

Students' Auditorium Management Software (SAMS)

TEST SUITE

Apurv Kumar (14CS10006) & Aniket Choudhary (14CS30004)

Contents

1	Black Box Testing	2
1.1	BBT for database connection method	2
1.2	BBT for seat status method	3
1.3	BBT for login method	3
1.3.1	Show Manager	3
1.3.2	Sales person	4
1.3.3	Accountant	5
1.4	BBT for change account details method	6
1.5	BBT for Add Employee method	7
1.6	BBT for Add Expenditure method	7
1.7	BBT for Book Ticket method	8
1.8	BBT for Cancel Ticket method	8
1.9	BBT for Add Show method	10
1.10	BBT for Event Details method	10
1.10.1	Event Expenditures	11
1.10.2	Event Sales	11
1.10.3	Event Status	12
1.10.4	Event's Seat Status	12
1.11	BBT for Transactions method	13
1.12	BBT for Sales per Employee method	14
1.13	BBT for Yearly balancesheet method	15
2	White Box Testing	16
2.1	WBT for testing Login	16
2.2	WBT for testing AddShows	17
2.3	WBT for testing EditAccountDetails	18
3	User Interface Testing	19
3.1	Home Interface	20
3.2	Show Manager Interface	24
3.3	Sales Person Interface	24
3.4	Accountant Interface	41

Black Box Testing

Black box testing is a software testing technique in which functionality of the software under test (SUT) is tested without looking at the internal code structure, implementation details and knowledge of internal paths of the software. This type of testing is based entirely on the software requirements and specifications. In Black Box Testing we just focus on inputs and output of the software system without bothering about internal knowledge of the software program.

We followed the following sequence of steps for black box testing of our Software

- Re-examination of requirements and specification of the software.
- Equivalence classes of all possible valid and invalid inputs for various scenarios are created.
- Proper inputs from each possible equivalence class is chosen and entered in the software.
- Expected output of those inputs are determined by the tester.
- The test cases are executed and any variation in output from that expected is duly noted, re-examined and re-tested.

We have done functional blackbox testing on our software. we have checked all the functional abilities of the software as per the SRS and software design documents. We have also tried to check for as many boundary cases and equivalence classes as possible

BBT for database connection method

This testing is for the database connection method and checks for event when database is available and when it is not available due to connection failure or other technical reasons.

Input	Expected Output	Actual Output	Notes
Connection to the Database in the given url is not available	An error message should be displayed	Error message in the console but not in the software	No error is displayed in the software software remains in previous state
Connection to the Database at the given url is available	No error message should be displayed	No error message is displayed and software works normally	No bug found

BBT for seat status method

This testing is for the seat status button on the home page, which displays the booked and available seats for the selected event.

Input	Expected Output	Actual Output	Notes
No Event exists, but the seat status button is clicked	An error message should be displayed	An error message is displayed stating "no events created"	No bug found
Event exists, but is not selected in the table	An error message should be displayed	An error message stating "No event selected" is displayed	No bug found
Events exist, and an event selected from the table	No error message should be displayed and seat status should be displayed	No error message is displayed and seat status with every seat's relative position is displayed graphically	No bug found

BBT for login method

This testing uses the login method for showmanager, sales person and accountant in various scenarios.

Show Manager

This testing checks the show manager login method in different cases

Input	Expected Output	Actual Output	Notes
No show manager account created and login is clicked	Error Message should be displayed	An error message is displayed stating "No show manager account is created" and user is taken to add show manager frame	No bug found
Show Manager account is created, but entered username or password is wrong	Error message should be displayed	An error message is displayed stating "Username or Password is incorrect. Try again!"	No bug found
Show manager account is created, and entered username or password is correct	No Error message should be displayed and the user should be logged in	No error message is displayed and the user is successfully logged in	No bug found

Sales person

This testing checks the sales person login method in different cases

Input	Expected Output	Actual Output	Notes
No sales person account created and login is clicked	Error Message should be displayed	An error message is displayed stating "No sales person account is created create a sales person account first"	No bug found
Sales person account is created, but entered username or password is wrong	Error message should be displayed	An error message is displayed stating "Username or Password is incorrect. Try again!"	No bug found
Sales person account is created, and entered username or password is correct	No Error message should be displayed and the user should be logged in	No error message is displayed and the user is successfully logged in	No bug found

Accountant

This testing checks the sales person login method in different cases

Input	Expected Output	Actual Output	Notes
No accountant account created and login is clicked	Error Message should be displayed	An error message is displayed stating "No accountant's account is created create a sales person account first "	No bug found
Accountant's account is created, but entered username or password is wrong	Error message should be displayed	An error message is displayed stating "Username or Password is incorrect.Try again !"	No bug found
Accountant's account is created, and entered username or password is correct	No Error message should be displayed and the user should be logged in	No error message is displayed and the user is successfully logged in	No bug found

BBT for change account details method

This testing is for change login details for show manager, sales person and accountant
This testing checks the change account details method in show manager, sales person and accountant interface in different cases on entering the required data and clicking the update button

Input	Expected Output	Actual Output	Notes
Entry for mobile number is not a number	An error message should be displayed	An error message is displayed stating "Entry for mobile is not a number"	No bug found
Entry for email is not a valid email format	An error message should be displayed	An error message is displayed stating "Entry for email is not a valid"	No bug found
Entry for username is blank	An error message should be displayed	An error message is displayed stating "username or password cannot be empty"	No bug found
Entry for password is blank	An error message should be displayed	An error message is displayed stating "username or password cannot be empty"	No bug found
Entry for username is not blank and entry for email and mobile number is valid	No error message should be displayed and account should be updated	No error message is displayed stating and the account information is updated	No bug found
Entry for name is blank	An error message should be displayed	No error message is displayed and account is updated successfully	Bug found
Entry for username is already existing username	An error message should be displayed	An error message is displayed stating "Entered username already exists !"	No bug found

BBT for Add Employee method

This testing tests the add employee (sales person and accountant) method of show manager interface.

Input	Expected Output	Actual Output	Notes
Entry for mobile number is not a number	An error message should be displayed	An error message is displayed stating "Entry for mobile is not a number"	No bug found
Entry for email is not a valid email format	An error message should be displayed	An error message is displayed stating "Entry for email is not a valid"	No bug found
Entry for username is blank	An error message should be displayed	An error message is displayed stating "username or password cannot be empty"	No bug found
Entry for password is blank	An error message should be displayed	An error message is displayed stating "username or password cannot be empty"	No bug found
Entry for username is not blank and entry for email and mobile number is valid	No error message should be displayed and new account should be created	No error message is displayed stating and a new account is created	No bug found
Entry for name is blank	An error message should be displayed	No error message is displayed and account is created successfully	Bug found
Entry for username is already existing username	An error message should be displayed	An error message is displayed stating "Entered username already exists !"	No bug found

BBT for Add Expenditure method

This testing checks for the add expenditure method called on clicking the add expenditure button in the accountant interface after selecting an event from the event's table.

Input	Expected Output	Actual Output	Notes
No event is selected from the event's table	An error message should be displayed and expenditure shouldnt be added	An error message stating "No event is selected" is displayed and returned	No bug found
An event is selected from the event's table	No error message should be displayed and expenditure should be added	No error message is displayed and the expenditure is added successfully after getting type and amount from the user	No bug found

BBT for Book Ticket method

This testing checks for the Book Ticket method called on clicking the book ticket button in the sales person interface after selecting an event from the event's table.

Input	Expected Output	Actual Output	Notes
No event is selected from the event's table	An error message should be displayed and ticket shouldnt be booked	An error message stating "No event is selected" is displayed and returned	No bug found
An event is selected from the event's table	No error message should be displayed and sales person should be able to book tickets easily	No error message is displayed and a new frame pops up where the sales person can select the seats to be booked and click ok after which a pdf with the tickets pops up	No bug found

BBT for Cancel Ticket method

This testing checks for the Cancel Ticket method called on clicking the cancel ticket button in the sales person interface after selecting a transaction id (from a list of all available transaction ids) from the popup dialog.

Input	Expected Output	Actual Output	Notes
Entered transaction id is for a show which has passed	An error message should be displayed and the salesperson should'nt be able to cancel tickets	An error message is displayed stating "The event has passed" and the transaction is not cancelled	No bug found
Entered transaction id is for a show which hasn't passed	No error message should be displayed and the salesperson should be able to cancel tickets	No error message is displayed and the sales person is asked for confirmation after which the entered transaction is removed and the confirmation is generated in the form of a pdf	No bug found
Entered transaction id is for a show which is more than 3 days from current date	No error message should be displayed and the salesperson should be able to cancel tickets and provide appropriate refund amount to costumer	No error message is displayed and the sales person provides them with refund after deducting 5 rupees for cancellation charges and gives confirmation in the form of generated pdf	No bug found
Entered transaction id is for a show which is less than 3 days from current date but more than 1 day from the current date	No error message should be displayed and the salesperson should be able to cancel tickets and provide appropriate refund gives amount to costumer	No error message is displayed and the sales person provides them with refund after deducting 10/15 rupees for cancellation charges and confirmation in the form of generated pdf	No bug found
Entered transaction id is for a show which is less than 1 day from current date	No error message should be displayed and the salesperson should be able to cancel tickets and provide appropriate refund amount to costumer	No error message is displayed and the sales person provides them with refund after deducting 50 percent for cancellation charges and gives confirmation in the form of generated pdf	No bug found

BBT for Add Show method

This testing is for add show method in the show manager interface. The following cases arise when the add button is clicked in the add shows frame.

Input	Expected Output	Actual Output	Notes
The entered date-time for the event is before current date-time	An error message should be displayed and event should not be added	An error message stating "Invalid date-time is entered" is displayed and event is'nt added	No bug found
The entered date-time for the event is after current date-time	No error message should be displayed and event should be added successfully	No error message is displayed and the event is added successfully	No bug found
The entered data for the name is blank	An error message should be displayed and event should not be added	No error message is displayed and the event is added successfully	Bug found

BBT for Event Details method

This testing checks the various event details method in the show manager interface. The various event's details that can be accessed are event sales, event transactions, event status and seat status.

Event Expenditures

This testing is for the event's expenditure method in the show manager interface. This is called on selecting an event from the event's table and clicking on the event expenditure button. This outputs a table consisting of all the expenditures (type, amount and id of the accountant) for the selected event

Input	Expected Output	Actual Output	Notes
No Event exists, but the event expenditure button is clicked	An error message should be displayed	An error message is displayed stating "no events created"	No bug found
Event exists, but is not selected in the table	An error message should be displayed	An error message stating "No event selected" is displayed	No bug found
Events exist, and an event selected from the table	No error message should be displayed and event expenditure should be displayed	No error message is displayed and selected event's expenditure is displayed in a table	No bug found

Event Sales

This testing is for the event's sales method in the show manager interface. This is called on selecting an event from the event's table and clicking on the event sales button. This outputs a table consisting of all the sales (tickets booked/cancelled with their amount and sales person id) for the selected event

Input	Expected Output	Actual Output	Notes
No Event exists, but the event sales button is clicked	An error message should be displayed	An error message is displayed stating "no events created"	No bug found
Event exists, but is not selected in the table	An error message should be displayed	An error message stating "No event selected" is displayed	No bug found
Events exist, and an event selected from the table	No error message should be displayed and event sales should be displayed	No error message is displayed and selected event's sales is displayed in a table	No bug found

Event Status

This testing is for the event's status method in the show manager interface. This is called on selecting an event from the event's table and clicking on the event status button. This outputs a table consisting of all the sales details for the selected event like the percentage of available seats of different types, total sales of the event, total expenditures of the event and seats of various kinds presently available/booked

Input	Expected Output	Actual Output	Notes
No Event exists, but the event sales button is clicked	An error message should be displayed	An error message is displayed stating "no events created"	No bug found
Event exists, but is not selected in the table	An error message should be displayed	An error message stating "No event selected" is displayed	No bug found
Events exist, and an event selected from the table	No error message should be displayed and event status should be displayed	No error message is displayed and selected event's status is displayed in a popup	No bug found

Event's Seat Status

This testing is for the event's seat status method in the show manager interface. This is called on selecting an event from the event's table and clicking on the event's seat status button. This displays the seats along with their relative position and status graphically

Input	Expected Output	Actual Output	Notes
No Event exists, but the seat status button is clicked	An error message should be displayed	An error message is displayed stating "no events created"	No bug found
Event exists, but is not selected in the table	An error message should be displayed	An error message stating "No event selected" is displayed	No bug found
Events exist, and an event selected from the table	No error message should be displayed and seat status should be displayed	No error message is displayed and seat status with every seat's relative position is displayed graphically	No bug found

BBT for Transactions method

This testing checks the transactions method in the show manager interface. All the transactions are shown in a tabular form on pressing this button. A popup asks the user for the sales person id for which the user wants to see the transaction table

Input	Expected Output	Actual Output	Notes
No transactions are created before pressing this button	An error message should be displayed	An error message is displayed stating "No transactions is created and a blank table is displayed"	Bug found No table should've been displayed in this case
Transactions are created before pressing this button	No error message should be displayed	No error message is displayed and the transaction table is displayed	No Bug found

BBT for Sales per Employee method

This testing checks the sales per employee method in the show manager interface. All the transactions by each sales person are shown in a tabular form on pressing this button.

Input	Expected Output	Actual Output	Notes
Transactions/Salesperson are created before pressing this button	No error message should be displayed	No error message is displayed and the sales per employee table is displayed	No Bug found
No transactions are created before pressing this button	An error message should be displayed	An error message is displayed stating "No transactions is" created and a blank table is displayed	Bug found No table should've been displayed in this case
No sales person is created before pressing this button	An error message should be displayed	No error message is displayed and no table is displayed	Bug found An error should've been displayed in this case

BBT for Yearly balancesheet method

This testing checks the yearly balancesheet method in the show manager interface. All the transactions and expenditures are shown in a tabular form of a particular year which is read from the user on pressing this button. A popup asks the user for the year for balancesheet table is to be displayed

Input	Expected Output	Actual Output	Notes
No transactions are created before pressing this button	An error message should be displayed	An error message is displayed stating "No transactions is created and a" blank table is displayed	Bug found No table should've been displayed in this case
Transactions/expenditures are created before pressing this button	No error message should be displayed	No error message is displayed and the balancesheet table is displayed	No Bug found
No expenditures are created before pressing this button	An error message should be displayed	An error message is displayed stating "No expenditures is created and a" blank table is displayed	Bug found No table should've been displayed in this case
If the entered year doesn't contain any evnts	An error message is expected	An error message stating "No events in the entered year" is displayed	No bug found

White Box Testing

In the White Box Testing, we will verify that all the paths in a function are correct through basic path testing. To conduct White Box Testing , knowledge of underlying programming language is essential. Current day software systems use a variety of programming languages and technologies and its not possible to know all of them. Black box testing gives abstraction from code and focuses testing effort on the software system behaviour. Some White box tests for different functions are as follows:

WBT for testing Login

```
287     String sql = "SELECT id,username,password FROM salesperson WHERE `username`=";
288     ResultSet rs = stmt.executeQuery(sql);
289     //STEP 5: Extract data from result set
290     String password = null;
291     int id=0;
292     while(rs.next()){
293         id=rs.getInt("id");
294         password = rs.getString("password");
295     }
296     rs.close();
297
298     if(yField.getText().equals(password)){
299         JOptionPane.showMessageDialog(frmWelcome,"Login Successful !");
300         SalesPersonHome frame = new SalesPersonHome(id);
301         frame.setVisible(true);
302         frmWelcome.dispose();
303     }
304     else JOptionPane.showMessageDialog(frmWelcome,"Login Unsuccessful Try Again")
```

Figure 2.1: Login Code

Path	Path Expected Output	Path Actual Output	Notes
287-298-299-300	Login Successful Message display	Login Successful Message Display	No bug found
287-298-304	Login Unsuccessful Message display	Login Unsuccessful Message Display	No bug found

WBT for testing AddShows

```
158     Connection con = test.getConnection();
159     Statement stmt = null;
160     String name = textField.getText();
161
162     String datetime=datePicker.getEditor().getText();
163     Date current = new Date();
164     long diff=(sdf.parse(datetime).getTime()-current.getTime());
165     if(diff<0) {
166         JOptionPane.showMessageDialog(contentPane, "Event date-time is
167         not valid message display");
168     }
169     System.out.println("Show is Valid");
```

Figure 2.2: Add New Show Code

Path	Path Expected Output	Path Actual Output	Notes
158-162-164-165-166-167	Event date-time is not valid message display	Event date-time is not valid message Display	No bug found
158-162-164-165-169	Show is valid	Show is valid	No bug found

WBT for testing EditAccountDetails

```
235     try {
236         mobileNumber = Long.parseLong(tfMobile.getText());
237     } catch (java.lang.NumberFormatException ea) {
238         JOptionPane.showMessageDialog(contentPane, "Entered input for Mobile Number is not a number");
239         return;
240     }
241     String email = tfEmail.getText();
242     String EMAIL_REGEX = "^[\\w-_\\.]+[\\w-_\\.]+@[\\w]+\\.[\\w]+[\\w]$";
243     Boolean b = email.matches(EMAIL_REGEX);
244     System.out.println("is e-mail: "+email+" :Valid = " + b);
245     if(!b) {
246         JOptionPane.showMessageDialog(contentPane, "The email format is invalid !");
247         return;
248     }
249     String address = tfAddress.getText();
250     String username = tfUsername.getText();
251     String password = tfPassword.getText();
252     if(username.equals("")&&password.equals("")) {
253         JOptionPane.showMessageDialog(contentPane, "Username and password field cannot be empty");
254         return;
255     }
256     stmt = con.createStatement();
257     String sql = "INSERT INTO salesperson(id,name,gender,age,mobilenumber,email,address,password)";
258     stmt.executeUpdate(sql);
259     JOptionPane.showMessageDialog(contentPane, "The Details have been updated successfully");
```

Figure 2.3: Add New Show Code

Path	Path Expected Output	Path Actual Output	Notes
236-238	Entered mobile number is not a number message display	Entered mobile number is not a number message display	No bug found
236-242-243-245-246	Email Format is not valid message display	Email Format is not valid message display	No bug found
236-242-243-245-250-251-252-253-254	Username and Password field cannot be empty	Username and Password field cannot be empty	No bug found
236-242-243-245-250-251-252-259	The Details have been updated successfully	The Details have been updated successfully	No Bug Found

User Interface Testing

In software engineering, user interface testing is the process of testing a product's graphical user interface to ensure it meets its specifications. This is normally done through the use of a variety of test cases.

For our software we tried to test all the components of the GUI executing all the functionalities of the software interface by interface. We determined a set of expected outputs for each possible input and cross-checked them with the actual output we got from our software to ensure a comprehensive interface by interface test is carried out.

Here, we have attached several screenshots with the inputs entered and the expected outputs for those inputs as follows:



What is Tested?

All Components which an accountant is going to use.

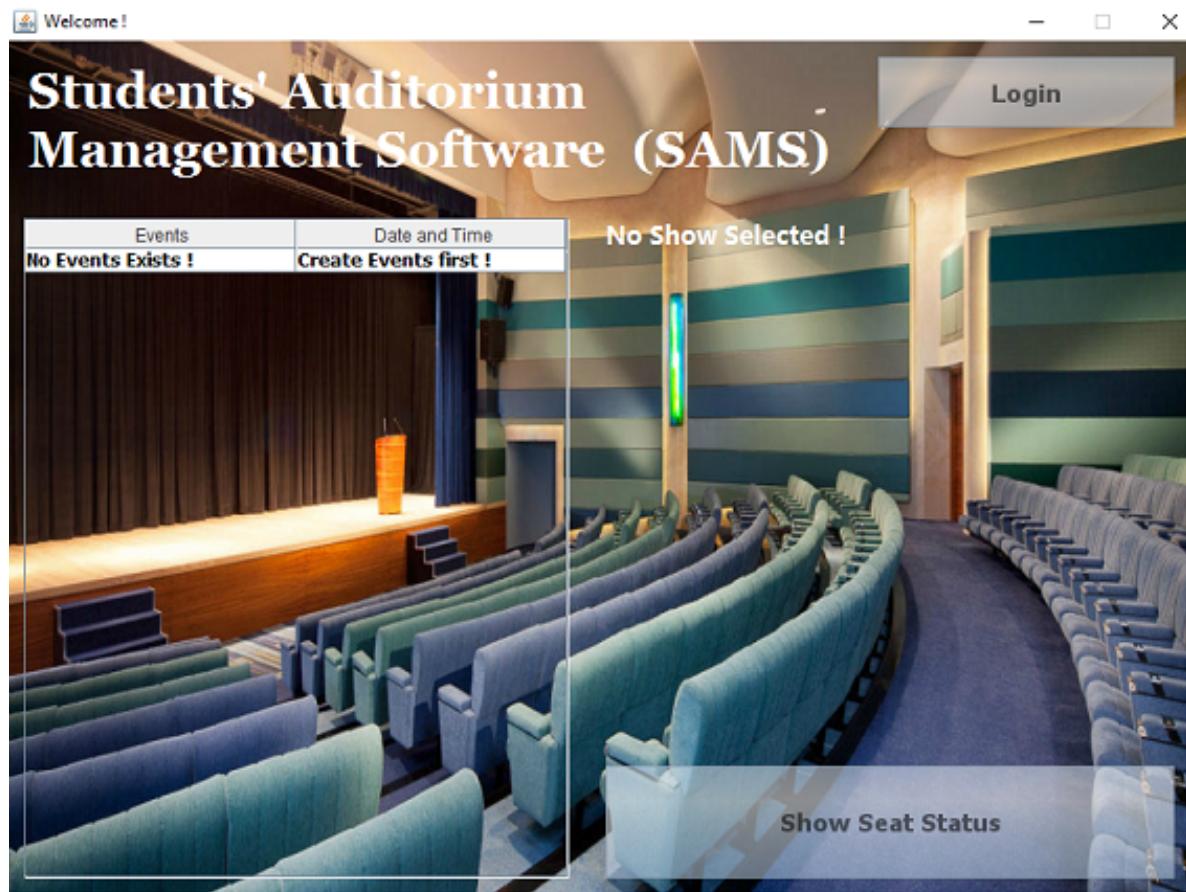
Accountant can change his/her details by clicking on the button 'Change Login Details'.

Accountant can add expenditures to the event by selecting an event first and then by clicking on button 'Add Expenditures'.

Accountant can log out by clicking on button 'Log Out'.

Figure 3.1: What is tested ?

Home Interface



Input :

- User can click on 'Log In' button.
- User can see Show Status by first selecting an event and then clicking on 'Show Seat Status'.

Output :

- After clicking on 'Log In' button a Dialog Box appear having three options 'Sales Manager', 'Sales Person' and 'Accountant'.
- Seat Status dialog box appear after clicking on 'Show Seat Status'.

Effective Results :

All results match the expected results.

Figure 3.2: Home interface



Input: The user clicks on the login button on the home page and is provided with three options as the account type to be logged in (Assuming that account is created). The user then enters the username and password and presses ok button.

Expected Output: The user should be able to log in successfully on providing correct username and password.

Effective Result: If the an account of the type already exists and the username and password entered is correct, then the user is taken to the respective home page of the account type selected.

Figure 3.3: Login Button clicked, Account type selected



Input: The user after clicking the login button on the home page enters wrong username, password or both.

Expected Output: The software should display an error message and not allow the user to access their account.

Effective Result: The software displays an error showing "Login unsuccessful, Try Again !" and on clicking ok button the user is taken back to the home page and is not given access to the account.

Figure 3.4: Login Button clicked, Account type selected, Username/Password entered is incorrect

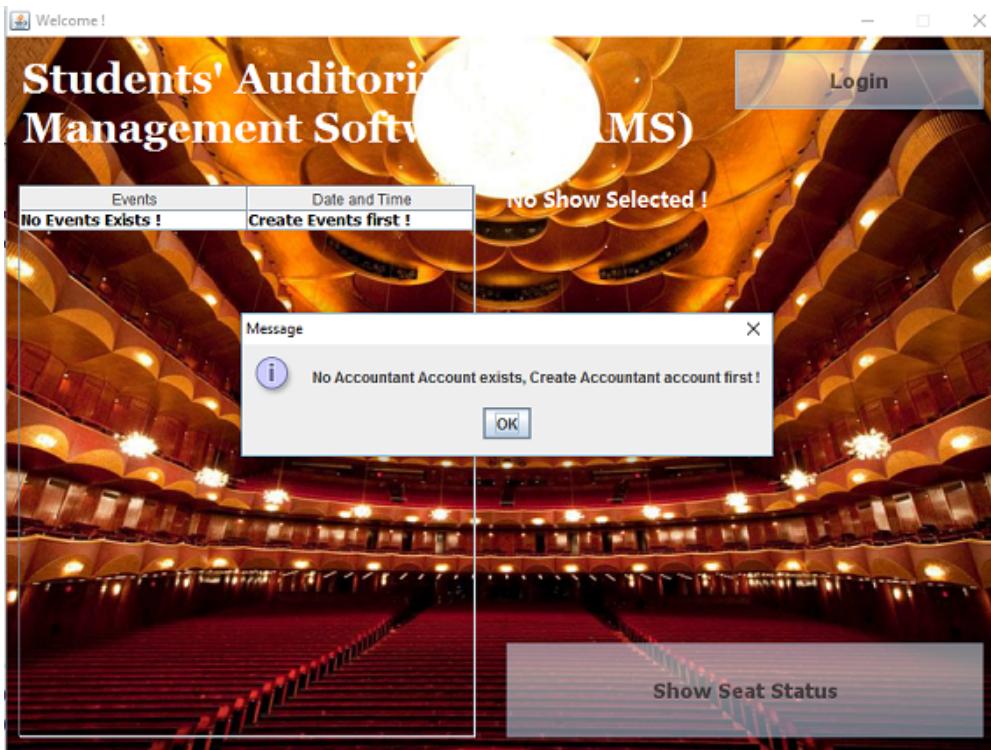


Figure 3.5: Login Button clicked, Accountant Selected

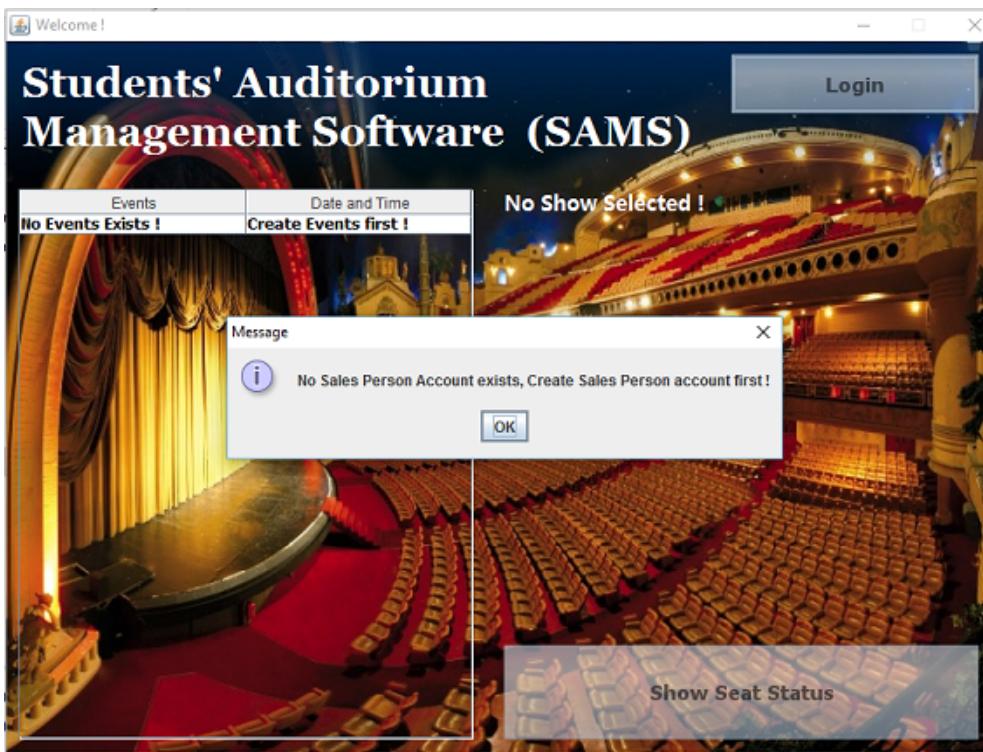


Figure 3.6: Login Button clicked, Sales Person selected



Figure 3.7: Login Button clicked, Show Manager Selected

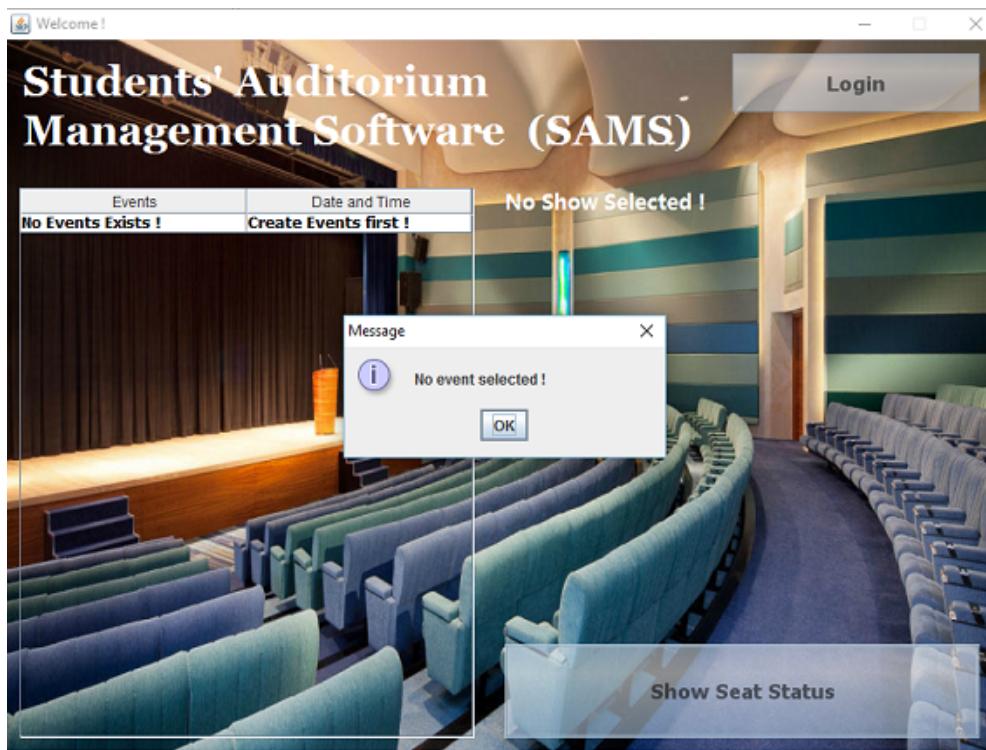
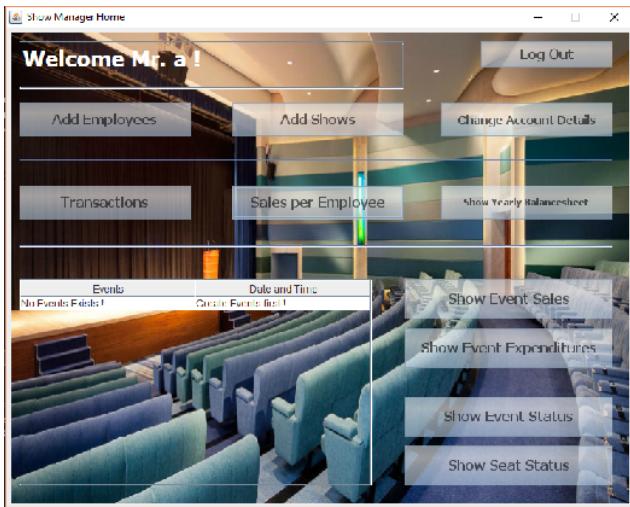


Figure 3.8: Seat Status clicked without selecting an event from the Event table

Show Manager Interface

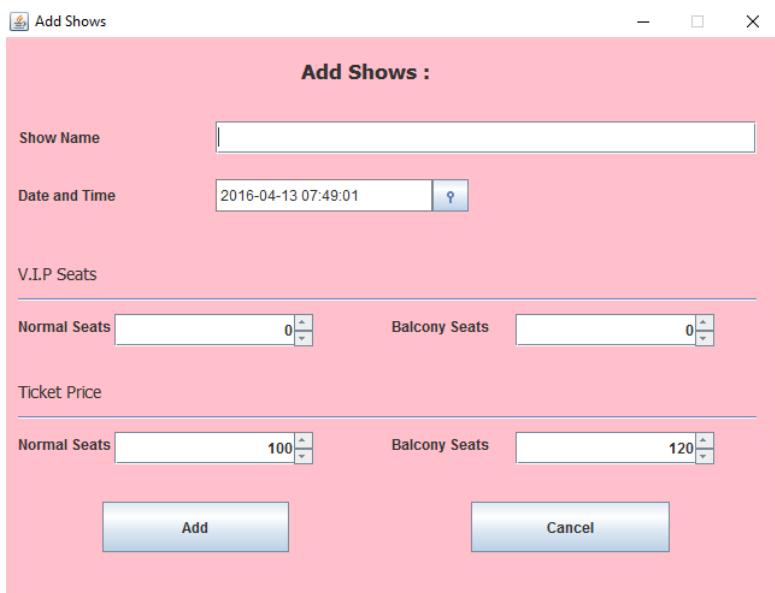


The show manager after logging in successfully from the home interface can perform several functions including changing account details like:

- event's status (percentage of seats booked, total sales/expenditures etc)
- event's seat status and sales and expenditures of the event
- The show manager can also access yearly datasheet which displays all the expenditures/sales for a particular year
- All transactions by all sales person or a particular sales person can be seen by the sales person and sales per employee can also be accessed by the show manager

Figure 3.9: Manager Interface

Sales Person Interface



Input :
-User enters Show Name, Date, Time, Number of VIP Seats and Ticket Price for Normal and Balcony Seats.
Expected Output :
- Show gets updated accordingly.
Effective Results :
-All results match the expected output.

Figure 3.10: Creating a new Event.

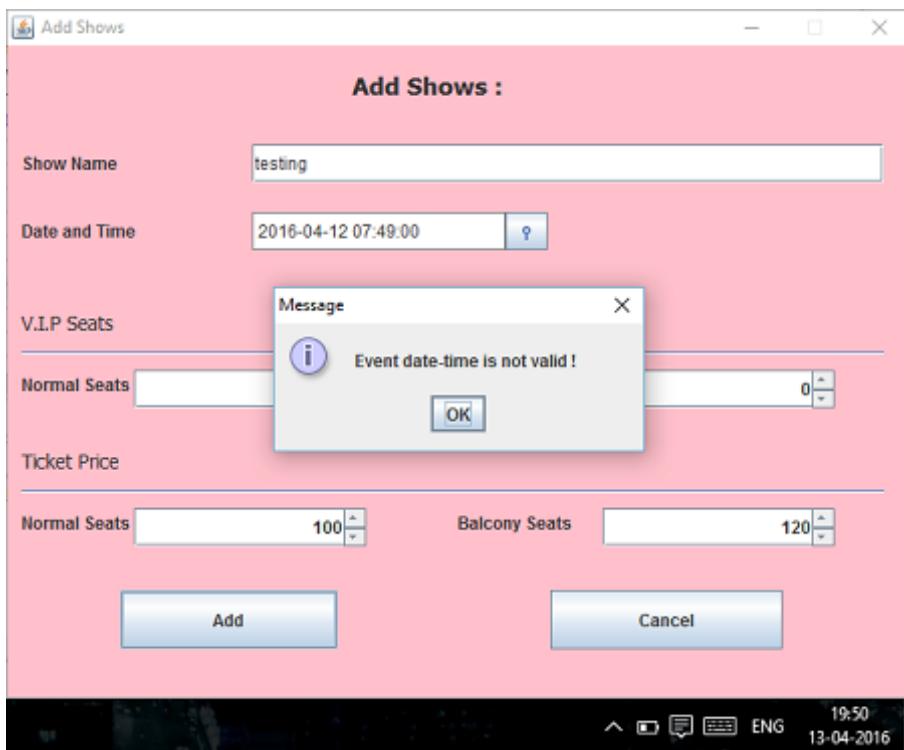


Figure 3.11: Event Time should be greater than current time.

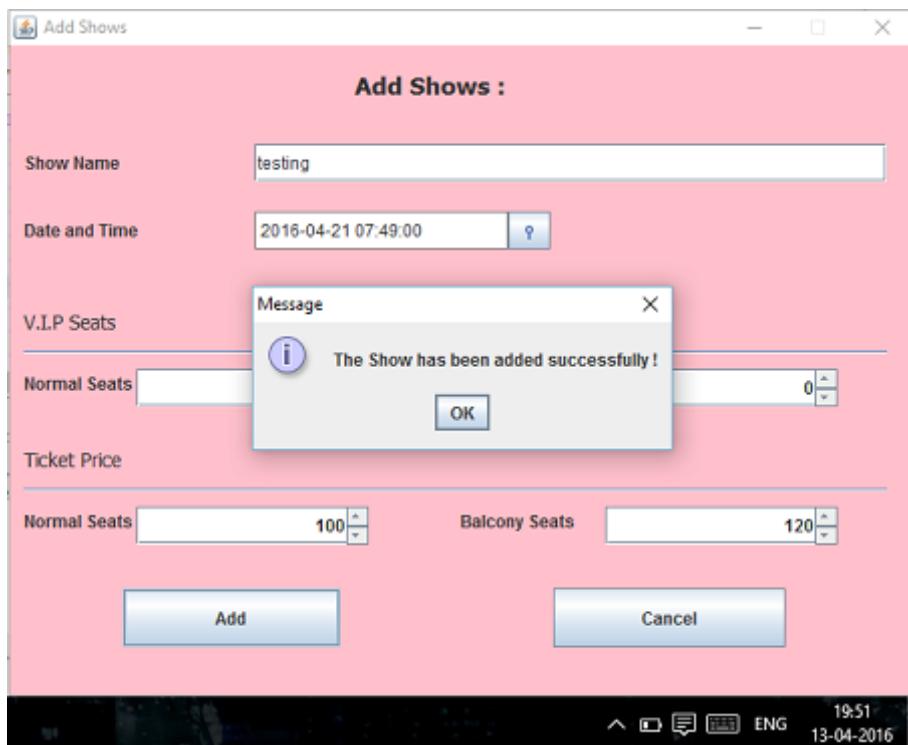
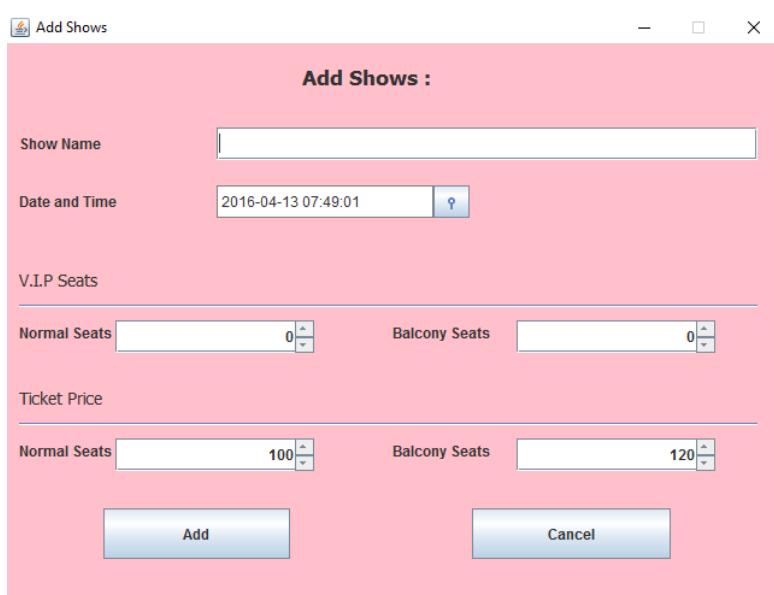


Figure 3.12: Event added successfully.



Input :
-User enters Show Name, Date, Time, Number of VIP Seats and Ticket Price for Normal and Balcony Seats.
Expected Output :
- Show gets updated accordingly.
Effective Results :
-All results match the expected output.

Figure 3.13: Invalid Email format while changing account details.

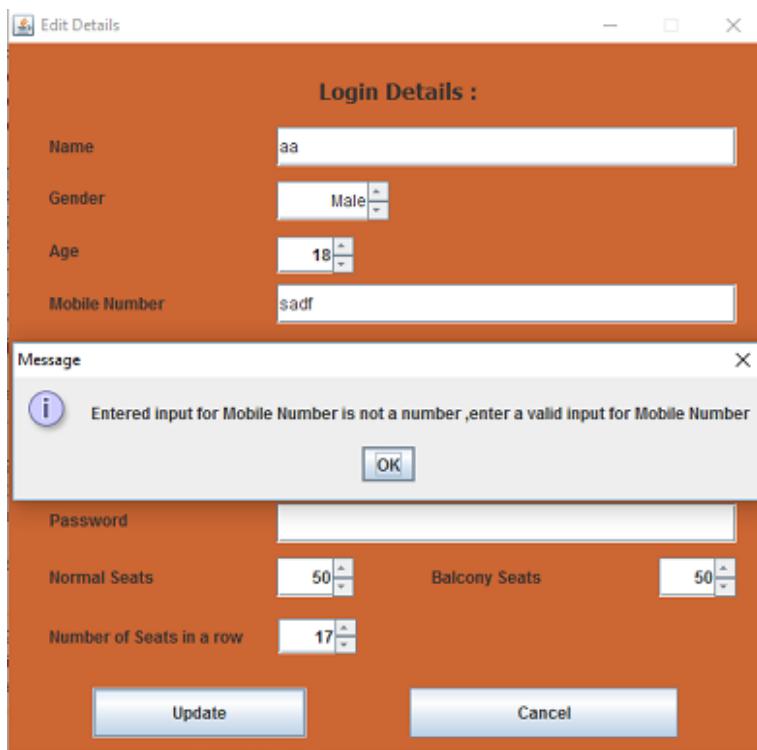


Figure 3.14: Mobile Number is not valid.

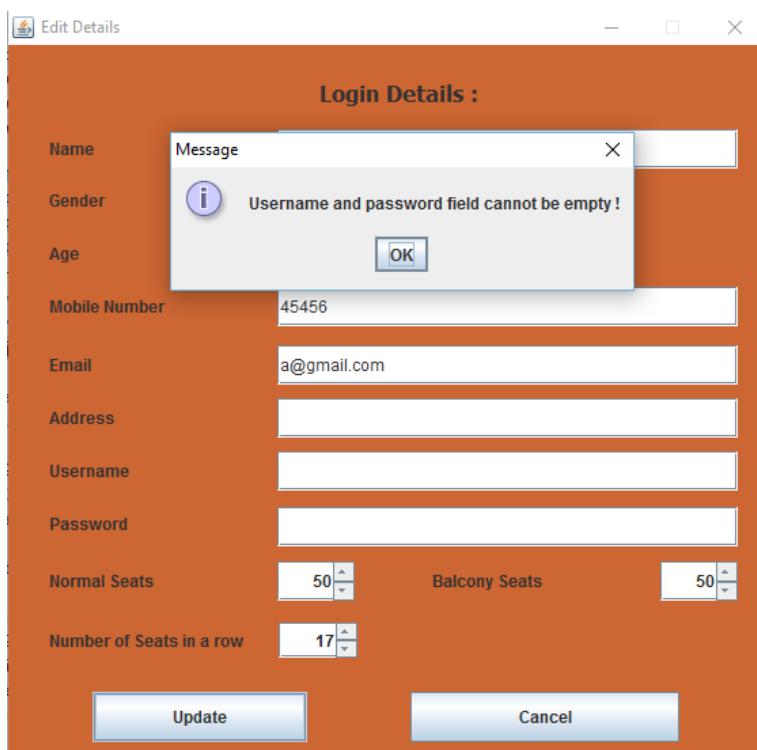


Figure 3.15: Username or Password Field is empty.

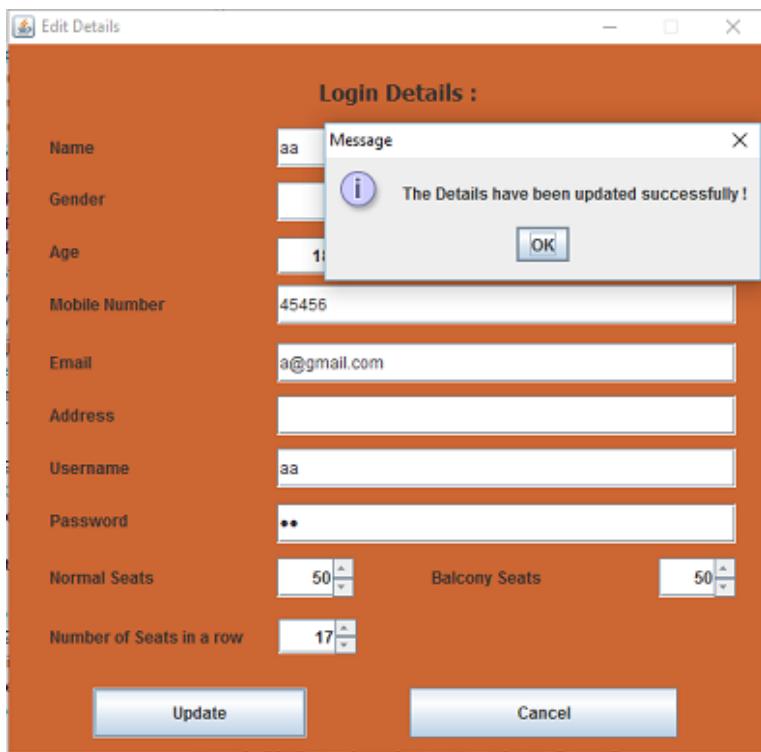


Figure 3.16: Account Details updated successfully.

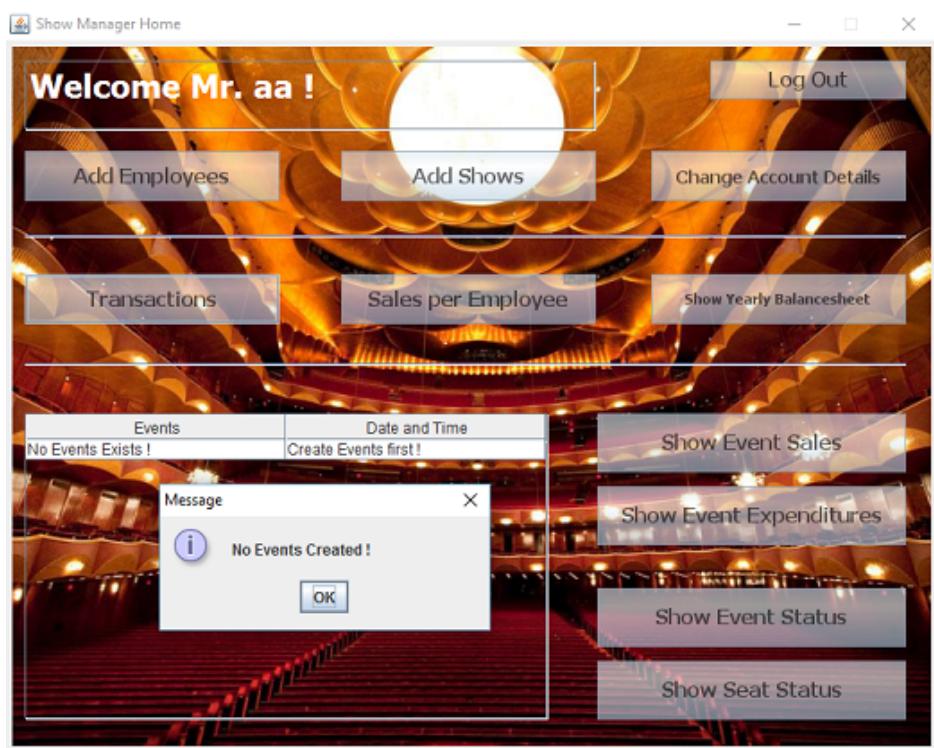


Figure 3.17: Create Shows first.

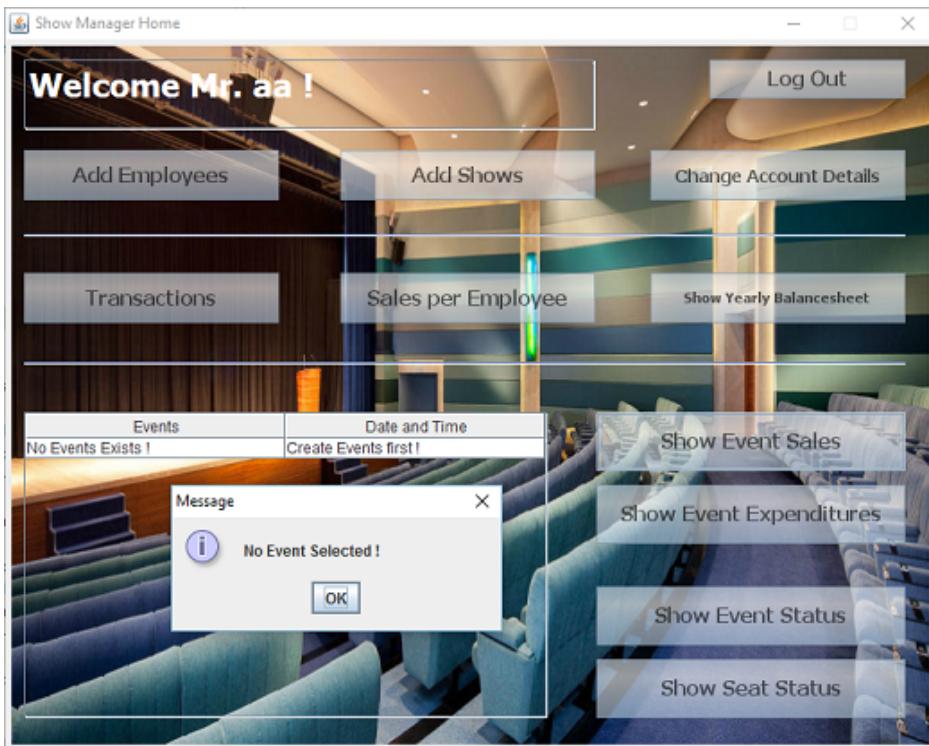
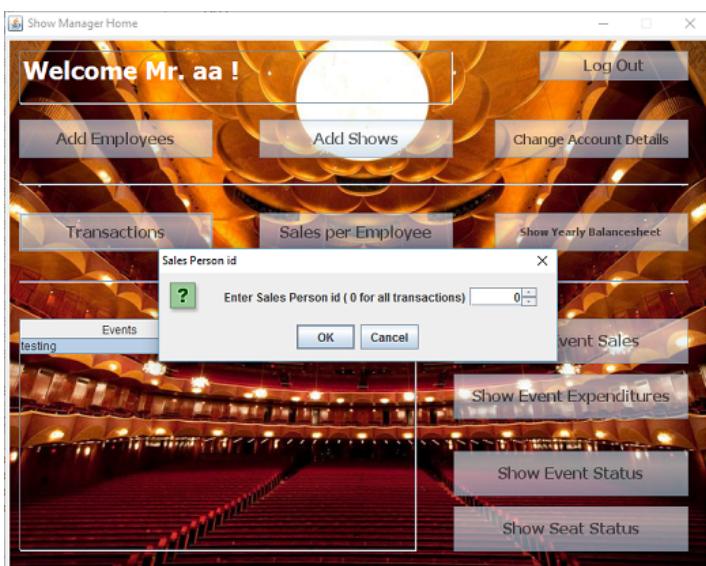


Figure 3.18: Select an Event first.



Input :

-User enters the Sales Person Id to see his/her transaction.

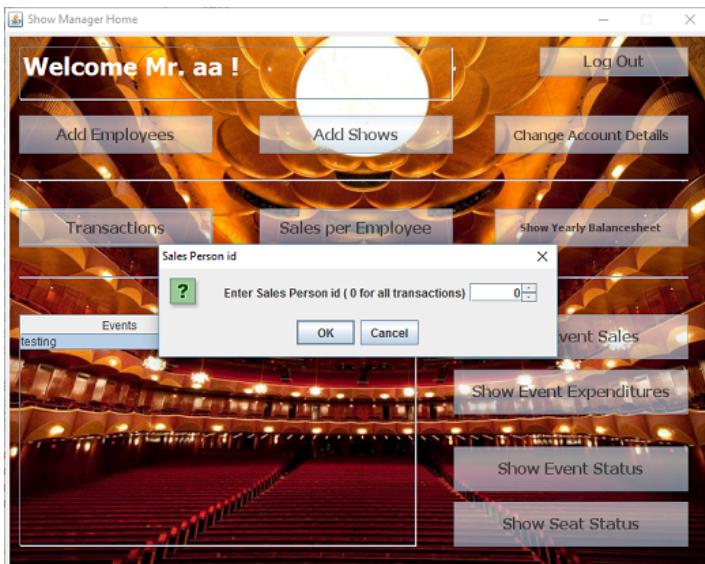
Output :

-A table appears which has all the transactions of that sales person.
-If user enters 0 then transaction for all the sales persons appear.

Effective Results :

-All results matches the expected results.

Figure 3.19: Enter Sales Person Id to see transaction list.



Input :

-User enters the Sales Person Id to see his/her transaction.

Output :

-A table appears which has all the transactions of that sales person.

-If user enters 0 then transaction for all the sales persons appear.

Effective Results :

-All results matches the expected results.

Figure 3.20: List of transaction for selected Sales Person.



Input :

-User enters the year for which he/she wants to see the balancesheet.

Output :

-If events are not available for that year then message box appear displaying 'No Events for this year'.

-Otherwise a table appear containing all the expenditures for entered year.

Effective Results :

All results match the expected results.

Figure 3.21: Choose Year for which you want to see yearly balancesheet.

Data Table

id	name	datetime	Expenditure	Sales
1	testing	2016-04-21 07:49:00.0	0	0
2	testing2	2016-04-30 07:59:00.0	100000	765

OK

Figure 3.22: Balance Sheet for the whole year.

Data Table

seatype	seatid	spid	price
0	3	1	100
0	5	1	100
0	6	1	100
0	24	1	100
2	59	1	120
2	78	1	120
2	88	1	120
2	56	1	5
0	37	2	100
0	40	2	100
2	71	2	120
2	66	2	120

OK

Figure 3.23: Sales for selected event.

Data Table

id	name	Total Sales
1	sales person	765
2	sp2	0

OK

Figure 3.24: Sales List for selected Sales Person.

Data Table

type	amount
Sets	100000
Costumes	100

OK

Figure 3.25: Expenditure Table for selected Event.

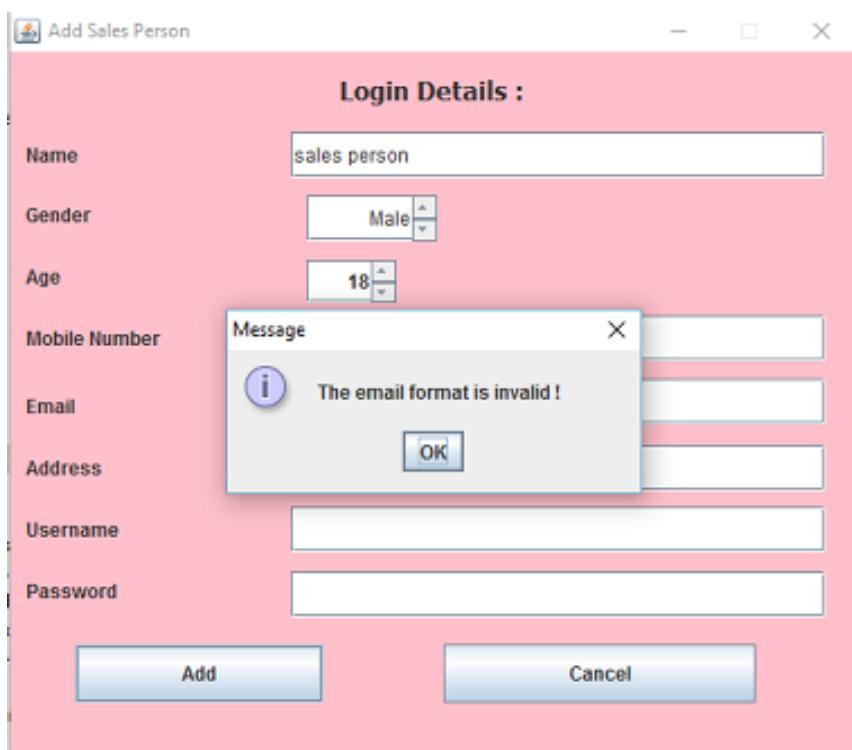


Figure 3.26: Change Account details clicked and email format is invalid

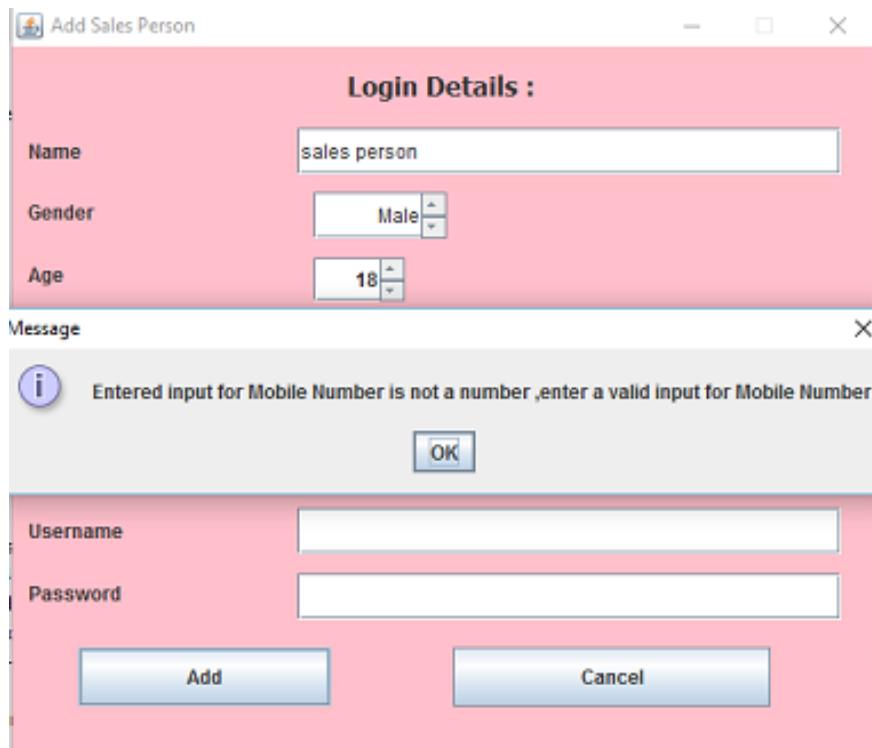


Figure 3.27: Mobile number entered is not a number

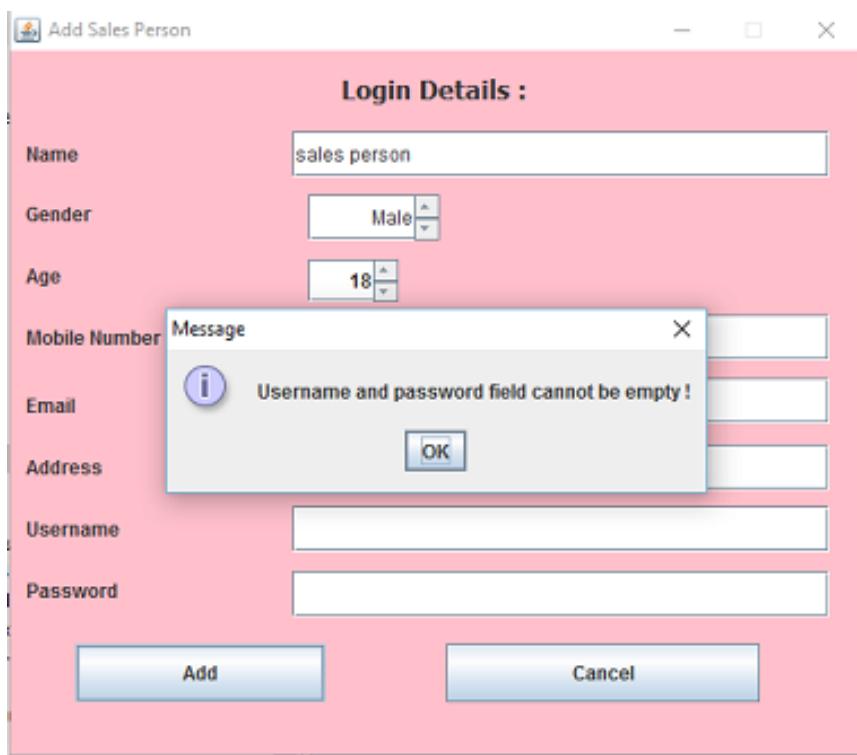


Figure 3.28: Username and/or password is entered blank

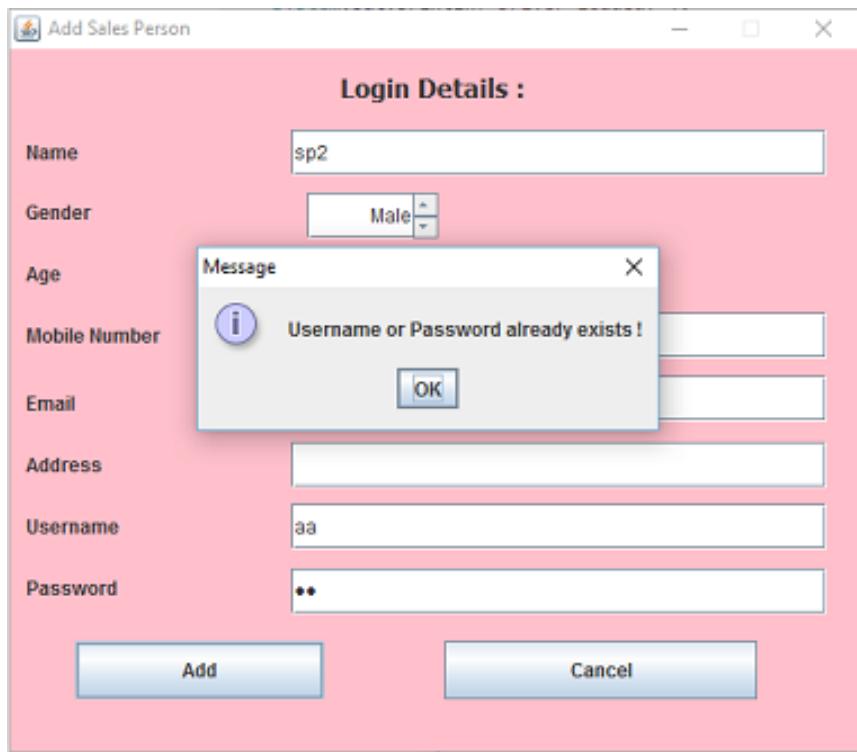
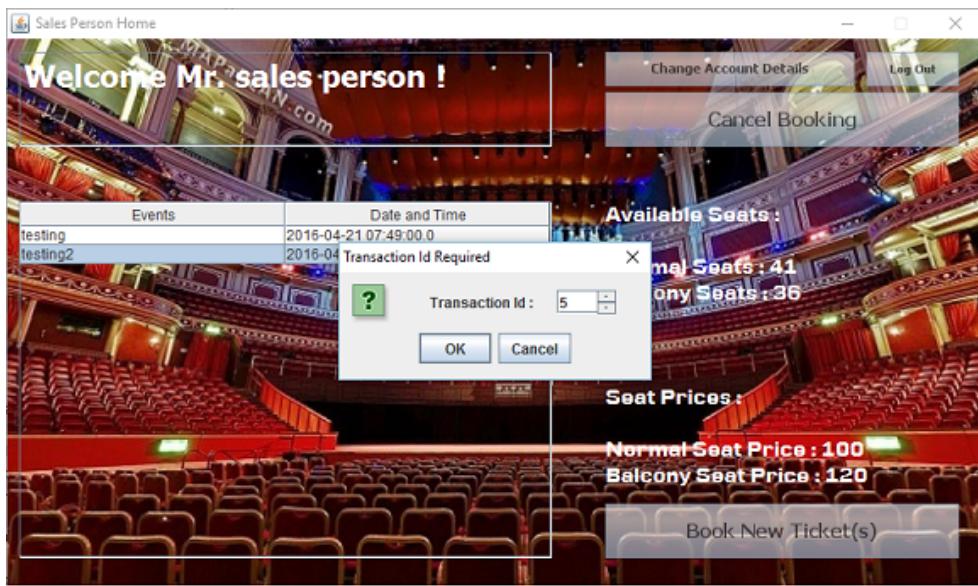


Figure 3.29: username already exists

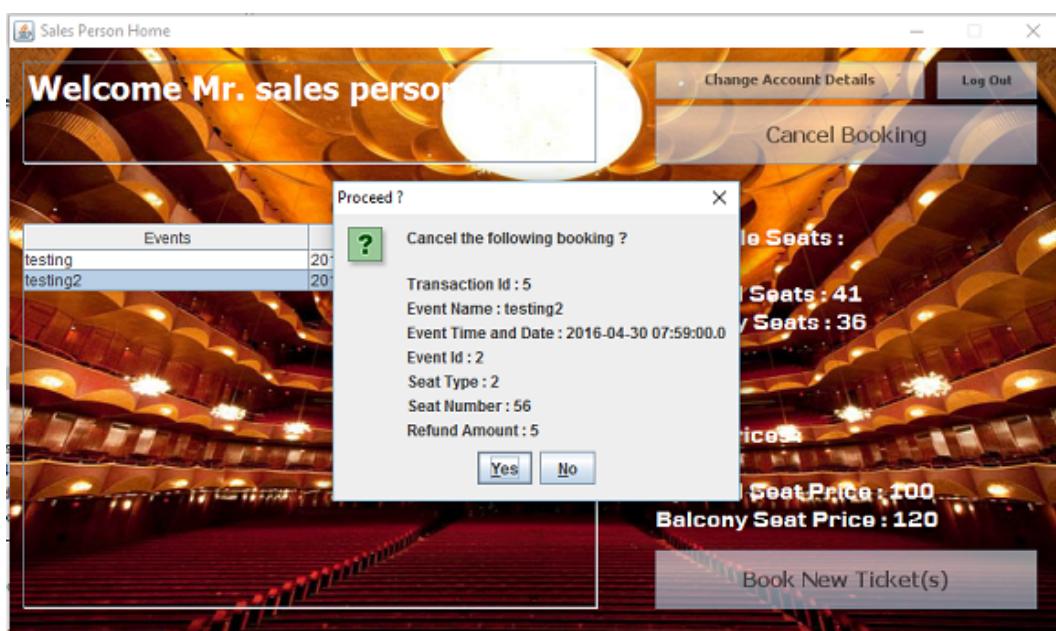


Input :
-User has to enter his/her transaction Id printed on ticket and press the button 'OK'.
-User can also go back to previous window by pressing the button 'Cancel'.

Output :
- A dialog box appears to confirm the cancellation.

Expected Result :
All results match the expected Outputs.

Figure 3.30: cancellation menu



Input :
-User can click on button Yes or No.
Expected Output :
-If 'Yes' button is pressed then ticket gets cancelled. Database updates accordingly. Final Cancel Receipt appears on the screen.
-If 'No' button is pressed then user come back to the previous window.

Figure 3.31: confirmation

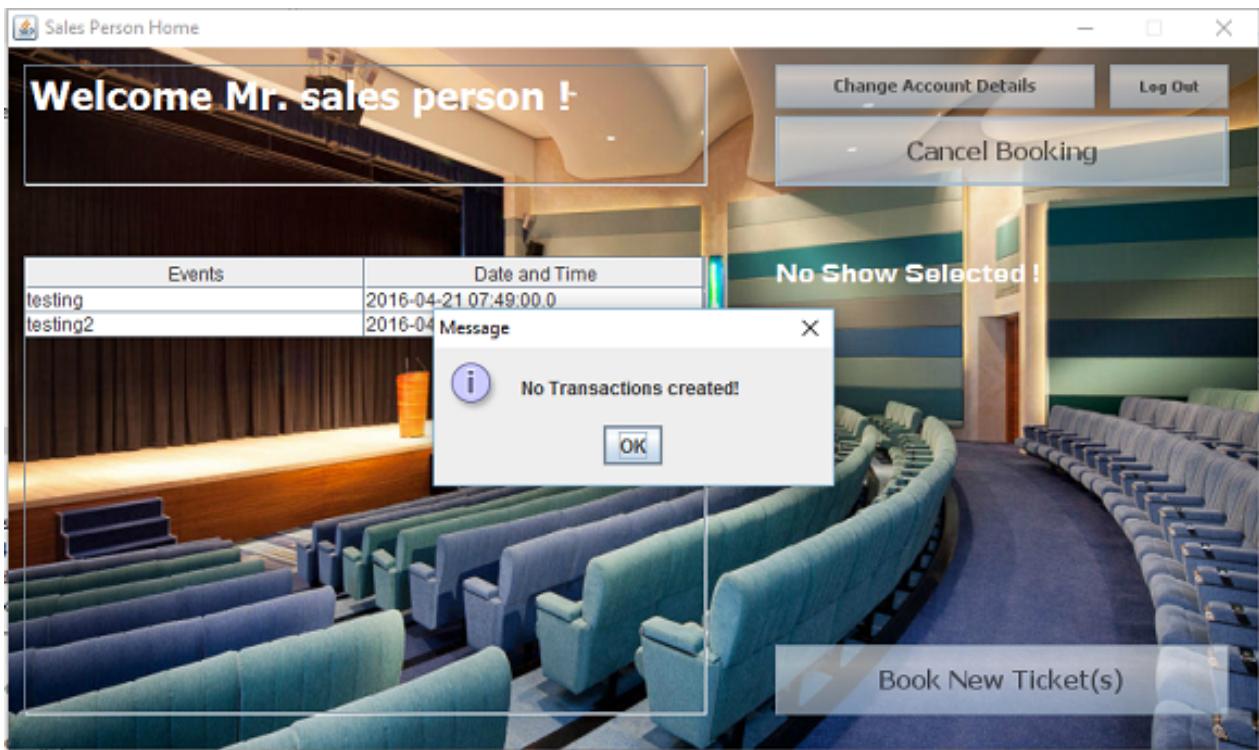


Figure 3.32: Transaction error

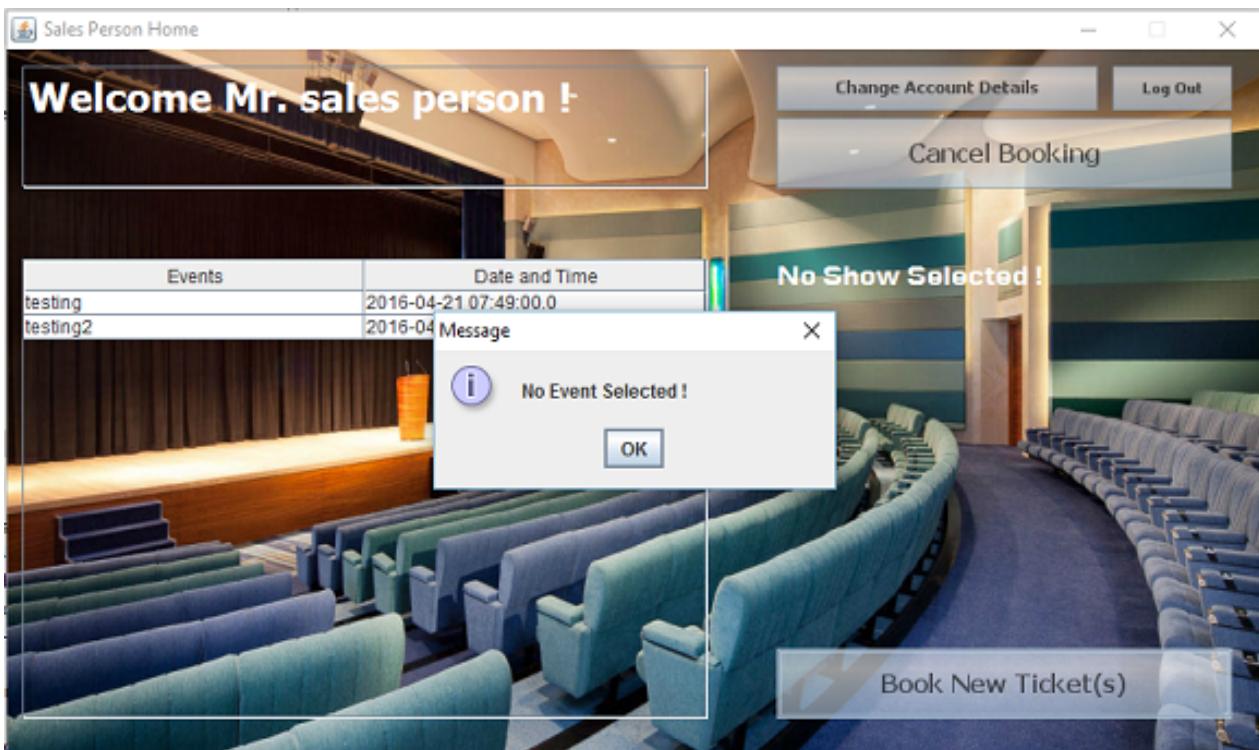


Figure 3.33: No event selected

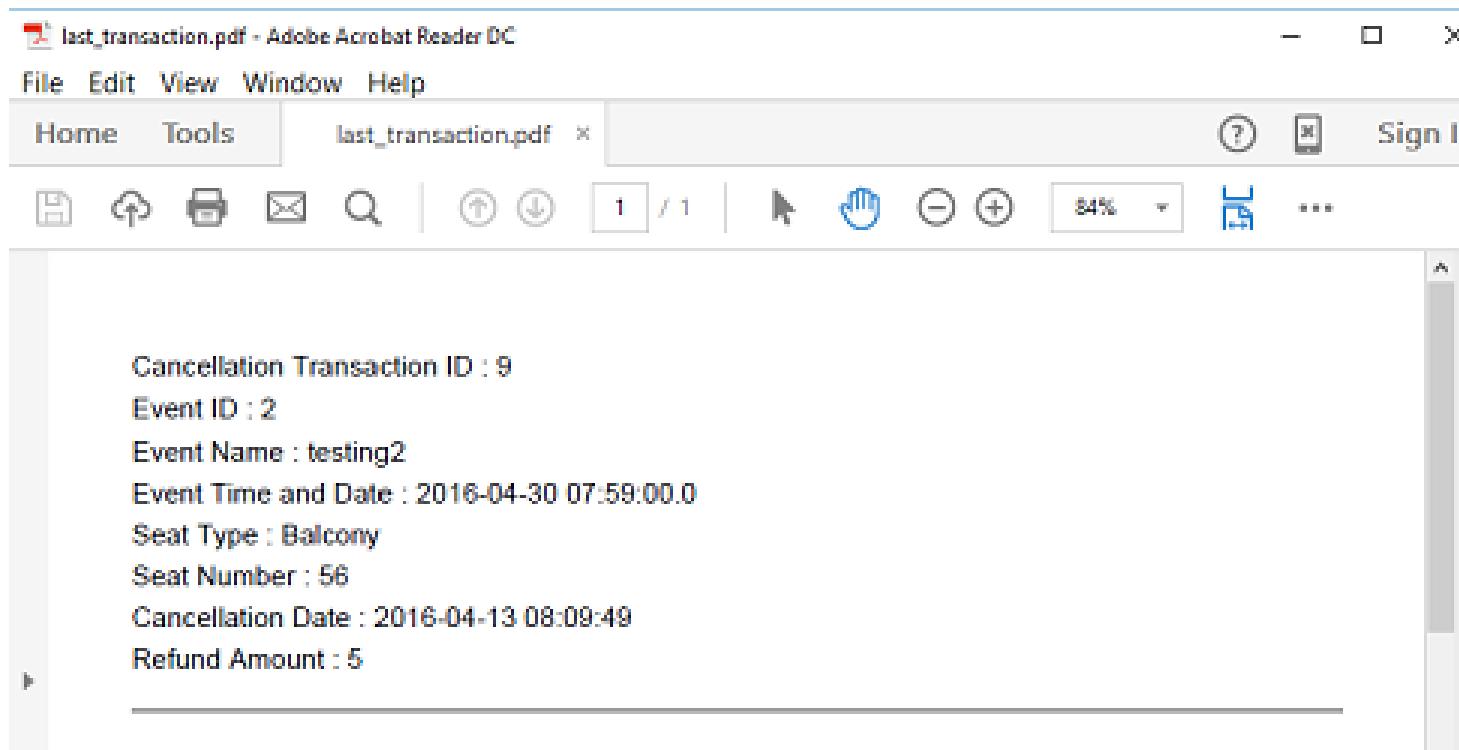


Figure 3.34: cancel receipt

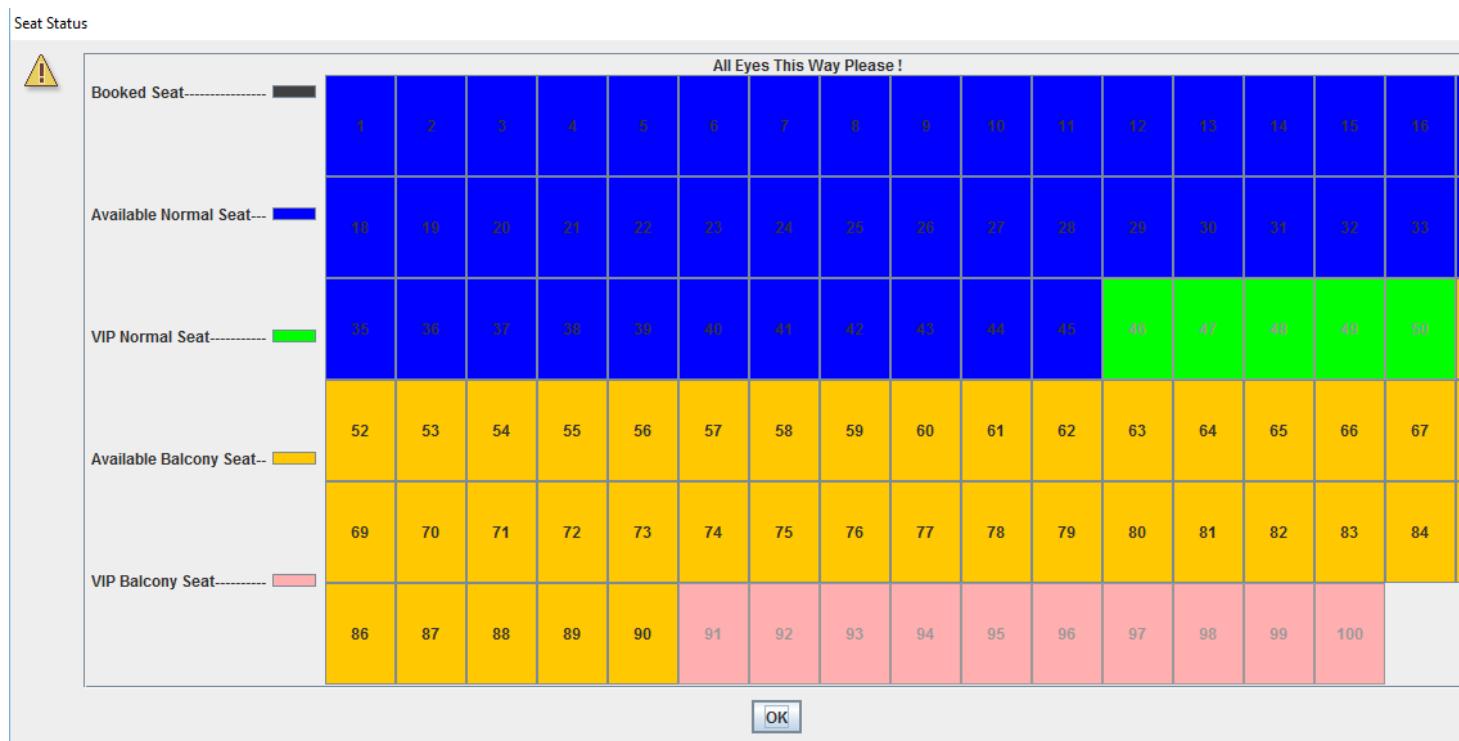


Figure 3.35: Booking

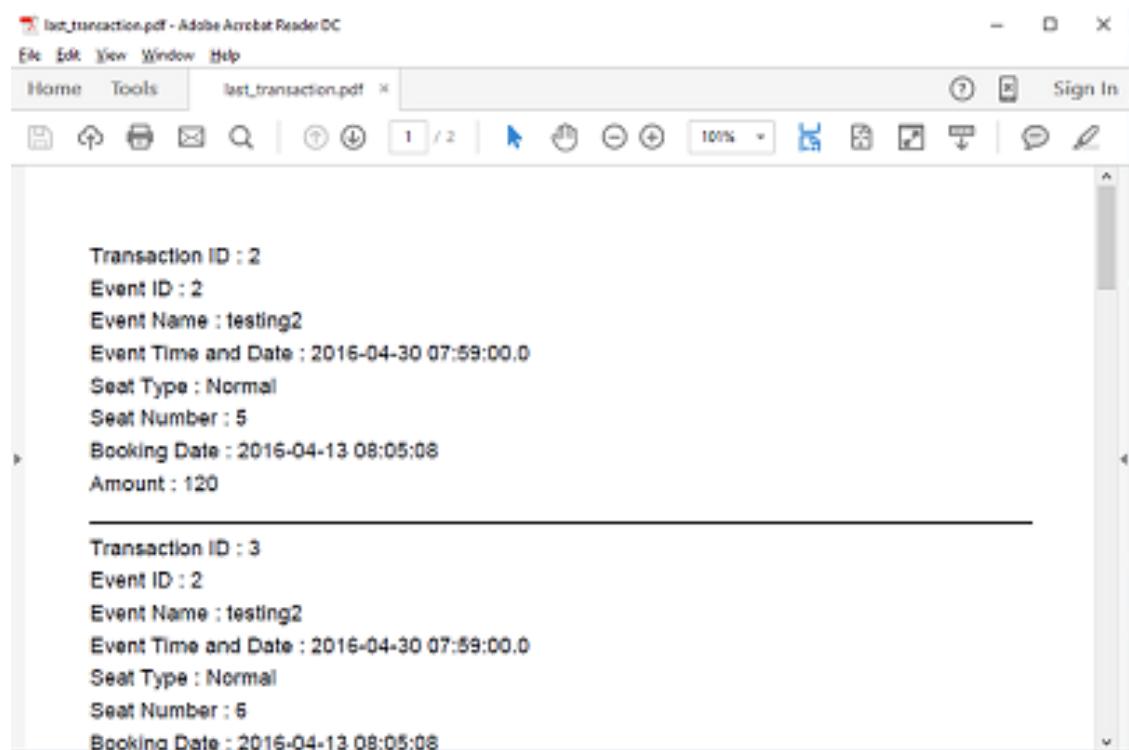


Figure 3.36: Booked Tickets

Accountant Interface



What is Tested?

All Components which an accountant is going to use.

Accountant can change his/her details by clicking on the button 'Change Login Details'.

Accountant can add expenditures to the event by selecting an event first and then by clicking on button 'Add Expenditures'.

Accountant can log out by clicking on button 'Log Out'.

Figure 3.37: Accountant Home

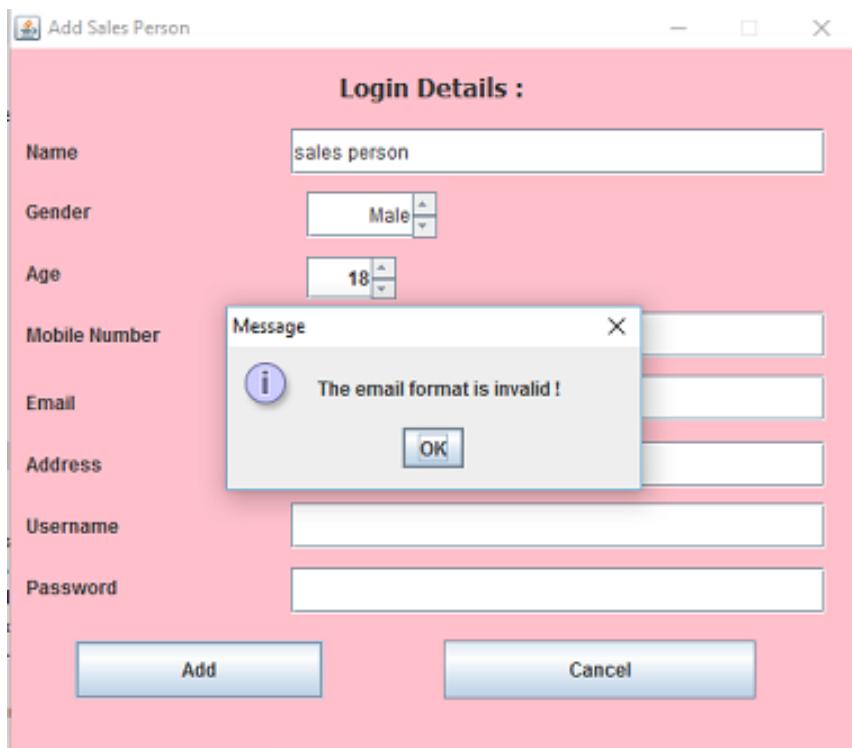


Figure 3.38: Entered email is not valid.

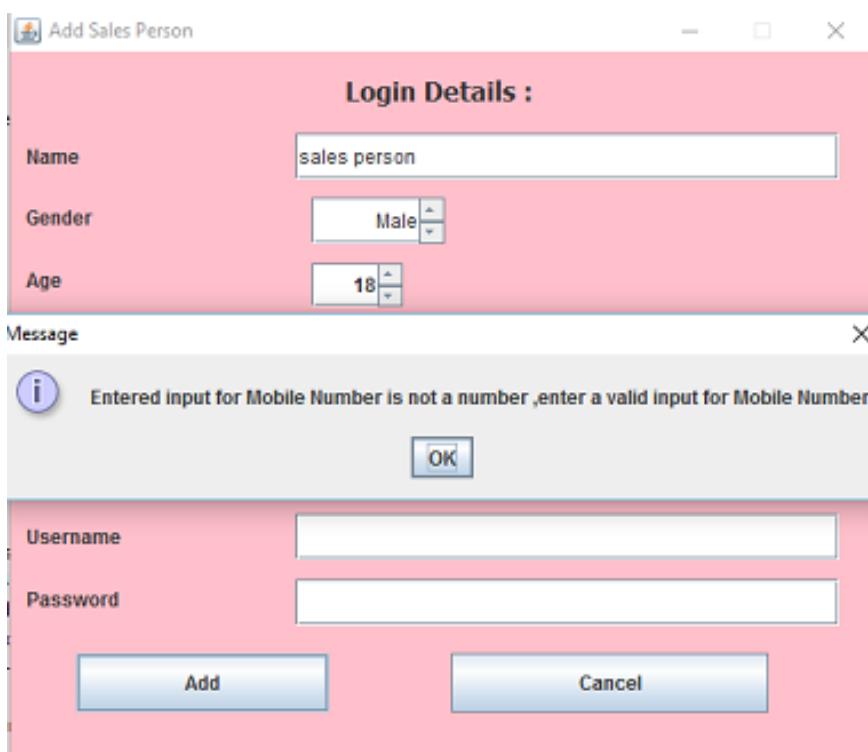


Figure 3.39: Entered Mobile Number is not valid.

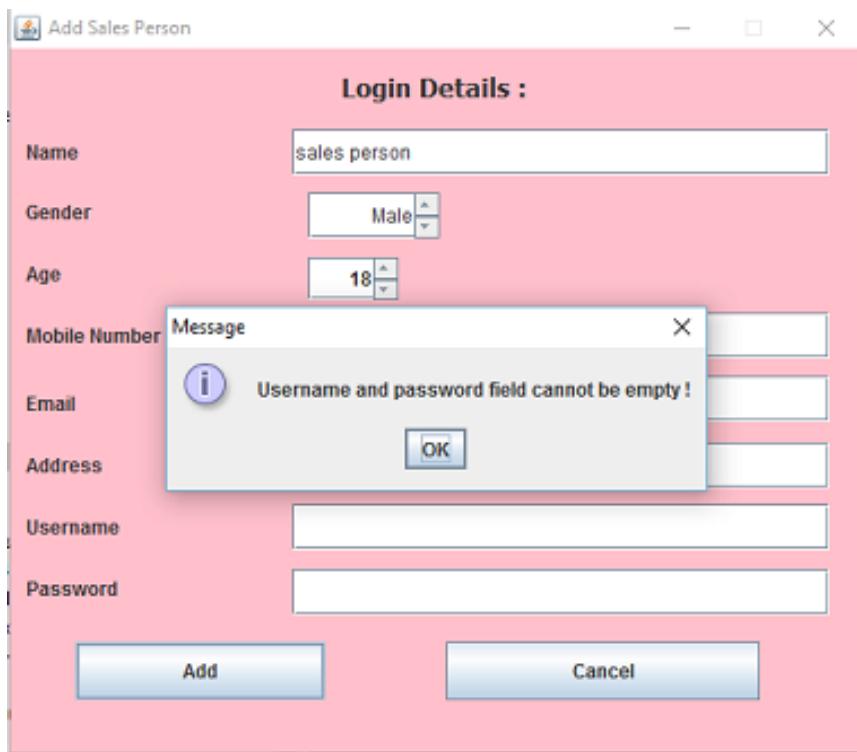


Figure 3.40: Username or Password field is empty.

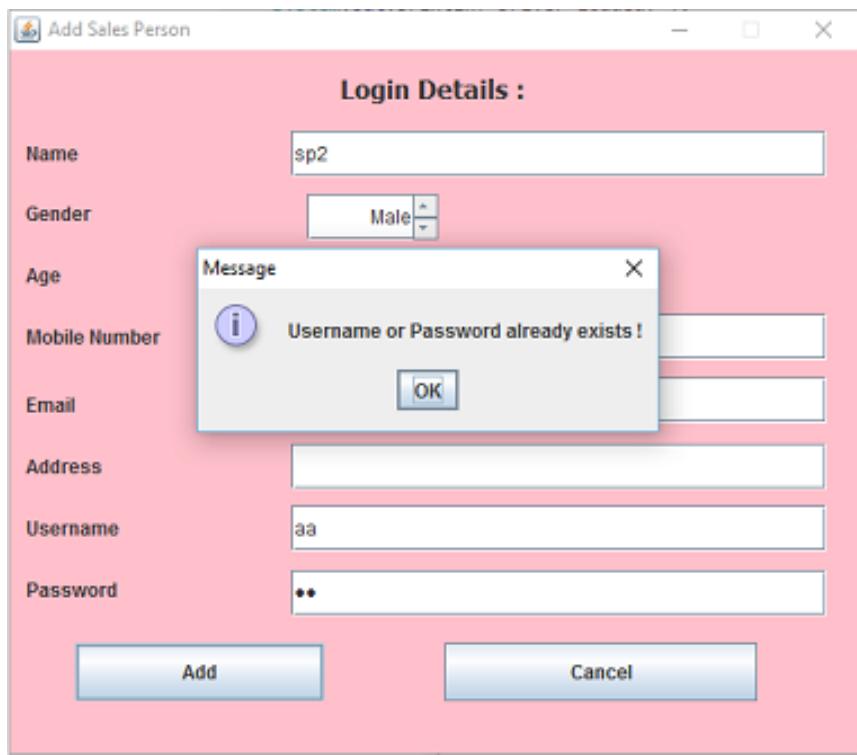


Figure 3.41: Username or Password already exists. Try something different.

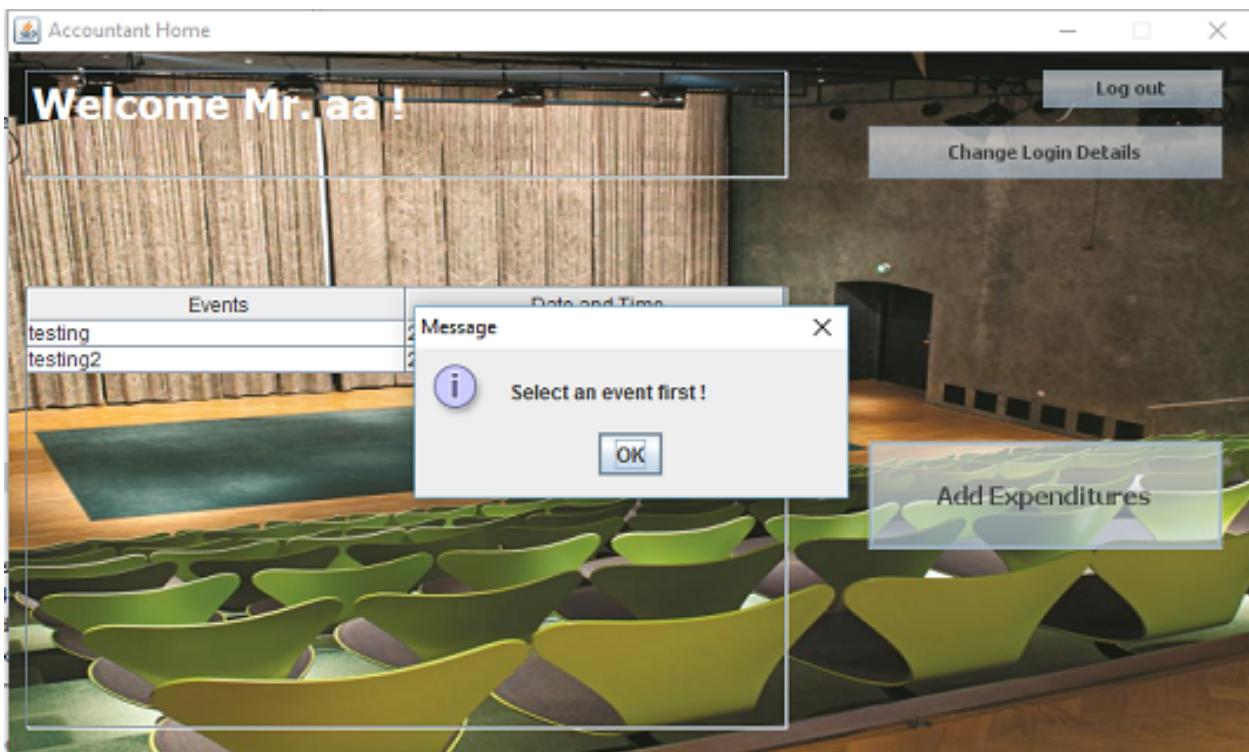
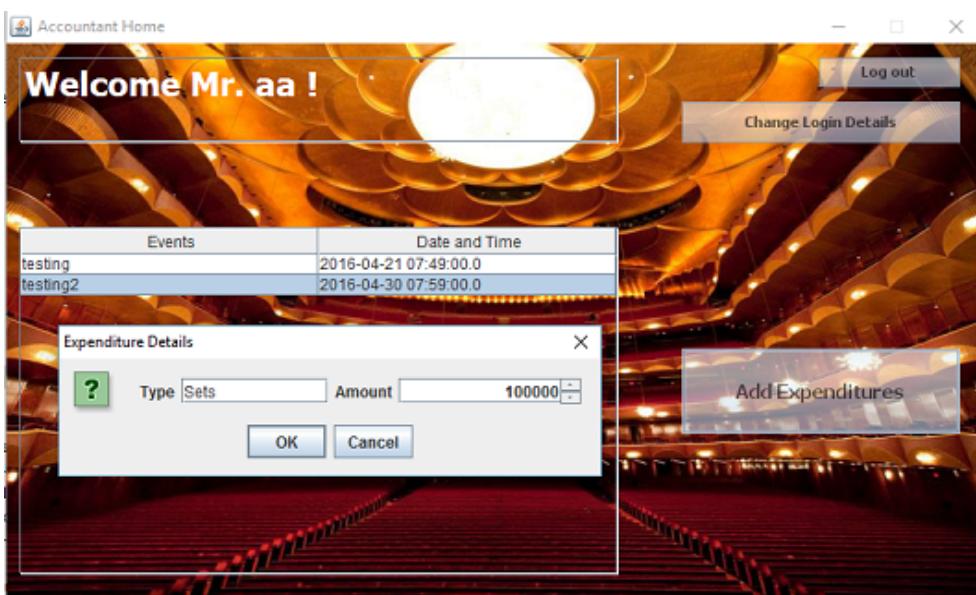


Figure 3.42: First Choose an Event then click on 'Add Expenditure'



Input :
-User enters Type and Amount of Expenditure.
Expected Output :
-The Expenditure gets updated accordingly.
Effective Results :
All Results match the expected results.

Figure 3.43: Expenditure Menu

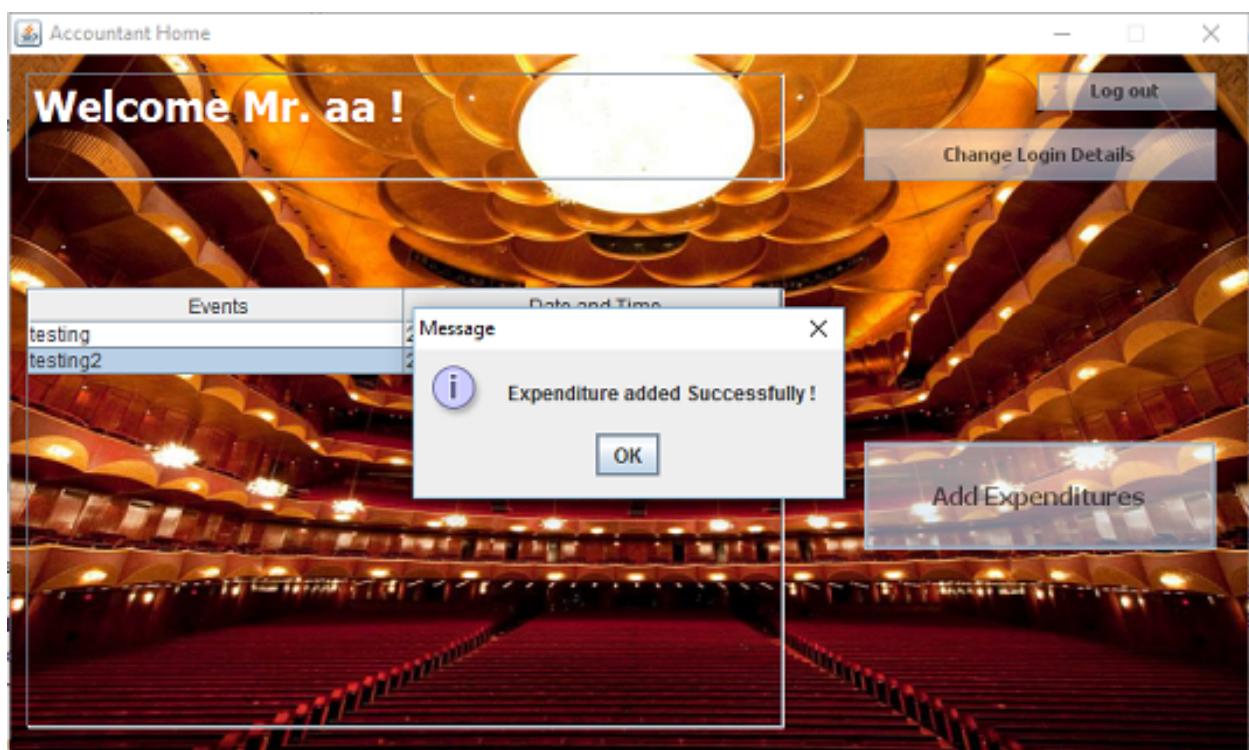


Figure 3.44: Expenditure Added Successfully.