Modern Application Development -2 Project Report

IITM BS Degree Program

Name: Satyaki Goswami **Roll Number:** 21F2001257 **Email:** SATYAKI GOSWAMI

About Myself: I am a dedicated student who has recently started delving into the world of WebDev. After this project, I would like to explore more of this field.

1. Description

SyncIt is a platform for connecting Sponsors and Influencers, enabling Influencers to earn money through advertising. Designed for the MAD - 2 Project (May 2024 term), it features role-based logins for Influencers, Sponsors, and Admins. The dynamic registration page adjusts options based on the chosen role. Sponsors can create and manage campaigns and ad-requests, while Influencers can send ad-requests for campaigns they like. The system uses models for Influencers, Sponsors, Campaigns, and Ad-Requests.

2. Technologies Used

- **Flask :** This Python web framework has been used for backend development.
- **Flask-Security :** Provides user session management and authentication
- **Flask-RESTful** : To create REST APIs for data exchange
- **Celery:** It has been used for queuing tasks
- **Vue 2:** This JavaScript framework has been used to build interfaces
- **Flask-SQLAlchemy :** This library provides an object relational mapper.
- **SQLite3 :** Provides a lightweight and simple database storage engine
- **HTML/CSS/Jinja2 :** It has been used to define the content and structure of the front end

3. Database Schema Design

- **User Table (user)**
  inf_id (PK; int), username (str; unique; nn), password (str; nn), active (bool), fs_uniquifier (str; nn)
- **Role Table (role)**
  inf_id (PK; int), name (str; unique; nn), description (str)
- **RoleUsers Table (role_users)**
  inf_id (PK; int), user_id (FK; int), role_id (FK; int)
- **Influencer Table (inf_info)**
  id (PK; int), email (str; nn), full_name (str; nn), category (str; nn), niche (str; nn), no_of_followers (int; nn), flag_status (str, default: “Unflagged”), user_id (FK; int)
- **Sponsor Table (spon_info)**
  id (PK; int), spon_type (str; nn), email (str; nn), full_name (str; nn), industry (str; nn), evaluation (int; nn), flag_status (str; default: “Unflagged), user_id (FK; int),
- **Campaign Table (campaign)**

id (PK; int), spon_id (FK; int; nn), name (str; nn), description (txt; nn), sdate (date; nn), edate (date; nn), budget (int; nn), visibility (str; nn), completion_status (str; default: “Active”), flag_status (str; default: “Unflagged”)

- **Adreq Table (adreq)**
  id (PK; int), camp_id (FK; intr; nn), inf_id (FK; int; nn), messeges (txt; default:”DEFAULT TEMPLATE”), requirements (txt; nn), payment (int; nn), req_status (str; default: “PENDING”), sender (str; default: “WAIT”)

LINK TO THE ER DIAGRAM

4. API Design

APIs for user, influencer, sponsor, campaign, and ad request have been created for CRUD operations. These have been created using Flask-RESTful, following the principles of structured data exchange.

API RESOURCES ENDPOINTS

5. Architecture and Features

- SyncIt uses a Model-View-Controller (MVC) architecture. The “models.py” file in the backend folder handles database operations (Model), HTML templates manage the frontend (View), and “app.py” and “controllers.py” in the root folder handle routing and business logic (Controller).
- APIs have been used for structured data exchange. At the same time, Flask-Security takes care of authentication and tokenisation.
- Sponsors can create, update, or delete campaigns and generate multiple ad requests for each campaign, with options to update or delete those requests as well. They have the ability to search for Influencers using filters and manage ad requests from Influencers by accepting or rejecting them. They can receive ad requests from Influencers as well.
- Influencers can search for campaigns (with filtering features) and view the details of each public campaign (private campaigns created by sponsors are not visible through search). They can accept or reject ad requests and at the same time, send ad requests to the Sponsors as well.
- The Admin can monitor all the activities taking place on SyncIT. They can approve Sponsors, flag Sponsors - barring them from creating campaigns or being displayed through the Influencer’s search bar. When a campaign is flagged, the corresponding Sponsor won’t be able to create ad requests for it. Similarly, if an Influencer is flagged, they won’t be shown in the Sponsor’s search bar and they won’t be able to send an ad request. The Admin can also keep track of Influencer and Sponsor statistics.
- Periodic tasks like monthly report and daily reminders are also scheduled.

6. Presentation Video

LINK TO THE PRESENTATION VIDEO