Modern Application Development -2

By – Ameya Raj

Roll - 21f1003202

eGrocery

Description

The eGrocery web app is a platform designed for both Users and the Business owners. Users can signup, login, Search, Buy, See Purchase History, Edit Profile, Give Feedback. Business Owners have two Roles - Store Manager and Admin. Store Manager could signup, Login (Only when Signup is Approved by Admin), Add Product, Edit Product, Export sales and Inventory as CSV, Submit Request to Admin to Add a New Category, Send Messages to Admin, Graphical view of the Sales and Stock charts. Admin can Create a New Category (only for Admin), Edit Category, Approve/Reject -> (Signup requests, Add Category Requests, Messages) from Store Manager. Asynchronous processing of tasks using Celery -> Sending Reminders to users to visit website, Sending Purchase History as Monthly Activity Report to the Users. Data storage using SQLite. Caching with Redis.

Technologies Used

The app is built using the Flask web framework for the backend. Key Flask extensions include Flask-JWT-Extended for user authentication, and Flask-SQLAlchemy for database management. VueJS is used for the frontend UI, along with Bootstrap for styling, vue-router for routing.Redis is employed for caching, Flask-Caching for caching, flask_mail for sending emails, flask_cors for enabling cross-origin resource sharing, requests for making http requests, and Redis along with Celery are utilized for handling batch jobs.

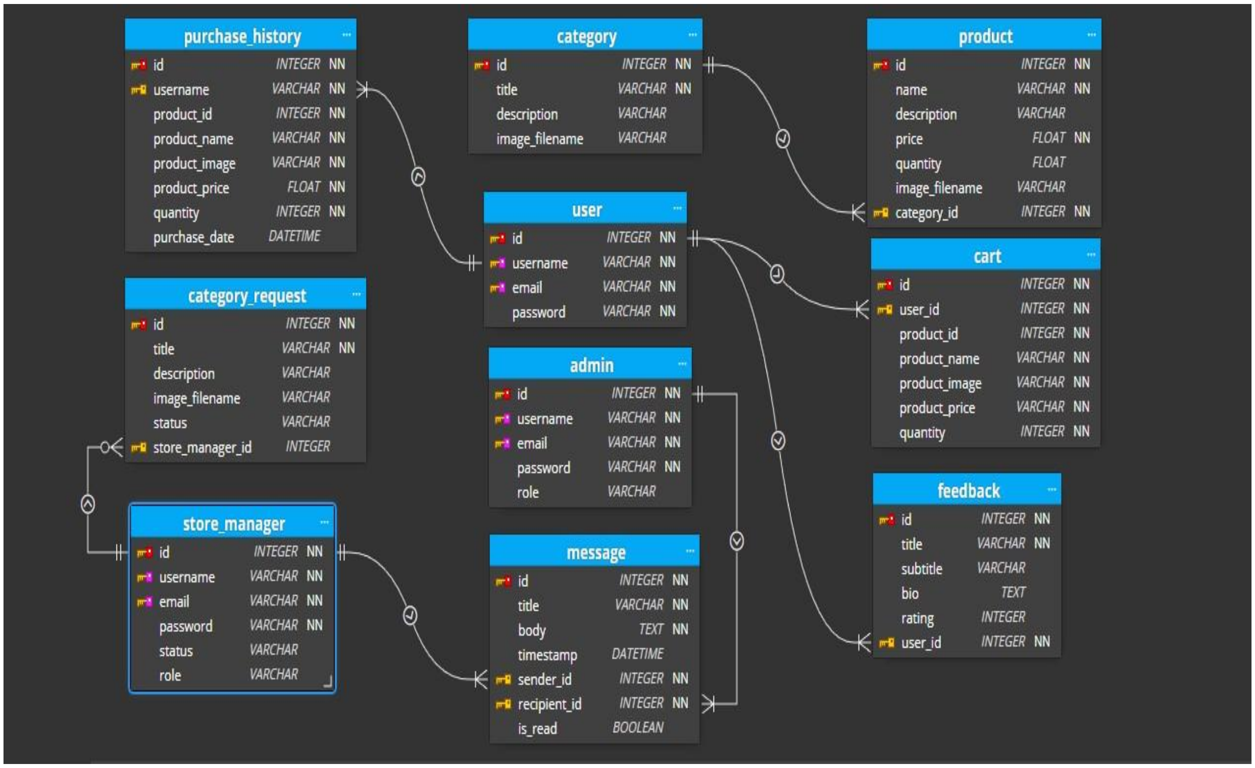
API Design

The API has been designed using Flask. The app consists of app.py and send.py. app.py consists of all the functions related to Signup, Login of Users, Store Managers and Admin. Handling Requests, Displaying Products/Categories, Adding Product/Category, Deleting Product/Category, Editing Product/Category, Fetching Product/Category, Approving/Rejecting Requests, Sending Messages, Providing access based on roles, Exporting files, Searching, caching using Redis, models. Send.py consists of all functions related to scheduling jobs through celery and Redis.

Architecture and Features

The grocery app boasts a comprehensive set of features that enhance user experience and administrative control. The registration process requires users to create accounts using a username and password, with the passwords securely encrypted using generate_password_hash and stored in the database. The login system provides a secure authentication mechanism for users. The home page features a dynamic navbar with links to admin login, user registration, user login, view cart, and logout. After logging in, users can access additional features such as viewing their cart, update and see profile , and purchase history and also gave their feedback . Product search functionality allows users to find specific items quickly. The cart section provides a detailed view of added products, allowing users to remove items and proceed to checkout. The purchase history offers detailed display of the user's buying history. For admin and managers, the dashboard provides an overview of products, including stocks, prices, and descriptions organized by sections. Admins can manage sections, add, edit, or delete and store manager can add edit and delete products, and explore graphical representations of total stock and see sales report. Overall, the app seamlessly integrates user-friendly features with powerful administrative tools for efficient grocery management.

DB Schema Design



purchase_history		
id	INTEGER	NN
username	VARCHAR	NN
product_id	INTEGER	NN
product_name	VARCHAR	NN
product_image	VARCHAR	NN
product_price	FLOAT	NN
quantity	INTEGER	NN
purchase_date	DATETIME	

category		
id	INTEGER	NN
title	VARCHAR	NN
description	VARCHAR	
image_filename	VARCHAR	

product		
id	INTEGER	NN
name	VARCHAR	NN
description	VARCHAR	
price	FLOAT	NN
quantity	FLOAT	
image_filename	VARCHAR	
category_id	INTEGER	NN

user		
id	INTEGER	NN
username	VARCHAR	NN
email	VARCHAR	NN
password	VARCHAR	NN

cart		
id	INTEGER	NN
user_id	INTEGER	NN
product_id	INTEGER	NN
product_name	VARCHAR	NN
product_image	VARCHAR	NN
product_price	VARCHAR	NN
quantity	INTEGER	NN

category_request		
id	INTEGER	NN
title	VARCHAR	NN
description	VARCHAR	
image_filename	VARCHAR	
status	VARCHAR	
store_manager_id	INTEGER	

admin		
id	INTEGER	NN
username	VARCHAR	NN
email	VARCHAR	NN
password	VARCHAR	NN
role	VARCHAR	

feedback		
id	INTEGER	NN
title	VARCHAR	NN
subtitle	VARCHAR	
bio	TEXT	
rating	INTEGER	
user_id	INTEGER	NN

store_manager		
id	INTEGER	NN
username	VARCHAR	NN
email	VARCHAR	NN
password	VARCHAR	NN
status	VARCHAR	
role	VARCHAR	

message		
id	INTEGER	NN
title	VARCHAR	NN
body	TEXT	NN
timestamp	DATETIME	
sender_id	INTEGER	NN
recipient_id	INTEGER	NN
is_read	BOOLEAN	

Presentation Link: http://lnkiy.in/21f1003202_MAD2_Presentation