

Requirements

Functional Requirements:

1. Drone Operations:

- The drone must avoid all no-fly zones during its flight.
- The drone must deliver each order in the exact order provided by the REST API.
- The drone must pick up and deliver pizzas in under 60 seconds of runtime requirement.
- The drone must return to its starting location after completing deliveries.
- Only valid orders that have correct data and are within constraints must be delivered.
- The drone must hover at the pickup and delivery locations for one move.
- The drone must fly over buildings whenever possible to minimise risks.

2. Order Management:

- The system must validate all orders first before processing
 - valid credit card information.
 - correct pizza count, each order cannot exceed 4 pizzas but must contain at least 1.
- Orders with invalid data must be rejected.
 - closed restaurants
 - invalid payment details
- All pizzas in a single order must come from the same restaurant.
- Only current day deliveries are to be processed.

3. Routing and Pathfinding:

- The flightpath must avoid illegal routes, such as leaving and re-entering the central area.
- The system must optimise routes to minimise the number of moves.
- The pathfinding algorithm must use compass-based 16-directional constraints.

4. Output and Logging:

- The system must generate three output files for each session:
 - deliveries-YYYY-MM-DD.json containing delivery details and statuses.
 - flightpath-YYYY-MM-DD.json recording each move of the drone.
 - drone-YYYY-MM-DD.geojson visualising the flightpath.
- Each order must include the order number, status, validation code, and cost.

5. Error Handling:

- The system must handle invalid orders gracefully, without crashing.

- Errors in order validation must be flagged with appropriate error codes.
-

Non-Functional Requirements:

- The algorithm runtime must not exceed 60 seconds per delivery session.
- The drone must handle high volumes of daily orders efficiently.
- The drone must be available at all times.
- The system must ensure valid orders are always delivered accurately.
- There should be no delivery for invalid orders or during errors.
- The system must adhere to all safety and privacy regulations, including compliance with no-fly zones.
- All calculations for distances and paths must use double-precision floating-point numbers for precision.
- Sensitive order details, such as credit card numbers, must not be logged or stored inappropriately.
- The application must handle data securely and avoid exposing vulnerabilities during API interactions.
- The system must recover gracefully from REST API failures or incomplete data.
- Delivery sessions should proceed with partial data where possible.

Measurable Quality Attributes:

1. Performance:

- Algorithm runtime for computing the flightpath must not exceed 60 seconds per delivery session.

2. Accuracy:

- Flightpath calculations must be within a positional accuracy of 0.00015 degrees.

3. Reliability:

- The drone must avoid all no-fly zones with zero violations in a single delivery session.

4. Maintainability:

- Test coverage for unit tests must exceed 90%, to make sure the system is robust.

5. Compliance:

- 100% compliance with regulatory requirements, such as avoiding no-fly zones and privacy constraints.

6. Availability:

- The system must gracefully handle 100% of API errors, such as timeouts or invalid endpoints, without crashing.

7. Error Handling:

- All invalid orders must be flagged with an appropriate validation error code (e.g., INVALID, RESTAURANT_CLOSED).