# ENSF 614

## Advanced System Analysis and Software Design

Fall 2022

## Design Package


**Group 1:**

Stewart Pratt - 30073940

Robert Njie - 30020243

Justin Nguyen - 30042258

Florian Bache - 10075304

Fizzah Malik - 10122276


**Submission Date:**

December 4, 2022

# Table of Contents


**System Use Case Diagram** 3

**System Activity Diagram** 4

**Scenarios** 5

- Use Case 1: Login 5
- Use Case 2: Sign up 5
- Use Case 3: Select Theatre 6
- Use Case 4: Search Movie 6
- Use Case 5: Select Showtime 6
- Use Case 6: Select Seat(s) 6
- Use Case 7: Make Payment 7
- Use Case 8: Cancel Ticket 7
- Use Case 9: Email Receipt and Ticket 8
- Use Case 10: Pay Account Fee 8
- Use Case 11: Email Movie News 9
- Use Case 12: Update Database 9

**List of Candidate Objects from Use Case Scenarios** 10

- Nouns 10
- Verbs 11

**State Transition Diagrams** 13

- Ticket - Justin Nguyen 13
- Payment - Stewart Pratt 14
- Cancel Ticket - Fizzah Malik 15
- Email Receipt and Ticket - Robbie Njie 15
- Login - Florian Bache 16

**System Interaction Diagrams** 17

- Login - Justin Nguyen 17
- Sign Up - Stewart Pratt 18
- Make Payment - Florian Bache 18
- Cancel Ticket - Fizzah Malik 19
- Select Seat(s) - Robbie Njie 19

**Design Level Class Diagram** 20

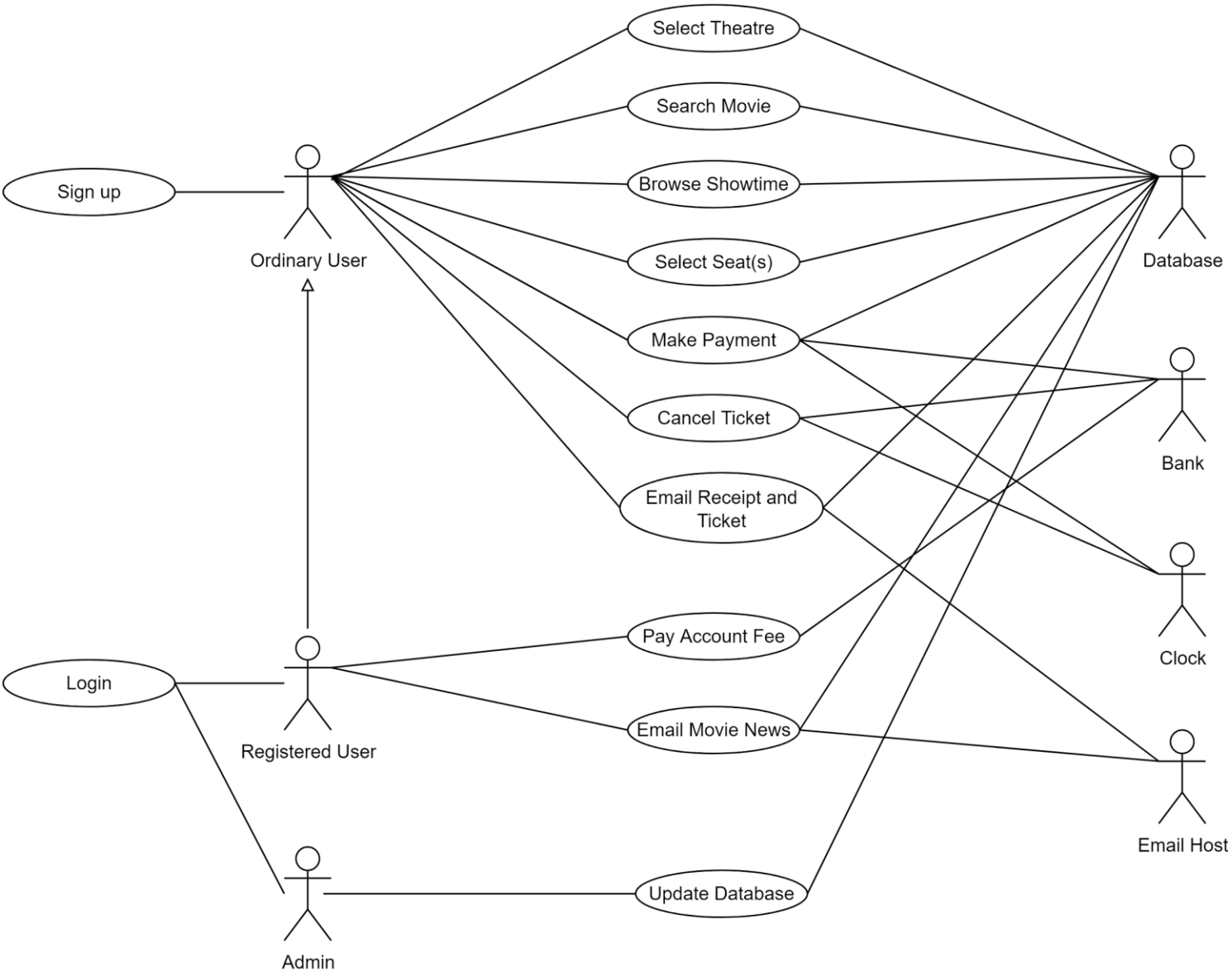
- Class Diagram 21
- Theatre classes 21

Showtime classes 21
Movie classes 22
SeatMap classes 22
Seat classes 23
User classes 23
Ticket classes 24
Payment classes 24
Email classes 25
Admin classes 25

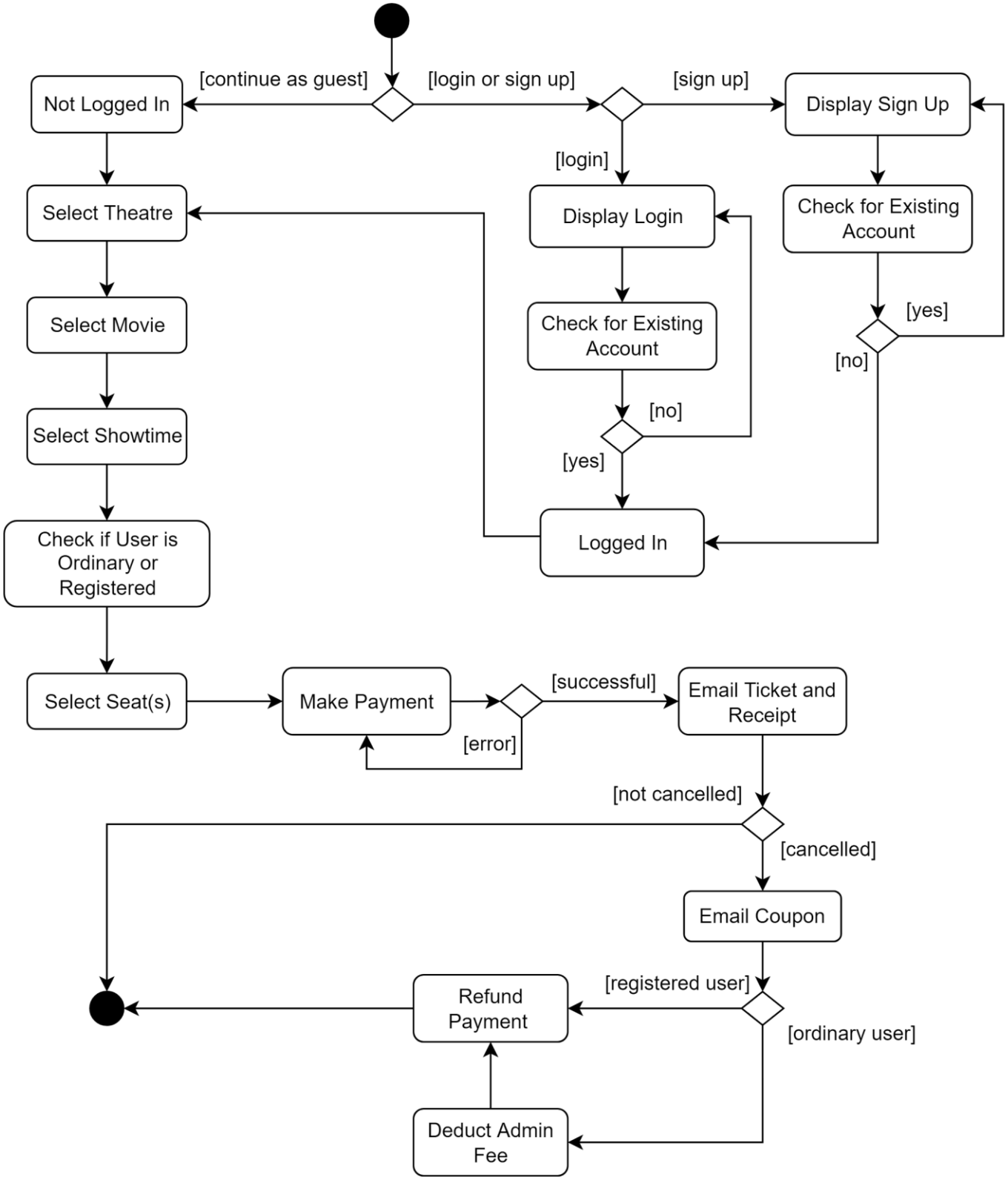
**Package Diagram** 26

**Deployment Diagram** 27

# System Use Case Diagram



# System Activity Diagram



# Scenarios

## Use Case 1: Login

This scenario starts when a registered_user or admin wants to login to the app by $clicking$ on Login.

A LoginView window appears with two text fields for EmailAddress and Password. The window also displays two buttons, Submit and Sign_up.

The registered_user then enters their Email Address and Password and $clicks$ the Submit button.

An admin will enter their Username and Password and $click$ the Submit button.

The system then $checks$ the given user information against the Database to see if the user exists. If successful, close the popup window, otherwise display an error message.

## Use Case 2: Sign up

This scenario starts when an ordinary_user wants to sign up for an account by $clicking$on Sign_up.

A SignupView window appears with five text fields for FirstName, EmailAddress, Password, ConfirmPassword, CardholderName, CreditCardNo, ExpiryDate, and CVV. The window also displays a button called Submit.

The ordinary_user then enters their information and $clicks$ the Submit button.

The system then $checks$ the given user information against the Database to see if the registered_user already exists. If registered_user does not exist, $charge$ the CreditCard the annual fee, otherwise display an error_message. If payment is successful, close the window, otherwise display an error_message.

## Use Case 3: Select Theatre

This scenario starts when a user $selects$ the Theatres_tab.

Then the user can $scroll\ through$ the Theatre_list and $select$ a Theatre.

A Movie_list will appear on the screen and a user will $select$ a Movie.

## Use Case 4: Search Movie

This scenario starts when a user $selects$ a Theatre.

Then the user can $scroll\ through$ the Movie_list to $search$ for a Movie and $select$ it.

A Showtime_list will $display$ and the user can $select$ a Showtime.

## Use Case 5: Select Showtime

This scenario starts when a user $selects$ a Movie.

Then the user can $scroll\ through$ the Showtime_list and $select$ a Showtime.

A Seat_map will $display$ and the user will $select$ a number of Seats.

## Use Case 6: Select Seat(s)

This scenario starts when a user has $selected$ a Theatre, Movie, and a Showtime.

The user will then be shown a Seat_map and the Theatre, Movie and Showtime.

The user will then $select$ available_seats.

The user will then $select$ proceed button.

## Use Case 7: Make Payment

This scenario starts when a user has *selected* a Theatre, Movie, Showtime, and available_seats and *clicked* proceed.

The user will be shown a payment_page where the Showtime, Theatre, Movie and Total_cost will be *displayed*.

The user will also be shown CardholderName, CreditCardNo, ExpiryDate, and CVV as input fields to make a payment and an input field EmailAddress to *receive* confirmation.

The user will *enter* their information in the input fields.

The user will then *select* confirm and pay.

The system will *verify* that the payment was successful.

The user will then be *displayed* a confirmation page of the ticket information and payment summary.

## Use Case 8: Cancel Ticket

This scenario can start when a user *selects* cancel_button.

The user will *enter* their ticket_information into the cancel_window. The ticket_information will be *retrieved* from the Database. This includes theatre, movie, showtime, and seat.

If the cancel_button was *clicked* is more than 72 hours before showtime, a popup confirmation window will be *displayed* to the user. If user *presses* decline or the hour to showtime is less than 72 hours the system will exit use case.

If the user will *selects* confirm_and_cancel the user group will be *checked*.

If the user is an ordinary_user, the system *checks* the total_cost of the Ticket and 85% of full payment is *returned* to the user as credit.

If the user is an registered_user, the system *checks* the total_cost of the Ticket and the full payment is *returned* to the user as credit.

The system will then *update* the Database to *add* the credit and Date of Cancellation as well as *remove* the Ticket.

## Use Case 9: Email Receipt and Ticket

This scenario starts when a user has *selected* confirm_and_pay or confirm_and_cancel.

If the user has *selected* confirm_and_pay. The system *retrieves* the user EmailAddress, Ticket, and Receipt from the Database.

The Ticket and Receipt is then *emailed* to the user EmailAddress.

If the user has *selected* confirm_and_cancel. The system *retrieves* the user EmailAddress, Receipt from the Database.

The Receipt is then *emailed* to the user EmailAddress.

## Use Case 10: Pay Account Fee

This scenario starts when a new day begins. The Database is *checked* to see if it has been one year since a registered_user first registered.

If such a registered_user is *found*, their EmailAddress, CardholderName, CreditCardNo, ExpiryDate, and CVV is *retrieved* from the Database.

Payment *charges* $20 to their CreditCard.

The EmailHost sends an Email to the registered user containing the Receipt.

## Use Case 11: Email Movie News

This scenario starts when MovieList is $updated$ with a new Movie.

The new Movie's information is $written$ into an Email.

The EmailAddresses of all registered users are $retrieved$ from the Database. They are $sent$ the Email with the Movie news.

## Use Case 12: Update Database

This scenario starts when an admin $clicks$ add_movie, remove_movie, add_registered_user, remove_registered_user, add_staff, or remove_staff on the adminView.

The appropriate object will be $added$ or $removed$ from the Database. The admin will then be $notified$ of success.

# List of Candidate Objects from Use Case Scenarios

## Nouns

- registered_user (Use Case 1, 2, 8, 10, 11)
- admin (Use Case 1, 12)
- Login (Use Case 1)
- LoginView (Use Case 1)
- EmailAddress (Use Case 1)
- Password (Use Case 1)
- Submit (Use Case 1, 2)
- Sign_up (Use Case 1)
- Database (Use Case 1, 2, 8, 9, 11, 12)
- ordinary_user (Use Case 2, 8)
- Sign_up (Use Case 2)
- SignupView (Use Case 2)
- FirstName (Use Case 2)
- EmailAddress (Use Case 2, 7, 9, 10, 11)
- Password (Use Case 2)
- ConfirmPassword (Use Case 2)
- CardholderName (Use Case 2, 7, 10)
- CreditCardNo (Use Case 2, 7, 10)
- ExpiryDate (Use Case 2, 7, 10)
- CVV (Use Case 2, 7, 10)
- CreditCard (Use Case 2, 10)
- error_message (Use Case 1, 2)
- user (Use Case 3, 4, 5, 6, 7, 8, 9)
- Theatres_tab (Use Case 3)
- Theatre_list (Use Case 3)
- Theatre (Use Case 3, 4, 6, 7, 8)
- Movie_List (Use Case 3, 4, 11)
- Movie (Use Case 3, 4, 5, 6, 7, 8, 11)
- Showtime_List (Use Case 4, 5)
- Showtime (Use Case 4, 5, 6, 7, 8)
- Seat_map (Use Case 5, 6)
- Seats (Use Case 5, 8)
- available_seats (Use Case 3, 6, 7)
- proceed (Use Case 6, 7)
- payment_page (Use Case 7)
- Total_cost (Use Case 7, 8)

- Payment (Use Case 7, 8, 10)
- confirmation (Use Case 7)
- information (Use Case 7)
- input_fields (Use Case 7)
- confirm_and_pay (Use Case 7, 9)
- ticket_information (Use Case 7)
- payment_summary (Use Case 7)
- cancel_button (Use Case 8)
- ticket_information (Use Case 8)
- confirm_and_cancel (Use Case 8, 9)
- credit (Use Case 8)
- Date_of_Cancellation (Use Case 8)
- Ticket (Use Case 8, 9)
- Receipt (Use Case 9, 10)
- Email (Use Case 10, 11)
- EmailHost (Use Case 10)
- add_movie (Use Case 12)
- remove_movie (Use Case 12)
- add_staff (Use Case 12)
- remove_staff (Use Case 12)
- add_registered_user (Use Case 12)
- remove_registered_user (Use Case 12)

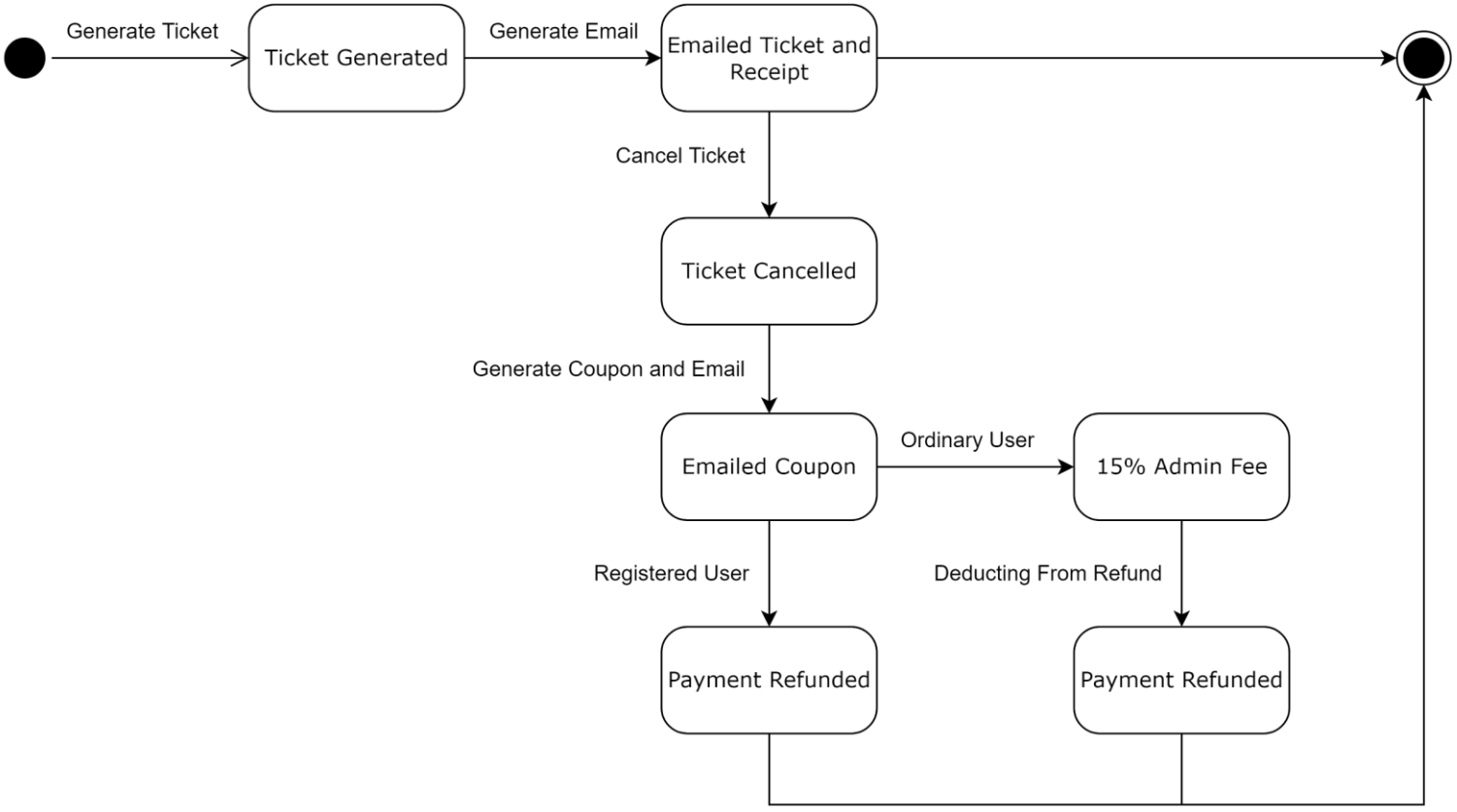
## Verbs

- click (Use Case 1, 2, 7, 8, 12)
- check (Use Case 1, 2, 8, 10)
- charge (Use Case 2, 10)
- select (Use Case 3, 4, 5, 6, 7, 8, 9)
- scroll through (Use Case 3, 4, 5)
- display (Use Case 3, 4, 5, 7, 8)
- search (Use Case 4)
- receive (Use Case 7)
- enter (Use Case 7, 8)
- verify (Use Case 7)
- retrieve (Use Case 8, 9, 10, 11)
- press (Use Case 8)
- return (Use Case 8)
- update (Use Case 8, 11)
- add (Use Case 8, 12)
- remove (Use Case 8, 12)

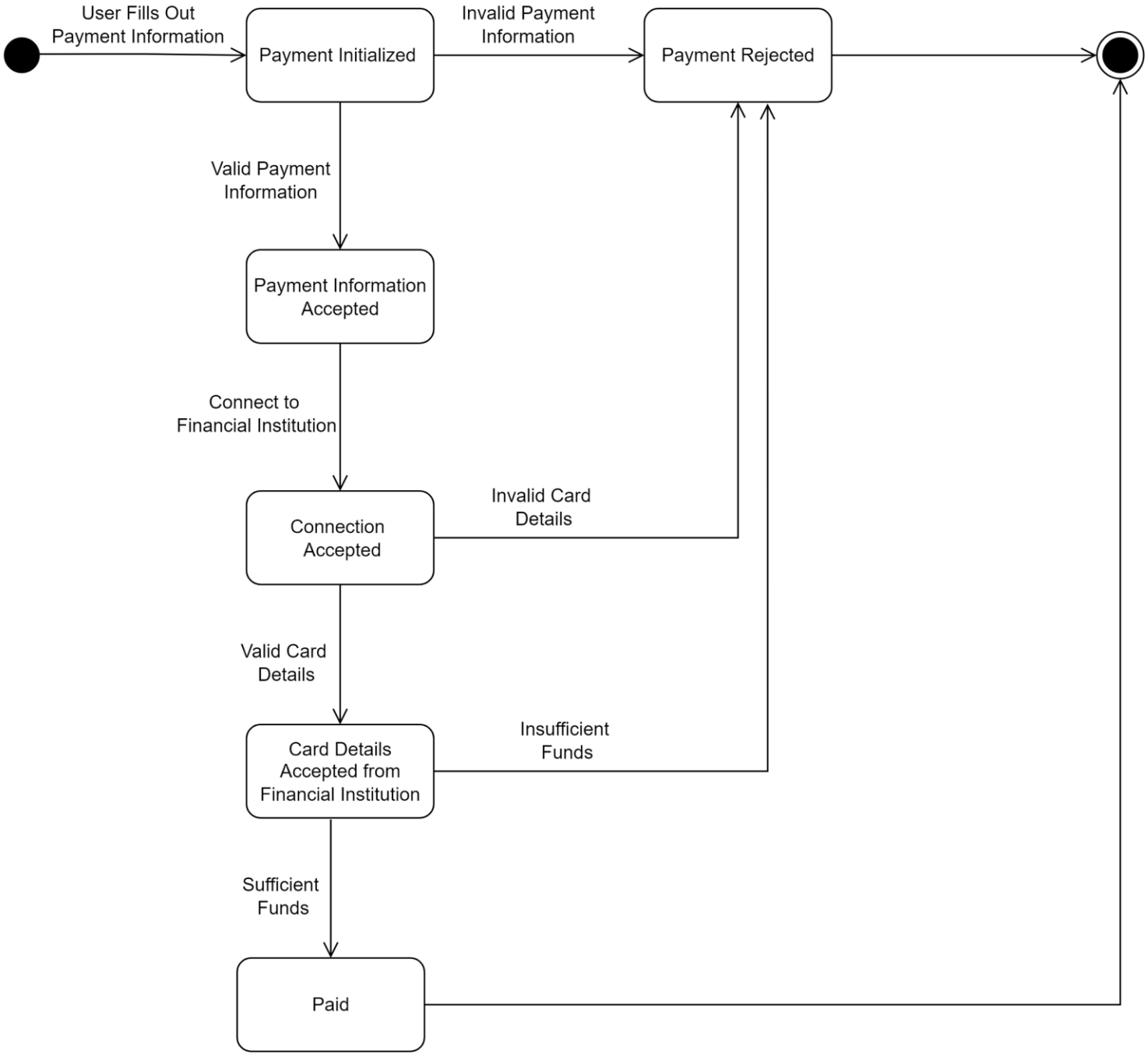
- email (Use Case 9)
- charge (Use Case 10)
- write (Use Case 11)
- send (Use Case 11)
- notify (Use Case 12)

# State Transition Diagrams

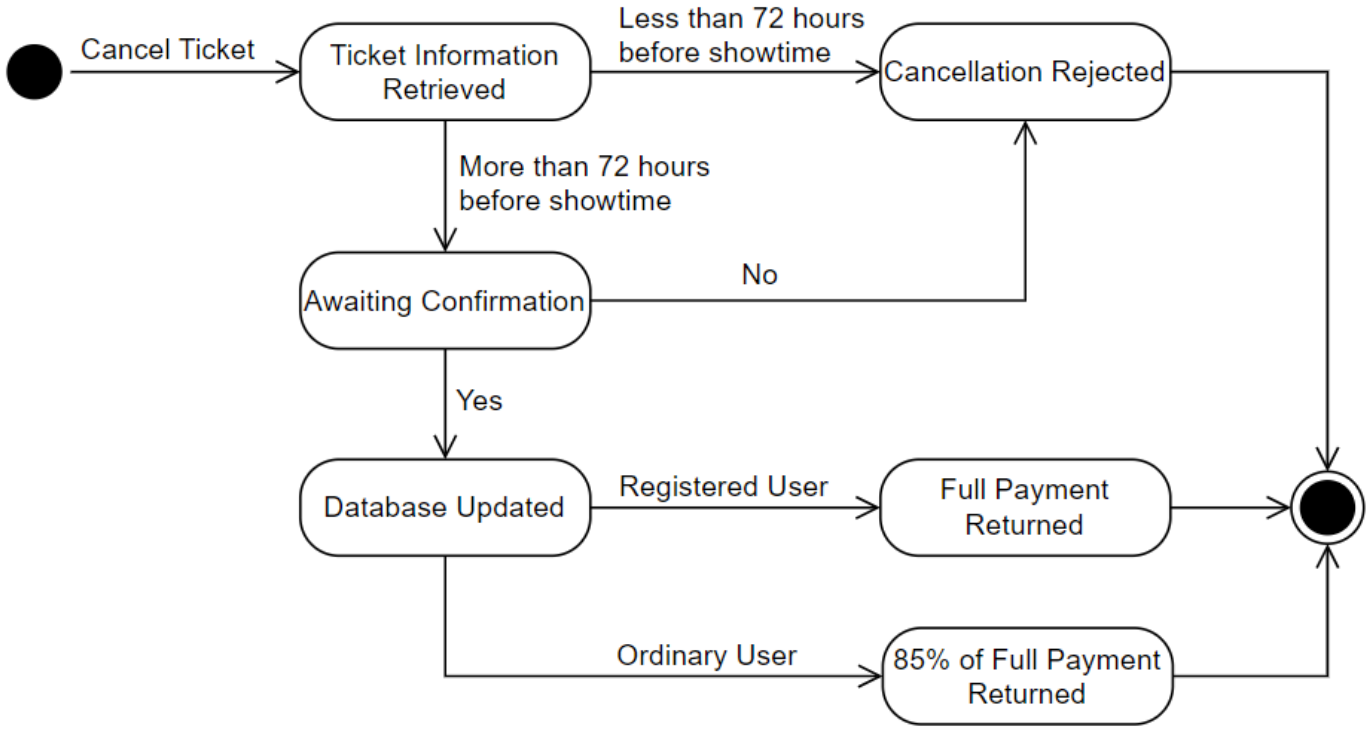
## Ticket - Justin Nguyen



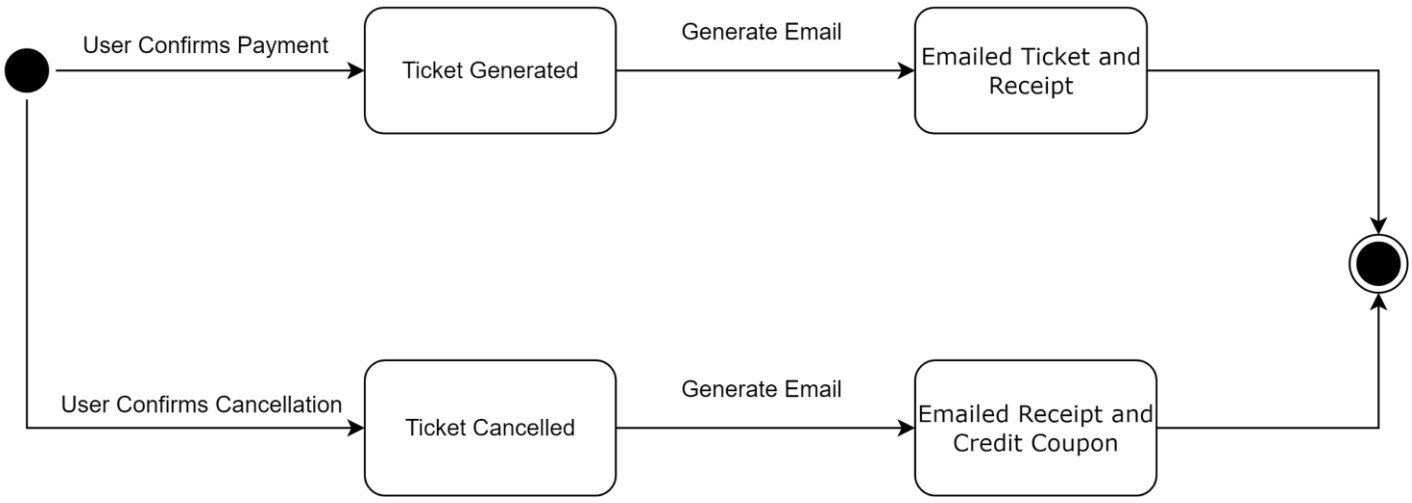
# Payment - Stewart Pratt



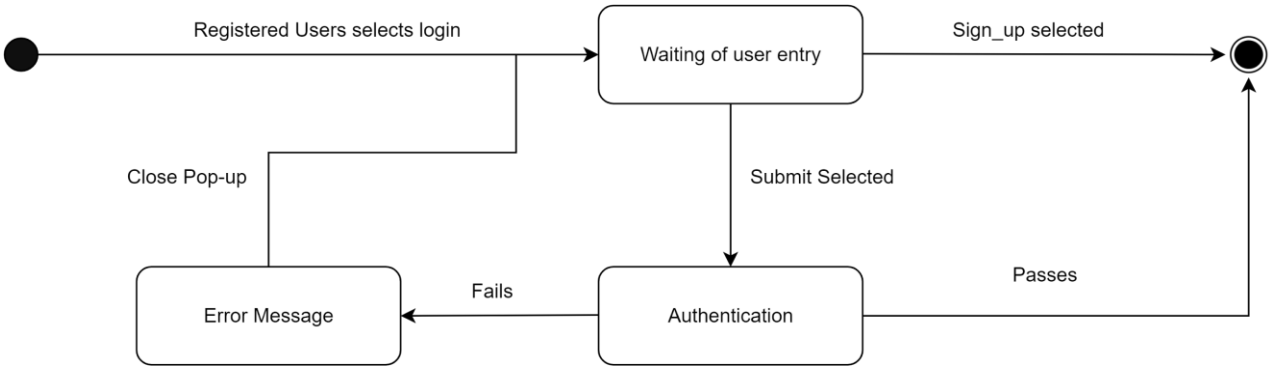
## Cancel Ticket - Fizzah Malik




## Email Receipt and Ticket - Robbie Njie

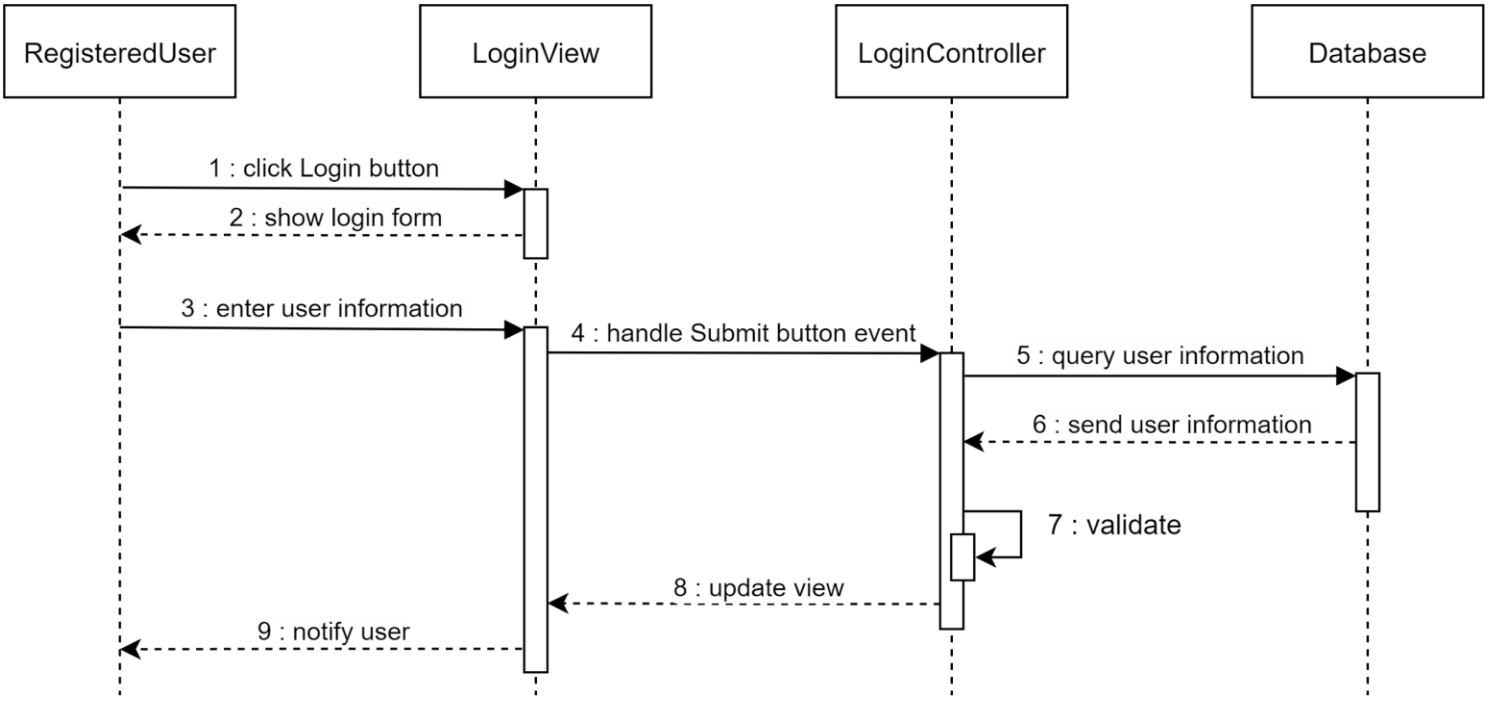


# Login - Florian Bache

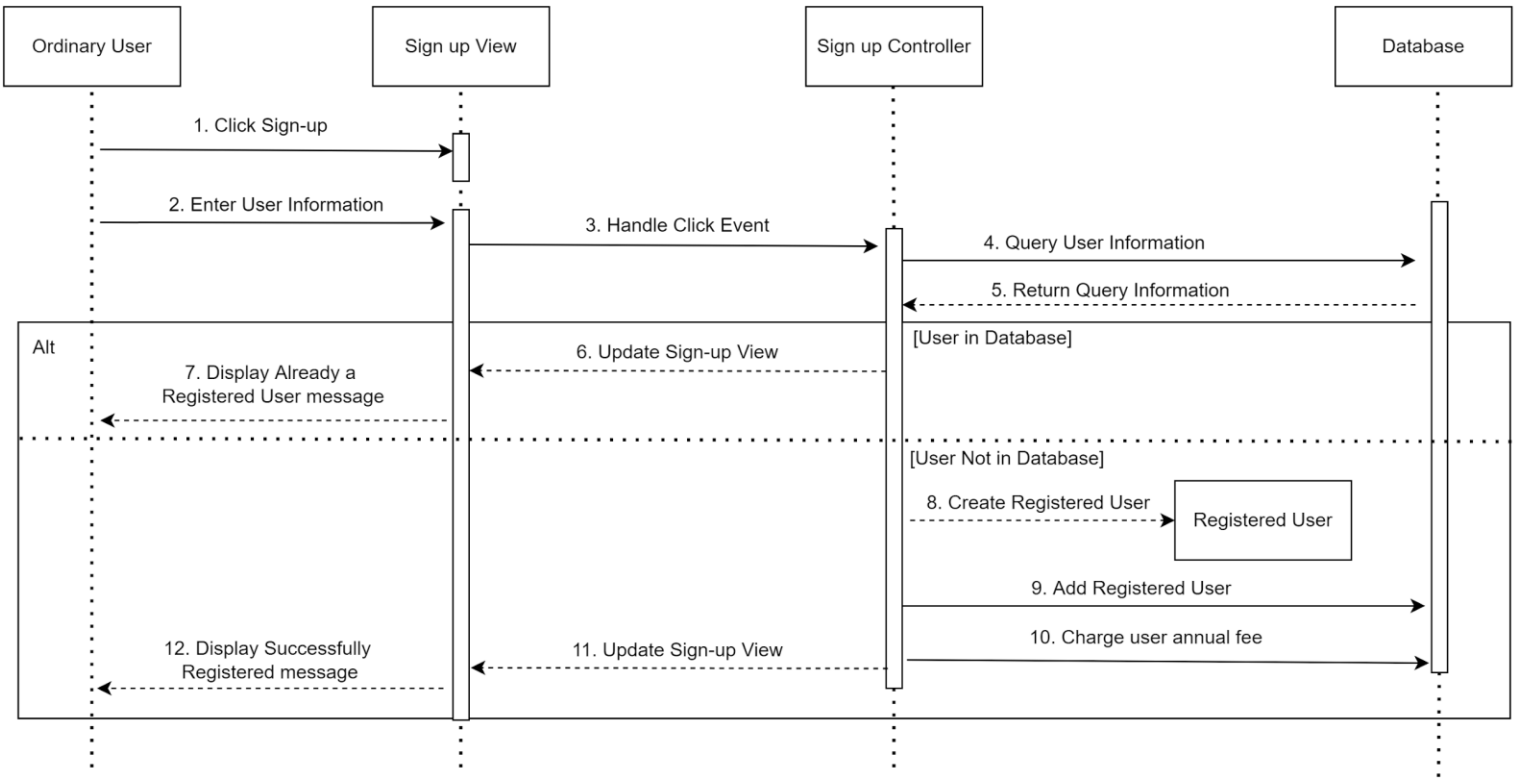


# System Interaction Diagrams

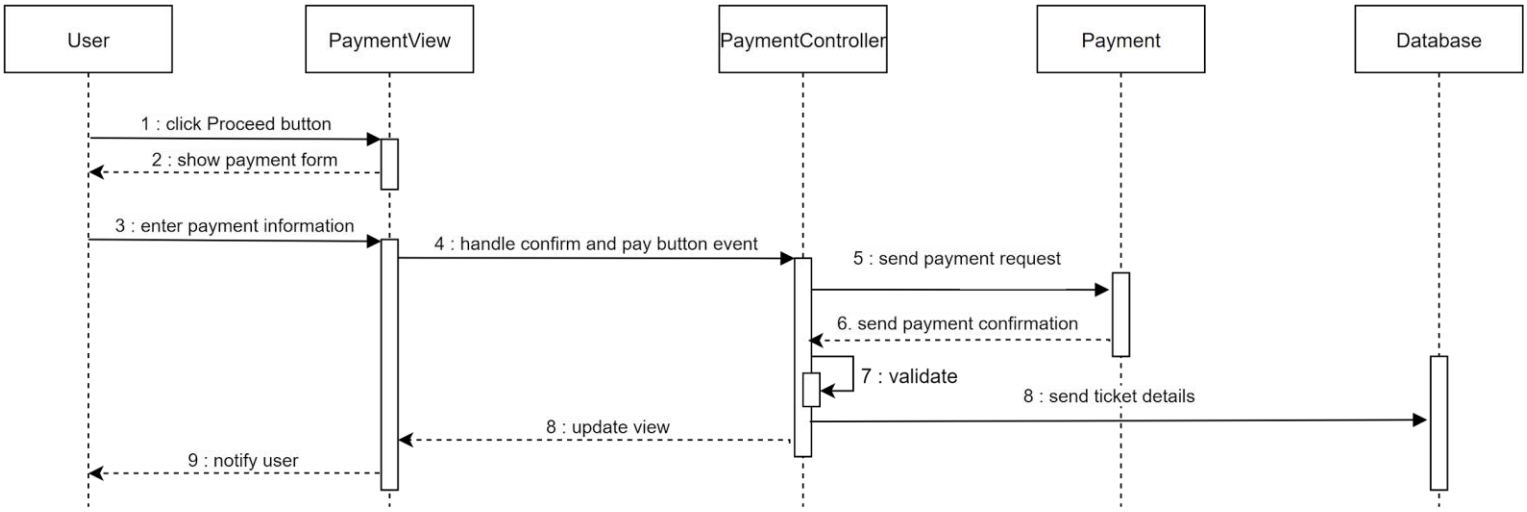
## Login - Justin Nguyen



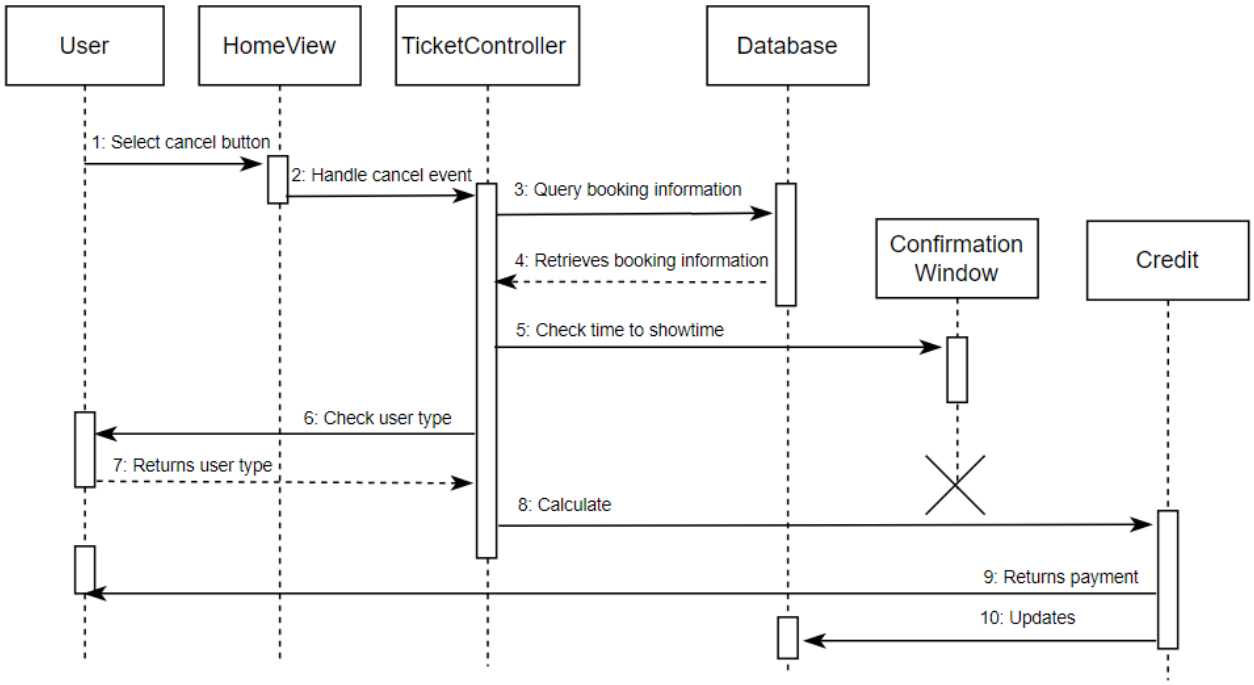
## Sign Up - Stewart Pratt



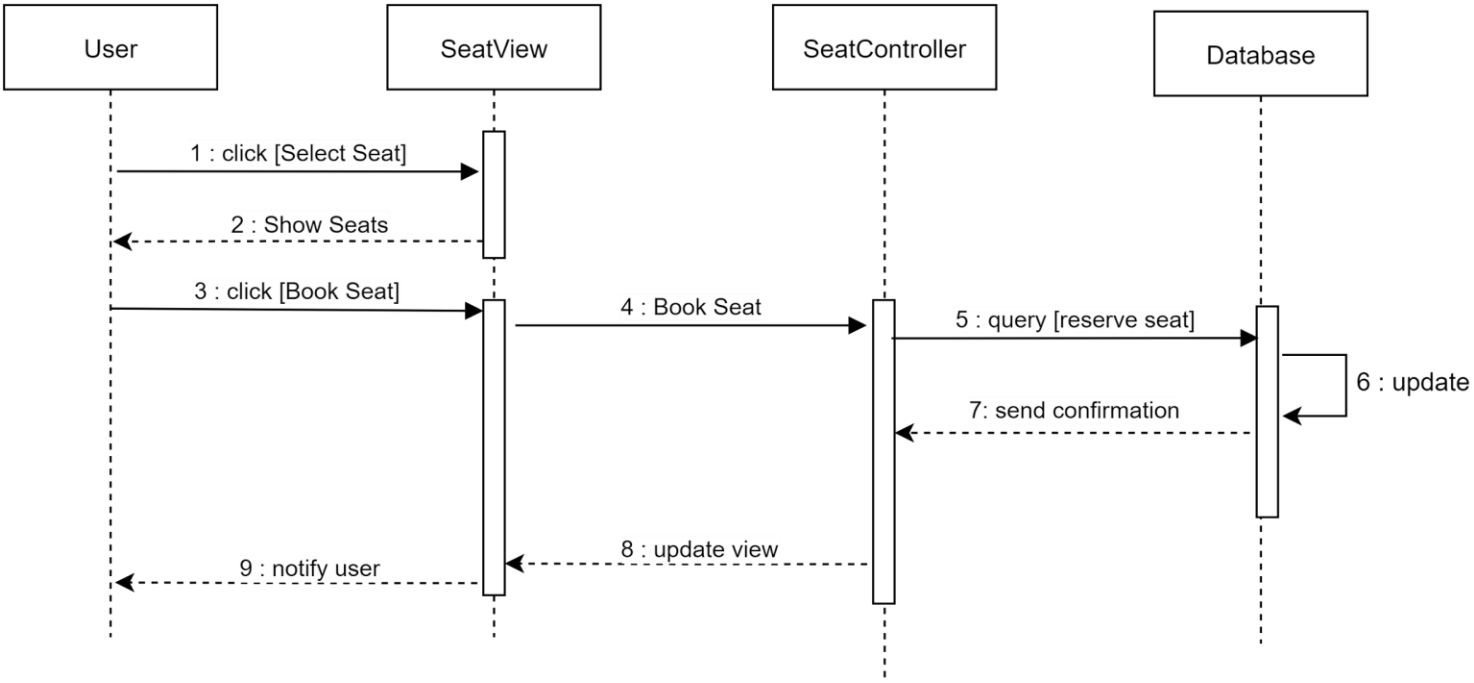

## Make Payment - Florian Bache



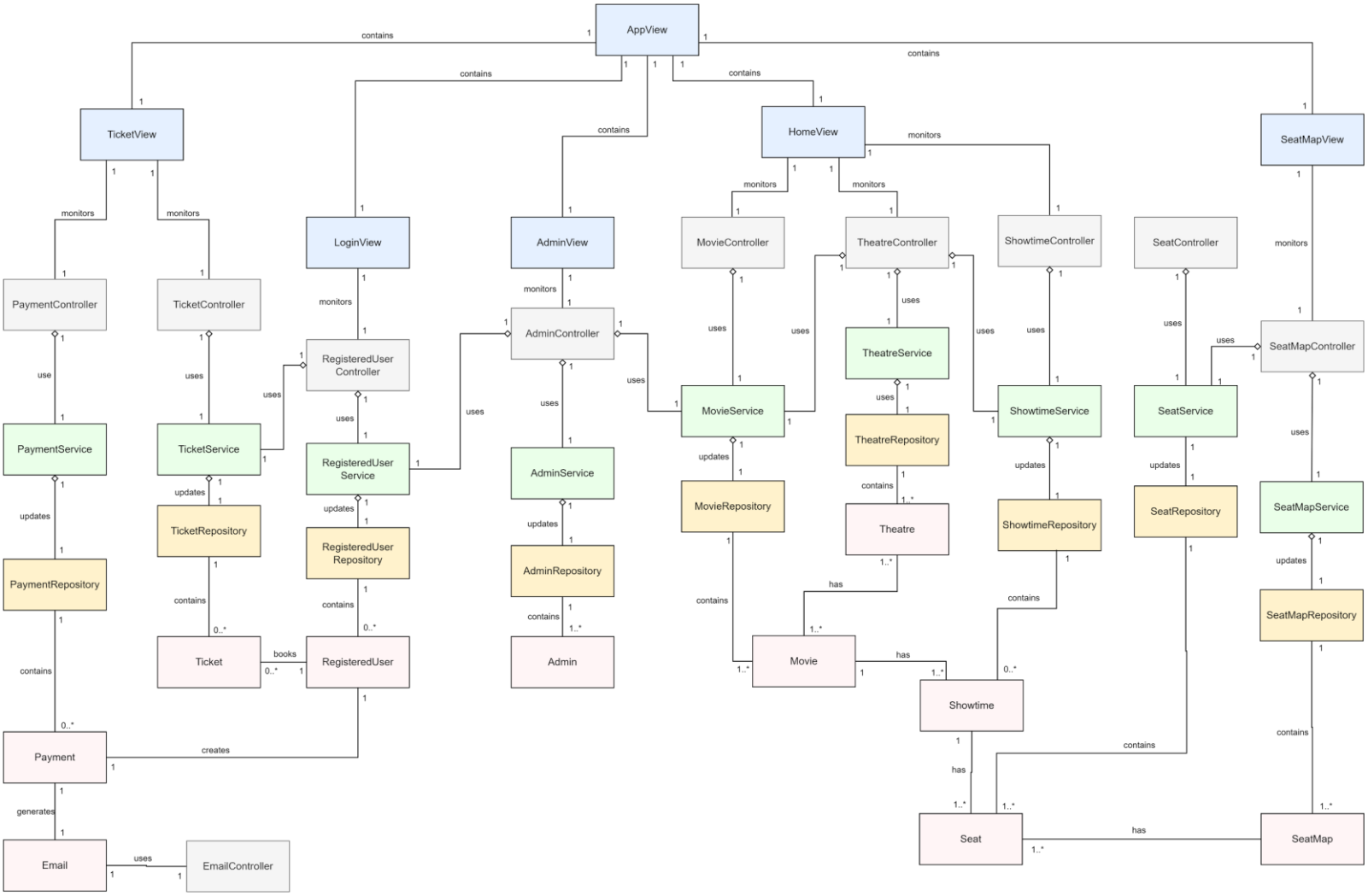
## Cancel Ticket - Fizzah Malik




## Select Seat(s) - Robbie Njie

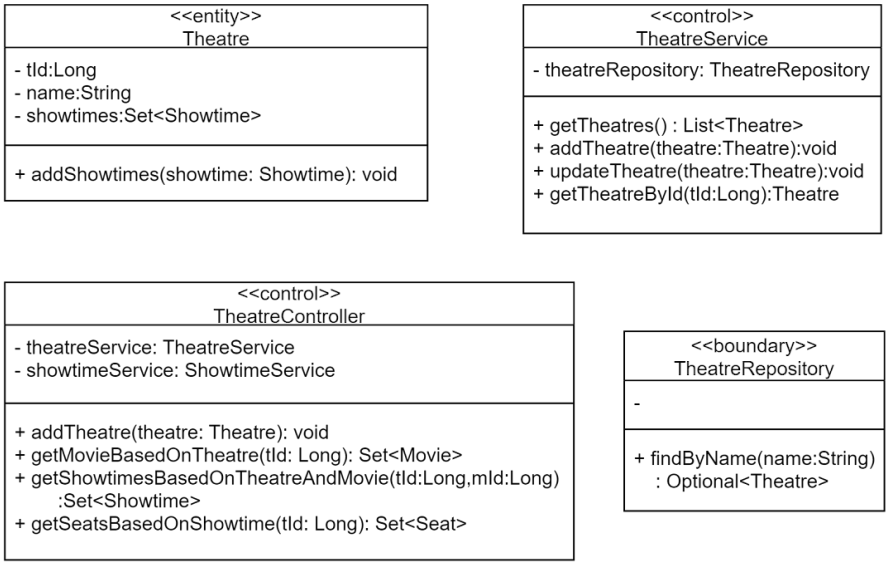


# Design Level Class Diagram

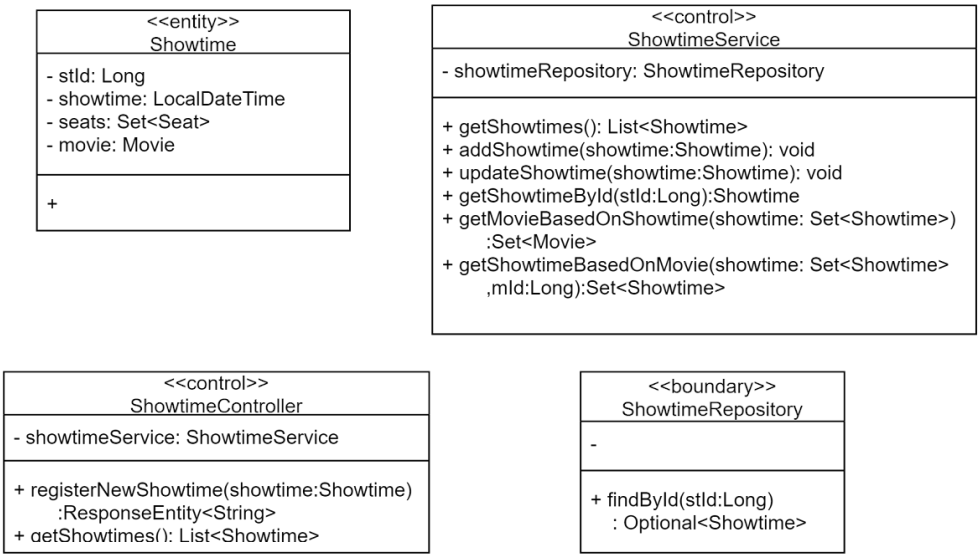


# Class Diagram

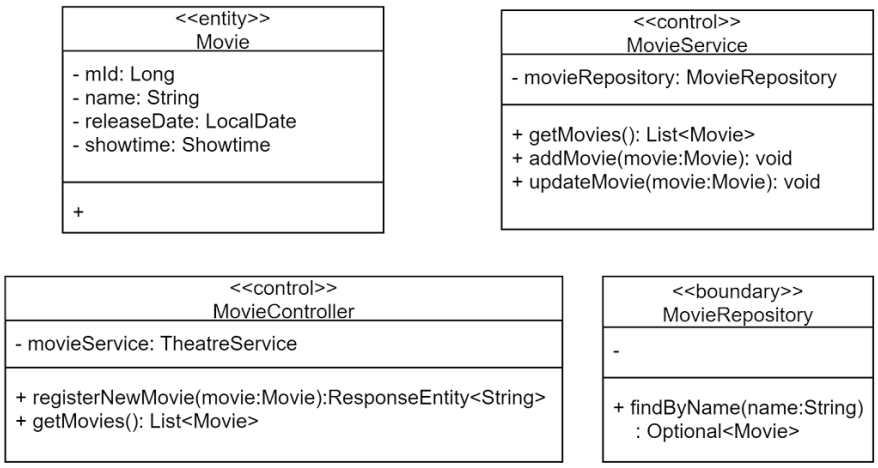
## Theatre classes



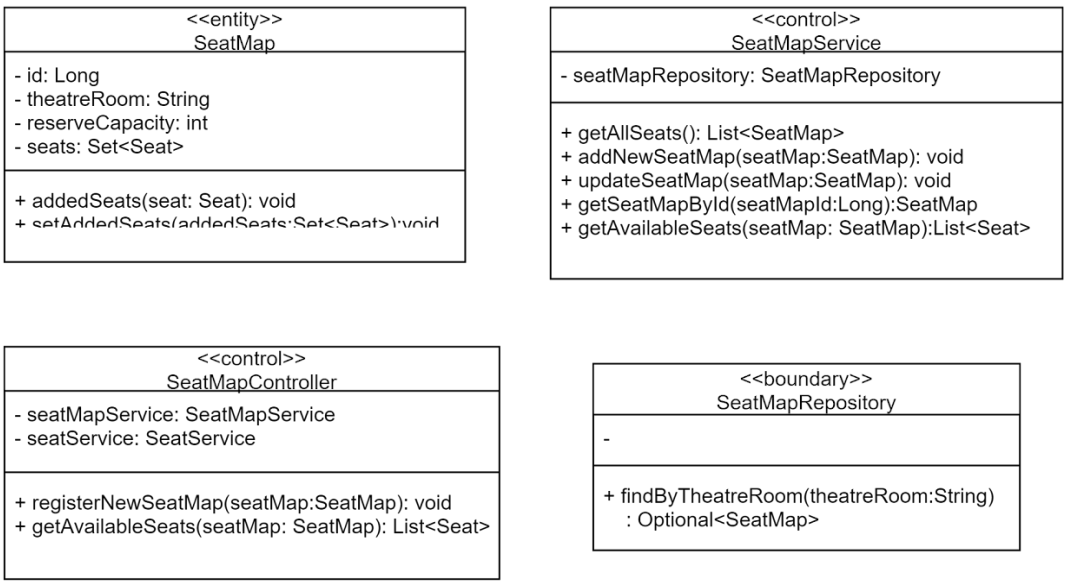

## Showtime classes



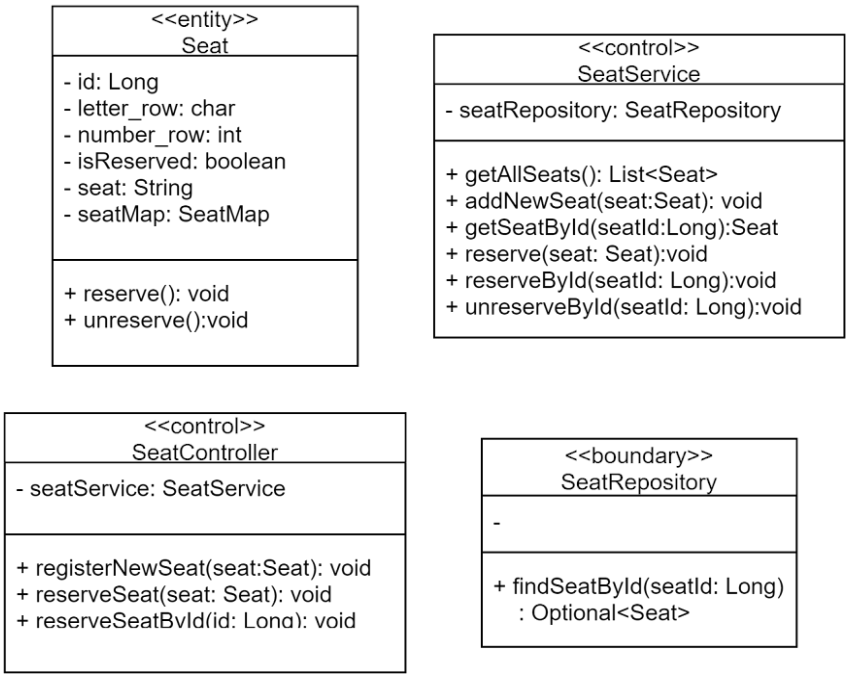
# Movie classes



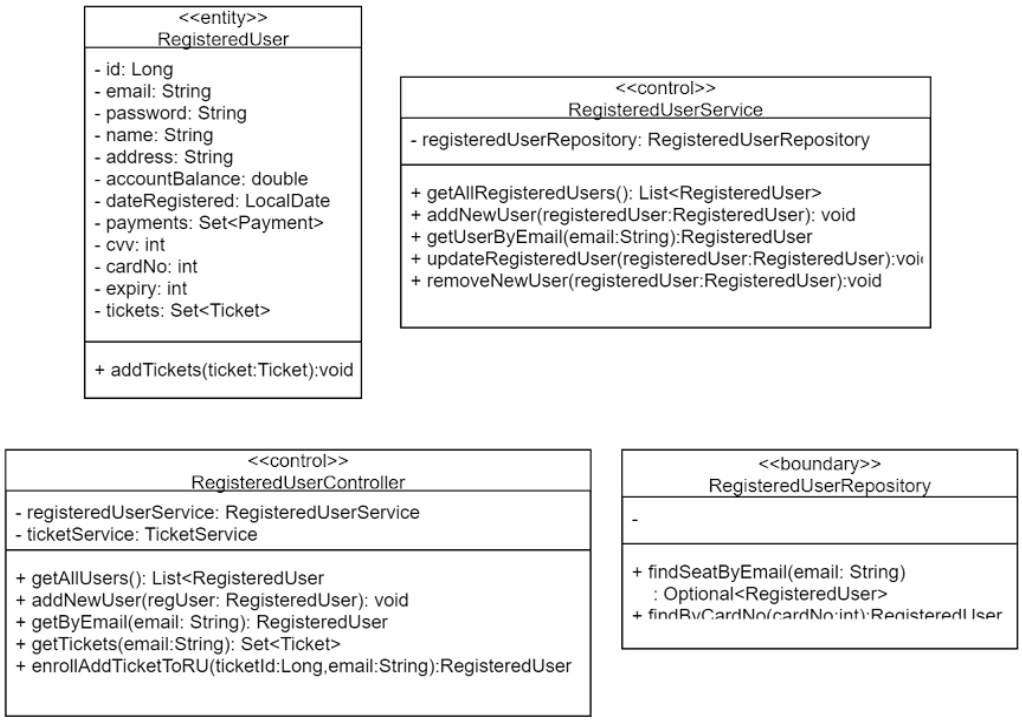

# SeatMap classes



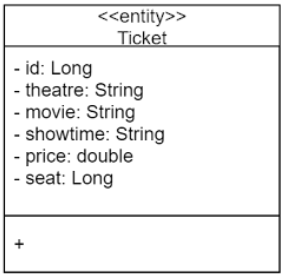
## Seat classes

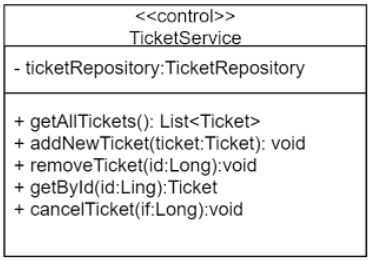


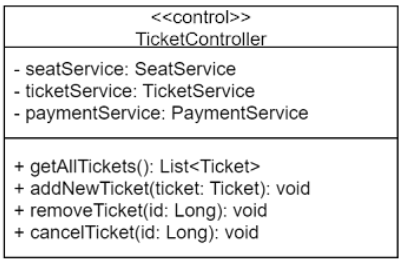

## User classes

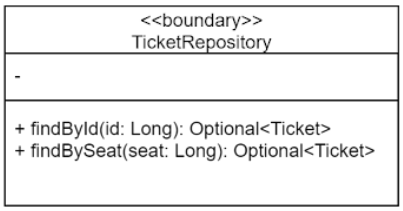


# Ticket classes

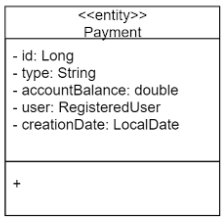


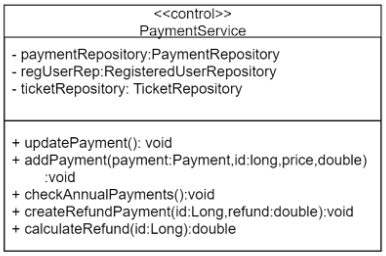



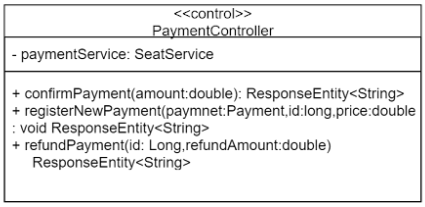



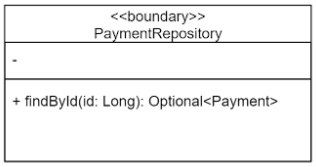




# Payment classes

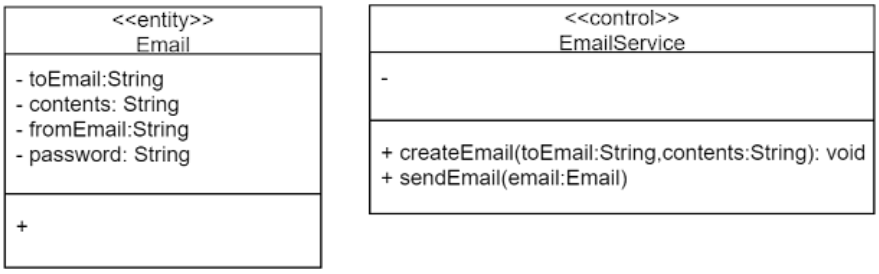


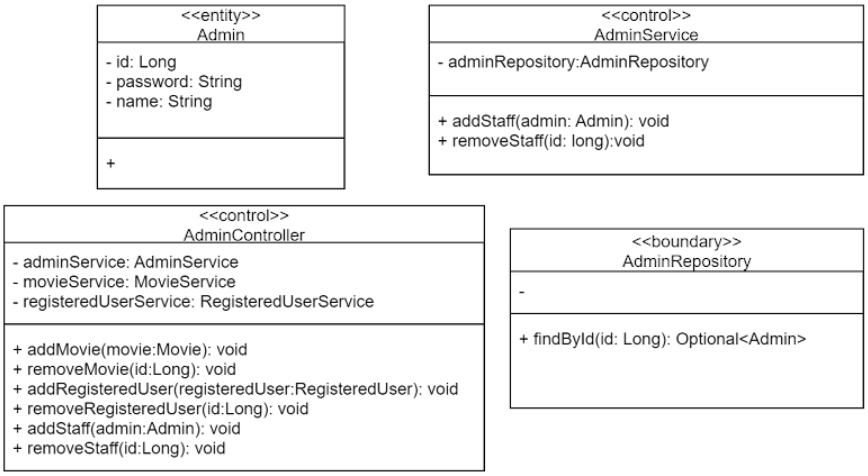



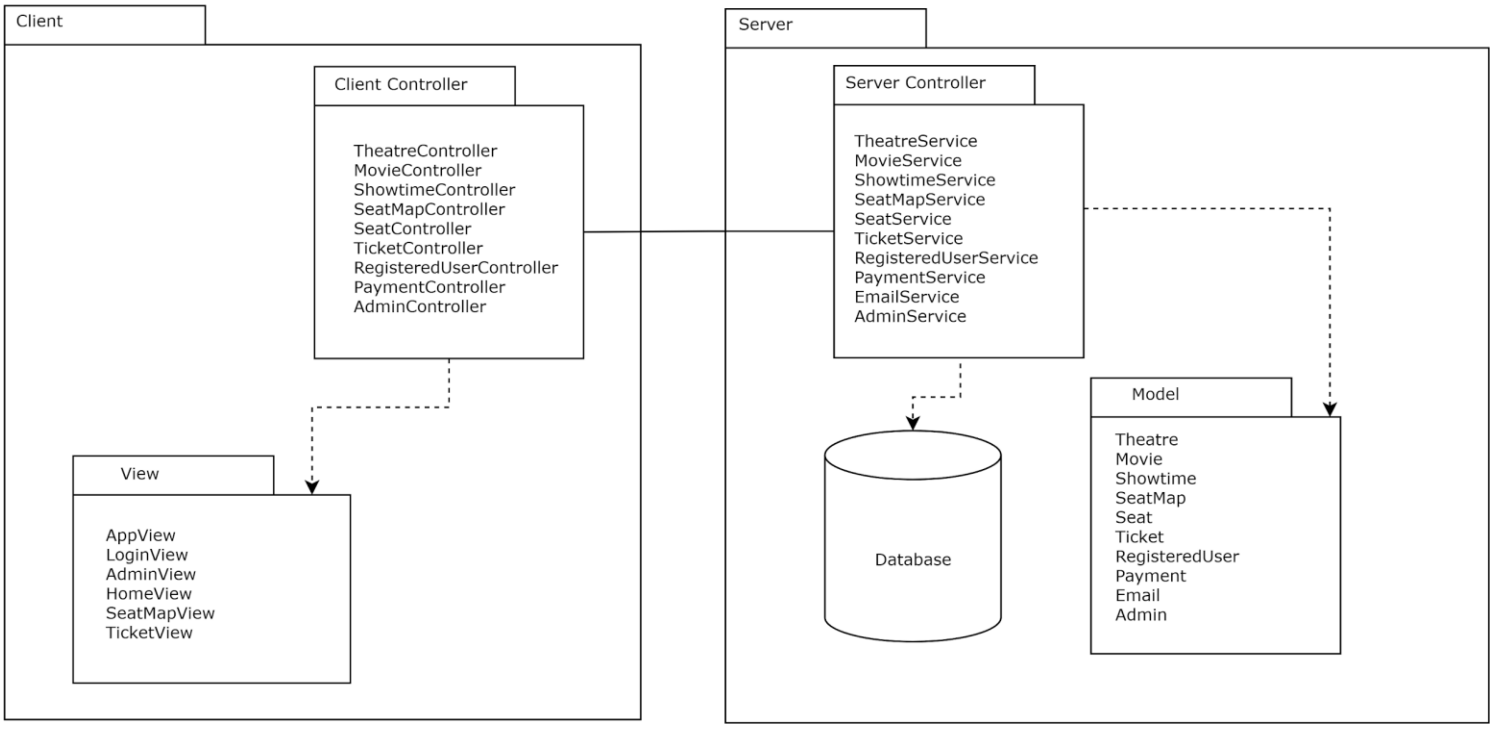
## Email classes




## Admin classes



# Package Diagram



# Deployment Diagram