

Problem: Design an app for calling taxis (e.g. Uber).

Identify Objects and Behaviors:

Thing (Object) / Entity:

Information (Data) / (State)

Services (Behaviors) / Actions

Things:

Consumer:

Data: name, phoneNumber, emailAddress, password

Behaviors: signUp, signIn, tryAgain, hasAnAccount, hasSignedIn,  
addPaymentMethods, bookTaxi, cancelOrder, pay

Trip:

Data: pickupPlace, destination, pickupTime, carTypeChoose

Behavior:

Driver:

Data: name, rate

Behaviors: confirmOrder, cancelOrder, driveToPickup, driveToDestination

App:

Data: maps, carLocations

Behaviors: estimatePrice, matchTaxi, noticeDriver, calculatePrice, autoPay,  
recordRoute

Taxi:

Data: carType, licensePlate, driverInformation

Behavior:

Sequence of invoking behaviors in Objects:

CallTaxi:

Consumer luqing,

Trip trip,

Driver driver,

App Uber,

Taxi taxi,

if luqing.hasSignedIn

luqing.bookTaxi -> trip: connected

if connected is true

```

Uber.estimatePrice -> trip: estimation
Uber.matchTaxi -> trip, driver, taxi : response
Uber.noticeDriver -> driver
if driver.cancelOrder
    luqing.tryAgain
else
    driver.confirmOrder -> trip : approved
    driver.driveToPickup -> trip.pickupPlace, trip.pickupTime
    driver.driveToDestination -> trip.destination
    Uber.calculatePrice -> Uber.recordRoute : price
    luqing.pay -> price: response
    if response is true
        Uber.autoPay
    else
        luqing.tryAgain
    end
end
end
else
    if luqing.hasAnAccount
        luqing.signIn -> phoneNumber, password : response
        if response.success
            luqing.hasSignedIn
        else
            luqing.tryAgain
        end
    else
        luqing.signUp -> name, phoneNumber, emailAddress, password : account
        if account.success
            luqing.hasSignedIn
        else
            luqing.tryAgain
        end
    end
end
end
end

```