Quantified Self Application - Report

Name : HARISH SAHADEV M
Roll No. : 21F1005856
E-mail : 21f1005856@student.onlinedegree.iitm.ac.in

I'm from Kerala, but currently living in Chennai. MAD-I has introduced me to the exciting world of App Dev. which I'm looking forward to learning. I like to take a stroll at the beaches of Chennai during my free time.

Description

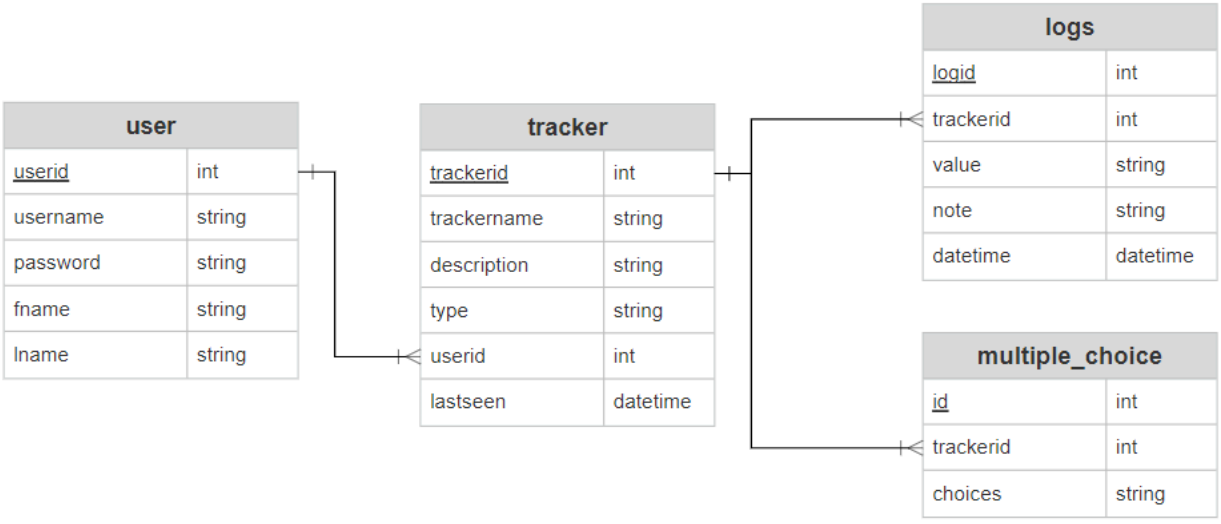
The project is about making a tracker application that can: Create a tracker of choice, Add logs into the tracker, Access the recorded logs using graphs and trend lines, and Edit or Delete both trackers and logs. Every user can create an account and log in using it.

Technologies Used

The project is mainly built on using Flask Framework using Python. The HTML part is covered by Jinja2 templating engine with major styling done using Bootstrap CSS framework.

- *Flask* - This app. Is developed using Flask micro web framework
- Jinja2 - Templating engine to render HTML pages.
- *Flask-SQLAlchemy* - Is used to define the DB models and query the data from the database.
- *SQLAlchemy* - ORM that connects the application to the DB.
- *Flask-WTF* - Used for defining the forms for Registration and Login Forms. Front-end validation for the same.
- *WTForms* - Forms for Registration and Login Pages.
- *Werkzeug* - WSGI web application library
- *Matplotlib* - For plotting graphs and trendlines of tracker logs.

DB Schema Design



Architecture

The project is implemented using Python- Flask framework. Rendering of the HTML is done using Jinja2 templating engine with Bootstrap 5 for most of the styling.

The python implementation of the app is split into the following files -

- Main.py - Initialization with flask and running the application

Application folder-

- Config.py - Configuring the SQLite database file to the application
- Controllers.py - Codes for all the routes of the application are written in this file with the required validations.
- Database.py - Initialization of declarative base from SQLAlchemy.
- Forms.py - Registration and Login form classes are defined here.
- Models.py - Model Classes for all the tables in the database are defined.

Db_directory folder-

- Holds the SQLite file for database storage.

HTML implementation -

- In templates folder.

Static folder-

- Background image
- Dynamically generated graph for tracker logs

Features

Default -

- Registration of new User
- Login of existing user
- Tracker Types - Numerical, Multiple-Choice and Boolean
- CRUD on trackers
- CRUD on logs

Additional -

- Validation for Registration and Login using Flask-WTF
- All styling using Bootstrap 5

Video

https://drive.google.com/file/d/1P95MtCAc4e28BwT8WqD8SIHITRLcVm8f/view?usp=sharing

Replit link

https://quantifiedself-mad-i-project.harishsahadev.repl.co/