

Problem Statement: Movie Ticket Reservation API

Models

RoleModel

| Field Name | Datatype | Primary Key | Comments |
|------------|----------|-------------|-----------------------------|
| id | Integer | Yes | Autoincrement |
| rolename | String | No | Values: "CUSTOMER", "ADMIN" |

UserModel

| Field Name | Datatype | Primary Key | Foreign Key | Comments |
|------------|----------|-------------|-------------|-----------------|
| id | Integer | Yes | No | Autoincrement |
| username | String | No | No | - |
| email | String | No | No | Unique |
| password | String | No | No | - |
| role | Integer | No | Yes | FK to RoleModel |

MovieModel

| Field Name | Datatype | Primary Key | Comments |
|------------|----------|-------------|--------------------------------------|
| id | Integer | Yes | Autoincrement |
| title | String | No | - |
| category | String | No | Example: "Action", "Comedy", "Drama" |
| basePrice | Double | No | Price per ticket |
| available | Boolean | No | Default: True |

ReservationModel

| Field Name | Datatype | Primary Key | Foreign Key | Comments |
|-------------|----------|-------------|-------------|--|
| id | Integer | Yes | | Autoincrement |
| customerId | Integer | No | | FK to UserModel |
| movieId | Integer | No | | FK to MovieModel |
| quantity | Integer | No | No | - |
| discountPct | Double | No | No | Default: 0 |
| gstPct | Double | No | No | Default: 18 |
| serviceFee | Double | No | No | Default: 50 |
| totalAmount | Double | No | No | Calculated: (quantity * basePrice - discount) + gst + serviceFee |

| reservationDate | String | No | Default: Current date |
| status | String | No | Default: "reserved" |

JWT Authentication

- JWT token required in Authorization header:

Authorization: Bearer <JWT_TOKEN>

- Roles (CUSTOMER / ADMIN) are identified from the token.
- All endpoints except /login require authentication.

Endpoints

1. POST /login

Input Request:

```
`json
{
  "email": "cust1@gmail.com",
  "password": "cust123$"
}
```

Expected Output (200 OK):

```
`json
{
  "email": "cust1@gmail.com",
  "access": "<access_token>",
  "refresh": "<refresh_token>",
  "status": 200
}
```

2. POST /reservation/create (CUSTOMER only)

Input Request:

```
`json
{
  "movieId": 201,
  "quantity": 3,
  "discountPct": 10
}
```

Manipulations:

- $\text{discount} = (\text{quantity} \times \text{basePrice}) \times (\text{discountPct}/100)$
- $\text{gst} = (\text{quantity} \times \text{basePrice} - \text{discount}) \times (\text{gstPct}/100)$
- $\text{totalAmount} = (\text{quantity} \times \text{basePrice} - \text{discount}) + \text{gst} + \text{serviceFee}$

Expected Output (201 CREATED):

```
`json
{
  "id": 9001,
  "customerId": 1,
  "movieId": 201,
  "quantity": 3,
  "discountPct": 10,
  "gstPct": 18,
  "serviceFee": 50,
  "totalAmount": 13286,
  "reservationDate": "11/12/2025",
  "status": "reserved"
}
```

3. GET

/reservation/list?category=Action&minAmount=5000&maxAmount=15000&status=reserved

(Both roles)

Expected Output (200 OK):

```
`json
[
  {
    "id": 9001,
    "customerId": 1,
    "movieId": 201,
    "category": "Action",
    "quantity": 3,
    "totalAmount": 13286,
    "status": "reserved"
  }
]
```

4. PATCH /reservation/update/{id} (ADMIN only)

Input Request:

```
`json
{ "status": "confirmed" }
```

Expected Output (200 OK):

```
`json
{
  "id": 9001,
  "customerId": 1,
  "movieId": 201,
  "totalAmount": 13286,
  "status": "confirmed"
},
`
```

5. DELETE /reservation/delete/{id} (Conditional access)

Expected Output (204 No Content):

```
`json
{ "message": "deleted successfully" }
`
```

Response Codes

- 200 OK – Success
- 201 CREATED – Resource created
- 204 No Content – Deleted successfully
- 400 Bad Request – Invalid input / Not found
- 403 Forbidden – Permission denied

✓ This new problem statement adds extra manipulations:

- Discount + GST + service fee combined.
- Query params with four filters: category, minAmount, maxAmount, status.