Requirements

Non-Functional

Application should run in less than 60 seconds (Performance)
Application should exit gracefully when errors are found (Robustness)
System should run under different amounts of stress (Robustness)
Readable Code (Maintainability)

Functional

Orders must be validated before being processed (unit)
Drone Route must be valid (unit)

- Must not go through no fly zones
- Once the drone is back in the central area it can't leave
- After picking up food the drone must return to the Central Area in the quickest route possible
- Drone must hover at restaurant and Appleton when picking up/delivering the pizzas

Drone moves should always be the same length (unit)
Drone moves should be in the 16 compass directions (unit)
Drone routes should only be generated for valid orders (unit)
Orders must be processed in the order they were received (unit)
System must receive orders, restaurants, central area and no fly zones from an external Rest API (integration)
The application must generate three files when run (system)

- deliveries-date.json (every order with the order no, order status, order validation code, cost in pence after the system has been run)
- drone-date.geojson (list of coordinates traversed by the drone in GeoJson format)
- flightpath-data.json (array of JSON records with a record for each drone move)

Application must run from the command line and accept two inputs (system)

- Valid url
- yyyy-mm-dd date