**Donation System**

**Practical Assessment**

**Sofin Wadhwaniya**

**Overview**

This project is an assessment task which is a Django-based donation system that allows users to make secure donations via an online payment gateway. Users authenticate using OTP via Fast2SMS, and all API endpoints are protected using token-based authentication. The system includes a donation history feature, data visualization, and is fully containerized using Docker.

**Features**

* **User Authentication**: Phone number + OTP authentication using Fast2SMS.
* **Authentication**: Secure API access with token-based authentication.
* **Donation Processing**: Users can donate via an integrated payment gateway (Razorpay).
* **Payment History**: Users can view their donation history.
* **Data Visualization**: A dashboard displaying monthly donation statistics.
* **Dockerized Deployment**: Runs in a Docker container.
* **Filtering & Sorting**: Filter donations based on amount, date range, and transaction ID.

**Technologies Used**

* **Backend**: Django, Django REST Framework (DRF)
* **Authentication**: Token Based, Fast2SMS for OTP
* **Database**: SQLite
* **Payment Gateway**: Razorpay
* **Containerization**: Docker & Docker Compose
* **Data Visualization**: Matplotlib & Pandas

**Installation**

**1. Clone the Repository**

git clone https://github.com/So-fin/Donation-System.git

cd donation-system

**2. Install Dependencies**

Create a virtual environment and install dependencies:

python3 -m venv env

source env/bin/activate

pip install -r requirements.txt

**3. Run Migrations**

python manage.py migrate

**4. Start the Server**

python manage.py runserver

The application will be available at http://127.0.0.1:8000/.

**Running with Docker**

docker-compose up --build

The app will be running on http://localhost:8000/.

**API Endpoints**

* POST /api/donation/send-otp/ - Register user with phone number (sends otp)
* POST /api/donation/login/ - Login with phone number and OTP & generate token.
* POST /api/donation/razorpay-donation-order/ - Make a donation (requires token)
* POST /api/donation/ create-order-webhook/ - Razorpay Webhook
* GET /api/donation/ - View donation history (requires token)
* GET /api/donation/donation-history/?amount\_min=&amount\_max=&date\_from=&date\_to= - Filter donations (requires token)
* GET /api/donation/donation-visualization/ - View donation analytics (requires token)

**Data Visualization**

* A **monthly donations dashboard** is available.

**Architecture Overview**

1. **User Authentication**: Users authenticate via OTP (Fast2SMS) and receive a token.
2. **Donation Processing**: Secure transactions through Razorpay.
3. **Payment History & Analytics**: Users can track their donations and visualize trends.
4. **Docker Deployment**: Simplifies installation and deployment across environments.
5. **Filtering & Sorting**: Users can filter donations based on amount and date.