צוות הפרויקט:

שי שילה, 204684914, אמיר

אופק דאידה, 315143958, אמיר

נתיב שלום, 207353426, אמיר

יובל אמר, 318620721, מירה

המתרגל אליו מוגש הפרויקט: אמיר לויט.
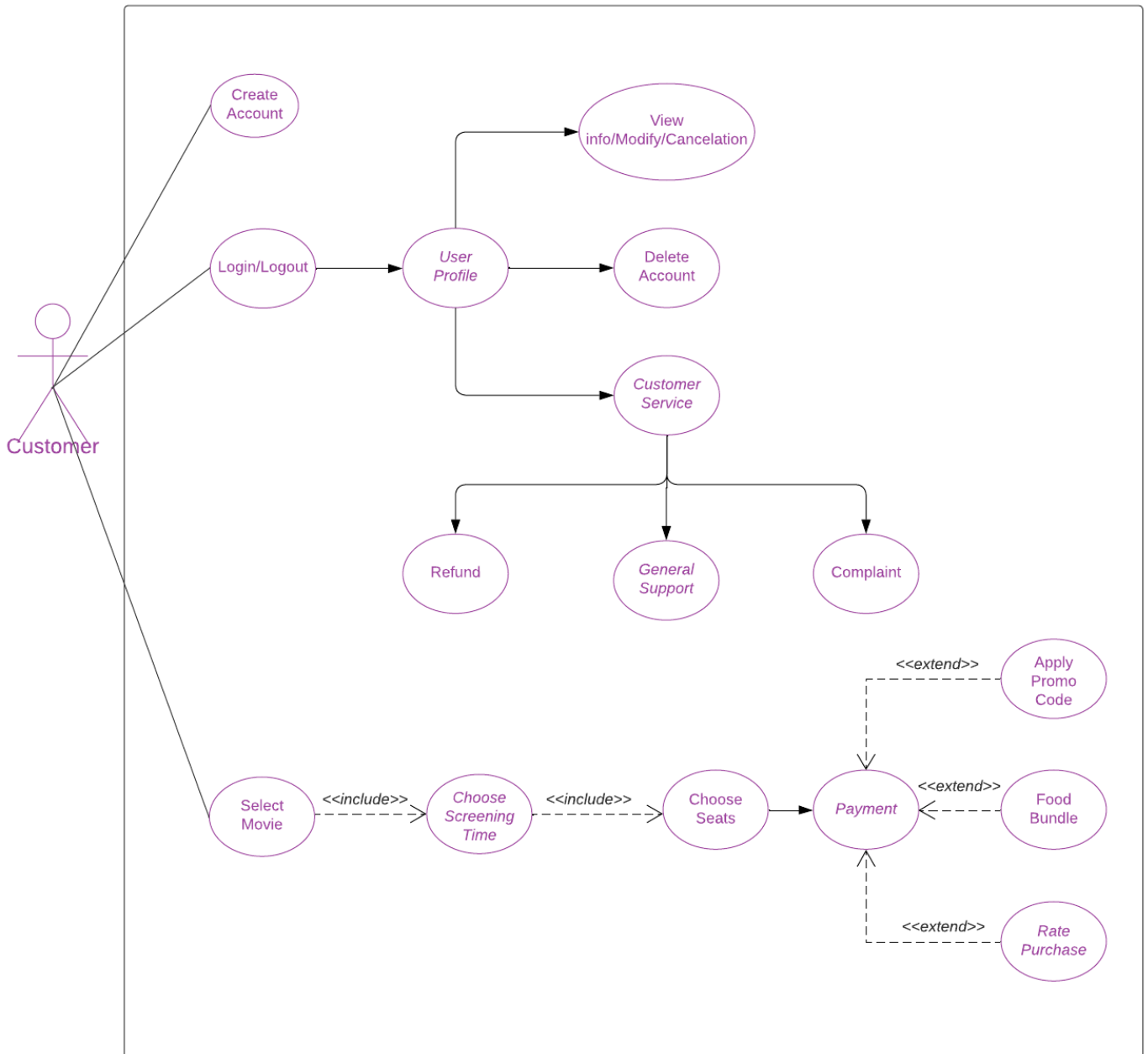
SeatFinder

מסמך ארכיטקטורה

Use-Case Diagrams

1)

2)

Admin

- Login/Logout
- Add New Movie
- Edit Movie Details
- Modify Cinema Locations
- Manage Accounts
  - Customer Accounts
  - Employee Accounts
- View Customer requests
  - Approve Request
  - Deny Request
- Remove Movie
- Financial Mangement
  - View Report
  - Export Report
  - Manage promotions and discounts
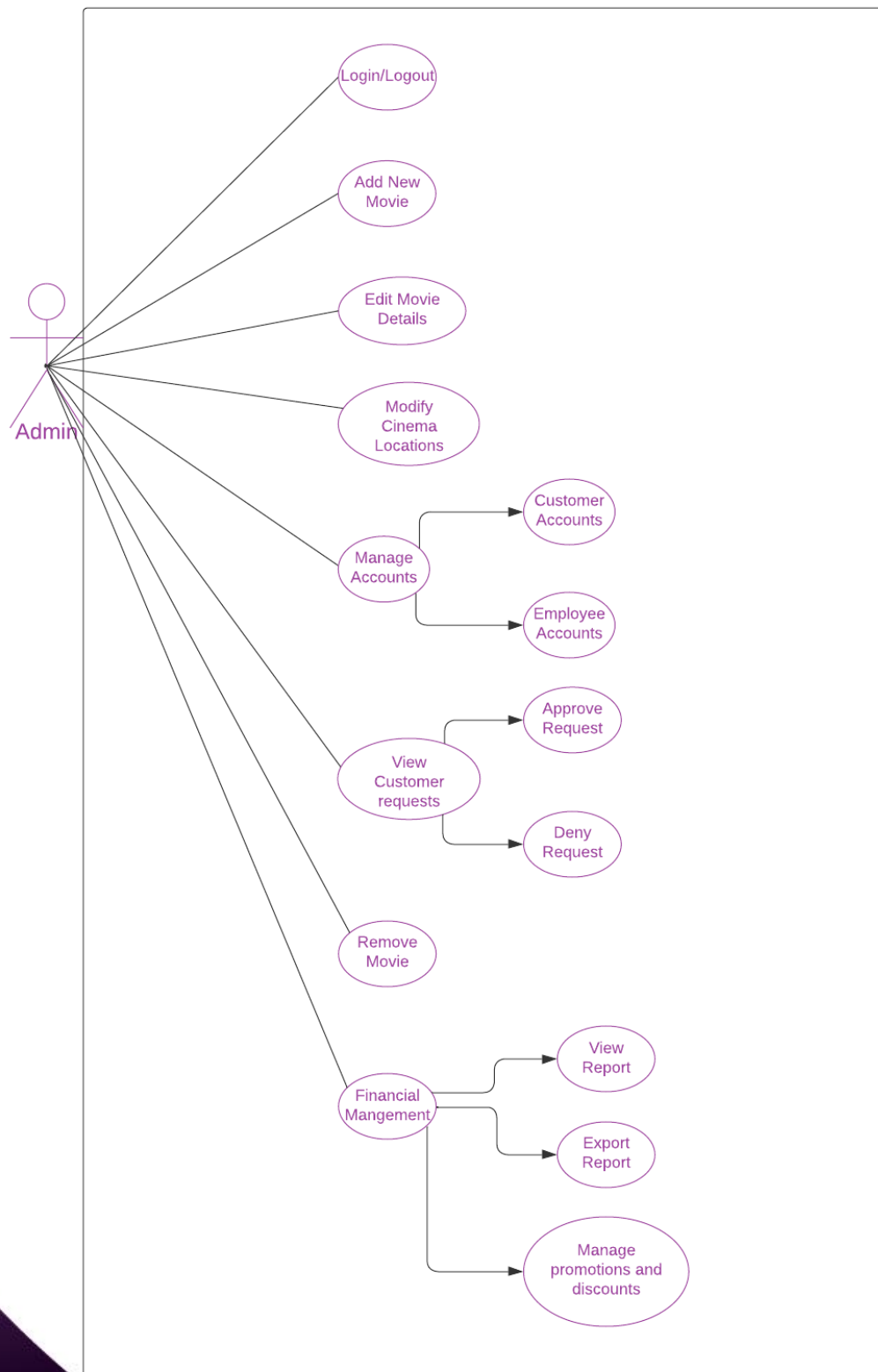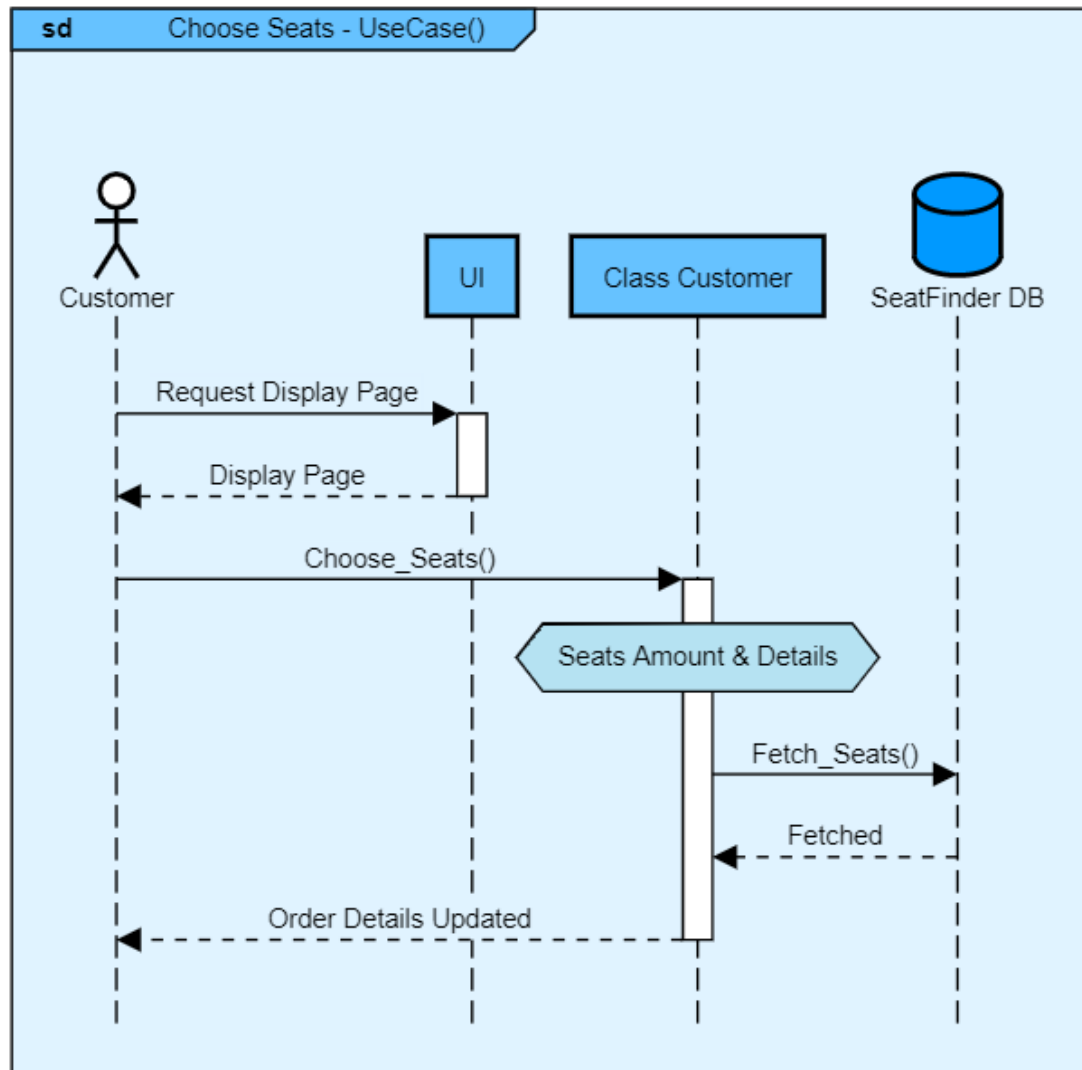
Sequence Diagrams

1)


```
title Choose Seats

actor Customer
participant UI #66c2ff
participant Class Customer #66c2ff
frame #66c2ff           Choose Seats - UseCase()
database SeatFinder DB #0099ff

Customer->UI:Request Display Page
activate UI
Customer<--UI:Display Page
deactivate UI

Customer->Class Customer:Choose_Seats()
activate Class Customer
abox over Class Customer #B5E2F2:Seats Amount & Details
Class Customer->SeatFinder DB:Fetch_Seats()
SeatFinder DB-->Class Customer:Fetched
Class Customer-->Customer: Order Details Updated
deactivate Class Customer
```

2)
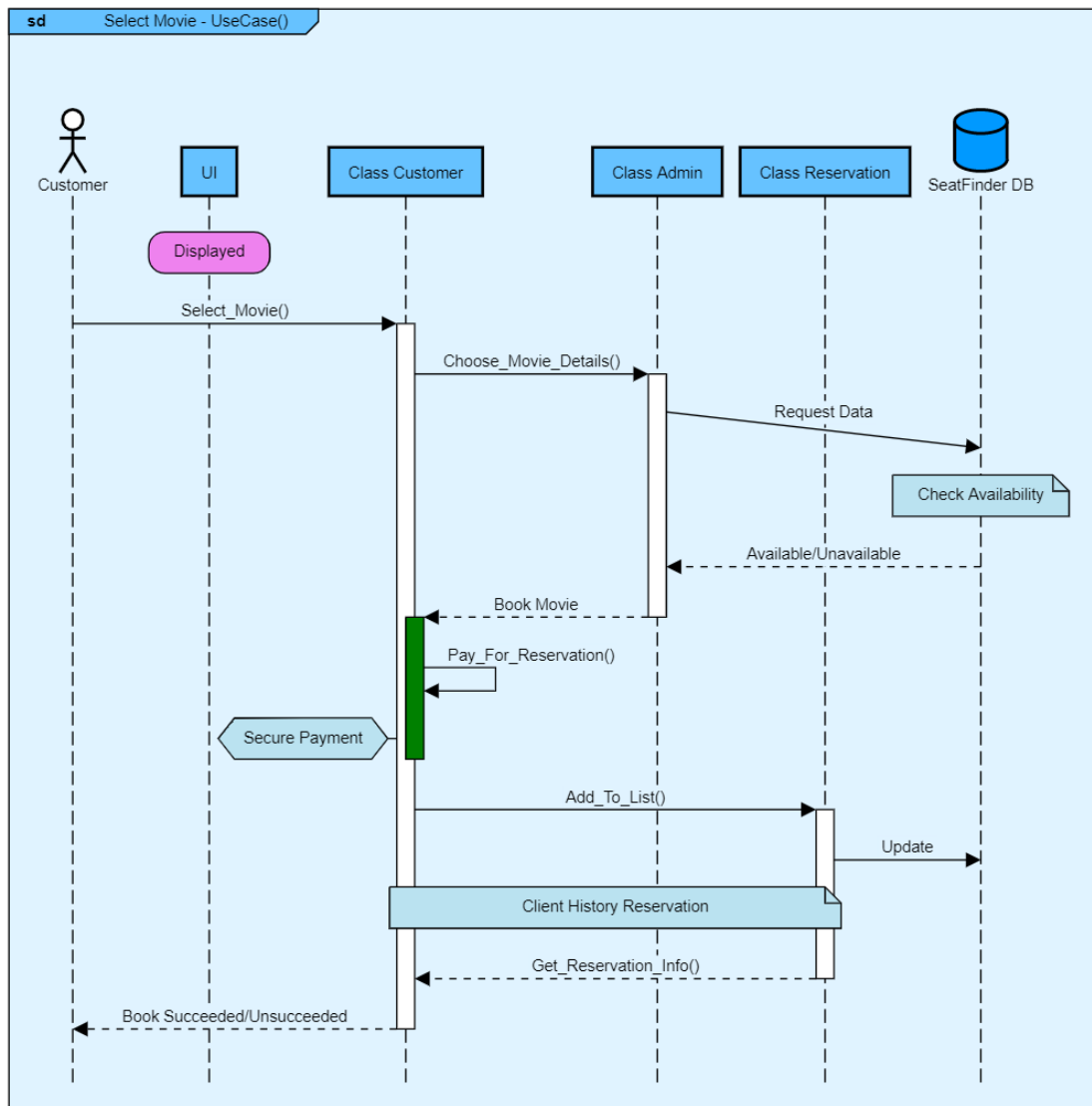

```
title Select Movie

actor Customer
participant UI #66c2ff
participant Class Customer #66c2ff
participant Class Admin #66c2ff
participant Class Reservation #66c2ff
frame #66c2ff            Select Movie - UseCase()
database SeatFinder DB #0099ff

rbox over UI #violet:Displayed

Customer->Class Customer:Select_Movie()
activate Class Customer
Class Customer->Class Admin:Choose_Movie_Details()
activate Class Admin
Class Admin->(2)SeatFinder DB:Request Data

note over SeatFinder DB#BAE2F0:Check Availability
SeatFinder DB-->Class Admin: Available/Unavailable
Class Admin-->Class Customer:Book Movie
deactivate Class Admin
activate Class Customer #green

Class Customer->Class Customer: Pay_For_Reservation()
abox left of Class Customer#BAE2F0: Secure Payment
deactivate Class Customer

Class Customer->Class Reservation:Add_To_List()
activate Class Reservation
Class Reservation->SeatFinder DB:Update
note over Class Customer,Class Reservation#BAE2F0:Client
History Reservation
Class Reservation-->Class
Customer:Get_Reservation_Info()
deactivate Class Reservation
Class Customer-->Customer:Book Succeeded/Unsucceeded
deactivate Class Customer
```

3)
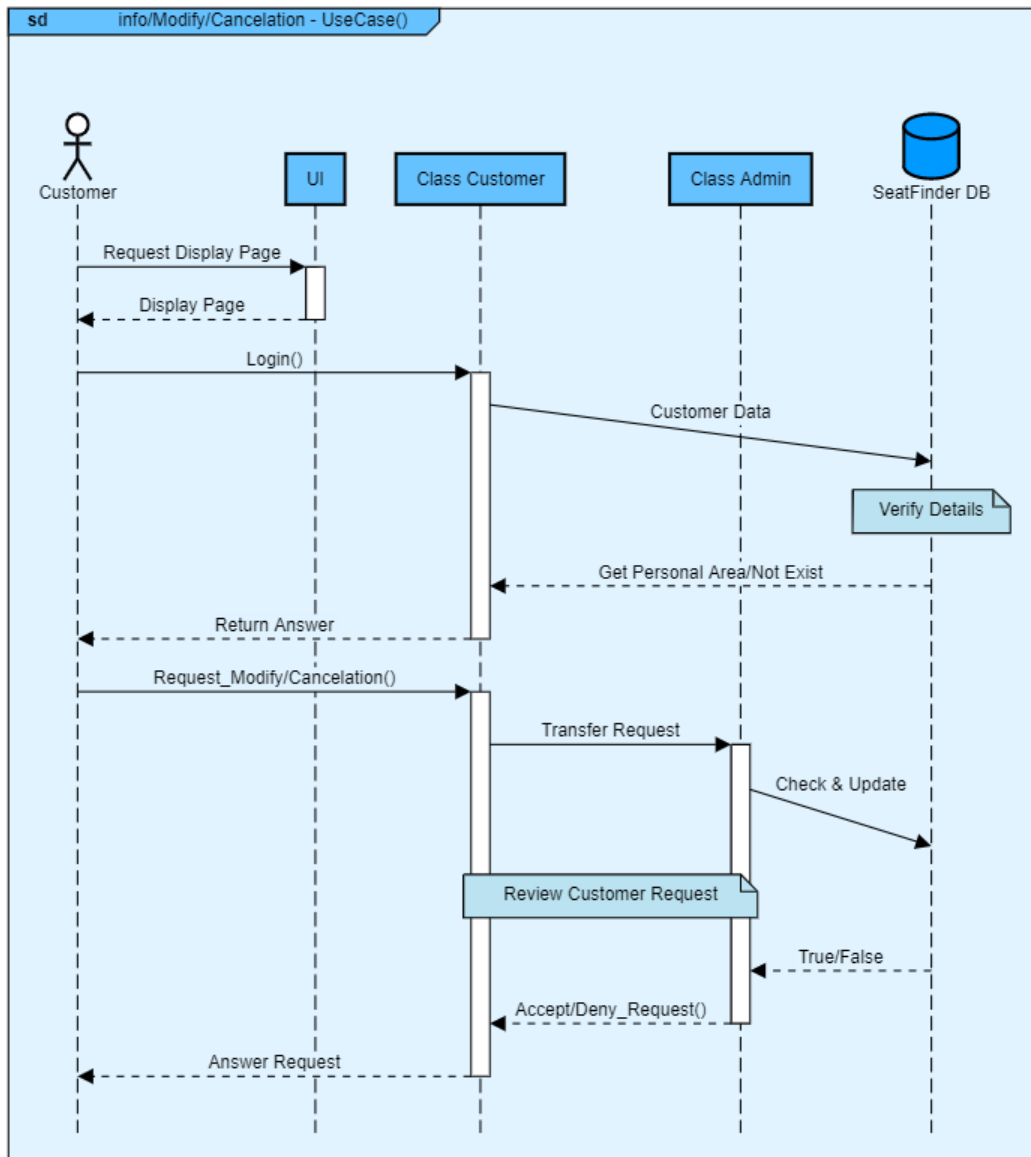

```
title view info/Modify/Cancelation

actor Customer

participant UI #66c2ff
participant Class Customer #66c2ff
participant Class Admin #66c2ff
frame #66c2ff                    info/Modify/Cancelation -
UseCase()
database SeatFinder DB #0099ff


Customer->UI:Request Display Page
activate UI
UI-->Customer:Display Page
deactivate UI

Customer->Class Customer:Login()
activate Class Customer
Class Customer->(3)SeatFinder DB:Customer Data
note over SeatFinder DB #BAE2F0:Verify Details
Class Customer<--SeatFinder DB:Get Personal Area/Not
Exist
Customer<--Class Customer:Return Answer
deactivate Class Customer


Customer->Class Customer:Request_Modify/Cancelation()
activate Class Customer
Class Customer->Class Admin:Transfer Request
activate Class Admin
Class Admin->(3)SeatFinder DB:Check & Update
note over Class Admin,Class Customer#BAE2F0: Review
Customer Request
Class Admin<--SeatFinder DB:True/False
Class Customer<--Class Admin:Accept/Deny_Request()
deactivate Class Admin
Customer<--Class Customer:Answer Request
deactivate Class Customer
```

Class Diagram

Ticket
- reservation: Reservation
- price: Double (decimal)
- seat: Seat
- ticket_id: Int
- status: string

+ getReservation() : Reservation
+ setReservation(reservation: Reservation) : void
+ getTicketNumber() : int
+ setTicketNumber(ticket_number: int) : void
+ getPrice() : decimal
+ setPrice(price: decimal) : void

Customer
- username: String
- password: String
- email: String
- phone: String
- reservations: List

+ getName() : string
+ setName(name: string) : void
+ getAge() : int
+ setAge(age: int) : void
+ getGender() : string
+ setGender(gender: string) : void
+ getEmail() : string
+ setEmail(email: string) : void
+ getPassword() : string
+ setPassword(password: string) : void
+ getReservationHistory() : list of Reservation
+ addReservation(reservation: Reservation) : void
+ deleteReservation(reservation: Reservation) : void

Movie
- title: String
- description: String
- duration: Int
- rating: Int

+ getTitle() : string
+ setTitle(title: string) : void
+ getDescription() : string
+ setDescription(description: string) : void
+ getDuration() : int
+ setDuration(duration: int) : void
+ getRating() : int
+ setRating(rating: int) : void

Showtime
- time: Datetime
- price: Double (decimal)
- movie: Movie
- theater: Theater

+ getTime() : datetime
+ setTime(time: datetime) : void
+ getPrice() : decimal
+ setPrice(price: decimal) : void
+ getMovie() : Movie
+ setMovie(movie: Movie) : void
+ getTheater() : Theater
+ setTheater(theater: Theater) : void

Reservation
- showtime: Showtime
- seats: List

+ getShowtime() : Showtime
+ setShowtime(showtime: Showtime) : void
+ getSeats() : list of Seat
+ setSeats(seats: list of Seat) : void

Payment
- reservation: Reservation
- amount: Double (decimal)
- date: Datetime

+ getReservation(): Reservation
+ setReservation(reservation: Reservation): void
+ getAmount(): decimal
+ setAmount(amount: decimal): void
+ getDate(): datetime
+ setDate(date: datetime): void

Admin
- username: String
- password: String
- email: String

+ getName() : string
+ setName(name: string) : void
+ getUsername() : string
+ setUsername(username: string) : void
+ getPassword() : string
+ setPassword(password: string) : void
+ addMovie(movie: Movie) : void
+ deleteMovie(movie: Movie) : void
+ addTheater(theater: Theater) : void
+ deleteTheater(theater: Theater) : void
+ addShowtime(showtime: Showtime) : void
+ deleteShowtime(showtime: Showtime) : void
+ getRevenue() : decimal

Theater
- name: String
- location: String
- seating_plan: Dictionary

+ getName() : string
+ setName(name: string) : void
+ getLocation() : string
+ setLocation(location: string) : void
+ getSeatingPlan() : dictionary
+ setSeatingPlan(seating_plan: dictionary) : void

Seat
- showtime: Showtime
- row: String
- number: Int
- status: String
- theater: Theater

+ getRow() : string
+ setRow(row: string) : void
+ getNumber() : int
+ setNumber(number: int) : void
+ getStatus() : string
+ setStatus(status: string) : void

MVC Architectural Diagram

Controller

Admin	Customer
Login/Logout	Login/Logout
Add/Remove Movie	Select Movie
Edit Movie Details	Select Seats
View Customer Request	Choose Screening Time
View/Export Reports	User Profile
Accept/Deny Request	Payment
Check Avalibility	Modify/Cancelation
Manage Accounts	Delete Account

User Actions

Uptades

Uptades

Conroller changes

View

SeatFinder UI

Dynamic Page

Send User Request

Model

SeatFinder DB

SeatFinder System