Django Feedback System- Step-by-Step Implementation

Step 1: Set Up the Project in VS Code

Install Prerequisites

Ensure that Python and pip are installed.

Install Virtual Environment

pip install virtualenv

Create a Virtual Environment

mkdir feedback_project && cd feedback_project python -m venv venv

Activate the Virtual Environment

• Windows: venv\Scripts\activate

Mac/Linux: source venv/bin/activate

Install Django and Start the Project

pip install django django-admin startproject feedback system.

Step 2: Create the Feedback App

Create a Django App

python manage.py startapp feedback

Register the App in settings.py

```
INSTALLED_APPS = [
   'django.contrib.admin',
   'django.contrib.auth',
   'django.contrib.contenttypes',
   'django.contrib.sessions',
   'django.contrib.messages',
   'django.contrib.staticfiles',
   'feedback', # New app
]
```

Step 3: Define the Feedback Model

Define Model (models.py in feedback app)

Apply Migrations

python manage.py makemigrations python manage.py migrate

Step 4: Create the Feedback Form

Define Form (forms.py in feedback app)

Step 5: Create Views

Define Views (views.py in feedback app)

Step 6: Set Up URLs

Define App URLs (urls.py in feedback app)
Include App URLs in feedback_system/urls.py

Step 7: Create the Template

Create feedback.html in feedback/templates/feedback/

Step 8: Add CSRF Protection

Django includes CSRF protection by default. The {% csrf_token %} in the form ensures security.

Step 9: Add Feedback to Admin Panel

Register Model in admin.py

Create Superuser and Access Admin Panel

python manage.py createsuperuser

Visit /admin/ to manage feedback.

Final Steps

Run the Server

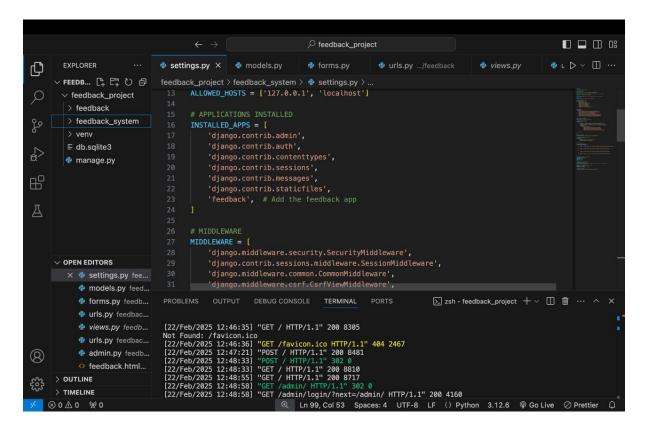
python manage.py runserver

Access the Application

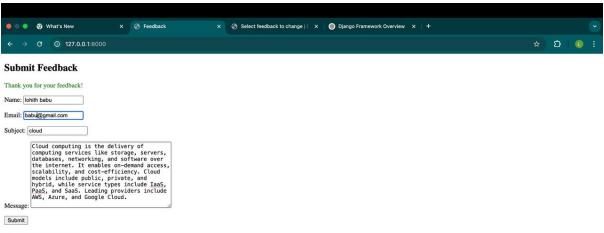
• Visit: <u>http://127.0.0.1:8000/</u>

• Visit: http://127.0.0.1:8000/admin/feedback/feedback/

IMPLEMENTED CODE



WFB PAGE



- **Previous Feedback**
 - harshith: The web, or World Wide Web, is a global system of interconnected documents and resources accessed via the internet. It enables users to browse websites, share information, and communicate online. The web operates through browsers, HTTP, and hyperlinks, facilitating instant access to multimedia content and online services worldwide.

 Lohith: Django is a high-level Python web framework that promotes rapid development and clean, pragmatic design. It follows the Model-View-Template (MVT) architecture and includes built-in features like authentication, ORM, and admin panel. Django emphasizes security, scalability, and reusability, making it ideal for building robust web applications efficiently.

ADMIN PAGE

