Problem: Design an app for calling taxis

Identify Objects and Behaviors:

Thing(Object)/Entity:

Information (Data)/(State):

Services (Behaviors)/Actions

Things:

Passengers:

Data: Phone, Name, Location, Destination

Behaviors: inputOriginAndDestination, search, pay, cancel, reviews

Driver:

Data: Name, rating, Phone, location

Behaviors: drive, pickUpPassengers, noticeTrafficSystem

TrafficSystem:

Data: RoadCondition, Route

Behaviors: getRoadCondition, getRoute,

CreditCard:

Data: Number, name, company, expiry, security code

Behavior:

CreditCardCompany

Data: name

Behavior: authorize transaction

Vehicle:

Data: brand, color, license plate number

Behavior:

Internet:

Data: Group of Apps( Collection of Apps)

Behavior: searchForElectronicApps

ElectronicApp:

Data: Uber, Lyft

Behavior: search, display, match, compare

Sequence of invoking behaviors on Objects:

MakeAnOrder:

Passenger emily,

Driver billy

TrafficSystem TS,

Internet internet,

ElectronicApp Uber,

Vehicle billy’s vehicle,

CreditCard emily’s card, billy’s card

CreditCardCompany visa

Route route

If Internet.isAvailable

emily.loginToElectronicAppInInternet->Internet: authorize

if( authorize is true)

emily.inputOriginAndDestination -> Uber

Uber.searchForMatchedDriver -> location, billy: matched

billy = driver

if matched is true

TS.getRoadCondition and getRoute -> route

if emily.agreeToStartThisRoute

emily.waitForTaxi-> brand, color, license plate number, Uber

else

emily.cancelTheTransaction - >Uber

end

if route.ended

emily.payForTheTransaction -> Uber: orderConfirmation

response = orderConfirmation

end

else

emily.orderAgain

end

else

emily.registerANewAccount

end

else

emily.browseAppAfterAWhileBack

end