

Room Sharing App

API Reference - Rooms Booking (sharing) App

Admin APIs

GET /admin/listUsers

- **Description:** Fetch all registered users (admin only)
- **Params:** `pass` (String) - admin password
- **Returns:** `List<UserModel>`
- **Status:** 200 OK or 401 Unauthorized

GET /admin/listRooms

- **Description:** Fetch all registered rooms (admin only)
- **Params:** `pass` (String) - admin password
- **Returns:** `List<RoomsModel>`
- **Status:** 200 OK or 401 Unauthorized

DELETE /admin/deleteUser/{userId}

- **Description:** Permanently delete a user
- **PathVariable:** `userId` (Integer)
- **Returns:** Success message

DELETE /admin/deleteRoom/{roomId}

- **Description:** Permanently delete a room
- **PathVariable:** `roomId` (Integer)
- **Returns:** Success message

User APIs

POST /users/add

- **Description:** Register a new user
- **Body:** UserModel
- **Returns:** Created UserModel

GET /users/{id}

- **Description:** Get a user by ID
- **PathVariable:** id (Integer)
- **Returns:** UserModel

GET /users/exists/{id}

- **Description:** Check if a user exists
 - **Returns:** true or false
-

Room APIs

POST /rooms/add

- **Description:** Add a new room
- **Body:** RoomsModel
- **Returns:** Saved RoomsModel

GET /rooms/roomDetails?id={id}

- **Description:** Get full room details (Room + Features + Reviews)
- **Returns:** RoomsDTO

GET /rooms/city?param={city}

- **Description:** Get all rooms in a city
- **Returns:** List<RoomsModel>

PUT /rooms/avl/{id}

- **Description:** Update room availability
 - **PathVariable:** id
 - **Body:** Boolean (true/false)
 - **Returns:** Success or failure message
-

Booking APIs

GET /bookings/user/{userId}

- **Description:** Get bookings by a specific user
- **Returns:** `List<BookingsModel>`

POST /bookings/book

- **Description:** Book a room
- **Body:** `BookingsModel`
- **Returns:** Booking confirmation or error if already booked

DELETE /bookings/cancel

- **Description:** Cancel a booking
- **Body:** `BookingsModel`

GET /bookings/admin/all?pass={adminPass}

- **Description:** Get all bookings (admin only)
 - **Returns:** `List<BookingsModel>`
-

Complaint APIs

POST /complaints/add

- **Description:** Add a complaint by a user about a room
 - **Params:** `userId`, `roomId`
 - **Body:** `ComplaintsModel`
 - **Returns:** Success message
-

Review APIs

POST /reviews/add

- **Description:** Add a review for a room
 - **Body:** `ReviewsModel`
 - **Returns:** Success message
-

Additions for Controller Enhancements

AdminController ->

BookingController ->

ComplaintsController ->

ReviewController ->

RoomsController ->
