Author

HARSH PAL
21f1002562
21f1002562@student.onlinedegree.iitm.ac.in
I had done my schooling from army public school no.1 Roorkee and I'm pursuing my second degree in Bsc physical science from Delhi university.

Description

On this kanban app you can make a list of different fields on which you want to work and each list or field stores different cards or tasks which have to be done in that list or field. App also told you summary or details of done, pending, time left ,time taken and etc in doing tasks or cards.

Technologies used

1. python = 3.8.0,<3.9
2. Flask = 2.2.2
3. Flask-SQLAlchemy =^2.5.1
4. Jinja
5. SQLAlchemy = 1.4.41
6. Flask-RESTful = ^0.3.9
7. Google graphs
8. SQLAlchemy = 1.4.41
9. Flask-Login = ^0.6.2
10. fastapi = 0.82.0

These modules are used because I got friendly with them throughout the MAD 1 course and modules like Flask-login and fastapi provide easy ways of implementing my ideas. This is the minimum requirement to make a task tracking app.

DB Schema Design

Database contain 3 table :-
a) Member table have one to many relationship with List table
b) List table have one to many relationship with Card table
c) Card table

1. Member (which contain details of each user who logged in)
Columns info of member as follows:-

A) id (integer, primary key) :- Contain unique id for each member
B) name (string , not nullable) :- Contain Name for each member
C) email (string , should be unique) :- Contain unique Email for each member
D) Lists (to create relationship with next table List)

2. List (which contain details lists)
Columns info of List as follows:-

A) id (integer, primary key) :- Contain a unique id for each list.
B) list_name (string , not nullable) :- Contain Name of list .
C) date (DateTime format ,default= list form time) :- Contain Date and time at which this list is formed.
D) member_id(integer,foreignkey for Member table) :- Contain Id of member to which this card belongs to.
E) cards (to create relationship with next table Card)

3. Card (which contain details of each card)

A) id (integer, primary key) :- Contain a unique id for each card.
B) task (string of 30 words, not nullable) :- Contain task or title of each card
C) content (string of 500 words, not nullable) :- Contain content of card.
D) have_date (boolean,default=True) :- Contain boolean value to get info related to card crossed due date or not
E) status (boolean,default=False) :- Contain boolean value to get info related to task Done or not
F) form_date (Integer, nullable) :- Contain Date and at which card is formed
G) due_date (Integer,not nullable) :- Contain Date and at which card crossed it's due date
H) done_date (Integer,not nullable) :- Contain Date and at which card task is done

API Design

There is three section of APIs links :-

1) For login in Member
(This section is to get information,create account ,delete account and change password of member who's signup or want to signup in app)
2) For lists on kanban board
(This section is to get information,create ,delete and modify lists)
3) For cards in list
(This section is to get information,create ,delete and modify cards)

Architecture and Features

```
├── kanban :-  ( is folder where our application code is)
│   ├── pycache__
│   │     ├── __init__.cpython-310.pyc
│   │    ├── Accounts.cpython-310.pyc
│   │    ├── controllers.cpython-36.pyc
│   │    ├── api.cpython-310.pyc
│   │    ├── Controllers.cpython-310.pyc
│   │    ├── Models.cpython-310.pyc
│   │    ├── Stat.cpython-310.pyc
│   │    ├── __init__.cpython-37.pyc
│   │    └── validation.cpython-310.pyc
│   │
│   ├── static :-  (Default “static” files folder used to front end)
│   │     ├── assets
│   │     │     └──img
│   │     ├── css
│   │     │     └── styles
│   │     └── js
│   │           └── scripts.js
│   ├── templates :-  (Default flask templates folder)
│   │     ├── info.html
│   │     ├── home.html
│   │     ├── layout.html
│   │     ├── list.html
│   │     ├── login.html
│   │     ├── sign_up.html
│   │     ├── view.html
│   │     └── index.html
│   │
│   ├── __init__.py
│   ├── Accounts.py
│   ├── controllers.py
│   ├── api.py
│   ├── Controllers.py
│   ├── Models.py
│   ├── database.py
│   ├── myalldata.sqlite3  :- (it is my  app database)
│   └── Stat.py
└── App.py  :- ( application initiator just run this file to run app   )
│
```

├── **kanban Api.yaml :- (** *you can work with api part* **)**
└── **requirements.txt**

After the login app shows you the home page from where you can add lists.

A) List have option to view (from where you can see statistic of tasks in list which is implemented by with the help of google graphs) and List have option of delete and edit list option

B) After clicking on edit , app will open the page where you find the options of:

   a) Changing list name

   b) Adding , deleting and changing status of cards

   c) Moving the card from one list to other by providing card title and list name

Video

https://drive.google.com/file/d/1_emh4G-mzC11eNVOa77i4Oh7F-MRjES3/view?usp=sharing