

ALL ENDPOINT FUNCTIONALITIES:

Here's a detailed checklist:

Requirement	Covered?	Location
✓ Hotel Chains with address, phone, email, number of hotels	✓	HotelChainController.java HotelChainController
✓ Hotels with contact info, address, star rating, and relation to chains	✓	HotelController.java HotelController
✓ Rooms with price, capacity, amenities, view, extendability, damage	✓	RoomController.java + amenities supportRoomController
✓ Customers with ID type, address, registration date	✓	CustomerController.java CustomerController
✓ Employees with SSN/SIN, roles, address, hotel, and manager role	✓	EmployeeController.java EmployeeController
✓ Room Availability Check	✓	BookingController.java → /bookings/availability BookingController
✓ Booking Flow (Create, View, Confirm, Cancel)	✓	BookingController.java BookingController
✓ Renting Flow (Walk-in and From Booking)	✓	RentingController.java RentingController
✓ Check-In / Booking to Renting	✓	RentingController.java → /from-booking RentingController
✓ Check-Out (updates dateOut)	✓	RentingController.java → /checkout RentingController

✓ Search Rooms with Filters (capacity, amenities, date, view)	✓	<code>RoomController.java</code> → <code>/rooms/searchRoomController</code>
✓ Delete HotelChain, Hotel, Room	✓	All 3 controller files support <code>@DeleteMapping</code> <code>HotelChainController</code> <code>HotelControllerRoomController</code>
✓ Maintain Booking/Renting History (archive)	✓	Data is persisted in DB (no cascade delete from booking/renting to customer or room)
✓ Prevent Orphaned Rooms/Hotels/Chains	✓	DB foreign key constraints ensure referential integrity
✓ Manager Assigned to Each Hotel	✓	Managed via <code>EmployeeController</code> and schema design

🟡 Optional/Nice-to-Have (Not Required but Valuable)

Customer booking lookup via email	✓	<code>/bookings/customer/{email}</code> <code>BookingController</code>
Rental history by customer	✓	<code>/rentings/customer/{email}</code> <code>RentingController</code>
Room amenities update	✓	<code>/rooms/{id}/amenities</code> <code>RoomController</code>
Secure all routes via <code>SecurityConfig</code>	✓	Confirmed during previous steps



Hotel Booking & Management System – API Documentation



Authentication



Login

POST `/employees/login`

Log in as an employee or manager and receive a JWT token.

Request Body:

json

CopyEdit

```
{  
  "email": "alice.manager@hotel.com",  
  "password": "password123"  
}
```

Response: JWT Token (string)



Hotel Chain Endpoints



Get All Hotel Chains

GET `/hotel-chains`



Get Hotel Chain by ID

GET `/hotel-chains/{id}`



Create Hotel Chain (*Manager Only*)

POST `/hotel-chains`

Body:

json

CopyEdit

```
{  
  "name": "Luxury Stay",  
  "address": "123 Chain Blvd",  
  "phone": "111-222-3333",  
  "chainNumber": "LS-001"  
}
```

 Update Hotel Chain

PUT `/hotel-chains/{id}`

 Delete Hotel Chain

DELETE `/hotel-chains/{id}`

Hotel Endpoints

 Get All Hotels

GET `/hotels`

 Get Hotel by ID

GET `/hotels/{id}`

 Create Hotel (*Manager Only*)

POST `/hotels`

Body:

json

CopyEdit

```
{
```

```
{
  "name": "Luxury Downtown",
  "address": "123 Street",
  "phone": "111-222-0000",
  "category": "Luxury",
  "starCount": 5,
  "hotelChain": {
    "id": 1
  }
}
```

Update Hotel

PUT `/hotels/{id}`

Delete Hotel

DELETE `/hotels/{id}`

Room Endpoints

Get All Rooms

GET `/rooms`

Get Room by ID

GET `/rooms/{id}`

Search Rooms by Hotel ID

GET `/rooms/hotel/{hotelId}`

Check Room Availability

GET `/rooms/available?checkIn=YYYY-MM-DD&checkOut=YYYY-MM-DD`

Search Rooms by Filters

POST `/rooms/search`

Example Body:

json

CopyEdit

```
{
  "capacity": "double",
  "view": "sea",
  "wifi": true,
  "ac": true,
  "checkInDate": "2025-05-01",
  "checkOutDate": "2025-05-04"
}
```

+ Create Room (*Manager Only*)

POST `/rooms`

 Update Room (*Manager Only*)

PUT `/rooms/{id}`

× Delete Room (*Manager Only*)

DELETE `/rooms/{id}`

 Update Room Amenities (*Manager Only*)

PUT `/rooms/{id}/amenities`

Customer Endpoints

 Get All Customers

GET `/customers`

 **Get Customer by ID**

GET */customers/{id}*

+ Create Customer

POST */customers*

 **Update Customer**

PUT */customers/{id}*

✗ Delete Customer

DELETE */customers/{id}*

Employee Endpoints

 **Get All Employees (*Manager Only*)**

GET */employees*

 **Get Employee by ID**

GET */employees/{id}*

+ Create Employee (*Manager Only*)

POST */employees*

 **Update Employee (*Manager Only*)**

PUT */employees/{id}*

✗ Delete Employee (*Manager Only*)

DELETE */employees/{id}*



Booking Endpoints

✓ Check Availability

GET

`/bookings/availability?roomId=1&checkInDate=2025-04-01&checkOutDate=2025-04-03`

+ Book Room (No Auth Required)

POST `/bookings`

Body:

json

CopyEdit

```
{
  "customer": {
    "fullName": "John Doe",
    "email": "john@example.com",
    "phone": "555-5555",
    "address": "123 St",
    "idType": "Driver License"
  },
  "roomId": 1,
  "checkInDate": "2025-04-01",
  "checkOutDate": "2025-04-03"
}
```



Get All Bookings (*Employee/Manager Only*)

GET `/bookings`



Get Bookings by Customer (*Employee/Manager Only*)

GET `/bookings/customer/{email}`



Update Booking Status (*Employee/Manager Only*)

PUT `/bookings/{id}/status`

Body:

json

CopyEdit

```
{
  "status": "Confirmed"
}
```

Renting Endpoints

Rent Room Without Booking (*Walk-In*)

POST `/rentings`

Body:

json

CopyEdit

```
{
  "roomId": 1,
  "employeeId": 2,
  "customer": {
    "fullName": "Guest",
    "email": "guest@example.com",
    "phone": "555-5555",
    "address": "Lobby",
    "idType": "SSN"
  },
  "dateOut": "2025-04-05T12:00:00",
  "payment": 400.00
}
```

Rent Room from Booking (*Check-In*)

POST `/rentings/from-booking`

Body:

json

CopyEdit


```
{  
  "bookingId": 5,  
  "employeeId": 2,  
  "dateOut": "2025-04-05T12:00:00",  
  "payment": 500.00  
}
```

 Get All Rentings (*Manager Only*)

GET `/rentings`

 Get Rentings by Customer (*Employee/Manager*)

GET `/rentings/customer/{email}`

 Check-Out Room (*Employee/Manager*)

PUT `/rentings/{id}/checkout`



HotelChainController – API Test Cases & Responses

Base route: `/hotel-chains`

1. GET /hotel-chains

✓ Test Case: Get all hotel chains

Request:

bash

CopyEdit

```
curl -X GET http://localhost:8080/hotel-chains
```

Expected Response:

json

CopyEdit

```
[
  {
    "id": 1,
    "name": "Luxury Stay",
    "address": "123 Chain Blvd, City A",
    "phone": "111-222-3333",
    "chainNumber": "LS-001"
  },
  {
    "id": 2,
    "name": "Budget Inn",
    "address": "789 Cheap Rd, Town B",
    "phone": "444-555-6666",
    "chainNumber": "BI-002"
  }
]
```

2. GET /hotel-chains/{id}

✓ Test Case: Get a specific hotel chain by ID

Request:

bash

CopyEdit

```
curl -X GET http://localhost:8080/hotel-chains/1
```

Expected Response:

json

CopyEdit

```
{
  "id": 1,
  "name": "Luxury Stay",
  "address": "123 Chain Blvd, City A",
  "phone": "111-222-3333",
  "chainNumber": "LS-001"
}
```

3. POST /hotel-chains



Requires manager token



Test Case: Create a new hotel chain

Request:

bash

CopyEdit

```
curl -X POST http://localhost:8080/hotel-chains \
-H "Authorization: Bearer <manager-token>" \
-H "Content-Type: application/json" \
-d '{
  "name": "Modern Hotels",
  "address": "45 Innovation Ave, City X",
  "phone": "222-333-4444",
  "chainNumber": "MH-003"
}
```

```
}'
```

Expected Response:

json

CopyEdit

```
{  
  "id": 3,  
  "name": "Modern Hotels",  
  "address": "45 Innovation Ave, City X",  
  "phone": "222-333-4444",  
  "chainNumber": "MH-003"  
}
```

4. PUT /hotel-chains/{id}

 Requires manager token

 Test Case: Update an existing hotel chain

Request:

bash

CopyEdit

```
curl -X PUT http://localhost:8080/hotel-chains/3 \  
-H "Authorization: Bearer <manager-token>" \  
-H "Content-Type: application/json" \  
-d '{  
  "name": "Modern Hotels Intl.",  
  "address": "99 Updated St, City X",  
  "phone": "222-333-9999",  
  "chainNumber": "MH-003"  
}'
```

Expected Response:

json

CopyEdit

```
{
  "id": 3,
  "name": "Modern Hotels Intl.",
  "address": "99 Updated St, City X",
  "phone": "222-333-9999",
  "chainNumber": "MH-003"
}
```

5. DELETE /hotel-chains/{id}



Requires manager token



Test Case: Delete a hotel chain

Request:

bash

CopyEdit

```
curl -X DELETE http://localhost:8080/hotel-chains/3 \
-H "Authorization: Bearer <manager-token>"
```

Expected Response:

css

CopyEdit

204 No Content



HotelController – API Test Cases & Responses

Base route: /hotels

1. GET /hotels

✓ Test Case: Get all hotels

Request:

bash

CopyEdit

```
curl -X GET http://localhost:8080/hotels
```

Expected Response:

json

CopyEdit

```
[
  {
    "id": 1,
    "name": "Luxury Stay Downtown",
    "address": "456 Main St, City A",
    "phone": "111-000-1111",
    "category": "Luxury",
    "starCount": 5,
    "hotelChain": {
      "id": 1,
      "name": "Luxury Stay"
    }
  },
  {
    "id": 2,
    "name": "Budget Inn North",
    "address": "789 North Rd, City B",
    "phone": "222-111-9999",
    "category": "Economy",
    "starCount": 3,
    "hotelChain": {
```

```
    "id": 2,  
    "name": "Budget Inn"  
  }  
}  
]
```

2. GET /hotels/{id}

✓ Test Case: Get a specific hotel by ID

Request:

bash

CopyEdit

```
curl -X GET http://localhost:8080/hotels/1
```

Expected Response:

json

CopyEdit

```
{  
  "id": 1,  
  "name": "Luxury Stay Downtown",  
  "address": "456 Main St, City A",  
  "phone": "111-000-1111",  
  "category": "Luxury",  
  "starCount": 5,  
  "hotelChain": {  
    "id": 1,  
    "name": "Luxury Stay"  
  }  
}
```

3. POST /hotels

 Requires **ROLE_Manager**

 **Test Case: Create a new hotel**

Request:

bash

CopyEdit

```
curl -X POST http://localhost:8080/hotels \
-H "Authorization: Bearer <manager-token>" \
-H "Content-Type: application/json" \
-d '{
  "name": "City Center Inn",
  "address": "900 Central Ave, City C",
  "phone": "333-555-6666",
  "category": "Mid-range",
  "starCount": 4,
  "hotelChain": {
    "id": 1
  }
}'
```

Expected Response:

json

CopyEdit

```
{
  "id": 3,
  "name": "City Center Inn",
  "address": "900 Central Ave, City C",
  "phone": "333-555-6666",
  "category": "Mid-range",
  "starCount": 4,
  "hotelChain": {
    "id": 1,
```

```
    "name": "Luxury Stay"
  }
}
```

4. PUT /hotels/{id}

 Requires **ROLE_Manager**

 **Test Case: Update an existing hotel**

Request:

bash

CopyEdit

```
curl -X PUT http://localhost:8080/hotels/3 \
-H "Authorization: Bearer <manager-token>" \
-H "Content-Type: application/json" \
-d '{
  "name": "City Center Inn - Updated",
  "address": "901 Updated Blvd, City C",
  "phone": "333-555-9999",
  "category": "Mid-range",
  "starCount": 4,
  "hotelChain": {
    "id": 1
  }
}'
```

Expected Response:

json

CopyEdit

```
{
  "id": 3,
  "name": "City Center Inn - Updated",
```

```
"address": "901 Updated Blvd, City C",
"phone": "333-555-9999",
"category": "Mid-range",
"starCount": 4,
"hotelChain": {
  "id": 1,
  "name": "Luxury Stay"
}
}
```

5. DELETE /hotels/{id}

 Requires **ROLE_Manager**

 **Test Case: Delete a hotel**

Request:

bash

CopyEdit

```
curl -X DELETE http://localhost:8080/hotels/3 \
-H "Authorization: Bearer <manager-token>"
```

Expected Response:

css

CopyEdit

204 No Content

RoomController – API Test Cases & Responses

Base route: `/rooms`

1. GET /rooms

✓ Test Case: Get all rooms

Request:

bash

CopyEdit

```
curl -X GET http://localhost:8080/rooms
```

Expected Response:

json

CopyEdit

```
[
  {
    "id": 1,
    "roomNumber": "101",
    "price": 120.0,
    "capacity": "double",
    "extended": false,
    "problems": null,
    "view": "sea",
    "hotel": {
      "id": 1,
      "name": "Luxury Stay Downtown"
    },
    "amenities": {
      "id": 1,
      "wifi": true,
      "ac": true,
      "pool": false,
      "gym": false,

```

```
    "spa": false,  
    "parking": true,  
    "fridge": true,  
    "coffee": true  
  }  
}  
]
```

2. GET /rooms/{id}

✓ Test Case: Get a specific room by ID

Request:

bash

CopyEdit

```
curl -X GET http://localhost:8080/rooms/1
```

Expected Response: Same as above, for that room.

3. GET /rooms/hotel/{hotelId}

✓ Test Case: Get all rooms for a specific hotel

Request:

bash

CopyEdit

```
curl -X GET http://localhost:8080/rooms/hotel/1
```

4. GET /rooms/available

✓ Test Case: Get available rooms between dates

Request:

bash

CopyEdit

```
curl -X GET
```

```
"http://localhost:8080/rooms/available?checkIn=2025-04-01&checkOut=2025-04-05"
```

Expected Response: List of rooms not booked in that time frame.

5. POST /rooms/search

✓ Test Case: Filter rooms with search criteria

Request:

bash

CopyEdit

```
curl -X POST http://localhost:8080/rooms/search \
-H "Content-Type: application/json" \
-d '{
  "capacity": "double",
  "view": "sea",
  "wifi": true,
  "ac": true,
  "checkInDate": "2025-04-01",
  "checkOutDate": "2025-04-05"
}'
```

Expected Response: List of available rooms matching filters

6. POST /rooms

 Requires **ROLE_Manager**

 **Test Case: Create a new room**

Request:

bash

CopyEdit

```
curl -X POST http://localhost:8080/rooms \
-H "Authorization: Bearer <manager-token>" \
-H "Content-Type: application/json" \
-d '{
    "roomNumber": "202",
    "price": 175.0,
    "capacity": "single",
    "extended": false,
    "problems": null,
    "view": "mountain",
    "hotel": {
        "id": 1
    }
}'
```

Expected Response:

json

CopyEdit

```
{
    "id": 4,
    "roomNumber": "202",
    ...
}
```

7. PUT /rooms/{id}

 Requires **ROLE_Manager**

✓ **Test Case: Update room details**

Request:

bash

CopyEdit

```
curl -X PUT http://localhost:8080/rooms/4 \  
-H "Authorization: Bearer <manager-token>" \  
-H "Content-Type: application/json" \  
-d '{  
  "roomNumber": "202-A",  
  "price": 180.0,  
  "capacity": "double",  
  "extended": true,  
  "problems": "Leaky faucet",  
  "view": "mountain",  
  "hotel": {  
    "id": 1  
  }  
'
```

8. DELETE /rooms/{id}

 Requires **ROLE_Manager**

Request:

bash

CopyEdit

```
curl -X DELETE http://localhost:8080/rooms/4 \  
-H "Authorization: Bearer <manager-token>"
```

9. PUT /rooms/{id}/amenities

 Requires **ROLE_Manager**

 **Test Case: Update amenities for a room**

Request:

bash

CopyEdit

```
curl -X PUT http://localhost:8080/rooms/4/amenities \
-H "Authorization: Bearer <manager-token>" \
-H "Content-Type: application/json" \
-d '{
  "wifi": true,
  "ac": true,
  "pool": true,
  "gym": false,
  "spa": true,
  "parking": true,
  "fridge": true,
  "coffee": true
}'
```

Expected Response:

json

CopyEdit

```
{
  "id": 4,
  "wifi": true,
  "ac": true,
  ...
}
```



CustomerController – API Test Cases & Responses

Base route: `/customers`

1. GET /customers



May be manager-only depending on your config.



Test Case: Get all customers

Request:

bash

CopyEdit

```
curl -X GET http://localhost:8080/customers \
-H "Authorization: Bearer <manager-token>"
```

Expected Response:

json

CopyEdit

```
[
  {
    "id": 1,
    "fullName": "John Doe",
    "email": "john@example.com",
    "phone": "555-1234",
    "address": "123 Main St",
    "idType": "Driver License",
    "regDate": "2025-03-01"
  },
  ...
]
```

2. GET /customers/{id}

✓ Test Case: Get a customer by ID

Request:

bash

CopyEdit

```
curl -X GET http://localhost:8080/customers/1 \
-H "Authorization: Bearer <manager-token>"
```

Expected Response:

json

CopyEdit

```
{
  "id": 1,
  "fullName": "John Doe",
  "email": "john@example.com",
  "phone": "555-1234",
  "address": "123 Main St",
  "idType": "Driver License",
  "regDate": "2025-03-01"
}
```

3. POST /customers

✓ Test Case: Create a new customer

Note: This can also happen implicitly via bookings, but this is a direct creation route.

Request:

bash

CopyEdit

```
curl -X POST http://localhost:8080/customers \
-H "Content-Type: application/json" \
-d '{
  "fullName": "Alice Walker",
  "email": "alice@example.com",
  "phone": "555-9876",
  "address": "456 Elm St",
  "idType": "SSN",
  "regDate": "2025-04-01"
}'
```

Expected Response:

json

CopyEdit

```
{
  "id": 4,
  "fullName": "Alice Walker",
  "email": "alice@example.com",
  ...
}
```

4. PUT /customers/{id}

Test Case: Update a customer

Request:

bash

CopyEdit

```
curl -X PUT http://localhost:8080/customers/4 \
-H "Content-Type: application/json" \
-d '{
  "fullName": "Alice W. Walker",
```

```
"email": "alice@example.com",
"phone": "555-1111",
"address": "789 Oak Blvd",
"idType": "SSN",
"regDate": "2025-04-01"
}'
```

Expected Response:

json

CopyEdit

```
{
  "id": 4,
  "fullName": "Alice W. Walker",
  ...
}
```

5. DELETE /customers/{id}

 This may fail if foreign keys in bookings/rentings are not **ON DELETE SET NULL**.

Request:

bash

CopyEdit

```
curl -X DELETE http://localhost:8080/customers/4 \
-H "Authorization: Bearer <manager-token>"
```

Expected Response:

css

CopyEdit

204 No Content



EmployeeController – API Test Cases & Responses

Base route: `/employees`

1. POST /employees/login

✅ Test Case: Log in as an employee or manager

Request:

bash

CopyEdit

```
curl -X POST http://localhost:8080/employees/login \
-H "Content-Type: application/json" \
-d '{
  "email": "alice.manager@luxurystay.com",
  "password": "password123"
}'
```

Expected Response:

json

CopyEdit

```
"eyJhbGciOiJIUzI1NiJ9..."
```

A valid JWT token string.

2. GET /employees



ROLE_Manager only

✓ Test Case: Get all employees

Request:

bash

CopyEdit

```
curl -X GET http://localhost:8080/employees \
-H "Authorization: Bearer <manager-token>"
```

Expected Response:

json

CopyEdit

```
[
  {
    "id": 1,
    "fullName": "Alice Manager",
    "email": "alice.manager@luxurystay.com",
    "role": "Manager",
    "manager": true,
    "ssnNumber": "123-45-6789",
    "hotel": {
      "id": 1,
      "name": "Luxury Stay Downtown"
    }
  }
]
```

3. GET /employees/{id}

🔒 ROLE_Manager only

Request:

bash

CopyEdit

```
curl -X GET http://localhost:8080/employees/1 \
-H "Authorization: Bearer <manager-token>"
```

4. POST /employees

 **ROLE_Manager** only

 **Test Case: Create a new employee**

Request:

bash

CopyEdit

```
curl -X POST http://localhost:8080/employees \
-H "Authorization: Bearer <manager-token>" \
-H "Content-Type: application/json" \
-d '{
  "fullName": "Bob Clerk",
  "email": "bob.clerk@luxurystay.com",
  "password": "password123",
  "role": "Employee",
  "manager": false,
  "ssnNumber": "987-65-4321",
  "hotel": {
    "id": 1
  }
}'
```

Expected Response:

json

CopyEdit

```
{
  "id": 2,
  "fullName": "Bob Clerk",
```



```
...  
}
```

5. PUT /employees/{id}

 **ROLE_Manager** only

Request:

bash

CopyEdit

```
curl -X PUT http://localhost:8080/employees/2 \  
-H "Authorization: Bearer <manager-token>" \  
-H "Content-Type: application/json" \  
-d '{  
  "fullName": "Bob Clerk Jr.",  
  "email": "bob.clerk@luxurystay.com",  
  "password": "password123",  
  "role": "Employee",  
  "manager": false,  
  "ssnNumber": "987-65-4321",  
  "hotel": {  
    "id": 1  
  }  
'
```

6. DELETE /employees/{id}

 **ROLE_Manager** only

Request:

bash

CopyEdit

```
curl -X DELETE http://localhost:8080/employees/2 \
-H "Authorization: Bearer <manager-token>"
```

Expected Response:

css
CopyEdit
204 No Content



BookingController – API Test Cases & Responses

Base route: `/bookings`

1. GET /bookings/availability

✓ Test Case: Check if a room is available

Request:

bash
CopyEdit
curl -X GET
"http://localhost:8080/bookings/availability?roomId=1&checkInDate=2025-04-01&checkOutDate=2025-04-05"


Expected Response:

json
CopyEdit
true

or

json
CopyEdit
false

2. POST /bookings

 Public (no auth)

 **Test Case: Create a booking as a customer**

Request:

```
bash
CopyEdit
curl -X POST http://localhost:8080/bookings \
-H "Content-Type: application/json" \
-d '{
  "customer": {
    "fullName": "John Doe",
    "email": "john.doe@example.com",
    "phone": "555-1111",
    "address": "101 Booking Rd",
    "idType": "Driver License"
  },
  "roomId": 1,
  "checkInDate": "2025-04-01",
  "checkOutDate": "2025-04-05"
}'
```

Expected Response:

```
json
CopyEdit
{
  "id": 7,
  "bookingDate": "2025-03-25",
```

```
{
  "checkInDate": "2025-04-01",
  "checkOutDate": "2025-04-05",
  "status": "Pending",
  "room": {
    "id": 1,
    "roomNumber": "101"
  },
  "customer": {
    "id": 5,
    "fullName": "John Doe"
  }
}
```

3. GET /bookings

 Requires **ROLE_Manager** or **Employee**

 **Test Case: View all bookings**

Request:

bash

CopyEdit

```
curl -X GET http://localhost:8080/bookings \
-H "Authorization: Bearer <manager-or-employee-token>"
```

4. GET /bookings/customer/{email}

 Requires **ROLE_Manager** or **Employee**

Request:

bash

CopyEdit

```
curl -X GET
http://localhost:8080/bookings/customer/john.doe@example.com \
-H "Authorization: Bearer <manager-or-employee-token>"
```

Expected Response:

json

CopyEdit

```
[
  {
    "id": 7,
    "status": "Pending",
    "checkInDate": "2025-04-01",
    "checkOutDate": "2025-04-05"
  }
]
```

5. PUT /bookings/{id}/status

 Requires **ROLE_Manager** or **Employee**

 **Test Case: Update booking status**

Request:

bash

CopyEdit

```
curl -X PUT http://localhost:8080/bookings/7/status \
-H "Authorization: Bearer <manager-or-employee-token>" \
-H "Content-Type: application/json" \
-d '{"status": "Confirmed"}'
```

Expected Response:

json

CopyEdit

```
{  
  "id": 7,  
  "status": "Confirmed"  
}
```

RentingController – API Test Cases & Responses

Base route: `/rentings`

1. POST /rentings

 Requires `ROLE_Manager` or `Employee`

 Test Case: Rent a room (walk-in customer)

Request:

bash

CopyEdit

```
curl -X POST http://localhost:8080/rentings \  
-H "Authorization: Bearer <employee-token>" \  
-H "Content-Type: application/json" \  
-d '{  
  "roomId": 1,  
  "employeeId": 2,  
  "customer": {  
    "fullName": "Walk-In Guest",  
    "email": "walkin@example.com",  
    "phone": "555-2222",  
    "address": "Front Desk",
```

```
    "idType": "SIN"
  },
  "dateOut": "2025-04-06T12:00:00",
  "payment": 400.00
}'
```

Expected Response:

json

CopyEdit

```
{
  "id": 12,
  "dateIn": "2025-03-25T14:02:00",
  "dateOut": "2025-04-06T12:00:00",
  "payment": 400.0,
  "room": { "id": 1 },
  "customer": { "id": 10 },
  "employee": { "id": 2 },
  "booking": null
}
```

2. POST /rentings/from-booking

 Requires **ROLE_Manager** or **Employee**

 **Test Case:** Rent a room from an existing booking (check-in)

Request:

bash

CopyEdit

```
curl -X POST http://localhost:8080/rentings/from-booking \
-H "Authorization: Bearer <employee-token>" \
-H "Content-Type: application/json" \
-d '{
```

```
"bookingId": 7,  
"employeeId": 2,  
"dateOut": "2025-04-06T12:00:00",  
"payment": 500.00  
'}
```

Expected Response:

json

CopyEdit

```
{  
  "id": 13,  
  "booking": { "id": 7, "status": "Checked-In" },  
  "room": { "id": 1 },  
  "customer": { "id": 5 },  
  "employee": { "id": 2 },  
  "payment": 500.00  
}
```

3. GET /rentings

 Requires **ROLE_Manager**

 **Test Case: Get all rentings**

Request:

bash

CopyEdit

```
curl -X GET http://localhost:8080/rentings \  
-H "Authorization: Bearer <manager-token>"
```

Expected Response:

List of all active and historical rentings.

4. GET /rentings/customer/{email}

 Requires **ROLE_Manager** or **Employee**

 **Test Case: Get rentings by customer email**

Request:

bash

CopyEdit

```
curl -X GET
```

```
http://localhost:8080/rentings/customer/walkin@example.com \
-H "Authorization: Bearer <employee-token>"
```

5. PUT /rentings/{id}/checkout

 Requires **ROLE_Manager** or **Employee**

 **Test Case: Checkout a rental**

Request:

bash

CopyEdit

```
curl -X PUT http://localhost:8080/rentings/12/checkout \
-H "Authorization: Bearer <employee-token>"
```

Expected Response:

json

CopyEdit

```
{
  "id": 12,
  "dateOut": "2025-03-26T10:15:00",
  "booking": {
```

```
    "id": 7,  
    "status": "Checked-Out"  
  }  
}
```