Functional requirements:

* The system should be able to pick up orders from restaurants and drop it off at Appleton Tower by finding a path avoiding no-fly zones.
* The system should mark the orders as valid but not delivered if and only if there is nothing wrong with the order. I.e., credit card number, cvv, expiry date is valid, etc.
* The system should be able to have the drone deliver orders such that it maximises the number of orders delivered in a day.
* The system should be able to fly the drone in 16 compass directions.
* When the drone is picking up or dropping off the order, the system should have it hover for one move.
* The system should ensure that the drone only picks up one order at a time.
* The system should not send the drone out if the moves required for the trip is more than the drone’s battery level.
* The system should be able to generate a visibility graph of the provided restaurants, no-fly zones, and Appleton Tower.

Measurable attributes:

* Wait time
* Drone utilisation
* Execution time
* Drone availability
* MTBF

Non-functional requirements:

* The system should be secure and keep the users’ data private and encrypted.
* The pathfinding algorithm of the system should be able to plan and plot an optimal path for the day in less than 60 seconds.
* The system should operate all day; thus, the system should have 100% availability.
* The drone should not give off too much noise and greenhouse gas pollution
* The drone should ideally fly over buildings.