

 Problem Statement: Event Ticket Booking 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

TicketModel

Field Name	Datatype	Primary Key	Foreign Key	Comments
id	Integer	Yes		Autoincrement
customerId	Integer	No		FK to UserModel
eventName	String	No	No	-
category	String	No	No	Values: "VIP", "Regular", "Student"
quantity	Integer	No	No	-
basePrice	Double	No	No	Price per ticket based on category
discountPct	Double	No	No	Default: 0
serviceFee	Double	No	No	Default: 100
gstPct	Double	No	No	Default: 12
totalAmount	Double	No		Calculated: (quantity * basePrice - discount) + gst + serviceFee
bookingDate	String	No		Default: Current date
status	String	No		Default: "booked"

 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 /ticket/create (CUSTOMER only)

Input Request:

```
'json
{
  "eventName": "Rock Concert",
  "category": "VIP",
  "quantity": 2,
  "discountPct": 5
}'
```

Manipulations:

- basePrice auto-set based on category:
 - VIP → 5000
 - Regular → 2500
 - Student → 1000
- discount = (quantity * basePrice) * (discountPct/100)
- gst = (quantity * basePrice - discount) * (gstPct/100)
- totalAmount = (quantity * basePrice - discount) + gst + serviceFee

Expected Output (201 CREATED):

```
'json
{
  "id": 5001,
  "customerId": 1,
  "eventName": "Rock Concert",
  "category": "VIP",
  "quantity": 2,
  "basePrice": 5000,
  "discountPct": 5,
  "serviceFee": 100,
  "gstPct": 12,
  "totalAmount": 10690,
  "bookingDate": "06/12/2025",
  "status": "booked"
}
```

.

3. GET /ticket/list?category=VIP&minAmount=5000&maxAmount=15000&status=booked
(Both roles)

Expected Output (200 OK):

```
'json
[
  {
    "id": 5001,
    "customerId": 1,
    "eventName": "Rock Concert",
    "category": "VIP",
    "quantity": 2,
    "totalAmount": 10690,
    "status": "booked"
  }
]
```

.

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

Input Request:

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

Expected Output (200 OK):

```
'json
{
```

```
"id": 5001,  
"customerId": 1,  
"eventName": "Rock Concert",  
"category": "VIP",  
"quantity": 2,  
"totalAmount": 10690,  
"status": "confirmed"  
}  
`
```

5. DELETE /ticket/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:

- Auto-set basePrice by category.
- Discount + GST + service fee combined into totalAmount.
- Query params with four filters: category, minAmount, maxAmount, status.