Test Plan

1. **Introduction**
   1. Test Plan Objectives
      1. The project we are testing is a delivery system that has different trucks that deliver on three different routes in the city. We are testing to ensurethe precise allocation of packages to delivery trucks. The three trucks represented by blue, yellow and green lines start from the same point and travel through the route, we have to figure out the shortest posssible path from the three truck routes to the destination. We expect to get the most optimised path from the route to the destination which also takes into consideration any building obstacles on the way. We also have to take into account the truck’s capacity and package’s sizes are not exceeding the maximum limit.
2. **Scope**
   1. **In this section you can describe what will be tested and what will not be tested.**

**We will test everything to ensure that it is the most efficient solution to the problem which involves testing the following scopes –**

**(i)We will test that the truck and the point of destination selected is the shortest possible distance by implementing a valid algorithm.**

**(ii)We will test to ensure that the truck’s capacity is not exceeding the maximum limit.**

**(iii)We will ensure that the size of the boxes is within limits.**

**(iv)Ensuring to implement a valid delivery mechanism that promises on time delivery.**

**(v)We have to ensure that the route chosen successfully avoids the obstacles.**

1. **Test Strategy**

**Testing is performed to ensure that the program is delivered in a seamlessly working state to the client.**

**1. Sytem test – In order to ensure its functionality, we will run the whole program and ensure**

**that the results produced by all the components are as expected. This covers testing both**

**the functional and non-functional code of the application.**

**2.Performance test – In order to implement this we will build test cases that will involve**

**scenarios in which the efficiency of the code is tested. The speed, responsiveness and**

**overall quality of the code are put under scrutiny.**

**3.Security Test : In this case, we will check that no person can take unauthorized access into our application. In the case of the milestone, we are delivering our app to a delivering company so it is important to ensure that no one can login without a username and password and also there should be measures against breaking into the system.**

**4.Automated Testing : In order to check that the code is bug-free, we can test all the functions by unit testing in visual studio. This way it will be easy to test a large number of test cases by simply changing the values and it makes the process faster and more efficient.**

**5. Stress and Volume Test : If there are more orders than the system can handle then there should be some measures to avoid crashing of the app. In order to ensure this, we will put our application under high loads of user entry and test the limits to which we can increase our load handling.**

**6. Documentation Test : It is imporatant to ensure that the project requirements are being met and in order to ensure that, we can maintain documentation. Another way to ensure successful completion is creating test cases for documentation purposes and collaborating with other team members.**

**7. User Acceptance Test : In the end, we are building the app to be used by the user in order to fulfill its business requirements so it is important to ensure that the end-user is satisfied with the result of the application.**

**Test Design Process Overview :**

**1. Understanding Requirements : We need to understand the result that is expected out of project before starting out with it. Identifying the parts which need thorough testing in order to implement the solution effectively.**

**2. Building Traceability Matrix : Tracebility matrix allows us to perform forward traceabilty where we look up the requirements of the project and find the related tests as well as reverse traceability where we can look up a test and see the associated requirements that it needs to fulfill. It is essential in large projects in order to ensure that software has been implemented to fulfill every requirement listed by the client and also the test cases cover the aspects of the application that require testing.**

**3. Prepare Test Cases : The test case ensure that we cover all the aspects of the business requirements hence delivering a end-product that meets the requirements of the client. Having the test case laid out ensures that the testing team can have access to them at any time and check them again once the code is modified to ensure that they still hold true.**

**4. Peer Review : Group collaboration is the key to the success of this project. Actively taking part in each other’s work and helping each other out would lead to good teamwork and hence a well tested and implemented application. After completing a few tasks, it is important to hold reveiws on each otherr’s work so that we can stay upto task and collaborate with other members of the team better.**

1. **Environment Requirements**

**Hardware** - Dedicating one laptop for all testing requirements. In the group, we have 5 members so we can easily allocate the testing requiremnts to one person and ensure that their workstation is upto the task. Another important task is to ensure backup of the test data. If we lose our test data, it would be difficult to modify or work with the application in the future, so it is essential that we have a good backup method for all the testing scenarios.

**Software** – The test data needs to be tracked in order for us to see the changes made by all the group members. For this we have softwares like jira that will be used to assign tasks and for code tracking purposes. For the actual testing part, we will be using Visual Studio Debugger and writing the test outputs in a test cases document which will be an excel file. Other than this we will be working with github and pushing our code onto the repository so that everyone can collaborate effectively and learn to work with this professional model of code handling.

**Test Harness** – This invlove the process of having a testing framework at hand to be able to work with various conditions of the program. We should have test data that tests for all the edge cases and covers all the aspects of the business requirement. Also once the test data is created, one has to ensure that it is saved and stored safely for future use. Another great way to ensure that our test data is saved and accessible at any is version control. It allows us to maintain mutiple version of the software. This enables the recovery from mistakes, if someone accidently caused a bug in the original code and that resulted in the creation of many files, now we would not know which files to delete and which to use, this when version control comes into play. It simplifies our process by marking down all the files in a particular release. Remote version control keeps a copy of the version software on a network based server as well as on your local computer as well. To implement this in our project we will be using Github in order to keep track of our code and in case any mistake is made, we will be able to recover from that effectively without any problem.

1. **Execution Strategy**
   1. **Entry Criteria** : One should be able to start testing if the program begins execution. This can be ensured if the build of the program has succeded and is available for debugging to begin. The test environment which includes hardware and software components should be ready for execution for the testing to be executed smoothly. The test data should be ready for checking as the testing cannot be started without deciding on the test data. The documents to record the test data should also be made available as it is important to keep track of all the test data and document it accurately for future reference.

**Exit Criteria** : We can stop testing when we reach 100% in the main functions like finding the best possible route and truck capacity but other functions like order details and package size can be approved even if it has 95% test success. It is important that the test cases pass all the important edge like if there is a obstacle then that has to be avoided and the route has to change accordingly to ensure that the application delivers its results successfully.

* 1. You can describe the severity of defects in this section and break them down into severity levels of:
     1. **critical** which cause the system to crash or produce anomalous results,

**-**Functional Defects : They do not let the user perform the task that the application was built for. This renders the application useless and also results in customer disatisfaction.

**-**Data Corruption : If there is loss of data due to some bug in the program that can lead to severe problems as protection of user data holds the utmost importance.

**-**The application should perform with the same efficiency across all platforms regardless of the web browsers as providing a good user experrience is essential in testing.

* + 1. **high** which causes lack of program functionality and might have a work around,

**-**Incorrect Logic : Implementing the correct logic in order to calculate the shortest distance for the delvering company is of the utmost importance as they will be able to efficiently perform their task and mange their orders better if they save time on their delivering route. It is important to ensure that the route is free of any building obstacles.

**-**Unreliable Error Handling : If there is some misinformation or mismatched information being entered in the system, it shooud be able to produce the correct error messages in order to guide the user in a better way so that they can navigate their way across the application in a better way.

**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

**-**Misalignment with the requirements : It is important to the consider the fact that all aspects of the program should align with the business requirements of the client. If the application is working fine but still has some parts missing or logic incomplete then the requirements of the client remain incomplete and the purpose is not solved.

* + 1. **Low** which might be an unclear error message or some other minor error that has minimum impact on functionality

**-**Lack of accessibility compliance : All the users regardless of whether they are disable would be given the same access across the application.

**-**Inadequate Documentation : The data that we collect is for our own reference and future use to handle any updates in the program but it does not become a barrier in the performance of the application.

* + 1. **Cosmetic** which is something that makes the user interface less than optimal but still perfectly functional.

**-**User Experience : If the user is not able to use the application with minimum effort then that is a reason that can lead to the application being rendered invalid or useless

* 1. **Test Reporting**

(i)In our group, we have decided that the activity of each individual will be monitored through jira. Tickets will be assigned to each individual and they would have to work through the task assigned to them well in time so that other team members can review the work and make any changes necessary. Jira platform will act as a update and catchup place for all the testers as well, we can easily stay updated on the work that our other team members are performing. Asking for help can also be done on jira as everyone would try to stay as active on that platform as possible. The code would be pushed and commited to github after the individual is satisfied with the implementation This structure helps everyone in check and also provides reports on each others work. Other than this, all the test data will be wriiten in a excel sheet and a scrum report will be submitted in the end giving a detailed report on the contributions of each individual in the group.

(ii) When a tester finds a bug in the program, that has to be updated on jira, making it a issue that is of the type bug. Then a new ticket can be assigned to the person who has to fix the bug. This way everyone will stay in loop as to what the bug was and how th eother person is working to resolve that bug.

d. The developers and testers can interact by commenting on the jira issues. They can stay in touch with each other on that platform. Along with that, only the bug-free code should be pushed and committed into the git repository which is the responsibilty of each and every individual. A code with bugs would lead to further implementation on the same which would lead to critical errors and failure of execution so a clear communication is required.

**6. Test Schedule**

**Analysing and implementing code: 2 days**

**Creating whitebox and black box test cases: 1 day**

**Testing : 1 day**

**Fixing bugs : 2 days**

**Finalising all the management and resouces : 1 day**

**Completion : In approximately 6-7 days Control Procedures**

1. **Control Procedures**

**-Reveiws : Keeping regular reviews to ensure all the requirements are met and testing is completed within expected due timing.**

**-Bug Review Meetings:** Bug review meeting needs to be mandatory and regular so that software is bug free and testing members know what to fix and what more to implement

**-Change Request : It will be done to fix the bugs found and recover from it.**

**-Defect Reporting : Reporting by all teammates with proper documentation is necessary to keep track of progress.**

**8. Functions To Be Tested**

a) Finding accurate distance between points Functional testing : This is one of the key aspect of the software and finding accurate distance is necessary part of bulding this software.

Therefore, testing software to ensure this functional requirement is met efficiently.

b) Finding best route Functional testing : Testing to check if the software suggests the best and most efficient route out of all the possible paths. This testing will consider different start points, critical conditions and edge cases.

c) Finding best shipment allocation Functional testing : Testing to ensure that best truck is alloted to the package according to size, weight , path , destination and deliverable date. This testing will consider all valid and invalid test cases , all scenaiors, different sizes and destination, truck with maximum weight and situations involving trucks with varying capacity constraints.

d) Ensuring obstacle path is elimanted Functional testing : Testing to check that path with obstacles is elimanted out of the possible routes and software handles all the scenarios. In this testing will be done by considering multiple obstacles, path with shortest distance as well as obstacles and other exceptional cases

1. **Resources and Responsibilities**  
   - **Resources** : Resources we are grateful to have are code provided by college, access to github.senecacollege, Jira ( for management ) , clear and easy to follow guidelines and a dedicated team to conduct testing and bring out the best possible software for the problem.  
   - **Responsibilities :** The team comprise of a team leader : to manage all the tasks, to keep track of the progress, and to ensure all the responsiblities are met before deadline, Tester : to test the software and fix bugs, Coders : to code the rest part of the software as well as fix the bugs found by testers. Rest all the task are brought to completion with team efforts and hardwork.
2. **Deliverables**

- An analysis of the problem (no written artifacts produced).

- A series of data structures created as header files and stored in the repository.

- A test plan stored in the repository.

- Completed scrum report including reflection questions answered.

1. **Suspension / Exit Criteria**

**-The truck can only hold upto 1500 kilograms of cargo.**

**-The truck is capable of holding 48 cubic metres of boxes.**

**-If the closest route calculated is incorrect**

**-If the error messages are incorrect.**

1. **Resumption Criteria**

**After a error has been detected, it has to be resolved and made sure that there are no further bugs that will cause the program to terminate. It is important to perform testing again on the program in order to ensure that all the changes have been implemented harmoniously in the system.**

* 1. **Dependencies**  
     12.1 Personnel Dependencies : Precisely define the roles and responsibilities of each team member to ensure the seamless execution of their assigned tasks.  
     12.2 Software Dependencies : Given that each team member employs distinct software for development and testing, it is vital to maintain data accuracy and system stability while ensuring the accurate testing of the code.  
     12.3 Hardware Dependencies : The availability of hardware, such as computers, is indispensable for the successful execution of testing and development activities, thus ensuring the project's smooth operation.  
     12.3 Test Data & Database : : It is of utmost importance to ensure that the test data is suitable for testing purposes and appropriately stored within the database. This encompasses data related to weight, volume, and routes, with a particular focus on maintaining data integrity and verifying that performance aligns with the expected standards.

1. **Risks**  
   13.1. Schedule : It is important that each member communicates with each other in order to decide the timings of group discussion and share the updates on the piece of code they are working on currently in order to ensure a smooth execution of the program.  
   13.2. Technical : The technical challenges can mainly occur due to network issues but our group will ensure that all the work is performed well in advance. Other than this, all the repositories and methods of submission will be well established for each student so that no one has any problem in submitting their work.  
   13.3. Management : It is important that the group leader assigns specific tasks to each individual so that they have a sense of direction as to what they have to contribute to the project and complete their deliverable in time.  
   13.4. Personnel : We have enough members in our group to complete the tasks of testing successfully but still it is important to assign equal amount of tasks to everyone so that one person is not more burdened than others.  
   13.5 Requirements : Having a clear understanding of what is expected out of the task to be performed is a must before taking on any responsibility.
2. **Tools**

**1.Testing the program with Visual Studio Debugger**

**2.Testing the shorter functions using printf statements**

**3.Testing the program maintaining log files**

**4. Tracking and maintaining group discussions using jira**

**5.Creating versions of program using Github.**

1. **Documentation**

**It is important to maintain a full record of all the test data that will be implemented in the program for the successful execution of the output. This record keeping helps us in the future when we have to understand a particular bug in the program or introduce modification in the program. It also serves as a communication tool between team members, including testers and developers ensuring that they are upto date on the tasks. This improves the overall test strategies to a large extent as it serves as a map for the testing team as to what has been executed successfully and what can be improved.**

1. **Approvals**

It is essential that each member of the group has equal say in the project strategies and is satisfied with the end product. Before the submission of each milestone, every member has to approve of the final draft of the project in order for it to be considered a valid submission by the group.