

### Student Details:

**Name:** Milan Kumar Chandra

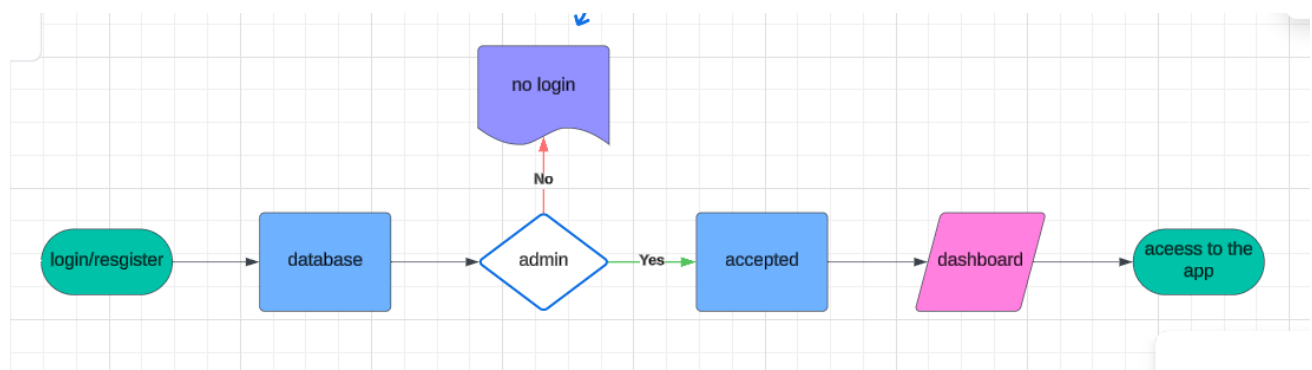
**Student ID:** 21f2000882

---

### Frameworks and Libraries Used:

- **Frontend:**
    - Vue.js, Bootstrap, Chart.js, JavaScript, HTML, CSS
  - **Backend:**
    - Flask, Python
  - **Database:**
    - SQLite3
  - **Other Tools:**
    - Celery and Redis for task management, Redis for caching
- 

### ER Diagram:



### API Resource Endpoints:

- **User API:**
  - POST /register - Register a new user
  - POST /login - User login
- **Admin API:**
  - GET /admin/monitor - Monitor all users
  - POST /admin/approve\_sponsors/<int:sponsor\_id> - Approve a sponsor
- **Sponsor API:**

- POST /sponsor/campaign - Create a new campaign
- PUT /sponsor/campaign/<int:campaign\_id> - Update an existing campaign
- POST /sponsor/ad\_request - Create a new ad request
- **Influencer API:**
  - GET /influencer/campaigns - Get influencer campaigns
  - POST /influencer/ad\_request/<int:ad\_request\_id> - Respond to an ad request
- **Problem Solving Approach**
  - To address the problem statement, I began by focusing on the backend development. I chose Flask as the framework for building the backend due to its simplicity and flexibility. My first task was to design and implement the API endpoints necessary for the project. This involved defining the various routes, handling requests, and ensuring that the backend was capable of interacting effectively with the database.
  - Once the backend was fully functional and the API endpoints were established, I shifted my focus to the frontend development. For the frontend, I utilized Vue.js, which allowed me to create a dynamic and responsive user interface. I integrated Vue with the backend API endpoints to ensure smooth communication between the frontend and backend, enabling users to interact with the system effectively.
  - Throughout this process, I conducted initial research and analysis to determine the best practices for API design and frontend development. This included studying similar projects, evaluating different frameworks and libraries, and considering design patterns that would enhance the overall performance and user experience of the application.