1. Hiking

1.1. Rent a ReachNow to drive to mountain rainier on weekends with friends

1.2. On the way, fill the gas

1.3. On the way, eat at restaurant for lunch

1.4. Buy the ticket to get into the park

1.5. Book a hotel or check in the hotel.

Hiker

Data: Name, Address, Phone number, Credit card

Behaviors: rent, drive, talk, pay, eat, buy, book, check

ReachNow

Data : URL, Car[] cars, BankAccount

Behaviors : search, sort, display, compare , placeTheOrder

Car

Data: Brand, Mile, Date, Price

Behaviors:

Friend

Data: Name, Address, Phone number

Behaviors: talk

Gas

Data: price, gallon

Behaviors:

Gas Station

Data: Gas, Location

Behaviors: fill

Restaurant

Data: Address, Phone number, Price, Review, Menu, Food, Drink

Behaviors: serve, cook

Park

Data: Location, Route, Nature, Operating hours, Gift shop, Current condition

Behaviors: sell

Ticket

Data: Price, Date

Behaviors:

Hotel

Data: Address, Phone number, Price, Distance from park, Available date, Review, Rate, Room

Description

Behaviors:

Sequence of Flow - Invoke Objects with Behaviors

Hiker peter

Friend1 kevin

Friend2 lily

Car bmw-suv

Gas gas

Gas Station shell

Restaurant chipotle

Park yellowstone

Ticket ticket

Hotel marriott

If ReachNow.isAvailable

peter.findDesitableCar(bmw-suv)

if (marriott.room > 0)

peter.book("marriott ", date, price);

end

if peter.hasFriends()

peter.inviteFrineds(kevin, lily)

friends.makeAResponse(peter)

end

if car is out of gas

shell.fillGas(gas, gallon)

end

if (peter.isHungry || kevin.isHungry || lily.isHungry)

while (!peter.isfull)

chipotle.serve(food, drink)

peter.eat(chipotle, price, food, drink)

while (!kevin.isfull)

chipotle.serve(food, drink)

kevin.eat(chipotle, price, food, drink)

while (!kevin.isfull)

chipotle.serve(food, drink)

lily.eat(chipotle, price, food, drink)

end

if ticket.isAvailable

yellowstone.sellTicket(ticket, price, date)

peter.buyTicket(ticket, price, date);

else

peter.drive(bmw-suv, address);

end

2. Organize a career fair(Suppose you are the organizer)

2.1. Search for different kinds of companies

2.2. Connect with employers

2.3.Invite companies that are willing to come

Organizer

Data: name

Behaviors: search, rent, host, talk

Company

Data: name, HR, phone

Behaviors: come

HR

Data: name, company

Behaviors: talk

Sequence of Flow - Invoke Objects with Behaviors

Organizer peter

Company company

HR hr

peter.search(company[])

peter.findDesirableCompany()

loop

peter.connect(hr)

if hr.agreeToCome(peter))

company.come == true

else

company.come == false

end

3. Order Pizza from Pizza Hut

3.1. Log in PizzaHut.com

3.2. select the perfect pizza

3.3. make the order

3.4. receive pizza

Pizza Hut

Data: URL, Phone number

Behaviors:

Pizza

Data: price,

Behaviors:

Myself

Data: Name, Address, Phone number

Behaviors: order

Sequence of Flow - Invoke Objects with Behaviors

if pizzahut.com is available

4. Design a code sharing platform (eg: Github).

4.1.user log in

4.2.

4.3.

4.4.

platform

Data: Name, Website

Behaviors:

Sequence of Flow - Invoke Objects with Behaviors

5. Design a soft-drink/snacks vending machine.

5.1.

5.2.

5.3.

5.4.

soft-drink/snacks

Data: Name, Price,

Behaviors:

Sequence of Flow - Invoke Objects with Behaviors