
Online Retail Store

Siddharth Gupta(2021355) and Tony Thomas(2021360)



Project Scope:

Our Project focuses on creating a website for an online retail store. The main objective of this project is to develop an online retail store with a robust end-to-end database application as the primary part of the back-end, which **allows customers to navigate through the product catalog and purchase products**. Like standard retail stores like BigBazar, Amazon, and Flipkart, we will sell customers products of different categories, such as electronics, groceries, etc, and offer them the best deals according to their requirements. They can find the products according to their needs, like products belonging to some Company, products that fall in some price range, etc, and make payments. There will be a similar setup for employees where their data will be stored if time permits. Different payment methods will be linked. In terms of data, **we will be storing customer information**, their personal information, phone number, email address and address, both office, temporary and residential, their previous deals, returns and orders, and rewards and coupons available. Data regarding products will be stored like description, price, reviews, and ratings, shipping/transportation details, discounts available.

Technical Requirements:

1. HTML, CSS, Javascript, React:

HTML and CSS will help us customize the **look and feel of the websites**. Javascript will help us render some effects and dynamism to the website, and ReactJS will be used to render some components, such as navigation bars, etc.

2. MySQL:

MySQL is a relational database management system and it will help us manage the database of our customers, middlemen, and admins through **adding, accessing, and**

processing the data stored in our computer database. The database entries will be sorted, created, and deleted with the help of this language.

3. Django/Flask, Python:

Python will be used to operate backends, and we will also use its libraries to render some utilities for easy management for the customer. Django will be used for creating interactive and dynamic web pages, or Flask will be used as it is lightweight and extensible, and this system helps us support APIs.

Constraints

- A customer(identified with email-id or mobile no.) can have only one account in the store application.
- One account can have only a single cart.
- One account can have multiple payment methods saved in the account, so this is a one-to-many relationship.
- The relationship between customers and products will be many-to-many as multiple customers can buy the same products and also one customer can buy multiple products.
- There can be multiple sellers for the same product, while a single seller can also sell multiple products so again this would be a many-to-many relationship.
- The order should be placed only after payment is confirmed. The cart should be emptied once order is placed.

Functional Requirements:

From Customer's side

1. **Customer Login/SignUp:** This feature will help the customer to log in or sign up if he is new to the platform
2. **Search the Products:** This feature helps the customer search the product
3. **View the Product:** This feature helps the customer view the product and displays its description and features
4. **Add to the Cart:** This feature adds the product to the user's cart and keeps track of all the products that were added to the cart
5. **View Cart:** Go to cart to check the added products or remove any.
6. **Buy the Cart:** This feature helps the user buy the entire cart and place the order.
7. **Payment Methods:** This feature helps the customer choose from one payment option, whether it is COD, UPI, Net Banking, Debit Card, etc
8. **Track the Order:** This feature helps the customer get the current status of the order where it is right now.
9. **Best Sellers:** This feature will feature the most frequently sold products
10. **Deals:** This feature will showcase deals according to special occasions.

From Admin's side

1. Enter as Admin
 2. Add Category
 3. Delete Category
 4. Add Product
 5. Delete Product
 6. Set Discount on Products
 7. Add deals
 8. Update details on products
 9. Sales analysis
-