CPS714 Project - ECommerce Management Application

Group 19

E-Commerce web application primarily built using ReactJS, Flask, and SQL DBMS. Additional features using API's (WhisperAPI etc).

Group Members

Member	Responsibilities
Thomas Cervantes	SCRUM Master, Frontend development
Samuel Habib	Backend development - Databases
Vraj Patel	Backend development - Flexible/Webapp
Abdulkadir Ahmed	Backend development - Databases
Gary Lai	Fullstack development - flexible/APIs
Minwoo (Thomas) Jo	Fullstack development - flexible/UI
Bryan Serra	Frontend development - UI/Data Vis
Jananakan Sivaloganathan	Backend development - flexible
Huy (David) Pham	Fullstack - Webapp functionality
Aser Khamis	Backend - Webapp logic

High Level Overview of Main Features

- Admin and customer views
    - Admin can manage their inventory of products
    - Admin view sales data ~~(data visualization through ReactJS)~~
        - Datavis too complex for time crunch, still able to view orders
    - Customers have shopping cart and checkout system
    - Customers can sort through products via hierarchical product types (for instance menswear/womenswear, clothing/shoes/accessories, clothing types, etc.)
    - Customer login and user-specific attributes
    - ~~Customer wishlist functionality~~
        - Also had to remove to meet allocated schedule
    - ~~Loyalty rewards through points~~
        - Extra feature for extra time only
- Timed/seasonal sales on the storefront
    - Replaced with discount factor functionality
- Cloud data storage
    - Stored on firebase
- Customer to admin chat/~~chat bot feature~~
    - ~~Chat bot through whisper API~~
        - Chat bot was too complex to implement within only 2 sprints, could be a project on its own
    - Ticketed message system for direct customer to admin interaction
- ~~Size prediction function (similar to h&m/zara sizing recommendations - api or general population data for clothes sizing~~
    - Transitioned to miscellaneous storefront for secondhand items and collectibles

Technologies

- Frontend: ReactJS
- Backend: Flask
- Database: Firebase NoSQL
- Additional APIs: ~~WhisperAPI~~, others as needed

Features and Views

Admin View
- Inventory Management: Admins can efficiently manage their product inventory, including adding, editing, and removing products.

~~- Sales Data Visualization: Utilizing React and Flask, we'll implement data visualization tools to provide admins with insightful sales data.~~

Customer View
- Product list view: View all products that are available for purchase. Customers can limit the products that they view via product categories (e.g. clothing or shoes or accessories) and sort the products by product attributes (e.g. price, popularity, release date)

- Account registration/email verification capabilities: secure account creation, 2FA through emailed verification codes and encryption on all sensitive data (passwords).

- Shopping Cart and Checkout System: Customers can add products to their cart and seamlessly proceed to checkout.

- User Authentication: Implement a secure user authentication system allowing customers to create accounts and log in.

- User-Specific Attributes: Customized user profiles enabling customers to save personal preferences, track order history, ~~and earn loyalty rewards.~~

~~- Wish List Functionality: Allow customers to create and manage wish lists for future purchases.~~

~~- Loyalty Rewards System: Reward customers with points for every purchase, encouraging repeat business.~~

- Timed/Seasonal Sales: Implement timed and seasonal discounts to boost customer engagement. **implemented discount feature instead

Additional Features

- Cloud Data Storage: Utilize cloud storage solutions for secure data storage, scalability, and accessibility.

- Customer-to-Admin ~~Chat/Chat Bot~~ Feedback: Enable real-time communication between customers and admins through a chat system, through ticketed messages and feedback inbox ~~possibly integrated with Whisper API for advanced chat bot capabilities~~.

- Ticketed Message System: Establish a ticketed message system for direct, organized communication between customers and admins.

- ~~Size Prediction Function: Incorporate an AI-driven sizing recommendation system, similar to H&M/Zara, utilizing either API integration or general population sizing data.~~