

Influencer Engagement and Sponsorship Coordination Platform

Name: Pooja Kumari

Roll no.: 22f1000470

Email: 22f1000470@ds.study.iitm.ac.in

I am an IITM BS student currently in the diploma level

Description

I have approached the given problem statement in the following 3 phases:

Phase 1: Finalizing the necessary information required for the proper and smooth functioning of the web app. Thereafter, designing a suitable database structure to cater for the information storage.

Phase 2: Creating the basic framework with the routes and functions implemented.

Phase 3: Styling and Aesthetics. Mainly focused on polishing the web pages to look cleaner and more user-friendly.

Modules / Technologies Used

- **Flask:** The entire web application is built on the Flask framework.
- **Jinja2 Template:** Webpages of the app follow templating through Jinja2 templating.
- **SQLite3:** Database created is of SQLite3 structure.
- **Bootstrap:** Basic styling for the web pages.
- **Flask-SQLAlchemy:** For creating the relational database used in the app.
- **Session:** For providing basic login and logout features in the app.
- **Matplotlib.pyplot:** For creating graphs in the Admin dashboard.

Database Design

Tables used in the Database:

- **User:** Stores information for Admin, Sponsors, and Influencers.
- **Campaign:** Stores details of campaigns created by sponsors.
- **AdRequest:** Tracks ad requests between campaigns and influencers.
- **Req_from_inf:** Stores requests sent to sponsors by influencers .

Many-to-Many Relationship: Used in the tables Campaigns and Influencers via the AdRequest table, since one campaign can have multiple influencers, and one influencer can participate in multiple campaigns.

Architecture

The project folder consists of 3 parts:

- **Template folder:** Consists of the HTML files.
- **Instance folder:** Database is generated and stored in the instance folder.
- **api.py file:** Routes for API and JSONified outputs.
- **Main.py:** Controllers and Tables formed in this.
- **Requirement.txt:** Contains name of all the requirements to be downloaded.

How to use the application

After extracting the zip folder, download the requirement.txt file and run `app.py`.

Features

Core Features:

- **User login:** Login form for Admin, Sponsor, and Influencer.
- **Campaign Management:** Sponsors can create, update, and delete campaigns.
- **Ad Request Management:** Sponsors can create, edit, and delete ad requests.
- **Influencer Profile Management:** Influencers can update their profiles and manage their received ad requests.
- **Search for Influencers/Campaigns:** Sponsors can search for influencers, and influencers can search for public campaigns.
- **Admin Dashboard:** Admin can monitor users, campaigns, ad requests, and flagged content.

Other Features:

- **API:** URL endpoints are defined for getting JSONified outputs for different request methods using Postman.
- **Form Validation:** Both success and error messages with suitable descriptions are added using Jinja.
- **Bootstrap:** Used for minor styling of the web application.
- **Login Framework:** A proper session-based login framework has been used to create new accounts for users, and a login page is also formed.

Drive Link for Presentation:

https://drive.google.com/file/d/11x4-yGAeewd3iMPRw_Rke0lZJWgnxrX/view?usp=drivesdk