Author

Dhruv Sharma

21F3002488

21f3002488@ds.study.iitm.ac.in

I am a budding python developer who loves Full Stack Project Development and pursuing Diploma in Programming from IIT Madras and in 3rd Year B.Tech. in Information Technology.

Description

Grocery Store Application is designed to be a multi-user platform, consisting of a store manager and regular users. The purpose of the app is to facilitate the buying and selling of groceries. Users can purchase multiple products, while the store manager can manage sections, categories, and products.

Technologies Used

- · Python (Programming Language)
- · Flask (Web Application Framework)
- · SQLAlchemy (ORM for database management)
- · HTML (front-end markup language)

Grocery Store Application is built using the Flask framework, a lightweight web application framework in Python. The database management is handled through SQLAlchemy, which provides an Object-Relational Mapping (ORM) system, simplifying interactions with the database. The front-end is designed using HTML templates with the help of Flask's built-in templating engine.

DB Schema Design

1.User Table : The user_data table represents registered users of the application. Users can create accounts and log in to the system. It contains the following columns.

- **u_id :** The primary key for each user, which uniquely identifies the user, auto incremented.
- **u_name :** The name of the user, represented as a string with a maximum length of 30 characters.
- **u_pass :** user password, represented as a String with a maximum length of 30 characters.

2. Manager Table: The manager_data table stores information about the store manager. It contains the following columns:

- **m_id :** The primary key for the manager, which uniquely identifies the manager.
- **m_name :** The name of the admin, represented as a String with a max length of 30 characters.
- **m_pass :** The password of the manager, represented as a String with a max length of 30 chars

3. Category Table : The category_table represents different categories of products available in the store. Each product belongs to a specific section. It contains the following columns:

- **cat_id:** The primary key for each section, which uniquely identifies it.
- **cat_name:** The name of the section, represented as a String with a max length of 30 characters

4. Product Table : The product_table represents individual grocery items available in the store. It contains the following columns:

- **product_id:** The primary key for each product, which uniquely identifies it.
- **category_id:** category id of which product belong
- **product_name**: product name, represented as a String with a max length of 30 characters.
- **uom_name:** unit of measure.
- **rate:** price of product, represented as a Float.
- **quantity:** total quantity in stock, represented as an Integer
- **expiry_date**: date of expiry of product.
- **price:** The price of the product

5.UOM Table : The uom_table for unit of measure.

- **uom_id** : primary key of uom_table
- **uom_name** = name in String of uom.

6. All_Orders Table : The all_orders table contains track of all the orders made through web app. It contains only important information about order.

- order_id = uniquely identifies each order.
- customer_id = stores the customer id of customer.
- grand_total= store total bill order of an order.

7.Order_details Table : The order_details stores all the details of recent order till the payment .Once payment is made these tables values are deleted.

- index=uniquely identify each product bought by user, primary key.
- category_id =category id of product bought
- product_id = product id of product bought
- product_name= name of product.
- rate = rate of product in float
- quantity = number of item bought
- price = total price of each product.(rate)*quantity result.

8.Summary Table : The summary_table is made for tracking the sales and helping the manager to make buying decisions about products.

- index=primary key of this table.
- product_id = product id of product.
- category_id = product id of product.
- quantity = quantity of product sold.

Architecture and Features

- **app.py** - contains all the imports and controllers
- **models.py -** contains all the models (classes) and sqlalchemy code.
- **templates folder** - contains all the HTML files.
- **static** - contains graph png image.
- **instance** - contains db.sqlite3 file.
- **env** - is the virtual environment files.

Video Link [Video Link : Dhruv Sharma MAD 1]