

## Task Management Application with Task Completion Report

We are enhancing the Task Management Application by allowing users to **submit a short report and their worked hours when they complete a task**. This will help track what was done on each task and how much time it took.

- When users **mark a task as completed**, they must also submit a **Completion Report** and **Worked Hours**.
- These reports will be visible to **Admins** and **SuperAdmins** for review and monitoring.
- This feature makes task tracking more transparent and improves accountability.

**Develop API endpoints and an Admin Panel utilizing Python and Django to enhance task management functionality.**

### 1. API Endpoints **(Create endpoints only, no need to integrate ):**

#### ❖ User Authentication:

- **JWT Authentication:** Implement JSON Web Token (JWT) authentication for the **User API** to ensure secure access.
  - Users can authenticate via username and password, receiving a JWT token for further requests.

#### ❖ Tasks APIs:

- **GET /tasks:** Fetch all tasks assigned to the logged-in user.
  - Only returns tasks for the logged user who is sending the request.
- **PUT /tasks/{id}:** Allows users to update the status of a task (*mark it as Completed*).
  - When setting a task status to Completed, users are required to submit a **Completion Report** and the **number of Worked Hours**.
- **GET /tasks/{id}/report:** Admins and SuperAdmins can view the **Completion Report** and **Worked Hours** for a specific task.
  - This is only available for tasks that are marked as **Completed**.

### 2. Admin Panel (Web Application for Admin and SuperAdmin – **Create your own admin panel using html templates.**):

- **SuperAdmin Features:**
  - Can view and manage all users (create, delete, assign roles).
  - Can view and manage all admin(create, delete, assign roles).

- Can assign user to the admin
- Can view and manage all tasks across users.
- Can view task reports submitted by users.
- **Admin Features:**
  - Admin can assign tasks to their users.
  - Can view and manage tasks assigned to their users.
  - Can view the completion reports submitted by users (including worked hours).
  - Cannot manage user roles but can manage tasks within their scope.

### 3.Task Workflow

#### 1. Roles and Permissions:

- **SuperAdmin:**
  - Can manage admin(create, delete, assign roles, and promote/demote admin).
  - Can manage users (create, delete,update).
  - Can access the full Admin panel with all privileges.
- **Admin:**
  - Can create, assign, view, and manage tasks.
  - Can view task completion reports for tasks assigned to users.
  - Has limited access to the Admin panel: Can manage tasks but not users.
- **User:**
  - Can view their assigned tasks, update their task status, and submit a completion report (including worked hours).
  - Can only interact with their own tasks through the User API.

#### 2. Task Model:

- **Fields:**
  - **Title:** The name of the task.
  - **Description:** A detailed description of the task.
  - **Assigned To:** The user to whom the task is assigned.
  - **Due Date:** The task's deadline.
  - **Status:** The current task status (Pending, In Progress, Completed).
  - **Completion Report:** A text field for the user to provide a report when marking the task as completed.
  - **Worked Hours:** Numeric field to track hours worked on the task, provided when marking the task as completed.

### 3. Completion Report and Worked Hours:

- When a user marks a task as **Completed** via the API or Admin panel, they must provide:
  - **Completion Report:** A brief description of what was done, challenges faced, and other relevant details.
  - **Worked Hours:** A numeric field where the user specifies how many hours they worked on the task.
- Admin and SuperAdmin users can view these reports for completed tasks in the Admin panel and via the API.

**Note:** Use SQLite as the database, and the [requirements.txt](#) must include.  
Read the task carefully. If you have any doubts, feel free to ask at any time.