Scenarios:

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.

2. Organise a career fair(Suppose you are the organiser)

2.1 design a webside for register the career

2.2 Predict the number of the people who will join this career fair

2.3 find a place that have enough room for them

2.4 invite the company who is interested for the career fair

2.5 design the timetable for the career fair

2.6 inform all the students who register this career fair

3. Order Pizza from Pizza Hut

3.1 make a call to the Pizza Hut to order the Pizza

3.1.1 ask the address of the Pizza Hut

3.1.2 look up the price of the pizza

3.1.3 tell them which pizza i would like to order

3.1.4 tell them when i would like to pick up the pizza

3.2 take the bus to the Pizza Hut on time

3.3 identify the order and get my pizza

3.4 return to home

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

4.1 make a requirement analysis for this platform

4.1.1 design the programming language for this platform

4.1.2 design the struture for this platform

4.1.3 find the type of the server for this platform

4.1.4 design what type of database we need to use(redis or mysql)

4.2 design the website for this platform

4.2.1 use Html or jsp to fill the website

4.3 design how to Connect the website with the database

4.4 invite the programmers to code this platform

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

5.1 show the GUI for our customer

5.1.1 the first Button for the gui is "Soft drink"

5.1.2 the second Button for the gui is "Snacks"

5.1.3 determine soft drink or snacks

5.2 look up the food information to choose the food

5.3 choose the amount of the food

5.4 pay for the food

5.5 the machine will give the food to you

Problem: Hiking

Identify Objects and Behaviors:

Thing (Object)/Entity:

Information (Data)/(State):

Services (Behaviors)/Actions

Things:

Person:

Data :name, Id ,money, phone number;

Behavior: rent, cars, fill gas, eat, buy ticket, book hotel

ReachNow:

Data:price,carName,address,time

Behavior:rentCars

Gas Station:

Data: price, volume. address ,time, name

Behavior; fill the gas according to the demand,

Restaurant;

Data; food price, food menu, address, time, name,

Behavior: provide the food for customer.

Hotel:

Data: Price, Duration, address, name,

Behavior: provide the room for customer

Sequence of invoking behaviors on Objects:

GoHiking

Person peter;

ReachNow rentcar;

GasStation station;

Restaurant food;

Hotel homelike;

If. Peter..wantToHike == true;

If peter.HaveCar == false;{

Rentcar,rentcar();

}

if(rentcar.car.gas == false){

station.fillTheGas();

}

if(peter..isHungry == true){

food.eatFood();

}

if(peter.needToHotel == true){

homelike.bookTheHotel();

}

Problem Organise a career fair(Suppose you are the organiser)

Identify Objects and Behaviors:

Thing (Object)/Entity:

Information (Data)/(State):

Services (Behaviors)/Actions

Things:

Person

Data :name, Id , , phone number;

Behavior: Join Career Fair

Website;

Data:name ;number

Behavior:register for career fair

House;

Data;price, address, roomNumber

Behavior: thePlaceOfcareerfair

Company:

Data:name;Person number;Isjoin

Behavior: presentTheCareer

Timetable:

Data;time

Behavior:theTimeforCareer

Organize Career Fair

Website RegisterWeb;

Company JoinCompany

House RoomForCareer

Person.Peter

Timetable Time

If (IsCareerFair ==true){

Num = Register.getNumber()

RoomForCareer.setroom(num);

If(Organzer.InviterCompany() == true){

JoinCompany.present;

}

}

Timetable(RoomeForCareer.gettime Company.gettime);

Return time;

Person.Settime(time);

Person.JoinCareer();

Problem: Order Pizza from Pizza Hut

Things

Person:

Data :name, Id ,money, phone number;

Behavior: order Pizza

Pizza :

Data :name,price, phone number;

Behavior: Pizza

Resturant

Data; food price, food menu, address, time, name,

Behavior: provide the PIzza for customer.

Bus:

Data:price,BusName,address,time

Behavior:take BUS;

Oder a pizza

Person peter;

Bus busName;

Restaurant PizzaShop;

Pizza pizza;

If (IsWantPizza == true){

Peter.order(Pizza shop);

Pizzashop.setPizzaName(Pizza.getPizzaname)

Pizzashop.setPizzaPricePizza.getPizzaPrice)

Pizzashop.setPizzatime(Pizza.getPizzatime)

}

peter.Settime(Pizza.getPizzatime)

peter.takeBus(busName)

if(peter.gettime() == pizzaShop.gettime()){

Peter.getPizaa(pizzashop);

}

peter.returnHome(bus);

Problem: Design a code sharing platform

Identify Objects and Behaviors:

Thing (Object)/Entity:

Information (Data)/(State):

Services (Behaviors)/Actions

Things;

Coder:

Data :name, Id , , phone number;language

Behavior: code the platform

ProgramLanguage:

Data :name,

Behavior: The coding language for the platform

Requirement:

Data :Structure;backend,frontEnd;

Behavior: the requirement for the platform

DataBase:

Data :name, DataType

Behavior: store the data for the platform

Design a code sharing platform

Coder peter;

ProgramLanguage C;

Requirement;

DataBase db;

If(coder.IsSetThePlatform == true){

Requirement.setRequriement();

C.setBacklanguage ( Requirement.getBackLanguage());

C.setFrontlanguage ( Requirement.getFrontLanguage());

db.setlanguage ( Requirement.getDatabaseLanguage());

coder.coding(C.getBacklanguage,C,getFrontLanguage,db.getlanguage);

coder.connect(Frontend,BackEnd);

}

Problem;Design a soft-drink/snacks vending machine.

Thing:

SoftDrink:

Data:price,DrinkName,number

Behavior:ProvidevDrinks.

Snacks:

Data:price,DrinkName,number

Behavior:ProvidevSnacks.

Buyer:

Data :,money, amount;choice

Behavior: buy product;

Design a soft-drink/snacks vending machine.

Buyer.peter;

Snacks snackName;

Drinks DrinkName;

If(peter.IsToBuy == true){

String choice = peter.choice

If(choice == snack){

snackName = peter.choice.getname

snackPrice = peter.choice.getprice;

snackAmount = peter.choice.getAmount;

peter.buy(snackName,snackPrice,snackAmount);

}

if(choice == drink){

drinkName = peter.choice.getname

drinkPrice = peter.choice.getprice;

drinkAmount = peter.choice.getAmount;

peter.buy(drinkName, drinkPrice, drinkAmount);

}

peter.havePaid();

}