

Home Assignment 1

Course: IS-5100-Application Engineering and Development

Instructor: Prof. Siva Dosapati

Name: Kuangying Li

Date: 9/13/2018

Question 1

Problem: Order a hotel online before a trip.

Identify Objects and Behaviors:

Thing (Object)/Entity:

Information (Data)/(State): Services (Behaviors)/Actions

Things:

Guest:

Data: Name, phone number, email, ID number

Behaviors: search, compare, book, cancel

Internet:

Data: Booking.com, Orbitz, Expedia, Hotels.com

Behaviors: searchForHotelBookingWebsites

HotelBookingWebsite:

Data: URL, Hotels

Behaviors: search, sort, compare, placeTheOrder

Hotel:

Data: Name, price, address, facilities, reviews

Behavior

CreditCard:

Data: Number, name, expired date, CVV

Behavior

CreditCardCompany:

Data: Name, phone number

Behavior: authorize transaction

Sequence of invoking behaviors on Objects:

OrderHotelOnline

Guest lee,

Internet myInternet,

HotelBookingWebsite orbitz,

Hotel myHotel,

CreditCard card,

CreditCardCompany chase,

BookingConfirmation response

If Internet.isAvailable

lee.searchInInternet ->myInternet, question: Collection of HotelBookingWebsite

//lee.findGoodWebisteFirstPage -> Collection of Websites: websiePageNumber=1;

Loop

If lee.findsNoPages

Break

end

lee.findGreatHotelBookingWebsite ->internet, question, page Number: website

orbitz = website;

If orbitz is not empty

break

else

pageNumber = pageNumber + 1

end

End

orbitz = website

If orbitz is not empty

Orbitz.searchForHotel -> name, price, address, facilities, reviews: Collection of Hotel

myHotel = Hotel

lee.bookHotel-> myHotel, creditCard, address, orbitz: bookingConformation

response = bookingConfirmation

else

lee.doNotFindHotel

else

Lee.doNotFindInternet

Question 2

Problem: Design an app for calling taxi.

Identify Objects and Behaviors:

Thing (Object)/Entity:

Information (Data)/(State): Services (Behaviors)/Actions

Things:

Passenger:

Data: Name, PhoneNumber, Destination

Behaviors: placeOrder, pay, rate

TaxiAppCompany:

Data: Lyft, Uber, Flywheel, Migo

Behaviors: receiveOrderFromPassenger, assignOrderToDriver, collectMoney, payToDriver

Driver:

Data: Name, DriverLicense, VehicleNumber

Behaviors: receiveOrderFromApp, drive, collectMoney, rate

CreditCaard:

Data: Number, name, expired date, CVV

Behavior

CreditCardCompany:

Data: Name, phone number

Behavior: authorize transaction

Sequence of invoking behaviors on Objects:

CallTaxiOnline:

Passenger lee,

TaxiAppCompany uber,

Driver fernando,

CreditCard card,

CreditCardCompany chase,

OrderConfirmation response

If PhoneInternet.isAvaliable

Lee.searchInInternet -> cellphoneInternet, question: Collection of TaxiApp

//lee.findGoodTaxiApp -> Collection of App: appName;

If lee.findsNoTaxiApps

break

end

lee.findGoodTaxiApp -> cellphoneInternet, question, appName: application;

uber = application;

if uber is not empty

uber.searchAvaliableDriverNearby -> name, driverLicense, vehicleNumber : Collection of

Driver;

fernando = driver

lee.placeOrder -> Fernando, creditCard, destination, uber: orderConfirmation

response = orderConfirmation

else

lee.findsNoDriverNearby

else

lee.findsNoInternet

Question 3

Problem: Design a job searching and posting platform.

Identify Objects and Behaviors:

Thing (Object)/Entity:

Information (Data)/(State): Services (Behaviors)/Actions

Things:

JobSeeker:

Data: Name, phone number, email, ID number

Behaviors: search, compare, apply

Internet:

Data: Indeed, LinkedIn, CareerBuilder, Glassdoor

Behaviors: searchForJob

RecruitmentWebsite:

Data: URL, Company, Position

Behaviors: search, sort, compare

Position:

Data: Duty, income, company

Behavior

Sequence of invoking behaviors on Objects:

SearchJobOnline

JobSeeker lee,

Internet myInternet,

RecruitmentWebsite indeed,

Position softwareEngineer,

CreditCard card,

CreditCardCompany chase,

Application response

If Internet.isAvailable

lee.searchInInternet ->myInternet, question: Collection of RecruitmentWebsite

//lee.findGoodWebisteFirstPage -> Collection of Websites: websiePageNumber=1;

Loop

 If lee.findsNoPages

 Break

 end

 lee.findGreatRecruitmentWebsite ->internet, question, page Number: website

 indeed = website;

 If indeed is not empty

 break

 else

 pageNumber = pageNumber + 1

 end

End

indeed = website

If indeed is not empty

 Indeed.searchForPosition -> duty, income, company: Collection of Position

 myPosition = SoftwareEngineer

 lee.searchJob-> softwareEngineer, indeed: application

 response = application

else

 lee.doNotFindJob

else

 Lee.doNotFindInternet

Question 4

Problem: Order food in a restaurant.

Identify Objects and Behaviors:

Thing (Object)/Entity:

Information (Data)/(State): Services (Behaviors)/Actions

Things:

Customer:

Data: Name, TabeNumber, food

Behaviors: order, eat, pay

Restaurant:

Data: Name, Location, OpeningHours, DishSystem, AveragePrice

Behaviors: cook, receivePayment

Dish:

Data: Name, price, taste

Behavior

CreditCard:

Data: Number, name, expired date, CVV

Behavior

CreditCardCompany:

Data: Name, phone number

Behavior: authorize transaction

Sequence of invoking behaviors on Objects:

OrderHotelOnline

Customer lee,

Restaurant chipotle,

Dish burrito

CreditCard card,

CreditCardCompany chase,

Receipt response

If lee.isHungry

lee.searchNearby -> nearbyRestuarant, question: Collection of NearbyRestuarant

//lee.findGoodRestuarant -> Collection of Websites: websiePageNumber=1;

chipotle = restuarant

If chipotle is not empty

chipotle.findDish -> name, price, taste: Collection of Dish

myDish = burrito

lee.orderFood-> myDish, creditCard: receipt

response = receipt

else

lee.doNotFindRestuarant

else

lee.doNotNeedFood

Question 5

Problem: Design a course registration platform.

Identify Objects and Behaviors:

Thing (Object)/Entity:

Information (Data)/(State): Services (Behaviors)/Actions

Things:

Student:

Data: Name, phone number, email, ID number

Behaviors: search, register

Internet:

Data: NeuRegisterPlatform

Behaviors: searchAndRegisterCourses

RegisterPlatform:

Data: URL, CourseName

Behaviors: search, sort, register, cancel

Course:

Data: Name, Number, Time, Classroom, Professor, TA

Behavior

Sequence of invoking behaviors on Objects:

OrderHotelOnline

Student lee,

Internet myInternet,

RegisterPlatform neu,

Course info5100,

RegisterConfirmation response

If Internet.isAvailable

lee.searchInInternet ->myInternet, question: Collection of NeuRegisterPlatform

//lee.findCourseList -> Collection of courseList: courseListPageNumber=1;

Loop

 If lee.findsNoPages

 Break

 end

 lee.findCourseList ->internet, question, page Number: course

 info5100 = course;

 If info5100 is not empty

 break

 else

 courseListPageNumber = courseListPageNumber + 1

 end

End

info5100 = course;

If info5100 is not empty

 NeuPlatform.searchForcourse -> name, number, time, classroom, professor, TA: Collection

of course

 Info5100 = Course

 lee.registerCourse-> infor5100: registerConformation

 response = registerConfirmation

else

 lee.doNotFindCourse

else

 Lee.doNotFindInternet