Final Test Report

|  |  |  |  |
| --- | --- | --- | --- |
| **Function/acceptance/requirement** | **Test Run** | **Bugs Fixed** | **Passed** |
| AddRoute | TF001 |  | Yes |
| PopulateMap | TF002 |  | Yes |
| TestEqPt | TF003 |  | Yes |
| DestinationNotInBuilding | TF004 |  | Yes |
| DestinationIsInBuilding | TF005 |  | Yes |
| DestinationOutOfBounds | TF006 |  | Yes |
| DestinationOnBoundaryNotBuilding | TF007 | Corrected building detection logic | Yes |
| DestinationOutOfBounds | TF006 |  | Yes |
| DestinationOnBoundaryNotBuilding | TF007 |  | Yes |
| DestinationOnBoundaryIsBuilding | TF008 | Fixed boundary conditions for building check | Yes |
| AllPositivePaths | TF009 |  | Yes |
| OneNegativePath | TF010 |  | Yes |
| BuildingNotInRoute | TF011 |  | Yes |
| EmptyRoute | TF012 |  | Yes |
| SingleBuildingPoint | TF013 |  | Yes |
| RandomValues | TF014 |  | Yes |
| SameNotBuildingPoint | TF015 | It was not checking if outside the map (not returning 0) | Yes |
| SameBuildingPoint | TF016 |  | Yes |
| validtruckPaths | TF017 |  | Yes |
| testTruckWeightCapacity | TF018 |  | Yes |
| validDestination | TF019 |  | Yes |
| validShipmentWeight | TF020 |  | Yes |
| validBoxSize | TF021 |  | Yes |
| shortestRouteOptimization | TF022 |  | Yes |
| validTruckVolume | TF023 |  | Yes |
| Checking Destination and Building | TF024 |  | Yes |
| Checking Destination and Building | TF025 | Couldn’t fix bug | No |
| Checking Destination and Building | TF026 |  | Yes |
| Checking Destination and Building | TF027 | Couldn’t fix bug | No |
| validTruckPath | TF028 |  | Yes |
| isDestinationBuilding | TF029 |  | Yes |
| testValidTruckPath | TF030 |  | Yes |

**Approximate Cost of Testing and Project Expenses**

1. **Wages of Testers:**
   * **Senior Tester:** $50/hour x 40 hours = $2,000
   * **Junior Tester:** $30/hour x 40 hours = $1,200
   * **Total Wages:** $3,200
2. **Cost of Equipment:**
   * **New Testing Servers:** $5,000
   * **Software Licenses:** $2,000
   * **Miscellaneous Hardware Upgrades:** $1,500
   * **Total Equipment Costs:** $8,500
3. **Development Tools and Resources:**
   * **IDE Licenses (e.g., Visual Studio):** $1,000
   * **Project Management Tools (e.g., Jira, GitHub):** $500
   * **Total Development Costs:** $1,500
4. **Total Cost of Testing and Project:**

* **Wages:** $3,200
* **Equipment:** $8,500
* **Development Tools:** $1,500

**Grand Total:** **$13,200**

**What Worked Well**

1. **Basic Functionality**: The core functionalities, such as adding routes, populating the map, and performing basic equality checks on points, were implemented successfully. These functionalities passed their respective unit tests, indicating that the foundational aspects of the application were solid. The TestAddRoute and TestPopulateMap functions, for example, performed as expected, adding routes to the map and populating the map with correct values.
2. **Destination and Building Checks**: The checks for whether a shipment’s destination is within a building or if it is within the bounds of the map worked as intended. The DestinationNotInBuilding, DestinationIsInBuilding, and DestinationOutOfBounds tests demonstrated that the application could accurately determine whether a shipment's destination was valid. This was crucial for ensuring that trucks did not attempt to deliver packages to invalid or inaccessible locations.
3. **Path Validity and Optimization**: The functions responsible for validating truck paths and optimizing routes also performed well in most scenarios. The validtruckPaths and shortestRouteOptimization functions passed their tests, indicating that the application could correctly determine the shortest path for deliveries and validate that the paths chosen were valid. This optimization is key to reducing delivery times and improving overall efficiency.
4. **Handling of Capacity Limits**: The application successfully managed the constraints of truck capacities in terms of both weight and volume. The testTruckWeightCapacity and validTruckVolume tests confirmed that the application could correctly determine whether a truck could accommodate a shipment based on its current load. This ensured that the trucks were not overloaded, preventing potential issues with delivery failures due to overcapacity.

**What Didn’t Work**

1. **Checking Destination with valid Building:** Two integration test cases passed for this but the other two didn’t pass. We had troubles fixing and debugging the bugs. Therefore, the main interface for the destination input didn’t work.
2. **Route Display Issues**: The program failed to display valid truck options and their corresponding shortest routes to the destination. This prevents users from selecting the best possible route for their shipment.
3. **Route Deviation Handling**: The program did not adequately handle or display route deviations, especially concerning the shortest route to the destination. This lack of functionality unabled evaluate potential diversions along the chosen route.