Drone Food Delivery Project

Team Members: Rishu Kumar, Mohammad Ahsan Nazmul,Phong Tran

Introduction

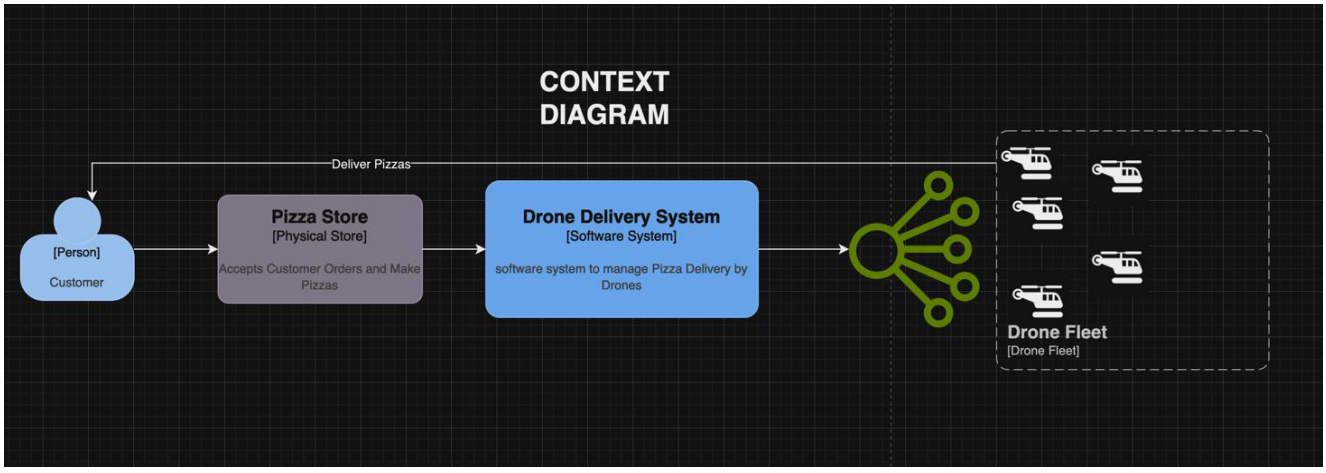
The Drone-based Food Delivery Service is designed to facilitate the automated delivery of pizzas using drones. This service consists of several components, including the Drone Control Center, Drone Device, Pizza Store, and Monitoring Service. The goal is to extend the existing system and implement a proof-of-concept (POC) using Spring Boot.

Features implemented and responsibilities

Component	Features implemented	Responsibility
Control Centre	Drone assignment CRUD operation on Drone devices	Manage overall drone operations
	Communication with Pizza store and Drone device	
Drone device(DroneUnit)	Execution of drone operations	Communicate with the Drone Control Center for assignment
Monitoring service	Drone monitoring	Collect data for tracking
Pizza Store	Communication with user, Control centre and drone unit.	Ordering and cancelling pizza.
Queue Service	Enqueue and Dequeue orders	Priorities orders
Drone Fleet Service	Request to Control Center	Handling Drone service request from Pizza Store

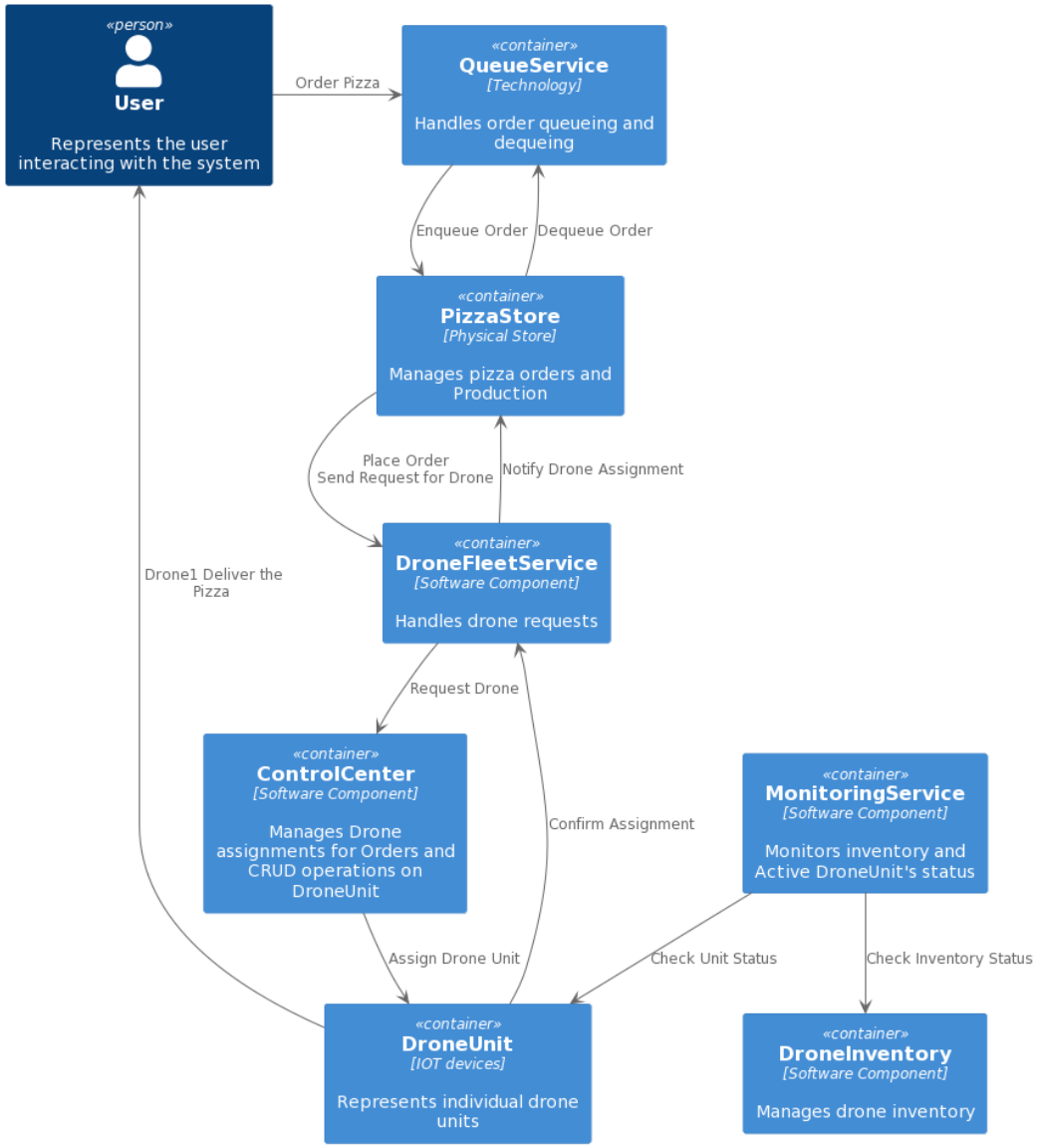
Context Diagram

This is the first layer of C4 model.



Container Diagram

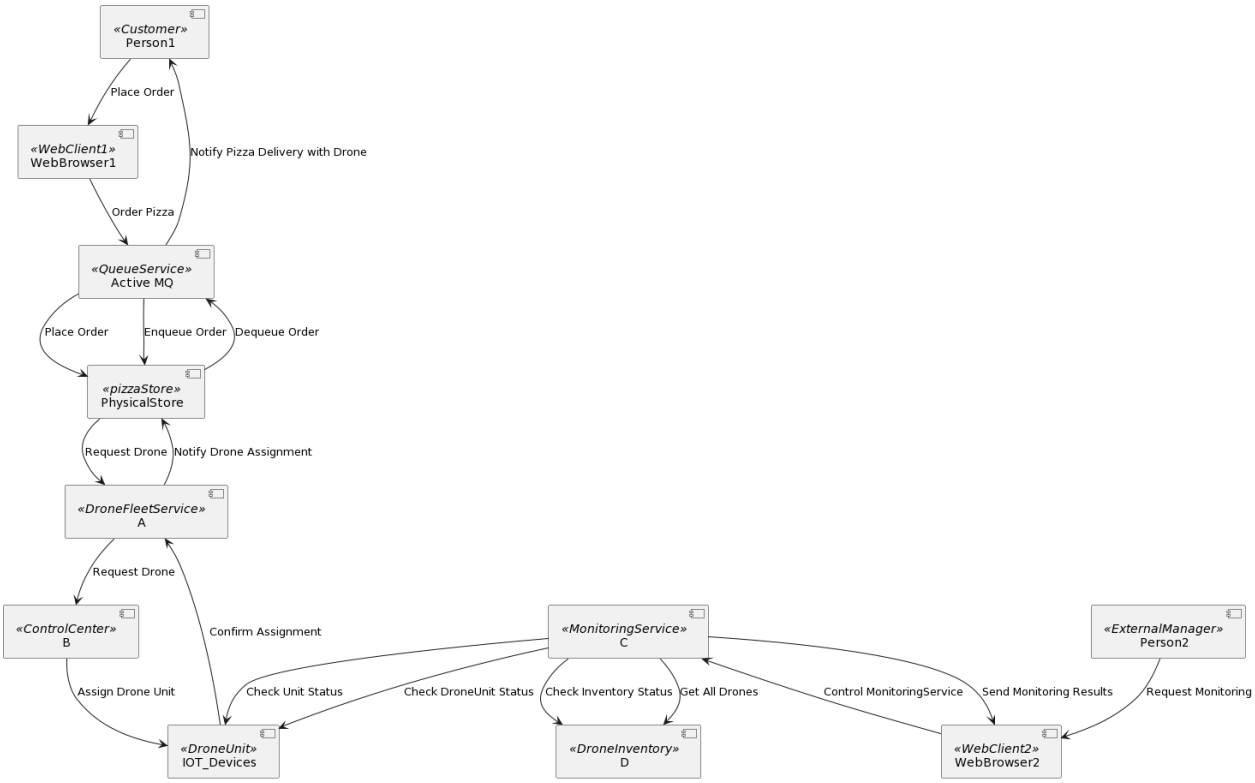
This is the 2nd layer of C4 model.

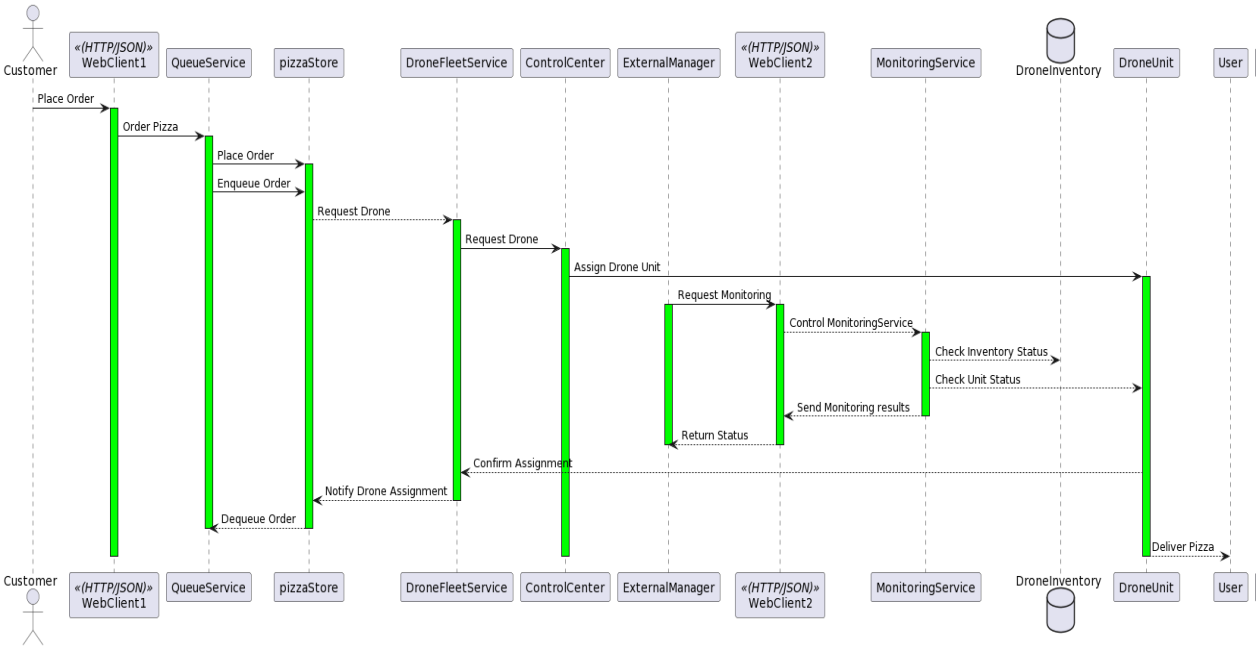


Component Diagrams

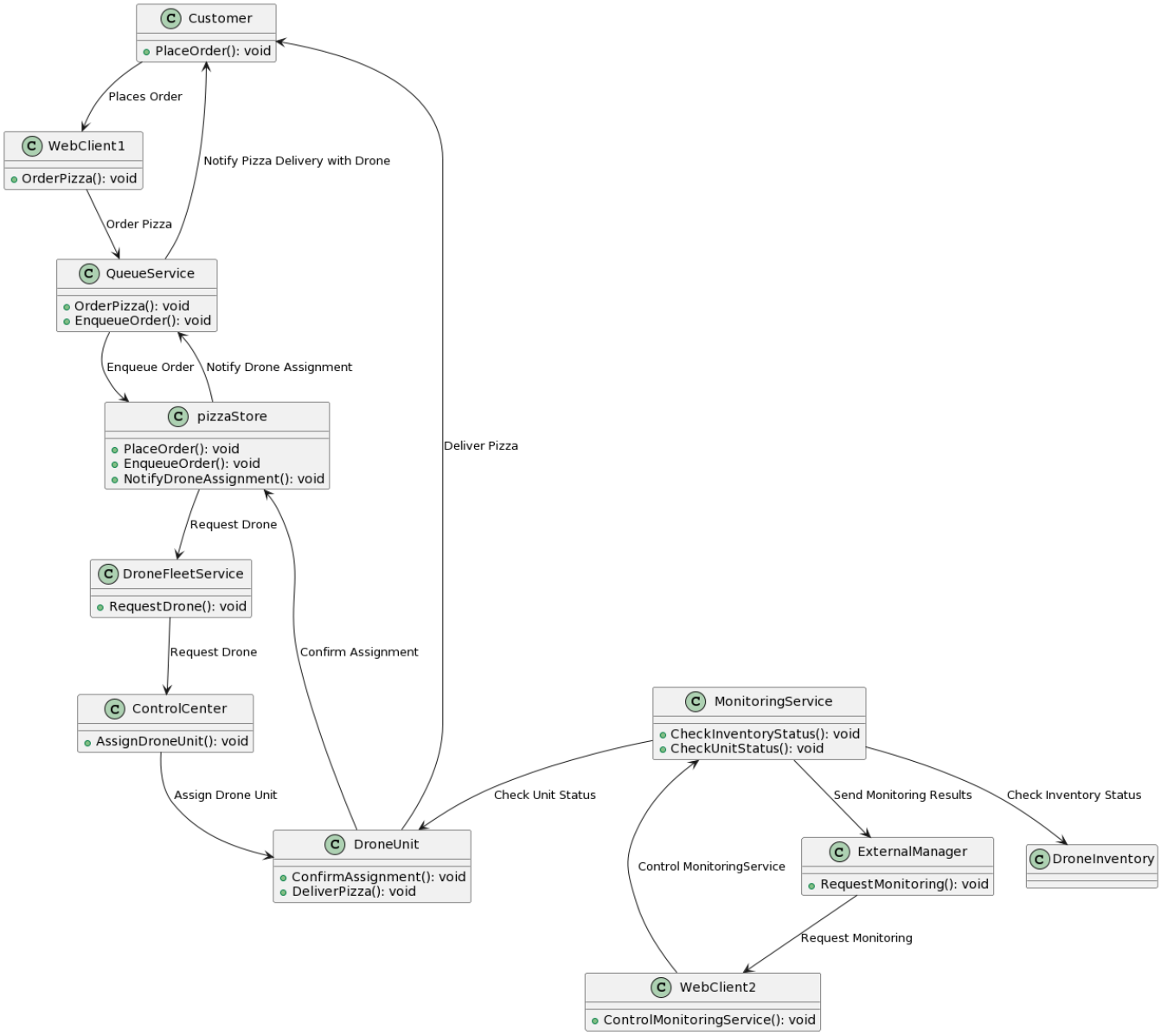
This is the 3rd layer of C4 model.

- **Components Interaction - Sequence diagram(s)**


«Customer»
Person1
Place Order
«WebClient1»
WebBrowser1
Notify Pizza Delivery with Drone
Order Pizza
«QueueService»
Active MQ
Place Order
Enqueue Order
Dequeue Order
«pizzaStore»
PhysicalStore
Request Drone
Notify Drone Assignment
«DroneFleetService»
A
Request Drone
«ControlCenter»
B
Confirm Assignment
«MonitoringService»
C
«ExternalManager»
Person2
Assign Drone Unit
Check Unit Status
Check DroneUnit Status
Check Inventory Status
Get All Drones
Control MonitoringService
Send Monitoring Results
Request Monitoring
«DroneUnit»
IOT_Devices
«DroneInventory»
D
«WebClient2»
WebBrowser2


Customer
«(HTTP/JSON)»
WebClient1
QueueService
pizzaStore
DroneFleetService
ControlCenter
ExternalManager
«(HTTP/JSON)»
WebClient2
MonitoringService
DroneInventory
DroneUnit
User
Place Order
Order Pizza
Place Order
Enqueue Order
Request Drone
Request Drone
Assign Drone Unit
Request Monitoring
Control MonitoringService
Check Inventory Status
Check Unit Status
Send Monitoring results
Return Status
Confirm Assignment
Notify Drone Assignment
Dequeue Order
Deliver Pizza

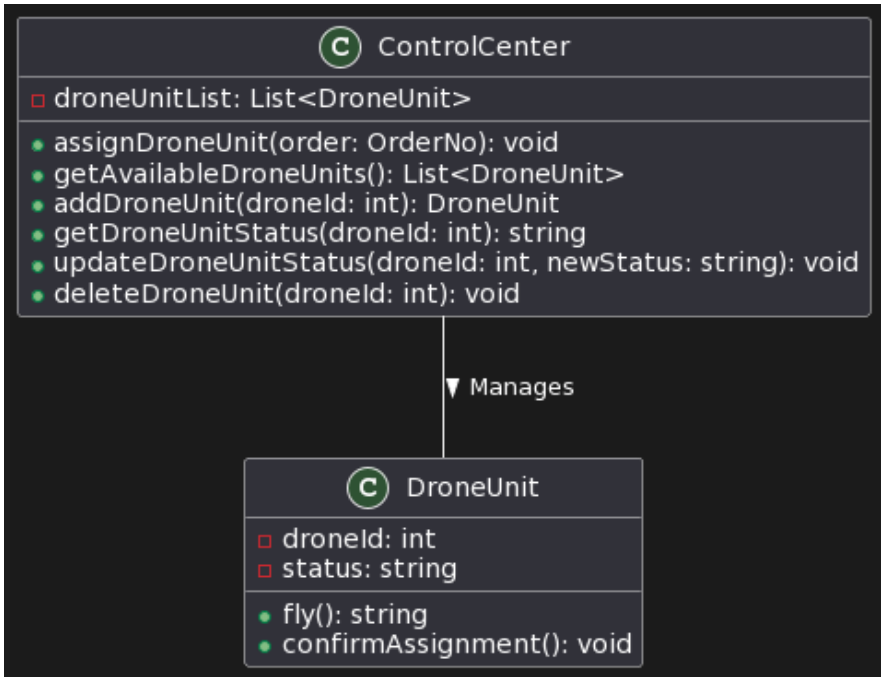
- **Components Interaction – Class diagram(s)**



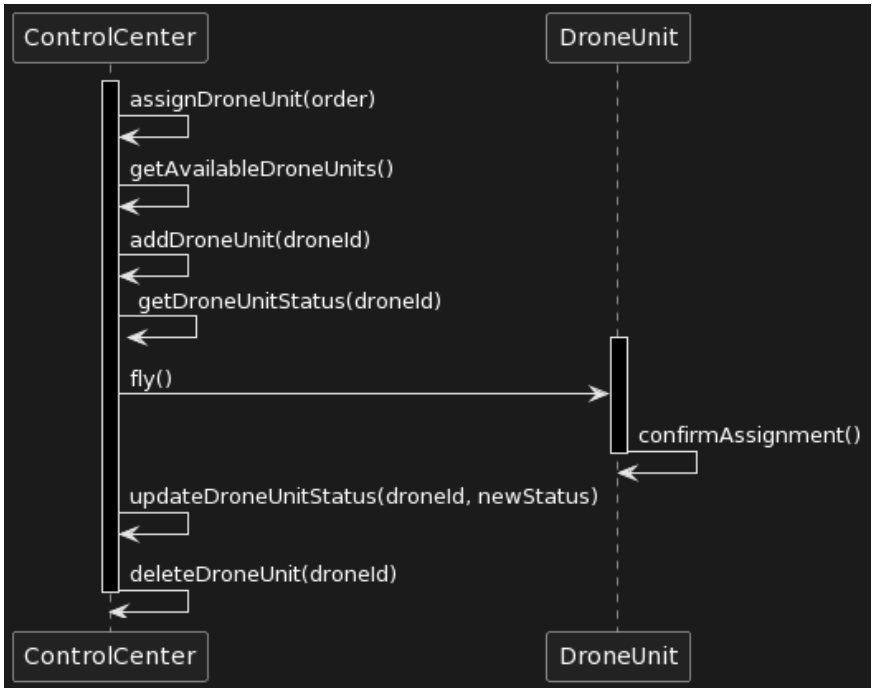
Code Diagram:

This is the 4th layer of C4 model.

- ***Code Diagrams for ControlCenter and DroneUnits components***



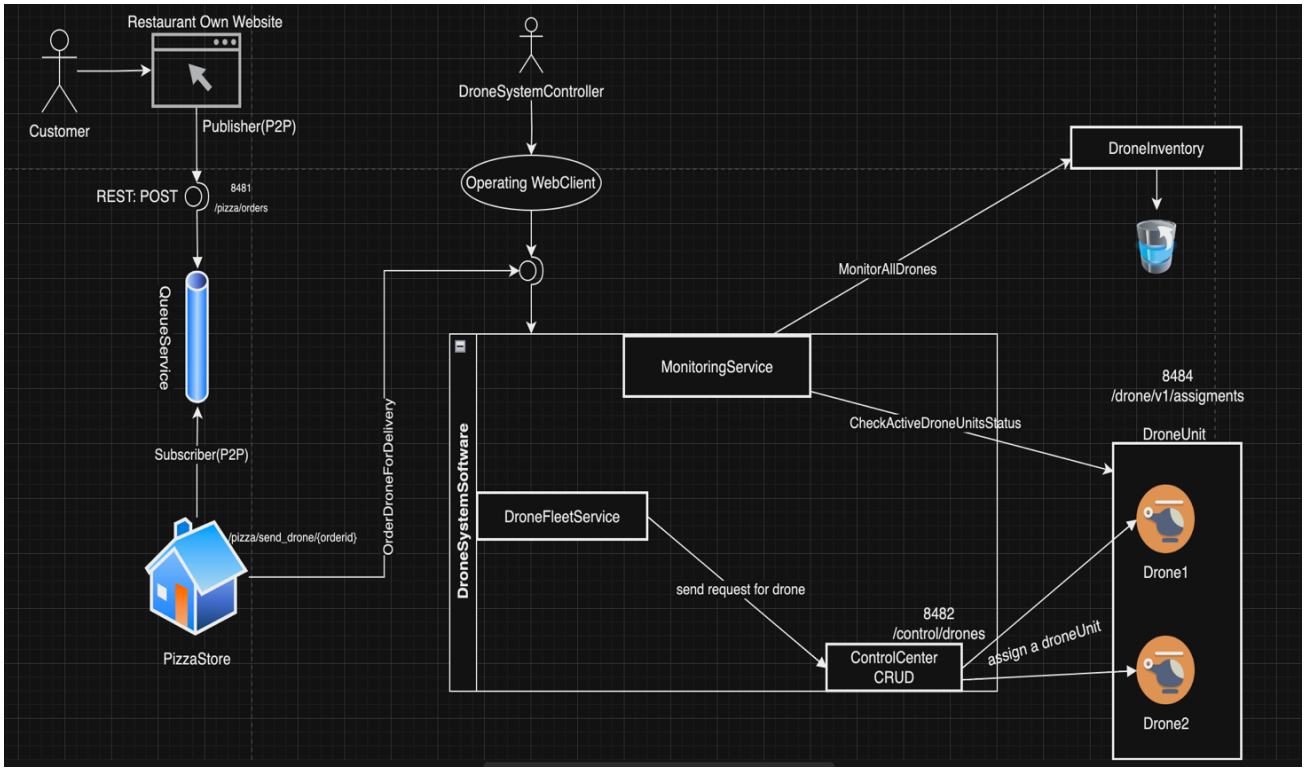
Class diagram



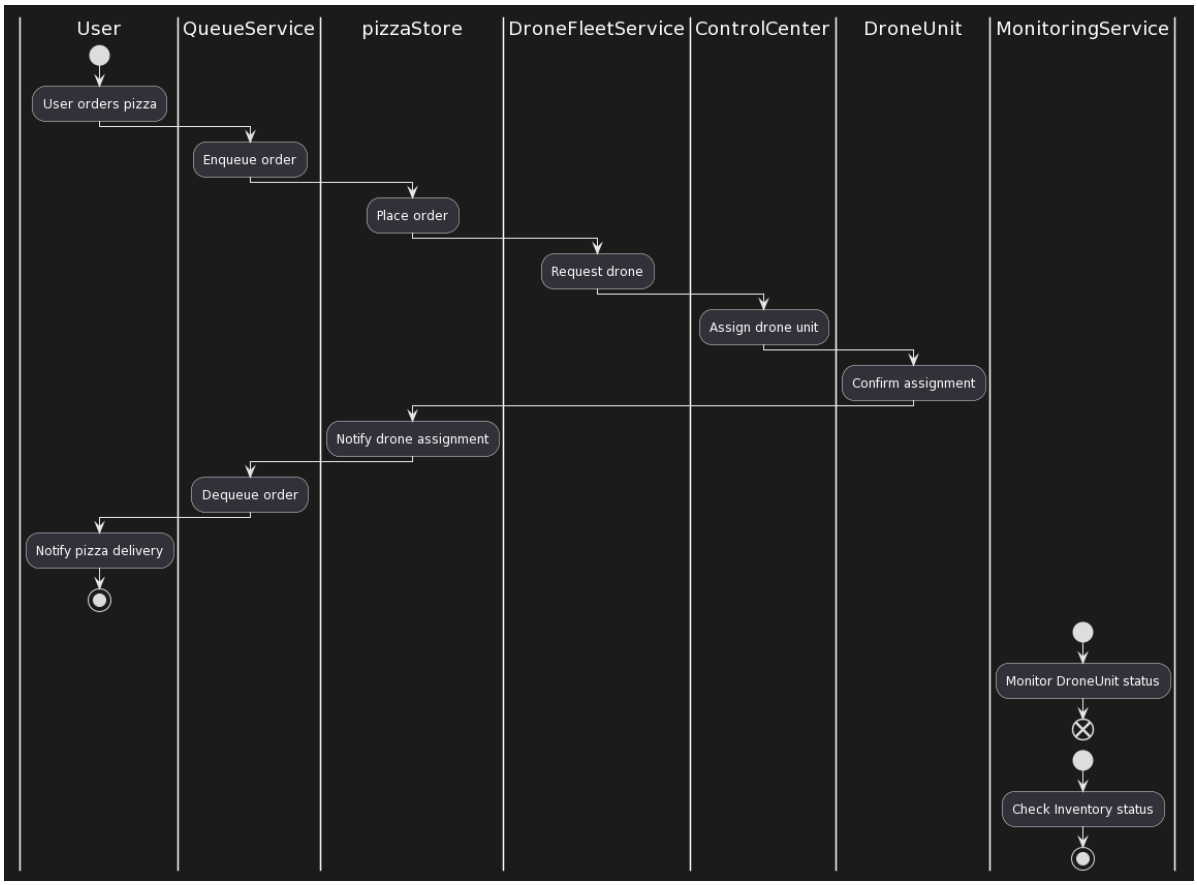
Sequence diagram

All Whole System Diagrams:

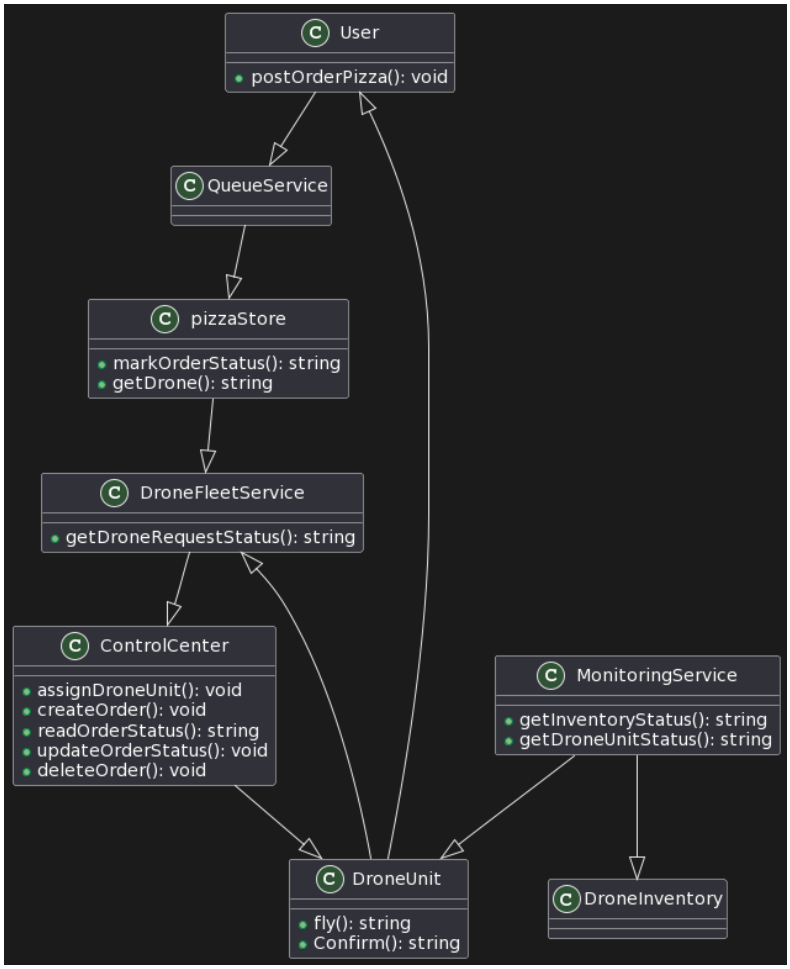
1. Overall Final Diagram



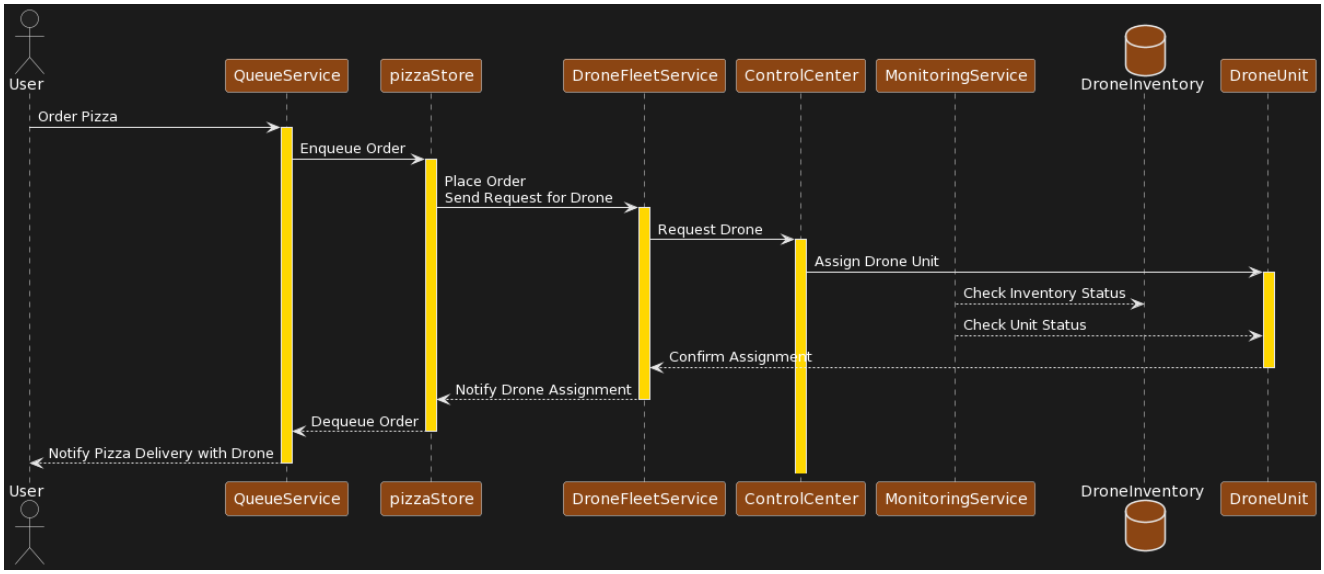
2. Activity Diagram



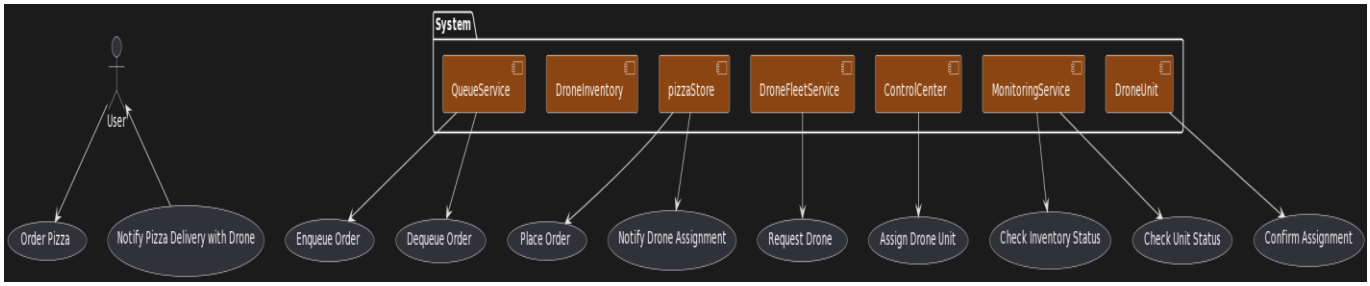
3. Class Diagram



4. Sequence Diagram



5. UseCase Diagram



Operating instructions

Installing, setting up and starting the containers

clone the code for the system

- git clone https://gitlab.tamk.cloud/sw-architectures-and-design-2023-rishu-kumar/final-sw-architecture-project-work.git
- Run it using spring boot

Testing instructions

Hands on commands to test the system with example requests. How to verify the result.

Console for hivemq: /opt/homebrew/opt/activemq/bin/activemq console

For Pizza store:

Place order: curl -X POST -H "Content-Type: application/json" -d '{"Order_id":"2", "Pizza_name":"Margherita", "size": "big"}' http://localhost:8481/pizza/orders

See order details: curl http://localhost:8481/pizza/orders

Cancel order: curl -X DELETE http://localhost:8481/pizza/orders/1

Message from control centre and drone unit: curl http://localhost:8481/pizza/send_drone/1

Control Center:

Assign drones: curl -X POST -H "Content-Type: application/json" -d '{"id":"1", "name":"droneverse", "capacity": 456}' http://localhost:8482/control/drones/

Drones details: curl http://localhost:8482/control/drones/

Update drone details: curl -X PUT -H "Content-Type: application/json" -d '{"id":"4", "name":"drone4", "capacity": 439}' http://localhost:8482/control/drones/

Remove drone: curl -X DELETE http://localhost:8482/control/drones/1

Fleet details : curl http://localhost:8312/fleet/v1/flights

Monitoring details : curl http://localhost:8312/fleet/v1/flights

Drone Unit:

Drone assignment: curl -X POST http://localhost:8484/drone/v1/assignments/ready

Drone details: curl : curl -X POST http://localhost:8484/drone/v1/assignments/ready