SUVANJAN DAS | 22f2001617@ds.study.iitm.ac.in | Indian Institute of Technology, Madras

Problem Statement: The problem was to create a digital platform where influencers and sponsors can interact, manage campaigns, and track their progress. The solution involves building a dashboard that allows both parties to register, manage their profiles, and engage in sponsorship activities. Additionally, an admin panel was created for overseeing and moderating the platform's activities. It's a platform to connect Sponsors and Influencers so that sponsors can get their product/service advertised and influencers can get monetary benefits.

Project Description: The project aimed to develop a web-based dashboard that provides a user-friendly interface for both influencers and sponsors to manage sponsorship campaigns. The platform offers various functionalities, including user registration, login, campaign management, ad requests, and an administrative dashboard. These features are designed to facilitate seamless interaction between influencers and sponsors, allowing them to easily manage their sponsorship engagements and track campaign progress.

The dashboard is divided into different sections for influencers, sponsors, and admins. Influencers can view and accept ad requests, while sponsors can create and manage campaigns. The admin panel provides oversight of platform activities, enabling the admin to monitor user activities and manage the overall platform efficiently.

Approach: The development process began with designing the database schema to structure the relationships between influencers, sponsors, campaigns, and ad requests. The backend logic was implemented using Flask, a lightweight web framework that allows for quick development and easy maintenance. Flask was chosen due to its flexibility and extensive ecosystem of extensions, which made it ideal for building the project.

The frontend was designed using Jinja2 templates integrated with Bootstrap, providing a responsive and aesthetically pleasing user interface. The use of Bootstrap ensured that the platform is accessible across different devices and screen sizes. SQLAlchemy was used as the Object-Relational Mapping (ORM) tool, enabling smooth interactions between the Python application and the underlying SQL database.

Frameworks Used:

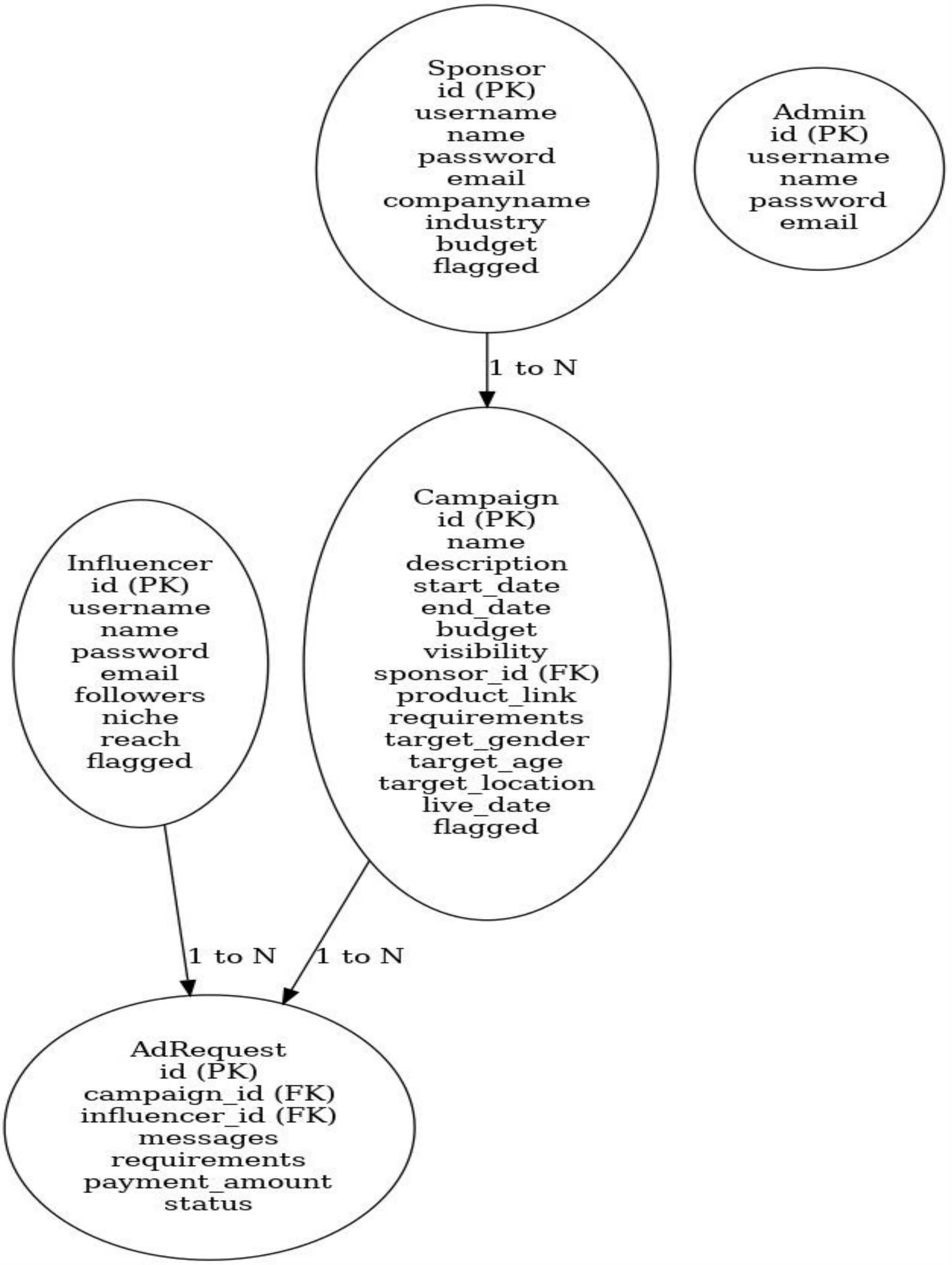
- Flask for application code.
- Jinja2 templates + Bootstrap for HTML generation and styling.
- SQL Alchemy for data storage.

Libraries Used:

alembic, blinker, certifi, charset-normalizer, click, colorama, Flask, Flask-Migrate, Flask-SQLAlchemy, greenlet, idna, itsdangerous, Jinja2, Mako, MarkupSafe, requests, SQLAlchemy, typing_extensions, urllib3, Werkzeug.

Presentation Drive Link :
https://drive.google.com/file/d/1QTXY1wSbKUTcnnT2EEIJGFnrfcOWESnT/view?usp=sharing

ER DIAGRAM.


Sponsor
id (PK)
username
name
password
email
companyname
industry
budget
flagged
Admin
id (PK)
username
name
password
email
1 to N
Campaign
id (PK)
name
description
start_date
end_date
budget
visibility
sponsor_id (FK)
product_link
requirements
target_gender
target_age
target_location
live_date
flagged
Influencer
id (PK)
username
name
password
email
followers
niche
reach
flagged
1 to N
1 to N
AdRequest
id (PK)
campaign_id (FK)
influencer_id (FK)
messages
requirements
payment_amount
status