Author

Satyabrata Dwivedy
21f1007010
21f1007010@student.onlinedegree.iitm.ac.in
I am a student at Veer Surendra Sai University of Technology, Burla, Odisha currently pursuing BTech. In Electrical engineering.

Description

We need to create an independent web application which keeps track of various day today activities. We have to create a backend (database) which keeps track of multiple trackers and their logs for multiple users and connect it to frontend

Technologies used

Flask -

- Flash - it used to show feedbacks
- Url_for - it used to link the pages
- RenderTemplate - it is used to render the templates created using jinja2
- Request - it is used get the form values from the pages
- SQLAlchemy - it is used to connect with the database and to run query
- Flask-login is used for secure authentication

Matplotlib-

- It is used to create the trendline

Bootstrap -

- Overall design of the web app is implemented with Bootstrap 5.1.

SQLite -

- SQLite3 and its appropriate model are implemented and connected to Flask.

HTML and CSS -

- Used to create the web pages

DB Schema Design

Tables (4)

Name	Type	Schema
log_table		CREATE TABLE "log_table" ("log_id" INTEGER, "tracker_id" INTEGER NOT NULL, "time" TEXT NOT NULL UNIQUE, "value" TEXT NOT NULL, "note" TEXT, "user_name" TEXT NOT NULL, "edited_date" TEXT, PRIMARY KEY("log_id" AUTOINCREMENT), FOREIGN KEY("user_name") REFERENCES "user"("user_name") ON DELETE CASCADE, FOREIGN KEY("tracker_id") REFERENCES "tracker"("tracker_id") ON DELETE CASCADE)
log_id	INTEGER	"log_id" INTEGER
tracker_id	INTEGER	"tracker_id" INTEGER NOT NULL
time	TEXT	"time" TEXT NOT NULL UNIQUE
value	TEXT	"value" TEXT NOT NULL
note	TEXT	"note" TEXT
user_name	TEXT	"user_name" TEXT NOT NULL
edited_date	TEXT	"edited_date" TEXT
sqlite_sequence		CREATE TABLE sqlite_sequence(name,seq)
name		"name"
seq		"seq"
tracker		CREATE TABLE "tracker" ("tracker_id" INTEGER UNIQUE, "user_name" NUMERIC NOT NULL, "tracker_name" TEXT NOT NULL, "description" TEXT, "tracker_type" INTEGER NOT NULL, "options" INTEGER, FOREIGN KEY("user_name") REFERENCES "user"("user_name") ON DELETE CASCADE, PRIMARY KEY("tracker_id" AUTOINCREMENT))
tracker_id	INTEGER	"tracker_id" INTEGER UNIQUE
user_name	NUMERIC	"user_name" NUMERIC NOT NULL
tracker_name	TEXT	"tracker_name" TEXT NOT NULL
description	TEXT	"description" TEXT
tracker_type	INTEGER	"tracker_type" INTEGER NOT NULL
options	INTEGER	"options" INTEGER
user		CREATE TABLE "user" ("id" INTEGER, "user_name" TEXT UNIQUE, "password" TEXT, "name" TEXT, PRIMARY KEY("id" AUTOINCREMENT))
id	INTEGER	"id" INTEGER
user_name	TEXT	"user_name" TEXT UNIQUE
password	TEXT	"password" TEXT
name	TEXT	"name" TEXT

user is created to keep track of user and their passwords
tracker is created to keep track of the different Trackers
log_table is created to keep track of the logs added by the user

API Design

```
@auth.route('/login',methods=['GET','POST'])
```
Authenticate user
```
@auth.route('/signup', methods=['GET','POST'])
```
New user sign up
```
@auth.route('/logout')
```
Logout user
```
@main.route('/')
```
Opens the index page
```
@main.route('/profile')
```
Shows the dashboard
```
@main.route('/addTracker',methods=['GET','POST'])
```
Add a tracker
```
@main.route('/deleteTracker/<track_id>')
```
Delete a tracker
```
@main.route('/editTracker/<track_id>',methods=['GET','POST'])
```
Edit a tracker
```
@main.route('/TrackerInfo/<track_id>',methods=['GET','POST'])
```
Information for Tracker
```
@main.route('/<track_id>/addLog',methods=['GET','POST'])
```
Add a log for tracker
```
@main.route('/deleteLog/<l_id>',methods=['GET','POST'])
```
Delete a log
```
@main.route('/editLog/<l_id>',methods=['GET','POST'])
```
Edit a log

Architecture and Features

The app has 6 parts. main.py containing the views and the run command,__ init__.py contains all the necessary thing for the database connectivity, authenticate.py contains code for the authentication of the user, models.py contain all the models of the tables from database
the Templates Folder containing HTML files and Static folder containing images, icons and CSS File

Features:
There is a complete feature for the Login and Sign-up.
There is a Sidebar and Navigation Bar in the Webpage to easily Navigate from any Page to Dashboard and logout.
Every Tracker's Multiple Logs are recorded and Shown in Tabular Form
Each Tracker and Log can be Edited and Removed
Bootstrap and CSS for a clean looking Webpage
Flask login is used for secure authentication

Video

https://drive.google.com/file/d/1SkV5QAExoo_m9_5nkREAvn0dmcDK_QhO/view?usp=sharing