

EERD:

user = (uname, birthday, fname, lname, address)

customer = (uname [fk1], rating, credit)

fk1: uname -> user.uname

employee = (uname [fk2], taxID, hired, salary, service)

fk2: uname -> user.uname

drone_pilot = (uname [fk3], licenceID, experience)

fk3: uname -> employee.uname

floor_worker = (uname [fk4])

fk4: uname -> employee.uname

store = (storeID, sname, revenue, manage [fk7])

fk7: manage -> floor_worker.uname, manage is non-null

order = (orderID, sold_on, request [fk8], storeID, droneTag [fk9])

fk8: request -> customer.uname, request is non-null

fk9: storeID, droneTag -> drone.storeID, drone.droneTag, storeID, droneTag is non-null

drone = (storeID [fk4], drongTag, rem_Trip, capacity, control [fk10])

fk5: storeID -> store.storeID

fk10: control -> drone_pilot.uname, control is non-null

item = (barcode, iname, weight)

employ = (uname[fk11], storeID [fk12])

fk11: uname -> floor_worker.uname

fk12: storeID -> store.storeID

contain = (orderID [fk13], barcode [fk14], price, quantity)

fk13: orderID -> order.orderID

fk14: barcode -> item.barcode

Unhandled Constraints

- Ensure that each user is also a customer, employee, or both.
- Ensure that a customer's credit is greater or equal to the total cost of their orders.
- Ensure that each floor worker is employed by at least 1 store
- Ensure that each employee is not a floor worker and drone pilot simultaneously.
- Ensure that each order contains at least 1 item.
- Ensure that the manager for the store also works for that store.
- Ensure that no manager can manage multiple stores.
- Ensure that no supervisee makes more (i.e., has a higher salary) than their manager.