| | | | |
|---|---|---|---|
| **Test Scenario Number:** | 005 | **Tested By:** | Ryan Bomalaski |
| **Sprint Number:** | 1 | **Application:** | main.py |
| **Tracker ID:** | ST-005 | **Time Estimation:** | 30 Minutes |
| **Module:** | N/A | **Type:** | Stepwise |

**Test Scenario and Requirements Description:** Tester will run test script test_005.sh to test Scenario 5.

**Prerequisites:**

- User has Collision Avoidance folder
- User has SQLite3 Installed

**Scenario Title:** Run Simulator for 120 Steps with resolution of 10 steps per second to simulate 12 seconds of flight time.

**Scenario Procedure:**

Using the provided scripts, the user will import the test airplanes to the python algorithm. Then the user will run the simulator for 120 steps.

| Scenario Steps: | Validation: |
|---|---|
| **Create Airplane Test Database:**<br>1. Open New Terminal<br>2. Navigate to …/collision_avoidance/test_scripts<br>3. Run command:<br>    1. ./test_005.sh | SQLite will initialize with test attributes. The terminal will open the python terminal (Denoted with the ">>>").*<br><br>* - Note: If this is the first set up of the table, two errors will appear. |
| **Create Simulator object and populate with Airplanes**:<br>1. Create a new simulator object with step count of 120 by typing the following command:<br>    1. sim = Simulator(120,10)<br>2. Populate the simulator with aircraft by running:<br>    1. sim.create_airplanes()<br>3. Confirm that two airplanes were created by running:<br>    1. sim.airplanes | A list of two airplane objects with the address in memory will appear. |
| **Run Simulator**:<br>1. In python environment, run the following command:<br>    1. sim.run_sim()<br>2. When the simulator is complete, run:<br>    1. exit() | The simulator will step through 120 steps, giving outputs for both airplanes. Upon exit, the user will be back at the linux terminal. |