

# **Software Design Specification (SDS)**

**Software Title:** Movie Ticketing System

**Team Members:** Ben Coon, Lucine Babikian, Raul Montes

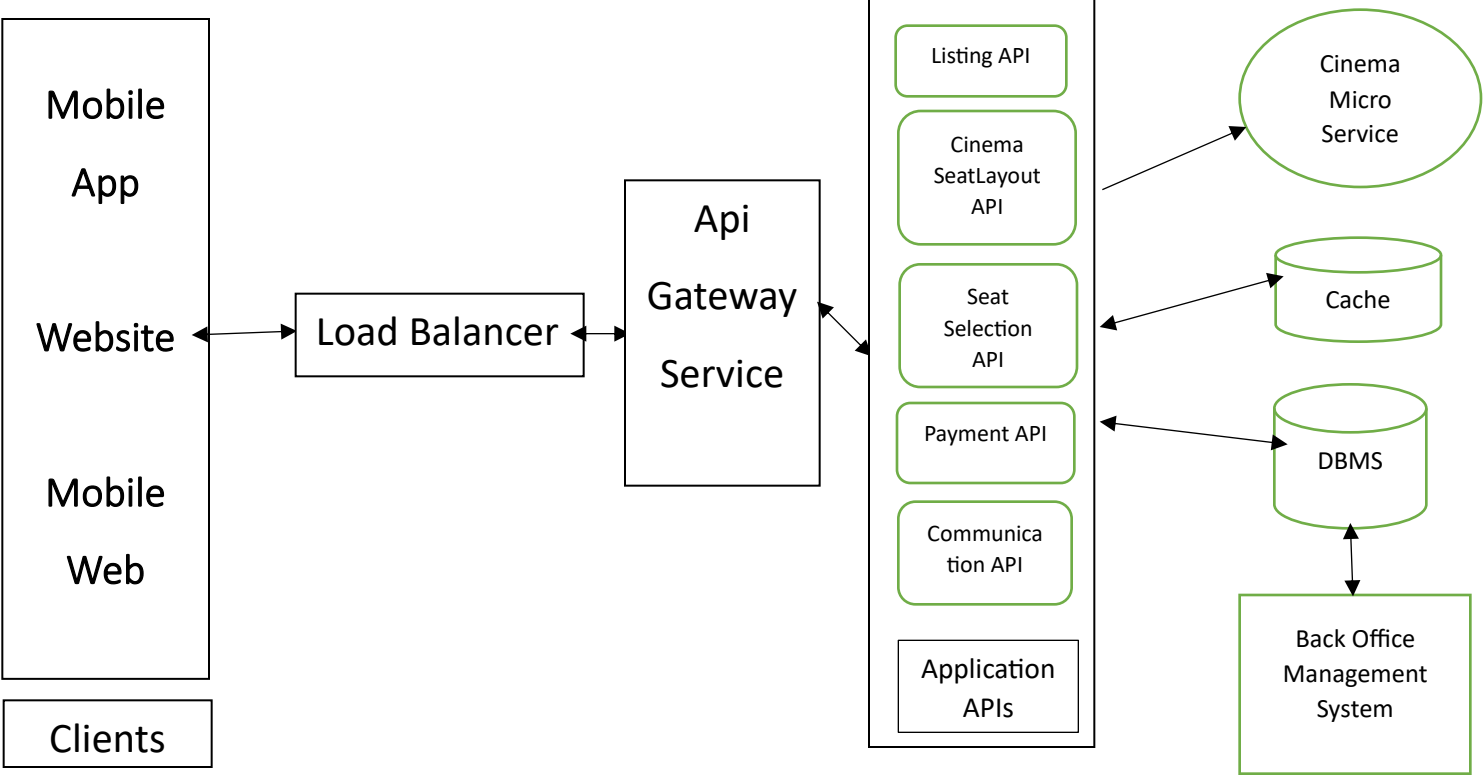
## **System Description**

### **Brief Overview of The System**

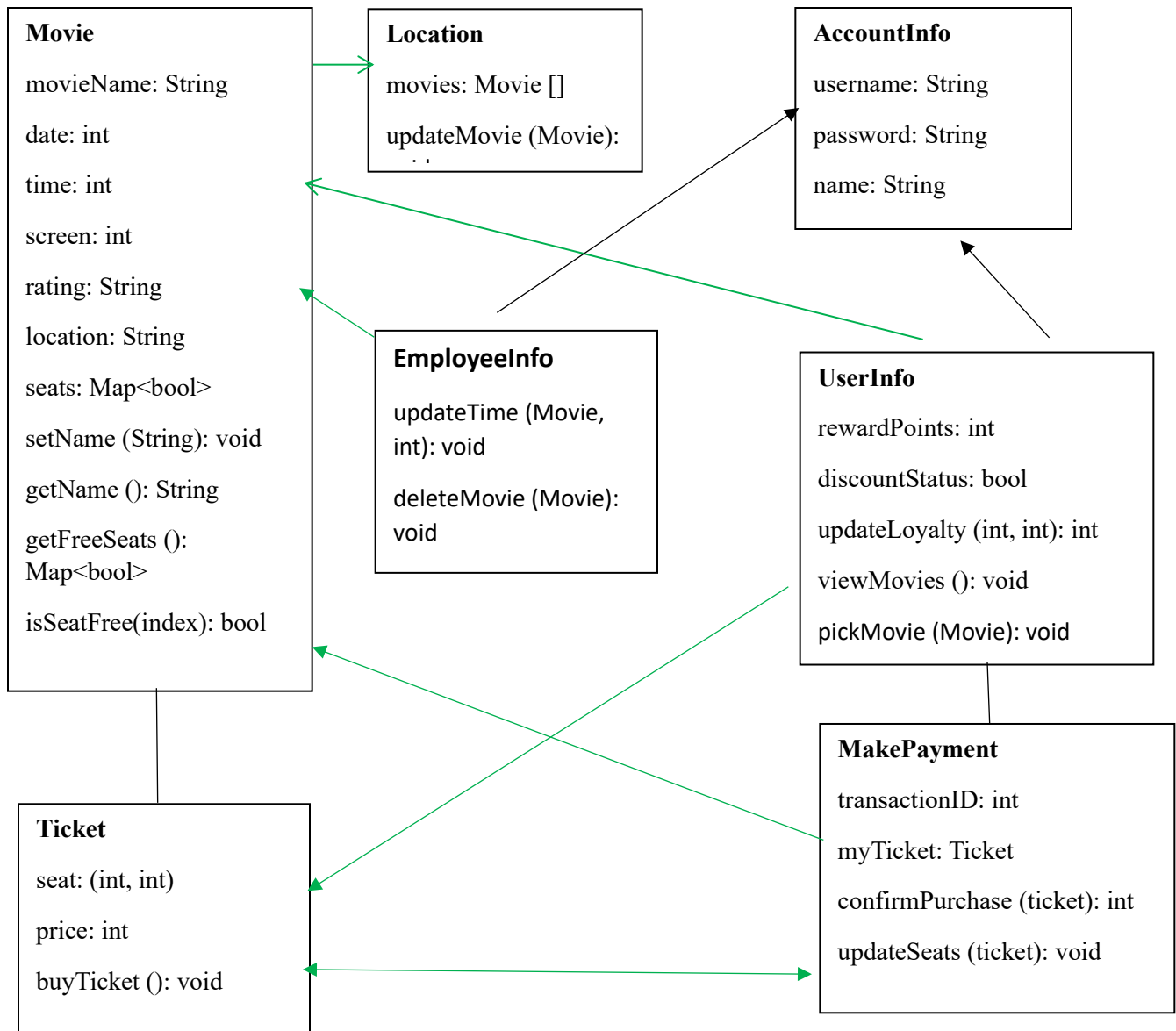
The Movie Ticketing System is designed to facilitate the online booking of movie tickets. The system allows users to browse movie listings, select showtimes, choose seats, and make payments securely. It also provides functionalities for theater administrators to manage movie schedules, seat layouts, and ticket pricing.

### **Software Architecture Overview**

### **Architectural Diagram**



UML Class Diagram



## Description of Classes

### Class Name: User

**Description:** Handles user-related operations such as registration, login, and profile management.

### Attributes:

- ✓ `userId: String` - Unique identifier for the user
- ✓ `username: String` - Username of the user
- ✓ `password: String` - Password for user authentication
- ✓ `email: String` - Email address of the user
- ✓ `phoneNumber: String` - Contact number of the user

#### **Operations:**

- ✓ `register (username: String, password: String, email: String, phoneNumber: String): Boolean`  
Registers a new user and returns true if successful.
- ✓ `Login (username: String, password: String): Boolean`  
Authenticates a user and returns true if credentials are valid.
- ✓ `updateProfile (email: String, phoneNumber: String): Boolean`  
Updates the user's profile information.

#### **Class Name: Movie**

**Description:** Represents a movie with all relevant details.

#### **Attributes:**

- ✓ `movieId: String` - Unique identifier for the movie
- ✓ `title: String` - Title of the movie
- ✓ `genre: String` - Genre of the movie
- ✓ `duration: int` - Duration of the movie in minutes
- ✓ `rating: float` - Average rating of the movie

#### **Operations:**

- ✓ `getDetails (movieId: String): Movie`  
Fetches the details of a movie based on the movieId.

- ✓ updateRating (movieId: String, rating: float): Boolean

Updates the rating of the movie.

### **Class Name: Show**

#### **Description:**

Represents a specific showtime for a movie.

#### **Attributes:**

- ✓ showId: String - Unique identifier for the show
- ✓ movieId: String - Identifier of the movie being shown
- ✓ theaterId: String - Identifier of the theater where the show is taking place
- ✓ showtime: DateTime - Date and time of the show
- ✓ availableSeats: int - Number of available seats for the show

#### **Operations:**

- ✓ getAvailableSeats (showId: String): int

Returns the number of available seats for a particular show.

Books a seat for a particular show and updates the available seats count.

### **Class Name: Theater**

**Description:** Represents a movie theater, including its details and the shows it offers.

#### **Attributes:**

- ✓ theaterId: String - Unique identifier for the theater
- ✓ name: String - Name of the theater

- ✓ location: String - Location of the theater
- ✓ totalSeats: int - Total number of seats in the theater

### **Operations:**

- ✓ getShows (theaterId: String): List<Show>  
Returns a list of shows available at the theater.
- ✓ updateSeatLayout (theaterId: String, totalSeats: int): Boolean  
Updates the seat layout of the theater.
- ✓ Class Name: Booking  
Description: Handles the booking of movie tickets.

### **Attributes:**

- ✓ bookingId: String - Unique identifier for the booking
- ✓ userId: String - Identifier of the user making the booking
- ✓ showId: String - Identifier of the show being booked
- ✓ seatsBooked: int - Number of seats booked
- ✓ totalPrice: float - Total price of the booking
- ✓ bookingStatus: String - Status of the booking (e.g., confirmed, pending, canceled)

### **Operations:**

- ✓ createBooking (userId: String, showId: String, seatsBooked: int): Booking  
Creates a new booking and returns the booking details.
- ✓ cancelBooking (bookingId: String): Boolean  
Cancels an existing booking and updates the booking status.

### **Class Name: Payment**

### **Description:**

Handles payment transactions for bookings.

**Attributes:**

- ✓ paymentId: String - Unique identifier for the payment
- ✓ bookingId: String - Identifier of the associated booking
- ✓ amount: float - Amount to be paid
- ✓ paymentMethod: String - Method of payment (e.g., credit card, debit card, PayPal)
- ✓ paymentStatus: String - Status of the payment (e.g., successful, failed, pending)

**Operations:**

- ✓ processPayment (bookingId: String, amount: float, paymentMethod: String): Boolean  
Processes the payment for a booking and returns true if the payment is successful.

## Development Plan and Timeline

### Partitioning of Tasks

The development tasks will be divided based on the main components of the system. Each team member will be responsible for specific components to ensure focused and efficient development.

Team Member	Responsibilities
Ben coon	User Management, Authentication, Seat Layout
Lucine Babikian	Movie and Show Management, Theater Management
Raul Montes	Booking System, Payment Processing

## Timeline

Milestone	Deadline
Requirement Specification	May 27 <sup>th</sup> 2024
Initial Design Draft	June 3 <sup>rd</sup> 2024
Final Design Submission	June 10 <sup>th</sup> 2024
Implementation Start	June 17 <sup>th</sup> 2024
Midpoint Review	June 21 <sup>st</sup> 2024
Testing Phase	June 24 <sup>th</sup> 2024
Final Review and Delivery	June 28 <sup>th</sup> 2024