

MAD II - Project Report

Student Details

Name: Vikram Singh Negi, Roll No: 22f3001905, Email: 22f3001905@ds.study.iitm.ac.in

Project Details

Question Statement: Influencer Engagement & Sponsorship Coordination Platform - V2

Problem Statement

My first observation was that there should be three different types of users: Sponsor, Influencer, and Admin. These users have different access controls, eg. only admin can view the admin dashboard. So, there was a need to implement role-based access control in both the frontend implemented with Vue and the backend implemented with Flask.

Simply implemented a similar database setup for the application as done in the MAD - I project. The major difference is token based authentication as we are not directly interacting with flask but rather using it to create APIs for our Vue front-end.

The frontend is completely built with industry standard web technologies like Vue 3 and other libraries such as vue-router and Pinia for state management. Frontend caching was implemented using localStorage API, which is available in all modern web browsers.

To create the backend jobs such as sending daily reminders to influencers, monthly campaign reports to sponsors, and a feature to download the campaigns data required the use of redis. For checking if the download is ready, short polling technique was used at client side.

I also made sure that both frontend and backend validation is being implemented. And added try except blocks wherever possible, eg. database error will be shown to the user as an internal server error.

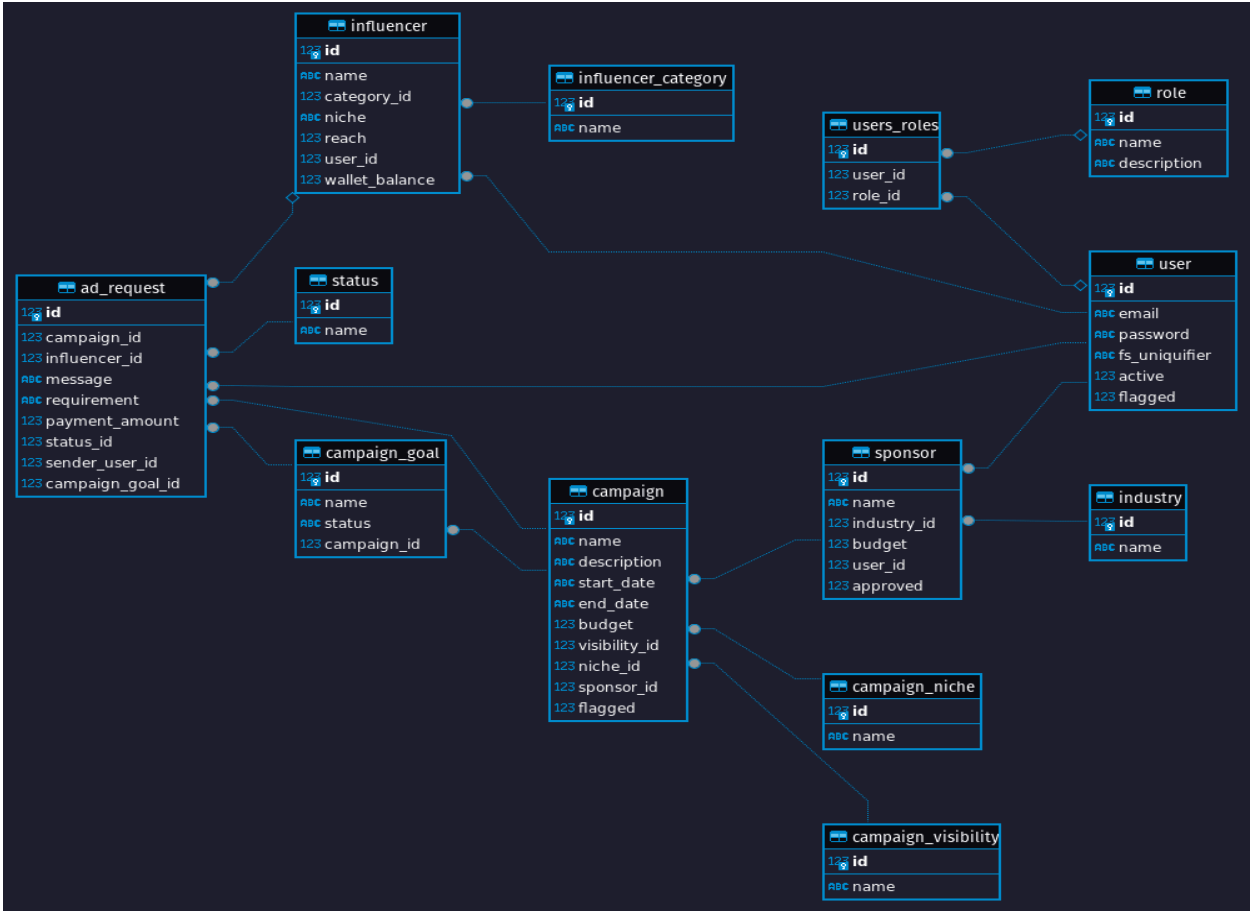
Frameworks and Libraries

- **Backend:** celery, Flask, Flask-Caching, Flask-Cors, Flask-Login, Flask-Mail, Flask-RESTful, Flask-Security-Too, Flask-SQLAlchemy, pandas, python-dotenv, redis
- **Frontend:** bootstrap, chart.js, pinia, vite-plugin-pwa, vue, vue-router

Presentation Video

Link:  MAD II - Project Video.mp4

Database Entity-Relationship (ER) Diagram



REST API Endpoints

Endpoint	Method
/user/create	POST
/campaign	POST
/campaign/:campaign_id	GET, PUT, DELETE
/ad-request	POST
/ad-request/:ad_request_id	GET, PUT, DELETE
/search/influencers	POST
/search/campaigns	POST