

## LO1 Support Documents (Requirements)

### 1.1 Range of requirements, a functional requirements, measurable quality attributes, qualitative requirements, ...

	Functional requirements	Measurable quality attributes	Qualitative requirements
Customers	<ol style="list-style-type: none"> <li>1. Log in and Log out</li> <li>2. Make orders</li> <li>3. Tracking orders in real time</li> <li>4. Cancel orders</li> <li>5. Refund</li> </ol>	<ol style="list-style-type: none"> <li>1. The orders should be delivered successfully.</li> <li>2. The orders need to be completed within the stipulated time.</li> <li>3. The user should be able to know how to use the application within a specific time.</li> </ol>	<ol style="list-style-type: none"> <li>1. Ease of use of the software (Nice looking User interface).</li> <li>2. User information can be kept securely, and personal information will not be disclosed.</li> </ol>
Restaurants	<ol style="list-style-type: none"> <li>1. Log in and Log out</li> <li>2. Receive orders</li> <li>3. Cancel orders</li> <li>4. Let restaurants update the menu and inventory information</li> </ol>	<ol style="list-style-type: none"> <li>1. The restaurant needs to know by what time to complete the order</li> <li>2. The user should be able to know how to use the application within a specific time.</li> </ol>	<ol style="list-style-type: none"> <li>1. Ease of use of the software (Nice looking User interface).</li> <li>2. User information can be kept securely, and personal information will not be disclosed.</li> </ol>
Delivery Company	<ol style="list-style-type: none"> <li>1. The drone route should be valid</li> <li>2. The drone should be able to go back to charging place when the battery is not enough to run all the orders.</li> </ol>	<ol style="list-style-type: none"> <li>1. Reliability: Mean Time Between Failure</li> <li>2. Performance: The drone should be able to find an optimal route to delivery certain number of orders before the battery runs out.</li> <li>3. Maintenance cost (drone maintenance, charging due to inefficient algorithm)</li> <li>4. Load size: The drone can carry objects of a certain weight</li> </ol>	<ol style="list-style-type: none"> <li>1. The user interface should be designed to be easy for the user to use.</li> <li>2. Excellent system security: Full encryption, password management</li> <li>3. A state machine that can intuitively see the status of the drone.</li> </ol>

Regulators	<ol style="list-style-type: none"> <li>1. The drone route should be valid</li> <li>2. Drone safety: the drone should be safe to collect, pass out the food.</li> <li>3. The ability to investigate and respond to accidents or incidents involving drones or community members</li> <li>4. The ability to share data with other regulators, authorities and agencies as necessary.</li> </ol>	<ol style="list-style-type: none"> <li>1. Flight restrictions: Regulators may require drones to fly below a certain altitude, stay within a specific airspace, or avoid flying over certain areas.</li> <li>2. Number of accidents or incidents involving drones or community members</li> <li>3. Number of complaints or reported issues related to the drone delivery system</li> <li>4. Compliance with existing regulations and laws regarding airspace, privacy, and security</li> </ol>	<ol style="list-style-type: none"> <li>1. Privacy of Residents nearby (No cameras when the drone is flying)</li> <li>2. User information security (No data leak)</li> <li>3. Safety and security: Regulators may require drones to have certain safety features, such as obstacle avoidance sensors, and to be equipped with technology to prevent unauthorized access or hacking.</li> </ol>
Neighbors and Communities	<ol style="list-style-type: none"> <li>1. Privacy protections for community members, such as obscuring sensitive information on packages</li> <li>2. Drones should guarantee very high safety and not crash and injure pedestrians</li> </ol>	<ol style="list-style-type: none"> <li>1. The number of drone crashes over a long period of time should be below a very small number</li> <li>2. Noise level of the drones, measured in decibels</li> <li>3. Number of community complaints or reported issues related to the drone delivery system</li> </ol>	<ol style="list-style-type: none"> <li>1. Safety and security for both the drones and the community members</li> <li>2. Privacy concerns for community members</li> <li>4. Transparency and communication with the community regarding the operation of the drones</li> </ol>

## 1.2 Level of requirements, system, integration, unit.

### System Test:

1. Log in and Log out for users (Customers and Restaurants)
2. Ease of use of the software (Nice looking User interface).
3. The orders need to be completed within the stipulated time
4. The user should be able to know how to use the application within a specific time.
5. User information can be kept securely, and personal information will not be disclosed.
6. Navigation: The drone delivery route cannot cross the no-fly zones.
7. The user interface should be designed to be easy for the user to use
8. The drone could delivery certain number of orders before the battery runs out.
9. Drone safety: the drone should be save to collect, pass out the food.
10. Excellent system security: Full encryption, password management
11. Reliability, Performance, maintenance
12. The number of crashes within a range of time and the drone noise pollution level

13. The ability to share data with other regulators, authorities and agencies as necessary.
14. The ability to investigate and respond to accidents or incidents involving drones or community members

**Integration Test:**

- Make orders, cancel orders, receive orders, update the menu and inventory information.
- Log in and Log out for users (Customers and Restaurants). (Interaction with database)
- The software should be able to give the estimation time of each order.