

**Software Requirements Specification**  
**Version 1.0**  
Annotated Version

08-03-2025

**Students' Auditorium Management  
System**



Aryan Vasudeva  
Lek Nath Ghalley  
Sumit Kumar  
Ananta Devraj Roul

## Table of Contents

Table of Contents.....	i
List of Figures.....	ii
1.0. Introduction.....	1
1.1. Purpose.....	2
1.2. Scope of Project.....	2
1.3. Glossary.....	3
1.4. Functional Requirements.....	3
1.5. Non-functional Requirements.....	4
1.6. References.....	5
1.7. Overview of Document.....	5
2.0.	
2.1. Overall Description.....	4
2.1 System Environment.....	4
2.2 Functional Requirements Specification.....	5
2.2.1 Reader Use Case.....	5
Use case: Search Article.....	5
2.2.2 Author Use Case.....	6
Use case: Submit Article.....	6
2.2.3 Reviewer Use Case.....	7
Use case: Submit Review.....	7
2.2.4 Editor Use Cases.....	8
Use case: Update Author.....	8
Use case: Update Reviewer.....	9
Use case: Update Article.....	9
Use case: Receive Article.....	10
Use case: Assign Reviewer.....	10
Use case: Receive Review.....	11
Use case: Check Status.....	12
Use case: Send Response.....	12
Use case: Send Copyright.....	13
Use case: Remove Article.....	14
Use case: Publish Article.....	14
2.3 User Characteristics.....	15
2.4 Non-Functional Requirements.....	15
3.0. Requirements Specification.....	17
3.1 External Interface Requirements.....	17
3.2 Functional Requirements.....	17
3.2.1 Search Article.....	17
3.2.2 Communicate.....	18
3.2.3 Add Author.....	18
3.2.4 Add Reviewer.....	19
3.2.5 Update Person.....	19
3.2.6 Update Article Status.....	20
3.2.7 Enter Communication.....	20
3.2.8 Assign Reviewer.....	21
3.2.9 Check Status.....	21
3.2.10 Send Communication.....	22
3.2.11 Publish Article.....	22
3.2.12 Remove Article.....	23
3.3 Detailed Non-Functional Requirements.....	23
3.3.1 Logical Structure of the Data.....	23
3.3.2 Security.....	25

Index..... 26

## **List of Figures**

Figure 1 - System Environment	4
Figure 2 - Article Submission Process	6
Figure 3 - Editor Use Cases	8
Figure 4 - Logical Structure of the Article Manager Data	23

## 1. Introduction

Students' Auditorium Management Software. Various types of social and cultural events are conducted in the students' auditorium. There are two categories of seats: balcony seats and ordinary seats.

Normally balcony seats are more expensive in any show. The show manager fixes the price of these two categories of seats depending on the popularity of a show. The show manager also determines the number of balcony and ordinary seats that can be put on sale, since for each show some seats are offered as complimentary gifts to different functionaries of the students' society and to VIPs. The show manager also enters the show dates, the number of shows on any particular date and the show timings.

The spectators book their seats in advance by paying the full ticket price to the authorized sales persons.

The spectators indicate the type of the seat and the computer should print out the ticket clearly showing the seat numbers. The spectators can cancel their booking before 3 clear days of the show. In this case the ticket price is refunded to them after deducting Rs.5/- as the booking charge per ticket. If a ticket is returned within 3 days and 1 day of a show, a booking charge of Rs.10/- is deducted for ordinary tickets and Rs.15/- is deducted for balcony tickets. On the last day of the show, there is a 50% deduction. The system should let the spectators query the availability of different classes of seats.

The show manager can query any time about the percentage of seats booked for various classes of seats and the amount collected in each case. The show manager creates login accounts for authorized sales persons.

When any authorized sales person logs in and makes a sale, the computer should record the sales person's id in the sales transaction. This information would help in computing the commission payable to each sales person and also the amount collected by each sales person. These data can be queried by the show manager.

## ***1.1 Purpose***

The purpose of a **Students' Auditorium Management System** is to efficiently manage the booking, scheduling, and utilization of an auditorium or event space, specifically for student-related activities.

## ***1.2 Scope of Project***

The **Student Auditorium Management System** is a web-based solution that simplifies the process of managing and reserving auditorium spaces in educational institutions. Here's some functions of this product:

1. **Easy Auditorium Booking**  
Students and faculty can easily book the auditorium for events, classes, meetings, or extracurricular activities through an intuitive platform.
2. **Real-Time Availability Check**  
Users can quickly check the availability of the auditorium, making it simple to plan events without the hassle of back-and-forth communication.
3. **Reservation Requests**  
Users can submit booking requests with details like event type, time, and required resources (e.g., projectors, microphones, or seating arrangements).
4. **Administrator Oversight**  
Administrators have full control to manage all reservations, approve or decline requests, and ensure smooth scheduling, avoiding conflicts and double bookings.
5. **Resource Allocation**  
The system ensures that all necessary resources (like tech equipment or seating arrangements) are properly allocated and managed, helping events run smoothly.
6. **Notifications and Reminders**  
Automated notifications keep users informed about the status of their reservations, any approvals, and upcoming events. Users will receive reminders to ensure no details are missed.
7. **Efficient Communication**  
The system provides a communication channel between students, faculty, and administrators, making it easier to resolve any issues or clarify details regarding bookings and resources.
8. **Centralized Database**  
A relational database tracks users, bookings, events, and resources, making it easy for administrators to access and manage information. This also supports detailed reporting.
9. **Reduced Administrative Workload**  
By automating the booking and management process, the system reduces the administrative burden, saving time and effort for staff while ensuring smooth operations.
10. **Improved Transparency and User Experience**  
The system provides a clear, transparent view of the booking process, making it easier for users to track and manage their reservations and improving their overall experience.

### ***1.3 Glossary***

Term	Definition
<b>Auditorium Management System</b>	Software designed to manage bookings, events, and financial transactions within an auditorium.
<b>User</b>	Any individual interacting with the system, including spectators, salespersons, clerks, and show managers.
<b>Spectator</b>	A person who books a seat to attend an event in the auditorium.
<b>Salesperson</b>	An authorized individual responsible for booking and canceling tickets on behalf of spectators.
<b>Clerk</b>	A system user who manages balance sheets and financial transactions related to the auditorium.
<b>Show Manager</b>	The individual responsible for managing events, setting seat pricing, and overseeing auditorium operations.
<b>Event</b>	A scheduled activity or performance held in the auditorium, available for booking.
<b>Seat</b>	A specific place in the auditorium categorized as either a balcony or ordinary seat.
<b>Booking</b>	The process of reserving a seat for an event through the system.
<b>Cancellation</b>	The process of revoking a booking with applicable refund policies.
<b>Balance Sheet</b>	A financial document that tracks revenue, expenses, and net income for an event.
<b>Payment Gateway</b>	An external service that processes online transactions securely.
<b>Refund</b>	The amount returned to a spectator upon canceling a ticket as per the cancellation policy.
<b>Transaction</b>	A financial action recorded in the system, such as a booking payment or refund.
<b>Authentication</b>	A security process that verifies a user's identity before granting access to the system.

## ***1.4 Functional Requirements***

### **1. Query Availability of Seats**

- Input: Event selection.
- Process: System checks the availability of seats for the selected event.
- Output: Displays available seats or a message if fully booked.

### **2. Book New Seat**

- Input: Event selection, seat selection, customer information.
- Process: System verifies seat availability. If available, books the seat and stores customer details for notification.
- Output: Confirmation of booking or message indicating unavailability.

### **3. Cancel Booking**

- Input: Event selection, seat selection.
- Process: System verifies booking and cancels it if valid.
- Output: Confirmation of cancellation and updated seat availability.

### **4. Prepare Balance Sheet**

- Input: Event selection (for new or existing balance sheet).
- Process: System generates or updates balance sheet based on event transactions.
- Output: Display or download balance sheet.

### **5. Add New Event**

- Input: Event details (date, time slot, guest list, duration, venue, etc.).
- Process: System creates and stores event details.
- Output: Confirmation of event creation.

### **6. Edit Event**

- Input: Event selection, updated details.
- Process: System modifies event details.
- Output: Confirmation of update.

### **7. Check Event Status**

- Input: Event selection.
- Process: System retrieves event status details.
- Output: Displays event status (booked seats, event progress, etc.).

### **8. Create New Personnel**

- Input: Personnel type (Clerk or Sales Person), personnel details.
- Process: System stores personnel details and assigns access.
- Output: Confirmation of personnel creation.

### **9. View Transaction Details**

- Input: Selection of specific salesperson or all transactions.
- Process: System retrieves transaction history.
- Output: Displays transaction details.

### **10. View Balance Sheet**

- Input: Event selection or full-year selection.
- Process: System retrieves balance sheet data.
- Output: Displays balance sheet.

## ***1.5 Non-Functional Requirements***

### **1. Performance Requirements**

- The system should process ticket bookings and cancellations within 2 seconds.
- The system should handle up to 1000 concurrent users without performance degradation.
- The response time for retrieving show details should not exceed 1 second.

### **2. Reliability & Availability**

- The system should be available 99.9% of the time to ensure uninterrupted booking and



- cancellation services.
  - It should have a backup mechanism to prevent data loss in case of a system failure.
  - The system should recover from failures within 5 minutes.
- 3. Usability Requirements**
  - The interface should be user-friendly with clear instructions for booking, cancellation, and querying seat availability.
  - The system should be accessible on both desktop and mobile devices.
  - It should support multi-language options for diverse users.
- 4. Security Requirements**
  - Only authorized salespersons should be allowed to book or cancel tickets.
  - The system should use role-based authentication to restrict access to administrative functions.
  - Payment transactions should be secured using SSL encryption.
  - The system should log all booking and cancellation activities for audit purposes.
- 5. Scalability Requirements**
  - The system should be designed to handle an increasing number of users as the college expands.
  - It should support multiple auditorium events simultaneously.
- 6. Maintainability & Extensibility**
  - The system should allow easy updates to ticket prices, show schedules, and seat categories.
  - The software should be modular, allowing future enhancements like online payment integration.
- 7. Data Integrity & Consistency**
  - The system should ensure accurate seat availability updates to prevent double bookings.
  - Refund transactions should be recorded and verified to prevent fraudulent claims.

## Data Dictionary

**User Table:**

Attribute Name	Data Type	Description	Constraints
User_ID	Integer	Unique identifier for users	Primary Key, Auto-increment
Username	Varchar(50)	Login username	Unique, Not Null
Password	Varchar(255)	Encrypted password	Not Null
Role	Enum	Defines user role	Not Null

**Event Table:**

Attribute Name	Data Type	Description	Constraints
Event_ID	Integer	Unique identifier for events	Primary Key, Auto-increment
Event_Name	Varchar(100)	Name of the event	Not Null
Event_Date	Date	Scheduled date	Not Null
Venue	Varchar(100)	Location of event	Not Null

### Booking Table:

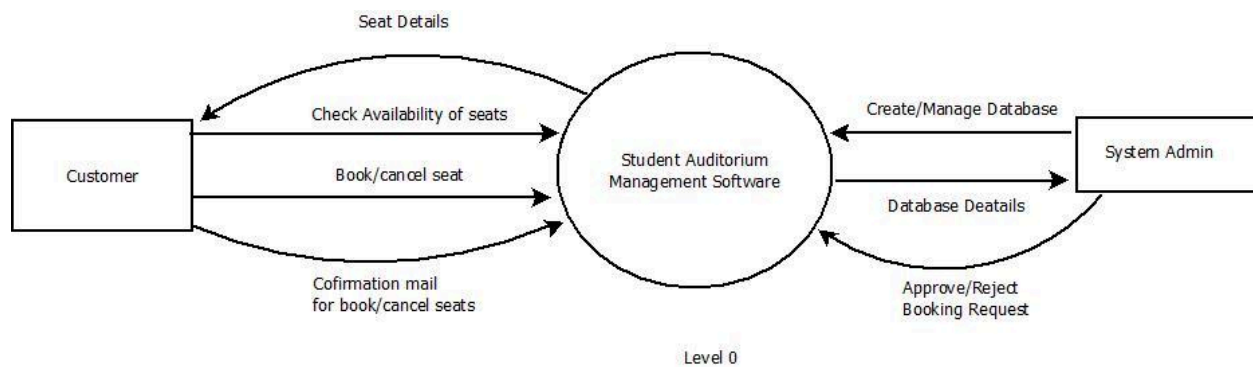
Attribute Name	Data Type	Description	Constraints
Booking_ID	Integer	Unique identifier for a booking	Primary Key, Auto-increment
User_ID	Integer	User who made the booking	Foreign Key (User Table)
Event_ID	Integer	Booked event	Foreign Key (Event Table)
Seat_ID	Integer	Booked seat	Foreign Key (Seat Table)

### Payment Table:

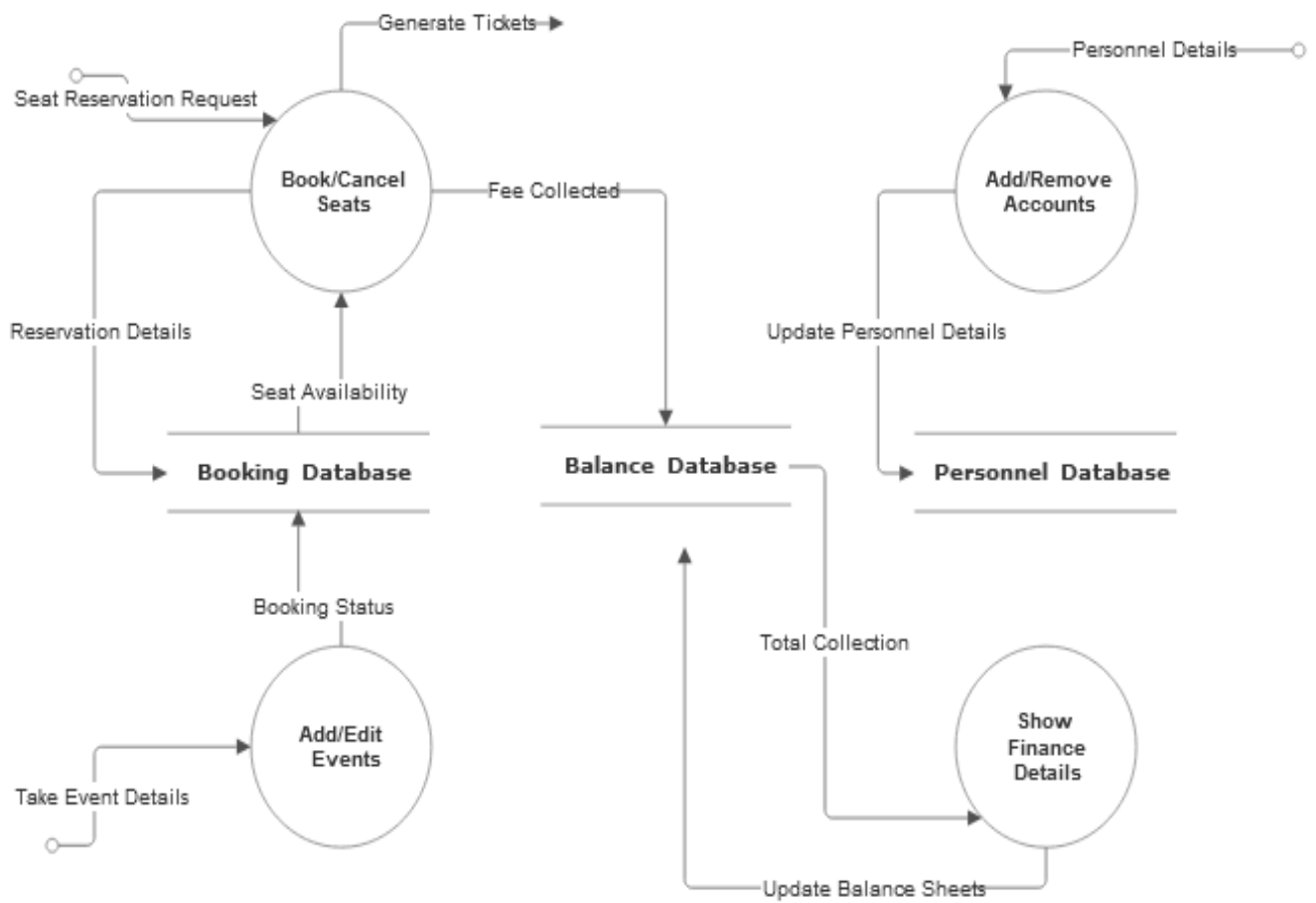
Attribute Name	Data Type	Description	Constraints
Payment_ID	Integer	Unique identifier for payments	Primary Key, Auto-increment
Booking_ID	Integer	Related booking	Foreign Key (Booking Table)
Amount	Decimal(10,2)	Total payment	Not Null
Payment_Status	Enum	Payment status	Default 'Pending'

## DFD(Data Flow Diagrams)

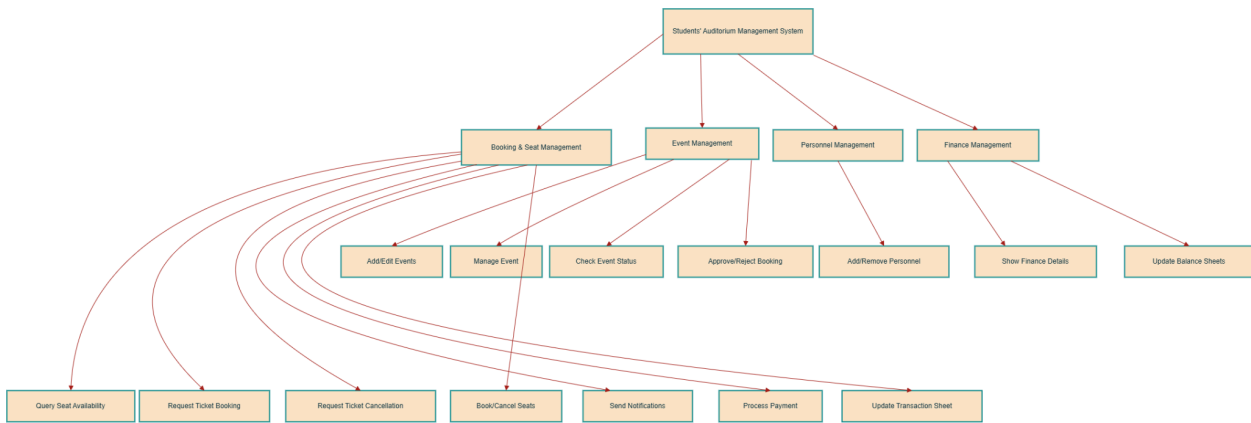
### Level 0:



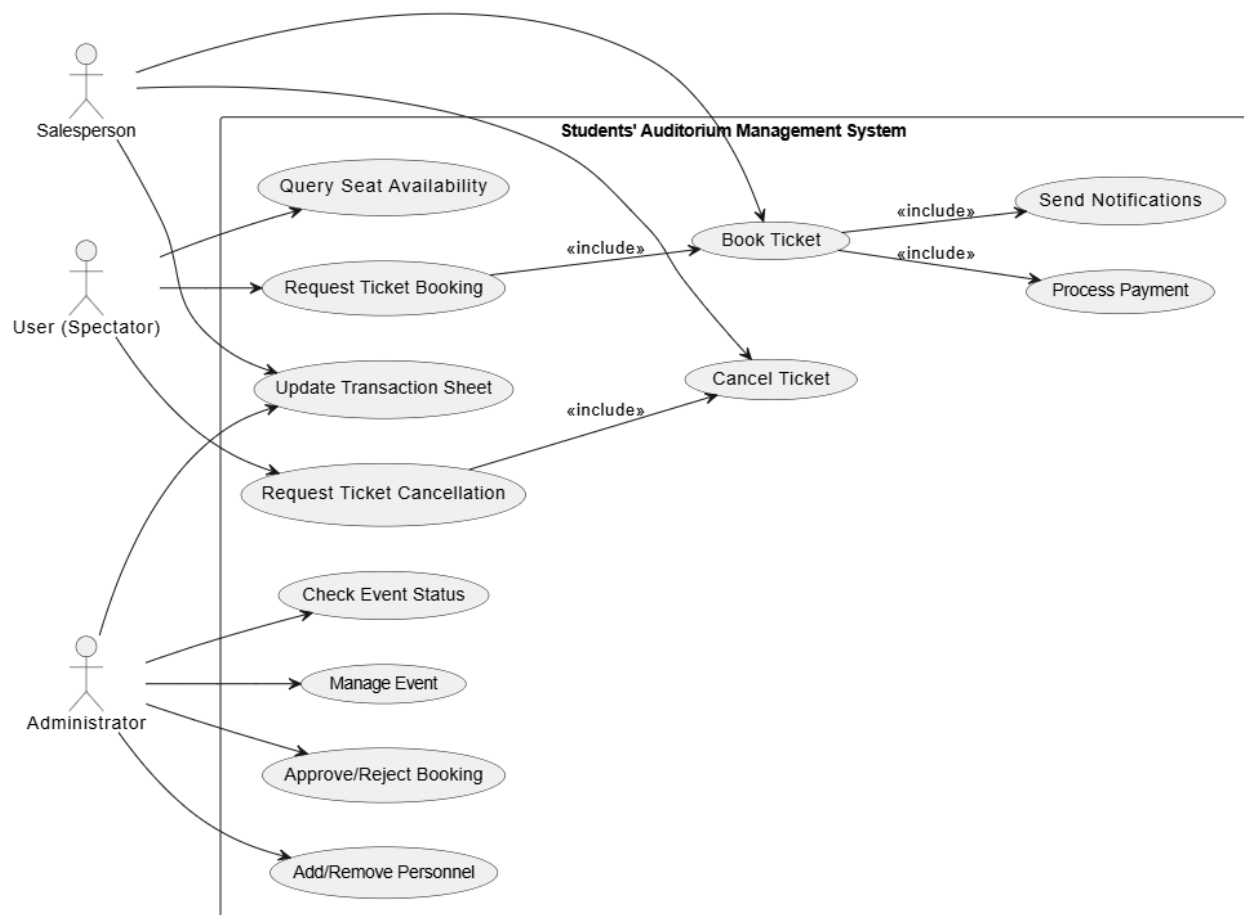
**Level 1:**

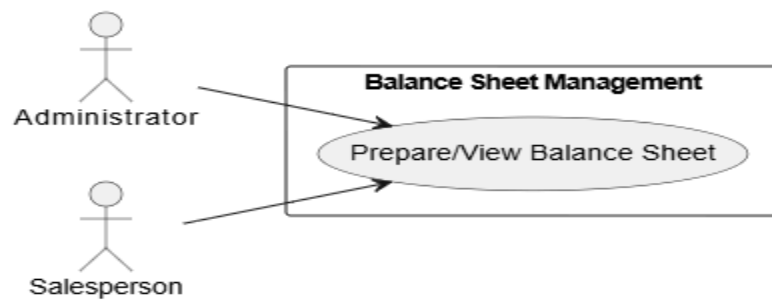
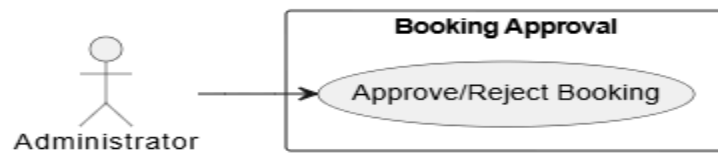
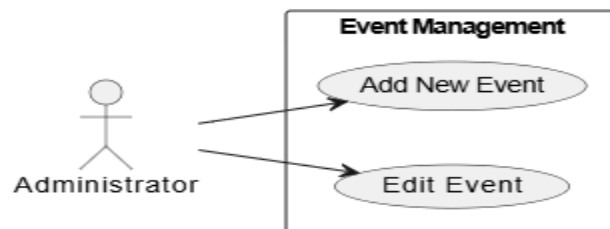
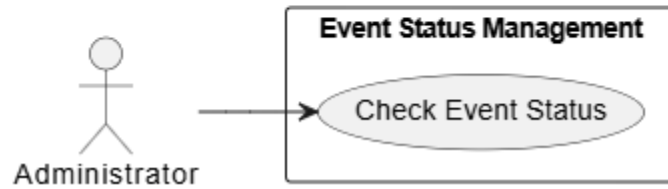


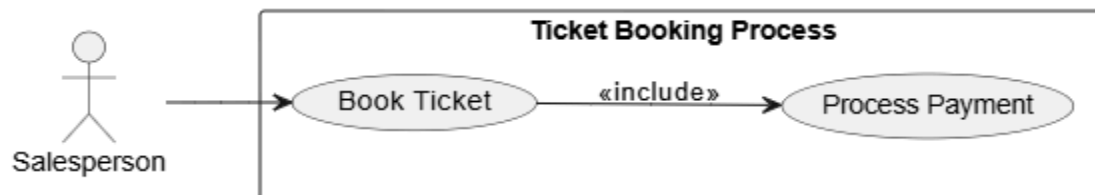
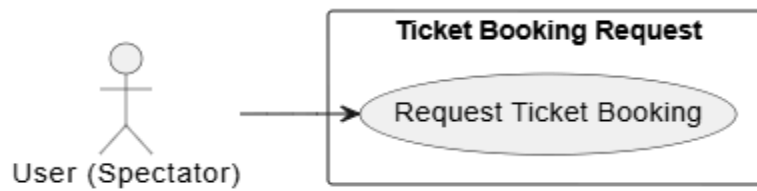
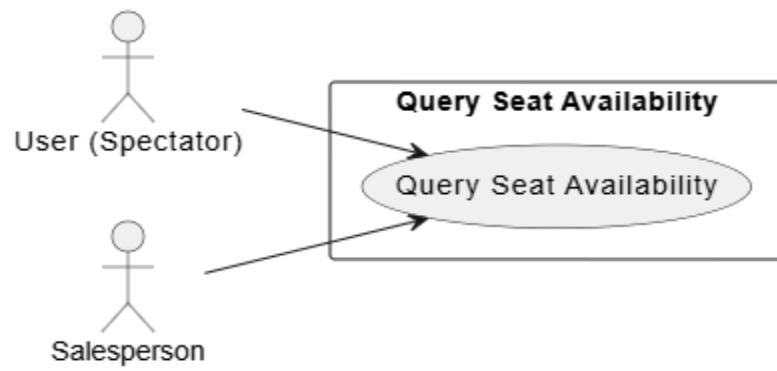
Structure Chart:

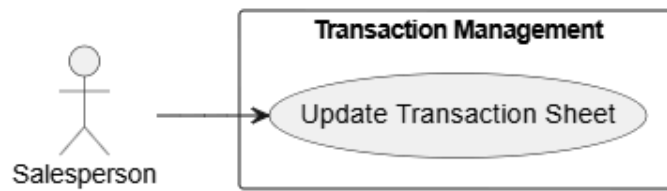
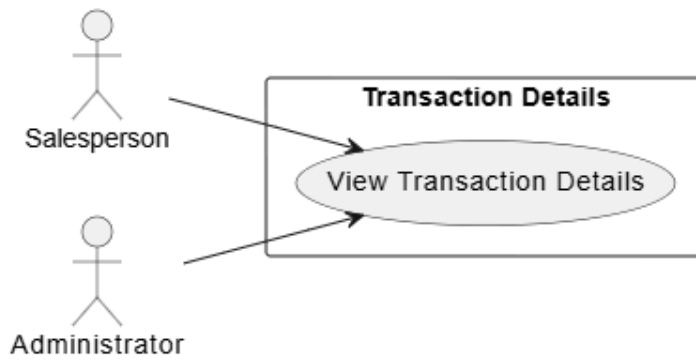
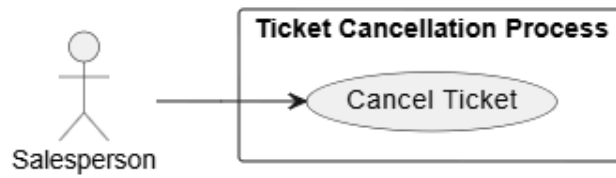
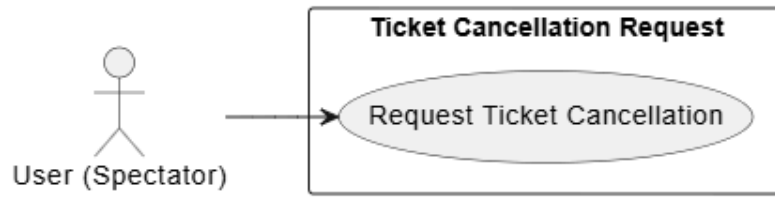


## Use Case Diagrams :



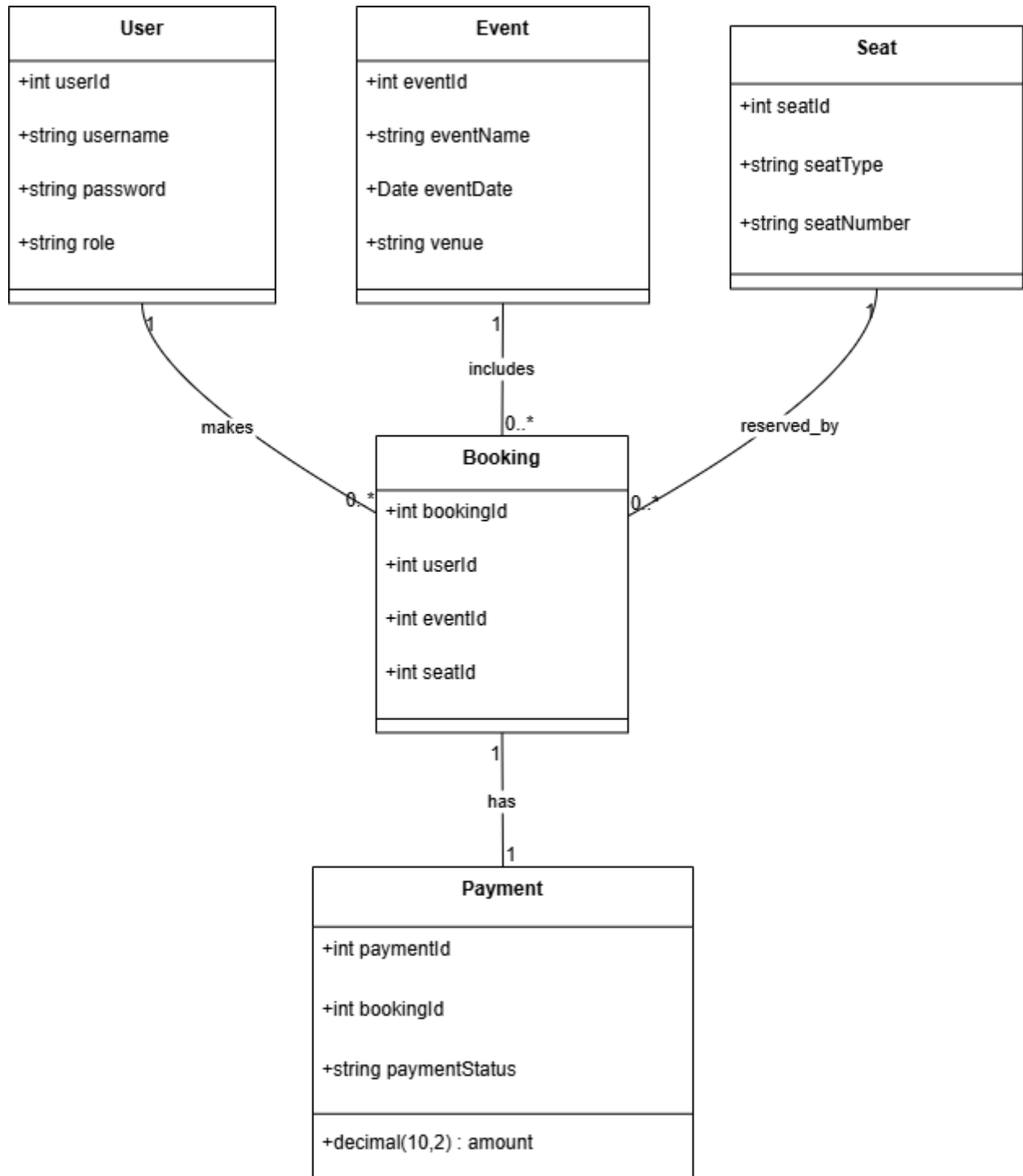




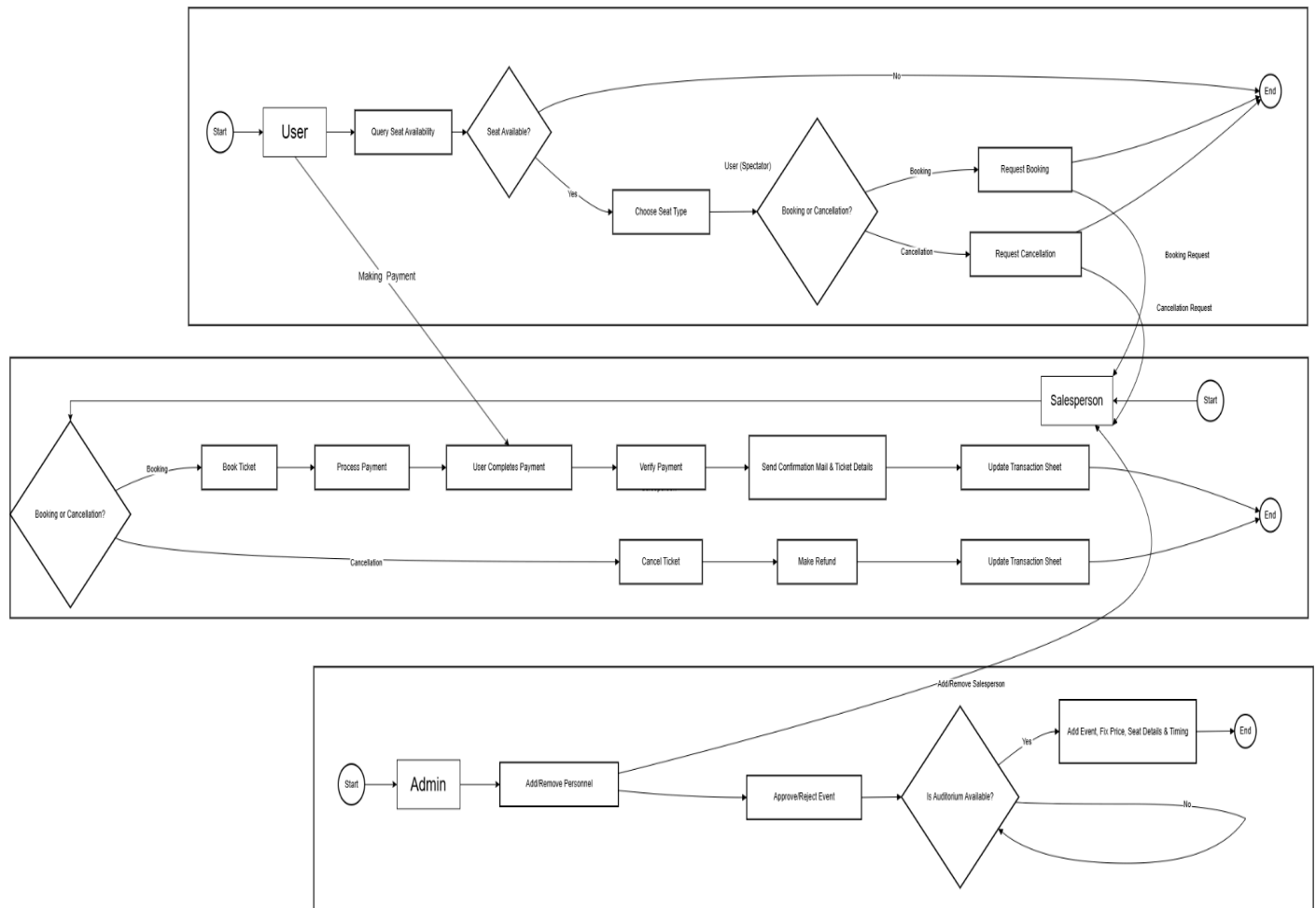




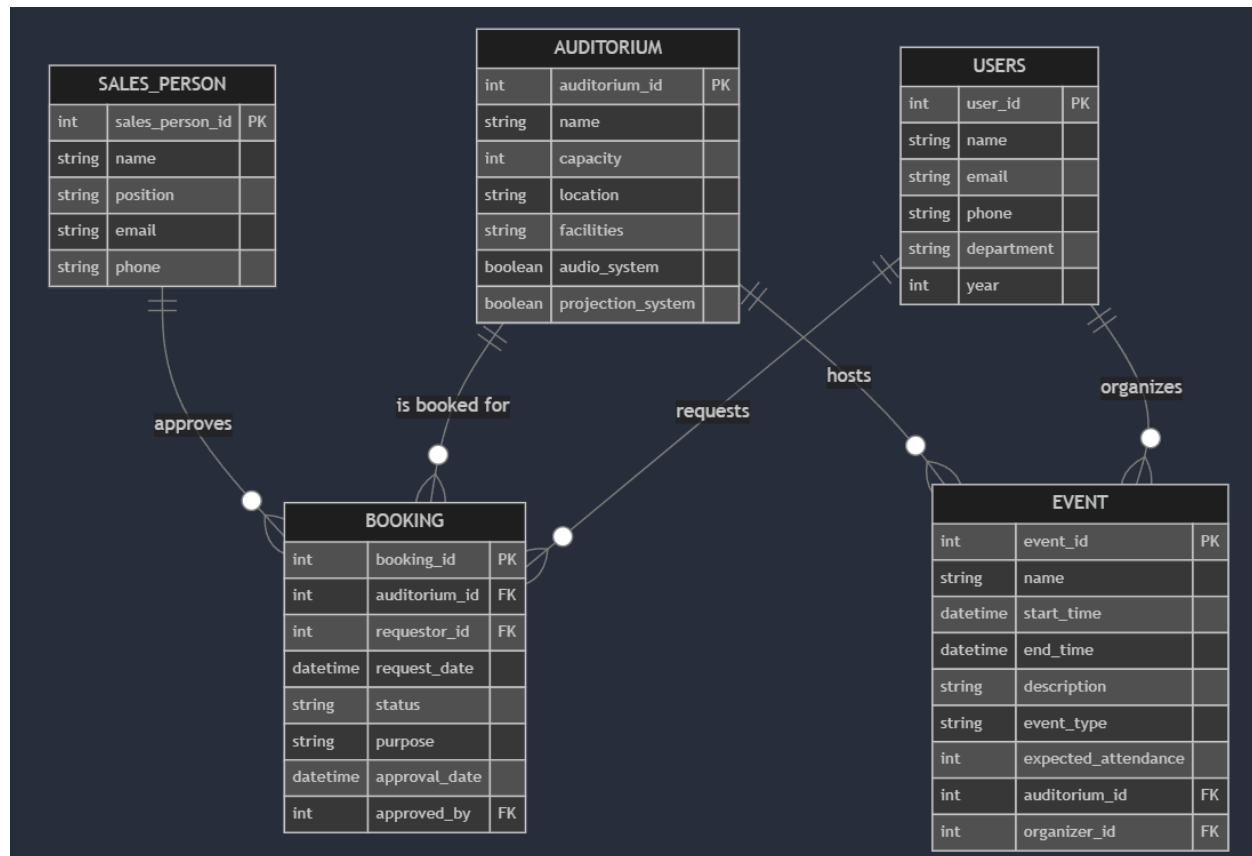
## Class Diagram:



## Activity Diagram :



## ER Diagram:



### 1.6 References

1. Fundamentals Of Software Engineering - Rajib Mall
2. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.
3. SE Lecture SASD (Provided by Prof. Durga Prashad Mohapatra)

### 1.7 Overview of Document

The system allows students and faculty to reserve the auditorium for academic or extracurricular activities, avoiding conflicts with a real-time availability calendar. Students can request space for events like club meetings, presentations, or workshops, while faculty can reserve it for classes or seminars. Administrators have the highest access level, managing user accounts, approving or denying bookings, and allocating resources such as projectors, microphones, or seating arrangements. The system enhances transparency by allowing all users to view upcoming bookings, preventing scheduling overlaps.