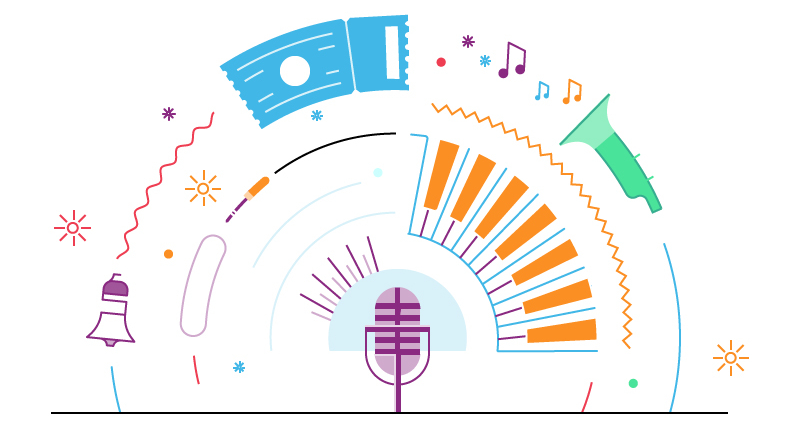
**Design a Movie Ticket Booking System**

**Let's design Movie Ticket Booking System**

**We'll cover the following:**

* [System Requirements](https://github.com/tssovi/grokking-the-object-oriented-design-interview/blob/master/object-oriented-design-case-studies/design-a-movie-ticket-booking-system.md#system-requirements)
* [Use Case Diagram](https://github.com/tssovi/grokking-the-object-oriented-design-interview/blob/master/object-oriented-design-case-studies/design-a-movie-ticket-booking-system.md#use-case-diagram)
* [Class Diagram](https://github.com/tssovi/grokking-the-object-oriented-design-interview/blob/master/object-oriented-design-case-studies/design-a-movie-ticket-booking-system.md#class-diagram)
* [Activity Diagrams](https://github.com/tssovi/grokking-the-object-oriented-design-interview/blob/master/object-oriented-design-case-studies/design-a-movie-ticket-booking-system.md#activity-diagrams)
* [Code](https://github.com/tssovi/grokking-the-object-oriented-design-interview/blob/master/object-oriented-design-case-studies/design-a-movie-ticket-booking-system.md#code)
* [Concurrency](https://github.com/tssovi/grokking-the-object-oriented-design-interview/blob/master/object-oriented-design-case-studies/design-a-movie-ticket-booking-system.md#concurrency)

An online movie ticket booking system facilitates the purchasing of movie tickets to its customers. E-ticketing systems allow customers to browse through movies currently playing and book seats, anywhere and anytime.

[](https://github.com/tssovi/grokking-the-object-oriented-design-interview/blob/master/media-files/movie-ticket-booking-system.png)  
Movie Ticket Booking System

**System Requirements**

Our ticket booking service should meet the following requirements:

1. It should be able to list the cities where affiliate cinemas are located.
2. Each cinema can have multiple halls and each hall can run one movie show at a time.
3. Each Movie will have multiple shows.
4. Customers should be able to search movies by their title, language, genre, release date, and city name.
5. Once the customer selects a movie, the service should display the cinemas running that movie and its available shows.
6. The customer should be able to select a show at a particular cinema and book their tickets.
7. The service should show the customer the seating arrangement of the cinema hall. The customer should be able to select multiple seats according to their preference.
8. The customer should be able to distinguish between available seats and booked ones.
9. The system should send notifications whenever there is a new movie, as well as when a booking is made or canceled.
10. Customers of our system should be able to pay with credit cards or cash.
11. The system should ensure that no two customers can reserve the same seat.
12. Customers should be able to add a discount coupon to their payment.

**Use Case Diagram**

We have five main Actors in our system:

* **Admin:** Responsible for adding new movies and their shows, canceling any movie or show, blocking/unblocking customers, etc.
* **FrontDeskOfficer:** Can book/cancel tickets.
* **Customer:** Can view movie schedules, book, and cancel tickets.
* **Guest:** All guests can search movies but to book seats they have to become a registered member.
* **System:** Mainly responsible for sending notifications for new movies, bookings, cancellations, etc.

Here are the top use cases of the Movie Ticket Booking System:

* **Search movies:** To search movies by title, genre, language, release date, and city name.
* **Create/Modify/View booking:** To book a movie show ticket, cancel it or view details about the show.
* **Make payment for booking:** To pay for the booking.
* **Add a coupon to the payment:** To add a discount coupon to the payment.
* **Assign Seat:** Customers will be shown a seat map to let them select seats for their booking.
* **Refund payment:** Upon cancellation, customers will be refunded the payment amount as long as the cancellation occurs within the allowed time frame.

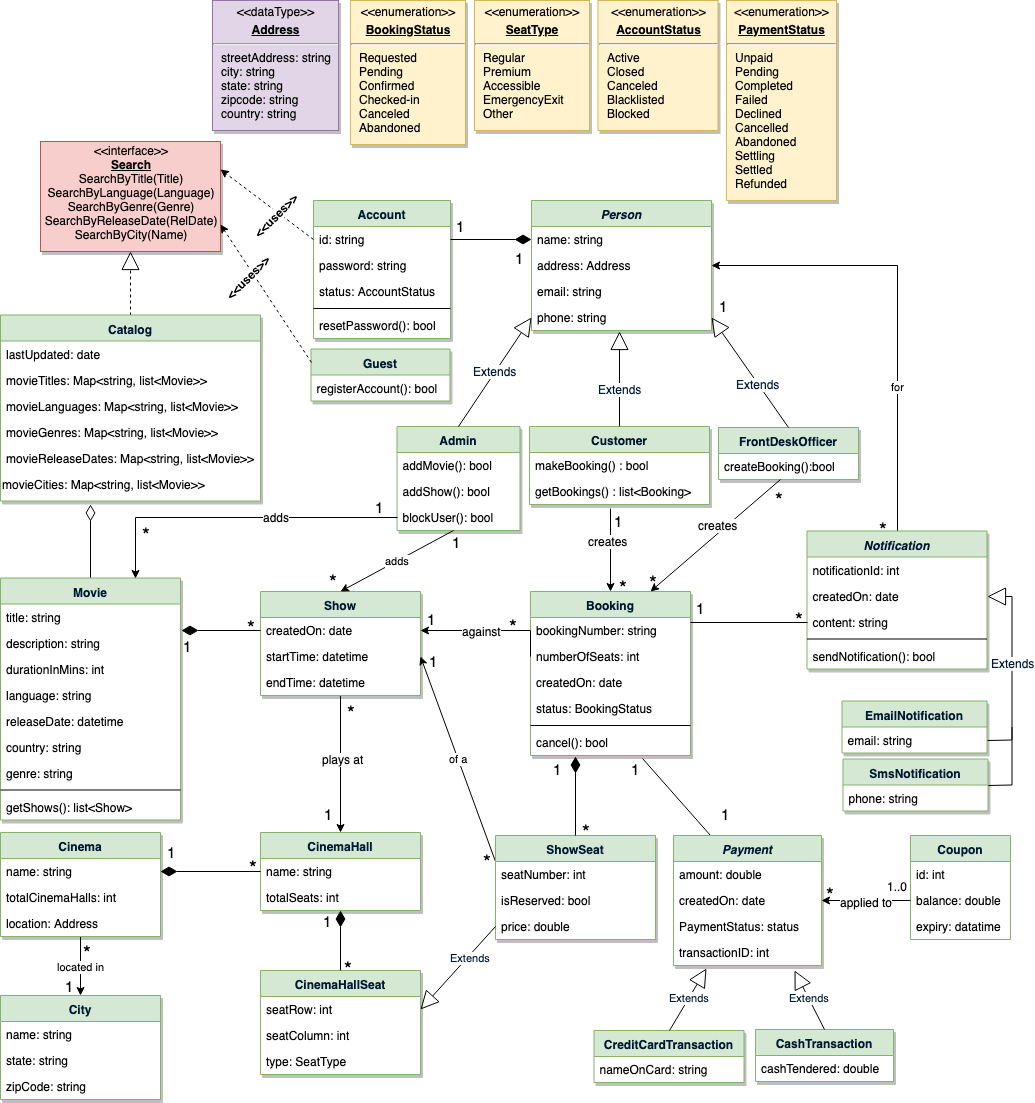
Here is the use case diagram of Movie Ticket Booking System:

Use Case Diagram for Movie Ticket Booking System

**Class Diagram**

Here are the main classes of the Movie Ticket Booking System:

* **Account:** Admin will be able to add/remove movies and shows, as well as block/unblock accounts. Customers can search for movies and make bookings for shows. FrontDeskOffice can book tickets for movie shows.
* **Guest:** Guests can search and view movies descriptions. To make a booking for a show they have to become a registered member.
* **Cinema:** The main part of the organization for which this software has been designed. It has attributes like ‘name’ to distinguish it from other cinemas.
* **CinemaHall:** Each cinema will have multiple halls containing multiple seats.
* **City:** Each city can have multiple cinemas.
* **Movie:** The main entity of the system. Movies have attributes like title, description, language, genre, release date, city name, etc.
* **Show:** Each movie can have many shows; each show will be played in a cinema hall.
* **CinemaHallSeat:** Each cinema hall will have many seats.
* **ShowSeat:** Each ShowSeat will correspond to a movie Show and a CinemaHallSeat. Customers will make a booking against a ShowSeat.
* **Booking:** A booking is against a movie show and has attributes like a unique booking number, number of seats, and status.
* **Payment:** Responsible for collecting payments from customers.
* **Notification:** Will take care of sending notifications to customers.

[](https://github.com/tssovi/grokking-the-object-oriented-design-interview/blob/master/media-files/mtbs-class-diagram.png)  
Class Diagram for Movie Ticket Booking System

UML for Movie Ticket Booking System

**Activity Diagrams**

**Make a booking:** Any customer can perform this activity. Here are the steps to book a ticket for a show:

Activity Diagram for Movie Ticket Booking System Make Booking

**Cancel a booking:** Customer can cancel their bookings. Here are the steps to cancel a booking:

Activity Diagram for Movie Ticket Booking System Cancel Booking