Author

Name:- Patil Dhairyasheel
Roll No:- 21f1006987
Email:- 21f1006987@student.onlinedegree.iitm.ac.in
About Me:-

I am a person who believes technology is the perfect solution for all upcoming problems mankind will be facing. Brainstorming about different things , working on myself and achieving goals related to daily processes are the only things that make me happy.

Description

A person has multiple goals going on in his life . Each goal has multiple tasks on an imminent or timely basis. This project acknowledges that and tries to build a tool to effectively administer these tasks . It keeps track of those tasks and creates lists for goals and subsequently cards for tasks in them.

Technologies used

Flask :- It is a useful tool used to create application code in Python .
Jinja2 :- Used to create customized html templates specific to user data.
Bootstrap 5 :- Enable us to advanced styling and designing of html pages.
SQLite :- Stored data in it.
SQLalchemy :- Communicates data to and fro from application code and sqlite database.

DB Schema Design

Database has three tables in it.
Table Name:- Users
Columns :-

user_id = This is an integer which auto increments and its primary key.
username = This is String, unique and not nullable
password = It is String and not nullable
name = It is of type String and not nullable
relationship1 = This has one to many relation with lists table. Because one user can have multiple lists.

Table Name:- Lists
Columns :-

list_id = This is an integer which auto increments,is unique and its primary key.
l_user_id=This is an integer which is foreign key from the users table.
name = It is of type String.
description = It is of type String.
cards=This has one to many relation with cards table. Because one list can have multiple cards.

Table Name:- Cards
Columns :-

card_id = this is an integer which auto increments,is unique and its primary key.
c_list_id =this is an integer which is foreign key from the lists table.
title = it is of type String.

content= it is of type String.

start = it is of type String. It contains the start date of tasks collected while generating in the database.

deadline= it is of type String. It contains the deadline set by the user for the particular task while creating it in the database.

complete = it is of type String.It is “0” if task is incomplete otherwise it will be the date of completion.

update= it is of type String. It will be null if not updated else will be the date when it was updated.

API Design

Login , Register and Delete user. Create,Read,Update,Delete (CRUD) on lists. CRUD on cards. Stats for list data in JSON format .Timeline for completion of tasks in JSON format. Barchart image for stats of list data. Trendline image for timeline of tasks.

Architecture and Features

Api controllers are in the **application/api.py** file .
Database configuration is in **application/config.py** file .
Controllers are in **application/controllers.py** file .
Base class used in models is in **application/database.py** file .
Models are in **application/models.py** file .
Extra functions used in controllers are defined in **application/script.py** file .
Validation errors used for api response are defined in **application/validation.py** file.
Database is in **db_directory/kb.sqlite3** file.
Static folder has all css,summary images, imported/exported csv files and .png files .
All templates are in the **templates** folder.
api.yaml contains api documentation.
local_run.sh script to run application.
local_setup.sh script to setup application.
main.py to base python script to run application.
readme.md describing how to run a project and project structure.
requirements.txt to be installed before running the project.

Users can register and then login with unique credentials. Each user will have a unique board. Users can create or import lists of tasks in the application of preset format. New cards can be created for list which have their own unique title, content ,deadline, complete status,update date if updated .Lists can be created,updated,downloaded, and deleted using buttons . Cards can be created, modified, deleted using buttons. They can also be moved from one list to another using the move button. Using summary button barchart specific to each list and trendline chart for all tasks would be seen. Finally using the logout button the user can log out of the app.

Video

https://drive.google.com/file/d/1P2dh5K7jFKuHKn8KD8lSNVQl5_HgNpWr/view?usp=sharing