MAD PROJECT

Author

Name: V S Swetha

Roll number :21f1001882

Email: 21f1001882@student.onlinedegree.iitm.ac.in

I am a second year B.Tech student from Chennai , India, with a nudge for data science .

Description

Sousveillance, is a website that allows users to track their daily habits, moods, etc to get a better understanding of themselves. This project is about having to develop a website that is Quantified Self Application. This website allows users to track any details about themselves for period and do self-analysis. The users can delete, edit or add new trackers and log data based on their need.

Technologies used

Languages used: Python, HTML, CSS, Bootstrap, JavaScript

Frameworks and libraries:

Flask – render_template (), request, redirect, flash, session, jsonify

Sqlite, json, SQLAlchemy

datetime, logging

DB Schema Design

activity db - db to manage and add new tracker

Columns: id (primary key, autoincrement) , activity, description, tracker_type, date, mselect, userid

user db- db to manage user login info and tracker details specific to user

Columns - id (primary key, autoincrement), name (not null), username (unique, not null), password(not null)

The constraints are specified in that way so that each username can be unique to access their respective page information

tracker db- db to manage tracker table of available trackers

Columns- id (primary key, autoincrement), tracker_id (foreign key), date (not null), tracker) value (not null), description (not null)

API Design

Unfortunately, due to lack of time and ability, no APIs could be incorporated in this project.

Architecture and Features

The templates folder contains all the html pages whereas the static folder contains two sub folders, CSS and images, which contains the CSS and images used respectively for the application.

The users have a choice to create their own trackers with different types. The website consists of

Home page:

The page contains a simple CSS animation with the website name. It also has Login and Sign-Up buttons which would take the user to the respective pages.

Login page:

The login page has username and password with client-side verification. A message is sent if username or password is wrong. Username and password accept all alphanumeric characters. If the username doesn't exist, the user has an option to sign up with a link.

Signup page:

The page asks for email, phone number, name, username and password. A message is sent if username already exists. Client-side validation is done so that user may not leave certain boxes empty and enters acceptable data.

Tracker page:

CRUD applications have been used to

- Create a new tracker
- Edit tracker name and description
- Delete tracker

These have been enlisted in a table format which also displays the last logged details. Each tracker name is a link to the respective page of the tracker which contains tracker information.

Logging page:

The logging page has again used the CRUD application to detail the list of logs made by the user for the tracker. Options such as editing log, adding new log and deleting log are available. It also displays a graph (using JavaScript) which displays a trend of the user's activities.

Video

https://drive.google.com/drive/folders/1u0-hi3Ej_4j8OQuVA5YKu5o60gOzDmjd?usp=sharing