Kanban Application V1 Documentation

Author

Paladugu Venkata Shabarish
21f1001346
21f1001346@student.onlinedegree.iitm.ac.in
I have completed my BTech in 2017 and have a work experience in different non technical domains. I am very much interested in this IITM BS degree and looking forward to learn more.

Description

Kanban Application is a Todo kind of application where we can log in as a specific user and add up to 5 lists in a row to display on the web page. Each List can contain multiple tasks with deadlines. The summary page will display the trendlines of completed tasks and not completed tasks.

Technologies used

flask - for app routing, templates rendering and redirecting, and flash messages display.
flask-SQLAlchemy - for connecting the SQLite database as a server with the flask application.
flask_restful - to implement CRUD operations using flask API.
json - for API communication and dumping & loading data from server to front end javascript

DB Schema Design

User Table Schema			
Column Name	**Column Type**	**Constraints**	**Reason**
id	Integer	Primary Key, Auto Increment	To uniquely identify the user with id
name	String	Not Null, Unique	

List Tabel Schema			
Column Name	**Column Type**	**Constraints**	**Reason**
id	Integer	Primary Key, Auto Increment	To uniquely identify the list with id
name	String	Not Null	
description	String		
update_date	String	Not Null	To record the last edited time
user_id	Integer	Foreign Key (user.id), Not Null	To identify the lists of particular user

Card Table Schema			
Column Name	**Column Type**	**Constraints**	**Reason**
id	Integer	Primary key, Auto Increment	To uniquely identify the card with id
name	String	Not Null	
content	String		
deadline	String	Not Null	Deadline for every task
toggle	String	Not Null	Toggle value of the card whether it's completed or not.
create_date	String	Not Null	To record the card create date
complete_date	String	Not Null	To record the card complete date
list_id	Integer	Foreign Key (list.id), Not Null	To identify the cards of a particular list.

API Design

API was designed for interaction with lists and cards using CRUD operations.

List has GET (fetches all lists for a particular user), POST (creates a new list), DELETE (deletes the existing list), PUT (updating the existing list)

Card has GET (fetches all cards for a particular list), POST (creates a new card), DELETE (deletes the existing card), PUT (updating the existing card).

YAML file for the API documentation is attached in the root directory.

Architecture and Features

In the root directory, there are folders as application which contains (api.py, validation.py, config.py, database.py, models.py, controllers.py), static which contains (favicon logo, image to be used, script.js), templates which contains all html files, test_database which contains databases and other files in the root directory are YAML file, local_run.sh, local_setup.sh, main.py, readme.md, requirements.txt.

In Kanban Application, a user can login with username and can add multiple lists in his dashboard. In a row, only 5 lists can be accommodate, rest will be added to the bottom row. A list can be edited and deleted. In each list, multiple cards can be added and will display inside each list.

It has the features as toggle button on each card which helps in mark as complete, deadline of each card can show the color coding based on the deadline cross time, each list shows the last updated time. Summary page shows the visual representation of how many tasks completed and not completed in a given list.

Video

https://drive.google.com/drive/folders/1iPLsBe5_QRlbogyghrVgN7_bFTgmL8NV?usp=sharing