**INFO-5100 APPLICATION ENGINEERING AND DEVELOPMENT**

**ASSIGNMENT-1**

**Manjula Bashini Prabhu**

(001831484)

**Problem 1:**

**Hiking:**

**Objects:**

Hiker

ReachNowService

Gas station

Restaurant

Mt.Rainier National Park

Hotel

**Attributes and Behaviours of the objects:**

**Hiker:**

Attributes: HikerName, Address, Mobile No

Behaviours: bookCar, searchCar , driveCar, findgasStation, fillGas, payGasBill, findRestaurant, orderFood, eatFood,payFoodBill, visitRainier, buyPass, hike, findHotel, reserveHotel, stay,checkout

**ReachNowService:**

Attributes: CarNo, AccessKey

Behaviours: AllowUserAccess

**GasStation:**

Attributes: GasAmount

Behaviours: GasFill, BillGenerate

**Restaurant:**

Attributes: RestaurantName, Food, FoodAmount, Quantity

Behaviour: getOrder, serve, billGeneration

**Mt.RainierNationalPark:**

Attributes: TicketNo,Amount

Behaviour: generatepass,provideEntry

**Hotel:**

Attibutes: hotelname,address

Behaviour: ProvideRoom, Billgeneration,HotelAmount.

**PSEUDOCODE:**

Hiker Jim

ReachNowService rnservice

Gasstation gstation

Restaurant rest

Mt.RainierNationalPark rainier

Hotel ht

If rnservice.isAvailable

Jim.bookCar

Jim.searchCar(CarNo)

Jim.enterKey(AccessKey)

Rnservice.AllowUserAccess(HikerName)

Jim.driveCar(CarNo)

Jim. findgasStation(Address)

gstation.gasFill

gstation.billGenerate

Jim.payGasBill(GasAmount)

Jim. findRestaurant(RestaurantName)

Rest.getOrder(HikerName)

Jim.orderFood(Food,Quantity)

Rest.serve(Food,Quantity)

Jim.eatFood(Food,Quantity)

Rest.billGeneration

Jim. payFoodBill(FoodAmount)

Jim.visitRainier

Jim.buyPass(TicketAmount)

If Ticket=”available”

Rainier. Generatepass(TicketNo)

Rainier.ProvideEntry

Jim.hike

Else

Jim.canthike

End If

Jim.findHotel(HotelName)

Jim.drivetoHotel(Address)

If room=”found”

Jim.reserveHotel

Jim.stay

Ht.ProvideRoom(Jim)

Jim.checkout

Ht.Billgenerate

Jim.payHotelBill(HotelAmount)

Else

Jim.stayatHome

End If

**Problem 2:**

**Organizing a Career Fair:**

**Objects:**

Organizer

Company

Attendee

Management

**Attributes and Behaviours of the objects:**

**Organizer:**

Attributes: Name

Behaviour: findEmployers, inviteEmployer, inviteStudents, seekPermission, chooseVenue, chooseTime

**Company:**

Attributes: Name

Behaviour: acceptInvite, giveInformation, collectResumes

**Attendee:**

Attributes: Name, ProgramInfo, GPA

Behaviour: attendJobFair, speaktoEmployer

**Management:**

Attributes:

Behaviour: Authorize,provideFacilities

**PSEUDOCODE:**

Organizer Tim

Company AbcCorp

Attendee Student

Management School

Employer=Tim. findEmployers(AbcCorp)

If Employer isAvailable

Tim.seekpermission(School)

School.authorize(Tim)

Tim.chooseVenue

Tim.chooseTime

School.provideFacilities

Tim.inviteEmployer(AbcCorp)

AbcCorp.acceptInvite(Tim)

Tim.inviteStudents(Attendees)

Students.attendJobFair

AbcCorp.giveInformation(Name, ProgramInfo, GPA)

Student.speaktoEmployer(AbcCorp)

AbcCorp.collectResumes(Name)

Else

Tim.cantOrganize

End If

**Problem 3:**

**Ordering Pizza from Pizza Hut**

**Objects:**

Customer

PizzaServiceWebSite

CreditCard

**Attributes and Behaviours of the objects:**

**Customer:**

Attributes: Name, Address, PhoneNo, Username, Password, Order, BillAmount

Behaviour: logIn, OrderPizza, PayBillAmount, EnterCardDetails, GiveDetails

**PizzaServiceWebsite:**

Attributes: PizzaOptions, Price,WebsiteAddress

Behaviour: LogInValidation, DisplayWebPage, GetOrder, GetCustomerDetails, CalculateBillAmount, AcceptPayment

**CreditCard:**

Attributes: CardNo,Name,CompanyName,Expiry,SecurityCode

Behaviour: ValidateDetails, AcceptPayment

**PSEUDOCODE:**

Customer Peter

PizzaServiceWebsite Pizzahut

CreditCard Chase

Peter.login(WebsiteAddress,Username,Password)

Pizzahut. LogInValidation

If login=”successful”

Pizzahut.DisplayWebPage

Peter.OrderPizza(PizzaOptions, Order)

Pizzahut.getOrder(Order)

Peter.giveDetails(Name,Address,PhoneNo)

Pizzahut.getCustomerDetails(Name,Address,PhoneNo)

Pizzahut.Calculatebillamount(Order)

Peter.PayBillAmount(BillAmount)

Peter.EnterCardDetails(CardNo,name,Expiry,SecurityCode)

Chase.ValidateDetails(CardNo,Name,Expiry,SecurityCode,BillAmount)

If Details=”correct”

Chase.AcceptPayment(BillAmount)

PizzaHut.ConfirmOrder

Else

Order is Denied

End If

Else

Login is denied

End If

**Problem 4:**

**Design a code sharing platform:**

**Objects**

User, Peers, CodesharingSystem

**Attributes and Behaviours of the objects**

**User:**

Attributes: username, password, codeid, code, comment, peername

Behavior: createAccount, deleteAccount, login, logout, CommitCode, amendCode, removeCode, addComment, deleteComment, addPeers, removePeers, searchPeers, sendRequest, ShipCode

**Peers:**

Attributes: corrections, comment, rating,

Behavior: suggestCorrections, addComment, addRating, acceptRequest, rejectRequest

**CodesharingSystem:**

Behaviour: loginvalidation, invokelogout, provideaccess, postcode, deletecode, enableuser, disableuser, postcorrections, postcomments, postrating

**Pseudocode:**

User jill

Peers jim

CodesharingSystem github

If jill=”new user”

jill.createAccount(username,password)

github.enableuser(jill)

Else

jill.login(username,password)

github.loginvalidation(username,password)

if login=”valid”

github.provideaccess(username)

jill.CommitCode(code)

github.postCode(code)

jill.searchpeers(jim)

if jim=”found”

jill.sendRequest(jim)

Case based on the option used

Case ”AcceptRequest”

jim.acceptRequest(jill)

Case ”Correctcode”

jim.suggestcorrections(corrections)

Case “AmendCode”

jill.amendCode(corrections,code)

Case “AddComment”

If jim=”wants to add comment”

jim.addComment(comment)

github.postComment(jim,comment,jill)

End If

If Jill=”want to add Comment”

jill.addComment(comment)

github.postComment(jill,comment,jim)

End If

Case “Addrating”

jim.addrating(jill,rating)

github.postrating(rating,jim,jill)

Case “ShipCode”

Jill.shipCode(code)

End Case

Else

unable to add jim as peer

End If

Else

Login not valid

End If

End If

**Problem 5:**

**Design a soft drink/snack vending machine**

**Objects:**

User

Vendingmachine

**Attributes and Behaviours of the objects:**

**User:**

Attributes: foodcode, billamount, paidamount

Behavior: entercode, insertbill, insertcoin, cancel

**Vending machine:**

Attributes: food

Behavior: searchbycode, validateamount, rejectpayment, dispense, producechange

**Pseudocode:**

User Tom

Vendingmachine snackbar

Tom.entercode(foodcode)

Tom.insertbill(paidamount)

Snackbar.searchbycode(foodcode)

Snackbar.validateamount(billamount)

If snackbar=”foundfood” and paidamount>=”billamount”

Snackbar.dispense(food)

If billamount>”correctamount”

Snackbar.producechange(billamount,paidamount)

Else

Process is over

End If

Else

Snackbar.rejectpayment(paidamount)

Unable to dispense

End If