

Online movie ticket booking platform

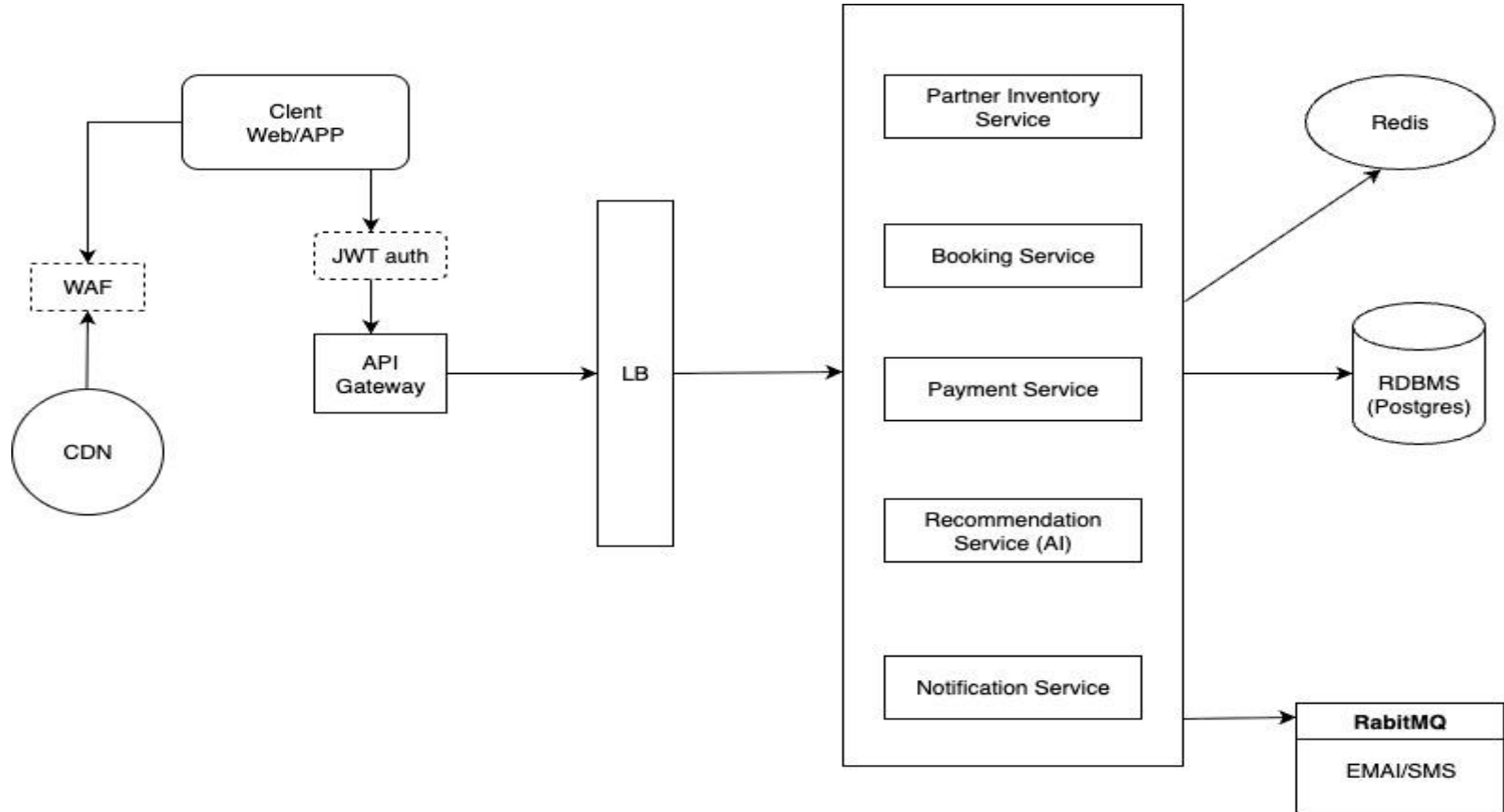
Functional Requirements

- **Multi-Tenancy** (Treat each theatre chain or organizer as a **partner**)
- Theatre chains can register.
- Partners add their show(performances), screens, and seat layout (rows/seats).
- Partners set show, date/time, language, and base ticket price for each screen.
- Customers pick a city/show/date, see movies and showtimes.
- When a customer selects seats, they are held for **5 minutes (seat lock)** so two people don't book the same seat.
- Apply Discount.
- Customers pay online, on success they get a booking confirmation by email/SMS.
- **AI- Recommendation** movies/shows.

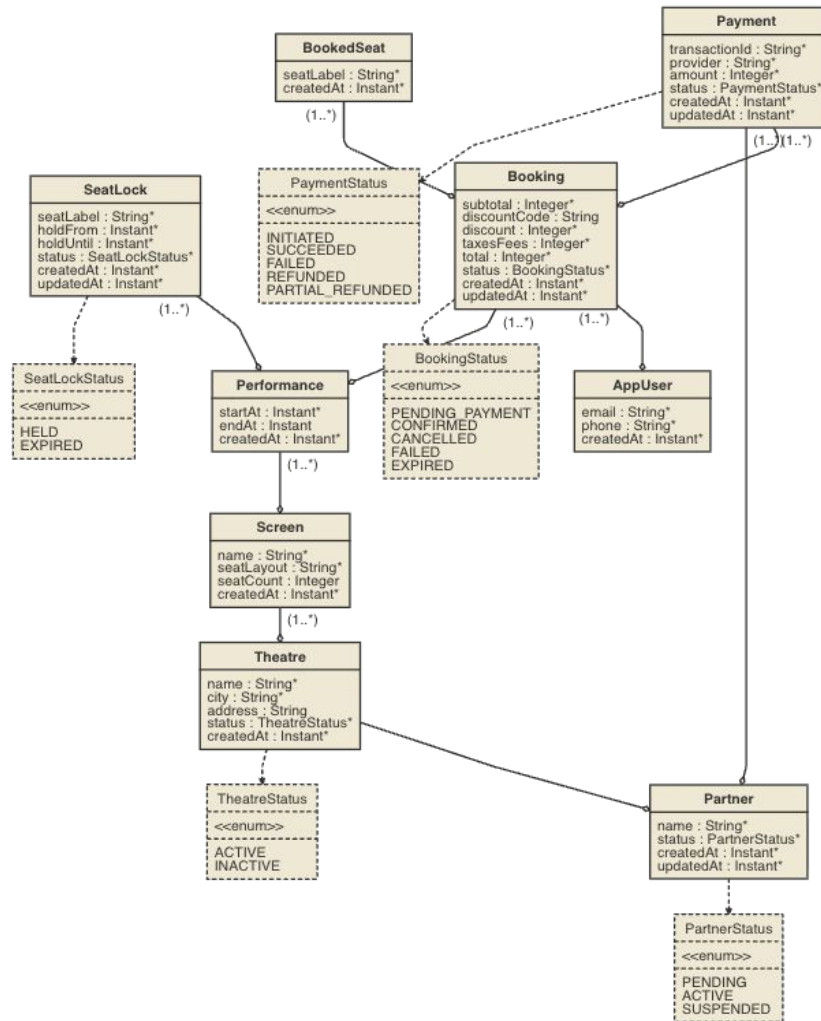
Non Functional Requirements

- Performance
- Availability
- Resilience - (**resilience4j**)
- Security (**Oauth2/Auth0**)
- Observability - **Spring Boot Actuator (health, liveness, readiness)**
- Maintainability & Delivery (CI/CD)
- Scalability (The system should be scalable to a growing user base)

High level Architecture



Data Model



Tech Stack

- Java 21
- Spring Boot 3.5.5
- Docker
- Gradle
- MySQL 8
- Redis 7
- RabbitMQ
- AI: Spring AI (**spring-ai-starter-model-ollama**)
- **Resilience4j**
- Testing: JUnit 5, Mockito
- Docs: springdoc-openapi / **Swagger UI**

API's

localhost:8084/swagger-ui/index.html#

Swagger
powered by SMARTBEAR

/v3/api-docs

Explore

OpenAPI definition v0 OAS 3.0

[/v3/api-docs](#)

Servers

http://localhost:8084 - Generated server url

theatre-controller

GET /api/v1/theatres

POST /api/v1/theatres

show-controller

GET /api/v1/shows

POST /api/v1/shows

screen-controller

GET /api/v1/screens

POST /api/v1/screens

GET /api/v1/screens/{screenId}

Browse Shows

HTTP

CatchMyShow / Browse Shows

Save

Share

GET

http://localhost:8084/api/v1/shows?city=Pune&showId=1&date=2025-09-07

Send

Params

Authorization

Headers (6)

Body

Scripts

Tests

Settings

Cookies

Query Params

<input checked="" type="checkbox"/> Key	Value	Description	...	Bulk Edit
<input checked="" type="checkbox"/> city	Pune			
<input checked="" type="checkbox"/> showId	1			
<input checked="" type="checkbox"/> date	2025-09-07			
<input type="checkbox"/> Key	Value	Description		

Body

Cookies

Headers (5)

Test Results

200 OK

498 ms

277 B

Save Response

{ } JSON

Preview

Visualize

```
1  {
2    "id": 1,
3    "screenId": 1,
4    "startAt": "2025-09-07T19:30",
5    "endAt": "2025-09-07T22:30",
6    "createdAt": "2025-09-07T22:56:15"
7  }
```


Create Booking

HTTP CatchMyShow / Create Booking Save Share

POST http://localhost:8083/api/v1/bookings Send

Params Authorization Headers (8) **Body** Scripts Tests Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL **JSON** Beautify

```
1 {
2   "showId": 1,
3   "seats": ["A20", "B20"],
4   "subtotal": 700,
5   "discountCode": "WELCOME10",
6   "discount": 50,
7   "taxesFees": 100,
8   "total": 750,
9   "userId": 1,
10  "ttlSeconds": 300
11 }
```

Body Cookies Headers (6) Test Results 201 Created • 2.88 s • 268 B • Save Response

{} JSON Preview Visualize

```
1 {
2   "id": 5,
3   "status": "PENDING_PAYMENT",
4   "showId": 1,
5   "userId": 1,
6   "total": 750
7 }
```

AI Recommendation

```
localhost:8080/api/v1/recommendations?userId=123&city=Pune
Pretty print ☒

{
  "items": [
    {
      "movieId": 4,
      "title": "Kolkata Crimes",
      "score": 0.63,
      "reason": "genre match"
    },
    {
      "movieId": 2,
      "title": "Monsoon Melodies",
      "score": 0.56,
      "reason": "language match"
    },
    {
      "movieId": 6,
      "title": "Chai & Chess",
      "score": 0.56,
      "reason": "language match"
    },
    {
      "movieId": 3,
      "title": "Quantum Heist",
      "score": 0.5,
      "reason": "language match"
    },
    {
      "movieId": 8,
      "title": "Street Beats",
      "score": 0.5,
      "reason": "language match"
    }
  ]
}
```

Security and OWASP (Future Scope)

- Input Validations
- Secrets Management (Vault)
- Refined Exception handling
- Auth0 = Authorization Server (OIDC)
- Services = OAuth2 Resource Servers (JWT)
- RBAC: Roles → Permissions → Tokens (permissions claim)
- Protect priority endpoints; add @PreAuthorize
- SAST using SonarQube
- DAST & API Testing
- GDPR (PII scrubbing)

Roadmap

- CI/CD (GitHub Actions)
- Payments integration
- Idempotency
- Kafka for CDC/events & analytics
- Search & discovery (movie metadata, filters)
- AI - Real user/movie data and behavioral embeddings.
- Observability - (Grafana/Prometheus)
- i18n (time zones, currencies), multi-region rollout

Thank you