold the shipment as well as finding a truck which is going to go as close as possible to the destination of the package. The trucks can divert slightly from their assigned routes and pass through any of the white parts of the grid to deliver the package.

Once we complete the whole program’s coding part, we are supposed to move to the testing part where we will test the whole program through different testing and debugging methods to make sure the program is bugs free. We will perform a series of tests which include Black box Tests, White Box Tests, Integration tests and acceptance tests. The whole project is intended to give us a practical knowledge of How to test a software in a professional team environment.

**2. Scope**

The complete scope of the project includes testing every single piece of code. The code that is already given to us is not tested. We will add some more code to the program which needs to be tested as well.

Once the coding part is completed, we will go through a series of tests which includes all or most of the testing methods we have learned so far.

**3. Test Strategy**

The testing phase will be the most challenging phase of the project as the code will go through many type of tests. Here is a list of different type of test which will be performed :

(1) System Test

(2) Performance Test

(3) Security Test

(4) Automated Test

(5) Stress and Volume Test

(6) Recovery Test

(7) Documentation Test

(8) Beta Test

(9) User Acceptance Test

**4. Environment Requirements**

Environment requirements refer to the specific hardware, software, and network configurations needed to conduct software testing effectively. These requirements are necessary to ensure that the testing environment accurately reflects the end-user environment in which the software will ultimately be deployed.

environment requirements for this project include:

Operating system: The software needs to be tested on different operating systems, such as Windows, MacOS, or Linux.

Hardware: The software needs to be tested on different hardware configurations, such as different processors, memory, or storage devices.

Network: The software needs to be tested in different network environments, such as LAN, WAN, or mobile networks.

By ensuring that the testing environment meets all of the necessary requirements, software testers can accurately identify and address any issues that may arise before the software is released to end-users.

It may not be possible to create and test in all the above specified environments as per the scope of the project but all the above specified requirements apply for most of the software testing projects.

**5. Execution Strategy**

The execution strategy is the plan for how software tests will be performed. It includes the criteria for when a test will start and finish. For example, a test might be considered complete if it passes 95% of the test cases, or if there are no severe or critical defects found during testing.

For the scope of this project, the test is considered to be completed if it is completely bug free (or at least 99% bug free).

The entry criteria for a test is having a complete set of test cases and a stable environment to perform the testing. The exit criteria is meeting the pass rates or resolving all critical defects before the testing is complete.

The execution strategy helps ensure that the testing process is consistent and thorough, and that any issues with the software are identified and addressed before it is released to end-users.

The error severity level of the bugs found can be in the ascending order like : Cosmetic, Low, medium, high, critical. Here is the brief explanation of each term :

* + 1. **critical** which cause the system to crash or produce anomalous results,
    2. **high** which causes lack of program functionality and might have a work around,
    3. **medium** which is a bug which D crates degrades the quality of a system but often has a work around to give the desired functionality
    4. **Low** which might be an unclear error message or some other minor error that has minimum impact on functionality
    5. **Cosmetic** which is something that makes the user interface less than optimal but still perfectly functional.
  1. **Test Reporting**

Test reporting refers to the process of documenting and communicating the results of software testing. This includes what reports should be produced, how often they should be produced, who should receive them, and what information they should contain.

In this project, the testing part will be divided among all group members and everyone will perform each test in their environment. All the tests reports will be submitted to the group leaders who will further check the test reports thoroughly and document each and every thing professionally. The report will include all the tests that are passed as well as all the test which failed with a detailed description of what made them to pass or fail.

A group discussion will be done on the tests which failed so that we can figure out answers of the following questions :

What is the exact issue ?

Where is it located ?

How can it be fixed ?

What did we learned from it ?

What needs to be taken care of ?

This group discussion will help all the members to learn and work effectively while putting the best team efforts to deliver a high quality program to the end user.

**6. Test Schedule**

|  |  |
| --- | --- |
| **WEEK-1 [24TH FEB TO 3RD MAR]** | **Repositories and Jira Account Setup by all team members [DONE]** |
| **WEEK-2 [4TH MAR TO 16TH MAR]** | **Create complete Project Plan and add new Data Structures to the program. [DONE]** |
| **WEEK-3 [17TH FEB TO 23RD MAR]** | **Design required functions and its specifications. Start designing the Black box tests specifications as well.** |
| **WEEK-4 [24TH MAR TO 31ST MARCH]** | **Complete the implementations of the functions and perform Black Box tests.** |
| **WEEK-5 [1ST APR TO 7TH APR]** | **Create, Implement and Execute Integration Tests** |
| **WEEK-6 [8TH APR TO 14TH APR]** | **Perform Acceptance tests on the complete program.** |

**7. Control Procedures**

Control procedures are an important aspect of software testing that help to ensure that the testing process is efficient and effective. There are several different control procedures that are commonly used in software testing, including:

Reviews : We will have a weekly group meeting to review and modify any changes in the test plan.

Bug Review Meetings : We will have meeting to resolve meetings for the bugs determined. The purpose of these meetings is to identify the root cause of the defect and determine the appropriate course of action to resolve it.

Change Request : After discussion in the group meeting, it will be decided if there will be any change necessary to be implemented in the project.

Defect Reporting : This step will involves documenting and tracking defects that are identified during testing. This includes capturing information about the defect, such as its severity and impact, and tracking its status through to resolution.

By implementing these control procedures, our team will ensure that defects are identified and resolved in a timely manner, and that the software meets the necessary quality standards.

**8. Functions To Be Tested**

Many of the functions were already given to us and many more functions will be created by our group. All of them needs to be tested in every possible way. All the functions will be tested as per the tests mentioned above in the testing strategy.

**9. Resources and Responsibilities**

Resources refer to the tools, equipment, and personnel required to perform testing activities. Responsibilities refer to the tasks, roles, and activities that individuals or teams are responsible for in the testing process.

Here are some of resources and responsibilities of our group in testing this project:

**Resources:**

(1) Testing tools such as test management software, automated testing tools, and defect tracking software.

(2) Test data such as sample data.

**Responsibilities:**

(1) Test planning, including creating test plans, test cases, and test scenarios.

(2) Test execution, including running test cases and recording test results.

**Test management, including managing the testing process, coordinating with development teams, and ensuring that testing is completed on schedule :**

(1) Test automation, including creating and maintaining automated test scripts and frameworks.

(2) Test analysis, including analysing test results to identify trends, patterns, and areas for improvement.

(3) Test environment management, including ensuring that the test environment is set up correctly and maintained throughout the testing process.

(4) Test data management, including managing and creating test data and ensuring that it is accurate and up-to-date.

**10. Deliverables**

We need to deliver the following at the end of the project delivery date:

(1) Bug free Source Code

(2) Documentation (group contract form, test plan etc)

(3) Test Cases, Obtained result

**11. Suspension / Exit Criteria**

Exit criteria are the conditions that must be met before the testing phase can be completed and the software can be released. These criteria include:

>>> The completion of all planned test cases and the achievement of predetermined test objectives.

>>> The resolution of all critical defects and the confirmation that all other identified defects have been addressed satisfactorily.

**12. Resumption Criteria**

Resumption criteria in software testing refer to the set of conditions or requirements that must be met to resume testing after it has been suspended. These criteria are established to ensure that testing can be resumed effectively and that the testing objectives can still be achieved.

Resumption criteria may vary depending on the reason for the suspension, but they generally include:

(1) Resolution of the issue that caused the suspension: Any issues that caused the suspension of testing, such as hardware or software problems, must be resolved before testing can resume.

(2) Restoration of the testing environment: The testing environment must be restored to its original state or reconfigured as necessary before testing can resume.

(3) Re-evaluation of the test plan: The test plan must be reviewed to ensure that it is still valid, and that any changes or adjustments needed to reflect the suspension have been made.

Resumption criteria are important to ensure that testing can be resumed effectively and that the results obtained after the suspension are reliable.

Top of Form

**13. Dependencies**

Dependencies in software testing refer to the factors or elements that are necessary for testing activities to be completed successfully. These dependencies may be related to people, processes, tools, or infrastructure, and they can have a significant impact on the testing process.

Dependencies in this project are:

(1) Availability of test data: The testing process may require specific test data, such as user information or specific data sets, which need to be available before testing can begin.

(2) Access to hardware and software: The testing process may require specific hardware or software configurations to be available to complete testing successfully.

**14. Risks**

There might be several risks that must be considered while testing this project :

(1) Inadequate test coverage: If the test coverage is insufficient, it may not be possible to identify all defects, and the software may not meet the quality requirements.

(2) Inaccurate requirements: If the requirements are inaccurate, incomplete, or ambiguous, it may be difficult to design and execute effective tests.

(3) Unforeseen changes: If there are changes in the project scope, requirements, or schedule, it may be necessary to re-plan or adjust the testing process.

(4) Defect leakage: If defects are not identified and addressed effectively, they may be carried forward into subsequent phases, leading to increased costs and reduced quality.

(5) Technical challenges: If the software being tested is complex, or if there are challenges with the testing tools, infrastructure, or environment, it may be difficult to complete testing activities effectively.

Identifying and managing risks is an important part of the software testing process. Risk identification should be done early in the project planning phase, and a risk management plan should be developed to address potential risks. Top of Form

Bottom of Form

**15. Tools**

**These are the tools used for this project :**

**(1) Visual Studio**

**(2) GitHub**

**(3) Jira Software**

**(4) Git Clone**

**(5) Tortoise Git**

**16. Documentation**

**Here is a complete list of documentations that are to be created and edited while working with this project :**

**(1) Group Contract Form**

**(2) Project Test Plan**

**(3) Test strategy template**

**(4) Traceability Matrix Template**

**(5) Test Report Analysis**

**17. Approvals**

**This project is to be approved by our Subject Professor Mr. Ziad Diab.**