# Advanced Task Management Panel - Project Requirements

## 1. Project Overview

Develop a Task Management System using Python Flask where users can create, assign, update, and track tasks. The system will support role-based access control (Admin & User) and allow users to attach files to tasks. A password reset feature via email will also be included.

## 2. Core Features

### 2.1 User Management

✔ User Registration (Sign Up)  
✔ User Login  
✔ User Logout  
✔ Password Reset via Email  
✔ Role Management:  
 - Admin: Manage all users and tasks  
 - User: Manage only their own tasks

### 2.2 Task Management

✔ Create new tasks  
✔ List all tasks  
✔ Update task status (To Do / In Progress / Completed)  
✔ Delete tasks  
✔ Attach files to tasks (PDF, PNG, JPG, etc.)

## 3. Technical Requirements

### 3.1 Backend (Server-Side)

✔ Python Flask (for REST API)  
✔ Flask-SQLAlchemy (for database management)  
✔ Flask-Login (for user authentication)  
✔ Flask-Mail (for email notifications, using SMTP)  
✔ Flask-Uploads (for file handling)  
✔ SQLite (Development), PostgreSQL/MySQL (Optional for Production)

### 3.2 Frontend (User Interface)

✔ HTML, CSS, JavaScript  
✔ Bootstrap or Tailwind CSS  
✔ Jinja2 Template Engine

## 4. Usage Scenario

1. A user registers and logs into the system.  
2. If they forget their password, they can reset it via email.  
3. Users can create, update, and manage their tasks.  
4. Users can upload files related to their tasks.  
5. Admin users can:  
 - Manage users (delete, change roles, reset passwords, etc.)  
 - View and manage all tasks.

## 5. Development & Deliverables

✔ A GitHub repository will be created for version control.  
✔ A README.md file will be included with setup and usage instructions.  
✔ The source code will be well-documented with comments.