

## Grocery Store V2 - Application Report

by Navjot (21f2000931)

Frameworks used: Flask, VueJS, Bootstrap, SQLite, Redis, Celery

### Overview:

Grocery Store V2 is a modern web application developed as a part of the Modern Application Development - II course. It is designed to facilitate the buying of groceries with multiple user roles, including admin, store managers, and regular users. The application leverages Flask for the API, VueJS for the UI, Bootstrap for styling, SQLite for the database, Redis for caching, and Celery for handling background jobs.

### Base Requirements:

Authentication and Authorization:

#### 1. User Signup and Login:

- Users can sign up and log in using JWT-based Token Authentication.
- Token-based authentication is used to secure user access.

#### 2. Admin Login:

- Admin has a dedicated login and authentication process.
- Admin access is restricted and requires appropriate authentication.

#### 3. Store Manager Signup and Login:

- Store managers have a separate signup and login process.
- New store manager sign ups require approval from the admin.

### Section/Category and Product Management:

#### 1. Category and Product Management:

- Admin can manage sections/categories, including creation, editing, and removal.
- Store managers can create, edit, and remove products within their allocated sections/categories.
- Store managers can request the admin to create, edit and remove categories.

#### 2. Search for Category/Product:

- Users can search for products based on sections/categories.
- Search functionality includes filtering by price, categories, products.

## Shopping Cart and Buying Products:

### 1.Shopping Cart for User:

- Users can add multiple products to their cart, even from different categories.

### 2.Buy Products:

- Users can view and buy multiple products from one or more sections/categories.
- The system displays product availability and calculates the total amount for the transaction.

## Backend Jobs:

### 1.Daily Reminder Jobs:

- Scheduled job sends daily reminders via email to users who haven't bought anything in the last 24 hrs.

### 2.Monthly Activity Report:

- Monthly progress reports are generated in HTML and Pdf formats and sent via email to users.
- Reports include user orders, total expenditure, etc of the last month.

### 3.User Triggered Async Job - Export as CSV:

- Store managers can trigger an asynchronous job to export product details in CSV format.
- The export includes product details like name, stock remaining, price, units sold, etc and sent via email to that Store manager.

## Performance and Caching:

### 1.Performance and Caching:

- Caching is implemented to improve performance where required.
- Cache expiry is configured to maintain data accuracy.

### 2.API Performance:

- API endpoints are optimized for performance.

## Additional Features:

### 1.Well-Designed PDF Reports:

- Users have the option to choose between HTML and PDF reports.
- PDF reports are well-designed and provide a comprehensive overview.

### 2.Single Responsive UI:

- The UI is designed to be responsive, ensuring a consistent user experience across devices.

### 3.Add to Desktop Feature:

- Users can add the application to their desktop for quick access.

## Database Models:

### 1.Database Models:

- Users, Sections, Category Requests, Products, User Cart, and Shopping History tables are well-defined to capture user and product-related data.

## Run Commands:

- To start the application: python main.py
- Celery worker and beat schedule: celery -A main.celery worker --beat --loglevel=info
- Redis server: redis-server
- I have used Ubuntu wsl for Windows OS

## Conclusion:

Grocery Store V2 is a robust and feature-rich application that meets the specified requirements. It provides a seamless shopping experience for users, efficient management for store managers, and robust administration capabilities. The inclusion of background jobs, caching, and performance optimizations ensures a smooth and responsive user experience. The application is a testament to modern application development practices and leverages a combination of powerful frameworks and technologies to deliver a reliable and user-friendly solution.

## Demo link:

[https://drive.google.com/file/d/1Q-0Rg1W6bRbnZdeb\\_Q\\_h0zTBaiS9RmFY/view?usp=sharing](https://drive.google.com/file/d/1Q-0Rg1W6bRbnZdeb_Q_h0zTBaiS9RmFY/view?usp=sharing)