

# INFLUENCER ENGAGEMENT AND SPONSORSHIP COORDINATION PLATFORM – VERSION 2

## Final Project Report

BY M. MAHADEVAN

23f2005072 | Modern Application Development-II | Sept, 2024Term

### AUTHOR :

Name– M.Mahadevan

Roll no– 23f2005072

E-Mail– [23f2005072@ds.study.iitm.ac.in](mailto:23f2005072@ds.study.iitm.ac.in)

### DESCRIPTION OF PROJECT :

This influencer engagement and sponsorship coordination platform\_V2 connects the influencers and sponsors to market or advertise a product. Here sponsors can get their product/service advertised and influencers can get monetary benefit. This app meets the demands of sponsors, influencers.

### TECHNOLOGIES USED :

- **Flask** – Backend framework for building the web application.
- **SQLAlchemy** – ORM(Object-Relational Mapping) tool for database interactions.
- **Vue.js** – Used for building frontend user interfaces.
- **Axios** – HTTP client for API requests.
- **Celery** – Used for managing batch jobs and background tasks.
- **SQLite** – Database for storing application data.
- **Flask Security** –Provides token based authentication.
- **Flask-RESTful** – Used for building API's.
- **MailHog** – Testing mail functionalities.
- **Flask SMTP Mail** – Used for sending mails.
- **Chart.js** – Used for dynamic and responsive analytics.

### ARCHITECTURE :

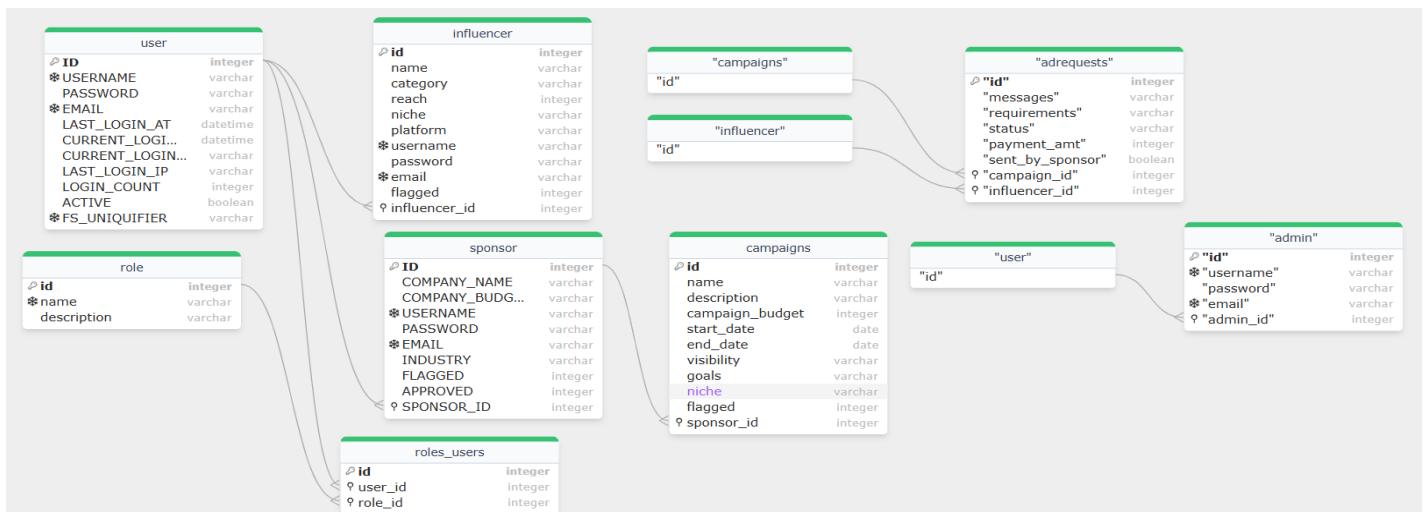
The description of the files inside IESCP\_V2 are as follows

- **app.py** – It initiates the web app and starts the server.
- **routes** – This folder contains all the resources which are used to perform operations in the web app. All the operations are written here like statistics calculation, plotting of graph and all functionalities are implemented here.
- **models.py** – This file contains all the database classes of the application.
- **frontend**– This folder contains all the frontend code for Sponsor, Influencer, Admin.
- **task.py** – This contains all the background jobs definition.
- **mail\_service.py** – This has the configuration for sending mails.
- **initial\_db.py** – This has the code for initializing the database

## API:

- '/admin/login'
- '/sponsor/login'
- '/sponsor/register'
- '/influencer/login'
- '/influencer/register'
- '/sponsor\_resources'
- '/admin/sponsor/<int:id>/approve'
- '/admin/sponsor/<int:id>/flagging'
- '/admin/sponsor/<int:id>/unflagging'
- '/admin/influencer/<int:id>/flagging'
- '/admin/influencer/<int:id>/unflagging'
- '/campaign\_resources'
- '/admin/campaign/<int:id>/flagging'
- '/admin/campaign/<int:id>/unflagging'
- '/admin/overall\_stats'
- '/sponsor\_campaign'
- '/sponsor/campaign/<int:id>'
- 'sponsor/adrequests/<int:sponsor\_id>'
- '/search/campaigns'
- /sponsor/<int:sponsor\_id>/statistics'
- '/all\_active/influencers'
- 'all\_public/campaigns'
- '/influencer/ad\_request'
- '/influencer/update\_details/<int:influencer\_id>'
- '/influencer/<int:influencer\_id>/statistics'
- '/search/influencers'
- '/sponsor/ad\_request/<int:sponsor\_id>'
- '/export\_csv'
- '/get-csv/<id>'

## DB SCHEMA DESIGN :



## VIDEO LINK :

[https://drive.google.com/file/d/1\\_hJWbBw1qLwg8wWTK2ShPUE24-hbA261/view?usp=sharing](https://drive.google.com/file/d/1_hJWbBw1qLwg8wWTK2ShPUE24-hbA261/view?usp=sharing)