**Requirements**

R1: drone can deliver pizzas to the customer.

R2: drone can reject invalid orders.

R3: all the URL should be safe and valid before use.

R4: all flight paths shall not enter the no-fly-zones.

R5: route generated should be short, better to be the best route.

R6: processing time for algorithm should be as less as possible with maximum of 1 minute.

**Basic Plan**

Steps for all the tests:

* Apply system test to test R1.
* Apply unit test to test R2.
* Apply security test to test R3.
* Apply unit test to test R4.
* Apply performance test to test R5.
* Apply performance test to test R6.

Firstly, when the program is basically built, we need to do the system test to check if the program can run properly which means drone can deliver pizzas and provide a record with all the delivery made. Then, we need to add more functionalities, unit test will be used now to check if the drone can reject all the invalid orders and security test is able to make sure the given URL is valid and safe. After that, when we add the first edition of the algorithm into the software, a unit test will be needed to check if the route generated now can avoid all the no-fly-zones. If it passes the test, we can further improve the algorithm. After each edition of the algorithm implemented in the software, we need to use performance test to check is the algorithm provided any better routes than before and if the routes are good enough. Finally, when we finish the algorithm, another performance test will check if the processing time is short and keep improving the running time of the algorithm.

**Potential Vulnerabilities**

In order to check if the system rejects the invalid orders (R2), we need to know that the order is invalid, however, I need to do the validation check of the orders in my program as well, so if my validation is wrong then even the system rejects all the invalid orders correctly, those orders may not be invalid in the first place.

For R4, We only have one set of no-fly-zone data to use, so if there are other unexpected shape of the no-fly-zones, drone may not be able to avoid them.

**Scaffolding and Instrumentation**

R1: we need at least a week of data to test if drone can deliver the orders now.

R2: we should view the order files and pick up or write some invalid orders for testing

R3: consider some valid but wrong URL which can’t lead to the right file

R4: find the no-fly-zone file and paste it into the geojson in order to see where they are, also need the flight path output file in different days.

R5: print out the steps taken by the route to digitally see the improvements, also input the flight path output file into the map website to visually see the improvements.

R6: need to set a place in the output file to record the processing time or print them during tests.