IITM Modern Application Development- 1

Influencer Engagement & Sponsorship Co-ordination Platform

B. Hemanth Sai Vardhan- 23f1001157

Description:

This is a Influencer Engagement and Sponsorship management system for multiple users. Both influencer and Sponsors can sign up and login in the application. influencers and sponsors will be redirected to their respective home pages. Sponsors can create campaigns categorizing them into various niches. They will also be able to manage ad requests by setting goals, specifying requirements, and negotiating payment terms with influencers. Influencers will have the ability to search for public campaigns relevant to their niche and view all ad requests. They can accept or reject requests and negotiate payment amounts, ensuring fair collaboration with sponsors. The application will also include powerful search features to help sponsors find influencers based on niche, reach, and other metrics. The simple design and straight forward user interface make it easier for the users to navigate around the pages.

Database Design:

Entity Identification:

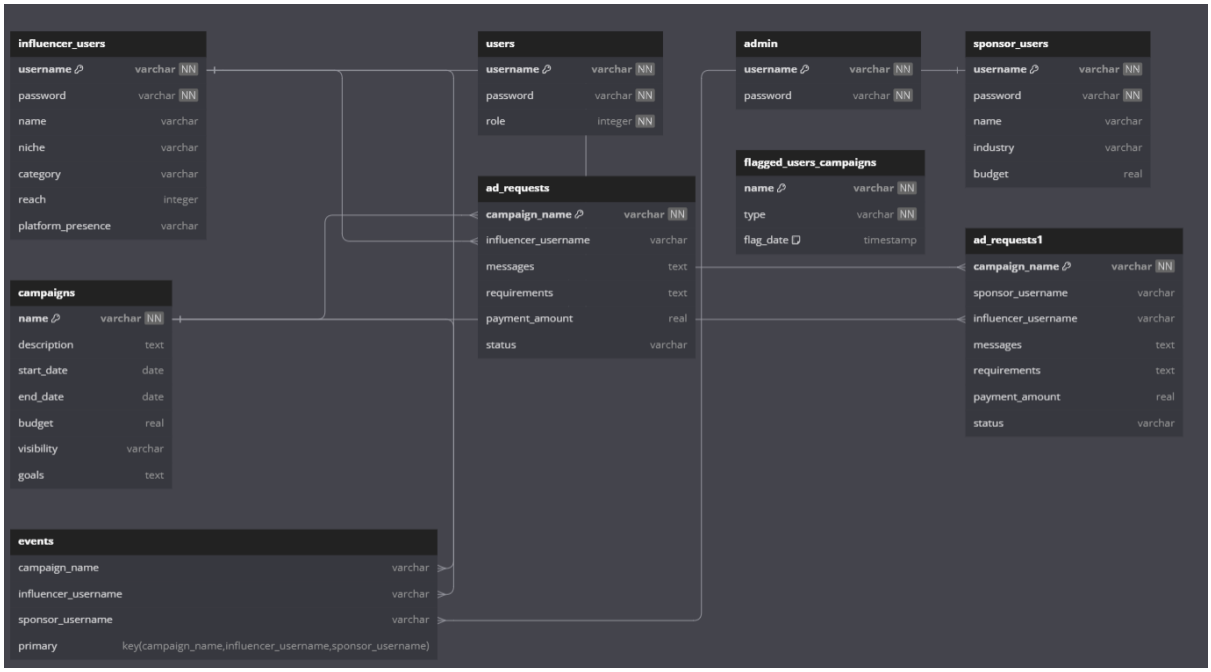
Entities and Attributes:

1. **Influencer User**
    - Attributes: username, password, name, niche, category, reach, platform_presence
2. **Admin**
    - Attributes: username, password
3. **Sponsor User**
    - Attributes: username, password, name, industry, budget
4. **Campaign**
    - Attributes: name, description, start_date, end_date, budget, visibility, goals
5. **Ad Request**
    - Attributes: campaign_name, influencer_username, messages, requirements, payment_amount, status
6. **Ad Request1**
    - Attributes: campaign_name,sponsor_username, influencer_username, messages, requirements, payment_amount, status
7. **Flagged Users/Campaigns**
    - Attributes: name, type, flag_date
8. **Events**

    .Attributes:campaign_name,influencer_username,sponsor_username

Schema:

Considering the above entity model, schema diagram has been constructed



Relationships:

1. In the schema, campaign_name in ad_requests and ad_requests1 refers to the name in campaigns.
2. influencer_username in ad_requests and ad_requests1 refers to the username in influencer_users.
3. Managed through the status field in ad_requests and ad_requests1 (e.g., 'accepted', 'rejected').

Tech Stack:

● Flask – Flask is a backend framework for python. It includes flexible core functionality and an extensive ecosystem of supported modules like FlaskSQL Alchemy for database access, Flask-Login for session management, and Flask-RESTful for API development.

● SQLite – SQLite is a embedded database engine written in C. It's selfcontained, i.e. it doesn't require a separate server process.

● Jinja 2 – Jinja is a fast, expressive, extensible templating engine. Special placeholders in the template allow writing code similar to Python syntax. Then the template is passed data to render the final document.

API Design: ●

GET request – For rendering html pages and return data from sqlite server.

POST request – For add and edit operations on the database. Used in adding users, books, sections, editing books, sections etc...

DELETE request – For deleting operations performed on the database.

VIDEO LINK: https://drive.google.com/file/d/1Qrk3CRivCWT4h8bOskG0F9JUZEBjAIWs/view?usp=sharing