

Author

Prasad Patra

22f3000994

22f3000994@ds.study.iitm.ac.in

I am also a BTech student from NIT Rourkela. I like coding and learning new tech stacks.

Description

1. Develop a Flask-based web application that allows users to browse and search for products, add items to their cart, place orders, and enables administrators to manage orders and product inventory.

2. Implement database models for products, users, carts, orders, and related relationships, and create frontend views for product listing, search, cart management, order placement, and admin order management using Flask, SQLAlchemy, and HTML templates.

Technologies used

Flask: Flask is a micro web framework for Python that allows you to build web applications easily.
render_template: A function in Flask to render HTML templates for generating dynamic web pages.

request: An object in Flask that represents the incoming HTTP request.

session: An object in Flask used to manage user session data across requests.

redirect: A function in Flask for redirecting users to a different route or URL.

url_for: A function in Flask to generate URLs for routes in a consistent manner.

json: A module in Python for working with JSON data.

os: A module in Python for interacting with the operating system, used here for generating a secret key.

datetime: A module in Python for working with dates and times.

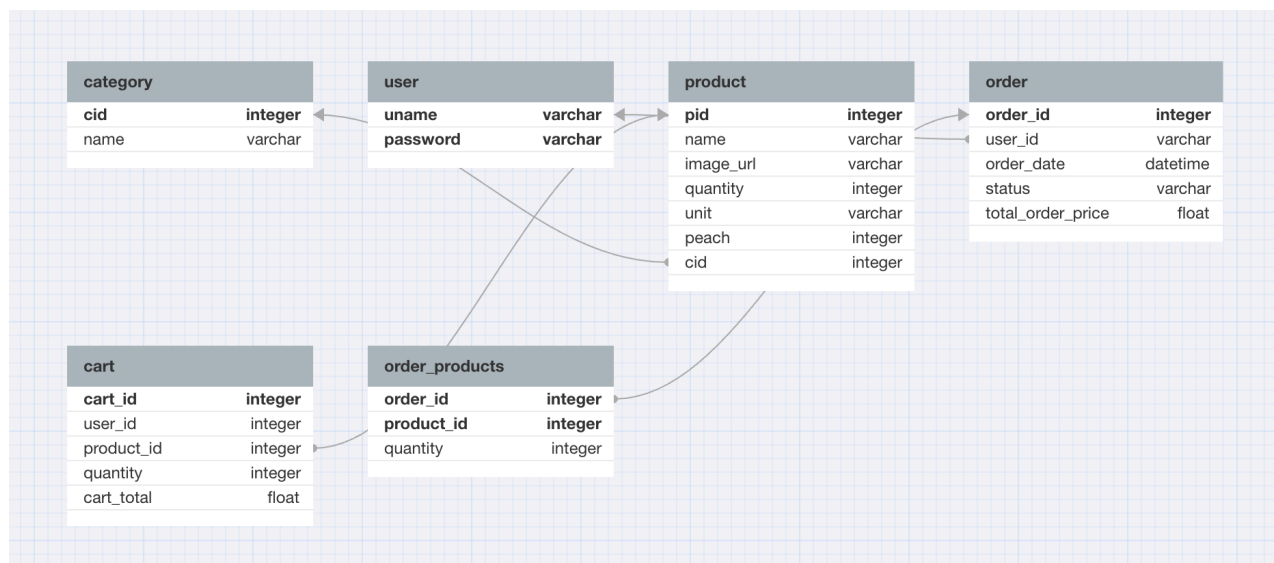
Flask-SQLAlchemy: An extension for Flask that simplifies database interactions by integrating SQLAlchemy, a SQL toolkit and Object-Relational Mapping (ORM) library.

Each of these technologies plays a specific role in building web applications with Flask, from handling HTTP requests to rendering templates and managing databases.

Bootstrap: Bootstrap is a popular front-end framework that provides pre-designed HTML, CSS, and JavaScript components for creating responsive and visually appealing web pages.

Jinja2: Jinja2 is a templating engine for Python that is used in Flask to dynamically generate HTML, XML, or other markup languages within templates, allowing you to insert dynamic content into your web pages.

DB Schema Design



Category Table (category):

cid: A unique identifier for each category. This allows you to categorise products and manage them more efficiently.

name: The name of the category. It provides a descriptive label for the category.

User Table (user):

uname: Unique username for each user. This is used as a primary identifier for users.

password: The password for each user. It's important for user authentication.

Product Table (product):

pid: A unique identifier for each product. This allows you to uniquely identify products in the system.

name: The name of the product. Describes the product to users.

image_url: URL pointing to an image of the product. Used to display product images.

quantity: The available quantity of the product. Helps track available stock.

unit: The unit of measurement for the product (e.g., "kg", "piece"). Specifies how the product is measured.

peach: Price per unit of the product. Represents the price of each unit.

cid: Foreign key referencing the category table. Connects products to their respective categories.

Order Table (order):

order_id: Unique identifier for each order. Allows you to track and manage orders individually.

user_id: Foreign key referencing the user table. Associates orders with the user who placed them.

order_date: Date and time when the order was placed. Helps track order history.

status: The status of the order (e.g., "Pending", "Confirmed", etc). Tracks the order's processing stage.

total_order_price: The total price of the order. Sum of prices for all products in the order.

Cart Table (cart):

cart_id: Unique identifier for each cart entry. Helps manage individual cart items.

user_id: Foreign key referencing the user table. Associates cart items with the user.

product_id: Foreign key referencing the product table. Links cart items to the corresponding product.

quantity: The quantity of the product added to the cart. Helps track how many of each product the user wants.

cart_total: The total price of the items in the cart. Sum of prices for all products in the cart.

Order_Products Table (order_products):

order_id: Foreign key referencing the order table. Associates order items with the order they belong to.

product_id: Foreign key referencing the product table. Links order items to the corresponding product.

quantity: The quantity of the product ordered in this specific order. Tracks how many of each product were ordered.

Architecture and Features

app.py: This is the main file of your Flask application.

db_model: This file defines the structure of your database.

config.json: This file stores admin details or configurations for your project.

requirements.txt: This text file lists all the external modules and libraries your project relies on.

instance: This directory likely holds instance-specific files, such as database configurations.

static: This folder contains static assets like CSS files and images.

templates: This directory holds your HTML templates.

Video :

