Modern Application Development – 2

PROJECT REPORT

Growcery – A Grocery Store Application

Jeevan Aji George 21f1007034

21f1007034@ds.study.iitm.ac.in

I am well versed in programming methodologies and have intermediate level experience in web development. I have completed my diploma level courses and all set to enter the degree stage next term.

Description

This report is created to document an application for a grocery store web app. The customers should be able to buy products from market and store managers can sell them accordingly. The admin controls requests from store managers regarding updation or deletion of categories.

Technologies Used

- Flask for API (related extensions like security, RESTful etc.)
- VueJS for frontend
- Bootstrap for styling
- Redis for caching
- Celery for jobs

Database Schema

Roles_users

id *INTEGER NOT NULL PRIMARY KEY*	user_id *INTEGER REFERENCES "user" ("id")*	role_id *INTEGER REFERENCES "role" ("id")*

Users

id *INTEGER NOT NULL PRIMARY KEY*	username *VARCHAR*	email *VARCHAR UNIQUE*	password *VARCHAR(255)*	active *BOOLEAN*	sname *VARCHAR*	fs_uniquifier *VARCHAR(255) NOT NULL UNIQUE*

Roles

id *INTEGER NOT NULL PRIMARY KEY*	name *VARCHAR(80) UNIQUE*	description *VARCHAR(255)*

All_products

id *INTEGER NOT NULL PRIMARY KEY*	p_name *VARCHAR NOT NULL*	category *VARCHAR NOT NULL*	rem_units *INTEGER NOT NULL*	added_date *DATE NOT NULL*	expiry_date *DATE NOT NULL*	rate *INTEGER NOT NULL*	unit_SI *VARCHAR NOT NULL*

All_orders

id *INTEGER NOT NULL PRIMARY KEY*	user_id *INTEGER NOT NULL*	p_id *INTEGER NOT NULL*	order_date *DATE NOT NULL*	quantity *INTEGER NOT NULL*	price *INTEGER NOT NULL*

Request_queue

id *INTEGER NOT NULL PRIMARY KEY*	sname *VARCHAR NOT NULL*	operation *INTEGER NOT NULL*	name *VARCHAR NOT NULL*	name2 *VARCHAR*

Reply_queue

id INTEGER NOT NULL PRIMARY KEY	r_id INTEGER NOT NULL	sname VARCHAR NOT NULL	operation INTEGER NOT NULL	name VARCHAR NOT NULL	name2 VARCHAR	status INTEGER NOT NULL

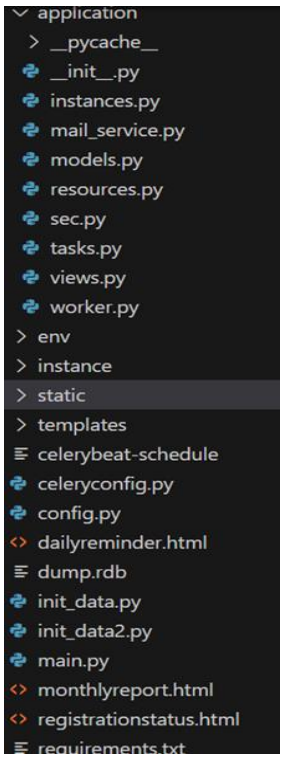
Registration_request

id INTEGER NOT NULL PRIMARY KEY	username VARCHAR NOT NULL	email VARCHAR NOT NULL	sname VARCHAR NOT NULL	password VARCHAR NOT NULL	allowed INTEGER

Category_allowed

sname VARCHAR NOT NULL PRIMARY KEY	category VARCHAR NOT NULL PRIMARY KEY

Architecture and Features



First, we have to create python environment and install dependencies using requirements.txt.

The application is run by running main.py. We have to run celery worker and celery beat for jobs. We have to start redis-server for caching. We have to start MailHog for sending mails.

The application folder contains all the business logic of the application.

mail_services.py handles mail related services, models.py contain the database schema, resources.py contain API endpoints, tasks.py contain all functions related to backend, views.py contains renders for HTML templates.

Instance contains database file which can be viewed using SQLite viewer.

Static contains all javascript , svg , and image files.

Templates contain html files

Init_data.py and init_data2.py are files for initializing database tables

Video Presentation of the project:

https://drive.google.com/file/d/1HAYnKTH-MB_MIzawnAIi1XWX0JQtOlb5/view?usp=drivesdk