1. Create a New Booking

Request

- **Endpoint:** `POST /bookings/{transaction_id}`
- **JSON Payload:**

```json
{
  "show_id": 12,
  "customer_id": 1001,
  "seats": "B2,B3",
  "amount": 90
}
```

Response

- **Status Code:** `201 Created`
- **JSON Response:**

```json
{
  "id": 101,
  "show_id": 12,
  "customer_id": 1001,
  "seats": "B2,B3",
  "amount": 90,
  "is_paid": false,
  "is_used": false,
  "datetime": "2023-10-22T19:00:00"
}
```

2. Cancel a Booking

Request

- **Endpoint:** `DELETE /bookings/{booking_id}/{transaction_id}`
- **Path Parameters:**
  - `booking_id`: 101
  - `transaction_id`: "txn89012"

Response

- **Status Code:** `200 OK`
- **JSON Response:**

```
{
  "id": 101,
  "show_id": 12,
  "customer_id": 1001,
  "seats": "B2,B3",
  "amount": 90,
  "is_paid": false,
  "is_used": false,
  "datetime": "2023-10-22T19:00:00"
}
```

3. Pay for a Booking

Request

- **Endpoint:** `PUT /bookings/{booking_id}/pay/{transaction_id}`
- **Path Parameters:**
  - `booking_id`: 101
  - `transaction_id`: "txn89013"

Response

- **Status Code:** `200 OK`
- **Response:** The server returns a PDF file of the ticket (not in JSON format).

4. Validate a Booking

Request

- **Endpoint:** `PUT /bookings/{booking_id}/validate/{transaction_id}`
- **Path Parameters:**
  - `booking_id`: 101
  - `transaction_id`: "txn89014"

Response

- **Status Code:** `200 OK`
- **JSON Response:**

```
{
  "id": 101,
  "show_id": 12,
  "customer_id": 1001,
  "seats": "B2,B3",
  "amount": 90,
  "is_paid": true,
  "is_used": true,
  "datetime": "2023-10-22T19:00:00"
}
```

5. Inform Users of Show Cancellation

Request

- **Endpoint:** `PUT /shows/{show_id}/cancel/{transaction_id}`
- **Path Parameters:**
  - `show_id`: 12
  - `transaction_id`: "txn89015"

Response

- **Status Code:** `200 OK`
- **JSON Response:**

```
"25 users affected and notified"
```

6. Get a Booking by ID

Request

- **Endpoint:** `GET /bookings/{booking_id}/`
- **Path Parameters:**
  - `booking_id`: 101

Response

- **Status Code:** `200 OK`
- **JSON Response:**

```
{
  "id": 101,
  "show_id": 12,
  "customer_id": 1001,
  "seats": "B2,B3",
  "amount": 90,
  "is_paid": true,
  "is_used": false,
  "datetime": "2023-10-22T19:00:00"
}
```

7. Get All Bookings

Request

- **Endpoint:** `GET /bookings/all`

Response

- **Status Code:** `200 OK`
- **JSON Response:**

```
[
  {
    "id": 100,
    "show_id": 10,
    "customer_id": 1000,
    "seats": "A1,A2",
    "amount": 85,
    "is_paid": true,
    "is_used": true,
    "datetime": "2023-10-21T18:30:00"
  },
  {
    "id": 101,
    "show_id": 12,
    "customer_id": 1001,
    "seats": "B2,B3",
    "amount": 90,
    "is_paid": true,
    "is_used": false,
    "datetime": "2023-10-22T19:00:00"
  }
]
```

8. Get Bookings by User ID

Request

- **Endpoint:** `GET /bookings/user/{user_id}/`
- **Path Parameters:**
  - `user_id`: 1001

Response

- **Status Code:** `200 OK`
- **JSON Response:**

```
[
  {
    "id": 101,
    "show_id": 12,
    "customer_id": 1001,
    "seats": "B2,B3",
    "amount": 90,
    "is_paid": true,
    "is_used": false,
    "datetime": "2023-10-22T19:00:00"
  },
  {
    "id": 102,
    "show_id": 15,
    "customer_id": 1001,
    "seats": "C1,C2",
    "amount": 95,
    "is_paid": false,
    "is_used": false,
    "datetime": "2023-10-23T20:00:00"
  }
]
```

9. Get All Bookings for a Show

Request

- **Endpoint:** `GET /bookings/show/{show_id}/`
- **Path Parameters:**
  - `show_id`: 12

Response

- **Status Code:** `200 OK`
- **JSON Response:**

```
[
  {
    "id": 101,
    "show_id": 12,
    "customer_id": 1001,
    "seats": "B2,B3",
    "amount": 90,
    "is_paid": true,
    "is_used": false,
    "datetime": "2023-10-22T19:00:00"
  },
  {
    "id": 103,
    "show_id": 12,
    "customer_id": 1003,
    "seats": "A1,A2",
    "amount": 90,
    "is_paid": false,
    "is_used": false,
    "datetime": "2023-10-22T19:00:00"
  }
]
```

10. Get Seat Matrix for a Show

Request

- **Endpoint:** `GET /bookings/show/{show_id}/seatmatrix`
- **Path Parameters:**
  - `show_id`: 12

Response

- **Status Code:** `200 OK`
- **JSON Response:**

```
[
  [0, 1, 0, 0, 1],
  [1, 1, 0, 0, 0],
  [0, 0, 0, 1, 1]
]
```

In this matrix, '1' represents a booked seat, and '0' represents an available seat.

11. Simple Test Endpoint

Request

- **Endpoint:** `GET /hello`

Response

- **Status Code:** `200 OK`
- **Response:**

```
"Hello World!"
```