1.1 - Range of Requirements

Functional Requirements

1. Drones should not be able to fly over no-fly zones *(no fly zones are defined as rectangles in LngLat space)*. This especially includes corner cutting
2. The service should be stateless.
3. No medicine should be delivered with a cost exceeding the total cost associated with that MedDispatch.

$$Total\_Cost = Initial\_Cost + moves \cdot cost\_per\_move + Cost\_Final$$

    1. Note that costs are distributed *pro-rata*, meaning if there are two deliveries in a route with a total cost of 22, this should be treated as a cost of 11 per delivery.
4. Must be able to query drones based on their capabilities, and these queries must return correct results
5. Drones can only move with an angle that is a multiple of 22.5˚
6. Drones can only move by *exactly* 0.00015˚ in a given direction.
7. Must be able to query the service to retrieve a flight path given a list of MedDispatch
8. A drone flight should deliver all medDispatches assigned to it
    1. A "delivery" counts as a drone hovering (The same location twice in the flight path) within 0.00015˚ of the delivery location.
9. Ensure flight paths begin and end at the same service point.
10. Return 400 if invalid data is passed into an endpoint, otherwise return 200
11. Endpoints must be correctly exposed (the majority have api/v1/ in as a prefix)
12. System must retrieve external information from another REST service given by a url found in the ILP_ENDPOINT environment variable.

Measurable

1. No response for an endpoint should take more than 30 seconds
2. Good usability - endpoints shouldn't require too much info
3. Good availability - start time should not be long (< 10 seconds), and crashes should be rare or non-existent
4. System must have good scalability - technically able to be parallelised.

Qualitative

1. Code should be well structured/readable
2. Code should be robust (not crash due to invalid input or runtime errors)

3. Always return appropriate HTTP status codes (200 or 400).