MAD Project I - Grocery Store Web Application

Name - Rekha

Roll Number - 21f1006795
Email Id - 21f1006795@ds.study.iitm.ac.in

I am a Diploma Student and by getting inspired by Machine Learning courses, I am currently learning MI, also I have some experience of web development as I have worked before as React Frontend Developer for some time.

Here is my project's System Design and Model Details

1. **Overview:** The Grocery store web applications are designed to provide an online platform for customers to browse and purchase grocery products. So, while building this application I kept this point in my mind and tried to develop it as an online grocery application like Blinkit, Zepto.
   System Design of this application is as follows - customers can create an account, see all of the available products of all categories, add them to cart and can complete the order by paying a total amount. An admin can add categories, products, and do the same thing as customers.

2. **System Components:**
   - User Interface: I have built a user-friendly interface for customers and admin to interact with application.
   - Server: To get data from the database, I have used HTTP requests and responses, and to show them used proper business logic.
   - Database: To store data like users, products, categories, cart details etc in the database, I have used **Sqlite.**

3. **Models:**

This application includes the following models:

a) User Model:

- Fields: user_id, username, password, role
- Description: I have built Registration and Login interfaces for users and admins.

b) Product Model:

- Fields: product_id, name, price, unit, image_url, expiry_date, quantity, category_id, created_at and updated_at
- Description: For the Product Section, I have an add_product section which takes all above mentioned inputs and shows the latest added products first in the dashboard. Only admins can add products.

c) Category Model:

- Fields: category_id, name
- Description: In the Category section, admins can add categories. In Dashboard, clicking on any category will show a product for that category.

d) Cart Model:

- Fields: cart_id, user_id, admin_id, product_id, product name, price, unit, image, quantity, created_at
- Description: Users can add any product from any category in cart, and in cart user can update quantity of any product and remove product from cart.
- I have created an **Order section**,where all ordered products will be present.

4. System Design: I have used the CRUD **API** for the Product and Category section. An admin can add, edit and delete any product and category.

5. Technologies: For Frontend - HTML, CSS, and javascript functions. For Backend - Python with Flask to handle APIs. For Database - Sqlite.