Author

J.Sai Satya Narayana
21f3001439@ds.study.iitm.ac.in
Hello, I am Satya.I am pursuing BTech CSE from SRMIST Chennai and BS in Data Science and Applications from IIT Madras(online degree). I am always keen to explore new things related to Tech and avidly inclined towards solving real-life problems by applying my knowledge of Computer Sciences.

Description

My Project is about the Grocery Store App. Here the Admin can add, edit or delete a Category. He can also add products into respective categories and edit their details. He can also delete the items in a category. Users can shop for items of the same or different Category.

Technologies used

Python and its Flask extension. Modules in a flask like Flask, request, redirect, render_template, url_for, flask_cors, flask_sqlalchemy (SQLALchemy()), flask_restful. The main purpose of using Flask is to run the app and render templates on user's requests, sqlalchemy is for the database and flask_restful for creating RESTful API for the application, and Bootstrap, CSS, and Javascript for user interface styling.

DB Schema Design

Class User

- username(string , primary key)
- email(string)
- Password(string)

Class Admin

- username (string, primary key)
- password (string)

Class Category

- id (integer, primary key)
- name (string)
- product_relation

Class Product

- id(integer , primary key)
- name(string)
- unit(string)
- rate(string)
- quantity(integer)
- manufacture_date(string)
- expiry_date(string)
- category_id(integer , foreign key)

Class Association

- category_id(integer , foreign key , primary key)
- product_id(integer , foreign key , primary key)

Class Cart

- id(integer , primary key)
- name(string)
- product_id(integer , foreign key)
- product_name(string)
- req_quantity(integer)
- product_rate(string)
- product_unit(string)

Class Bought

- id(integer , primary key)
- name(string)
- product_id(integer , foreign key)
- product_name(string)
- req_quantity(integer)
- product_rate(string)
- product_unit(string)

API Design

GET method

- For category - /api/category/{id}
- For product - /api/product/{id}

POST method

- For category - /api/category/
- For product - /api/product/

PUT method

- For category - /api/category/{id}
- For product - /api/product/{id}

DELETE method

- For category - /api/category/{id}
- For product - /api/product/{id}

Architecture and Features

In the project python file, the model.py python file consists of different models like Category, Product, Association, User, Admin, Cart, and Bought. The app.py contains all the controllers. I have implemented the login logic for User and Admin. Admin can add, edit, and delete Categories and Products. Admin can also see the orders placed by the users. Users can shop by category and search products based on the name, price and manufacture date of items. Users can see all the items in the cart with a grand total and can able to buy. Users can see his/her order history in my account section.

Video

https://drive.google.com/file/d/1djHYBTG37ya6aZsfEr-xSBapag3Er5nw/view?usp=sharing