



## **Open Ended Activity Report**

**On**

### **Movie Ticket Booking System**

**Object Oriented Programming with C++ (19ECSC204)**  
**Object Oriented Programming with C++ Lab (18ECSP203)**

Submitted by

<b>Name</b>	<b>Roll no</b>	<b>SRN</b>
Apoorva Jinde	044	01FE18BCS044
Ashutosh Pratap Singh	053	01FE18BCS053

**Team Number: 4A18**

Faculty In-charge:

Prof. K. M. M. Rajashekharaiiah

**SCHOOL OF COMPUTER SCIENCE & ENGINEERING**

HUBLI – 580 031 (India).

Academic year 2019-20

<b>1.</b>	<b>Introduction</b>	
	1.1	Overview of the problem statement
	1.2	Features of Application 1.2.1 Booking Tickets 1.2.2 Adding New Shows 1.2.3 Ticket Cancellation
<b>2.</b>	<b>Design</b>	
	2.1	Class Diagram
	2.2	Class Diagram Description 2.2.1 Class ServiceProvider 2.2.2 Class Theatre 2.2.3 Class Show 2.2.3.1 Class Movie 2.2.3.2 Class LiveEvent 2.2.4 Class Session 2.2.5 Class Ticket 2.2.6 Class Auditorium 2.2.7 Class Seat 2.2.7.1 Class Platinum 2.2.7.2 Class Gold 2.2.7.3 Class Silver 2.2.8 Class Customer 2.2.9 Class Snack

		2.2.10 Class Exception
	2.3	Factory Design Pattern
<b>3.</b>	<b>Unit Test Plan</b>	
	3.1	Booking Test Case 1: Successful Booking Test Case 2: Failed Booking
	3.2	Adding Shows Test Case 3: Adding Shows To Selected Theatre
	3.3	Ticket Cancellation Test Case 4: Cancelling an available Ticket Test Case 5: Cancelling an unavailable Ticket
	3.4	Customer Registration Test Case 6: Successful Registration Test Case 7: Failed Registration
<b>4.</b>	<b>Implementation</b>	
	4.1	Results 4.1.1 Main Menu 4.1.2 Add Shows 4.1.3 View Theatres 4.1.4 View Shows 4.1.5 View and Booking Ticket 4.1.6 Cancel Ticket

## **1.1 Overview of the problem statement**

In this application we can book the seats in the multiplex service centre based on the entered location where the movie of our choice is available. We can book the snacks from the menu in the theatre so that queue is avoided during the visit. The movie has its different session time in different theatre. We are having different types of seats in each auditorium and all theatres are containing those auditoriums. Service provider is having all those theatres registered. So customer can select the desired location from service provider, he can check all theatres present in his city, all shows available in his city and can book the tickets and snacks and get the payment and ticket details easily. All exceptions are handled.

## **1.2 Features of Application**

### **1.2.1 Booking Tickets**

This application allows the user to book show tickets of his choice in a more convenient and easy way. The user will enter his location and he will get all the available theatres in his city. Then the user will be allowed to choose the theatre he wants and as soon as he chooses a theatre then he will get the list of all available shows in the theatre then he can choose the show accordingly. User will be given an option for snacks as well if he chooses that then according to the show, seat and the snack selected he will be charged and the ticket will be generated.

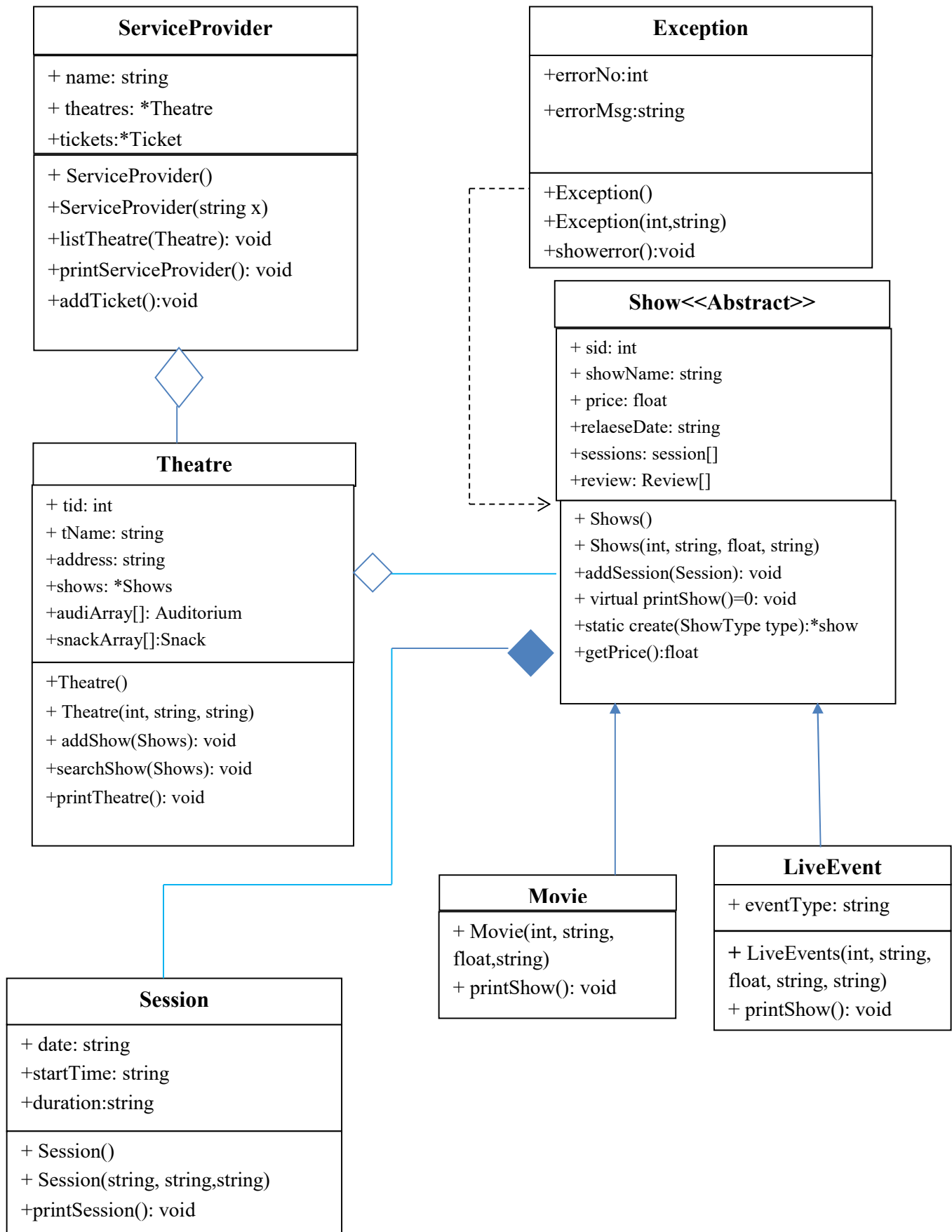
### **1.2.2 Add New Shows**

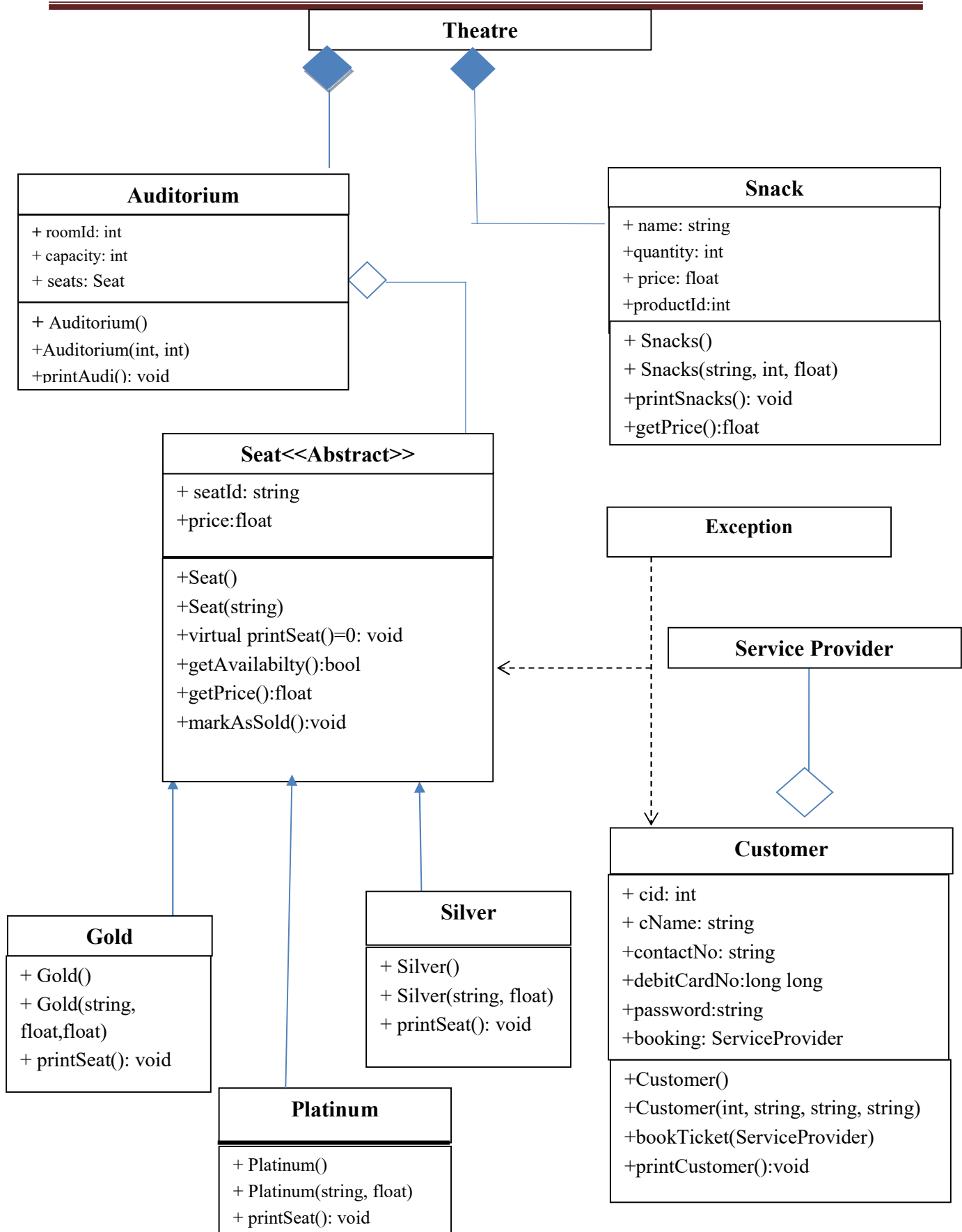
This will allow the administrator to add show in his theatre. The user can easily select the type of show he wants to add, then he will be asked to enter the details of the show like date, time and so on. He can set the price of the movie and all the details added by the administrator will be shown to the customer for ticket booking.

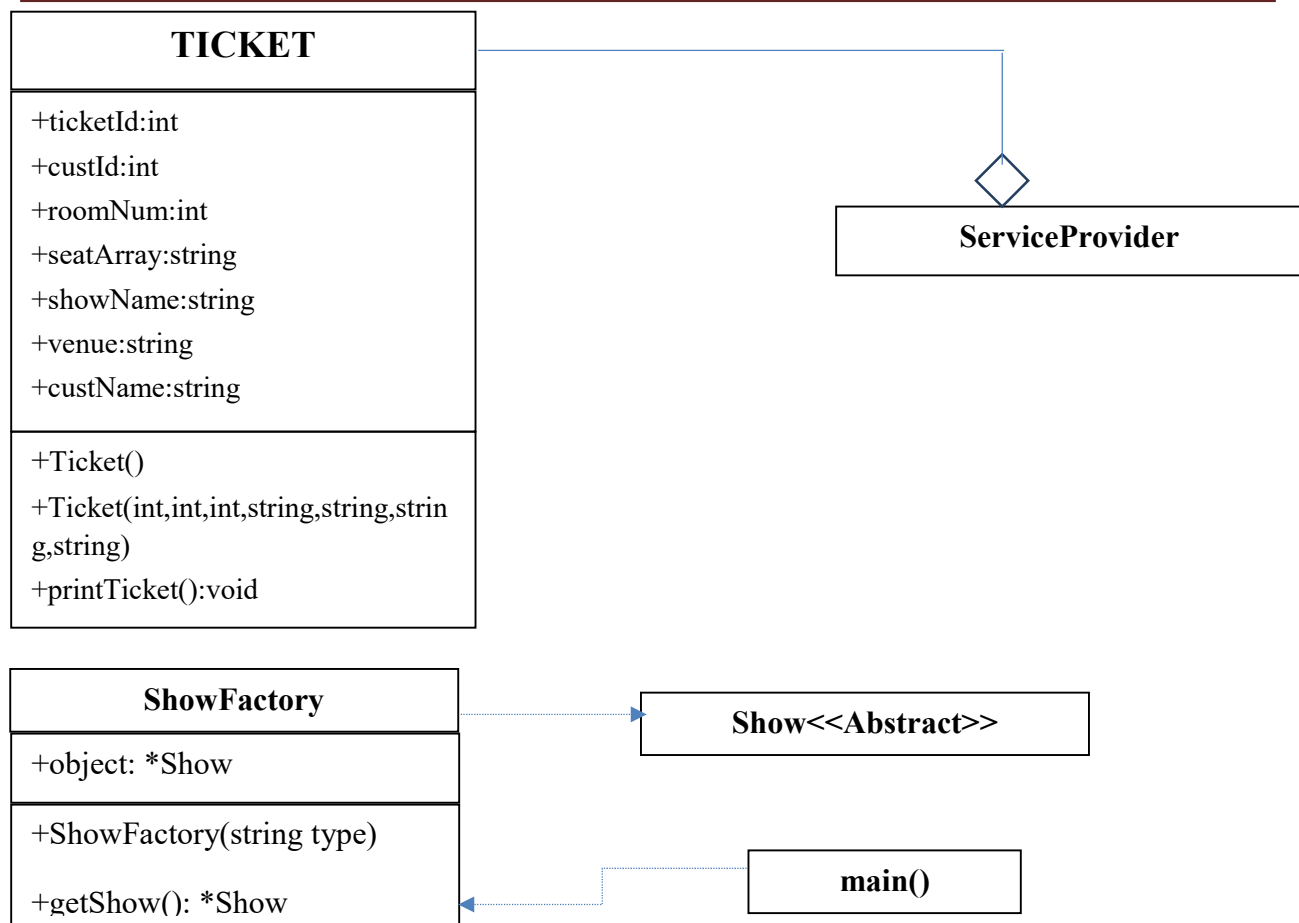
### **1.2.3 Ticket Cancellation**

We will also provide the facility of cancellation of ticket. The user will be asked about his ticket Id then as per the ticket ID the ticket will get cancelled and the user will get back his money accordingly and that seat will be visible to other customers then.

## **2.1 Class Diagram**

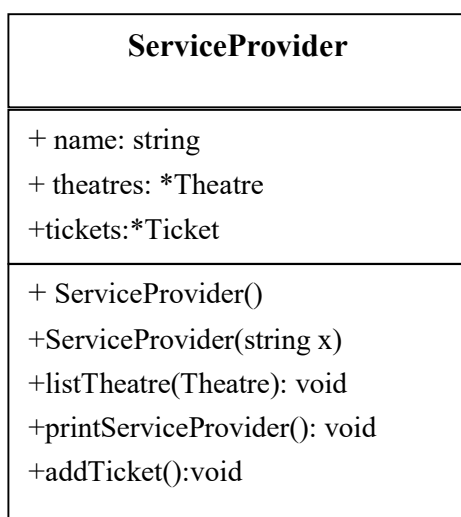






## 2.2 Class Diagram description

### 2.2.1.ServiceProvider



This is the class from which customer can access each and every feature of ticket booking. In this class we are having all theatres registered to it, it keeps the record of all the tickets booked and we can list all the theatres associated with the booking system, as the customer accesses the service provider he can book the ticket and all the tickets are stored.

### 2.2.2.Theatre

This class is associated with many other classes and helps the user in many ways. This class contains all the shows associated with each theatre. Admin can use this class to select his theatre and thus he can update and add the shows on daily basis thus those shows will be visible to the customers. The class also contains an auditorium array which stores the rooms associated with each theatre and thus by accessing that array we show the seat map to the customers while ticket booking from service provider. As the theatre also provides the feature of snack booking so it also contains an array of snack which stores the snacks and shows the user while booking. This class has functions for printing the theatres, rooms, and all the seat maps associated with it.

Theatre
+ tid: int + tName: string +address: string +shows: *Shows +audiArray[]: Auditorium +snackArray[]:Snack
+Theatre() + Theatre(int, string, string) + addShow(Shows): void +searchShow(Shows): void +printTheatre(): void

### 2.2.3. Show

Show is an abstract class which from which two classes are derived i.e. LiveEvent and Movie. With the help of this class we can create new shows and we can get the price of any category of the show which helps us to calculate the ticket price while the customer books the ticket. Print function for show is pure virtual function. This class is also having session as show owns session. Session stores the timing and date of each type of show. This class is also associated to a



factory design that is showFactory through which we create different types of show as per the user convenience.

<b>Shows&lt;&lt;Abstract&gt;&gt;</b>
+ sid: int + showName: string + price: float +relaeseDate: string +sessions: session[]
+ Shows() + Shows(int, string, float, string) +addSession(Session): void + virtual printShow()=0: void +static create(ShowType type):*show +getPrice():float

#### 2.2.3.1. Movie

Movie is the derived class from base class show which is an abstract class having same features as that of base class. This class is having function to display the movie created.

<b>Movie</b>
+ Movie(int, string, float,string) + printShow(): void

#### 2.2.3.2. Live Event

LiveEvent is the derived class from base class show which is an abstract class having same features as that of base class. This class is having function to display the event created

<b>LiveEvent</b>
+ eventType: string
+ LiveEvents(int, string, float, string, string) + printShow(): void

#### 2.2.4.Session

Session is a class which stores the timings and date of a particular show.

<b>Session</b>
+ date: string +startTime: string +duration:string
+ Session() + Session(string, string,string) +printSession(): void

#### 2.2.5.Ticket

Ticket class helps to print all the details like room number, venue, cutomer id and show details.

<b>TICKET</b>
+ticketId:int +custId:int +roomNum:int +seatArray:string +showName:string +venue:string +custName:string
++Ticket() +Ticket(int,int,int,string,string,stri ng,string) +printTicket():void

### 2.2.6. Auditorium

Auditorium is a class contained in the class theatre i.e theatre owns the auditorium. It describes about the no of seats in every particular room where the shows are held. It helps to access the seats as this class owns seats thus helps to print the details of room as well as the seats associated with respective room.

<b>Auditorium</b>
+ roomId: int + capacity: int + seats: Seat
+ Auditorium() + Auditorium(int, int) + printAudi(): void

### 2.2.7. Seat

Class Seat is used for describing the rows and columns of all the seats in their respective auditoriums. Seat is an abstract class from that class we have three derived classes Gold, Silver and Platinum. These are the three category of seats which the user can select as per his convenience. Each category is having different price.

In Seat class we have function like getPrice(), getAvailabilty(), markAsSold(), which help us to get the price, check whether the seat is available or not and to mark those seats which are booked.

<b>Seat&lt;&lt;Abstract&gt;&gt;</b>
+ seatId: string + price: float
+ Seat() + Seat(string) + virtual printSeat(): void + getAvailability(): bool + getPrice(): float + markAsSold(): void

#### 2.2.7.1. Platinum

Platinum is a class inherited from its base class Seat and has the same features as that of base class.

<b>Platinum</b>
+ Platinum() + Platinum(string, float) + printSeat(): void

#### 2.2.7.2. Gold

Gold is a class inherited from its base class Seat and has the same features as that of base class.

Gold
+ Gold() + Gold(string, float,float) + printSeat(): void

### 2.2.7.3.Silver

Silver is a class inherited from its base class Seat and has the same features as that of base class.

Silver
+ Silver() + Silver(string, float) + printSeat(): void

### 2.2.8.Customer

Customer is a class which is having the details of every customer with the help of this class we can manage ticket booking, ticket cancellation and payment based on no of seats types of seats along with the snacks taken. This class is associated with the service provider with has a relationship thus user can easily book ticket and avail all facilities quickly with the help of this class.

With this class we can even cancel class the tickets and free the seats allocated to the customer after cancellation

Customer
+ cid: int + cName: string +contactNo: string +debitCardNo:long long +password:string +booking: ServiceProvider
+Customer() +Customer(int, string, string, string) +bookTicket(ServiceProvider) +cancelTicket(int) +printCustomer():void

### 2.2.9. Snack

Snack is a class which describes information about the various food items present in respective theatres. This class is owned by class theatre class.

Snack
+ name: string +quantity: int + price: float +productId:int
+ Snacks() + Snacks(string, int, float) +printSnacks(): void +getPrice():float

### 2.2.10. Exception class:

This class is used to handle all the exception throughout the code. User cannot use the Id of the preexisting user, user has to put a valid mobile number and debit card number, if some seat is allocated already to someone the user will get a message of seat unavailable, if there is no theatre present in user's city he will get a message to try a new location.

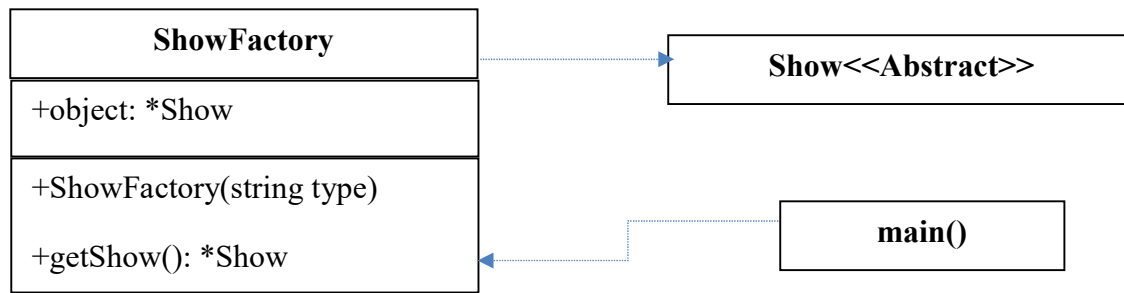
Exception
+errorNo:int +errorMsg:string
++Exception() +Exception(int,string) +showerror():void

## 2.3 Factory Design Pattern

- DEF:- The Factory Method pattern is a design pattern used to define a runtime interface for creating an object. It's called a factory because it creates various types of objects without necessarily knowing what kind of object it creates or how to create it.
- Usage :- Factory method is suitable for this scenario because Shows objects are to be created as per customer demand and the type of shows may be added of new type in the near future we need not change, so to create the objects of the required show type during the run time interface it becomes easier.

We have implement factory method to create objects of derived class. Based on user input it will create object of required subclass. Here we are having show factory for creating various types of shows based on user input i.e. the administrator can add all categories of show easily.

We have chosen this method because if in future the show category increases we can simply alter the category on our side and thus the client will not get affected by this. User can simply create new show based on his choice every time and thus can easily add the shows in theatres.



### 3. Unit Test Plan

#### 3.1 Booking

In this feature first the Customer fills his details and login credentials and then according to his current entered location list of theatres, shows, seats, and snacks will be available for him so that he can book ticket for his preferred show in a preferred theatre along with all the various facilities.

##### Test Case 1: Successful Booking

First customer enters his details then he proceeds for booking as per the available shows.

##### Sample Input:

Enter City:

Lucknow

Enter Theatre ID:

1

Enter Show ID:

1

Enter Number of Person(s):

1



Select The Seat(s):

A01

Enter 1 for Snacks

1

Enter The Snack ID:

2

Expected output:

Ticket Booked Successfully

//Ticket Details will be printed in the output

Actual Output:

```
4
Enter your ID:1
Enter your name:ASHUTOSH
Enter your mobile number:9984795100
Enter your debit card number:1234567898765432
Enter your password:12345678
Enter your city:LUCKNOW
*****
Theatres Available In Your City Are:

Theatre Name:SRS Theatre ID:1 Theatre Address:LUCKNOW

Theatre Name:CINEPOLIS Theatre ID:2 Theatre Address:LUCKNOW

Theatre Name:WAVE Theatre ID:3 Theatre Address:LUCKNOW

Theatre Name:PVR Theatre ID:4 Theatre Address:LUCKNOW

Enter the theatre id:1
The list of shows in the preferred theatre are:
Movie ID: 1
Movie Name: AVENGERS
Movie Base Price: 200
Movie Release Date: 23-MAY-2020

Show Date:30-MAY-2020
Show Slots are:9:00AM   Show Slots are:12:00PM   Show Slots are:6:00PM
Enter the show ID to be booked:1
The Available seats are:
Room ID:1 Total Seating Available:9

Seat & Price Info
A01 120 30
A02 120 30
A03 120 30
M01 160 35
M02 160 35
M03 160 35
V01 200 40
V02 200 40
V03 200 40

Enter the number of people:
```

```
Enter the number of people:2
Enter the seat id to be selected:A01
Seat Available
A02
Seat Available
Book your snacks:
Enter 1 or 0
1
Snacks available are:
Product ID:1:Name:BURGER :Quantity:1 Price:100
Product ID:2:Name:COLD DRINK :Quantity:1 Price:200
Product ID:3:Name:POPCORN :Quantity:1 Price:300
Enter the number of snacks:1
Enter the snack Id to be booked:
1
540
*****
Ticket ID:1
Venue:SRS
Audi Number:1
Customer ID:1
Customer Name:ASUTOSH
Show Name:KGF
Total Payment:540
Selected seats are:
A01 A02
*****
1.ADD SHOWS TO THE THEATRE
2.LIST OF SHOWS IN THE THEATRES
3.LIST ALL THE THEATRES
4.BOOK MOVIE TICKET
5.CANCEL TICKET
7.EXIT(0)
```

### Test Case 2: Unsuccessful Booking

Customer enters a city but there no theatre exists thus he will be asked to try some new location.

#### Sample Input:

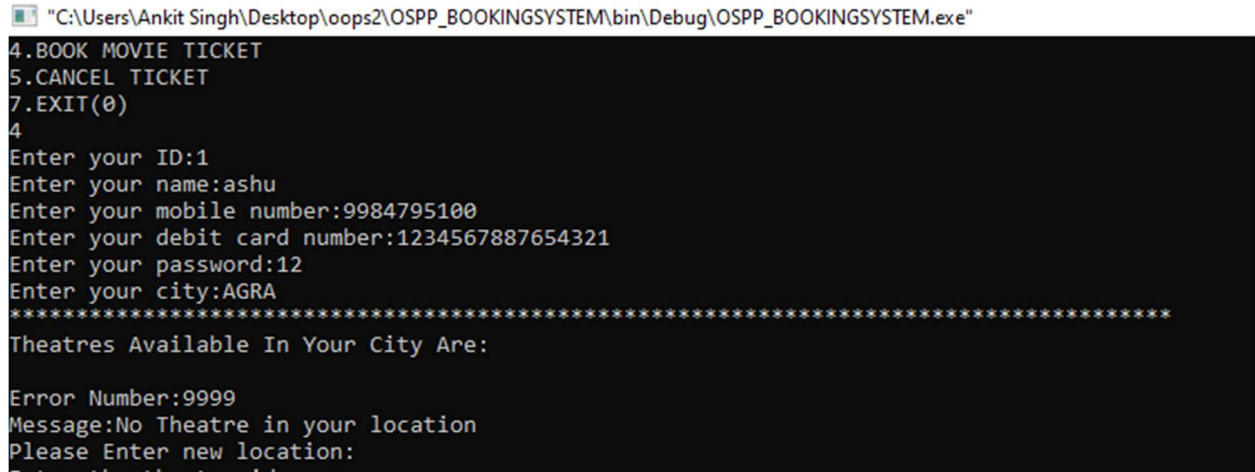
Enter City:

Agra

#### Expected Output:

No Theatre Available in Your City.

Please Try Some New Location.

**Actual Output:**

```
"C:\Users\Ankit Singh\Desktop\oops2\OSPP_BOOKINGSYSTEM\bin\Debug\OSPP_BOOKINGSYSTEM.exe"
4.BOOK MOVIE TICKET
5.CANCEL TICKET
7.EXIT(0)
4
Enter your ID:1
Enter your name:ashu
Enter your mobile number:9984795100
Enter your debit card number:1234567887654321
Enter your password:12
Enter your city:AGRA
*****
Theatres Available In Your City Are:

Error Number:9999
Message:No Theatre in your location
Please Enter new location:
```

**3.2 Adding Shows**

This option is for owner of theatres who manages shows on daily basis they can add shows for each day along with the date and time.

**Test Case 3: Adding Shows in The Selected Theatre****Sample Input:**

Enter the theatre Id:

1//Users Choice

Enter the type of show:

1 //Users Choice

Enter the Show ID:

1

Enter The Show Name:

Avengers

Enter the Base Price:

200

Enter The Release Date:

23-may-2019

Enter the Date:

12-Jun-2020

Enter Sessions:

1:00PM

2:00PM

6:00PM

Expected Output:

Details Entered Successfully

Movie Details will be shown

Actual Output:

```

*****
1.ADD SHOWS TO THE THEATRE
2.LIST OF SHOWS IN THE THEATRES
3.LIST ALL THE THEATRES
4.BOOK MOVIE TICKET
7.EXIT(0)
1
-----THEATRES-----
1.SRS LUCKNOW
2.CINEPOLIS LUCKNOW
3.WAVE LUCKNOW
4.PVR LUCKNOW
5.FUN LUCKNOW
6.PHOENIX KANPUR
7.SRS KANPUR
8.SRS BANGALORE
9.SRS DELHI
10.OASIS HUBLI
11.URBAN HUBLI
-----
Enter the theatre where you want to add the show:1
What type of Show you want to add in the theatre:
Enter the choice:
1.Movie
2.Live Event
1
Enter the Movie details:
Enter the Movie ID:
1
Enter the name of the Movie:
AVENGERS
Enter the Base Price of the Movie:
200
Enter the Release Date of the Movie:
23-MAY-2020
Enter the date:30-MAY-2020
Enter the three session slots for the movie:9:00AM
12:00PM
6:00PM
Details Entered Successfully
Movie ID: 1
Movie Name: AVENGERS
Movie Base Price: 200
Movie Release Date: 23-MAY-2020

Show Date:30-MAY-2020
Show Slots are:9:00AM   Show Slots are:12:00PM   Show Slots are:6:00PM
*****
1.ADD SHOWS TO THE THEATRE
2.LIST OF SHOWS IN THE THEATRES
3.LIST ALL THE THEATRES
4.BOOK MOVIE TICKET
7.EXIT(0)

```

### 3.2 Cancellation of Tickets

If the customer wants to cancel the tickets for any reason the request will be processed and cancelled ticket will be shown.

#### Test Case 4: Cancelling an available ticket

##### Sample Input:

Enter the ticket id to be cancelled:

1

Expected Output:

Ticket Cancelled Successfully

Details will be shown

Actual Output:

```
1.ADD SHOWS TO THE THEATRE
2.LIST OF SHOWS IN THE THEATRES
3.LIST ALL THE THEATRES
4.BOOK MOVIE TICKET
5.CANCEL TICKET
7.EXIT(0)
5
Enter the ticketID to be cancelled:1
*****
Ticket ID: 1
Venue: SRS
Audi Number: 1
Customer ID: 1
Customer Name: APOORVA
Show Name: KGF
Total Payment: 520
Selected seats are:
A01
Ticket Cancelled
*****
```

**Test Case 4: Cancelling an unavailable ticket**Sample Input:

Enter the ticket id to be cancelled:

4

Expected Output:

No such ticket exists

Actual Output:

```
1.ADD SHOWS TO THE THEATRE
2.LIST OF SHOWS IN THE THEATRES
3.LIST ALL THE THEATRES
4.BOOK MOVIE TICKET
5.CANCEL TICKET
7.EXIT(0)
5
Enter the ticketID to be cancelled:4
No such ticket exists
*****
```

### 3.2 Customer Registration

Before booking ticket customer has to register himself with valid login credentials.

#### Test Case 6: Successful Registration

##### Sample Input:

Enter the Id:

1

Enter the name:

Ashutosh

Enter the mobile number:

9984795120

Enter the debit card number:

1234567898765432

Enter the city:

Lucknow

##### Expected Output:

Details Entered Successfully //Further according to location booking will start.

##### Actual Output:



```
4
Enter your ID:1
Enter your name:ASHUTOSH
Enter your mobile number:9984795100
Enter your debit card number:1234567898765432
Enter your password:12345678
Enter your city:LUCKNOW
*****
```

### Test Case 7: Failed Registration

#### Sample Input:

Enter the id:

1

#### Expected Output:

It will show the error that customer with the given Id already exists

User will be asked to enter an unique ID.

#### Actual Output:

```
1.ADD SHOWS TO THE THEATRE
2.LIST OF SHOWS IN THE THEATRES
3.LIST ALL THE THEATRES
4.BOOK MOVIE TICKET
5.CANCEL TICKET
7.EXIT(0)
4
Enter your ID:1
Error Number:404
Message:Customer with the ID already Exists
```

#### Sample Input:

Enter the id:

1

Enter name:

Apoorva

Enter Mobile Number:

1234

Expected Output:

Invalid Mobile Number

Enter mobile number

Actual Output:

```

*****
1.ADD SHOWS TO THE THEATRE
2.LIST OF SHOWS IN THE THEATRES
3.LIST ALL THE THEATRES
4.BOOK MOVIE TICKET
5.CANCEL TICKET
7.EXIT(0)
4
Enter your ID:1
Enter your name:APOORVA
Enter your mobile number:1234
Error Number:999
Message:Invalid Mobile Number
Enter your mobile number:

```

Sample Input:

Enter the id:

1

Enter name:

Apoorva

Enter Mobile Number:

9983451234

Enter Debit Card Number:

99876

Expected Output:

Invalid Debit Number

Enter Debit number

Actual Output:

```
1.ADD SHOWS TO THE THEATRE
2.LIST OF SHOWS IN THE THEATRES
3.LIST ALL THE THEATRES
4.BOOK MOVIE TICKET
5.CANCEL TICKET
7.EXIT(0)
4
Enter your ID:1
Enter your name:APOORVA
Enter your mobile number:1234
Error Number:999
Message:Invalid Mobile Number
Enter your mobile number:9984795100
Enter your debit card number:123445
Error Number:1100
Message:Invalid Debit Card Number
```

## 4. Implementation

### 4.1 Results

#### 4.1.1 Main Menu

```
*****
1.ADD SHOWS TO THE THEATRE
2.LIST OF SHOWS IN THE THEATRES
3.LIST ALL THE THEATRES
4.BOOK MOVIE TICKET
5.CANCEL TICKET
7.EXIT(0)
_
```

## 4.1.2 Add Shows

```
*****
1.ADD SHOWS TO THE THEATRE
2.LIST OF SHOWS IN THE THEATRES
3.LIST ALL THE THEATRES
4.BOOK MOVIE TICKET
7.EXIT(0)
1
-----THEATRES-----
1.SRS LUCKNOW
2.CINEPOLIS LUCKNOW
3.WAVE LUCKNOW
4.PVR LUCKNOW
5.FUN LUCKNOW
6.PHOENIX KANPUR
7.SRS KANPUR
8.SRS BANGALORE
9.SRS DELHI
10.OASIS HUBLI
11.URBAN HUBLI
-----
Enter the theatre where you want to add the show:1
What type of Show you want to add in the theatre:
Enter the choice:
1.Movie
2.Live Event
1
Enter the Movie details:
Enter the Movie ID:
1
Enter the name of the Movie:
AVENGERS
Enter the Base Price of the Movie:
200
Enter the Release Date of the Movie:
23-MAY-2020
Enter the date:30-MAY-2020
Enter the three session slots for the movie:9:00AM
12:00PM
6:00PM
Details Entered Successfully
Movie ID: 1
Movie Name: AVENGERS
Movie Base Price: 200
Movie Release Date: 23-MAY-2020
Show Date:30-MAY-2020
Show Slots are:9:00AM Show Slots are:12:00PM Show Slots are:6:00PM
*****
1.ADD SHOWS TO THE THEATRE
2.LIST OF SHOWS IN THE THEATRES
3.LIST ALL THE THEATRES
4.BOOK MOVIE TICKET
7.EXIT(0)
```

#### 4.1.3 View Theatres

```
*****
1.ADD SHOWS TO THE THEATRE
2.LIST OF SHOWS IN THE THEATRES
3.LIST ALL THE THEATRES
4.BOOK MOVIE TICKET
7.EXIT(0)
3
                                BOOK MY SHOW
-----
List of all the theatres associated

Theatre Name:SRS Theatre ID:1 Theatre Address:LUCKNOW
Theatre Name:CINEPOLIS Theatre ID:2 Theatre Address:LUCKNOW
Theatre Name:WAVE Theatre ID:3 Theatre Address:LUCKNOW
Theatre Name:PVR Theatre ID:4 Theatre Address:LUCKNOW
Theatre Name:FUN Theatre ID:5 Theatre Address:LUVKNOW
Theatre Name:PHOENIX Theatre ID:6 Theatre Address:KANPUR
Theatre Name:SRS Theatre ID:7 Theatre Address:KANPUR
Theatre Name:SRS Theatre ID:8 Theatre Address:BANGALORE
Theatre Name:SRS Theatre ID:9 Theatre Address:DELHI
Theatre Name:OASIS Theatre ID:10 Theatre Address:HUBLI
Theatre Name:URBAN Theatre ID:11 Theatre Address:HUBLI
*****
```

#### 4.1.4 View Shows

```
*****
1.ADD SHOWS TO THE THEATRE
2.LIST OF SHOWS IN THE THEATRES
3.LIST ALL THE THEATRES
4.BOOK MOVIE TICKET
7.EXIT(0)
2
*****
Enter the theatre number where you want the list of available shows:1
Movie ID: 1
Movie Name: AVENGERS
Movie Base Price: 200
Movie Release Date: 23-MAY-2020

Show Date:30-MAY-2020
Show Slots are:9:00AM Show Slots are:12:00PM Show Slots are:6:00PM
*****
```

#### 4.1.5 View and Booking Ticket

```
4
Enter your ID:1
Enter your name:ASHUTOSH
Enter your mobile number:9984795100
Enter your debit card number:1234567898765432
Enter your password:12345678
Enter your city:LUCKNOW
*****
Theatres Available In Your City Are:

Theatre Name:SRS Theatre ID:1 Theatre Address:LUCKNOW

Theatre Name:CINEPOLIS Theatre ID:2 Theatre Address:LUCKNOW

Theatre Name:WAVE Theatre ID:3 Theatre Address:LUCKNOW

Theatre Name:PVR Theatre ID:4 Theatre Address:LUCKNOW

Enter the theatre id:1
The list of shows in the preferred theatre are:
Movie ID: 1
Movie Name: AVENGERS
Movie Base Price: 200
Movie Release Date: 23-MAY-2020

Show Date:30-MAY-2020
Show Slots are:9:00AM Show Slots are:12:00PM Show Slots are:6:00PM
Enter the show ID to be booked:1
The Available seats are:
Room ID:1 Total Seating Available:9

Seat & Price Info
A01 120 30
A02 120 30
A03 120 30
M01 160 35
M02 160 35
M03 160 35
V01 200 40
V02 200 40
V03 200 40

Enter the number of people:
```

```

Enter the number of people:2
Enter the seat id to be selected:A01
Seat Available
A02
Seat Available
Book your snacks:
Enter 1 or 0
1
Snacks available are:
Product ID:1:Name:BURGER :Quantity:1 Price:100
Product ID:2:Name:COLD DRINK :Quantity:1 Price:200
Product ID:3:Name:POPCORN :Quantity:1 Price:300
Enter the number of snacks:1
Enter the snack Id to be booked:
1
540
*****
Ticket ID:1
Venue:SRS
Audi Number:1
Customer ID:1
Customer Name:ASUTOSH
Show Name:KGF
Total Payment:540
Selected seats are:
A01 A02
*****
1.ADD SHOWS TO THE THEATRE
2.LIST OF SHOWS IN THE THEATRES
3.LIST ALL THE THEATRES
4.BOOK MOVIE TICKET
5.CANCEL TICKET
7.EXIT(0)

```

#### 4.1.6 Cancel Ticket

```

1.ADD SHOWS TO THE THEATRE
2.LIST OF SHOWS IN THE THEATRES
3.LIST ALL THE THEATRES
4.BOOK MOVIE TICKET
5.CANCEL TICKET
7.EXIT(0)
5
Enter the ticketID to be cancelled:1
*****
Ticket ID: 1
Venue: SRS
Audi Number: 1
Customer ID: 1
Customer Name: APOORVA
Show Name: KGF
Total Payment: 520
Selected seats are:
A01
Ticket Cancelled
*****

```



