

# Modern Application Development1-Project

## Influencer Engagement and Sponsorship Coordination Platform (IESCP)

### Student Details

**Name:** Irina Agastin

**Student ID:** 23f2004169

**Email:** [23f2004169@ds.study.iitm.ac.in](mailto:23f2004169@ds.study.iitm.ac.in)

### Project Details

**Title:** Influencer Engagement and Sponsorship Coordination Platform (IESCP)

**Objective:** To create a platform that connects sponsors and influencers, enabling sponsors to advertise their products/services and influencers to gain monetary benefits.

### Problem Statement:

The goal is to develop a web application where sponsors can create and manage advertising campaigns and influencers can search for and participate in these campaigns. The application should facilitate seamless communication between the two parties while being monitored and managed by an admin.

### Approach:

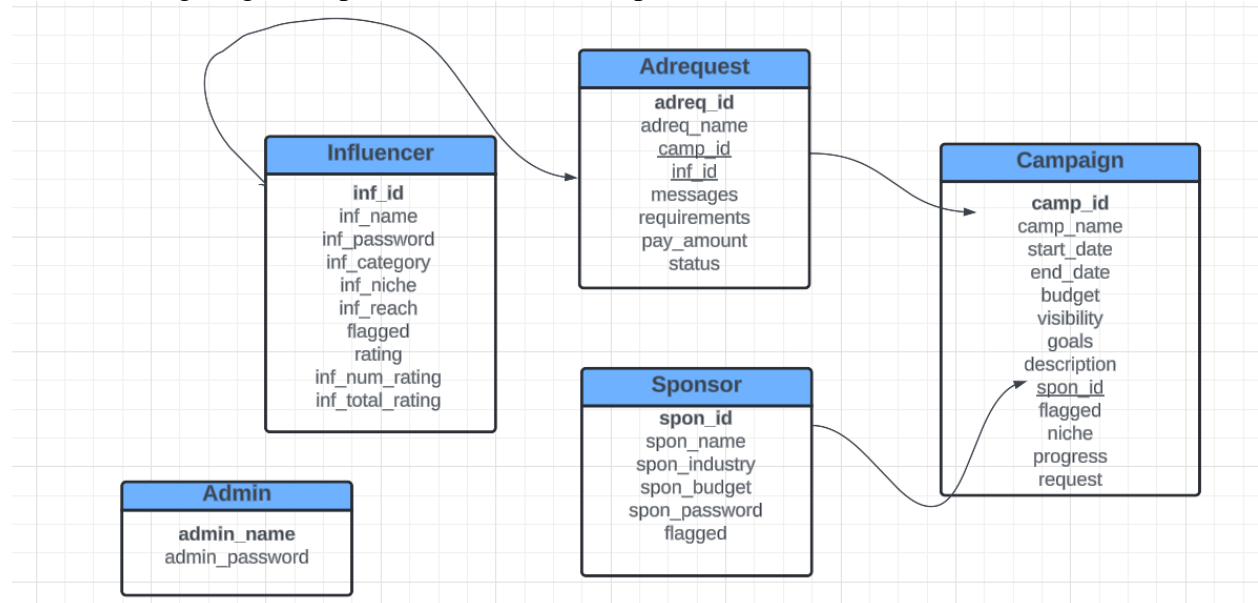
1. **User Login and register:** Implement a login/register system for all users
2. **Admin Dashboard:** Create a dashboard to monitor users, campaigns, ad requests, and flagged items and remove them from the platform if required.
3. **Campaign Management:** Enable sponsors to create, update, and delete campaigns.
4. **Ad Request Management:** Allow sponsors to create, edit, and delete ad requests for their campaigns.
5. **Search Functionalities:** Implement search capabilities for sponsors to find influencers and for influencers to find public campaigns and send requests to sponsors.
6. **Ad Request Actions:** Provide influencers with the ability to view, accept/reject, and negotiate ad requests.

### Frameworks and Libraries Used

- **Flask:** For the application backend and routing.
- **Jinja2:** For template rendering (dynamic content generation )
- **Bootstrap and css:** For responsive and aesthetic front-end design.
- **SQLite:** For data storage and management.

## ER Diagram

The following diagram represents the relationships between the main entities in the database:



## API Resource Endpoints

The application includes the following CRUD API endpoints for ad requests, and campaigns:

Create: POST/api\_create\_ad, , POST/api\_create\_camp

Read: GET/api\_read\_ad, GET/api\_read\_camp

Update: POST/api\_update\_ad, , POST/api\_update\_camp

Delete: POST/api\_delete\_ad, , POST/api\_delete\_camp

## Presentation Video

The following link provides access to a brief video presentation explaining the project's implementation and features:

[https://drive.google.com/file/d/1ycL6vLOiZmTaJyAp7aSMoTWDX4yFVxzT/view?usp=drive\\_link](https://drive.google.com/file/d/1ycL6vLOiZmTaJyAp7aSMoTWDX4yFVxzT/view?usp=drive_link)

## Conclusion

This project aims to provide a comprehensive platform for sponsors and influencers to collaborate efficiently. By leveraging Flask, Jinja2, Bootstrap, and SQLite, the application ensures a seamless user experience and robust data management.