1.1 Range of Requirements

Function Requirements

1. The system should receive orders via an external source and must validate customer orders based on validation rules, including:
    1. Credit card number is valid, Expiry date is valid, CVV is valid.
    2. The numbers of pizzas on the order do not exceed limit.
    3. The restaurant is opened on the order date.
    4. All pizzas on the order are from the same restaurant.
    5. All pizzas on the order are on the restaurant's menu.
    6. All pizza prices are correct.
    7. The total price should be the sum of each pizza's cost + a 100 pence delivery fee.
    8. Pizza order is not empty.
2. The system must be able to retrieve accurate restaurant data, including: name, location, menu, and opening days from an external API. The system must also retrieve coordinates for no-fly zone areas and the central area from an external API.
3. For valid orders, the system should be able to calculate the shortest most efficient flight path from the restaurant to Appleton Tower. Flight paths must be computed using straight-line segments of 0.00015 degrees, restricted to 16-point compass directions. The path must be valid, avoiding designated no-fly zones, and ensure that it does not exit the central area once entered, it must not exit. If the order is invalid, the system should not calculate a delivery path.
4. The system must be able convert the flight path into a valid GeoJSON structure which can be pasted in https://geojson.io/#map=2/0/20 and see the visual results.
5. The system must expose endpoints for order validation, path calculation, and health checks. The API must be reachable at http://server:8080/'request_name', and must not have any global prefix.

Performance Requirements and Measurable Attributes

1. Each REST API request should return a response within 500ms under normal load
2. The pathfinding algorithm should be optimized to calculate the most efficient valid route in under 200ms
3. The system must be able to handle at least 10 concurrent delivery requests without significant performance degradation, ensuring **scalability** under varying loads.
4. The drone should reach the destination within 0.00015 degrees accuracy.

Security Requirements

1. The system must reject improperly formatted order requests (input validation).
2. The system must not store or log raw credit card details for data privacy.
3. All endpoints should reject improperly formatted order requests, preventing the system from processing invalid or potentially harmful data.

Robustness Requirements

1. The system should handle external API failures.
2. The system must reject invalid GPS coordinates, such as extreme latitude or longitude values, to ensure navigational accuracy.
3. The system should be resilient to no-fly zone changes and be able to dynamically adapt if any changes to them occur.
4. An order should either be fully processed or rejected, preventing partial failures.