Test Plan Template

1. **Introduction**
   1. Test Plan Objectives
      1. **To analyze the end user usage of the program and determine the structures and functions that are already available and the ones that needed to be developed for desirable and complete functionality.**
      2. **To determine the tests and test cases needed and their goals. To analyze their scope, constraints, and functionalities and assign them to suitable members of the team for easier management of the expectations and efficient testing.**
      3. **To break down the overall objectives into more manageable parts determining and the specific objectives of each test and overcoming them one at a time for each team member.**
      4. **To identify the constraints and limitations of the resources, external systems and tools to get the best possible testing configuration and tester for each case to produce the most efficient outcome.**
      5. **To identify the risks and create strategies to mitigate them.**
2. **Scope**
   1. **(Pre-developed/ Available) Mapping and route functions in mapping.c file.**
   2. **(To be developed/ Unavailable) Delivery, shipment, and route functions in the (yet to be developed) delivery.c file.**
3. **Test Strategy**

|  |  |
| --- | --- |
| **Project Name**  SFT Project/Truck Shipment Route | **Author**  Prabhjot Singh |
| **Computing Environment**  Integrated Development Environment | **Software Type**  Visual Studio 2022 |
| **User Demographics**  Truck Shipment Associates | **Assumptions**  Users have web experience.  Users know how to navigate general user interface |
| **Purpose of Test**  Verify correct route and program's useability | **Phases of Testing**  Acceptance Testing |
| **Scope of Testing**  Straightforward and correct user interface.  Completeness of data  Route functions ensure points are added correctly | **Critical Success Factors**  User Interface is straightforward, useable, and correct.  Data is presented completely and correctly. |
| **Testing Types**  Unit Testing  Manual/Written Tests | **Tester Profiles**  Mukul Sharma  Divya Rana |
| **Development/ Test Tools**  Visual Studio Test Suite | |
| **Business / Operational Concerns**  Incorrect values can cause inefficiency to the shipment route. | |
| **Risks**  **Business**  **- Incorrect values can suggest incorrect/inefficient shipment route**  **Technical**  **- Incorrect values can suggest incorrect shipment route**  **Project**  **- Needs to be tested before next phase of development** | |
| **Other** | |

1. **Environment Requirements**
   1. The hardware that is available and will be used comprises the personal machines of the team members, all of which satisfy the minimum hardware requirements to support and run the software applications required for development and testing. All the finalized testing and testing requiring heavy configurations will be managed and once again performed by the Tech lead of the project as the specialized hardware peripherals are available to the tech lead.
   2. All the team members are using the same software (Visual studio) for development and management of the tests, and resources like Git and Jira are being used for version control and progress tracking. All this software and resources have already been made familiar to the team. No additional/external software is intended to be used throughout the project to minimize potential confusion and collaboration compromises that may occur from using different resources and tools.
2. **Execution Strategy**

**a. Criteria:**

i. The program must be functional.

ii. Any failed tests must have a Low or below severity level.

iii. Output outcomes must match projected values (iii).

**b. Severity Levels.**

i. Critical - Program crashes that prevent the program from compiling or memory leaks that result in an unanticipated crash.

ii. high - Program bugs or flaws that can impair or cripple functionality and lead to numerous failures or inaccurate results.

iii. medium - Output bugs or errors that result in inaccurate or erroneous results.

iv. Low - Smaller warning errors or defects that don't interfere with the program's functionality.

v. Cosmetic changes to the code that require rewriting or removing unnecessary variables.

**c. Test Reporting:**

i. Tests will be carried out with each iteration of code added to the repository.

ii. All group members will carry out tests to make sure the values provided by other members are correct.

iii. Group members will review the test findings to see if they contain any discrepancies.

**d. Quality Control**

Members of quality assurance will give developers advice on how to proceed to ensure that flaws or errors are found correctly for potential corrections and bug fixes.

1. **Test Schedule**
   1. **This is the section where you write layout a schedule for the testing and be able to give an estimate of how long the testing will take and approximately when it will be complete.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Schedule** | **Test** | **Description** | **Time spent for Completion** |
| **July 15, 2023** | **Unit Testing** | **Testing individual functions** | **4-6 hrs** |
| **July 20, 2023** | **Black Box Testing** | **Testing the program by comparing outputs given the test inputs** | **2-5 hrs** |
| **July 25, 2023** | **Integration Testing** | **Testing the program as a whole** | **3-5 hrs** |

1. **Control Procedures**

**6.1 Reviews**

-To guarantee accuracy, completeness, clarity, and adherence to project objectives, reviews of function specifications, design papers, and test plans are crucial.

- Various project team members, including developers, testers, and project managers, should take part in these reviews.  
**6.2 Bug Review Meetings**  
-Identified bugs and defects should be discussed and prioritized during regular bug review meetings.

-Project managers, members of the development team, and members of the quality assurance team ought to attend these meetings.

- Each bug should be evaluated for severity and importance before developers are assigned to address it.

**6.3 Change Request**

- Create a clear procedure for dealing with change requests during the testing stage.

- Clearly state the requirements for documentation and the standards for accepting or rejecting change requests.

- Assign members of the project team distinct duties for reviewing and carrying out modification requests.

**6.4 Defect Reporting**

- Establish a standardized reporting format for errors found during testing.

- Detailed information such as reproducibility steps, predicted and actual outcomes, and the severity level assigned should all be included in defect reports.

- Create a reliable system for monitoring and recording the correction of reported issues.

1. **Functions To Be Tested**

In order to select the best truck for delivering products depending on their weight, size, and location, this software testing assignment simulates the operations of a delivery service. The software does calculations to choose the best truck and decide whether any detours from the truck's route are required after receiving data regarding the weight, box size, and destination of each shipment.

The offered sample output illustrates the functionality of the program by displaying various shipment scenarios together with the appropriate truck assignments and detours.

1. **Resources and Responsibilities**  
   **8.1. Resources**

- Assure that each team member has access to laptops or PCs that are set up with the required hardware and software.

To enable thorough testing, test data and sample inputs should be made available.

- Team members should have access to the project's source code repository and pertinent materials.

- To promote efficient teamwork, communication and collaboration tools should be made available.  
**8.2. Responsibilities**

- Clearly outline each team member's job and responsibility as it relates to the testing process.

- Assign tasks and specify what each team member is expected to contribute.

- To promote effective coordination and collaboration, specify the precise duties of developers, testers, project managers, and quality assurance team members.

1. **Deliverables**

**- Test plan**

**- Test cases**

**-Test scripts for automated testing**

**- Data used for the tests**

**- test logs and reports**

**- errors/defects found**

**- what was done to resolve them**

**- test summary**

1. **Suspension / Exit Criteria**
   1. **If an encountered error/bug prevents further progress of affects the entire testing environment**
   2. **Any dependency constraint, such as network connectivity problems, hardware failures or unavailability of sample testing data.**
   3. **If the desired stability of tests is reached.**
   4. **If the suspension or exit of the project is approved by both the leads.**
2. **Resumption Criteria**
   1. **A working solution for the encountered bug is developed.**
   2. If the dependency constraints have been resolved by alternative connections/hardware or troubleshooting.
   3. If the leads approve the Resumption of the project.
3. **Dependencies**  
    12.1 Personnel Dependencies

**a. Project lead and tech lead to oversee the testing, coordinate the testing and ensuring the testers have the skills, and subject-matter expertise to execute the tests in adherence to the test plan.**

12.2 Software Dependencies

a. Version, configuration, and updates of the software being used for testing  
12.3 Hardware Dependencies

a. Database management and network, and connectivity requirements  
12.3 Test Data & Database

**a. Tests to be developed and performed by all team-members. The list of all tests and entire sample data is recorded and stored by the Project Lead.**

1. **Risks**  
   13.1. Schedule

a. Scheduling risks involving personal dependencies are to be reported to leads prior to the planned meeting or deadline (whichever is closer).

13.2. Technical

a. Any compatibility risks involving different versions of the same software or tools are to be resolved by ensuring the software versions and coordinating on any further updates or changes.

13.3. Management

a. Managements risks involving issues or problems caused by the poor decisions or improper task assignment by the project or the tech lead resulting in inefficiency or hindrance to the project are to be resolved immediately on a priority basis among the team (or by including the Professor if need arises).

13.4. Personal  
 a. Any potential knowledge or skill gap is to be eliminated before starting the testing or development, additional help from the team should be sought otherwise.

b. Compromises on personal health should be strictly avoided. Any fatigue, burnout or physical illness is to be discussed with the team.

13.5 Requirements

a. Any incomplete or ambiguous requirement guidelines are to be discussed and resolved at the time of receiving.

b. Any prioritization issues should be tracked and resolved by creating them as a new issue on Jira and assigning it to the leads.

1. **Tools**
   1. **Git/GitHub**
   2. **Visual Studio debugger**
   3. **Jira**
   4. **Visual Studio automated tester**
2. **Documentation**

**Test Plans**

**Test Logs**

**Test summary report**

**Error/bug report**

1. **Approvals**

**TEST PLAN**

Test Plan approval – All team members

**TEST CASES**

All the test cases prepared are to be approved by the Tech Lead.

**ERROR RESOLUTION**

If there are changes in the source code while resolving bugs, then the changes to the source code will be approved by the project manager.