

LAB 1 REPORT: REQUIREMENTS ELICITATION & MODELING

Project Name: Movie Ticket Booking System (ShopSphere Context)

Member Name: Nguyễn Đình Quyền - 23010485

Trần Hữu Kiên - 23010258

1. REQUIREMENTS ELICITATION

This section documents the functional, non-functional, and architecturally significant requirements (ASRs) derived from the project analysis¹.

1.1.

Actors²

Based on the system scenario, the following primary entities interact with the system:

- **Guest:** A visitor who has not logged in. Can search for movies and view showtimes.
- **Member (Customer):** A registered user who can perform transactions, such as booking tickets and viewing booking history.
- **Administrator (Admin):** The system manager responsible for managing movies, schedules, and maintaining the system.

1.2.

Functional Requirements (FRs)³

The system must provide the following top services to the users:

1. **Search & Browse Movies:** Users (Guests/Members) can search for movies by title, genre, or release date⁴.
2. **View Movie Details:** Display detailed information including trailer, synopsis, cast, and showtimes.
3. **Book Ticket (Make Purchase):** Members can select a movie, showtime, and seats to purchase tickets⁵.
4. **Manage Movies:** Admins can add, update, or delete movie information in the database.
5. **Manage Schedule:** Admins can schedule showtimes for specific movies in specific theater rooms.

1.3.

Non-Functional Requirements (NFRs) ⁶

The system must meet the following quality attributes:

1. **Performance:** The system must load the seat selection map within 2 seconds under normal load⁷.
2. **Availability:** The system must be available 99.9% of the time to ensure users can book tickets at any time⁸.
3. **Security:** User passwords and payment information must be encrypted and stored securely (e.g., using hashing and SSL/TLS)⁹.

1.4.

Architecturally Significant Requirements (ASRs) ¹⁰¹⁰¹⁰¹⁰

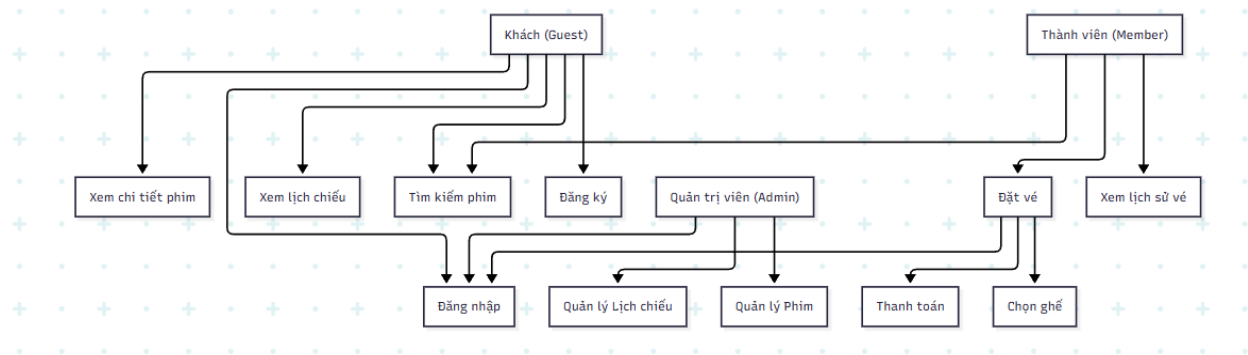
The following requirements will heavily influence the choice of software architecture:

ASR Name	Statement	Architectural Impact
High Concurrency (Seat Locking)	The system must handle situations where multiple users attempt to select the exact same seat simultaneously.	This requires a strict transaction management strategy and locking mechanisms (Optimistic or Pessimistic Locking) in the Business Logic or Database layer to prevent race conditions ¹¹ .
High Scalability	The system must handle a traffic surge (e.g., 10,000 concurrent users) during blockbuster movie releases ¹² .	This drives the decision towards a Microservices Architecture or a load-balanced environment, allowing the "Booking Service" to scale independently from other components ¹³ .
Data Consistency	Ticket inventory (seats) must be strictly consistent across all user views in real-time.	This impacts the Database Design , prioritizing ACID properties (Atomicity, Consistency, Isolation, Durability) over eventual consistency for the ticketing module ¹⁴ .

2. USE CASE MODELING ¹⁵

2.1. System Context

The diagram below represents the system boundary, actors, and main functionalities of the Movie Ticket Booking System¹⁶.



2.2.

Critical Use Case Description: "Book Ticket" ¹⁷

This section details the relationships for the primary flow "**Book Ticket**" (Make Purchase), as modeled in the diagram:

- **Primary Use Case: Book Ticket**
 - *Actor:* Member
 - *Description:* The process of selecting a movie and purchasing a ticket.
- **Include Relationships (<<include>>)¹⁸:**
 - **Book Ticket <<include>> Log In:** To book a ticket, the user *must* be identified by the system. If a Guest tries to book, the system forces the Log In process.
 - **Book Ticket <<include>> Select Seat:** Selecting a specific seat is a mandatory part of the booking process.
 - **Book Ticket <<include>> Payment:** A booking is not complete until the payment transaction is successfully processed.
- **Extend Relationships (<<extend>>)¹⁹:**
 - (*Optional - if you added it*) **Apply Coupon <<extend>> Payment:** During the payment step, a user *may* optionally choose to apply a discount code, but it is not required to complete the transaction.