1. **Design a course registration platform**

Objectives and behaviours

Internet Service:

Data: websiteAddress

Behaviours: connect

user:

data: username, password, listOfCoursesSelected

behavior: logIn, selectCourse, registerForCourse, payFees, selectLevel, dropCourse, selectTotalAmount,

addCourseToCourseList

portal:

data:

behavior: takeUserToListOfLevel, takeUserToListOfCourses, display fees, sendToMainPage,isEligible

level:

data: levelType

behavior:

course:

data: courseName, courseId, courseLevel

behavior: isAvailableForSelecting, isMandatory

financial administration:

data: amountToValidate

behavior: validateAmountTotal, validateAmountInstallment

CreditCard:

Data: name, bankName, cardNumber, address, securityCode, expiryDate

Behavior: chargeAmount, authorisedTransaction

Bank:

Data:

Behaviors: authorisedTransaction

HelpService:

Data: servicePhoneNumber

Behaviours: chatOnline, Question, Answer

Sequence of invoking behaviors on objects

Registering courseInPortal

User rekha;

Portal neu;

If the internet.isAvailable

Rekha.loginToApp-> username, password: connected

neu.takeUserTolistOfLevels

if selecting level

rekha.selectForLevel-> listOfLevel -> rekha.selectLevel : Grad

if course.isAvailableForSelecting

neu.takeUserToListOfCourse

rekha.selectCourse

rekha.addCourseToCourseList

rekha.selectTotalAmount

else

neu.takeUsertoListOfCourse

end

Loop: listOfCoursesSelected -> Course

if neu.isEligible : Course

bank.authoriseTransaction

end

else

neu.listOfCoursesSelected

1. **Order food in a food delivery app.**

Objectives and behaviours

Internet Service:

Data: Name, phoneNumber

Behaviours: connect

App:

Data: listOfRestaurants, map, listOfDeliveryBoys

Behaviors: askUserToSelectRestaurant, takeUserToListOfRestaurants, giveUserTimeExpected,

sendOrderToRestaurant, pickAndAssignOrderFreeDeliveryBoy

User:

Data: Name, phone, userLocation, destination, deliveryTime

Behaviors: logInToApp, searchforRestaurant, callForDelivery, writeReviews, cancelOrder, contactDeliveryBoy,

changeOrder, selectRestaurant, selectFoodfromRestaurant, placeTheOrder

DeliveryBoy:

Data: name

Behaviours: contactUser, deliveryToUser, isFreeToDeliver

Restaurant:

Data: phone, address, menu, timings

Behaviours: isAvailableForDeliveryNow, takeOrder, makeOrder, provideEstimatedTime

CreditCard:

Data: name, bankName, cardNumber, address, securityCode, expiryDate

Behaviors: chargeAmount, authorisedTransaction

Bank:

Data:

Behaviours: authorisedTransaction

HelpService:

Data: servicePhoneNumber, emailAddress

Behaviours: chatOnline, answerPhone

Sequence of invoking behaviors on objects

orderingFoodInApp

user rekha;

deliveryBoy boy;

app doordash;

if the Internet. isAvailable

rekha.loginToApp ->credentials: connected

doordash. takeUserToListOfRestaurants

if ordering now

rekha.searchForRestaurant-> listOfRestaurants -> rekha.selectRestaurant : Restaurant

if  restaurant.isAvailableForDeliveryNow

rekha.selectFoodfromRestaurant

rekha.placeTheOrder-> creditCard, address: Confirmation

doordash.sendOrderToRestaurant

doordash.pickAndAssignOrderFreeDeliveryBoy : deliverBoy

doordash.giveUserTimeExpected

deliverBoy.deliveryToUser

else

    doordash.takeUserToListOfRestaurants

end

bank.authoriseTransaction

1. **Design a platform for buying tickets of local events**

Objectives and behavior:

Internet Service:

Data: Name, phoneNumber

Behaviours: connect

App:

Data: List of events, map

Behaviours: askUserToSelectEvent, takeUserToListOfEvents, getEvent Confirmation

User:

Data: Name, phone, userLocation

Behaviors: logInToApp, searchForLocalEvents, writeReviews, cancelTicket,

selectEvent, bookTicketForEvent, placeTicketOrder, contactCustomerCare, checkForOffers, useOffer

Event:

Data: phone, address, timings

Behaviours: isAvailableForBooking, buyTickets

CreditCard:

Data: name, bankName, cardNumber, address, securityCode, expiryDate

Behaviors: chargeAmount, authorisedTransaction

Bank:

Data:

Behaviours: authorisedTransaction

Customer Care Service:

Data: servicePhoneNumber, emailAddress

Behaviours: chatOnline, answerPhone

Sequence of invoking behaviors on objects

buying TicketsInEvb

User rekha;

App evb;

if the internet.isAvailable

rekha.logInToApp -> credentials : connected

evb.takeUserToListOfEvents

if selecting event now

rekha.searchForLocalEvent-> listOfEvent -> rekha.selectEvent : Event

if  event.isAvailableForBookingNow

rekha.bookTicketForEvent

rekha.checkForOffers

if offer.isAvailable: offer

rekha.useOffer

rekha.placeTheOrder-> creditCard, address: Confirmation

else

evb.takeUserToListOfEvent

end

bank.authoriseTransaction

1. **Buy a computer from Amazon**

Objectives and behavior:

Internet Service:

Data: name, phoneNumber

Behaviours: connect

App:

Data: ListOfComputers

Behaviours: askUserToselectComputer, takeUserToListOfComputer, giveUserTimeExpected,

User:

Data: name, phone, userLocation, destination, deliveryTime

Behaviors: logInToApp, searchForComputers, writeReviews, cancelOrder, selectComputer, placeOrder, checkForOffers, useOffer, contactCustomerCare

Computer:

Data: computerId, computerType

Behaviours: isAvailableForDelivery

CreditCard:

Data: name, bankName, cardNumber, address, securityCode, expiryDate

Behaviors: chargeAmount, authorisedTransaction

Bank:

Data:

Behaviours: authorisedTransaction

HelpService:

Data: servicePhoneNumber, emailAddress

Behaviours: chatOnline, answerPhone

Sequence of invoking behaviors on objects

ordering computerFromAmazon

User rekha;

App amazon;

if the internet.isAvailable

rekha.logInToApp -> credentials : connected

amazon.takeUserToListOfcomputer

if ordering computer now

rekha.searchForComputer-> listOfComputer -> rekha.selectComputer : Computer

if  computer.isAvailableForDeliveryNow

rekha.selectComputer

rekha.placeTheOrder-> creditCard, address : Confirmation

rekha.checkForOffers

if offer.isAvailable: offer

rekha.useOffer

rekha.placeTheOrder-> creditCard, address : Confirmation

else

amazon.takeUserToListOfComputers

end

bank.authoriseTransaction

1. **Design an app for booking hotels**

Objectives and behavior:

Internet Service:

Data: Name, phoneNumber

Behaviours: connect

App:

Data: ListOfHotels, map

Behaviours: askUserToSelectHotel, takeUserToListOfHotel, getHotelConfirmation

User:

Data: Name, phone, userLocation

Behaviors: logInToApp, searchForHotels, writeReviews, selectHotel, bookingHotel, changeHotel, cancelBooking, contactCustomerCare, checkForOffer

Hotels:

Data: hotelName, address

Behaviours: isAvailableForBooking,

CreditCard:

Data: name, bankName, cardNumber, address, securityCode, expiryDate

Behaviours: chargeAmount, authorisedTransaction

Bank:

Data:

Behaviours: authorisedTransaction

HelpService:

Data: servicePhoneNumber, emailAddress

Behaviours: chatOnline, answerPhone

Sequence of invoking behaviors on objects

booking hotelInApp

User rekha;

App trivago;

if the internet.isAvailable

rekha.logInToApp -> credentials : connected

trivago.takeUserToListOfHotel

if selecting hotel now

rekha.searchForHotel-> listOfHotel -> rekha.selectHotel : Hotel

if  Hotel.isAvailableForBookingNow

rekha.selectHotel

rekha.bookHotel-> creditCard, address : Confirmation

rekha.checkForOffers

if offer.isAvailable: offer

rekha.useOffer

rekha.placeTheOrder-> creditCard, address : Confirmation

else

trivago.takeUserToListOfHotel

end

bank.authoriseTransaction