Design Document

for

Movie Ticket Reservation App

Prepared by Thien Nguyen, Sonia Obi, Remy Onyegbutulem, Kenechukwu Nwabueze

2020-11-21

1 Introduction

1.1 Purpose

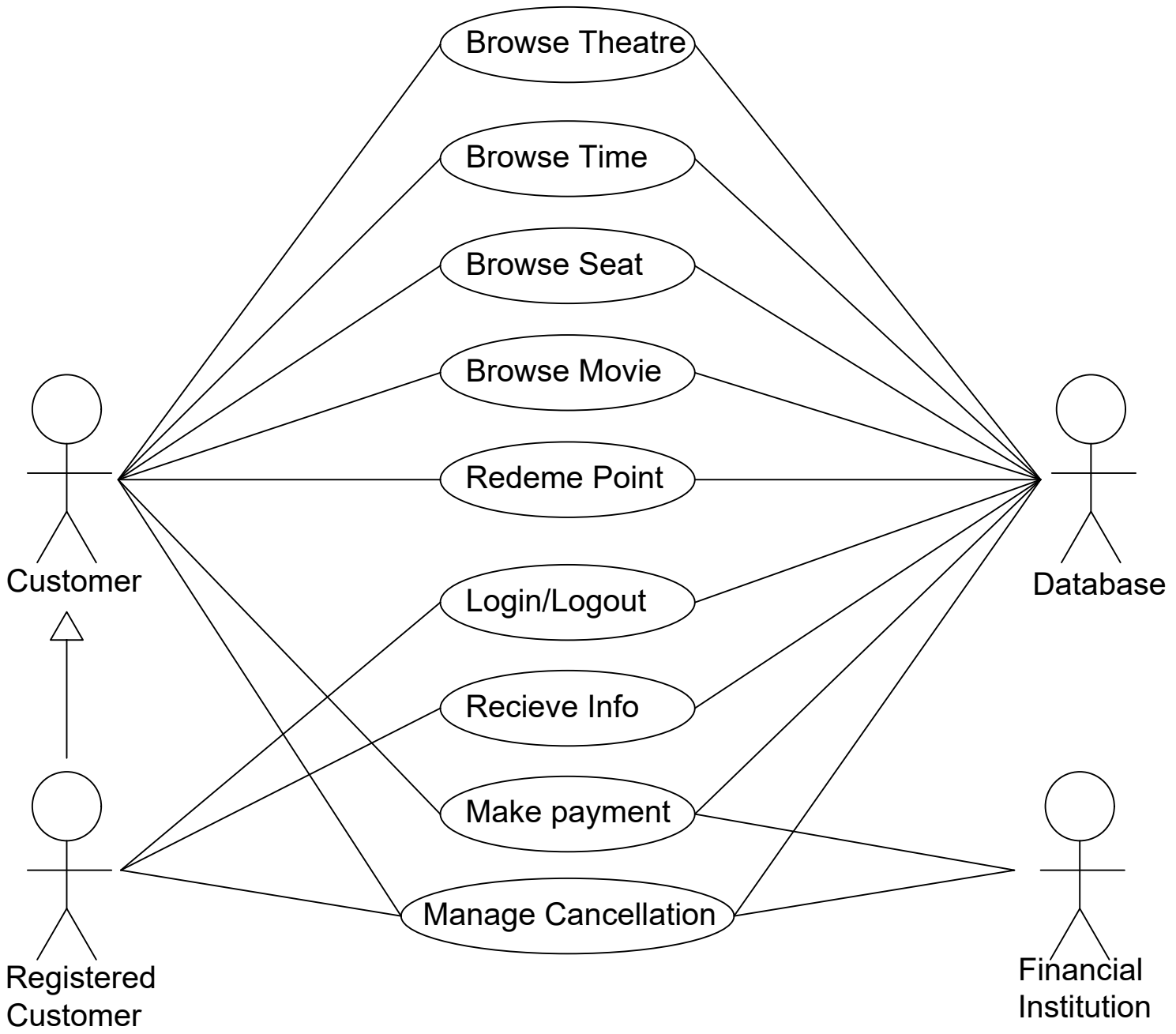
The purpose of this project is to develop a Movie Theatre Ticket Reservation application that allows user to reservation a movie ticket in a theatre. The user can also cancel their movie ticket and depending on the type of user will receive a voucher or full refund.

1.2 Scope

The objective of this app is to provide an easy guided experience for user. Users will be guided through all process of reserving a seat in a movie theatre. The app's front-end will handle interaction with the user and relay information to the back-end. The back-end will be responsible for business operations. The app also manages its own database automatically every time an operation is being done. The app will not be managed by an administrator, everything will be done online.

1.3 Use Case Diagram

The use case diagram for the app is shown next page. Each use case will be discussed in the next section



USE CASE DIAGRAM - V 1.0

GENERAL NOTES:

- Assume Database is already established and no modification is needed
- Assume no System Administrator is needed

LIST OF ACTORS:

- Customer
- Registered Customer
- Database
- Financial Institute

2 Project Descriptions

2.1 App Descriptions

The app will be a product that can manage the entire movie theatre reservation process. The application will rely on secondary actors such as financial institution as well as external database server to store information on the users, receipts, movies, theatres, ticket, seats.

2.2 Actor Descriptions

There are 4 specific actors identified that involved in the use cases for this app. They are listed below:

Ordinary User:

Ordinary users are ordinary customers that uses the app. This type of user does not need to log into the system. They can search for theatre, movie. showtime and seats. They can pay for the ticket via credit card and receive the copy of ticket and receipt through email. They also can cancel their ticket up to 72 hours prior to show and receive a credit (voucher) with %15 administration fee for future purchase up to maximum of 1 year.

Registered User:

Registered users must log into their accounts to be identified as registered users. Their information such as name, address, credit card's information is saved in the app's database. They have the same abilities as the ordinary user. However, they do not need to pay the 15% admin fee for cancelling the ticket. They can also receive movie announcements and allowed to reserve the first 10% of the seats. Their annual fee is $20.00.

Financial Institution:

Financial Institutions are institutions such as banks, credit union. They are responsible for processing payments via credit card. As such, the scope of this project will not involve payment processing.

Database Server:

Databases are entities that handles the storage and query of datums. Information related to the operation of the project will be stored as tables in a database. MySQL Server will be used to store and retrieve information.

2.3 Scenarios Description

Each use cases were studied through scenarios. From the scenarios, a list of nouns was found, Through careful considerations, a list of good object candidate was selected and the operations for them was discussed. The next page outlines the scenarios and the lists discussed.

SCENARIOS:

Browse Theatre - This use case begins when the user has decided to see a movie. At this point, the database displays a list of available theatres so that the user can select a theatre.

Browse Movie - This use case begins when the user selects the theatre that they would want to visit and wants to search for what movie to watch. The system then displays a list of available movies, and the user will then have the opportunity to select the movie that they want to watch.

Browse Time - This use case begins when the user has decided that they would like to see a movie. At this point, the system will display a list of showtimes for the movie they would like to see so that the user can select a showtime.

Browse Seat - This use case begins when the user has already selected a movie and showtime and wants to pick their seat. At this point, the system will display the seating chart, showing which seats are available/taken. When they choose the seats they would like for the movie, the seat is temporarily marked as taken to the user.

Make payment - This use case begins when the user has already selected a movie, showtime, and seat and has decided to proceed to checkout. At this point the system asks the user to select their financial institute, and prompts them to enter the user and bank information. When they submit their payment, the system confirms the completion of the process. The system will then send out an email containing the ticket and the receipt information. The ticket will have information on the movie, the showtime and the seat that the user selected.

Manage Cancellation - This use case begins after the user has already purchased the ticket and received a confirmation email with the receipt. When the user wants to cancel the reservation, the system will first check the time limit (72 hours). If 72 hours have not passed, then the system needs to check if the customer is a registered customer or an ordinary customer. If they are an ordinary customer, the system will refund for the price of the movie minus 15% for administration fee. If the user is a registered customer, the system will refund the full amount.

Redeem voucher - This use case begins when the user wants to redeem their voucher. The system will first look at the database, search for the voucher to validate. The system will then deduct the price of the movie based on the voucher.

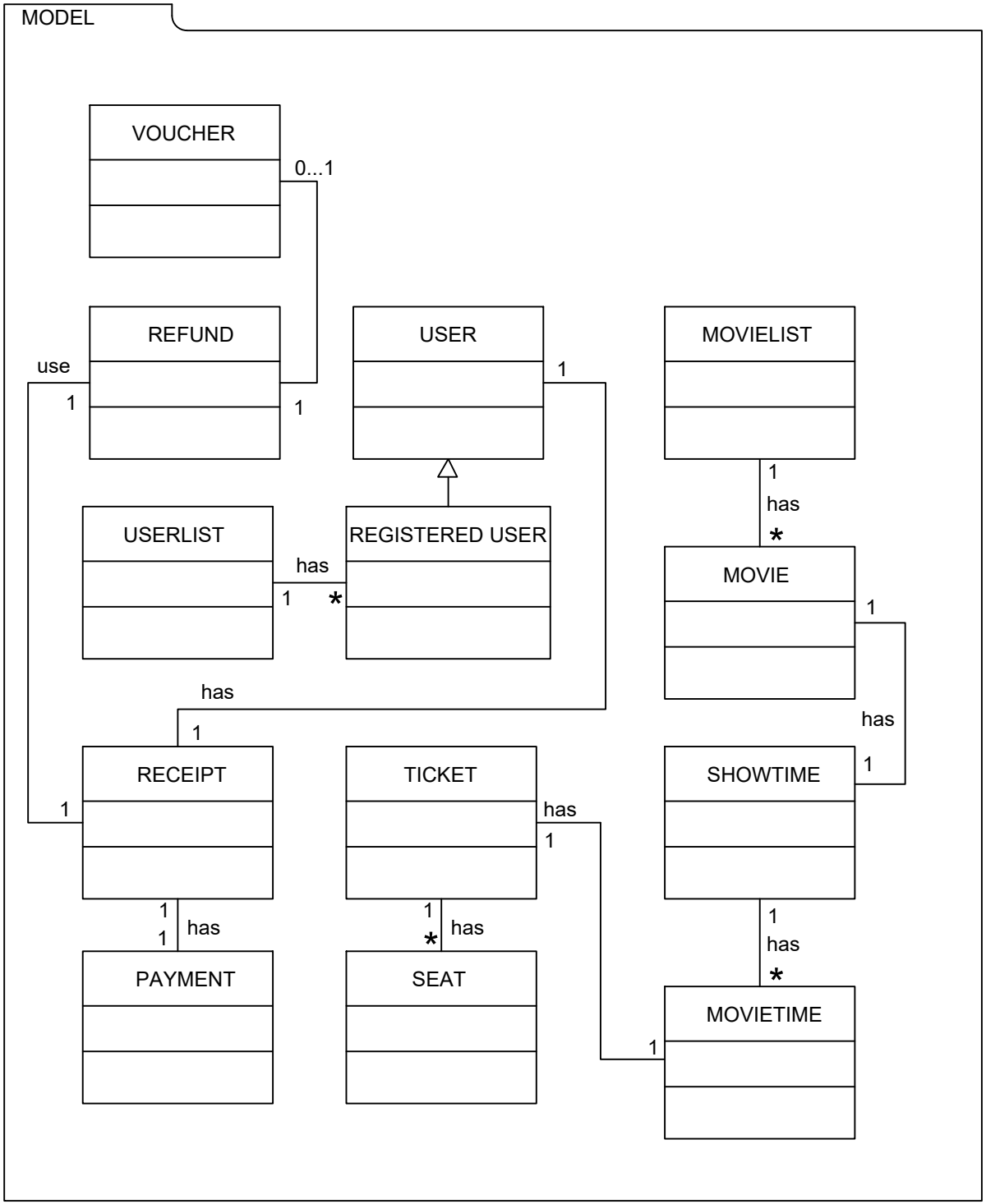
Login - This use case begins when the user wants to login into the system for registered user. The system will first collect the user information supplied, the search the database for a valid username and password. If the credential is valid, the system will grant access to the user..

LIST OF NOUNS:	LIST OF GOOD CANDIDATES OBJECTS:	LIST OF POSSIBLE OPERATIONS:
• Theatre • User info • Bank info • Movie info • Movie points • Refund • Payment • Tickets • Receipts • Seat • Showtime • Email • Customer • Password • Username • Database • Fee • Amount • Voucher • Credentials • Customer	• Theatre • User (Ordinary and Registered) • Movie • ShowTime • MovieTime • Seat • Ticket • Receipt • Payment • Refund • Voucher	• Save/Retrieve/Validate user info • Save/Retrieve bank info • Save/Retrieve/Send movie info • Add/Subtract movie points • Send refund • Receive/Send payments • Send/Validate tickets • Send receipts • Display/Reserve seats • Display/Select showtimes

3 Design Diagrams

The following table summarize the diagrams used for the design of the app:

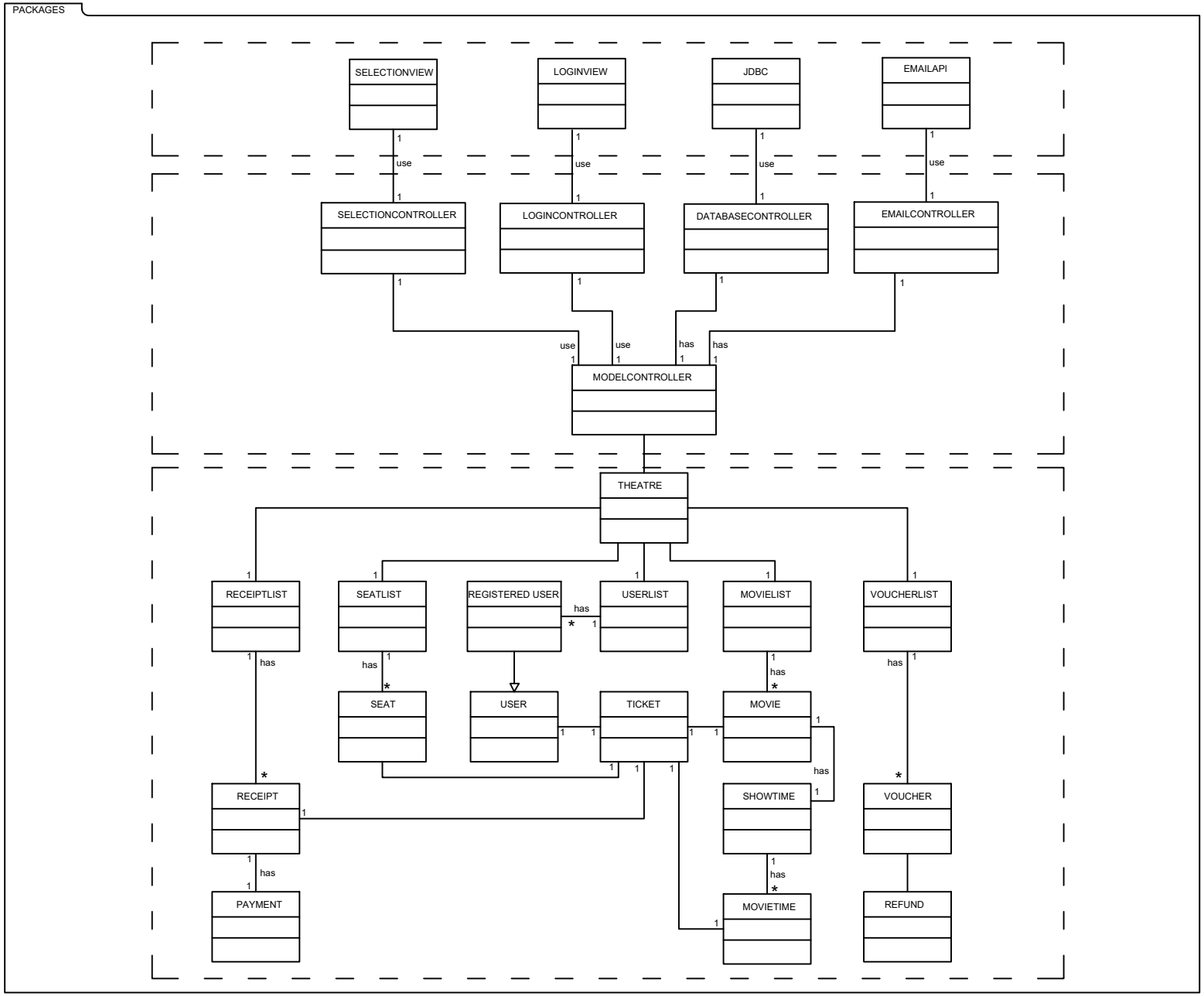
Diagram Title	Diagram Descriptions
Design Concept (Revision 0)	The very first concept diagram created
Design Concept (Revision 1)	The first revision of the original concept diagram. Additional boundary and control classes are added. The design split into 3 parts: view, control, and model.
Design Concept (Revision 2)	The latest revision of the design concept. This revision includes: futher refinement of each class, utilize strategy pattern for the refund process. Inheritance used for User class.
Class Details	This drawing shows all the classes with their details, only important operations and attributes are shown
Interaction Diagram (Thien)	Interaction diagram done by Thien
Interaction Diagram (Sonia)	Interaction diagram done by Sonia
Interaction Diagram (Kene)	Interaction diagram done by Kene
Interaction Diagram (Remy)	Interaction diagram done by Remy
State Transition Diagram	The transition diagram of the ticket object and payment object
Registered User Activity Diagram	The activity diagram for registered user
Ordinary User Activity Diagram	The activity diagram for ordinary user (note: Cancellation and Payment operations implemented differently)
Package Diagram	Diagram outlines the packages that the app will be using. Only major packages and classes are shown. For classes inside Model package, refer to the Design Concept and Class Details diagrams for more information. For this diagram, a client-server architecture is assumed
Deployment Diagram	The system deployment diagram



CONCEPTUAL DESIGN
VERSION 0.0

GENERAL NOTES:

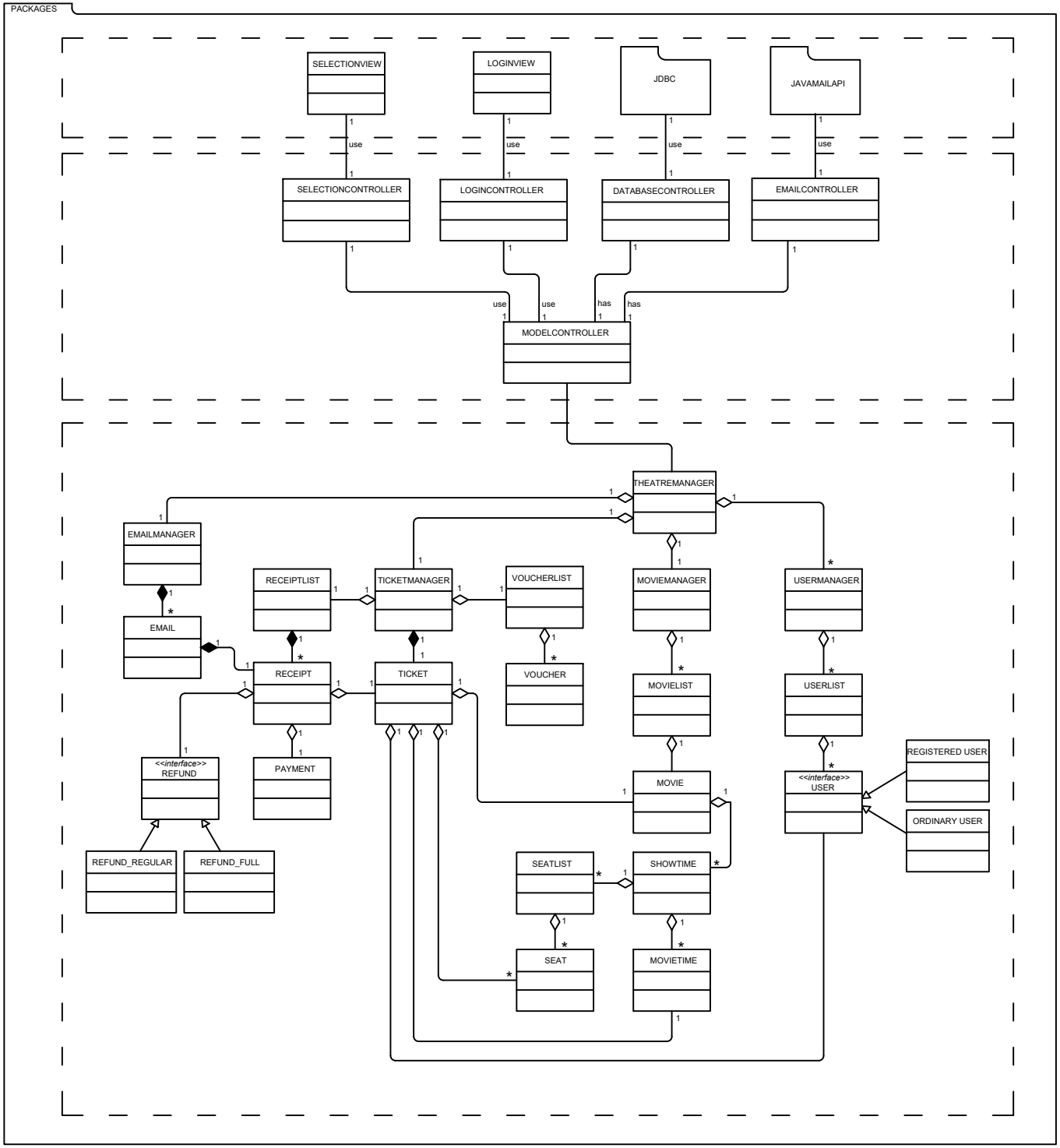
- Assume Database is already established and no modification is needed
- Assume no System Administrator is needed



CONCEPTUAL DESIGN
VERSION 1.0

GENERAL NOTES:

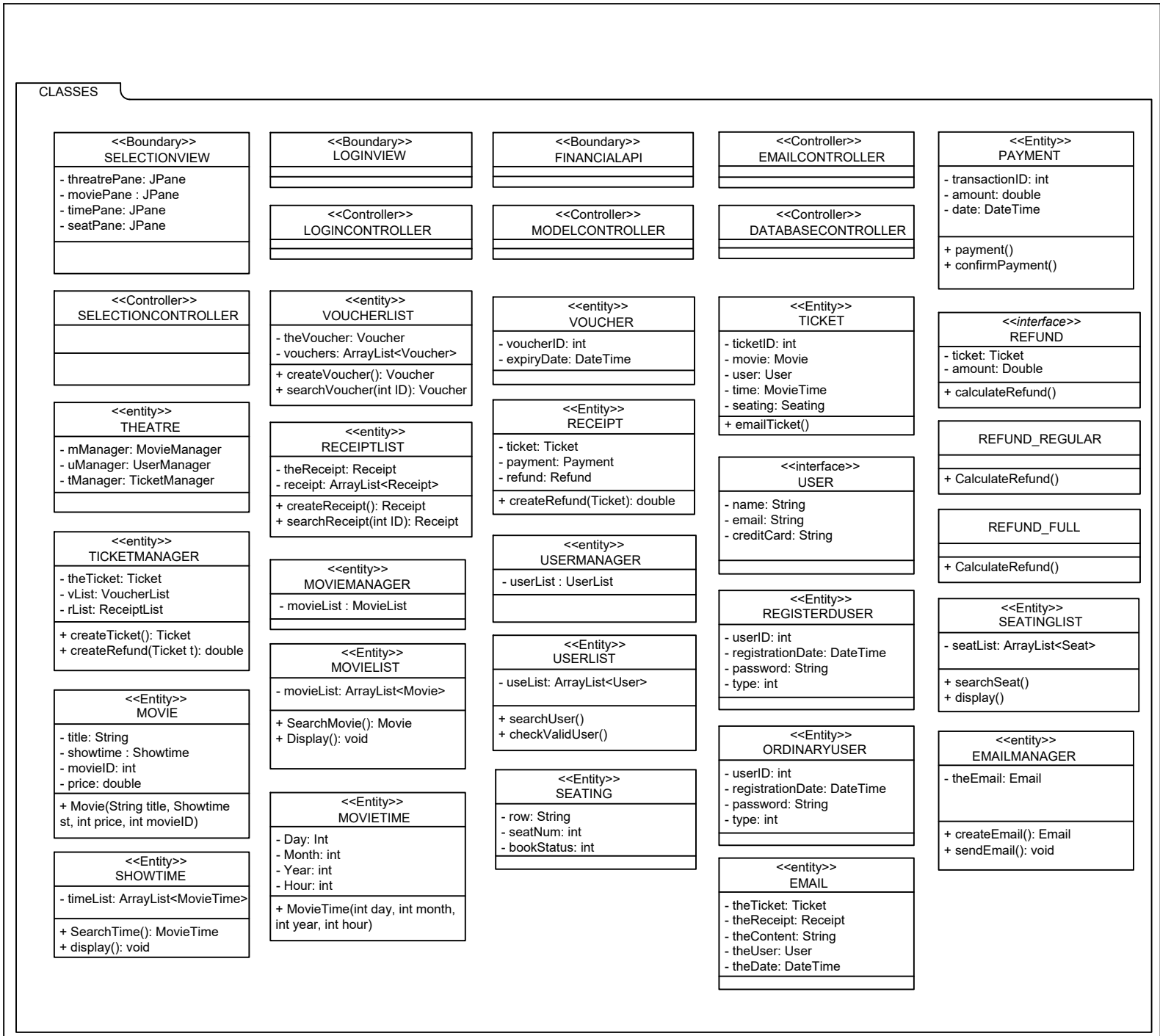
- Assume Database is already established and no modification is needed
- Assume no System Administrator is needed



CONCEPTUAL DESIGN VERSION 2.0

GENERAL NOTES:

- Assume Database is already established and no modification is needed
- Assume no System Administrator is needed



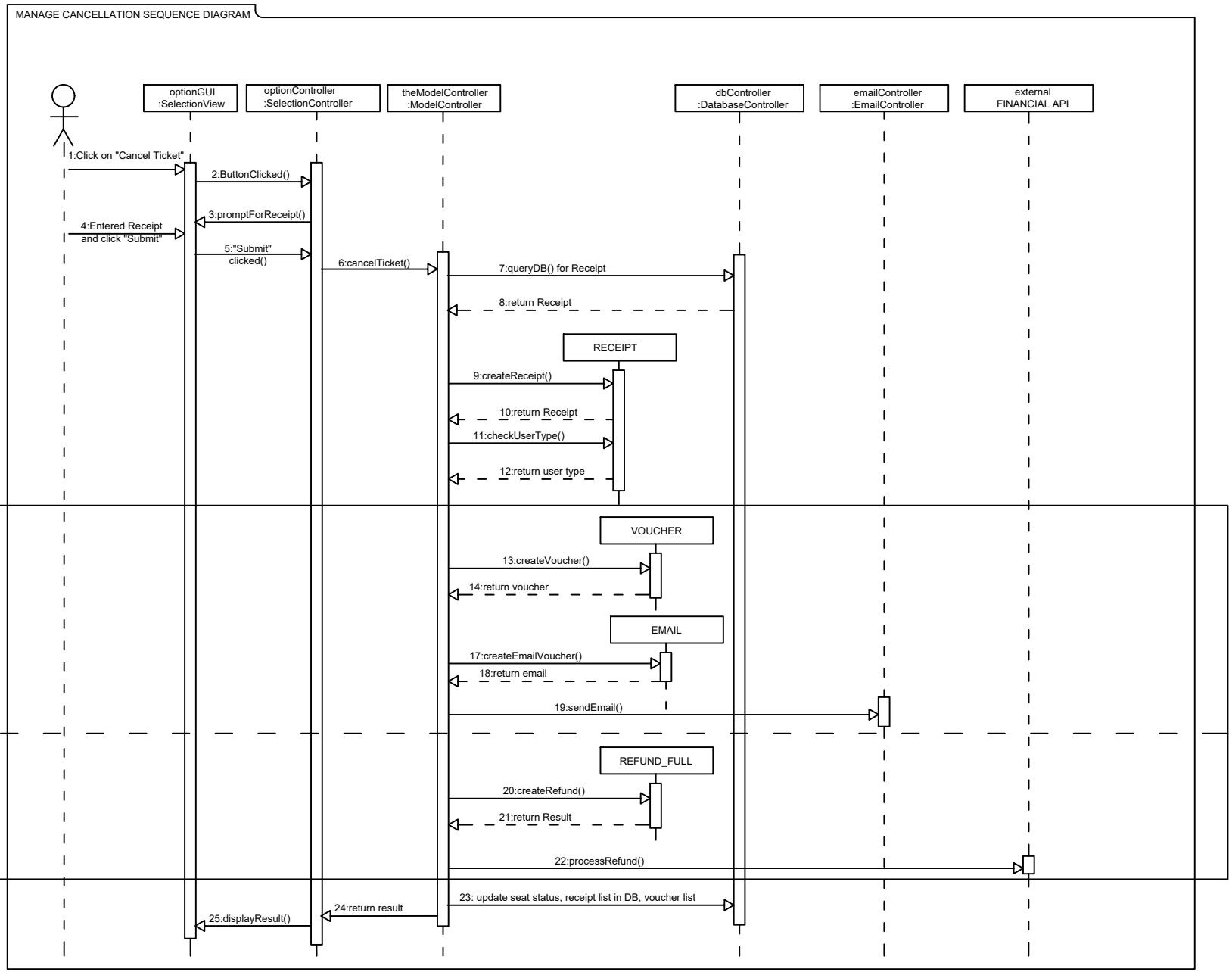
CLASSES DESCRIPTIONS

GENERAL NOTES:

- Assume Database is already established and no modification is needed
- Assume no System Administrator is needed

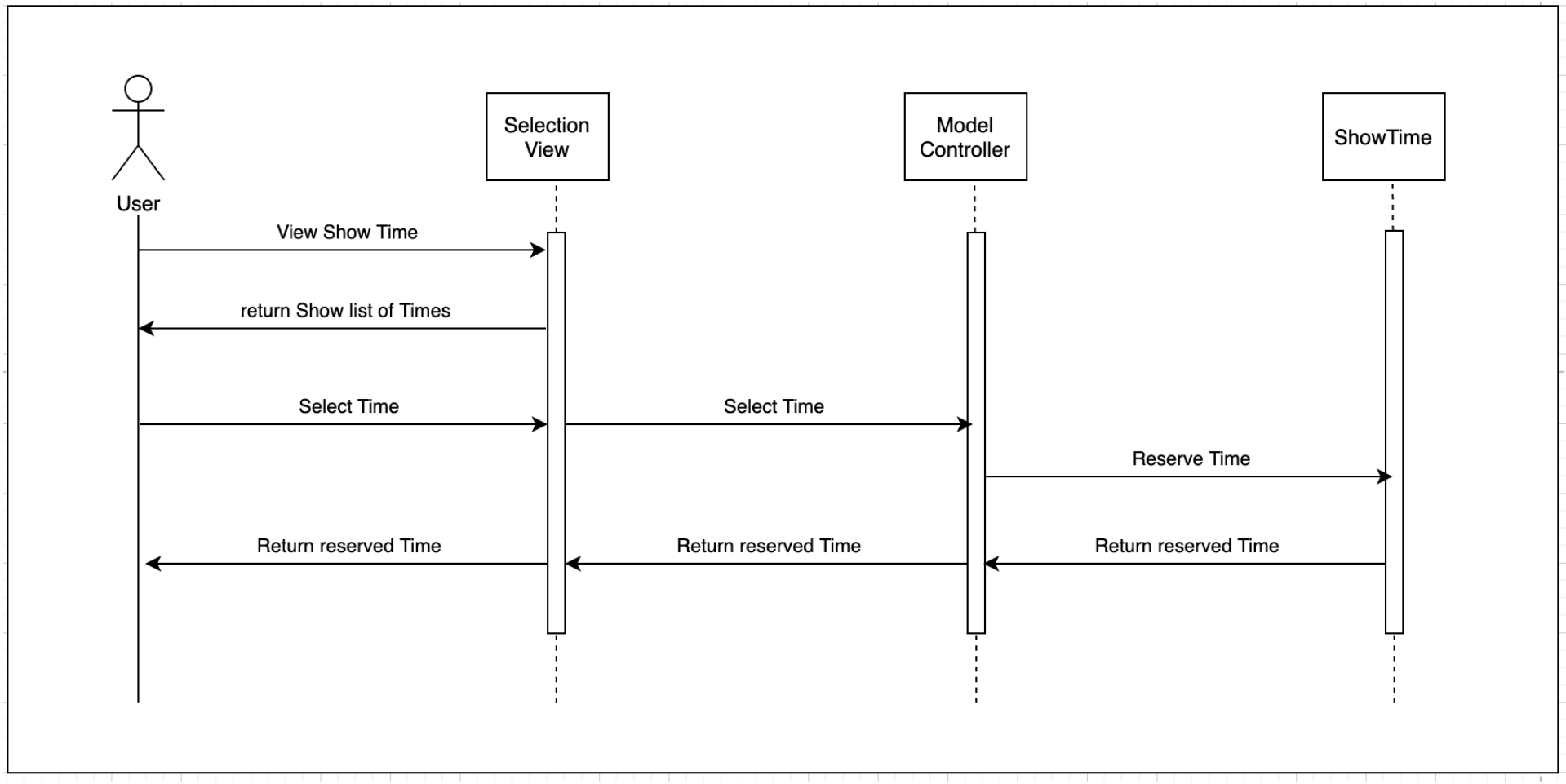
MANAGE CANCELLATION SEQUENCE DIAGRAM

-COMPLETED BY: THIEN NGUYEN



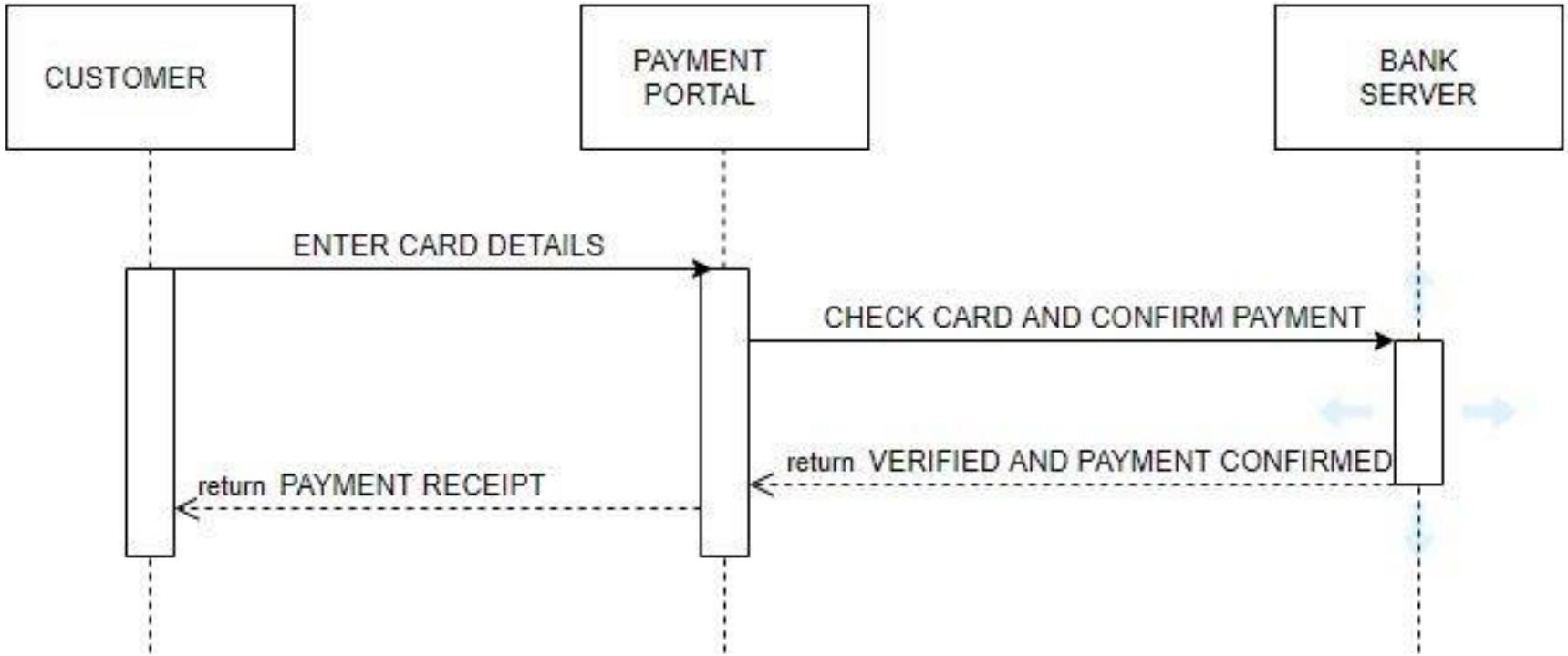
Browse Time Sequence Diagram

Completed by: Sonia Obi



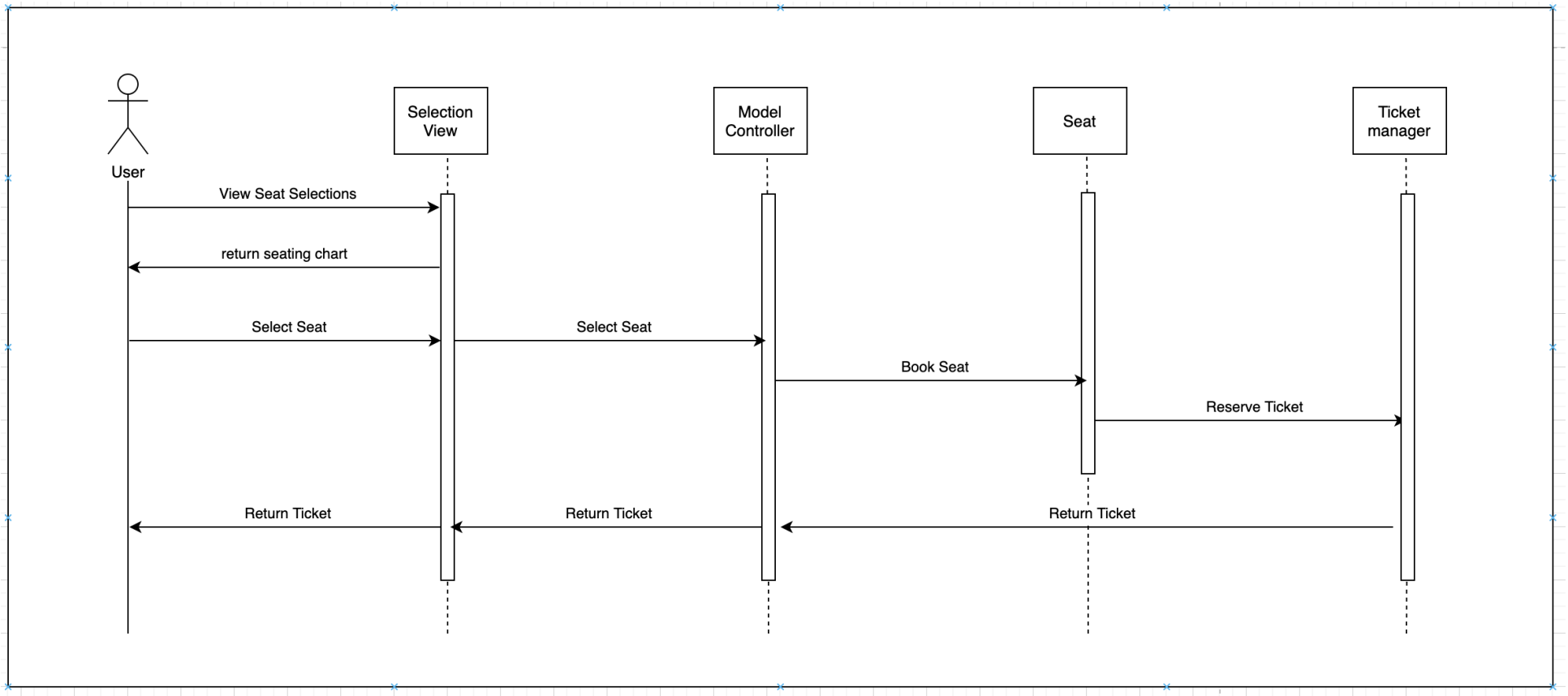
Make Payment Sequence Diagram

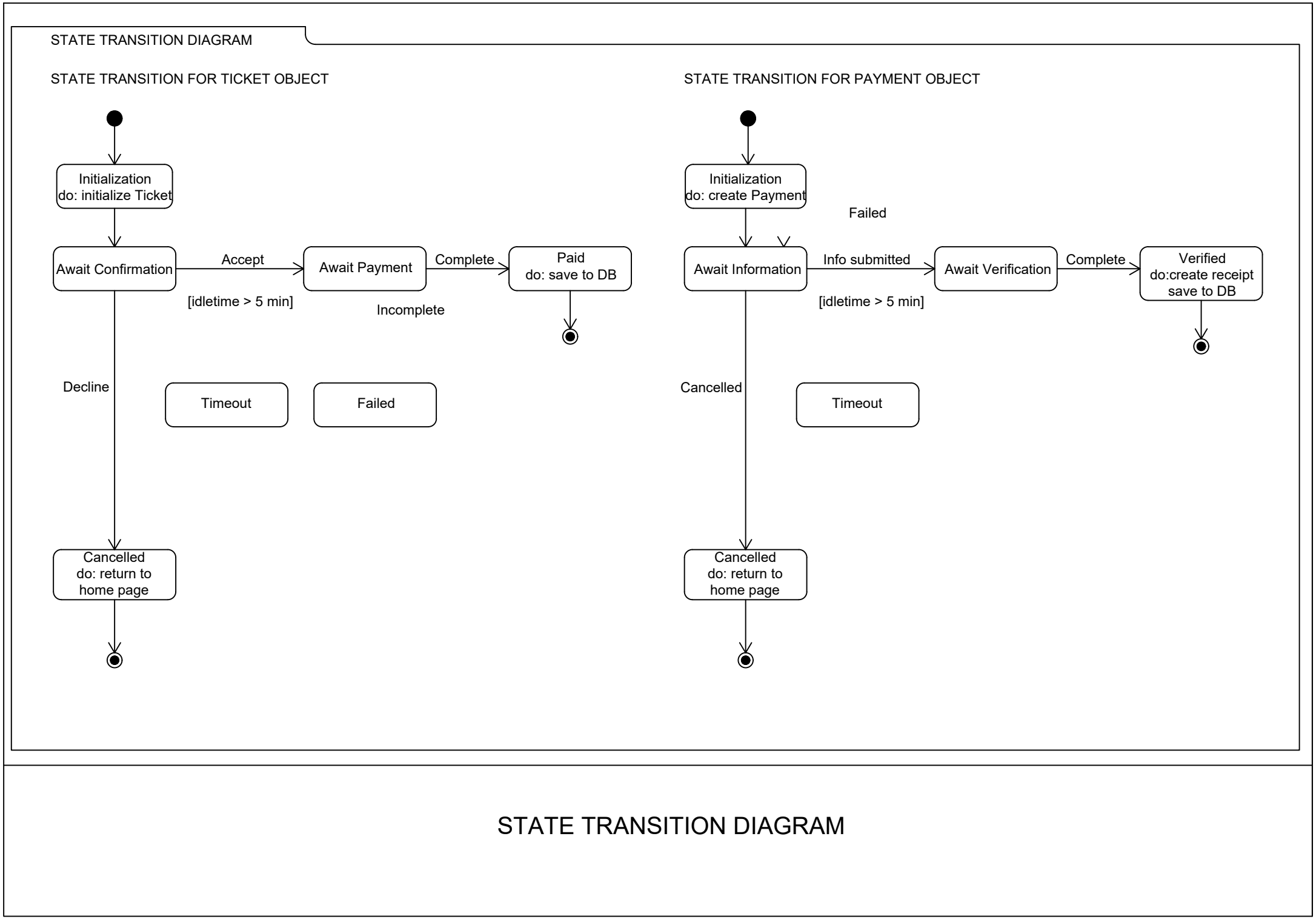
Completed by: Kenechukwu Nwabueze



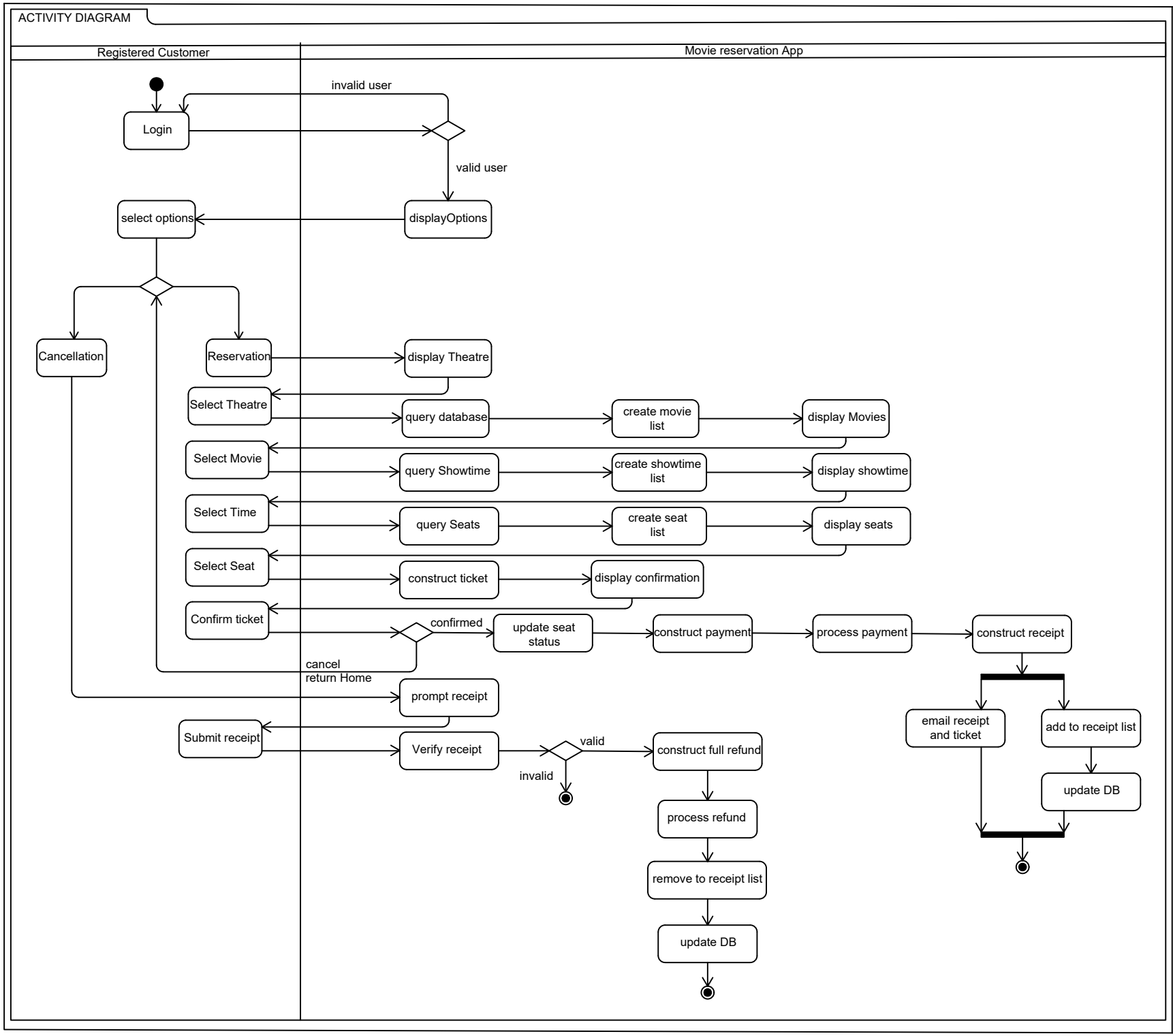
Browse Seat Sequence Diagram

Completed by: Remy Onyegbutulem





STATE TRANSITION DIAGRAM

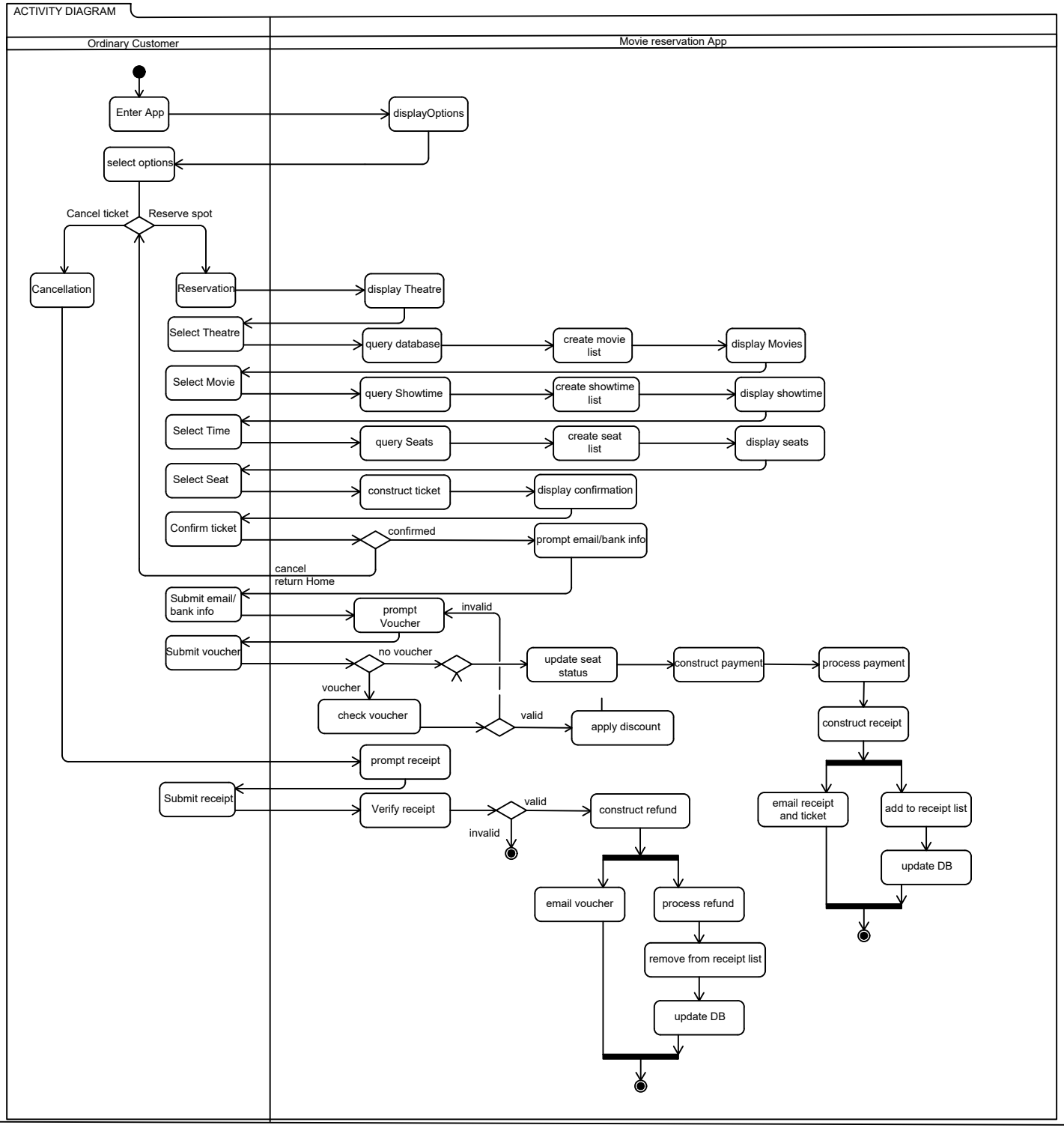


REGISTERED USER ACTIVITY DIAGRAM

ASSUMPTIONS:

-REGISTERED USER WONT NEED TO REDEME VOUCHERS

-PROCESS PAYMENT WILL BE HANDLED EXTERNALLY



ORDINARY USER ACTIVITY DIAGRAM

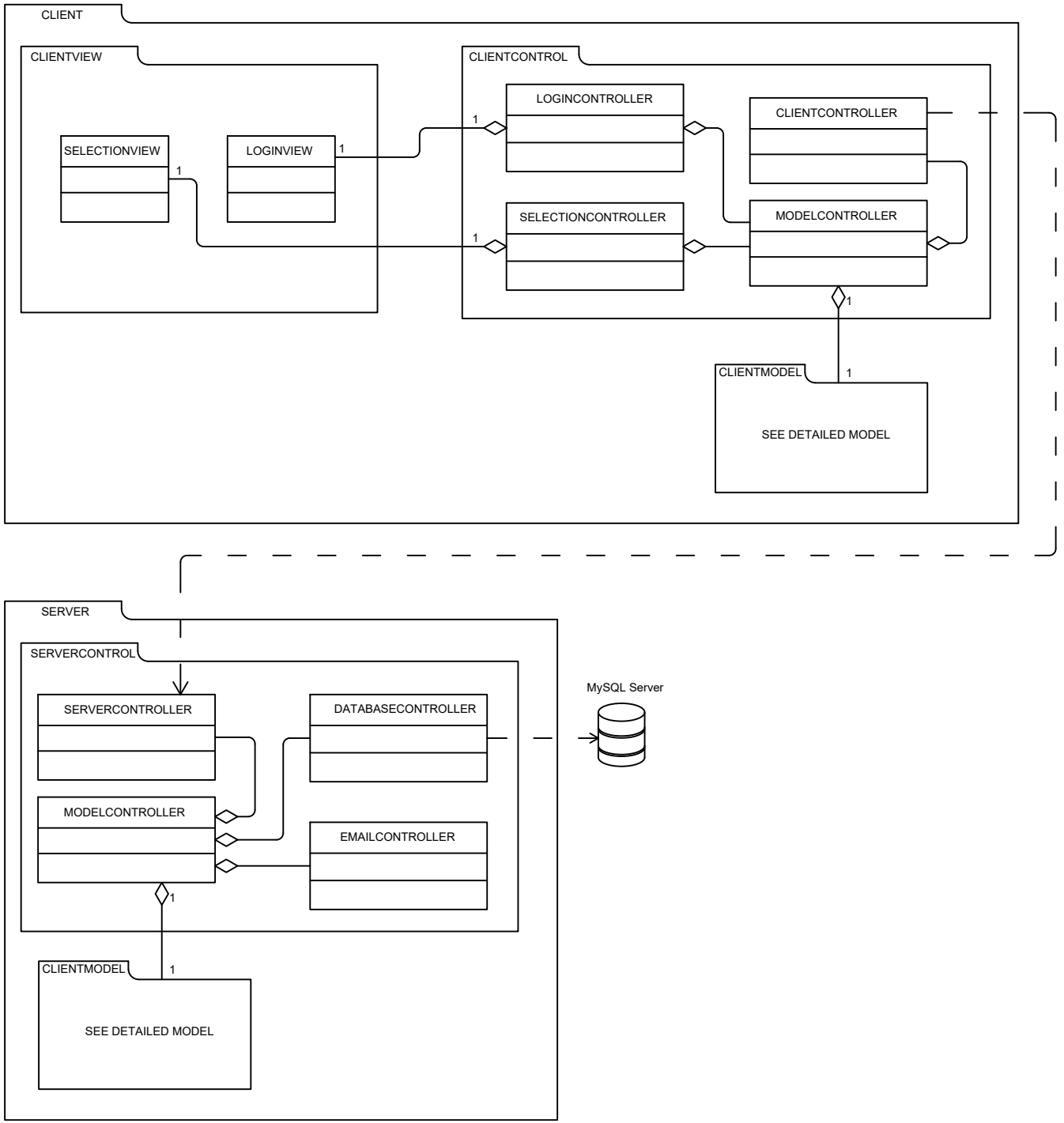
ASSUMPTIONS:

- ORDINARY USER WILL RECEIVE VOUCHER VIA EMAIL

PACKAGE DIAGRAM

NOTE:

-ONLY SHOWING MAJOR OPERATOR CLASSES. FOR MORE INFORMATION, PLEASE REFER TO DETAILED CLASS UML DIAGRAM



Deployment Diagram