Author

Amisha Ashok Das
21f1004853
21f1004853@ds.study.iitm.ac.in
Currently I'm a sophomore of Mba from National Institute of Technology , Rourkela. I'm data
Enthusiast and looking towards gaining expertise in management and data science .

Description

The Grocery Shopping Project is a digital platform that aims to improve and streamline shopping experience. Admins shall use cart management , keep a track of the stock and manage the inventory . Customers shall be able to shop around the various categories and products . An easy to use user friendly interface.

Technologies used

The technologies used in this project are: − Python libraries, HTML/CSS, Jinja,Bootstrap, Flask, Flask-Login, Flask-SQLAlchemy, and SQLite
Python is the core programming language used
Flask is the main framework used for the web app
Flask-Login is used for managing multiple user login and keeping a session alive
Flask-SQLAlchemy is the SQL toolkit used to connect with the database file
SQLite is the database used for the modelling of the application

DB Schema Design

There are 4 tables that has been used in this project - User, order, product , categories
User table - defines details about the users
Id -describes the count of user
Name- displays the name of the user
Password- user account and admin account password
Role- to check whether its admin or user
Email- to save the email id of admin and user
Order table- tracks the order placed
id: An auto-incrementing integer field that serves as the primary key for the table. Each row in the table will have a unique id.
order_num: A string field that represents the order number. Orders may have unique order numbers, but it's not enforced as a unique constraint based on your code.
user_name: A string field that stores the name of the user who placed the order.
date: A string field that stores the date of the order.

product_name: A string field that stores the name of the product being ordered.
quantity: An integer field that stores the quantity of the product ordered.
units: A string field that stores the unit measurement of the product (e.g., pieces, kg, etc.).
unit_price: An integer field that stores the price of a single unit of the product. It's marked as unique, which implies that each product should have a unique unit price.
total_product_amount: An integer field that stores the total amount for the product (unit_price * quantity). It's marked as unique, which seems unnecessary, as multiple orders might have the same total product amount.
total_amount: An integer field that stores the total amount for the order, including all products. It's marked as unique, which again seems unnecessary, as multiple orders might have the same total amount.
Product - This table stores information about products, such as their category, name, units, stock levels, pricing, description, image, and expiry date.
categories -This table stores information about various product categories, including their name, description, and associated products

Architecture and Features

Static-: Stores the images of respective products
- Templates: Stores the HTML templates created for this app
- Application- stores all the python files - config.py, controllers, database, models
- db_directory : groceryStore.sqlite3
- Requirements.txt: text file storing the required packages for this web application
- README.md: General instructions about how to start the application.

Video

Link to video