Author

Name: Balivada Sri Charitha
Roll number: 21f2000491
Email: 21f2000491@ds.study.iitm.ac.in

About: I am a recent graduate with a B.Tech degree in Electronics and communications Engineering, currently dedicated to pursuing a BS degree in Data Science and Applications at IITM. Fueled by youthful enthusiasm and curiosity. I am committed to expanding my knowledge and skills in these domains.

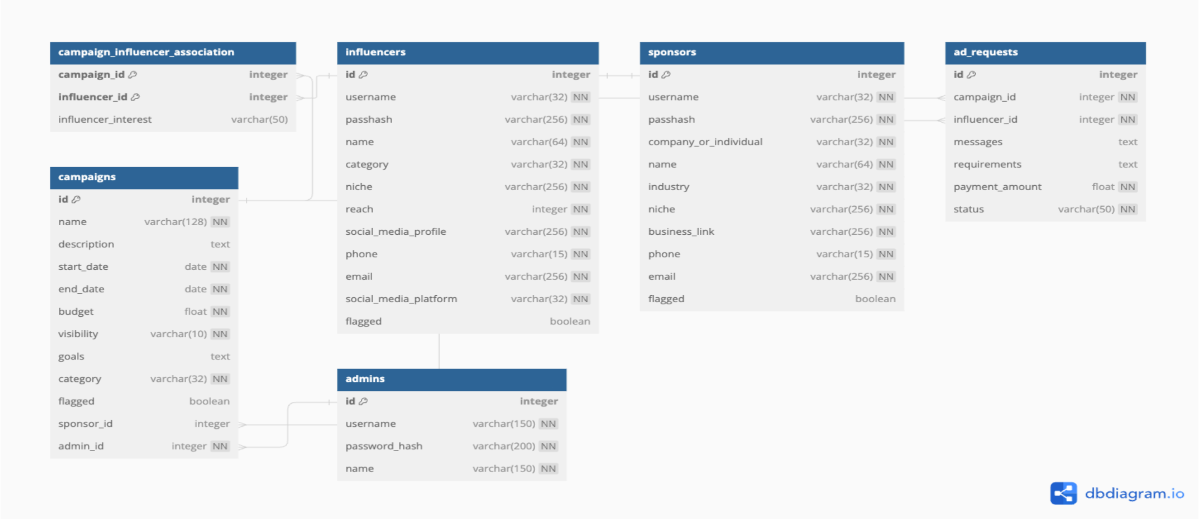
Description

The application CollabSphere is an all-inclusive campaign management and analysis platform for Sponsors and Influencers. The system provides a user-friendly interface for admins to track active campaigns, see detailed statistics, and control effective coordination between Sponsors and Influencers. Sponsors can create campaigns and send ad requests for campaigns to influencers with a defined payment amount. Influencers can track ad requests and review them, track campaign visibility, and ensure that campaign management is smooth and effective.

Technologies used

- Python (Used for creating the backend)
  - Flask (Used to create the web application)
  - Flask-SQLAlchemy (Used for interacting with the SQLite database)
  - Jinja2 (Used for templating in Flask)
- SQLite (Used for storing data)
- HTML5 (Used to create a structure for the front end)
- CSS3 (Used for custom styling)
- Bootstrap (Used for basic styling of the webpages)
- Matplotlib (Used for creating charts)

Database Schema Design



campaign_influencer_association		
campaign_id	integer	
influencer_id	integer	
influencer_interest	varchar(50)	

campaigns		
id	integer	
name	varchar(128)	NN
description	text	
start_date	date	NN
end_date	date	NN
budget	float	NN
visibility	varchar(10)	NN
goals	text	
category	varchar(32)	NN
flagged	boolean	
sponsor_id	integer	
admin_id	integer	NN

influencers		
id	integer	
username	varchar(32)	NN
passhash	varchar(256)	NN
name	varchar(64)	NN
category	varchar(32)	NN
niche	varchar(256)	NN
reach	integer	NN
social_media_profile	varchar(256)	NN
phone	varchar(15)	NN
email	varchar(256)	NN
social_media_platform	varchar(32)	NN
flagged	boolean	

admins		
id	integer	
username	varchar(150)	NN
password_hash	varchar(200)	NN
name	varchar(150)	NN

sponsors		
id	integer	
username	varchar(32)	NN
passhash	varchar(256)	NN
company_or_individual	varchar(32)	NN
name	varchar(64)	NN
industry	varchar(32)	NN
niche	varchar(256)	NN
business_link	varchar(256)	NN
phone	varchar(15)	NN
email	varchar(256)	NN
flagged	boolean	

ad_requests		
id	integer	
campaign_id	integer	NN
influencer_id	integer	NN
messages	text	
requirements	text	
payment_amount	float	NN
status	varchar(50)	NN

The database for CollabSphere is structured to support a platform where an Admin, along with two types of users "Sponsor" and "Influencer" interact with campaigns. Sponsors can create campaigns, and influencers can participate in these campaigns. The Admin and Sponsor tables have a one-to-many relationship with the Campaign table, while the Influencer table is linked to campaigns through a many-to-many relationship via the campaign_influencer_association table. Additionally, the Ad_requests table enables sponsors to invite specific influencers to participate in their campaigns. The Admin has comprehensive access to view all campaigns, flag inappropriate ones, and access statistics related to sponsors, influencers, and campaigns.

Architecture and Features

The project follows an iteration of the MVC model, embracing the essential idea of separation of concerns. All the models relative to each of the tables are defined in models.py. Controllers handling the functionality related to these models are distributed accordingly across specific route files, such as admin_routes.py, sponsor_routes.py, and influencer_routes.py. It also includes a templates folder where the html templates go, responsible for rendering the user interface and gluing the controllers into authentication mechanisms.
The project is structured inside a IESCP folder, which also contains configuration utility files such as config.py, routes.py, and other core functionalities. The SQLite database is also integrated seamlessly; the structure of the database schema is clearly mentioned in the models.py file.

Various features implemented on this application include the following:

Sponsors & Influencers Management: Company, influencer, and admin user login and signup pages. Secure password hashing and user authentication are implemented.

Campaign Management: A sponsor can create and view campaigns and manage them. An influencer can view open campaigns and apply to them.

Admin Moderation: Admin can see all the campaigns going on at the platform, mark inappropriate content, and see the platform statistics. Admin can also see sponsors and influencers and their details on the platform.

Request Management: Interested influencers, where influencers express interest in certain campaigns; AdRequest, where companies can invite specific influencers to participate in their campaigns are managed through the platform.

Filter : Influencers, sponsors and admin can filter relevant fields based on their category.

Video Link:
https://drive.google.com/file/d/1daEVbfVCrT7XBDWP2ITCtEuZNzHxmdj5/view?usp=sharing