# API Documentation

Base URL

http://localhost:5000/api

## Authentication and Authorization

All routes under /api/task are protected and require a valid JWT token in the request header.  
Include an Authorization header with the following format for protected routes:  
Authorization: Bearer <token>

## Endpoints

### User Authentication

#### 1. Sign Up

URL: /api/auth/signUp

Method: POST

Description: Register a new user.

Request Body:

{  
 "firstName": "Moaz",  
 "lastName": "Adly",  
 "email": "john@example.com",  
 "password": "Password123"  
}

Response:

- 201 Created if registration is successful.

- 400 Bad Request if the email already exists or if there is missing information.

#### 2. Login

URL: /api/auth/login

Method: POST

Description: Log in an existing user.

Request Body:

{  
 "email": "john@example.com",  
 "password": "Password123"  
}

Response:

- 200 OK with JWT token and user data if credentials are correct.

- 404 Not Found if email or password is incorrect.

### Task Management

#### 1. Get All Tasks for a User

URL: /api/task

Method: GET

Description: Retrieve all tasks for the authenticated user.

Headers: Authorization: Bearer <token>

Response:

- 200 OK with a list of tasks.

- 404 Not Found if the user is not found.

#### 2. Create Task

URL: /api/task

Method: POST

Description: Create a new task for the authenticated user.

Headers: Authorization: Bearer <token>

Request Body:

{  
 "title": "New Task",  
 "description": "Task details",  
 "completed": false,  
 "deadline": "2024-12-31T00:00:00Z"  
}

Response:

- 201 Created with the created task details.

- 404 Not Found if the user is not found.

#### 3. Edit Task

URL: /api/task/:taskId

Method: PUT

Description: Update an existing task for the authenticated user.

Headers: Authorization: Bearer <token>

Request Body: (optional fields to update)

{  
 "title": "Updated Task Title",  
 "description": "Updated task details",  
 "completed": true,  
 "deadline": "2024-12-31T00:00:00Z"  
}

Response:

- 200 OK with updated task details.

- 404 Not Found if the task is not found.

#### 4. Delete Task

URL: /api/task/:taskId

Method: DELETE

Description: Delete a specific task for the authenticated user.

Headers: Authorization: Bearer <token>

Response:

- 200 OK if deletion is successful.

- 404 Not Found if the task is not found.

## Error Codes

- 400 Bad Request: Validation errors or bad data in the request.

- 401 Unauthorized: Missing or invalid JWT token.

- 404 Not Found: Resource not found, such as user or task.

- 500 Internal Server Error: Unexpected server error.