Mohd Ikram

Roll no : 21f2000712
Email : 21f2000712@student.onlinedegree.iitm.ac.in
Info : I am from Prayagraj, Uttar Pradesh I am currently pursuing B.sc in Programming and Data Science as the only graduation degree as of now. I am a book reader, my favorite ones are Atomic Habits, The 12 rules of life and How to talk to anyone.

Quantified Self App

4th September 2022

DESCRIPTION

This Project is about creating a Self tracking web application, which allows the user to do all basic tracking activities for their personal development. In the project, backend and frontend are loosely packed and run on independent servers .For some automation backend jobs celery has been used and they run independently as well

TECHNOLOGIES USED

- **flask:** For creating and running app

- **flask_sqlalchemy** : For handling sql databases in the application including querying , pushing records and so on.
- **datetime :** for working with time and date
- **Celery:** For backend heavy tasks
- **Flask-restful:** For APIs
- **Flask-caching:** For caching at server end
- **Jwt:** For token based authentication
- **Smtplib :** For sending Mails
- **Weasyprint:** For Converting html into pdf

- **email:** For mail structuring

- **matplotlib.pyplot :** For plotting graphs in the app

DB SCHEMA DESIGN

Database has 3 tables, to prevent redundancy and better organization

- **users :** For containing all information about users
  - ❖ **user_id** (primary id), **email, password , name, age, backup** (For password reset)**, gender , trackers(**an array of all trackers of the user through relationship in table**)**
- **trackers :** For storing information about trackers
  - ❖ **tracker_id** (primary key), **tracker_name, description, tracker_type, setting, user_id** (Foreign key) , **logs(**an array of logs of tracker through relationship in table**)**

- **logs :** For storing logs information
  - ❖ **log_id** (primary key), **time, value, note, tracker_id (**Foreign key with reference to trackers table**)**

API DESIGN

I have created an API for CRUD functionality of users, trackers and logs and login and reset password . It is implemented using flask-restful module along with flask for app creation and flask-sqlalchemy for database handling along with flask-caching for caching and flask-security and jwt for token authentication

I have submitted a yaml file separately for the API Documentation

ARCHITECTURE AND FEATURES

The project has a main python source code file named "**main.py**", all other complementary python files and other files are under the directory application.

The frontend has been designed using Vue Cli and bootstrap theme for designing and bootstrap for styling. It runs on an independent server

The project has a login page, on login , it directs the user to the "Dashboard" of the user which displays all the trackers created by that user, along with the option to log entry in the tracker along with a log table and charts for each tracker.. The user can add, edit or delete trackers. There are 4 types of trackers to choose from , numerical, multiple choice, boolean, and time duration. User has the liberty to edit or delete the log entry created by the user. The user can download tracker details in csv format and can download logs data as csv too.

If a user wants to create an account , there is an option to sign up. If a user wants to reset his/her password, the user has an option in the login page to reset it using the backup code he/she might have given on time of account creation.

The app has celery jobs for daily alerts to log for missed entries and monthly report about the user trackers and logs each month

Video Link

[Project Presentation Video](#)