**PROJECT MANAGEMENT SYSTEM-DOCUMENTATION**

1. **User Registration and Login**
2. **Project Management**
3. **Task Management**
4. **Task Status & Remarks**
5. **Reports**

Everything will be done through server-side routing via query parameters or action types — all handled within PHP files.

**User Authentication**

**register.php**

* Form to collect email, password.
* Hash password using password\_hash().
* Save into users table.

**login.php**

* Verify with password\_verify().
* On success, start a session and store the user ID in $\_SESSION['user\_id'].

**logout.php**

* Just destroy the session.

**Project Management**

**projects.php**

* **List user projects**
* Use $\_SESSION['user\_id'] to show only their projects.

**create\_project.php**

* Form for project name & description.
* Insert into projects table linked to user\_id.

**edit\_project.php?id=...**

* Fetch project by id and populate form for editing.
* On submit, update the DB.

**delete\_project.php?id=...**

* Delete the project and related tasks.

**Task Management**

**create\_task.php?project\_id=...**

* Form for task title, description, initial status.
* Insert into tasks table linked to project\_id.

**edit\_task.php?id=...**

* Allow updating task title, description, or status.

**delete\_task.php?id=...**

* Delete the task and related remarks & status history.

**Update Task Status + Remarks**

**update\_status.php?task\_id=...**

* Form to:  
    
  + Change task status (dropdown)
  + Add a remark
* Insert into:  
    
  + task\_status\_history
  + task\_remarks

**Project Report**

**report.php?project\_id=...**

* Fetch all tasks under a project.
* For each task:  
    
  + List all status changes (from task\_status\_history)
  + List all daily remarks (from task\_remarks)

**Folder Structure (Recommended)**

bash

CopyEdit

/project-root

│

├── db.php                # DB connection

├── session.php           # Auth/session check

├── register.php

├── login.php

├── logout.php

│

├── projects.php

├── create\_project.php

├── edit\_project.php

├── delete\_project.php

│

├── create\_task.php

├── edit\_task.php

├── delete\_task.php

├── update\_status.php

│

├── report.php

└── ...

**DB Tables (Simplified)**

CREATE DATABASE IF NOT EXISTS project\_manager;

USE project\_manager;

CREATE TABLE users (

    id INT AUTO\_INCREMENT PRIMARY KEY,

    firstname VARCHAR(50) NOT NULL,

    lastname VARCHAR(50) NOT NULL,

    email VARCHAR(100) UNIQUE NOT NULL,

    password VARCHAR(255) NOT NULL,

    created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

CREATE TABLE projects (

    id INT AUTO\_INCREMENT PRIMARY KEY,

    user\_id INT NOT NULL,

    name VARCHAR(100) NOT NULL,

    description TEXT,

    created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

    FOREIGN KEY (user\_id) REFERENCES users(id) ON DELETE CASCADE,

    INDEX(user\_id)

);

CREATE TABLE tasks (

    id INT AUTO\_INCREMENT PRIMARY KEY,

    project\_id INT NOT NULL,

    title VARCHAR(100) NOT NULL,

    description TEXT,

    status ENUM('Pending', 'In Progress', 'Completed') DEFAULT 'Pending',

    created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

    FOREIGN KEY (project\_id) REFERENCES projects(id) ON DELETE CASCADE,

    INDEX(project\_id),

    INDEX(status)

);

CREATE TABLE task\_remarks (

    id INT AUTO\_INCREMENT PRIMARY KEY,

    task\_id INT NOT NULL,

    remark TEXT NOT NULL,

    remark\_date DATE NOT NULL DEFAULT CURRENT\_DATE,

    created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

    FOREIGN KEY (task\_id) REFERENCES tasks(id) ON DELETE CASCADE,

    INDEX(task\_id),

    INDEX(remark\_date)

);

CREATE TABLE task\_status\_history (

    id INT AUTO\_INCREMENT PRIMARY KEY,

    task\_id INT NOT NULL,

    status ENUM('Pending', 'In Progress', 'Completed'),

    changed\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

    FOREIGN KEY (task\_id) REFERENCES tasks(id) ON DELETE CASCADE,

    INDEX(task\_id),

    INDEX(status),

    INDEX(changed\_at)

);