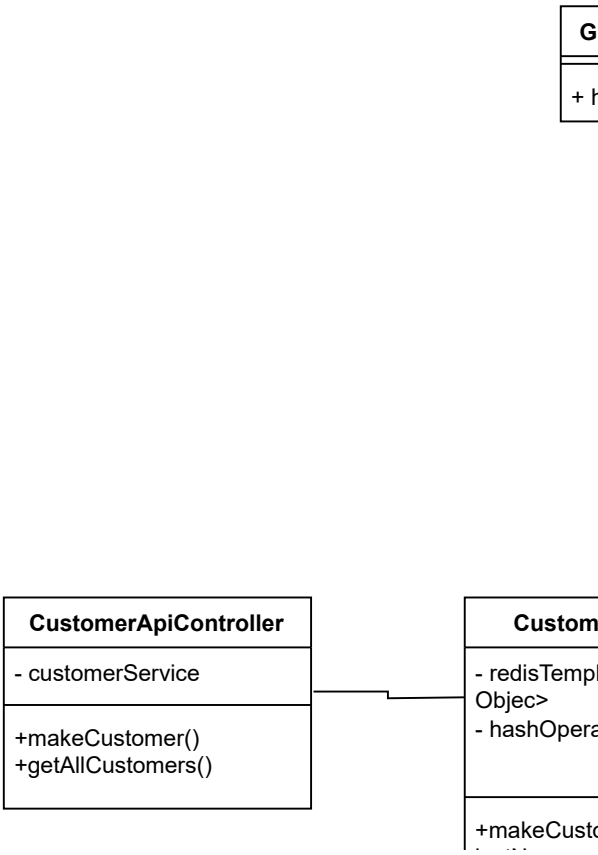
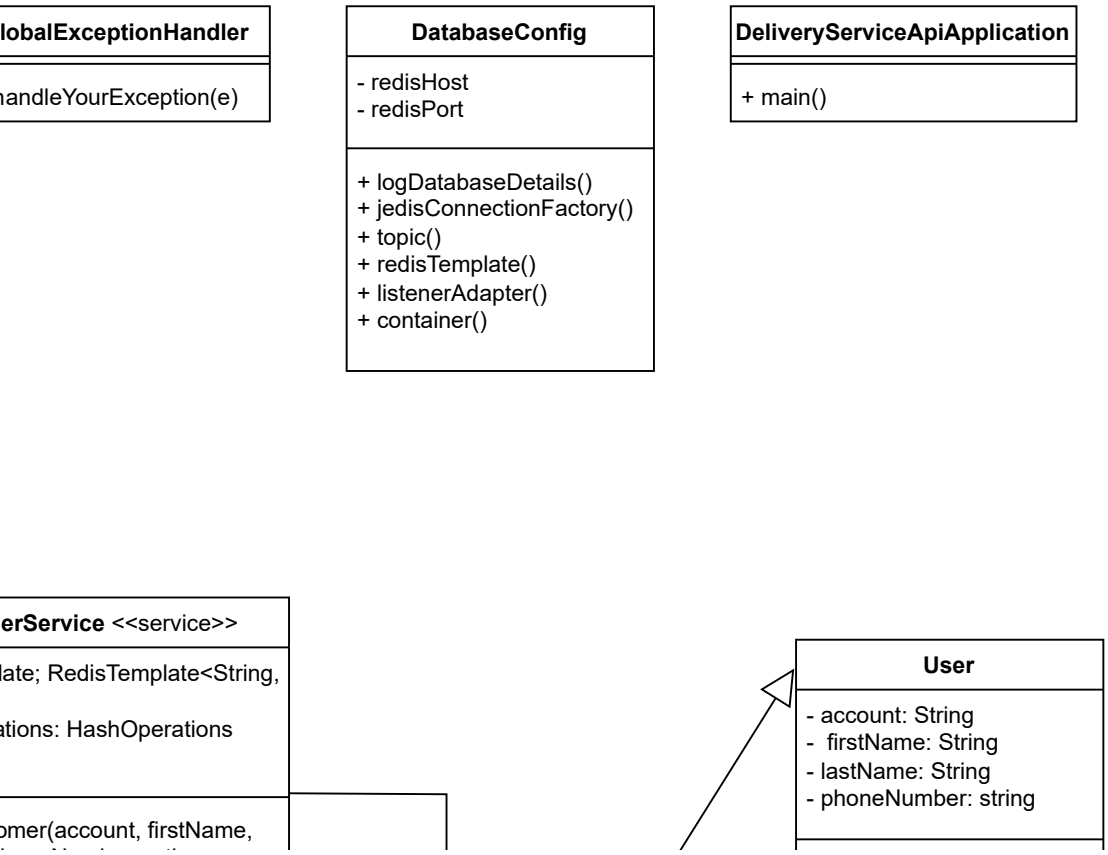
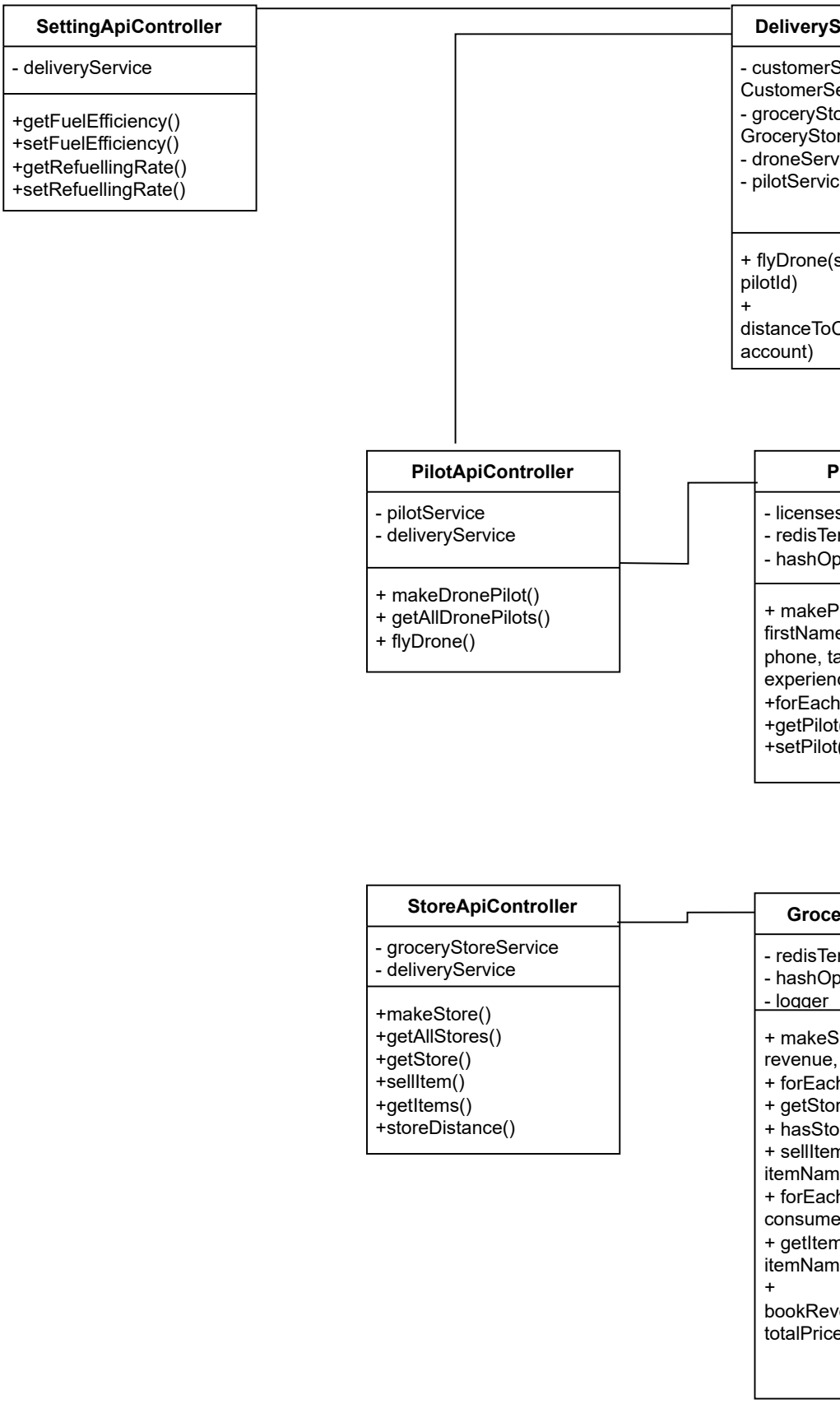
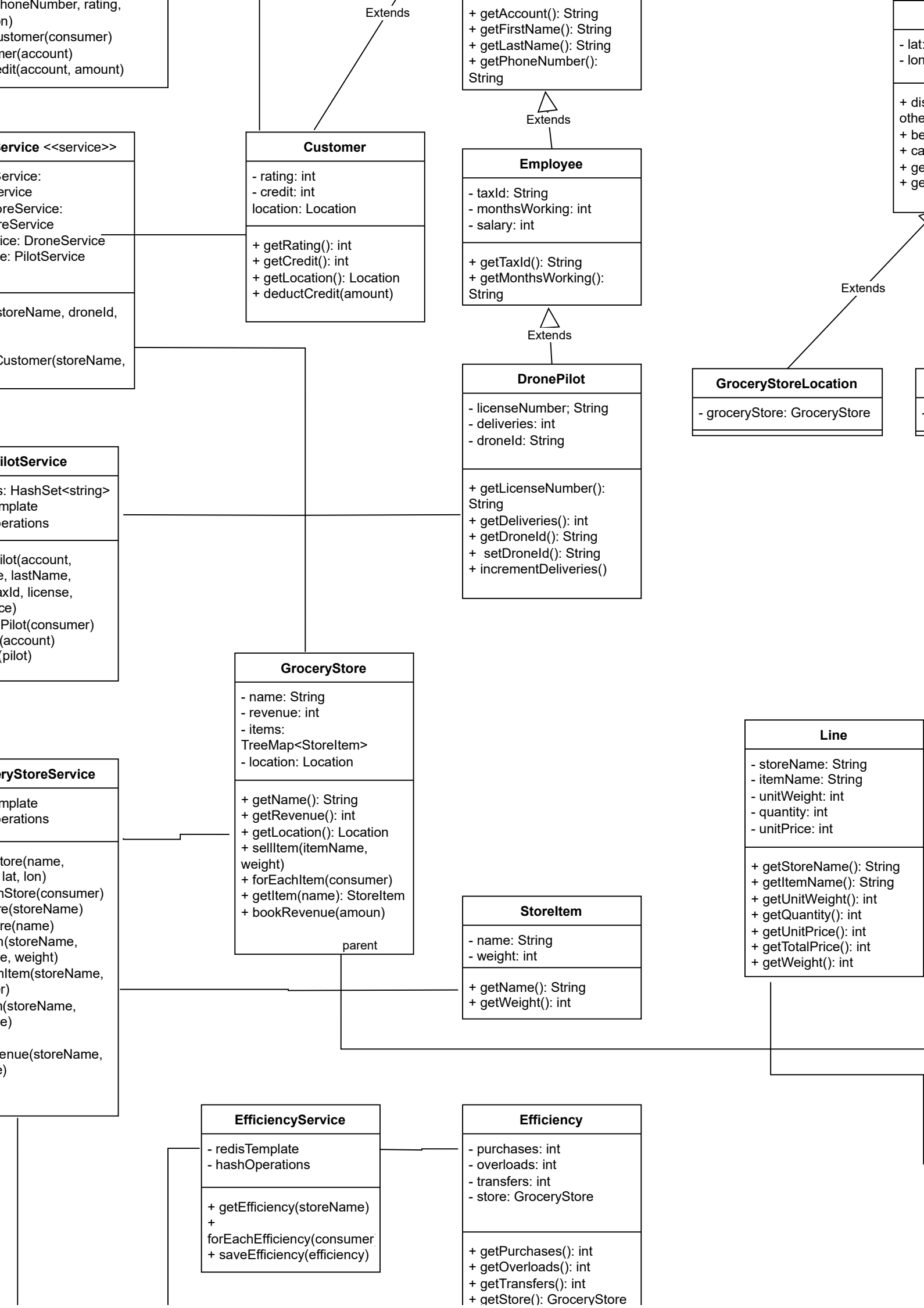
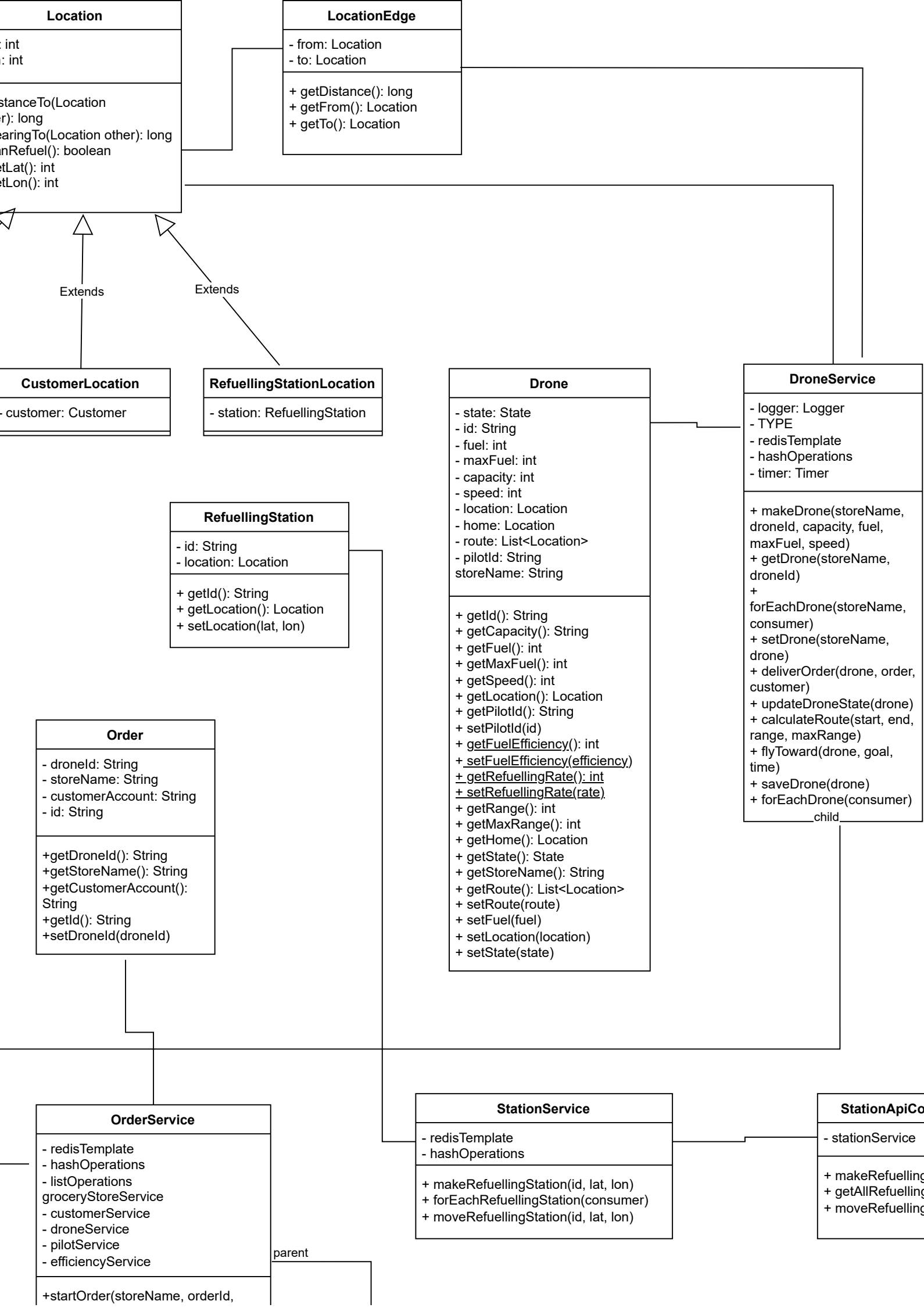
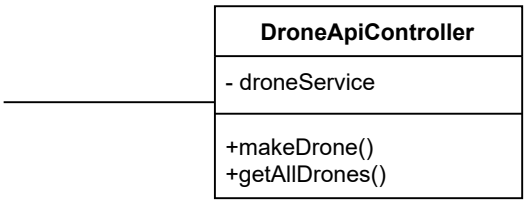

G
+ h
CustomerApiController
- customerService
+makeCustomer()
+getAllCustomers()
Custom
- redisTempl
Objec>
- hashOpera
+makeCusto

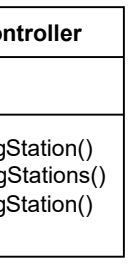

lobalExceptionHandler
handleYourException(e)
DatabaseConfig
- redisHost
- redisPort
+ logDatabaseDetails()
+ jedisConnectionFactory()
+ topic()
+ redisTemplate()
+ listenerAdapter()
+ container()
DeliveryServiceApiApplication
+ main()
erService <<service>>
late; RedisTemplate<String,
ations: HashOperations
omer(account, firstName,
User
- account: String
- firstName: String
- lastName: String
- phoneNumber: string

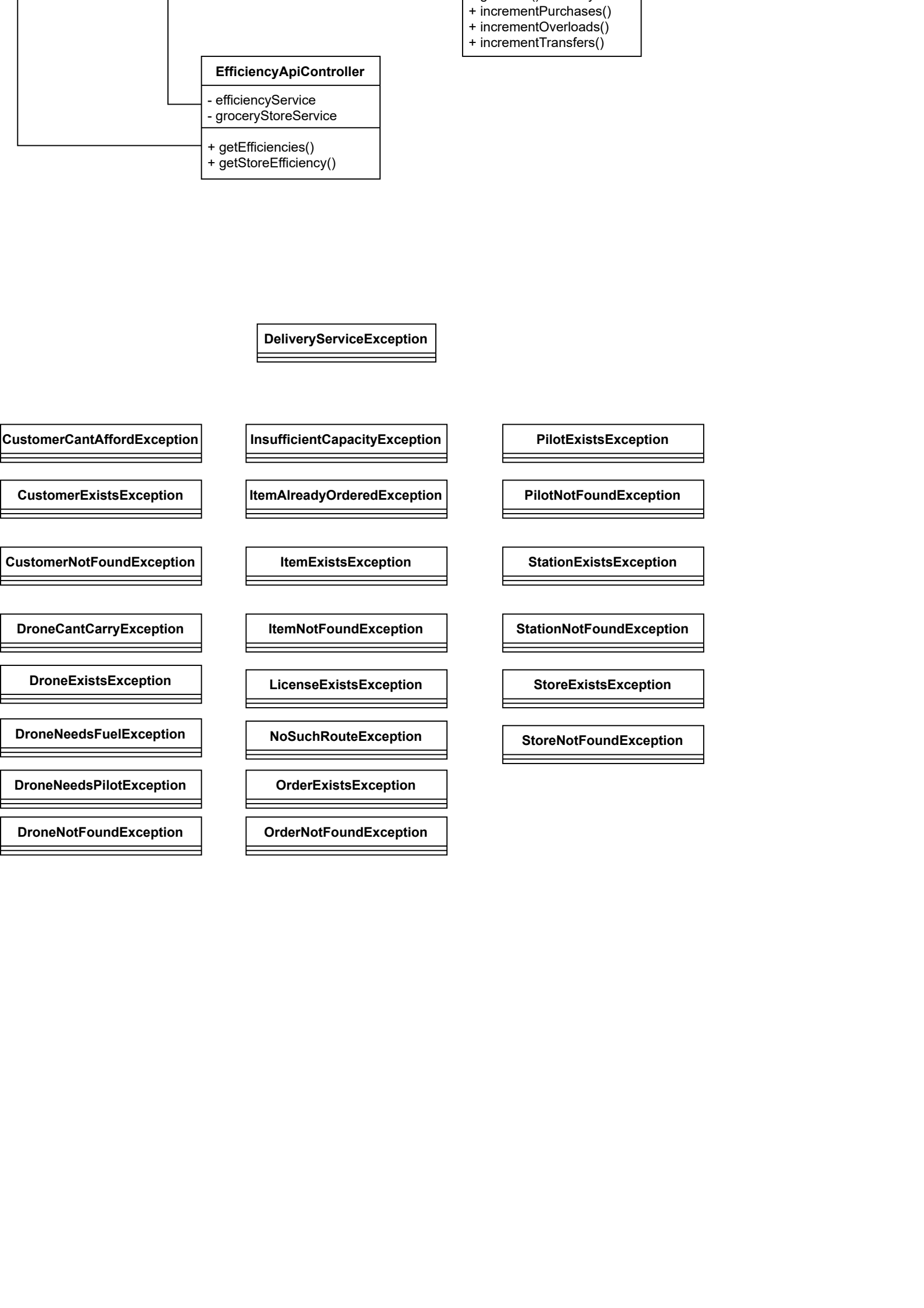

lastName, p
credit, lat, lo
+ forEachCu
+ getCustom
+ deductCre
SettingApiController
- deliveryService
+getFuelEfficiency()
+setFuelEfficiency()
+getRefuellingRate()
+setRefuellingRate()
DeliveryS
- customerS
CustomerSe
- grocerySto
GroceryStor
- droneServ
- pilotServic
+ flyDrone(s
pilotId)
+
distanceToC
account)
PilotApiController
- pilotService
- deliveryService
+ makeDronePilot()
+ getAllDronePilots()
+ flyDrone()
P
- licenses
- redisTer
- hashOp
+ makeP
firstName
phone, ta
experienc
+forEach
+getPilot
+setPilot
StoreApiController
- groceryStoreService
- deliveryService
+makeStore()
+getAllStores()
+getStore()
+sellItem()
+getItems()
+storeDistance()
Groce
- redisTer
- hashOp
- logger
+ makeS
revenue,
+ forEach
+ getStor
+ hasSto
+ sellItem
itemNam
+ forEach
consume
+ getItem
itemNam
+
bookRev
totalPrice


honeNumber, rating,
n)
ustomer(consumer)
mer(account)
edit(account, amount)
Extends
+ getAccount(): String
+ getFirstName(): String
+ getLastName(): String
+ getPhoneNumber():
String
Extends
- lat:
- lon
+ dis
othe
+ be
+ ca
+ ge
+ ge
ervice <<service>>
ervice:
ervice
oreService:
reService
ice: DroneService
e: PilotService
toreName, droneId,
Customer(storeName,
Customer
- rating: int
- credit: int
location: Location
+ getRating(): int
+ getCredit(): int
+ getLocation(): Location
+ deductCredit(amount)
Employee
- taxId: String
- monthsWorking: int
- salary: int
+ getTaxId(): String
+ getMonthsWorking():
String
Extends
Extends
DronePilot
- licenseNumber; String
- deliveries: int
- droneId: String
+ getLicenseNumber():
String
+ getDeliveries(): int
+ getDroneId(): String
+ setDroneId(): String
+ incrementDeliveries()
Extends
GroceryStoreLocation
- groceryStore: GroceryStore
ilotService
s: HashSet<string>
mplate
erations
ilot(account,
e, lastName,
axId, license,
ce)
Pilot(consumer)
(account)
(pilot)
GroceryStore
- name: String
- revenue: int
- items:
TreeMap<StoreItem>
- location: Location
+ getName(): String
+ getRevenue(): int
+ getLocation(): Location
+ sellItem(itemName,
weight)
+ forEachItem(consumer)
+ getItem(name): StoreItem
+ bookRevenue(amoun)
parent
Line
- storeName: String
- itemName: String
- unitWeight: int
- quantity: int
- unitPrice: int
+ getStoreName(): String
+ getItemName(): String
+ getUnitWeight(): int
+ getQuantity(): int
+ getUnitPrice(): int
+ getTotalPrice(): int
+ getWeight(): int
eryStoreService
mplate
erations
tore(name,
lat, lon)
nStore(consumer)
re(storeName)
re(name)
n(storeName,
e, weight)
nItem(storeName,
r)
n(storeName,
e)
enue(storeName,
e)
StoreItem
- name: String
- weight: int
+ getName(): String
+ getWeight(): int
EfficiencyService
- redisTemplate
- hashOperations
+ getEfficiency(storeName)
+
forEachEfficiency(consumer
+ saveEfficiency(efficiency)
Efficiency
- purchases: int
- overloads: int
- transfers: int
- store: GroceryStore
+ getPurchases(): int
+ getOverloads(): int
+ getTransfers(): int
+ getStore(): GroceryStore


Location
: int
: int
stanceTo(Location
r): long
earingTo(Location other): long
nRefuel(): boolean
tLat(): int
tLon(): int
LocationEdge
- from: Location
- to: Location
+ getDistance(): long
+ getFrom(): Location
+ getTo(): Location
Extends
Extends
CustomerLocation
- customer: Customer
RefuellingStationLocation
- station: RefuellingStation
Drone
- state: State
- id: String
- fuel: int
- maxFuel: int
- capacity: int
- speed: int
- location: Location
- home: Location
- route: List<Location>
- pilotId: String
storeName: String
+ getId(): String
+ getCapacity(): String
+ getFuel(): int
+ getMaxFuel(): int
+ getSpeed(): int
+ getLocation(): Location
+ getPilotId(): String
+ setPilotId(id)
+ getFuelEfficiency(): int
+ setFuelEfficiency(efficiency)
+ getRefuellingRate(): int
+ setRefuellingRate(rate)
+ getRange(): int
+ getMaxRange(): int
+ getHome(): Location
+ getState(): State
+ getStoreName(): String
+ getRoute(): List<Location>
+ setRoute(route)
+ setFuel(fuel)
+ setLocation(location)
+ setState(state)
DroneService
- logger: Logger
- TYPE
- redisTemplate
- hashOperations
- timer: Timer
+ makeDrone(storeName, droneId, capacity, fuel, maxFuel, speed)
+ getDrone(storeName, droneId)
+ forEachDrone(storeName, consumer)
+ setDrone(storeName, drone)
+ deliverOrder(drone, order, customer)
+ updateDroneState(drone)
+ calculateRoute(start, end, range, maxRange)
+ flyToward(drone, goal, time)
+ saveDrone(drone)
+ forEachDrone(consumer)
child
RefuellingStation
- id: String
- location: Location
+ getId(): String
+ getLocation(): Location
+ setLocation(lat, lon)
Order
- droneId: String
- storeName: String
- customerAccount: String
- id: String
+getDroneId(): String
+getStoreName(): String
+getCustomerAccount(): String
+getId(): String
+setDroneId(droneId)
OrderService
- redisTemplate
- hashOperations
- listOperations
groceryStoreService
- customerService
- droneService
- pilotService
- efficiencyService
+startOrder(storeName, orderId,
parent
StationService
- redisTemplate
- hashOperations
+ makeRefuellingStation(id, lat, lon)
+ forEachRefuellingStation(consumer)
+ moveRefuellingStation(id, lat, lon)
StationApiCo
- stationService
+ makeRefuelling
+ getAllRefuelling
+ moveRefuelling


DroneApiController
- droneService
+makeDrone()
+getAllDrones()


ntroller
gStation()
gStations()
gStation()


+ incrementPurchases()
+ incrementOverloads()
+ incrementTransfers()
EfficiencyApiController
- efficiencyService
- groceryStoreService
+ getEfficiencies()
+ getStoreEfficiency()
DeliveryServiceException
CustomerCantAffordException
CustomerExistsException
CustomerNotFoundException
DroneCantCarryException
DroneExistsException
DroneNeedsFuelException
DroneNeedsPilotException
DroneNotFoundException
InsufficientCapacityException
ItemAlreadyOrderedException
ItemExistsException
ItemNotFoundException
LicenseExistsException
NoSuchRouteException
OrderExistsException
OrderNotFoundException
PilotExistsException
PilotNotFoundException
StationExistsException
StationNotFoundException
StoreExistsException
StoreNotFoundException

droneId, account)
+forEachOrder(storeName, consumer)
+getOrder(storeName, orderId)
+requestItem(storeName, orderId, itemName, quantity, unitPrice)
+hasOrder(storeName, orderId)
+purchaseOrder(storeName, orderId)
+cancelOrder(storeName, orderId)
+transferOrder(storeName, orderId)
+forEachLine(storeName, orderId, consumer)
+canCarry(storeName, drone, quantity, weight)
+getDroneOrders(storeName, droneId)
+getOrderWeight(storeName, orderId)
+getOrderPrice(storeName, orderId)
+getCustomerOrders(account)
+canAfford(customer, quantity, unitPrice)
+getDroneOrderSize(storeName, droneId)

child

OrderApiController

- orderService
- publisher

+ makeOrder()
+ getAllOrders()
+ requestItem()
+ getItems()
+ cancelOrder()
+ transferOrder()
+ purchaseOrder()

parent

child

MessageSubscriber

+ onMessage()

MessagePublisher

+ publish(message)