**API Documentation for Project Management System**

**1. User Authentication**

**1.1 User Registration**

**POST** /api/register

* **Header**: Content-Type: application/json
* **Description**: Registers a new user with email, password, and name.
* **Request Body**:

{

"email": "user@example.com",

"password": "password123",

}

* **Response**:
  + **Success** (201 Created):

{

"message": "User registered successfully."

}

* **Failure** (409 Conflict):

{

"error": "Email already exists."

}

**1.2 User Login**

**POST** /api/login

* **Description**: Logs in an existing user with email and password and returns a JWT token.
* **Request Body**:
* {
* "email": "user@example.com",
* "password": "password123",
* }
* **Response**:
  + **Success** (200 OK):

{

"token":"eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJ5b3VyX2lzc3VlciIsImF1ZCI6InlvdXJfYXVkaWVuY2UiLCJpYXQiOjE3NDQyODQ1OTcsImV4cCI6MTc0NDI4ODE5NywiZGF0YSI6eyJpZCI6IjEiLCJlbWFpbCI6InRlc3RAZXhhbXBsZS5jb20ifX0.bOpd41C\_7ndo2pa6nUKGNTWPyvq1JQGELM\_HpPjjvyk"

}

* **Failure** (401 Unauthorized):

{

"error": "Invalid credentials."

}

**2. Projects**

**2.1 Create Project**

**POST** /api/projects/create

* **Description**: Creates a new project for the authenticated user.
* **Request Headers**:

Bearer <JWT\_TOKEN>

* + Authorization: Bearer <eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJ5b3VyX2lzc3VlciIsImF1ZCI6InlvdXJfYXVkaWVuY2UiLCJpYXQiOjE3NDQyODQ1OTcsImV4cCI6MTc0NDI4ODE5NywiZGF0YSI6eyJpZCI6IjEiLCJlbWFpbCI6InRlc3RAZXhhbXBsZS5jb20ifX0.bOpd41C\_7ndo2pa6nUKGNTWPyvq1JQGELM\_HpPjjvyk>
* **Request Body**:

{

"title": "My First Project",

"description": "This is a test project."

}

* **Response**:
  + **Success** (201 Created):

{

"message": "Project created successfully.",

"project\_id": 1

}

**Failure** (400 Bad Request):

{

"error": "Missing required fields."

}

**2.2 Update Project**

**PUT** /api/projects/update/1 { project\_id}

* **Description**: Update a project for the authenticated user.
* **Request Headers**:
  + Authorization: Bearer <eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJ5b3VyX2lzc3VlciIsImF1ZCI6InlvdXJfYXVkaWVuY2UiLCJpYXQiOjE3NDQyODQ1OTcsImV4cCI6MTc0NDI4ODE5NywiZGF0YSI6eyJpZCI6IjEiLCJlbWFpbCI6InRlc3RAZXhhbXBsZS5jb20ifX0.bOpd41C\_7ndo2pa6nUKGNTWPyvq1JQGELM\_HpPjjvyk>
* **Request Body**:

{

"title": "My First Projectfdf",

"description": "This is a test project."

}

* **Response**:
  + **Success** (200 OK):

{"message":"Project updated"}

**Failure** (400 Bad Request):

{

"error": "Missing required fields."

}

**2.3 Delete Project**

**DELETE**  /api/projects/delete/1 { project\_id}

* **Description**: Delete ptojetcs
* **Request Headers**:
  + Authorization: Bearer <eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJ5b3VyX2lzc3VlciIsImF1ZCI6InlvdXJfYXVkaWVuY2UiLCJpYXQiOjE3NDQyODQ1OTcsImV4cCI6MTc0NDI4ODE5NywiZGF0YSI6eyJpZCI6IjEiLCJlbWFpbCI6InRlc3RAZXhhbXBsZS5jb20ifX0.bOpd41C\_7ndo2pa6nUKGNTWPyvq1JQGELM\_HpPjjvyk>
* **Response**:
  + **Success** (200 OK):

{"message":"Project deleted"}

**2.4 GET Project**

**GET** /api/projects

**Description**: Get ptojetcs

* **Request Headers**:
  + Authorization: Bearer <eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJ5b3VyX2lzc3VlciIsImF1ZCI6InlvdXJfYXVkaWVuY2UiLCJpYXQiOjE3NDQyODQ1OTcsImV4cCI6MTc0NDI4ODE5NywiZGF0YSI6eyJpZCI6IjEiLCJlbWFpbCI6InRlc3RAZXhhbXBsZS5jb20ifX0.bOpd41C\_7ndo2pa6nUKGNTWPyvq1JQGELM\_HpPjjvyk>
* **Response**:
  + **Success** (200 OK):
* [{"id":"2","user\_id":"1","title":"fsjhfdjsdf","description":"","created\_at":"2025-04-10
* 17:21:30","updated\_at":"2025-04-10 17:21:30"}]

**3. Task**

**3.1** Create Task

**POST** /api/tasks/create

**Description**: Creates a new task for the authenticated user.

* **Request Headers**:
  + Authorization: Bearer <eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJ5b3VyX2lzc3VlciIsImF1ZCI6InlvdXJfYXVkaWVuY2UiLCJpYXQiOjE3NDQyODQ1OTcsImV4cCI6MTc0NDI4ODE5NywiZGF0YSI6eyJpZCI6IjEiLCJlbWFpbCI6InRlc3RAZXhhbXBsZS5jb20ifX0.bOpd41C\_7ndo2pa6nUKGNTWPyvq1JQGELM\_HpPjjvyk>
* **Request Body**:

{

"project\_id": 1,

  "title": "Write API documentation",

  "description": "Document all API endpoints for the project."

}

* **Response**:
  + **Success** (201 Created):

{"message":"Task created successfully.","task\_id":1}

* **Failure** (400 Bad Request):

{

"error": "Project not found."

}

**Failure** (403 Forbidden):

{"message":"Access denied to this project"}

**3.2** Update Task

**PUT**  api/tasks/status/1 {task\_id}

**Description**: updates task id for the authenticated user.

* **Request Headers**:
  + Authorization: Bearer <eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJ5b3VyX2lzc3VlciIsImF1ZCI6InlvdXJfYXVkaWVuY2UiLCJpYXQiOjE3NDQyODQ1OTcsImV4cCI6MTc0NDI4ODE5NywiZGF0YSI6eyJpZCI6IjEiLCJlbWFpbCI6InRlc3RAZXhhbXBsZS5jb20ifX0.bOpd41C\_7ndo2pa6nUKGNTWPyvq1JQGELM\_HpPjjvyk>
* **Request Body**:

{

"status": "In Progress"

}

**Response**:

* + **Success** (200 Ok):

{

"message":"Task status updated"

}

**Failure** (403 Forbidden):

{"message":"Access denied"}

**Failure** (400 Bad Request):

{

"error": "Invalid status."

}

**3.3** Add Remark

**POST**  api/tasks/remark/1 {task\_id}

**Description**: Add Remark for the created tasks.

* **Request Headers**:
  + Authorization: Bearer <eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJ5b3VyX2lzc3VlciIsImF1ZCI6InlvdXJfYXVkaWVuY2UiLCJpYXQiOjE3NDQyODQ1OTcsImV4cCI6MTc0NDI4ODE5NywiZGF0YSI6eyJpZCI6IjEiLCJlbWFpbCI6InRlc3RAZXhhbXBsZS5jb20ifX0.bOpd41C\_7ndo2pa6nUKGNTWPyvq1JQGELM\_HpPjjvyk>
* **Request Body**:

{

"remark": "Worked on JWT and validation.",

"remark\_date": "2025-04-10"

}

**Response**:

* + **Success** (201 Created):

{"message":"Remark added"}

**Failure** (400 Bad Request):

{

"error": "Task not found."

}

**3.4** View Task Remarks

**GET**  api/tasks/remarks/1 {task\_id}

**Description**: View Task Remarks

* **Request Headers**:
  + Authorization: Bearer <eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJ5b3VyX2lzc3VlciIsImF1ZCI6InlvdXJfYXVkaWVuY2UiLCJpYXQiOjE3NDQyODQ1OTcsImV4cCI6MTc0NDI4ODE5NywiZGF0YSI6eyJpZCI6IjEiLCJlbWFpbCI6InRlc3RAZXhhbXBsZS5jb20ifX0.bOpd41C\_7ndo2pa6nUKGNTWPyvq1JQGELM\_HpPjjvyk>

**Response**:

* + **Success** (200 Ok):

[{"id":"1","task\_id":"2","remark":"Worked on JWT and validation.","remark\_date":"2025-04-10","created\_at":"2025-04-10 18:41:11"}]

**Failure** (403 Forbidden):

{"message":"Access denied"}

**4. Get Task Reports for a Project**

**4.1** Task Reports

**GET** /api/tasks/report/1 {project\_id}

**Description**: Get Task Report for a Project.

* **Request Headers**:
  + Authorization: Bearer <eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJ5b3VyX2lzc3VlciIsImF1ZCI6InlvdXJfYXVkaWVuY2UiLCJpYXQiOjE3NDQyODQ1OTcsImV4cCI6MTc0NDI4ODE5NywiZGF0YSI6eyJpZCI6IjEiLCJlbWFpbCI6InRlc3RAZXhhbXBsZS5jb20ifX0.bOpd41C\_7ndo2pa6nUKGNTWPyvq1JQGELM\_HpPjjvyk>
* **Response**:
  + **Success** (200 Ok):

[{"id":"2","project\_id":"3","title":"Write API documentation","description":null,

"status":"In Progress","created\_at":"2025-04-10 18:19:28","updated\_at":"2025-04-10 18:21:29","status\_history":[{"id":"1","task\_id":"2","status":"In Progress","changed\_at":"2025-04-10 14:51:29"},{"id":"2","task\_id":"2","status":"In Progress","changed\_at":"2025-04-10 14:52:58"}],"remarks":[{"id":"1","task\_id":"2","remark":"Worked on JWT and validation.","remark\_date":"2025-04-10","created\_at":"2025-04-10 18:41:11"}]},{"id":"3","project\_id":"3","title":"Write API documentation","description":null,"status":"In Progress","created\_at":"2025-04-10 18:35:33","updated\_at":"2025-04-10 18:36:44","status\_history":[{"id":"3","task\_id":"3","status":"In Progress","changed\_at":"2025-04-10 15:06:44"}],"remarks":[]}]

**Failure** (404 Not Found):

{

"error": "Project not found."

}

**Failure** (403 Forbidden):

{"message":"Access denied"}

***API url:***

1. **POST** /api/register

2. **POST** /api/login

3. **POST** /api/projects/create

4. **PUT** /api/projects/update/1 { project\_id}

**5. DELETE**  /api/projects/delete/1 { project\_id}

**6. GET** /api/projects

**7. POST** /api/tasks/create

**8. PUT**  api/tasks/status/1 {task\_id}

**9. POST**  api/tasks/remark/1 {task\_id}

**10. GET**  api/tasks/remarks/1 {task\_id}

**11. GET** /api/tasks/report/1 {project\_id}