Theater Reservation Final Report

short line

Grayson Salyers, Ben Smith, Katlyn Murphy

# 

# 

## Contents

### I. Introduction.

### II. System Description

### III. System Requirements

### A. Functional requirements

### B. Non-functional requirements

### C. Domain Requirements (optional)

### IV. Use Case Diagram

### V. Domain Diagram

### VI. Sequence Diagrams

### VII. State Diagram

### VIII. Activity Diagrams

### IX. Database Design

### A. ER Schema

B. Table Schema

### X. Conclusion

### XI. Data Dictionary

## I. Introduction

#### Problem Statement

The Lexington Theatre is experiencing a large amount of reservations over the phone recently and now needs an online system to help with the amount of reservations they are receiving.

#### Proposal

We propose a user-friendly software system to allow customers to make a reservation, cancel a reservation, register as a new member, change personal information, and check events and their performance days. When making a reservation, the system should allow the customer to choose an event, select seats, make a payment, and get a confirmation number.

## II. System Description

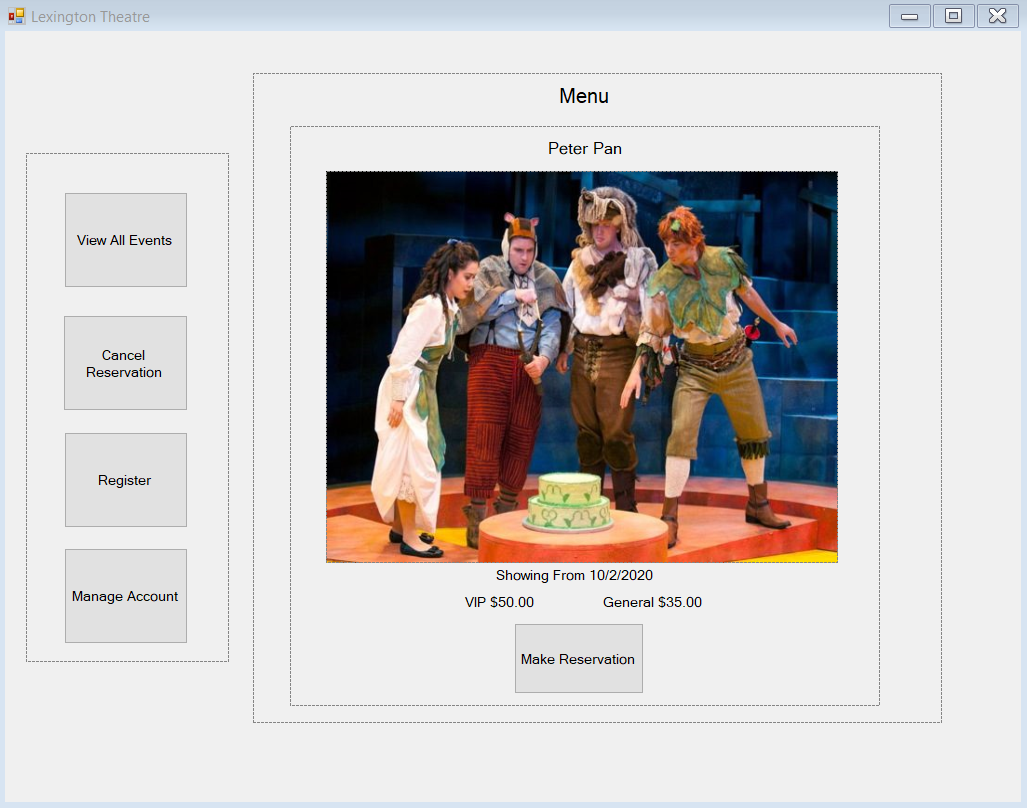
The online system allows members and non-members to make reservations online. The reservations shall be kept in a database. The customer should also be able to cancel a reservation with the system. The customer should be able to register as a member with the system. This will give them a 10% discount on tickets. The system should allow a customer to change their personal information. The system should also allow a customer to check events and their performance dates.

## III. System Requirements

#### Functional Requirements

##### R1. The system shall allow a customer to make a reservation for a performance.

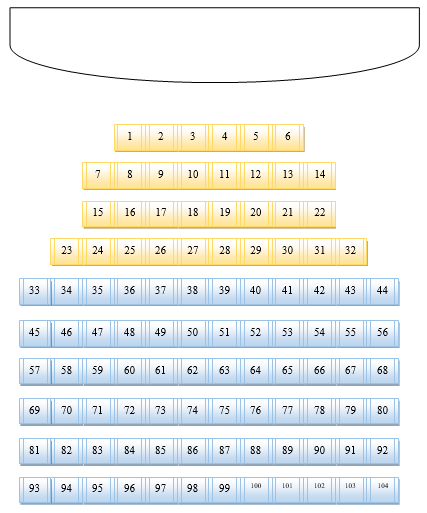
1.1 The system shall display a menu of events to the customer.



1.2 The customer shall click on the event they would like to attend.

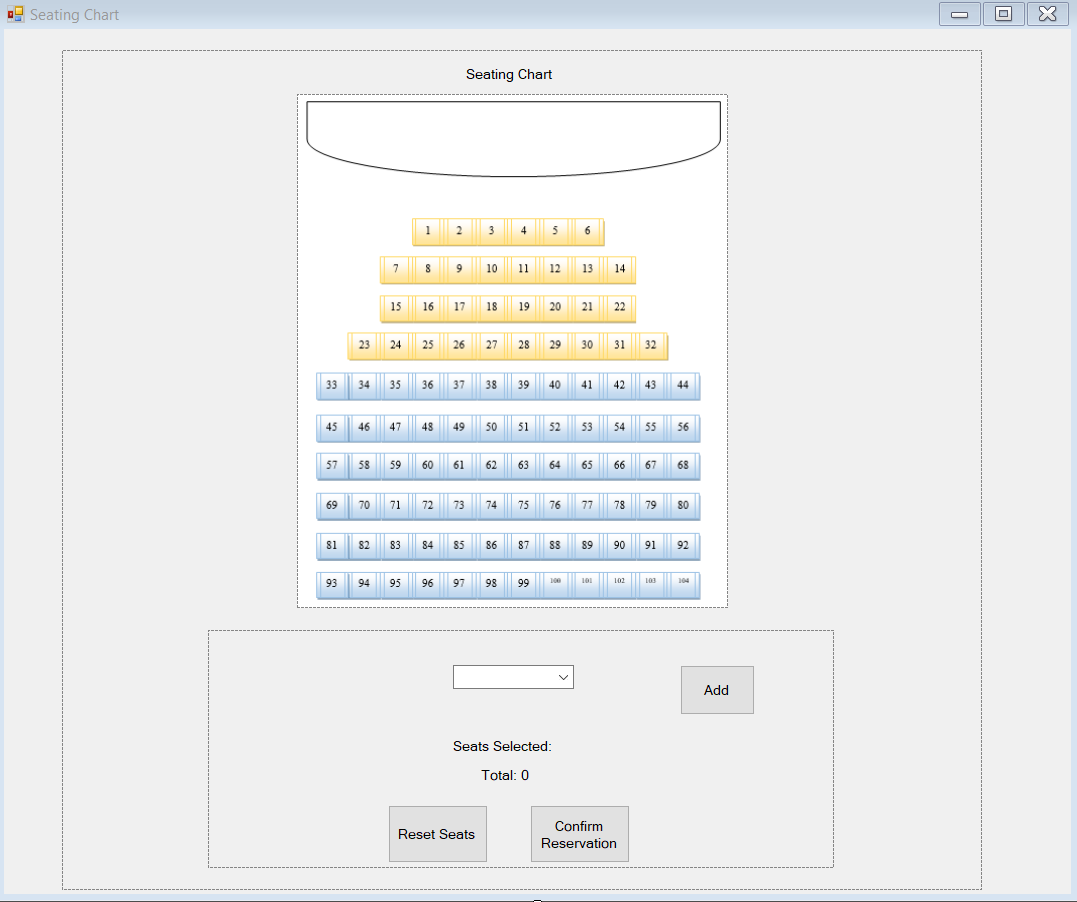


1.3 The system shall display the seating chart.



1.4 The user shall select up to four seats that they want.

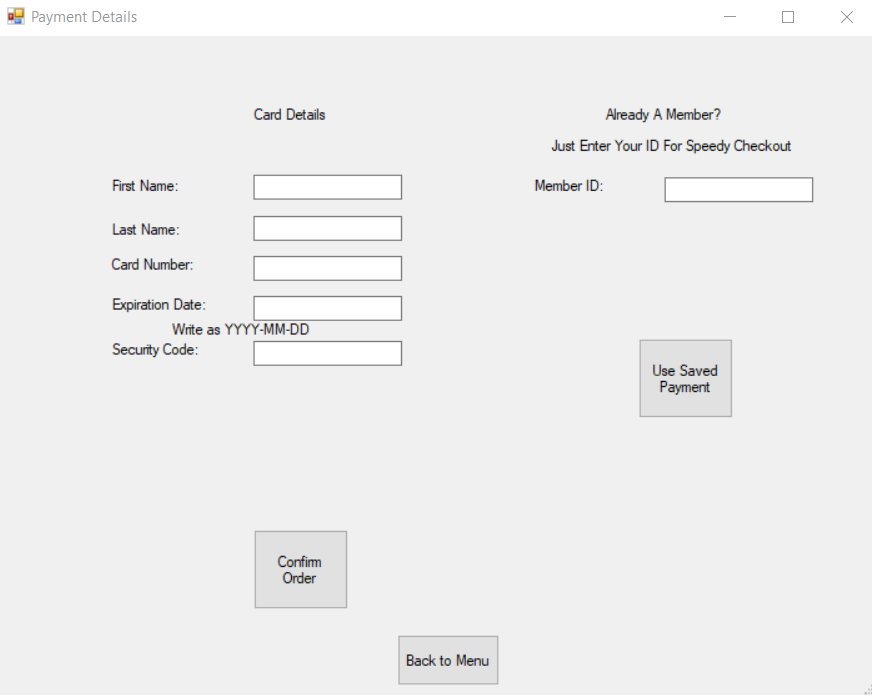
1.5 The user shall confirm the reservation.



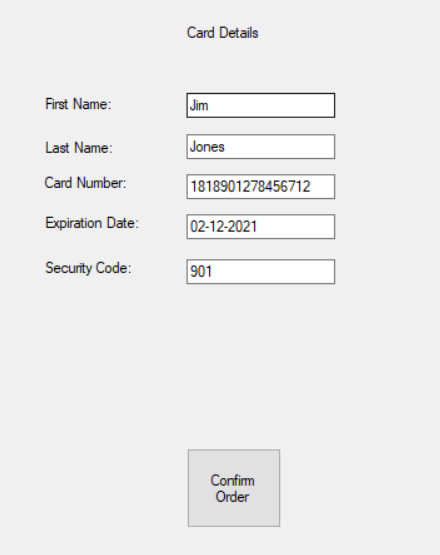
1.5.1 If the user has selected four seats the combo box will be unavailable to them.



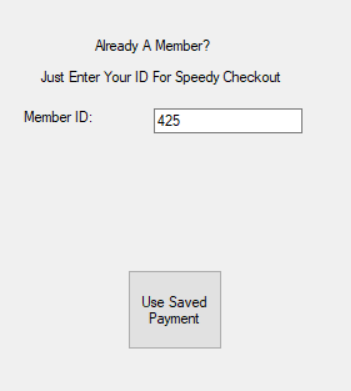
1.6 The system shall request payment from the customer or the membership ID.



1.7 The user shall enter their payment information.

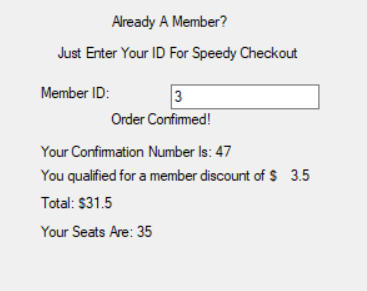


1.7.1 The user can enter in their member ID for automatic payment which shall give them a 10% discount.

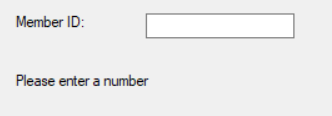


1.8 The system shall validate information the user has entered.

1.9 The system shall display their confirmation number for their order.

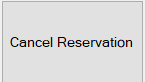


1.9.1 If the user tries to checkout without entering in either their Member ID or their card detail this error message will be displayed.

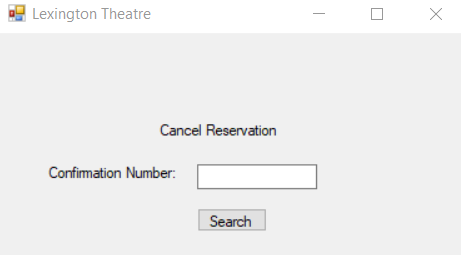


##### R2. The system shall allow a customer to cancel a reservation for a performance.

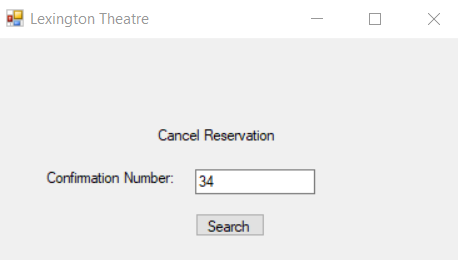
2.1 The customer shall click the “Cancel Reservation” button.



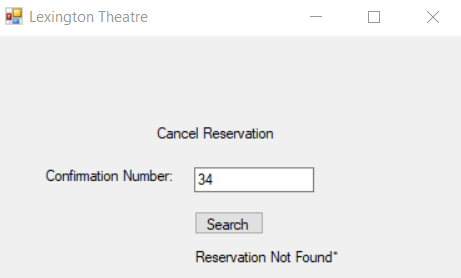
2.2 The system shall display a Cancel Reservation menu.



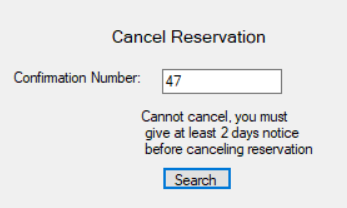
2.3 The customer shall input their reservation confirmation number.



2.3.1 If no reservation is found the system shall display “Reservation Not Found”.



2.3.2 If the user tries to cancel a reservation with less than two days notice this error will appear.



2.4 The system shall search for the reservation corresponding to the customer’s

confirmation number.

2.5 The system shall display the customer’s reservation and ask them to confirm

cancellation.

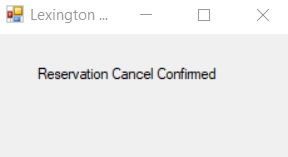


2.6 The customer shall confirm cancellation.



2.7 The system shall cancel the reservation and make the seats available again

and refund their card.

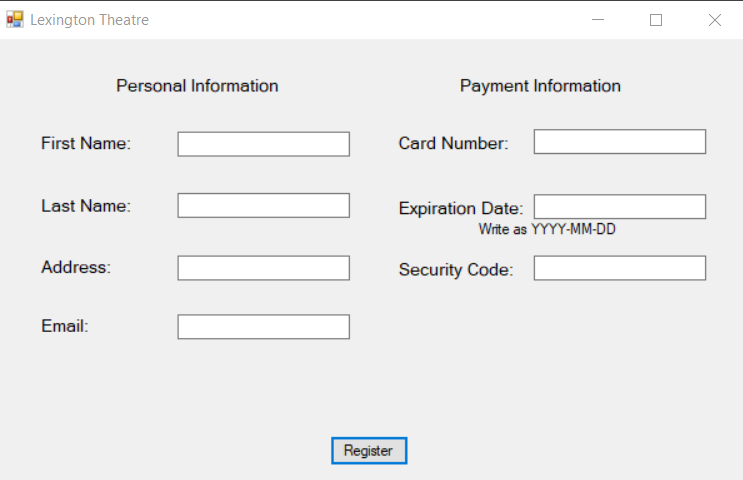


##### R3. The system shall allow a customer to register as a new member.

3.1 The customer shall click the “Register” button.

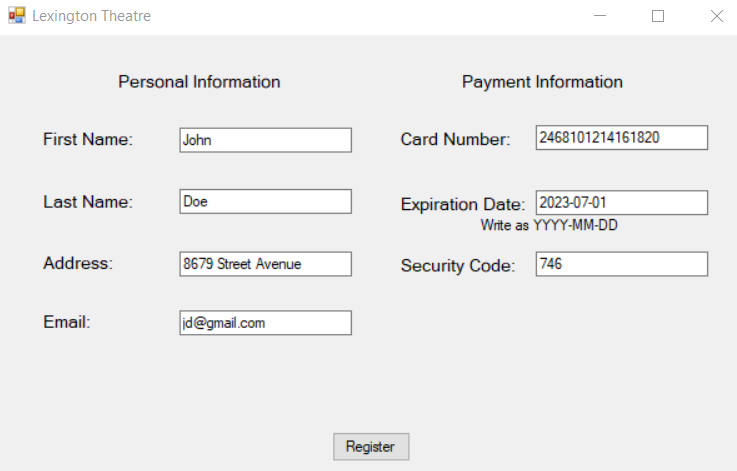


3.2 The system shall display the “Member Registration Form”.



3.3 The customer shall input all of the required information (name, address,

phone number, email, and payment information if they wish).



3.4 The user shall press the “Register” button.



3.5 The system shall make sure the person registering does not already have an

Account.

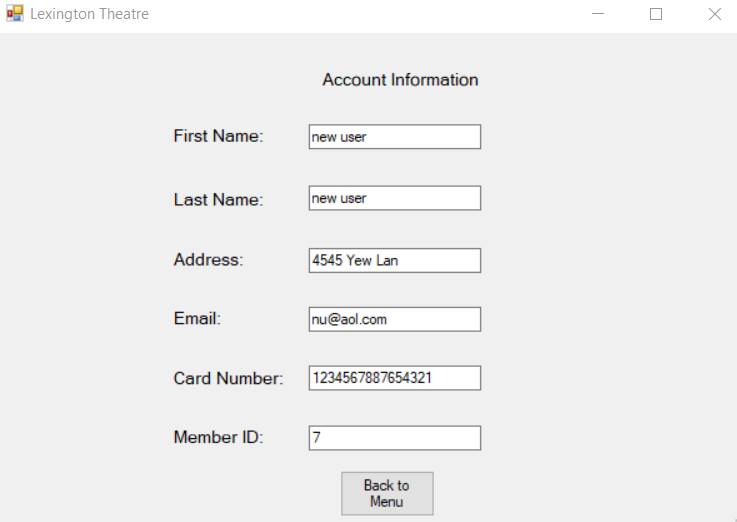
3.5.1 If the account already exists, the system shall display an “Account

Registration Failed” message.



3.6 If the account does not exist, the system shall generate and display

an ID number to the user along with their information.

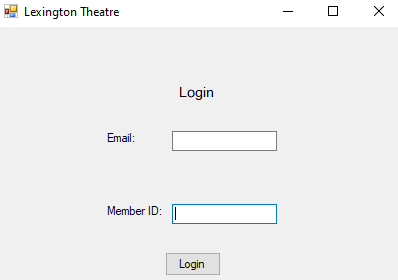


##### R4. The system shall allow a customer to change the personal information of their account.

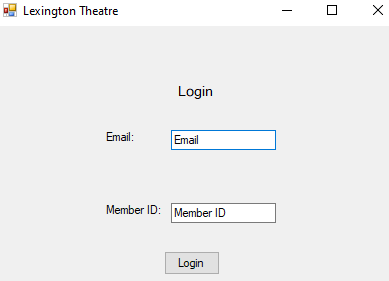
4.1 The customer shall click the “Update Account” button.



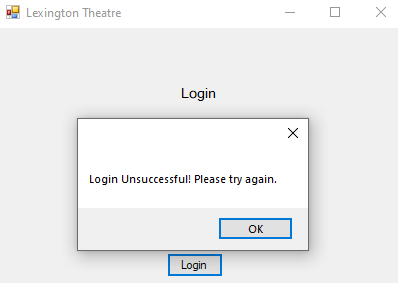
4.2 The system shall request login information.



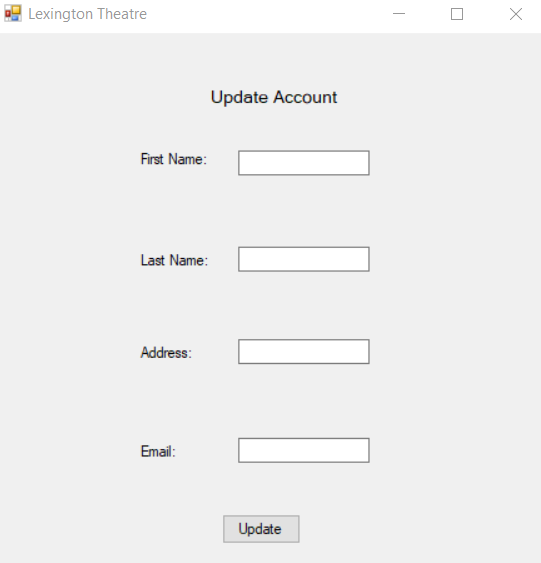
4.3 The customer shall login into the system.



4.3.1 If login was unsuccessful the system will display a “Login Unsuccessful” message.



4.4 The system shall validate the account, and ask the user to fill out the appropriate text fields to update.



4.5 The customer shall update all fields and press the “Update” button.



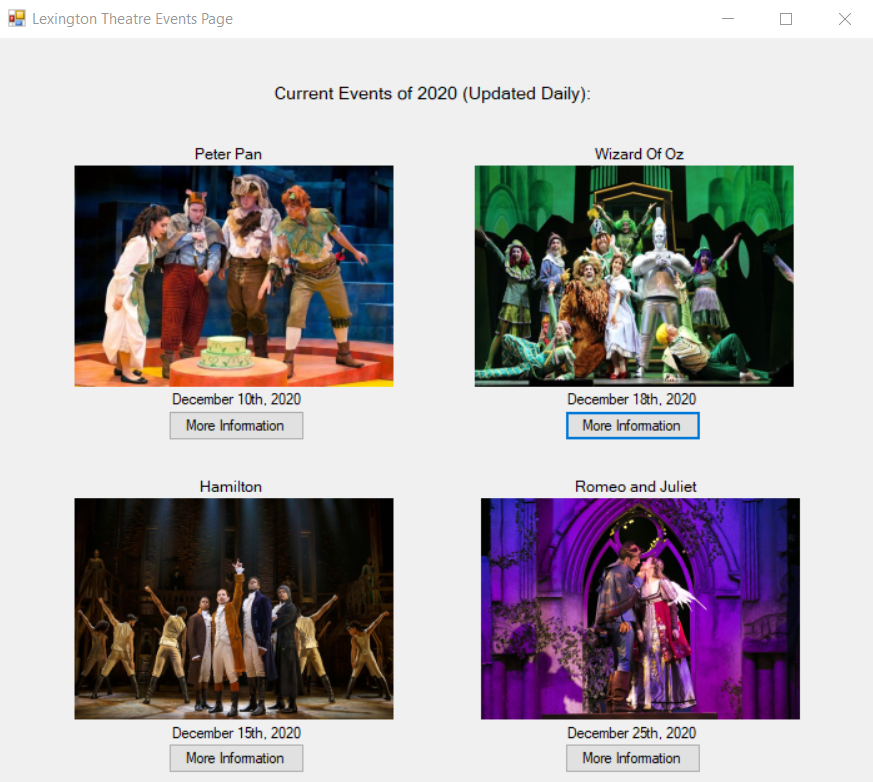
4.6 The system shall update all information and update the database.

##### R5. The system shall allow a customer to check events and their dates.

5.1 The customer shall click the “View Events” button.



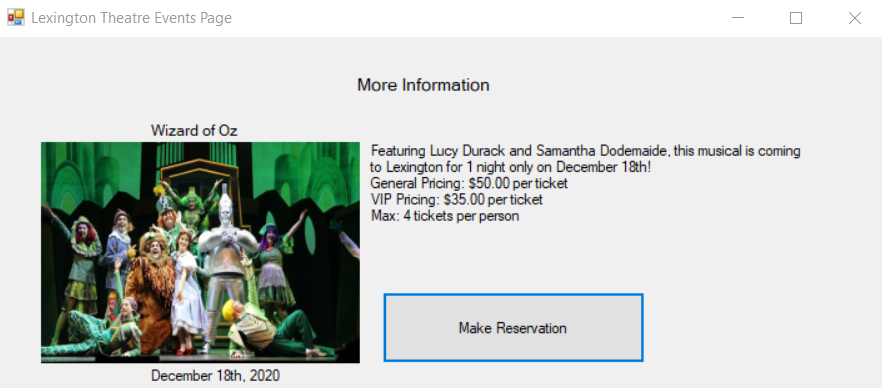
5.2 The system shall display the list of events that are scheduled up to two months to the customer including a name, picture, and dates.



5.3 The customer shall click “More Information” to view additional information such as description, pricing, and availability.



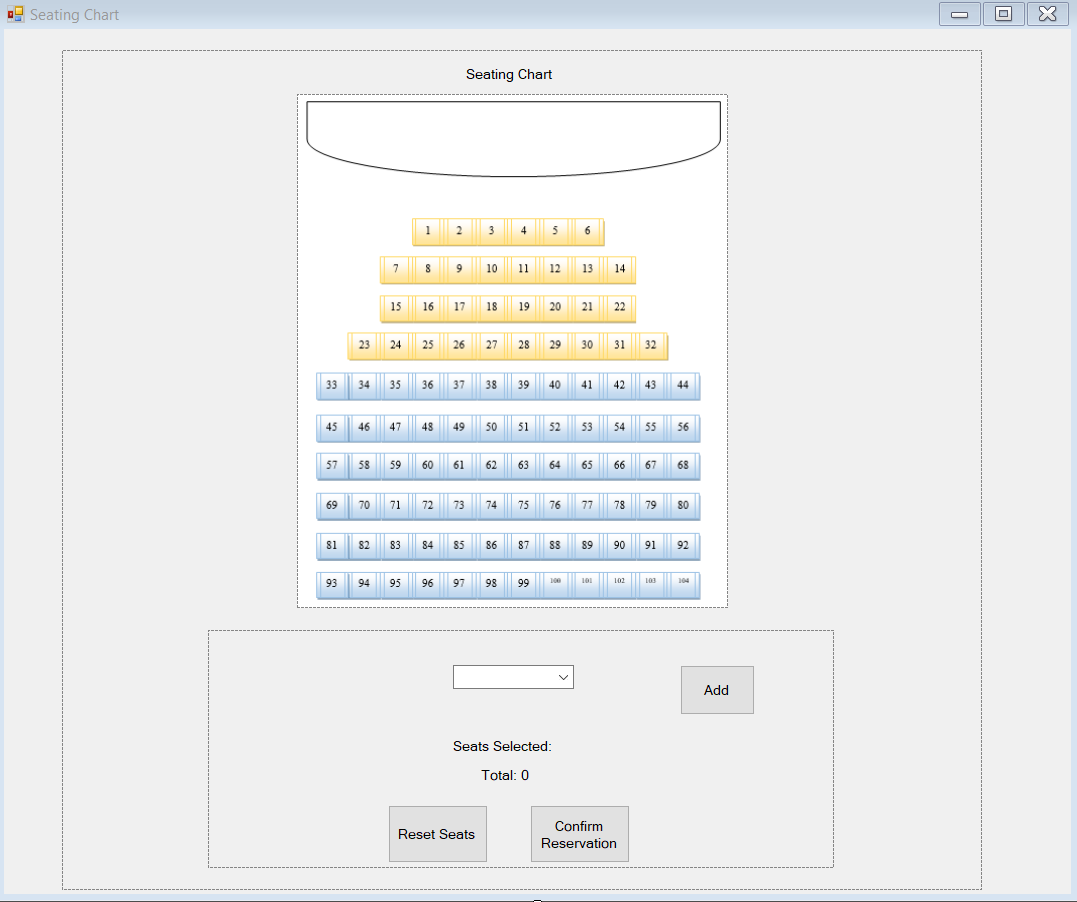
5.4 The system shall display the additional information such as event description and general pricing.



5.5 The customer shall press the “Make Reservation” button.



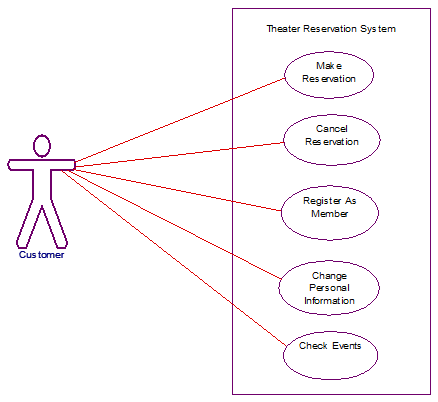
5.6 The system shall display the seating chart for the event and let the customer continue through with selecting seats and making payments for the selected event.



#### Non-Functional Requirements

#### Domain Requirements (Optional)

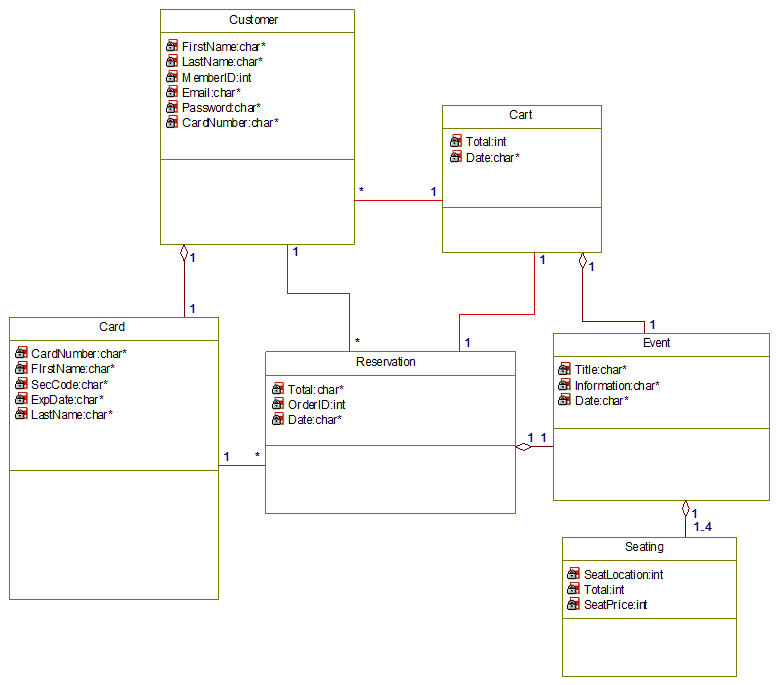
## IV. Use Case Diagram



This use case is to describe the services provided to the customers of the theater:

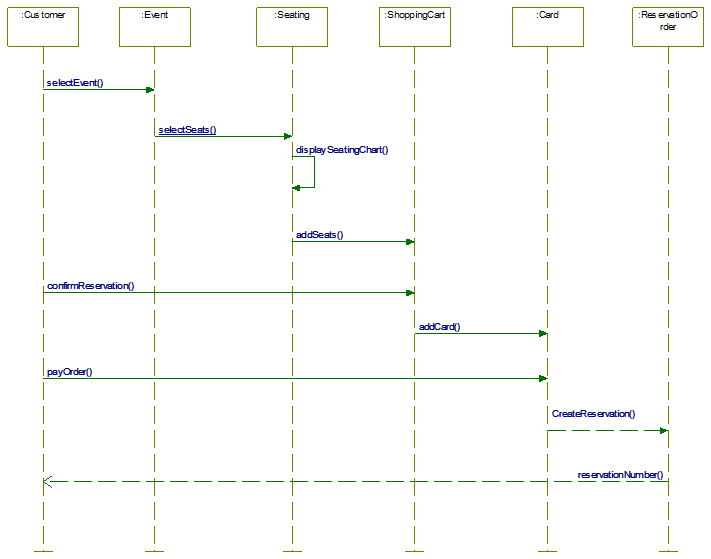
1. Use Case Make Reservation: The system shall allow a customer to make a reservation for a performance.
2. Use Case Cancel Reservation: The system shall allow a customer to cancel a reservation for a performance.
3. Use Case Register As Member: The system shall allow a customer to register as a new user.
4. Use Case Change Personal Information: The system shall allow a customer to change the personal information of their account.
5. Use Case Check Events: The system shall allow a customer to check events and their dates.

## V. Domain Diagram

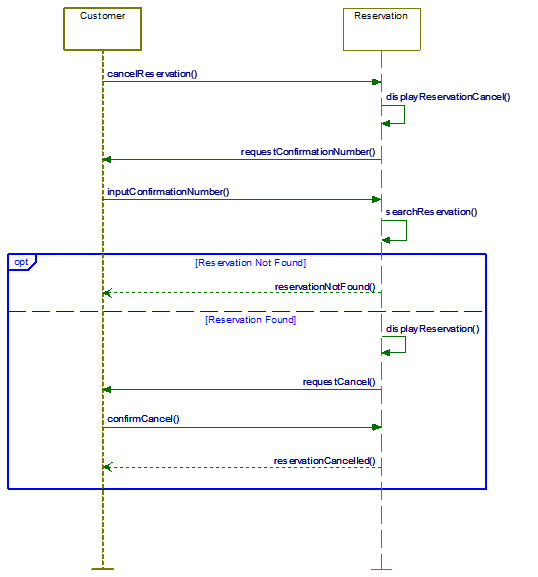


## VI. Sequence Diagrams

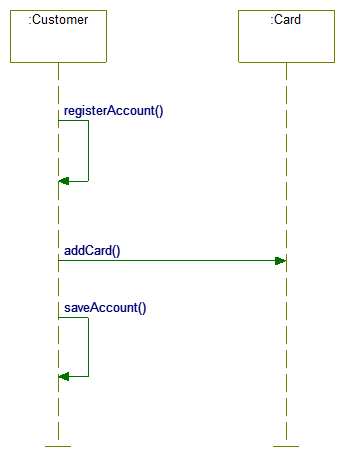
Use Case: Make Reservation



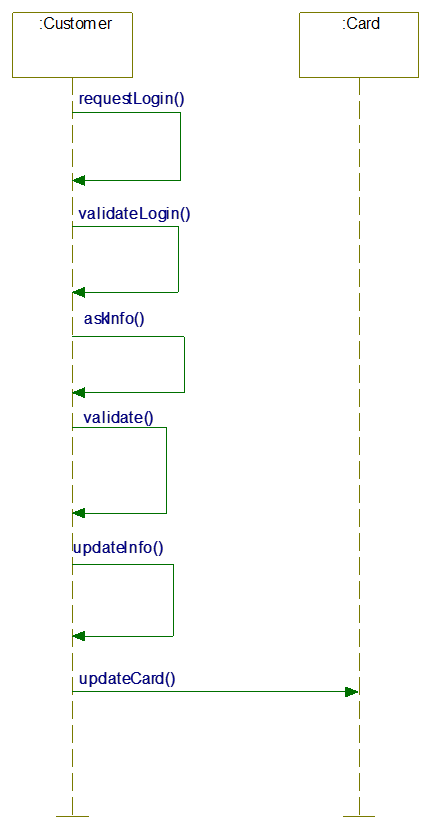
Use case: Cancel Reservation



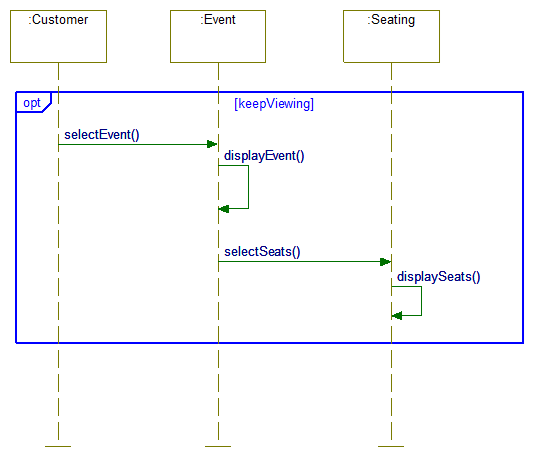
Use case: Register As Member



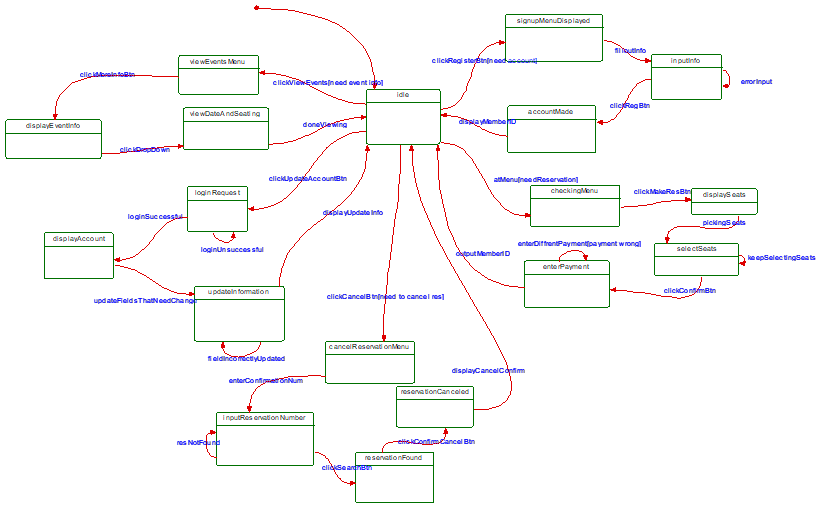
Use case: Change Personal Information



Use case: Check Events

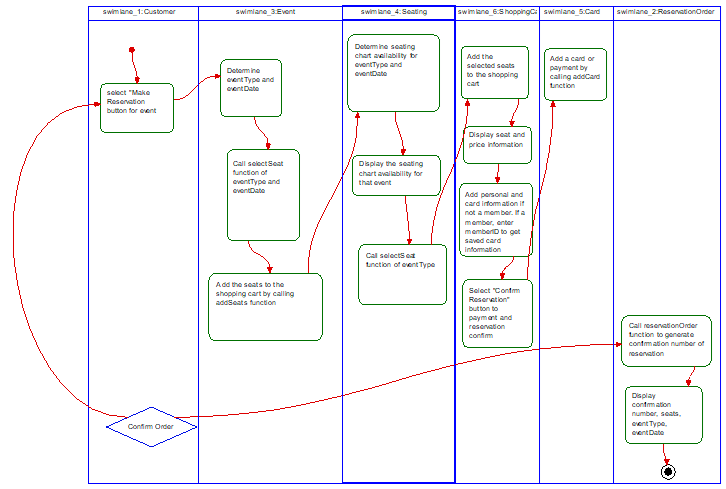


## VII. State Diagram

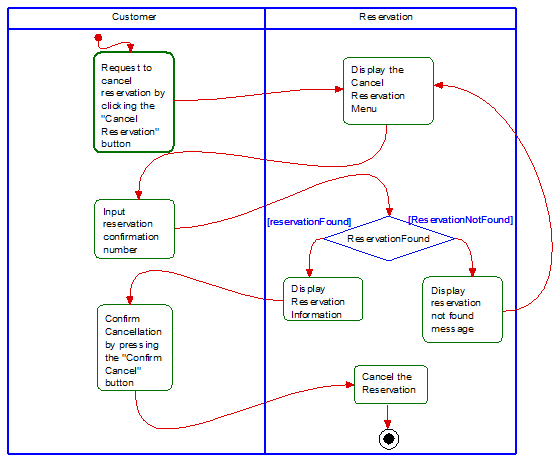


## VIII. Activity Diagrams

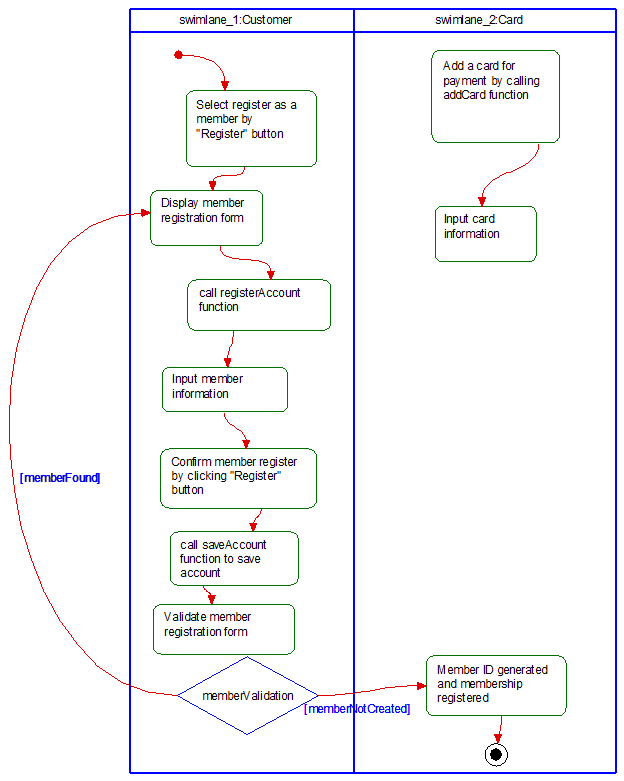
Make a Reservation Use Case:



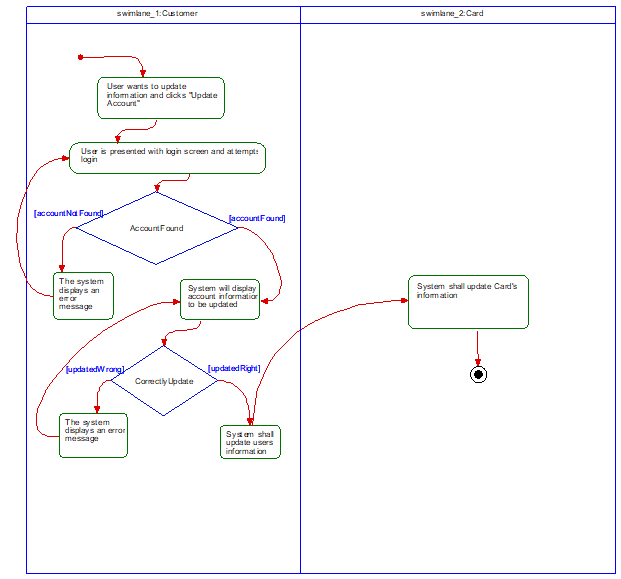
Cancel Reservation Use Case:



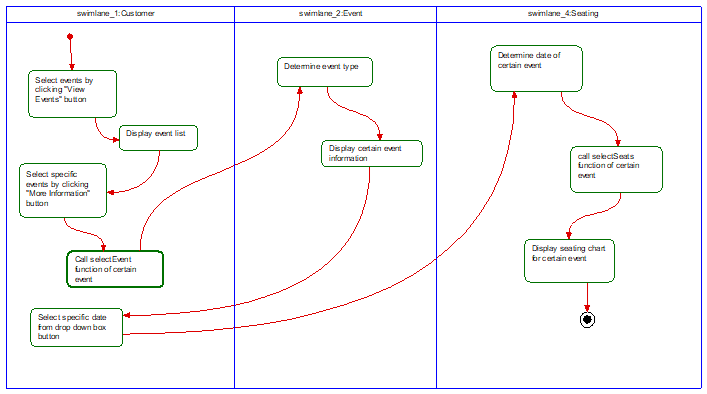
Register As Member Use Case:



Update Account Use Case:

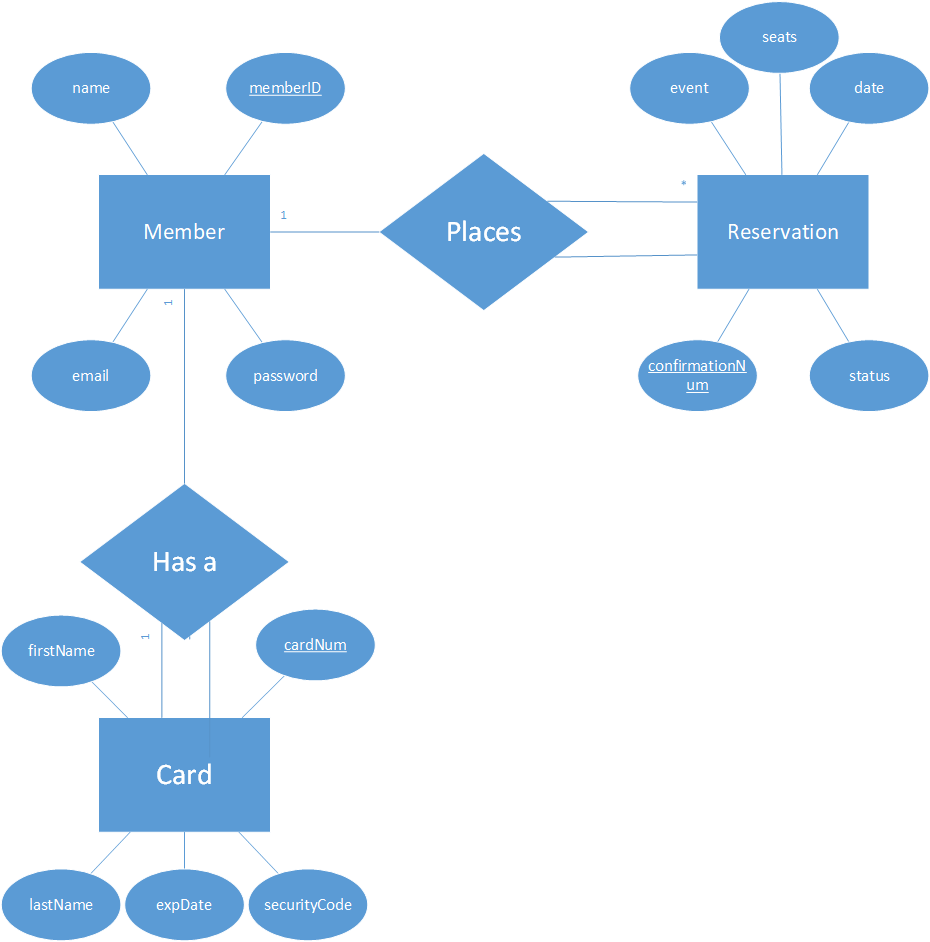


View Events Use Case:



## IX. Database Design

1. Er Schema



1. Table Schema

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| cardNum | firstName | lastName | expDate | securityCode | memberID  (foreign key) |

Card

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| memberID | name | address | email | cardNum  (foreign key) |

Customer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| confirmationNum | event | date | seats | memberID |

Reservation

## X. Conclusion

In conclusion we should be able to do all that is requested from the Lexington Theatre. With their key requests being that customers be able to make a reservation for a performance, let said customer cancel a reservation, be allowed to register as a member, change their membership information, and finally to allow the user to check upcoming events and dates. We will be able to complete their request all within the given frame time and budget.

## XI. Data Dictionary

Customer: A class in the system which creates an account with the customer’s personal information after confirmation. This is then saved in the database to access.

Cart: A class in the system which stores the type of play and the seats that the user has selected.

Reservation: A class in the system which generates a confirmation number when seats have been added and confirmed by payment. This is then saved in the database to access.

Card: A class in the system which stores a members card information.

Event: A class in the system which has details about the type play that is showing, this includes a brief description of the event, title, and the date the event will show.

Seating: A class in the system which has the seating location, seating price, and the total.