**19CSE204 Object Oriented Programming**

**CASE STUDY REPORT**

**Online Ticket Reservation System**

**Problem Statement :**

Rail transport as one of the most important means of transport, has played an important role in the

transport industry in China. With Chinese rapid economic development, the railway lines and passengers

have been increasing year by year in the country. With such a huge customer base, buying train tickets

problem has been very prominent. The electronic commerce could solve the problem of railway ticketing.

Introduced a new online ticketing system is not only technological innovation, but also will improve the

railway services, to a certain extent, solve the difficult problem of railway ticketing

Rail transport as one of the most important means of transport, has played an important role in the

transport industry in China. With Chinese rapid economic development, the railway lines and passengers

have been increasing year by year in the country. With such a huge customer base, buying train tickets

problem has been very prominent. The electronic commerce could solve the problem of railway ticketing.

Introduced a new online ticketing system is not only technological innovation, but also will improve the

railway services, to a certain extent, solve the difficult problem of railway ticketing

Rail transport as one of the most important means of transport, has played an important role in the

transport industry in China. With Chinese rapid economic development, the railway lines and passengers

have been increasing year by year in the country. With such a huge customer base, buying train tickets

problem has been very prominent. The electronic commerce could solve the problem of railway ticketing.

Introduced a new online ticketing system is not only technological innovation, but also will improve the

railway services, to a certain extent, solve the difficult problem of railway ticketing

Rail transport as one of the most important means of transport, has played an important role in the

transport industry in China. With Chinese rapid economic development, the railway lines and passengers

have been increasing year by year in the country. With such a huge customer base, buying train tickets

problem has been very prominent. The electronic commerce could solve the problem of railway ticketing.

Introduced a new online ticketing system is not only technological innovation, but also will improve the

railway services, to a certain extent, solve the difficult problem of railway ticketing

Rail transport as one of the most important means of transport, has played an important role in the

transport industry in China. With Chinese rapid economic development, the railway lines and passengers

have been increasing year by year in the country. With such a huge customer base, buying train tickets

problem has been very prominent. The electronic commerce could solve the problem of railway ticketing.

Introduced a new online ticketing system is not only technological innovation, but also will improve the

railway services, to a certain extent, solve the difficult problem of railway ticketing

The railway is one of the vital modes of transport in the world. These days, we see trains for long and short-distance travel, which provides a more relaxed lifestyle. The Online Ticket Reservation System will facilitate reserving the train tickets to travel from a particular source to the destination. These days the travelers can book the rail tickets online through Train Ticket Booking System effectively than holding up in long lines to acquire the ticket. This framework should incorporate the name of the train, source, goal, time, date of appearance or takeoff, and so on.

This project is related to online train reservation system. This system provides a platform to the customers to book or cancel a ticket in a train and admin to provide details of all the trains available.

The admin of the system (here IRCTC) provides the details of all trains that are running between stations across the country. The details include the type of the train (Express, Shatabdi, Super-Fast, etc.), its source and destination, its route, its timings, in between stopping stations, classes present in the train etc. The fare from one station to another in a specified train is also provided by the admin. A customer can login to his/her account and can select the source and destination stations for a specified date. Railway System provide the trains that travel through the selected stations are listed along with the availability of seats. From these, the customer can select a train, class that he intends to travel in. Passenger can Cancel the Tickets he booked and the money will be refunded to him by Railway System. If the seats are fully booked for a selected train and class, no availability status should be conveyed to the customer. Announcements can be made by the admin on the any changes in the status of trains (delay, cancelled, etc.).

**Actors:**

Passenger and Admin (Primary Actors)

Railway System (Secondary Actor)

**Task List :**

**Passenger :**

Login / Register to Book tickets Online.

Enquire Ticket Availability and Train Details.

Pay Fare Amount and Book Ticket by Filling Form.

Can Book the Food if Required.

Cancel Ticket and get Refund.

**Admin :**

Login through his Id into System for any changes to make.

Provides the details of the trains across the Country.

Make announcements about Status of the trains.

**Railway System :**

Availability of Seats for Selected train .

Refunds the Money when ticket gets cancelled by passenger

**Use case documents :**

**Use Case 1 :**

Use Case Name : User Login

Description : A customer if he is new to website then he registers an account or else he logins into the website(Railway System) with his proper credentials . Admin Logins to the system through his Id and Password.

Actors :

1.Customer.

2. Railway System.

3. Admin

Trigger : Customer/Admin logins into the website if his credentials match with the system.

Preconditions:

1. Customer/Admin should have Active Internet
2. Customer Should have any Identity Proof(Aadhar).

Basic Flow :

1.If a Customer is new to the website , he need to register by providing details like Name ,gender, mobile number , mail id , DOB , Id Proof (Aadhar no), Username and Password.

2.If Customer already having an account can login by providing username and password .

3.Railway system checks the credentials provided by customer are valid or not .If they are valid he is logged into the website else asked to provide correct username/password.

4.Admin need to provide his id with password to login into the website . if they are valid he is logged into the website else asked to provide correct username/password by Railway System.

Alternatives : User Can Change the Password by Clicking on Forgot Password through OTP of Registered Mobile Number / Mail ID

Exceptions : no failure condition.

Level : User Goal.

Post Conditions : User Successfully Logged into the Website .

Stakeholders : Nil

**Use Case 2 :**

Use Case Name : Train Enquiry

Description :

Customer enquires the train and seat availability on the website provided by the admin and railway system.

Actors :

1.Customer.

2. Railway System.

3. Admin

Trigger : Customer gets to know the availability of trains between two selected stations on Specific date provided by Admin and availability of Seats Provided by Railway system.

Preconditions:

1. Customer/Admin is Logged into the website with active internet.

Basic Flow :

1. Customer Selects the Starting and Destination stations on a specific date.
2. Admin provides the trains available in these routes with fare price and travel time .
3. Customer selects the Train of his Choice.
4. Railway System provides the availability of seats for a selected train with class.

Alternatives : If there is no availability of seats for a selected train the customer needs to change the date of journey or select other available trains.

Exceptions : If no train available to travel for selected starting and destinations stations then he cant travel through train .

Level : User Goal.

Post Conditions : Customer gets to know the details of required train.

Stakeholders : Nil

**Use Case 3 :**

Use Case Name : Ticket Booking and Cancellation.

Description:

Customer books the tickets for selected train and class by paying the fare to railway system. Railway System refunds the money paid by customer when he cancels the booked tickets.

Actors :

1.Customer.

2. Railway System.

3. Admin

Trigger : Ticket will be booked by Customer by paying the ticket fare for selected seats in the train and amount will be refunded by Railway System if ticket is Cancelled by the Customer.

Preconditions:

1. Customer/Admin is Logged into the website with active internet.
2. Customer Should have enough Money to Pay Ticket Fare.

Basic Flow :

1.Customer Selects the Class and Seats for selected train .

2.Provides the name and age of Passenger for each seat selected .

3.Fare will be Calculated According to the selected class and no.of seats by the Railway System and Displayed to the Customer.

4.Customer needs to do the Payment to Book the tickets through UPI/Debit or Credit/Netbanking.

5.After the Completion of Payment . Ticket Confirmation message and E-ticket will be sent to the registered Mobile number and email id.

6. If the customer wants to cancel the booked tickets he need to provide a reason and click on cancel tickets and confirm.

7.Then the Railway System Refunds the money payed by the customer to his bank account within a working day.

Alternatives :

NIL

Exceptions :

If a Failure Occurs During the Payment ,Customer needs to select the seats again and try to pay again .

If ticket amount is not refunded after cancellation, Can Contact to provided Support Mail .

Level : User Goal.

Post Conditions : Customer books/cancels the tickets Successfully.

Stakeholders : Indian Railways.

**Use Case 4 :**

Use Case Name : Food Order

Description : Customer Can Choose yes for if he wants to order food and No for if he doesn’t wants to order food while travelling in train. He needs to select food items and restaurant to order and also provide his train number with class and his seat to deliver at selected station.

Preconditions :

1. Customer’s train ticket is confirmed by railway system.
2. Customer should have active internet to order food.

Actors:

1.Customer.

2.Delivery Boy.

3.Railway System.

Basic Flow :

1.Customer after his ticket is confirmed, he can choose yes or no for the question **Is Food required ?** on the website.

2.If Customer Chooses yes then he needs to enter his train number , class and also his seatno .

3.Then He needs to select a pickup station at where he can receive the food he orders from selected restaurant at the station.

4.Railway System Generates the Total Bill and Customer needs to pay amount in advance To Book the Food Order.

5.After the Confirmation of Food Order Customer receives a Message From Railway System to His Registered Mobile Number and EmailID .

6.The message Consists the Details of Delivery Boy and his phone number through which customer can contact to deliver the food at his seat no in the train.

Alternatives :

1. The Customer Can even Book The Food Order While travelling in the Train On the Date of Journey.

Exceptions :

1. Customer Cannot Order His Food When He is at Pickup Station.
2. If Delivery Boy Cannot Reach the Customer On time and Misses the train then the Amount Paid By Customer will be refunded back by Railway system.

Level : User Goal.

Post Conditions:

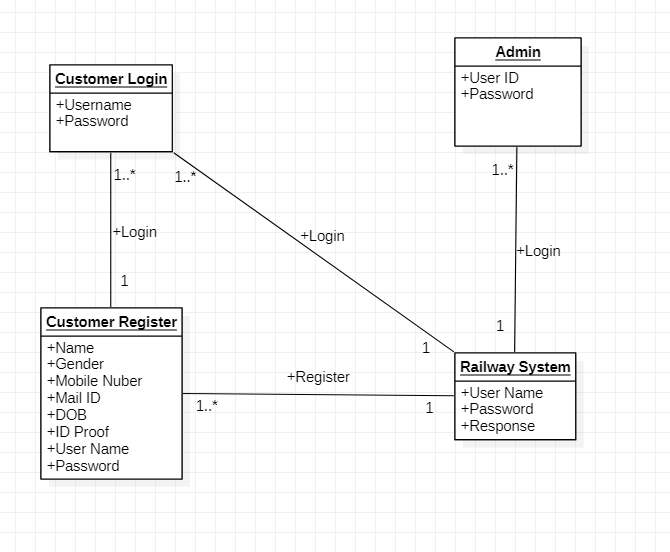
1.The Customer receives Food he ordered in Pickup station at his seat in the train From Delivery Boy.

Stake Holders :

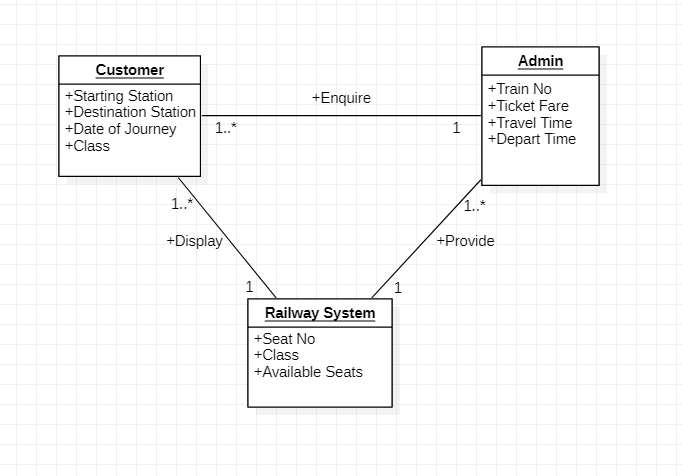
1. Delivery Boy
2. Restaurant

Object Diagrams :

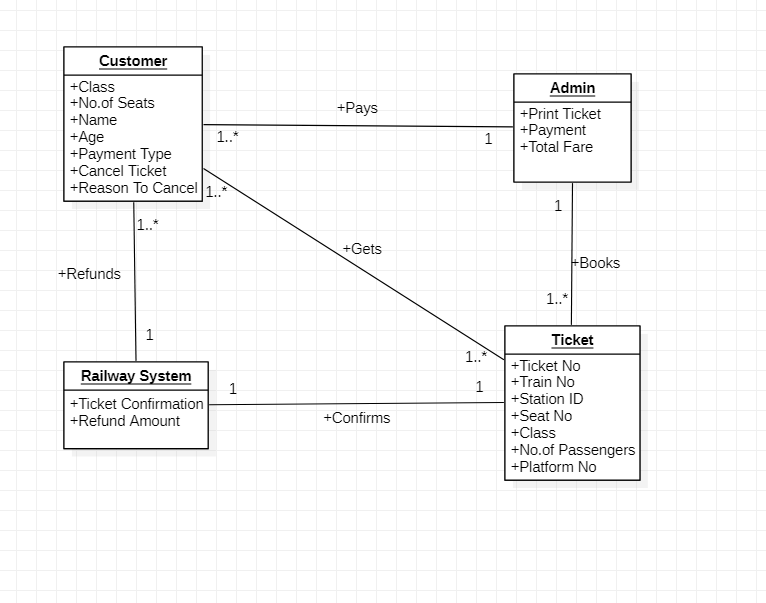
UseCase1 :



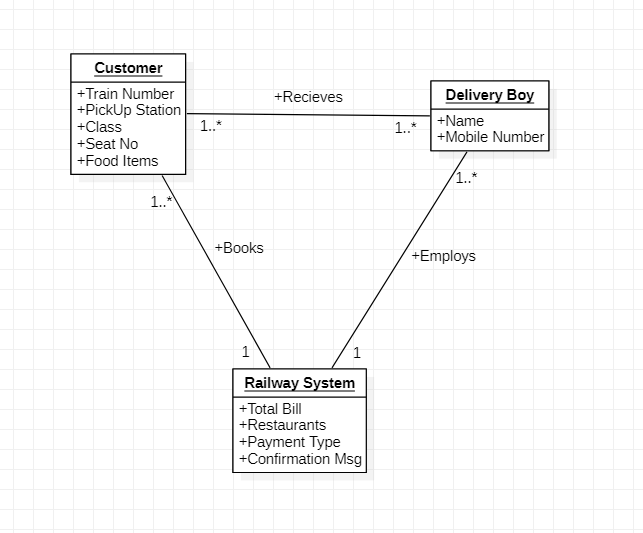
UseCase2 :



UseCase3 :

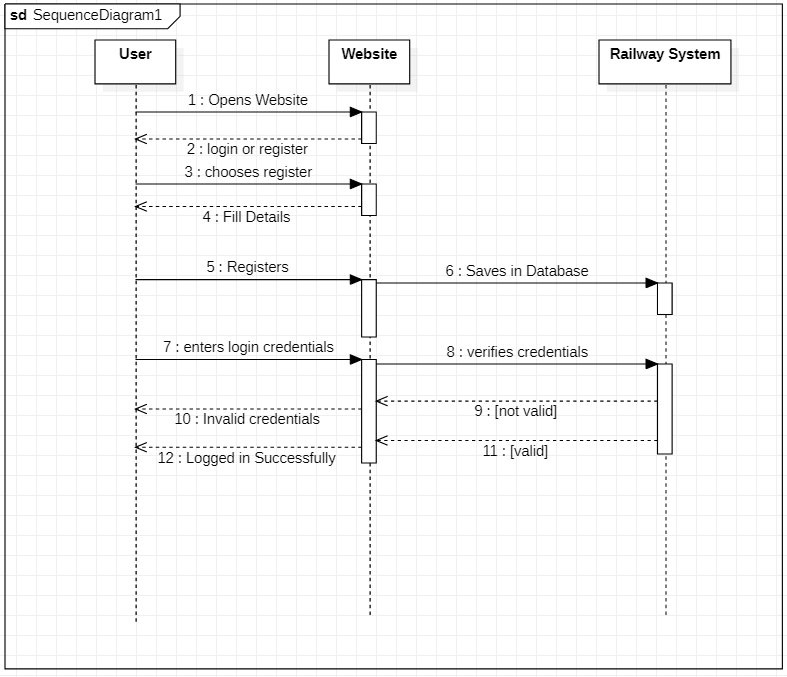


UseCase4 :

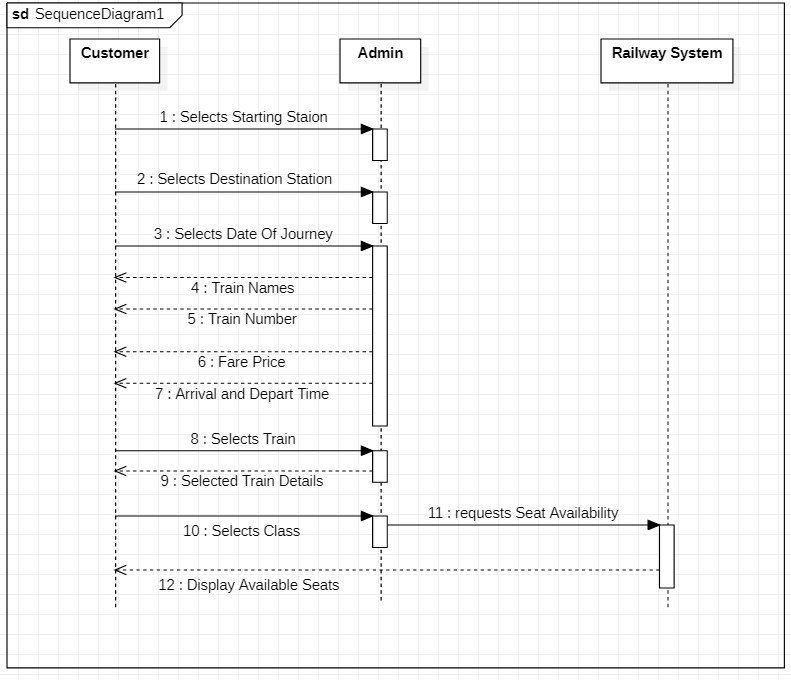


Time Sequence Diagrams :

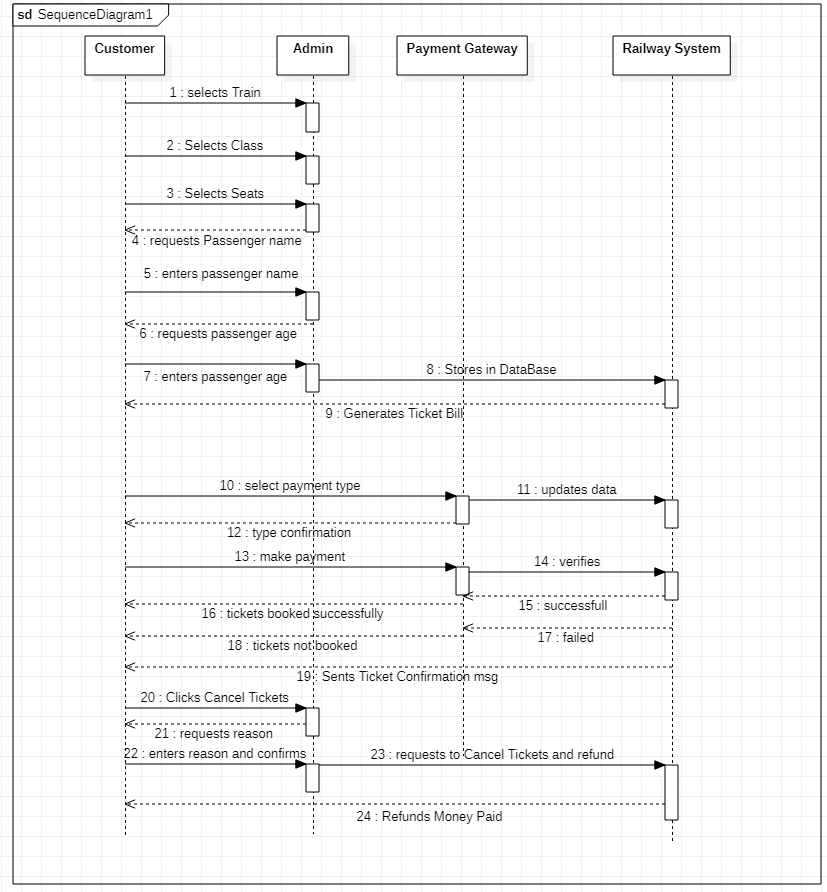
UseCase1 :



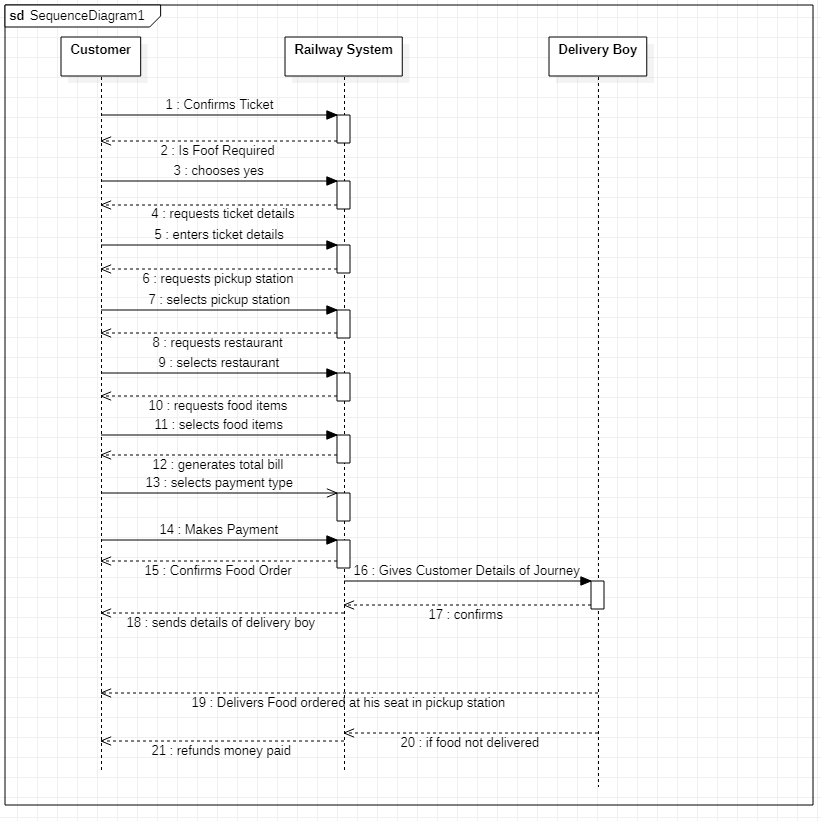
UseCase 2 :



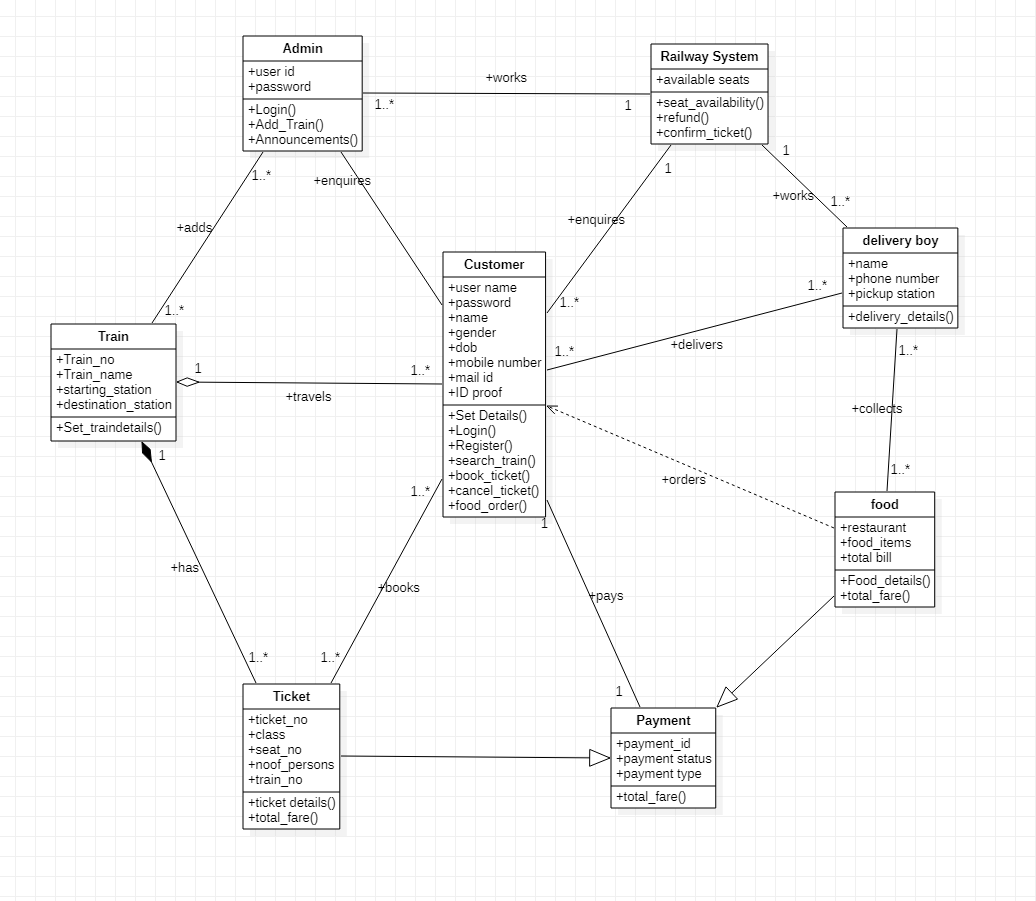
**UseCase 3 :**

****

UseCase4 :



Class Diagram :



**A.java**

**package** IndianRailways;

**import** java.io.IOException;

**public** **class** A {

**static** String *user*,*password*,*yn*,*search*,*again*,*choice*;

**static** **int** *to*=0,*y*=1,*z*=0,*end*=0,*r*=1,*x*;

**static** **int** *available*[] = **new** **int**[6];

**static** **int** *ticketI*[][] = **new** **int** [100][3];

**static** String *ticketS*[][] = **new** String[100][3];

**static** **double** *ticketD*[][] = **new** **double** [100][3];

**static** **double** *pay*[] = **new** **double**[20];

**static** **double** *change*[] = **new** **double**[20];

**public** **void** Passenger1() **throws** IOException

{

};

**void** Fare1()**throws** IOException

{

};

**void** Ticket1()**throws** IOException

{

};

**void** Exit1()

{

};

}

**B.java**

**package** IndianRailways;

**public** **interface** B {

**public** **void** getFood();

**public** **void** getSeats();

}

**IndianRailways.java**

**package** IndianRailways;

**import** java.io.\*;

**public** **class** IndianRailways **extends** A{

**public** **static** **void** main(String args[]) **throws** IOException {

BufferedReader in = **new** BufferedReader(**new** InputStreamReader(System.***in***));

**for**(**int** i=1;i<4;){

System.***out***.print("Enter Username: ");

*user* = in.readLine();

System.***out***.print("Enter Password: ");

*password* = in.readLine();

//"available[]" is the SEAT AVAILABLE//

//store 20 seats every destination [1-5]//

**for**(**int** o=1; o<=5; o++){

*available*[o]=20;

}

//if user and password are correct proceed to MAIN MENU//

**if**(*user*.equals("1") && *password*.equals("1")){

**for**(**int** x=1; x==1;){

//the MAIN MENU//

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.***out***.println("\* RAILWAY RESERVATION SYSTEM \*");

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.***out***.println("\* [1] Destination \*");

System.***out***.println("\* [2] Passengers \*");

System.***out***.println("\* [3] Billing \*");

System.***out***.println("\* [4] View \*");

System.***out***.println("\* [5] Food\_Order \*");

System.***out***.println("\* [6] Exit \*");

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

**for**(x=1; x==1;){

System.***out***.print("ENTER CHOICE: ");

*choice*=in.readLine();

//if CHOICE is "1" display the DESTINATION//

**if** (*choice*.equals("1")){

journey j=**new** journey();

System.***out***.println(j **instanceof** journey);

}

//if CHOICE is "2" proceed to Ticket Booking//

**else** **if** (*choice*.equals("2")){

A p=**new** Passenger();

p.Passenger1();

System.***out***.println(p **instanceof** Passenger);

}

**else** **if** (*choice*.equals("3")){

A f=**new** Fare();

f.Fare1();

System.***out***.println(f **instanceof** Fare);

}

**else** **if** (*choice*.equals("4")){

A t=**new** Ticket();

t.Ticket1();

System.***out***.println(t **instanceof** Ticket);

}

**else** **if** (*choice*.equals("5")){

Food\_Order f=**new** Food\_Order();

f.getFood();

System.***out***.println(f **instanceof** Food\_Order);

}

**else** **if**(*choice*.equals("6")){

A e =**new** Exit();

e.Exit1();

System.***out***.println(e **instanceof** Exit);

}

**else**{

System.***out***.println("Invalid Input!");

x=1;

}

**for**(*y*=1; *y*==1;){

**if**(*end*==1){

**break**;

}

System.***out***.print("Do you want another transaction? [Y/N]: ");

*yn* = in.readLine();

**if** (*yn*.equalsIgnoreCase("y")){

x=1;

*y*=0;

}

**else** **if** (*yn*.equalsIgnoreCase("n")){

System.***out***.println("\nThank You!!!");

x=0;

**break**;

}

**else**{

System.***out***.println("Invalid Input!!!");

*y*=1;

}

}

}

}

i=4;

}

**else**{

System.***out***.println("\nInvalid user or password!\n");

i++;

}

}

}

}

journey.java

**package** IndianRailways;

**public** **class** journey **extends** A {

journey()

{

//the DESTINATION DETAILS//

//display the "Destination", every destination "Fare", and the "Seat" available//

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.***out***.println("\* DESTINATION |FARE| SEAT \*");

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.***out***.println("\* 1.)CHENNAI | 600 | "+*available*[1]+" \*");

System.***out***.println("\* 2.)HYDERABAD | 170 | "+*available*[2]+" \*");

System.***out***.println("\* 3.)VIZAG | 150 | "+*available*[3]+" \*");

System.***out***.println("\* 4.)GOA | 200 | "+*available*[4]+" \*");

System.***out***.println("\* 5.)TIRUPATI | 250 | "+*available*[5]+" \*");

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

System.***out***.println("PWD, STUDENT, & SENIOR CITIZEN with 20% DISCOUNT!!!\n");

*x*=0;

}

}

Passenger.java

**package** IndianRailways;

**import** java.io.\*;

**public** **class** Passenger **extends** A {

**int** print=1;

@Override

**public** **void** Passenger1() **throws** IOException

{

BufferedReader in = **new** BufferedReader(**new** InputStreamReader(System.***in***));

//display first the Destination Details//

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.***out***.println("\* DESTINATION |FARE| SEAT \*");

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.***out***.println("\* 1.)CHENNAI | 600 | "+*available*[1]+" \*");

System.***out***.println("\* 2.)HYDERABAD | 170 | "+*available*[2]+" \*");

System.***out***.println("\* 3.)VIZAG | 150 | "+*available*[3]+" \*");

System.***out***.println("\* 4.)GOA | 200 | "+*available*[4]+" \*");

System.***out***.println("\* 5.)TIRUPATI | 250 | "+*available*[5]+" \*");

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

System.***out***.println("PWD, STUDENT, & SENIOR CITIZEN with 20% DISCOUNT!!!\n");

**if**((*available*[1]==0)&&(*available*[2]==0)&&(*available*[3]==0)&&(*available*[4]==0)&&(*available*[5]==0)){

System.***out***.println("Sorry, We don't have available seats for all Destination!");

*x*=0;

}

//inputing of Passenger's Name//

**else**{

**for**(*x*=1; *x*==1;){

System.***out***.print("\nENTER PASSENGER'S NAME: ");

*ticketS*[*z*][0] = in.readLine();

*x*=0;

//if Passenger's Name already used, display error and go back to Inputing//

**for**(**int** l=0; l<*z*; l++){

**if**(*ticketS*[l][0].equalsIgnoreCase(*ticketS*[*z*][0])){

System.***out***.println("Sorry, Passenger's name have already used!");

*x*=1;

}

}

}

//inputing of Destination//

//integers Only [1-5]//

**for**(*x*=1; *x*==1;){

System.***out***.print("ENTER DESTINATION [number]: ");

*to* = Integer.*parseInt*(in.readLine());

//if Inputed integers are "<1" or ">5", display error and go back to Inputing//

**if**(*to*<1 || *to*>5){

System.***out***.println("Invalid Input!");

*x*=1;

}

//if available seat is eqaul to "Zero", display error and go back to Inputing//

**for**(**int** d=1; d<=5; d++){

**if**(*to*==d){

**if**(*available*[*to*]==0){

System.***out***.println("Sorry, We don't have available seat!");

*x*=1;

}

*x*=0;

}

}

}

//convert the integer to string//

String dest[] = { " ", "CHENNAI", "HYDERABAD", "VIZAG", "GOA", "TIRUPATI"};

**double** fare[] = { 0,600,170,150,200,250};

//converted integer to string, transfer to storage array//

*ticketS*[*z*][1] = dest[*to*];

*ticketD*[*z*][0] = fare[*to*];

//inputing for Number of Passenger's//

**for**(*x*=1; *x*==1;){

System.***out***.print("HOW MANY PASSENGERS ARE YOU?: ");

*ticketI*[*z*][0] = Integer.*parseInt*(in.readLine());

//subtract the available seat by the the number inputed//

**for**(**int** p=1; p<=5; p++){

**if**(*to*==p){

print=1;

*available*[*to*] = *available*[*to*]-*ticketI*[*z*][0];

//if the subtracted available seat is "<0", display error//

//add the inputed number to the subtracted seat, to back the original seat//

//display the available seat and back to the inputing//

**if**(*available*[*to*]<0){

System.***out***.print("Sorry, We don't have seat available for " +*ticketI*[*z*][0] +" person\n");

*available*[*to*] = *available*[*to*]+*ticketI*[*z*][0];

System.***out***.print("We only have " +*available*[*to*] +" seat available\n");

*x*=1;

print=0;

}

**else**{

*x*=0;

}

}

}

}

//inputing for Number of Discounted Passenger's//

**for**(*x*=1;*x*==1;){

System.***out***.print("HOW MANY PASSENGERS HAVE DISCOUNT?: ");

*ticketI*[*z*][1] = Integer.*parseInt*(in.readLine());

**if**(*ticketI*[*z*][1]>*ticketI*[*z*][0]){

System.***out***.println("Invalid Input!");

System.***out***.println("No. of Passengers are only " +*ticketI*[*z*][0] +"!");

*x*=1;

}

**else**{

**break**;

}

}

//print out of passengers details....

**if**(print==1){

System.***out***.println("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.***out***.println("\* PASSENGER'S DETAILS \*");

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.***out***.println("PASSENGER'S NAME: " + *ticketS*[*z*][0]);

System.***out***.println("PASSENGER'S DESTINATION : " + *ticketS*[*z*][1]);

System.***out***.println("FARE PRICE: " + *ticketD*[*z*][0]);

System.***out***.println("NO. OF PASSENGERS: " + *ticketI*[*z*][0]);

System.***out***.println("NO. OF PASSENGERS WITH DISCOUNT: " + *ticketI*[*z*][1]);

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

*ticketS*[*z*][2]="0";

**double** discount=(*ticketD*[*z*][0]-(*ticketD*[*z*][0]\*0.2))\**ticketI*[*z*][1];

*ticketD*[*z*][2]= ((*ticketI*[*z*][0]-*ticketI*[*z*][1])\**ticketD*[*z*][0])+discount;

*x*=0;

}

*z*++;

}

}

}

Fare.java

**package** IndianRailways;

**import** java.io.\*;

**public** **class** Fare **extends** A {

**void** Fare1()**throws** IOException

{

BufferedReader in = **new** BufferedReader(**new** InputStreamReader(System.***in***));

**for**(*x*=1; *x*==1;){

System.***out***.print("ENTER PASSENGER'S NAME: ");

*search* = in.readLine();

**int** s=1;

**for**(**int** b=0;b<*z*;b++){

**if**(*search*.equalsIgnoreCase(*ticketS*[b][0])){

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.***out***.println("\* PASSENGER'S DETAILS \*");

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.***out***.println("PASSENGER'S NAME: " + *ticketS*[b][0]);

System.***out***.println("PASSENGER'S DESTINATION : " + *ticketS*[b][1]);

System.***out***.println("FARE PRICE: " + *ticketD*[b][0]);

System.***out***.println("NO. OF PASSENGERS: " + *ticketI*[b][0]);

System.***out***.println("NO. OF PASSENGERS WITH DISCOUNT: " + *ticketI*[b][1]);

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*");

s=0;

*x*=0;

**if**(*ticketS*[b][2].equals("x")){

System.***out***.println("Passenger's Already Paid!");

*x*=0;

}

**else**{

*ticketS*[b][2]="x";

**for**(*x*=1; *x*==1;){

System.***out***.println("\nPASSENGER'S TOTAL FARE: "+*ticketD*[b][2]);

System.***out***.print("ENTER AMOUNT TO PAY: ");

*pay*[b] = Double.*parseDouble*(in.readLine());

*change*[b]=*pay*[b]-*ticketD*[b][2];

**if**(*change*[b]<0){

System.***out***.println("Invalid Input!");

*x*=1;

}

**else**{

System.***out***.println("CHANGE: "+*change*[b]);

System.***out***.println("");

*x*=0;

}

}

}

}

}

**if** (s==1){

System.***out***.println("\nPASSENGER'S NAME NOT FOUND!\n");

**for**(**int** q=1; q==1;){

System.***out***.print("Do you wish to continue with this transaction? [Y/N]: ");

*again*=in.readLine();

**if**(*again*.equalsIgnoreCase("y")){

q=0;

}

**else** **if** (*again*.equalsIgnoreCase("n")){

q=0;

*x*=0;

}

**else**{

System.***out***.println("\nInvalid input!\n");

}

}

}

}

}

}

Ticket.java

package IndianRailways;

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStreamReader;

public class Ticket extends A {

void Ticket1()throws IOException

{

BufferedReader in = new BufferedReader(new InputStreamReader(System.in));

for(int sx=1; sx<=3;){

System.out.print("SEARCH PASSENGER'S NAME: ");

search = in.readLine();

int s=1;

for(x=0; x<=z; x++){

if(search.equalsIgnoreCase(ticketS[x][0])){

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.out.println("\* PASSENGER'S DETAILS \*");

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.out.println("PASSENGER'S NAME: " + ticketS[x][0]);

System.out.println("PASSENGER'S DESTINATION : " + ticketS[x][1]);

System.out.println("FARE PRICE: " + ticketD[x][0]);

System.out.println("NO. OF PASSENGERS: " + ticketI[x][0]);

System.out.println("NO. OF PASSENGERS WITH DISCOUNT: " + ticketI[x][1]);

System.out.println("TOTAL FARE PRICE: " + ticketD[x][2]);

if(ticketS[x][2].equals("x")){

System.out.println("PAY: " +pay[x]);

System.out.println("CHANGE: " +change[x]);

System.out.println("STATUS: PAID");

}

else{

System.out.println("STATUS: NOT PAID");

}

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*");

s=0;

sx=4;

}

}

if (s==1){

System.out.println("Passenger's Name not found!");

sx++;

}

}

}

}

**Food\_Order.java**

**package** IndianRailways;

**import** java.util.Scanner;

**public** **class** Food\_Order **extends** A **implements** B {

**static** Scanner *scan*= **new** Scanner(System.***in***);

**int** total;

**static** String[] *foodlist*={"1) Chicken Biryani ","2) Meals","3)Chapathi","4)Vada","5)Samosa"};

**static** **int** *number*;

**public** **void** getFood(){

System.***out***.println("Please select a Food Item");

**for**(**int** i=0;i<*foodlist*.length;i++){

System.***out***.println(*foodlist*[i]);

}

*number*=*scan*.nextInt();

System.***out***.println("Your Food Order is "+*foodlist*[*number*-1].substring(3,*foodlist*[*number*-1].length()));

getSeats();

}

**public** **void** getSeats(){

System.***out***.println("Please Choose the Quantity of Selected Food Item");

String seats=(*scan*.next());

**int** i=Integer.*parseInt*(seats);

System.***out***.println("Thank you,\nYou have selected "+seats+" Quantity of "+*foodlist*[*number*-1].substring(3,*foodlist*[*number*-1].length()) );

**if**(*number*==1)

{

total=150\*i;

}

**else** **if**(*number*==2)

{

total=120\*i;

}

**else** **if**(*number*==3)

{

total=60\*i;

}

**else** **if**(*number*==4)

{

total=50\*i;

}

**else** **if**(*number*==5)

{

total=20\*i;

}

**else**

{

System.***out***.println("Invalid choice");

}

System.***out***.println("Total Fare :"+total);

}

}

**Exit.java**

**package** IndianRailways;

**public** **class** Exit **extends** A{

**void** Exit1()

{

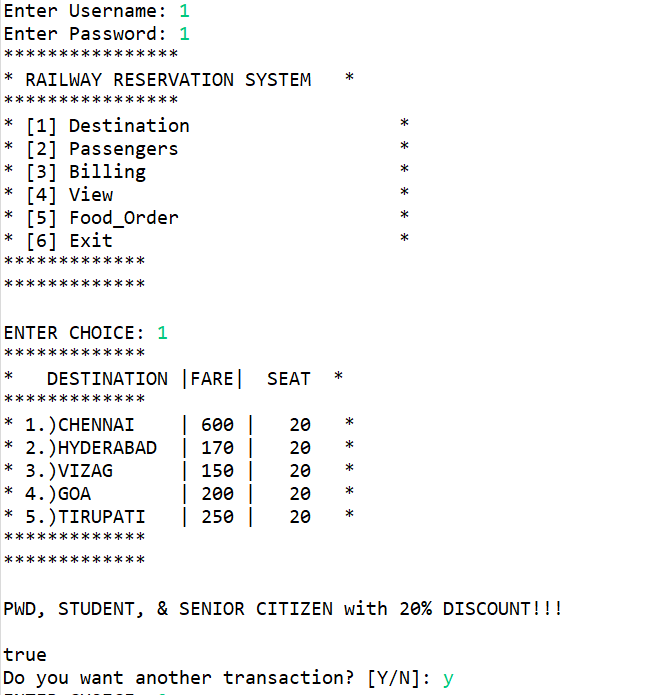
*end*=1;

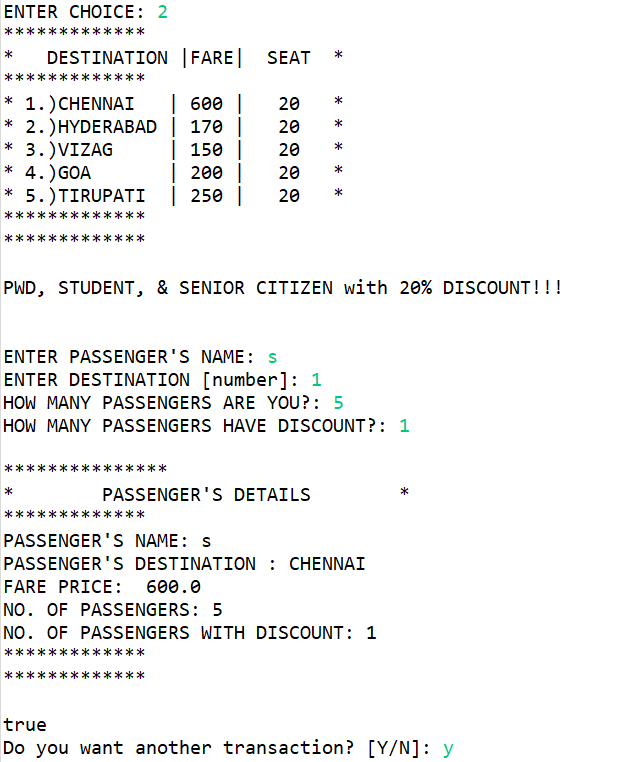
*x*=0;

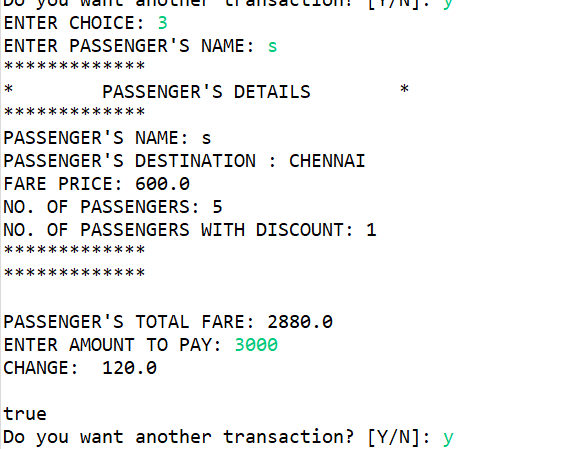
System.***out***.println("Thank You!");

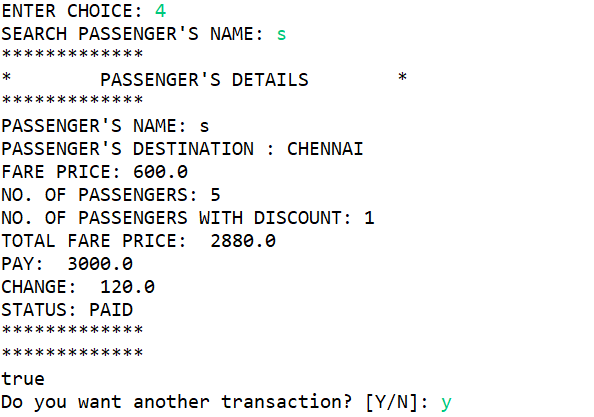
}

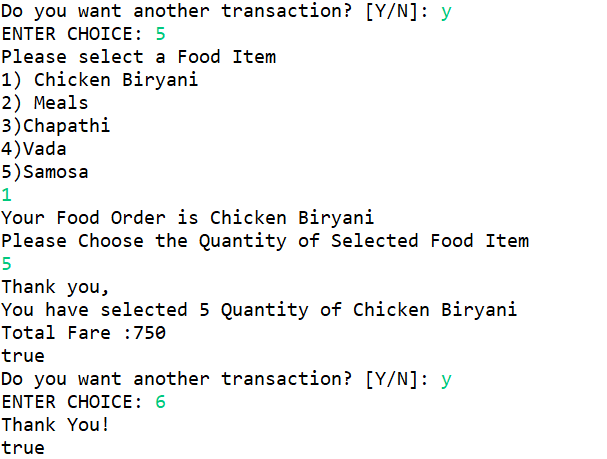
}











**Task done :**

|  |  |  |
| --- | --- | --- |
| **Roll no.** | **name** | **Task done** |
| cb.en.u4cse20310 | Boppudi Lokesh | Use case 1-document/ object model/ time sequence diagram,class diagram |
| cb.en.u4cse20312 | Chintha Pavan | Use case 3-document/ object model/time sequence diagram, class diagram |
| cb.en.u4cse20323 | Kammara Sai Sreeram | Use case 4-document/ object model/ time sequence diagram, class diagram |
| cb.en.u4cse20347 | Prathipati Sri Surya | Use case 2-document/ object model/ time sequence diagram, class diagram |