Project Title:

Complaint Management System

Project Description:

The Complaint Management System (CMS) is a web-based platform designed to help organizations, universities, or institutions manage and respond to user complaints efficiently.

It allows users to register complaints, track their status, and view responses from administrators. Administrators can review, assign, update, or resolve complaints in a structured and transparent way.

The system reduces manual paperwork, eliminates confusion caused by untracked issues, and provides a record of complaint handling for accountability and performance review.

Problem Statement:

In many organizations, complaints or issues are reported manually through emails, paper forms, or verbal communication. This often leads to delays in resolving issues, lost or ignored complaints, lack of tracking or accountability, and frustration among users who never receive updates.

Therefore, there is a need for a centralized, automated complaint management platform that keeps a transparent record of all reported issues and their resolutions.

Proposed Solution:

The Complaint Management System will be a Python Flask-based web application where users can log in and register complaints, while admins can manage them effectively.

Core Features:

- User Panel:

1. Register and log in to their account.
2. Submit a complaint with category, description, and optional attachments.
3. View the status of submitted complaints (Pending, In Progress, Resolved).
4. Receive updates when status changes.

- Admin Panel:

5. View all submitted complaints.
6. Assign complaints to relevant staff members or mark them as 'In Progress.'

7. Update complaint status to 'Resolved' after completion.
8. Send feedback or comments to the user.
9. Generate simple reports or summaries of complaint types and resolution rates.

- Additional Features:

10. Email or in-app notifications for status updates.
11. Simple dashboard showing complaint statistics.
12. Role-based authentication (User/Admin).

Technology Stack:

Component	Tool / Technology
Frontend	HTML, CSS, Bootstrap
Backend	Python (Flask framework)
Database	SQLite (or MySQL)
Version Control	Git + GitHub
Development Environment	VS Code
Optional Libraries	Flask-Login (for authentication), Flask-Mail (for notifications), SQLAlchemy (for ORM), Flask-WTF (for forms)