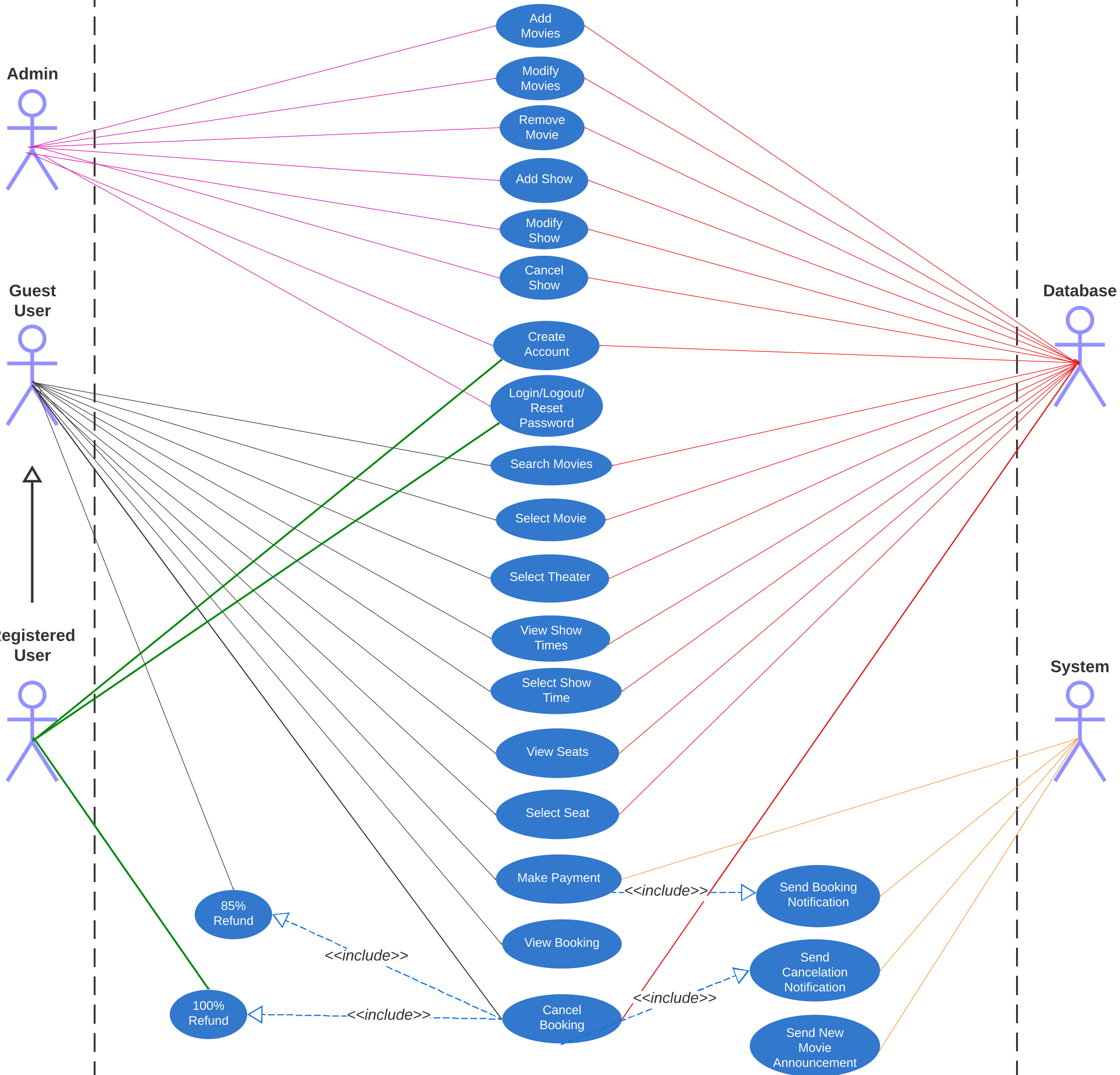


Movie Ticket Booking System



Scenarios

Create Account: User can become a Registered User by creating an account and entering their name, username, password, email and credit card number. Their information will be saved in the database.

Login: When User visits the movie app, they can Login if they have an existing Registered User account. Or continue as a Guest User.

Search Movies: User can view movies by browsing through a dropdown list.

Select Movie: User can select a movie from the movies dropdown list.

Select Theatre: User can select a theater from the theater dropdown list.

Select Showtime: User can select a showtime from the showtimes dropdown list.

View Seats: User can view available, selected and reserved seats graphically.

Select Seats: User can enter the row and column number for the desired available seat. The database will update accordingly to show the seat is selected.

Make payment: User can proceed to checkout and make payment. If it is Registered User there will be an option to use stored banking information from the database. If it is a Guest User, they will be required to enter their name, email and credit card number. The entered information will be validated. Payment will be processed.

View Booking: A receipt will pop up on the screen when the payment is processed. It will display details of the payment: Ticket ID, Movie Name, Theatre Name, Screening Room, and Seat.

Cancel Reservation: User can cancel a reservation up to 72 hours prior to the showtime by entering the ticket ID. A Registered User will receive 100% refund. A Guest User will receive an 85% voucher for a future purchase which is valid for 1 year. Database will be updated accordingly upon cancellation.

Send Reservation/Cancellation Notification: User will receive emails with their tickets and receipts for a reservation. For cancellations, Registered users will be notified of the 100% refund and Guest Users will be emailed a voucher.

Send Movie Announcement: Register Users will be emailed about new movie announcements.

Add Movie: Admin can add a new movie to the database.

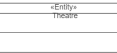
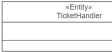
Modify Movie: Admin can change the details of a movie like the description.

Remove Movie: Admin can remove a movie from the database.

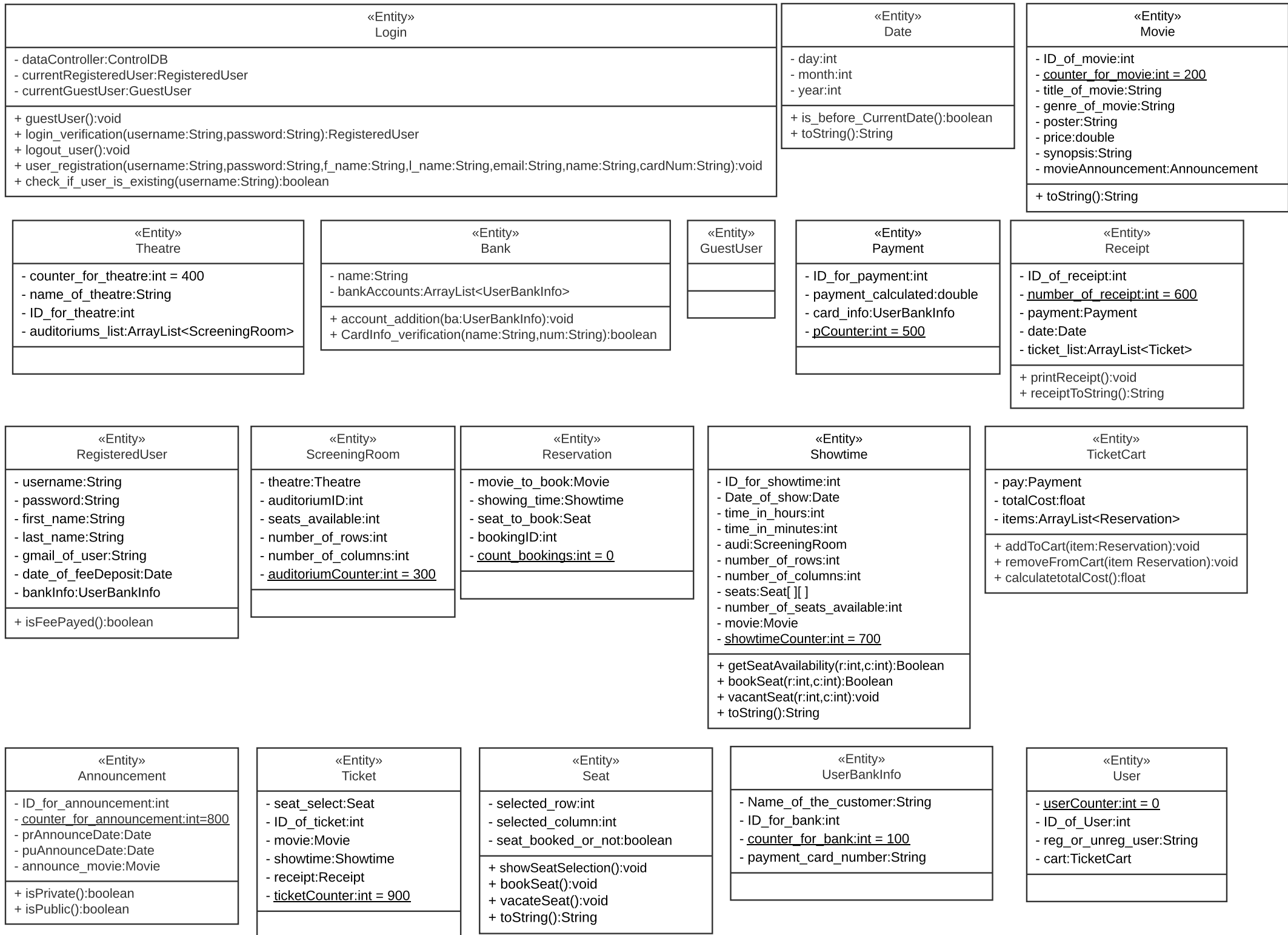
Add Show: Admin can add a showtime for a movie and update the database.

Modify Show: Admin can modify a showtime for a movie and update the database.

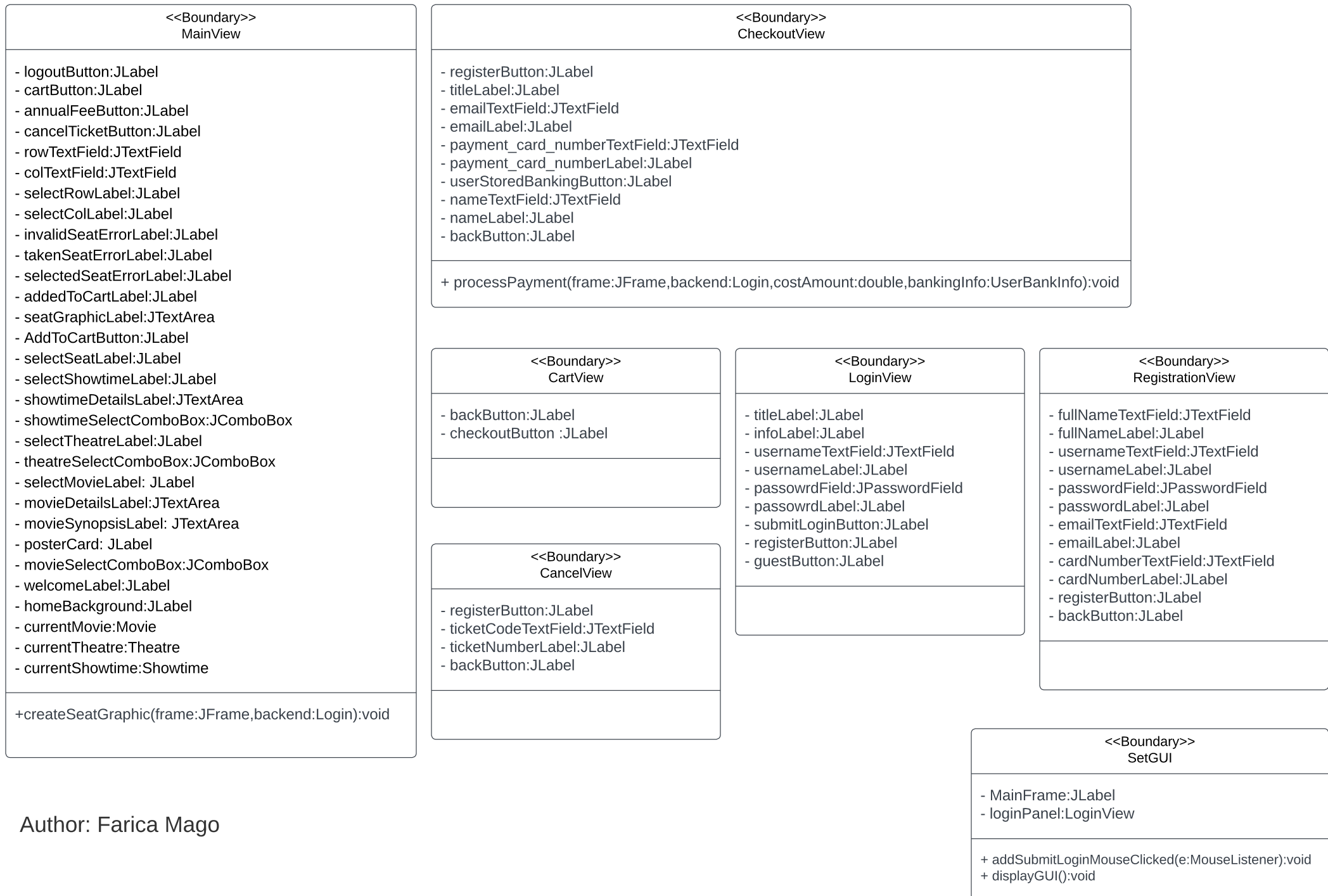
Remove Show: Admin can remove a showtime for a movie and update the database.



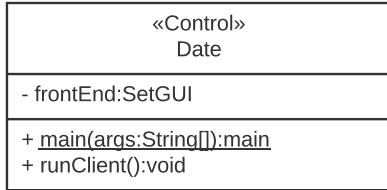
Class Diagrams: Major Attributes and Operations for Entity Classes



Class Diagrams: Major Attributes and Operations for Boundary Classes



Class Diagrams: Major Attributes and Operations for Control Classes



Author: Farica Mago

Class Diagrams: Major Attributes and Operations for Database Classes

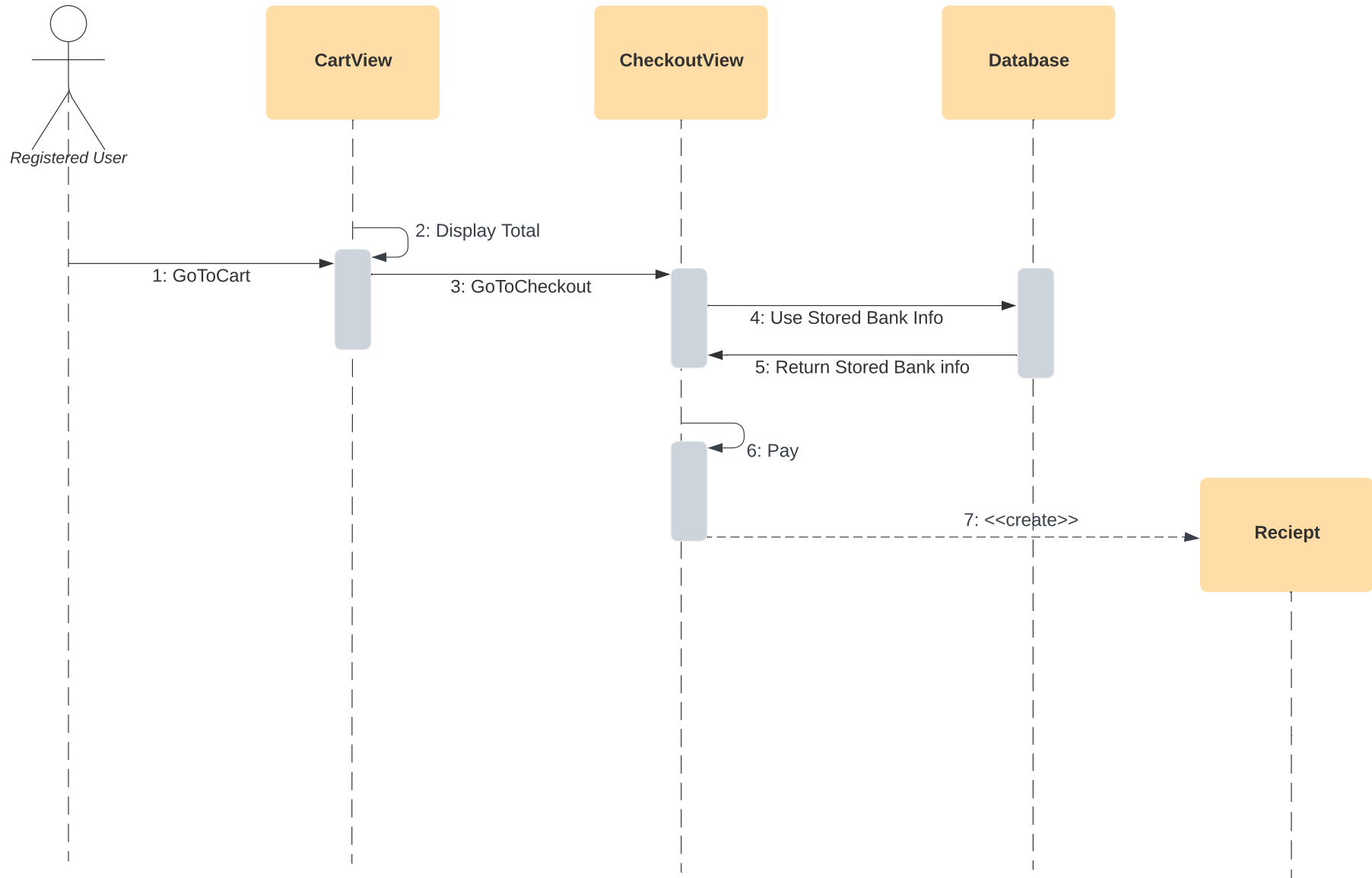
Farica | December 5, 2022

| ControlDB |
|--|
| <ul style="list-style-type: none">- <u>object:ControlDB</u>- list_of_users:ArrayList<RegisteredUser>- list_of_announcements:ArrayList<Announcements>- list_of_auditoriums:ArrayList<ScreeningRoom>- list_of_pay:ArrayList<Payment>- list_of_receipts:ArrayList<Receipt>- list_of_movies:ArrayList<Movie>- list_of_showtimes:ArrayList<Showtime>- list_of_theatres:ArrayList<Theatre>- list_of_banks:ArrayList<UserBankInfo>- list_of_tickets:ArrayList<Ticket>- inst:Bank |
| <ul style="list-style-type: none">+ findMovie(title_of_movie:String):Movie+ findTheatre(name:String):Theatre+ findAllShowtime(searchMovie:Movie,searchTheatre:Theatre):ArrayList<Showtime>+ findShowtime(searchMovie:Movie,searchTheatre:Theatre,date:Date):Showtime+ findTheatresPlayingMovie(searchMovie:Movie):ArrayList<Theatre>+ searchMovie(id:int):Movie+ searchTheatre(id:int):Theatre+ searchAuditorium(id:int):ScreeningRoom+ searchBankingInfo(id:int):UserBankInfo+ searchShowtimeInfo(id:int):Showtime+ addMovie(movie:Movie):void+ addShowtime(showtime:Showtime):void+ addTheatre(theatre:Theatre):void+ addTicket(ticket:Ticket):void+ removeTicket(ticket:Ticket):void+ addUser(user:RegisteredUser):void+ addAnnoucement(announcement:Announcement):void+ addAuditorium(auditorium:ScreeningRoom):void+ addBankingInfo(b:UserBankInfo):void+ addPayment(p:Payment):void+ addReceipt(list_of_receipts:Receipt):void |

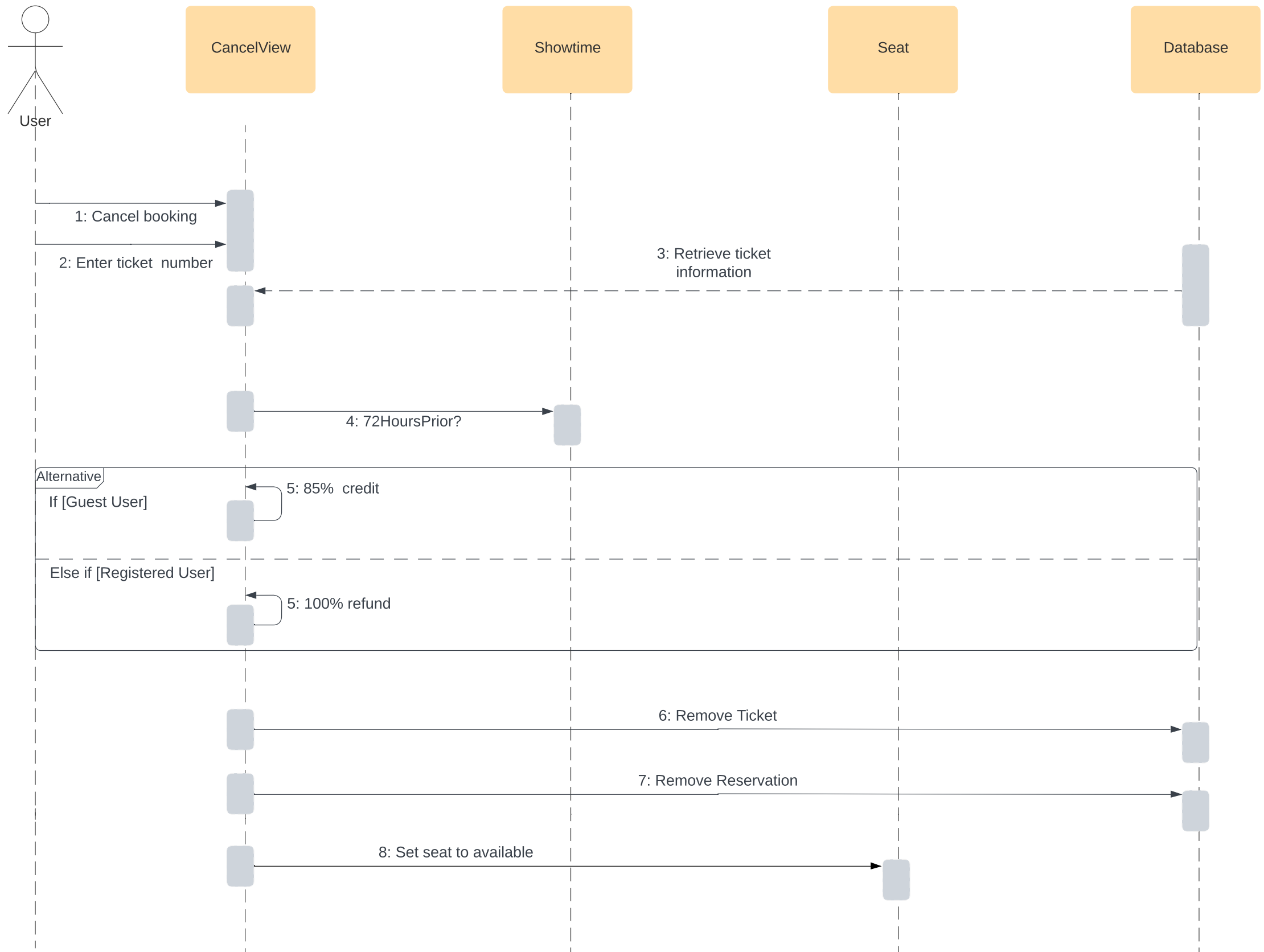
| ReadDB |
|---|
| <ul style="list-style-type: none">~ database_control:ControlDB |
| <ul style="list-style-type: none">+ loadDatabase():void+ gettingMovie():void+ gettingTheatres():void+ gettingAuditoriums():void+ gettingShowtimes():void+ gettingUsers():void+ gettingAnnouncements():void+ gettingBankInfo():void |

| UpdateDB |
|---|
| <ul style="list-style-type: none">~ databaseController:ControlDB |
| <ul style="list-style-type: none">+ write_to_Database():void+ writeRegisteredUser():void |

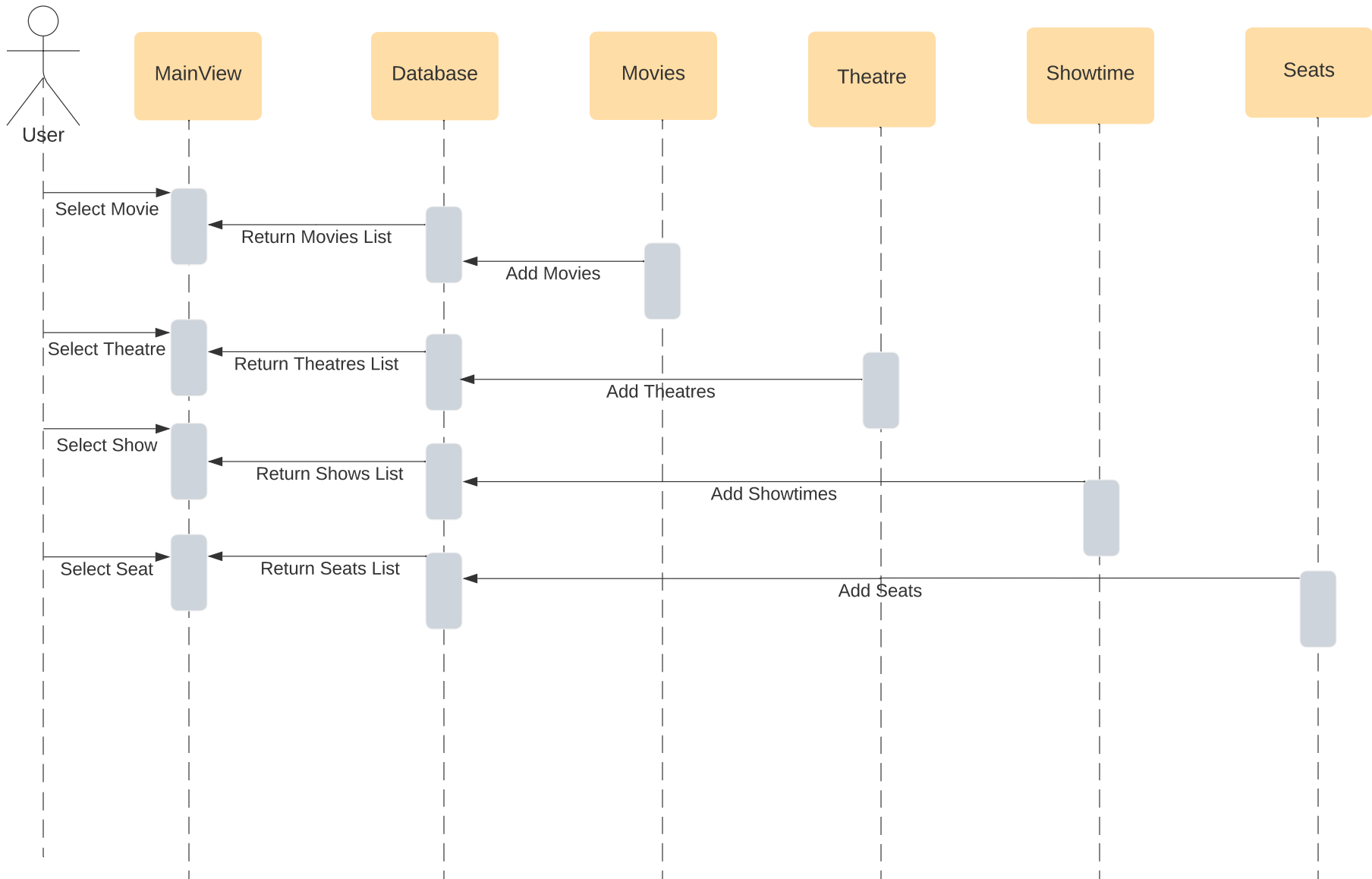
Registered User Makes a Payment



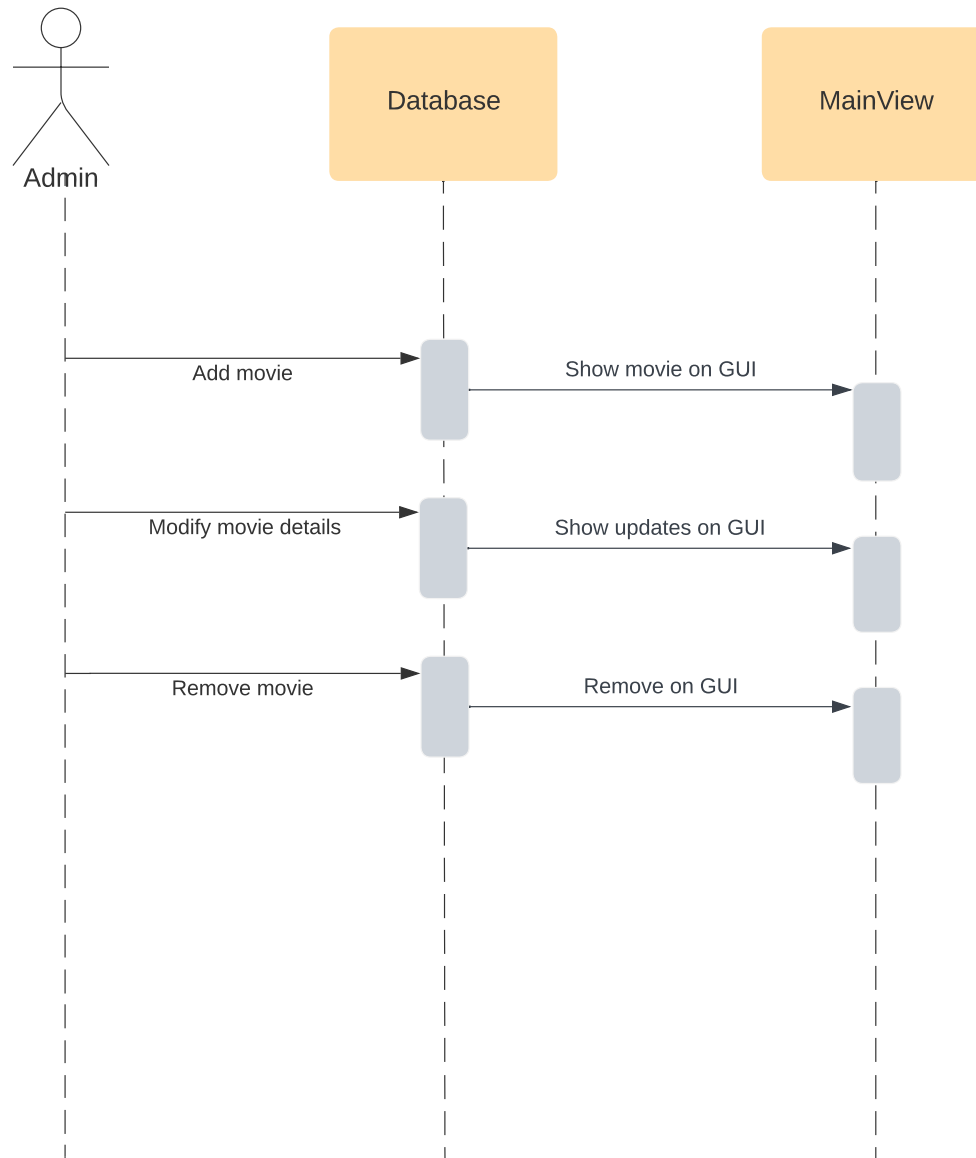
User Cancels a Ticket



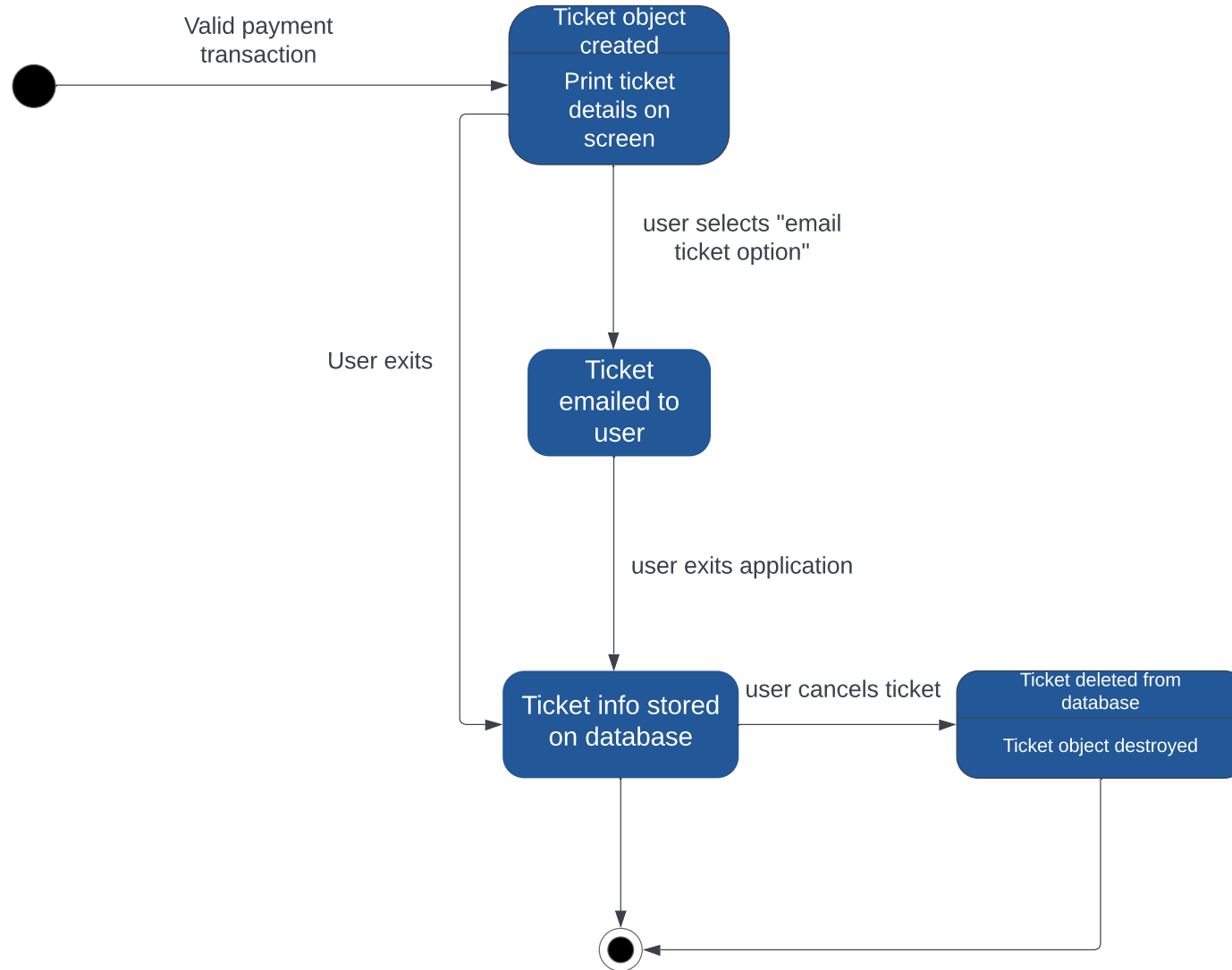
User selects movie, theatre, showtime and seat



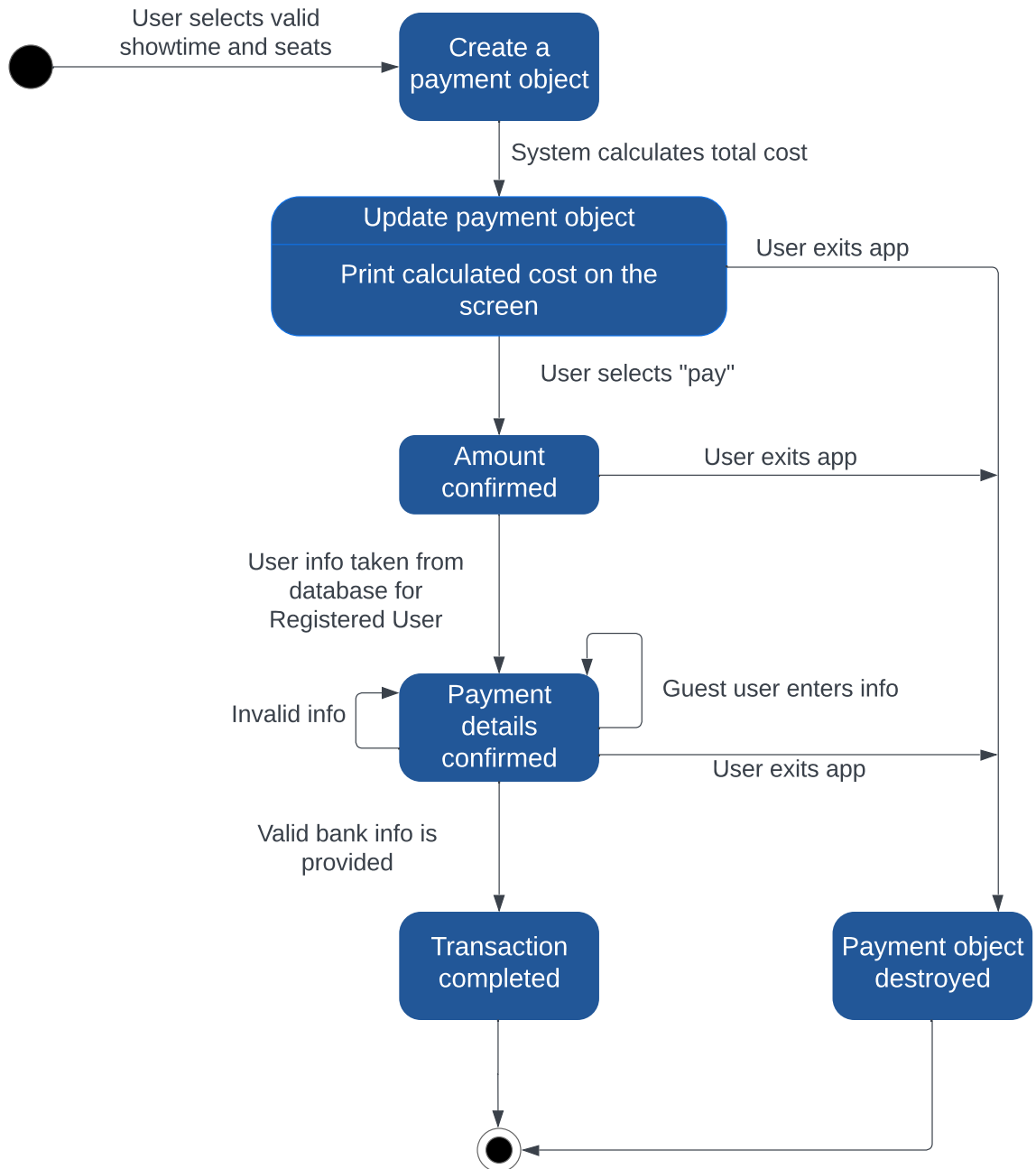
Admin Adds , Modifies and Removes Movies



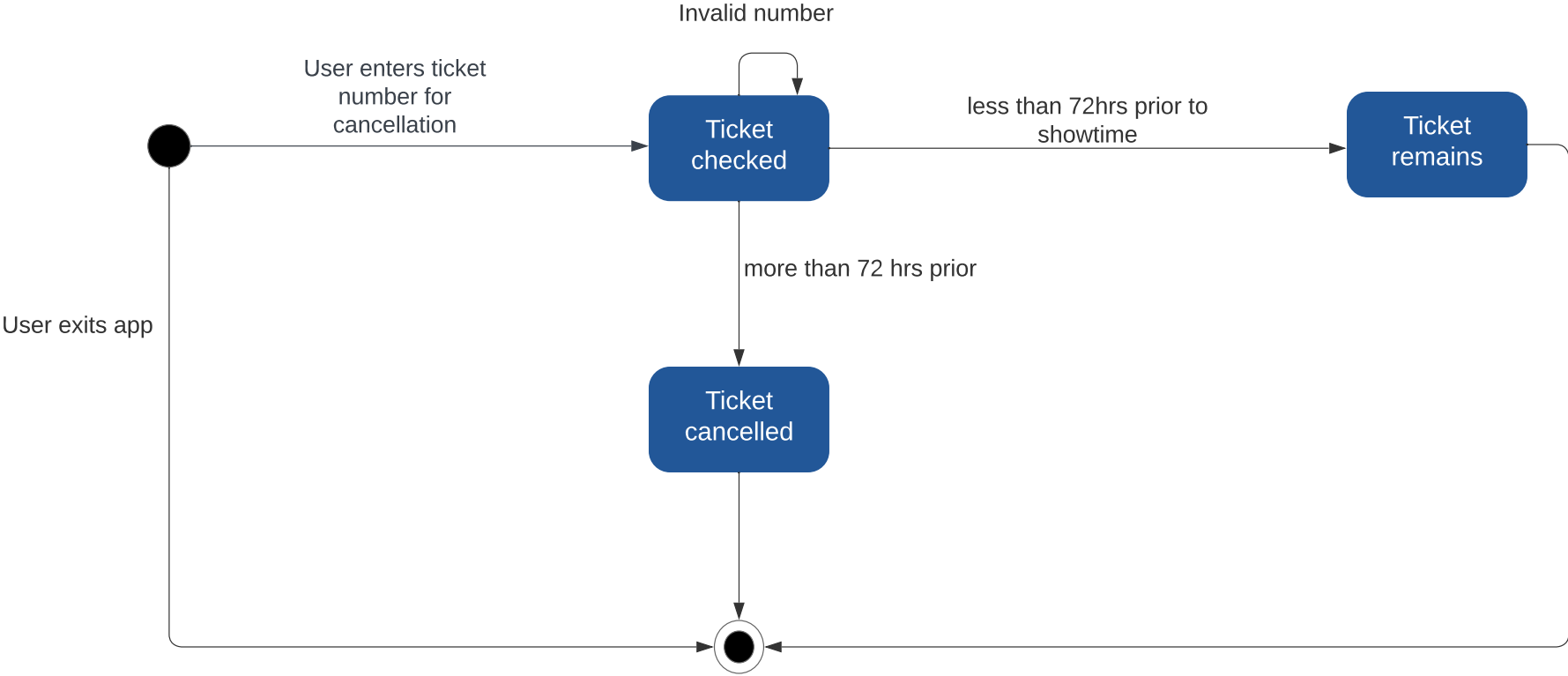
State Transition: Ticket Object



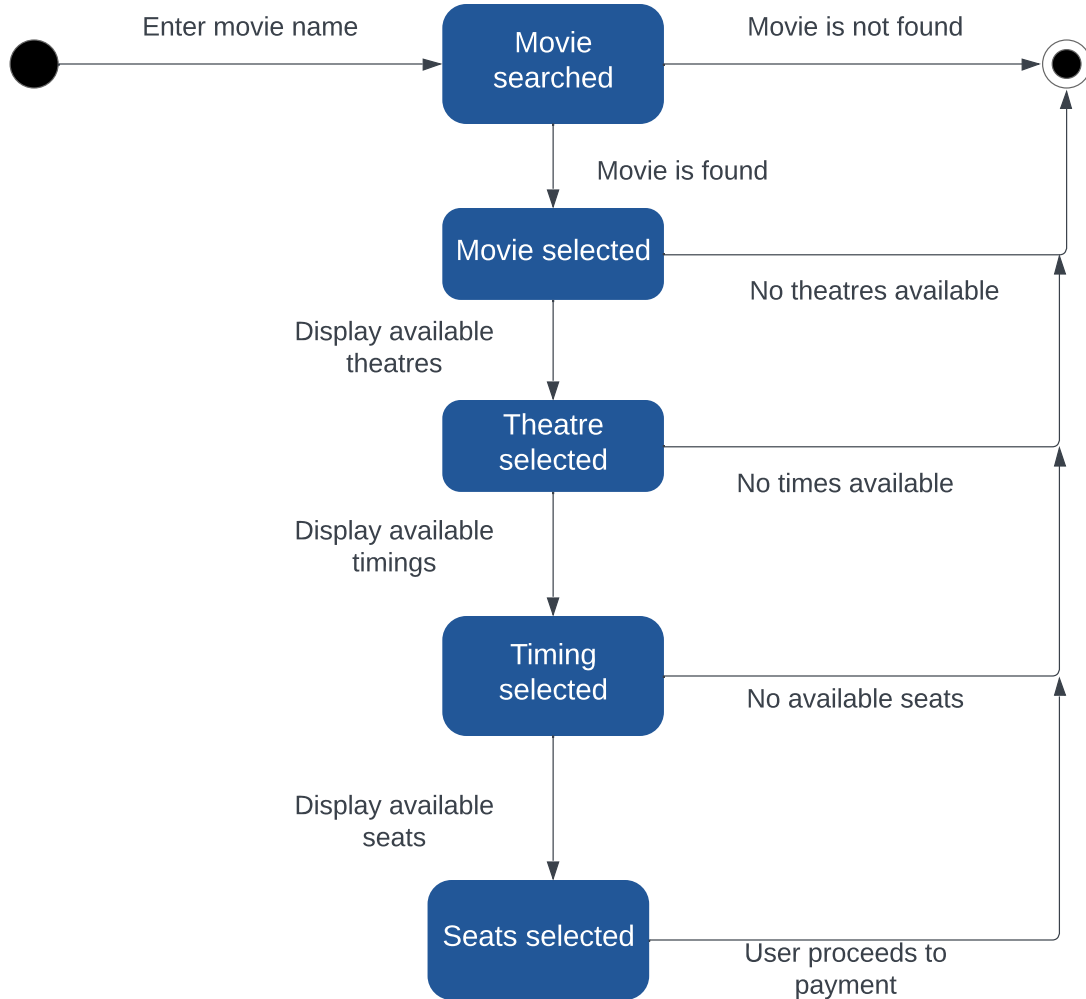
State Transition: Payment Object



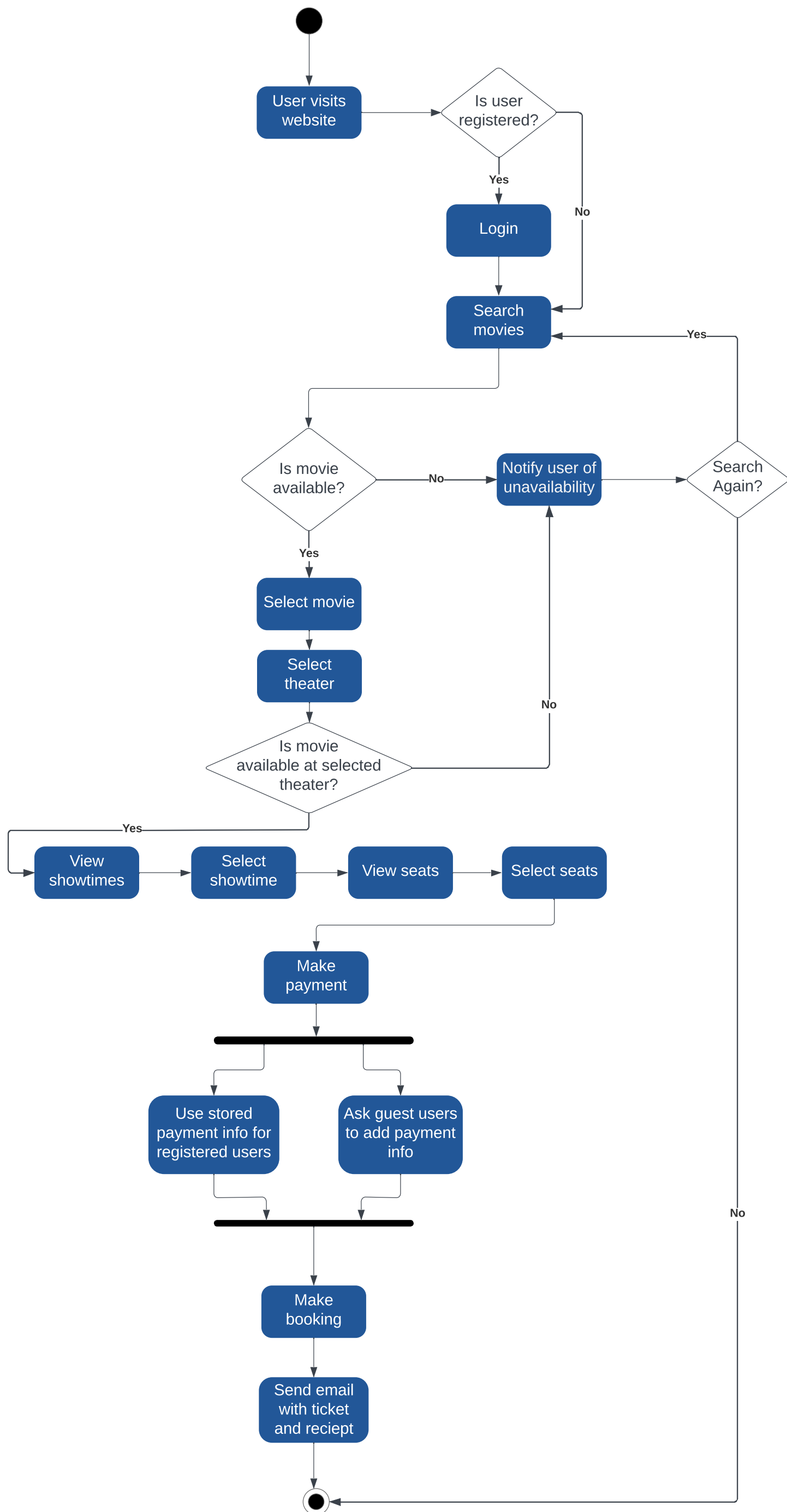
State Transition: Cancel Ticket



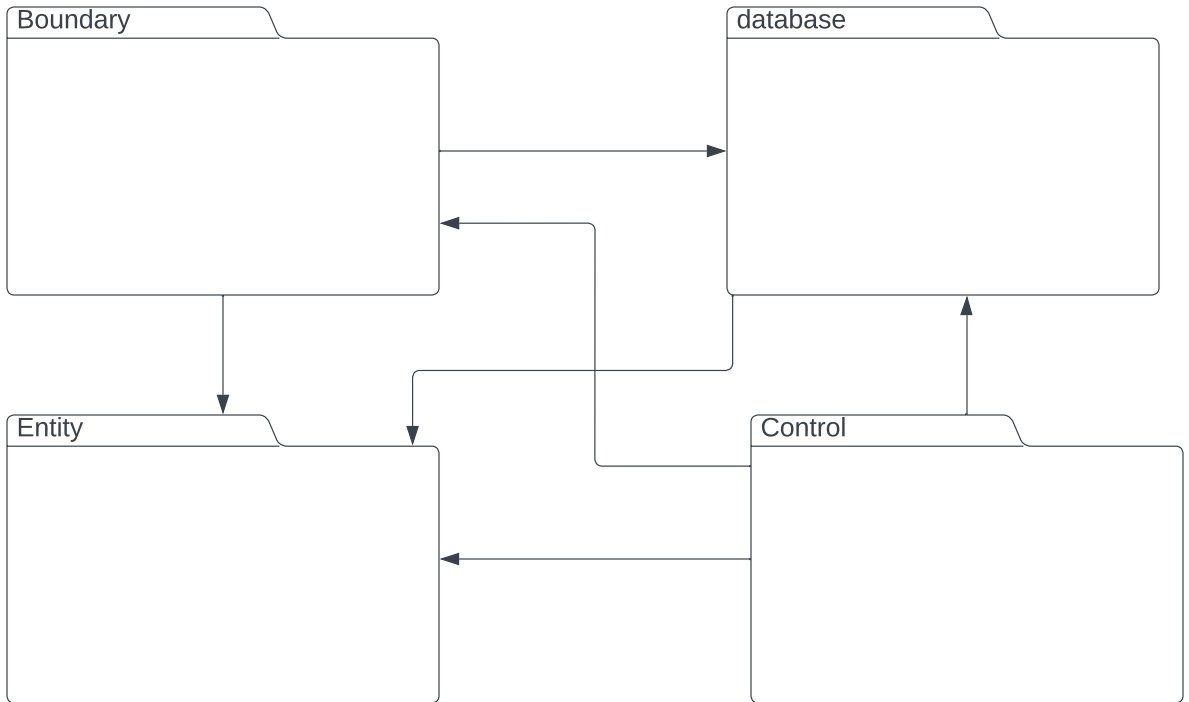
State Transition: Select Movie, Theatre, Showtime



System Activity Digram: Registered or Guest User makes a Movie Booking

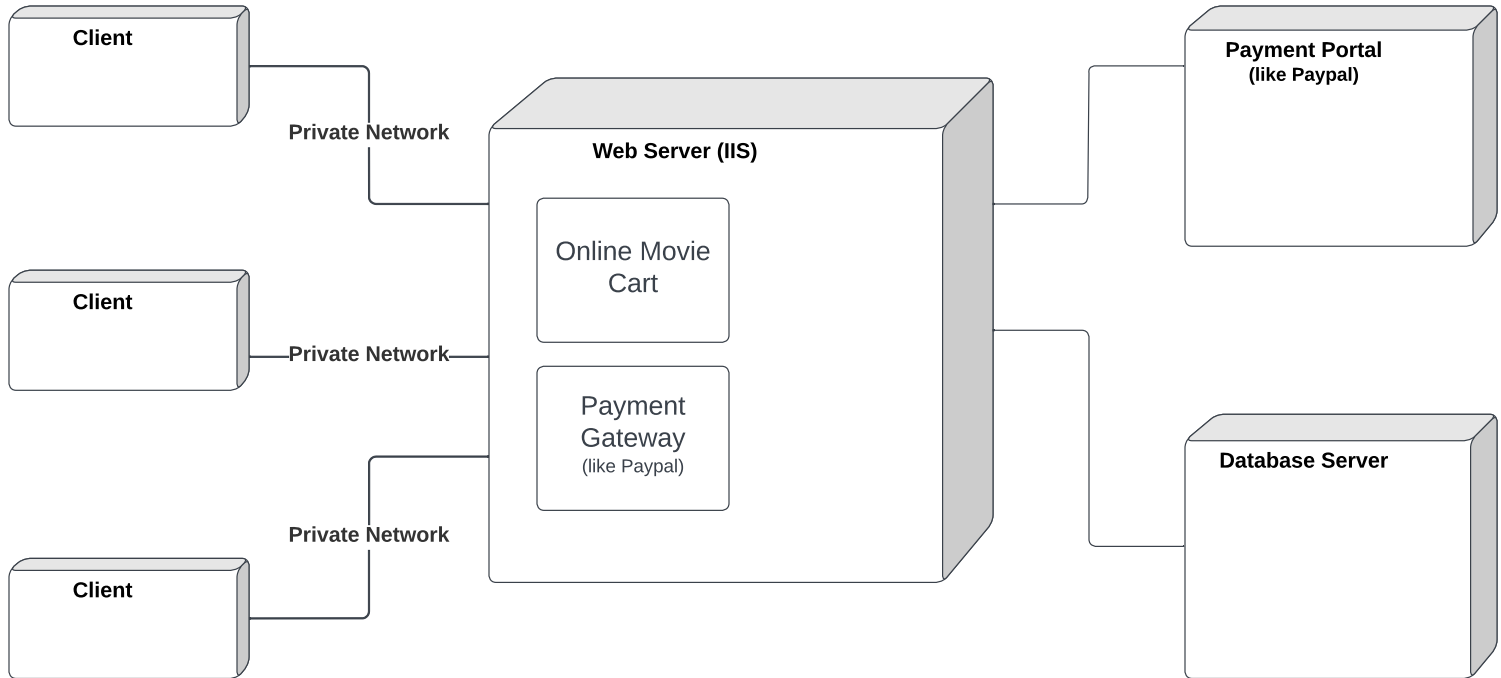


Package Diagram



Author: Aarushi Roy Choudhury

Deployemnt Diagram



Author: Aarushi Roychoudhury