**Ticket Kings Movie Theater Full Stack Development Project**

ENSF 614

Project Documentation

December 4, 2022

Group 3

Trevor Le – 30028725

Geer Ma – 10171071

Runze (Bill) Yu – 30045030

Nicolas Hirschfeld – 30172066

# Table of Contents

[Table of Contents i](#_Toc121062208)

[Table of Figures ii](#_Toc121062209)

[1. Use Case Diagram 1](#_Toc121062210)

[2. Use Case Scenarios 2](#_Toc121062211)

[Scenario: use case "login registered-user" 2](#_Toc121062212)

[Scenario: use case "register" 2](#_Toc121062213)

[Scenario: use case "select theater" 2](#_Toc121062214)

[Scenario: use case "search movie" 2](#_Toc121062215)

[Scenario: use case “view all movies” 2](#_Toc121062216)

[Scenario: use case "select movie" 2](#_Toc121062217)

[Scenario: use case "select showtimes" 3](#_Toc121062218)

[Scenario: use case "select seat" 3](#_Toc121062219)

[Scenario: use case "make payment" 3](#_Toc121062220)

[Scenario: use case "receive ticket/receipt " 3](#_Toc121062221)

[Scenario: use case "cancel ticket" 3](#_Toc121062222)

[3. List of Candidate Objects 4](#_Toc121062223)

[4. Class Diagram v.1 6](#_Toc121062224)

[5. Class Diagram v.2 7](#_Toc121062225)

[6. Sequence Diagrams 10](#_Toc121062226)

[Login Use Case: 10](#_Toc121062227)

[Select Seat Use Case: 10](#_Toc121062228)

[Select Movie Use Case: 11](#_Toc121062229)

[Make Payment Use Case: 11](#_Toc121062230)

[7. State Transition Diagrams 12](#_Toc121062231)

[Ticket Object: 12](#_Toc121062232)

[Payment Object: 12](#_Toc121062233)

[Select Movie Use Case: 13](#_Toc121062234)

[Login Use Case: 13](#_Toc121062235)

[8. System Activity Diagrams 14](#_Toc121062236)

[9. System Package Diagram 15](#_Toc121062237)

[10. System Deployment Diagram 16](#_Toc121062238)

# Table of Figures

[Figure 1: Use-case diagram for ticket king’s movie theater app. 1](#_Toc121062239)

[Figure 2: Class diagram representing the relationship between classes 6](#_Toc121062240)

[Figure 3: Detailed controller class diagram without relationships 7](#_Toc121062241)

[Figure 4: Detailed entity class diagram without relationships 8](#_Toc121062242)

[Figure 5: Class diagram for the frontend web development without relationships 9](#_Toc121062243)

[Figure 6: Login use case sequence diagram. 10](#_Toc121062244)

[Figure 7: Select seat use case sequence diagram 10](#_Toc121062245)

[Figure 8: Select movie use case sequence diagram 11](#_Toc121062246)

[Figure 9: Make payment use case sequence diagram 11](#_Toc121062247)

[Figure 10: State transition diagram for ticket object 12](#_Toc121062248)

[Figure 11: State transition diagram for payment object 12](#_Toc121062249)

[Figure 12: State transition diagram for selecting movie use case 13](#_Toc121062250)

[Figure 13: Login use case state transition diagram 13](#_Toc121062251)

[Figure 14: System activity diagram 14](#_Toc121062252)

[Figure 15: System package diagram 15](#_Toc121062253)

[Figure 16: System deployment diagram 16](#_Toc121062254)

# Use Case Diagram

Diagram

Description automatically generated

Figure : Use-case diagram for ticket king’s movie theater app.

# Use Case Scenarios

### Scenario: use case "login registered-user"

This scenario starts on the homepage. The user will clickon the login-button. They will be taken to a login-page where they inputtheir registered email and password. If entered correctly, the user will be directed to the homepage as a registered user. If the user is not a registered-user, the user will receive a pop-up error message. They will have to click okay to acknowledge the error, and try to enter their information again, or navigate to the registration page or back to the homepage to continue as a guest user.

### Scenario: use case "register"

This scenario starts after the user has navigated to the homepage and wants to become a registered user with Ticket Kings. The user will select on the registration-button in the header.  The user will then be taken to a page where they are required to submit in the following information: firstName, emailAddress, homeAddress, password, passwordVerification, creditCardNumber, creditCardExpiry, and CVV.  Once the user has entered the information, they will click on the register-button. This creates-an-account for the user, and a confirmation is sent to their email. The newly registered-user will then be directed to the homepage as a logged in registered-user and can search-for-a-movie, view-all-movies, cancel-ticket, view-ticket or log-out.

### Scenario: use case "select theater"

This scenario starts after the user has logged-in as a registered-user or continues as a guest-user. The user can then view all the available theaters and select the desired one. Once the user has selected a theater, they will be taken to the homepage for that specific theater where they can then search for a movie, view all movies, cancel a ticket.

### Scenario: use case "search movie"

This scenario starts after the user has logged-in as a registered-user or continues as a guest-user and has selected a theater. The user can input the movie-name or any letters belonging in the movie-name and search for it. If the movie is currently playing, the user can then select it, and they will be directed to the individual-movie-page where they can view and select showtime and seat.

### Scenario: use case “view all movies”

This scenario starts after the user has logged-in as a registered-user or continues as a guest-user and has selected a theater. The user can select the view-all-movies-button on the homepage to be taken to the movie-page where all the available movies will be listed and displayed. The user can then select a desired movie, which will direct the user to the individual-movie-page, where they can select a showtime and a seat.

### Scenario: use case "select movie"

This scenario starts when the user is on the movie page and all available movies are being displayed. The user will then be able to selectthe movie they want. A detailed, individual movie-page will be displayed, and the user can then view and select a showtime and a seat. Once the showtime and seat are selected, the user can then continue to purchase a ticket by proceeding to the payment-page.

### Scenario: use case "select showtimes"

This scenario starts once the user has selected a movie. The user will then be able to viewall the showtimes for the selected movie and selectone of the showtimes. A graphical representation of the available seats will then be displayed,and the user can select a seat.

### Scenario: use case "select seat"

This scenario starts when the user has selected a movie and a corresponding showtime. The user will be able to view all the available seats through a graphical representation, and then can select their desired seat. If a seat is not available, the seat is displayed red. Once the user has selected a seat, then the user can input their credit coupon code and proceed to pay for a ticket. Credit coupon code is not required, only option if the user has one available.

### Scenario: use case "make payment"

This scenario starts when the user has selected a showtime and seat for the desired movie. The user will see the price of the ticket, as well as other confirming details. The guest-user will be prompted to enter their payment-information such as name, email, Credit Card Number, CVV, and Expiry Date. If the user is a registered-user, they are not required to enter anything, and the ticket will be confirmed immediately on loading the purchase-page. Once payment-information is entered, the user will receivea ticket and receipt through email. The money for their ticket-purchase will be sent from the bank through their credit card transaction.

### Scenario: use case "receive ticket/receipt "

This scenario starts when the user has made a payment for a movie-ticket. The user will receivethe ticket as well as the receipt which contains a theater-number, seat-number, movie-time, movie-duration, movie-name, and ticket-price by email. The user will then be directed to the homepage where the user can perform any of the actions on the homepage that has previously been stated in the use-cases above. Specifically, if desired, now that the user has a ticket, they can cancel their ticket by inputting their ticket-code.

### Scenario: use case "cancel ticket"

This scenario starts after the user has made a payment, receivesthe ticket by email, but wants to cancel the ticket. The user will navigate to the Ticket Kings homepage. On the homepage they will be able to enterthe cancel ticket-code.  If the movie has not yet started, the cancellation request will proceed, and the user will have the money refundedin the form of a Ticket Kings credit, which will be emailed to the user. The user will not receive any refund for a movie that has already started. If the guest user cancels their ticket 72 hours prior to the movie, they will receive a credit-coupon-code with 15% administration fee for future purchase up maximum of one-year expiration date. Registered users do not have to pay 15% admin fee for cancelling their tickets. All coupons and cancelation-confirmations are sent via email.

Note: Admin features that are not included in the use case scenarios.

# List of Candidate Objects

|  |  |
| --- | --- |
| **Noun** | **Decision** |
| password | filtered(Attribute of registered User) |
| homepage | candidate object |
| user | filtered(Generalization of RU and GU) |
| registered-user | candidate object |
| guest-user | candidate object |
| theatre | candidate object |
| movie-name | filtered(Attribute of movie) |
| movie | candidate object |
| showtime | candidate object |
| seat | candidate object |
| ticket | candidate object |
| cost | filtered(Attribute of ticket) |
| payment | candidate object |
| payment-information | filtered(Attribute of payment) |
| credit card number | filtered(Attribute of payment) |
| CVV | filtered(Attribute of payment) |
| Expiry Date | filtered(Attribute of payment) |
| receipt | filtered(Same as payment) |
| theatre number | filtered(Attribute of ticket) |
| seat number | filtered(Attribute of ticket) |
| email | filtered(Attribute of registered user) |
| cancelation request | filtered(Method in payment) |
| money | filtered(Same as cost) |
| credit card | filtered(Attribute of payment) |
| bank | filtered(Actor) |
| login-button | filtered(button on homepage) |
| login-page | candidate object |
| registration button | filtered(button on homepage) |
| registration page | candidate object |
| header | candidate object |
| firstName | filtered(Attribute of user) |
| emailAddress | filtered(Attribute of user) |
| homeAddress | filtered(Attribute of registered user) |
| passwordVerification | filtered(Attribute of registered user) |
| creditCardExpiry | filtered(Attribute of user) |
| individual-movie-page | candidate object |
| movie-page | candidate object |
| payment-page | candidate object |
| credit coupon code | filtered(Attribute of credit) |
| theatre-number | filtered(Attribute of theatre) |
| movie-time | filtered(Attribute of movie) |
| movie-duration | filtered(Attribute of movie) |
| ticket-price | filtered(Attribute of ticket) |
| ticket-code | filtered(Attribute of ticket) |
| credit | candidate object |
| credit expiration date | filtered(Attribute of credit) |

# Class Diagram v.1

Diagram

Description automatically generated

Figure : Class diagram representing the relationship between classes

# Class Diagram v.2

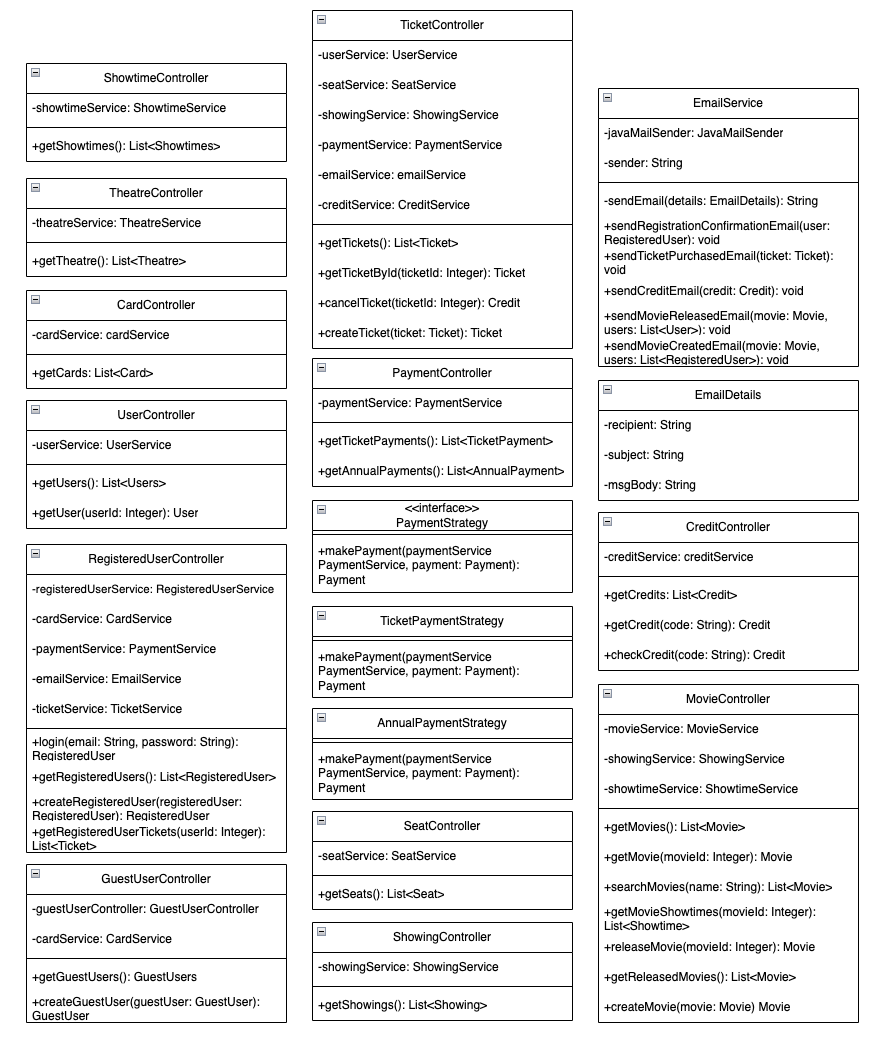


Figure : Detailed controller class diagram without relationships

Graphical user interface

Description automatically generated with low confidence

Figure : Detailed entity class diagram without relationships

Graphical user interface

Description automatically generated

Figure : Class diagram for the frontend web development without relationships

# Sequence Diagrams

### Login Use Case:

Diagram

Description automatically generated

Figure : Login use case sequence diagram.

### Select Seat Use Case:

Diagram

Description automatically generated

Figure : Select seat use case sequence diagram

### Select Movie Use Case:

Diagram

Description automatically generated

Figure : Select movie use case sequence diagram

### Make Payment Use Case:

Diagram

Description automatically generated

Figure : Make payment use case sequence diagram

# State Transition Diagrams

### Ticket Object:

Diagram

Description automatically generated

Figure : State transition diagram for ticket object

### Payment Object:

Diagram

Description automatically generated

Figure : State transition diagram for payment object

### Select Movie Use Case:

Diagram

Description automatically generated

Figure : State transition diagram for selecting movie use case

### Login Use Case:

Diagram

Description automatically generated

Figure : Login use case state transition diagram

# System Activity Diagrams

Diagram

Description automatically generated

Figure : System activity diagram

# System Package Diagram

Diagram

Description automatically generated

Figure : System package diagram

# System Deployment Diagram

Diagram

Description automatically generated

Figure : System deployment diagram