Author

TADIMALLA KARTEEK

22f1001543

22f1001543@ds.study.iitm.ac.in

I am currently pursuing **B.E (2nd Year) in ARTIFICIAL INTELLIGENCE & MACHINE LEARNING** from **UNIVERSITY COLLEGE OF ENGINEERING (OSMANIA UNIVERSITY-TELANGANA).** I am interested to pursue my **career in Data Science.**

Description

This is simple online application for shopping Groceries. A user can buy product from different sections. The application has administrator who in turn create, edit and delete Product Managers. The Product Manager is responsible to add, edit or delete the section and products. A user can search/ select all items available in particular section under search option drop down.

Technologies Used

1. **Flask Framework** Version 2.2.2 along with Jinja 2.

   Blueprint to make modular application. I used blueprint to expose my Templates folder and registered the Blueprint with the application example {authU = Blueprint ('authU', __name__, template_folder='templates')

   }

2. **Flask-SQL Alchemy** - for connecting the SQLite database as a server with the flask application.
3. **HTML 5, Bootstrap 5** – For styling the pages.
4. **JavaScript** – For dynamic calculation

DB Schema Design

1. **User**

   uid –> Unique ID for user identification - Integer, Primary Key, Auto Increment
   user –> Username for User - String, Unique, Not Null
   password -> String, Not Null
   email-> Email for Registering new user ->String, Not Null

2. **Admin**

   aid-> Unique ID for admin identification-> Integer, Primary Key, Auto Increment name->String, Not Null, Unique
   password->String, Not Null
   email->String, Not Null

3. **Manager**

   mid->Unique ID for manager identification- Integer, Primary Key, Auto Increment
   manager-> Manager Name-String, Unique, Not Null
   password->String, Not Null
   email->String, Not Null
   admin_id->Integer, Foreign Key admin.aid, Not Null

4. **Section**

   sid-> Unique ID for section identification- Integer, Primary Key, Auto Increment
   section-> Section Name-String, Unique, Not Null
   manager_id->Integer, Foreign Key manager.mid, Not Null

5. **Product**

pid-> Unique ID for product identification- Integer, Primary Key, Auto Increment
product-> Product Name-String, Unique, Not Null
unit-> Measurement of product-String, Not Null
price->Price of product- Integer, Not Null
quantity->Number of Products-Integer, Not Null
sectionid-> Integer, Foreign Key section.sid
manager_id->Integer, Foreign Key manager.mid

6. **Cart**

id-> Unique ID for cart item's identification- Integer, Primary Key, Auto Increment
user_id-> Integer, Foreign Key user.uid
product-> Name of product added to cart-String,Not Null
quantity->number of products, selected by user-Integer, Not Null
max_quantity->max quantity of products-Integer, Not Null
price->Float,Not Null
product_id->Integer, Foreign Key product.pid, Not Null

Architecture and Features

My application is following MVC (Model View Controller) Architecture Pattern. MVC provides a clear separation of concerns, which allows developers to work on different aspects of the application independently.

Model:

Model represents the application's core data and how it should be processed or changed.

View:

The View is responsible for displaying the data to the users.

Controller:

The Controller acts as an intermediary between the Model and the View. It receives user input from the View, processes that input, and updates the Model accordingly. It then retrieves data from the Model and passes it to the appropriate View for display.

VIDEO

https://drive.google.com/file/d/1lGatk-fSGZ9DzqLNdHelDr803hddl1AO/view?usp=sharing