4 Testing

Some of the unique testing challenges in our project include the complexity of testing communications between frontend and backend software as well as testing drone attributes in real-world conditions. Below is our testing plan to handle these challenges.

4.1 Unit Testing

As shown in the system architecture figure, there are several distinct modules in our system. While some of them are servicing different contexts, in each case, there are certain identifiable units of development and implementation that are planned to be a part of a final deliverable.

We will now list a few specific examples of identifiable elements:

1. Database

- Insertion of new "maps" i.e., models of geographical regions of interest. We expect that the database will be able to allow multiple grid models to be stored to avoid specification from scratch.
- Retrieval of the grid from the database. The database will be able to send stored grids to the desktop application to be used in a simulation.

2. Desktop Application

- Parameters within the desktop application such as the number of drones and the sizes of cells within the grids will be adjustable. For scalability, adding new parameters will not break any existing functionality.
- The flight planning algorithms should not impact the functionality of each other, even when new algorithms are implemented.

3. Drones Testing

- Numerical data to test the usage and life of a battery. Using a multimeter to get accurate reading of the battery usage in different situations like roaming and full speed of the drone.

4.2 Interface Testing

There are several identifiable levels of interface testing:

1. The interface used for planning and observing drone movement
    a. Testing various page formats and map interactivity…
2. The mobile interface for observing results and sending requests
3. Testing and making sure that the algorithms we are intending to use are optimized and efficient enough for a fleet of drones to work together while preserving energy expenditure.
4. We will also need to make sure that the data that is being used will be able to be used by the simulation to visualize a properly working drone surveillance system.

4.3 Integration Testing

There are several critical use cases that must be tested within our project including those listed below.

1. If the user inputs a predefined test area, the simulation will pull the test area from the database.
    a. The simulation and database will need to communicate with one another, this will be tested by confirming the two can successfully communicate with each other and information that is pulled from the DB can be seen on the simulation.
2. Communication between the database and the mobile application
    a. The requests for information can be tested with Volley acting as a faux mobile app for the purposes of simulating different requests and object types.

4.4 System Testing

We respectfully note that through integration tests 1 and 2, we will have achieved overall system testing. We will have captured units from different modules that need to be interfaced as well as module-wise integration towards enabling the functionality of the system.

4.5 Regression Testing

New additions should not impair the original functionality of the simulation, but instead should allow the user to have more options when using the simulation to get desired results. We need to ensure that the algorithms that we are using do not get affected in terms of effectiveness with additions of new features.

4.6 Acceptance Testing

As mentioned in section 4.4 (and 4.3), many of the functional requirements will be tested and evaluated as part of the integration and system testing. For example, for pulling a predefined testing area onto the simulation, the software will test for a successful communication with the backend and if the data was successfully pulled from the server. The client will be involved in the acceptance testing in both a functional and nonfunctional requirements sense. For example, the client will provide different activity scenarios in the geographical region of interest monitored by the drones.

4.7 Security Testing (if applicable)

While we recognize the importance of security protocols, this is not within the scope of this project. However, we will rely on existing mechanisms e.g., password protected accesses, basic encrypted communication protocols, etc.

4.8 Results

We will not have any quantifiable results this semester. In the next semester, we will follow Agile development.