

```
sequenceDiagram
    participant Interface as Interface Loop
    participant Delivery as Delivery Service
    participant Drone
    participant Restaurant
    participant Package
    participant Ingredient
    participant Current as Current Location
    participant Home as HomeBase Location
    participant Destination as Destination Location

    Interface->>Delivery: getDrone()
    Delivery-->>Interface: Drone ref
    Interface->>Drone: getCurrentLocation()
    Drone-->>Interface: Location ref
    Interface->>Drone: getHomeBase()
    Drone-->>Interface: Location ref
    Interface->>Drone: loadDroneFuel(payload)
    Drone-->>Interface: capacity
    Interface->>Delivery: setDronePayload(payload, barcode, quantity, price)
    Delivery->>Drone: getPayload()
    Drone-->>Delivery: payload
    Delivery->>Ingredient: getBarcode()
    Ingredient-->>Delivery: barcode
    Delivery->>Package: incrementQuantity(quantity)
    Package-->>Delivery: newPackage(quantity, unitPrice)
    Delivery-->>Package: Package ref
    Delivery->>Ingredient: getDronePayload(payload, Package)
    Ingredient-->>Delivery: incrementRemainingCapacity(quantity)
    Delivery->>Drone: getDrone()
    Drone-->>Delivery: Drone ref
    Delivery->>Drone: getCurrentLocation()
    Drone-->>Delivery: Location ref
    Delivery->>Destination: calculateDistance(destination)
    Destination-->>Delivery: distance
    Delivery->>Drone: getHomeBase()
    Drone-->>Delivery: Location ref
    Delivery->>Home: calculateDistance(homeBase)
    Home-->>Delivery: returnDistance
    Delivery->>Drone: getFuel()
    Drone-->>Delivery: fuel
    Delivery->>Destination: getSpacesLeft()
    Destination-->>Delivery: spacesLeft
    ALT [If distance > Fuel]
        Delivery->>Interface: displayErrorMessage()
    ALT [If distance + returnDistance > Fuel]
        Delivery->>Interface: displayErrorMessage()
    ALT [If spacesLeft = 0]
        Delivery->>Interface: displayErrorMessage()
    ALT [else]
        Delivery->>Drone: flyToDestination(destinationLocation)
        Drone->>Current: incrementSpacesLeft()
        Current-->>Drone: decrementSpacesLeft()
        Drone->>Destination: calculateDistance(destination)
        Destination-->>Drone: distance
        Drone->>Drone: useDroneFuel(distance)
        Delivery->>Drone: getCurrentLocation()
        Drone-->>Delivery: Location ref
        Delivery->>Drone: getDronePayload()
        Drone-->>Delivery: payload
        Delivery->>Restaurant: makePurchase(buyerDrone, buyerIngredient, quantity)
        Restaurant->>Package: getDronePrice()
        Package-->>Restaurant: price
        Restaurant->>Package: addPendingAmount()
        Package-->>Restaurant: completePurchase(buyerIngredient, quantity)
        Restaurant->>Current: addSalesAmount()
        Current-->>Restaurant: getDroneID()
        Restaurant-->>Current: quantity
        ALT [If ingredient quantity = quantity requested]
            Restaurant->>Ingredient: removeIngredient()
        ALT [else]
            Restaurant->>Package: decrementQuantity(quantity)
        END
    END
    END
```