**GROUP 5 : SHOPAHOLIC**

**SOFTWARE ENGINEERING PROJECT WRITE UP**

Keisuke Lester, Austin Thrash, Mahnoor Bokhari, Vamshi Ponnala, Mayank Gohil

---------------------------------------------------------------------------------------------------------------------

**OVERVIEW:**

Our goal for this project was to create a webpage that implements a database, and is available to anyone due to hosting it on a server. We decided to create a clothing brand website, where the customer can purchase clothing and accessories, named Shopaholic. Our site would have a login function, and two personas (The Customer and the Admin) that ensured it could function alike to a real clothing brand site. We were required to create a sorting function for both products and orders, which we implemented with a search bar as well as drop down sorting menus. Overall, although we had a slow start and difficulty adjusting to the project due to no prior experience, we were able to complete a fully functional web page that followed the rubrics requirements as well as some of our own touches, only with having to do a few reworks when working on the database connectivity and a restart regarding the frontend.

---------------------------------------------------------------------------------------------------------------------

**TEAM:**

Team Members:

Mahnoor Bokhari, Mayank Gohil, Keisuke Lester, Vamshi Ponnala, Austin Thrash

Project Manager: Keisuke Lester

The project manager is Keisuke, who will enable the group to stay on track, provide any revisions to team assignments if needed, facilitate meetings, and ensure the app is not behind in production. Keisuke will also write up any necessary documentation for the project, such as this project scope write-up, but will still communicate effectively with the group to have it proofread and ensure the team all agrees with what's written. The project manager will also work on creating the presentation flow and topics so that the presentation will go smoothly and convey everything we need to touch on.

Server Manager: Austin Thrash

Hosting the webpage publicly, one of the requirements, was fulfilled by Austin. The project made use of AWS, Amazon Web Services, for publicly hosting our webpage, using the EC2 free tier servers. The setup of the webpage server and connectivity was done by Austin, where he used one EC2 server for the front and backend, and another to host the mySQL database. Further on this he is using cookies and extensions to help deploy the site. However some of the connectivity of the backend and frontend became an entire group effort due to the issues we were having before. This being said Austin was the one who figured out the connection with what we had originally planned. Any added security, such as a certificate from certbot, or SSL will also be handled.

Backend Developers: Mahnoor Bokhari and Vamshi Ponnala

Both Mahnoor and Vamshi worked on the database; the database holds several different tables such as User login info, Product Info. Mahnoor and Vamshi also worked on the functions needed for each sorting type (ascending/descending by price, keyword search), tax calculations, and final price calculations. This also includes creating the testing database entries to fill the database with products (for testing and demo purposes). The backend was also helped out by Mayank as communication between these two parties were critical to ensure the webpage functioned together.

Frontend Developer: Mayank Gohil

Mayank worked on the frontend of the webpage, working closely with javascript to create a user friendly UI with all the content that we need, as well as learning how the database and javascript interact to display entries from the database onto the webpage. He drafted several templates and even had to restart due to accidental deletion of a previous frontend UI he had created. He got most of the basics working on the website to create a usable page on ReactJS. Throughout the course of the project however, he also contributed significantly to the backend as well, and worked alongside Austin and the database developers to create an interactive and dynamic webpage. One of the most notable contributions besides all of the CSS for the webpage was his ability to work side by side with the rest of the members, and working closely with Austin to help create a dynamic webpage that interacts with the backend.

I would like to note that these are generalized roles, as the project progressed, a lot of the members strayed from their original placements to help the webpage in parts we were struggling on. So although these were their focus, the webpage overall was a team effort, and every single part took advice and ideas from every member, causing us all to be involved with every part of its creation.

---------------------------------------------------------------------------------------------------------------------

**TIMELINE:**

Although this is not a formal timeline, there are descriptions of the process we took to create the project. The list is in order of what we completed and the issues we encountered while creating the project, such as things we had to change from the scope, or our understanding that we were not putting in as much effort as we needed to for the project to be completed within the time restraint.

WEEK 2

The first meeting between us was just getting acquainted with one another right after the group assignments were posted to ensure we stayed on task. To ensure a schedule we also met the next week as well, deciding the roles of each of our members according to their wishes or strengths in particular areas, however we did not banish the idea of crossover, it’s important that each part of the project has ample help from every member involved.

WEEK 3

The first task was creating the project scope, however none of our team members had any experience with creating a server based webpage, as well as no experience with databases which was one of the key factors behind the project. The project manager sent several emails to the professor to gain some insight on direction and went from there with the project scope. Although the project scope was a rough draft, you'll find we stayed quite true to it, only deviating due to slight logical errors as we didn't quite have the complete insight on creating a webpage when the scope was due.

WEEK 4-6

The next steps were a slow process, it’s obvious having full investment into the project was impossible due to each member's workload. We ensured to maintain communication and meet for updates, however our server manager was having difficulty setting up a working server. To fix this we reviewed several online video tutorials as well as him meeting with the professor to discuss what could have been going wrong in his process. After this he was able to set up a functional page, using wordpress as our knowledge over the subject was not advanced enough to connect ReactJS just yet. This was a great start for our project and helped us gain some confidence back.

WEEK 7-8

Our next step was to create a template for what the front end might look like. The project manager drafted up a list of features and descriptions of each feature to give the front end developer some insight to what needs to be created. Once the template was created, he started working on its actual development. Our database team had no experience with databases, so we placed two members onto it to help with learning by collaborating with each other and sharing knowledge. One database developer took it upon himself to enroll in a database course, and studied this over spring break with his own free time.

WEEK 9-11

With the return from spring break it was obvious our plans to keep a schedule and consistent meetings were not holding true, as well as being slightly behind in the project’s construction. With one month left for development time, we decided to step up our game and ensure we can power through to a finished working project. The frontend developer showed us his almost completed webpage, which was more than we had expected, however he also had a few issues as stated in the members list, in that his progress had been deleted prior and he had to restart from scratch, yet was still able to come out with an amazing UI. We created a github so that our members can view the code of the frontend provided.

-------------------------------------------------------------------------------------------------------

We finally set the deadlines for each member, and connect the database to the front end:

Austin (Deadline: 3rd Apr)

- Try to deploy the unfinished website onto the server

Noor & Vamshi (Deadline: 3rd Apr)

- Work on the database

- Create all the columns and put in the information needed for the site(Let me know if you need to know what info needs to go on the database)

Mayank (Deadline: Sunday; Latest: Monday)

- Finish up the website

- Work with the backend guys to connect the database with the website

Kei

- Work on the project write up (no rush)

-------------------------------------------------------------------------------------------------------

WEEK 12

Our next meeting took place on April 4th, to review what we had completed by the deadline. The backend met to work on connecting the database with the site as we thought doing this in person would help productivity! Austin created keys for the server so that everyone has access to it through ssh and can work on it together.

We then reserved a study suite room for Wednesday, and emailed to book our presentation. Within the meeting we placed two more deadlines:

Finished software deadline: April 20th

Presentation Preparation: April 21-24th

And then scheduled another meeting for Friday as well as a list of things we will try to achieve during the meeting:

Few things we need to try to finish are:

- Figure out a way to connect the database with front-end (Visual Studio Code)

- Add inventory to database (Vamshi)

- Read, write, update and pull data from the database (Once we can connect it)

However during this meeting we ran into a few problems and decided our original approach was not going to work, instead We decided that we are going to attempt to rebuild the website using react and javascript, since we ran into multiple problems using html and php. On top of this we are going to try and switch to firebase to host the database. Austin decided he would start on the admin view while Mayank works on the amazon clone watching an amazon clone tutorial video that could help us better understand what went wrong when connecting the frontend and backend. With that we booked another room for Monday.

On sunday Austin updated us that he had gotten great progress rewriting the webpage with reactJS, as we were finally able to have products into the database, while currently working on the login/registration.

WEEK 13

On monday we met as scheduled, in this meeting we discussed our shortcomings, and how we were able to progress over the weekend. Austin showed us how he was able to connect the original database with the frontend. We discussed having to write an API that would automate the process of adding products to the database, and reading them from the database. Further on this Austin talked about setting up routers so the data on each page would be consistent, as well as setting up a way to run both the client and server at the same time instead of separate. Mayank and Vamshi will be working on a mock database to fill, giving us room to test our website with valid data, and Noor would focus on creating the searching and sorting for the products since it's one of the required features.

-------------------------------------------------------------------------------------------------------

From this meeting we set up another meeting for Wednesday after class, and a goal list that we want to focus on for this week:

Deadline : Friday(15th)

Austin:

- Login setup

- Routers

Vamshi & Mayank:

- Set up the database (inventory)

Noor:

- Sorting/searching for the products

Mayank:

- Convert HTMLs to JSX

- Help Austin with backend

Kei (No Rush):

- Work on the presentation (maybe create slides??)

On Wednesday we met for another 2 hours in person, in this we created another set of updated deadlines for each of the members whilst making great progress.

\*\*(Updated)Deadline is Friday(15th)\*\*

Austin:

- Login setup

Vamshi:

- Set up the database (inventory)

- Find a way to parse products name for search (not case sensitive)

Noor & Austin:

- Sorting/searching for the products

Mayank:

- Create admin & user dashboard UX

- Create search bar and filter (for sorting)

- Figure out a way to let admin change product images which updates the database

- Help Austin with backend

Kei (No Rush):

- Work on the presentation (maybe create slides??)

-------------------------------------------------------------------------------------------------------

On Friday we met for the final time this week, to be motivated to get our work done by sitting in each other's presence. We booked a room for 3 hrs and made a decent amount of progress. Austin created a changelog of the current state of our webpage and what needs to be done

-------------------------------------------------------------------------------------------------------

Version Changes:

* admin dashboard loads all products on site. Displays title, price, image, and stock
* attempting to access unauthorized pages, i.e. not signed in user attempting to access user dashboard, should now properly redirect to home page
* ability to update product information stored in the database from the admin dashboard

Next Version:

* add and remove products from admin page
* add and remove items from user cart
* will work on preparing code to connect the front end and back end to new pages such as user dashboard and admin dashboard.

We also were able to finish a beta for the admin view

-------------------------------------------------------------------------------------------------------

WEEK 14

Our plan for week 14 was the same as the previous week due to our success. We planned to meet 3 times this week to wrap up on our webpage construction and shift into planning for the presentation. On Monday we were able to finish the construction and stylization of the search bar and planned out our final stretch goals :

Deadline: Apr 20th

-------------------------------------------------------------------------------------------------------

Austin:

- Finish up the checkout backend

- Work on the order part

Mayank:

- Create a user dashboard

- Add a coupons part in admin dashboard

- Fix admin dashboard styling

- Make changes to the UX/UI (After everything is done)

-------------------------------------------------------------------------------------------------------

On Wednesday two of our members were unable to attend the meeting, due to this, we agreed to have the meeting in person, but also have an online meeting later that night to work on what we need to get finished.

On Friday we met up to work on deployment and encountered a few issues, our entire meeting just consisted of working on the stylizing on some portions of the site (due to the translation changing some portions) and trying to get it deployed.

Finally on Sunday the Project Write up and slides were due and we finally got our site deployed and working. We are going to spend Sunday fine tuning the site and fixing any bugs/errors that we find with the way that it works.

---------------------------------------------------------------------------------------------------------------------

**PERSONA LIST:**

Since our project has two different personas we plotted out what the extent of each one should be, this is also briefed over in the Feature List section, but we thought it was important to include this to ensure we have well defined our personas. Without defined personas it's hard to imagine what the webpage could look like, and what different versions you'd need to make for the different types of accounts.

-------------------------------------------------------------------------------------------------------

Admin:

* One account, Username Admin
* Has view to the Order History and Pending Orders
* Has the ability to Edit,Remove, or Add products

Customer:

* Created User, must register if one doesn't have an account
* Unable to checkout if not logged in
* Ability to add to cart and browse products
* Ability to check out/purchase products

-------------------------------------------------------------------------------------------------------

---------------------------------------------------------------------------------------------------------------------

**FEATURE LIST:**

We composed a list of all the necessary features for the web page to help our Front End design a base webpage that can be used with all of the needed functions. By having a put together list of features, it was easy to identify what kind of physical things the webpage would need to have to ensure it works as intended. This was also to keep track of what still needed to be added and what is completely implemented.

-------------------------------------------------------------------------------------------------------

Necessary:

General:

* List of products page (page that will show all the products currently in the store)
* Products have images, price and quantity (most likely stored in database)
  + Ability to sort the products by Price, and by which is in stock (two sort features for the products list page)
  + A search bar to search for keywords (I think the search should be available anywhere on the site

Customer Persona | the normal view of the site:

* + Register for accounts (registration page)
  + Login to accounts (login page, button on the home to login which takes you to the login page, likely will have a way to reset your password, or register if you don't have an account)
  + Add items to cart (add to cart button under each product, grayed out and replaced with “sold out” if the product sells out)
  + Show items in the cart page (cart button on the page bar?)
  + Ability to check out the items in the cart (forces you to sign in before you check out)
  + Check out will include calculated taxes, ability to add discount codes, summary of order( summary of order will list the description and picture of every item added, and the price which has a tax added to it, the subtracts any discount codes, and shows the total of the order)
  + Ability to place the order (assuming we don't take credit card info)

Admin Persona | this version of the website will only be viewable by those who sign in using the admin login info:

* Login to admin account (likely only 1 account, don't need to register since it'll always exist)
* Add or remove products from the store (product list will now have an additional button to remove or add and products)
* Modify any products currently on the store (in addition to the remove button, an edit button)
* View current orders (not fulfilled)( maybe replace the cart button on the home bar?)
* List of all orders (fulfilled)
* Ability to sort orders by date, customer name, or price (I believe this is only for the fulfilled orders)

Additional:

Customer Persona:

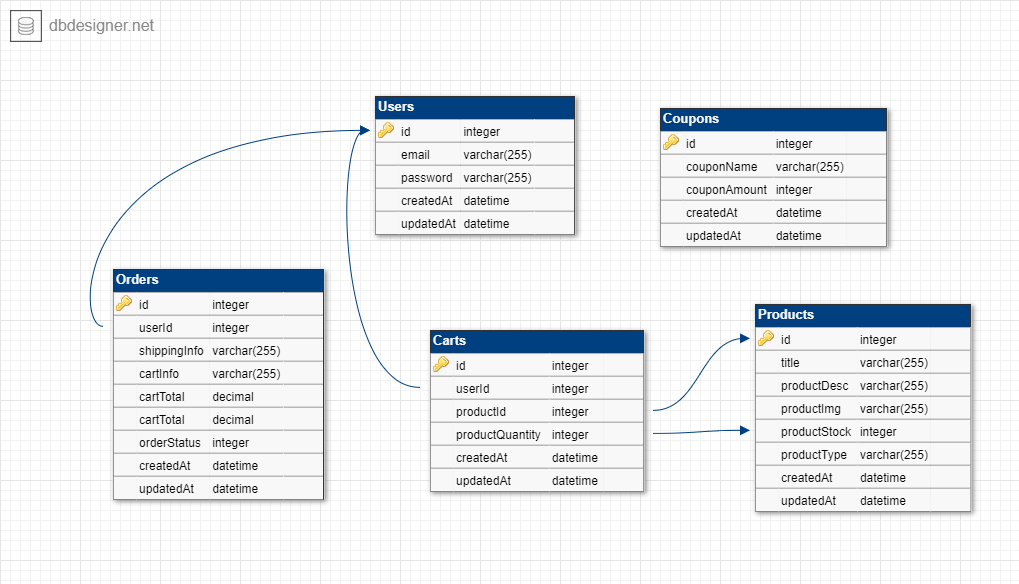
* Some way to wishlist products and view their wishlist

-------------------------------------------------------------------------------------------------------

---------------------------------------------------------------------------------------------------------------------

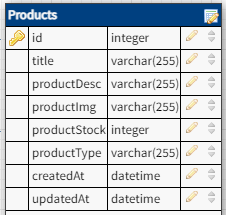
**DATABASE DEVELOPMENT:**

Our database had a very rocky start, the database was created on mySQL but our first version we had extreme difficulty getting it to connect with the front end. This caused many group meetings and revisions of the database and the set up of the entire server. However we finally got our original idea to work and from there built onto it. We used models to structure the database, such as users and products :



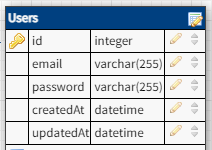
PRODUCTS:

Each product has several different entries and datatypes to help us maintain all the information about the product. This includes the stock of the product to show when the product has become sold out, as well as the product price, image, name, and description.



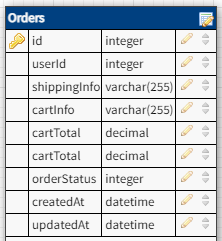
USERS:

Our user data type helps us recognize if the user is currently registered or not, and which user has what added to their cart. Users also have a password for their account, this password is actually hashed for greater security, besides this users have a username, and a boolean field that confirms if the user is an admin or not



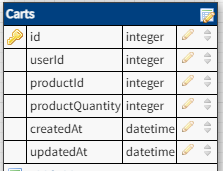
ORDERS:

Our order data type helps us sort orders for the admin view, as well as display them for the user dashboard. We added a userID and order status type to help track who placed the order and if it's been completed. The cart info is to show what was purchased and the Total shows the profit/charged amounts on the order



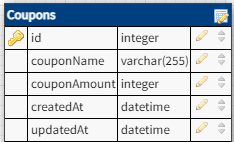
CARTS:

We created a cart database to ensure that each user would maintain the items previously added to their cart before, we have a cart ID and the user it's associated with, as well as the products and their quantities.



COUPONS:

Finally, our last datatype is coupons, we created this to store any current coupons so that we can always check if a coupon name is valid at check out.



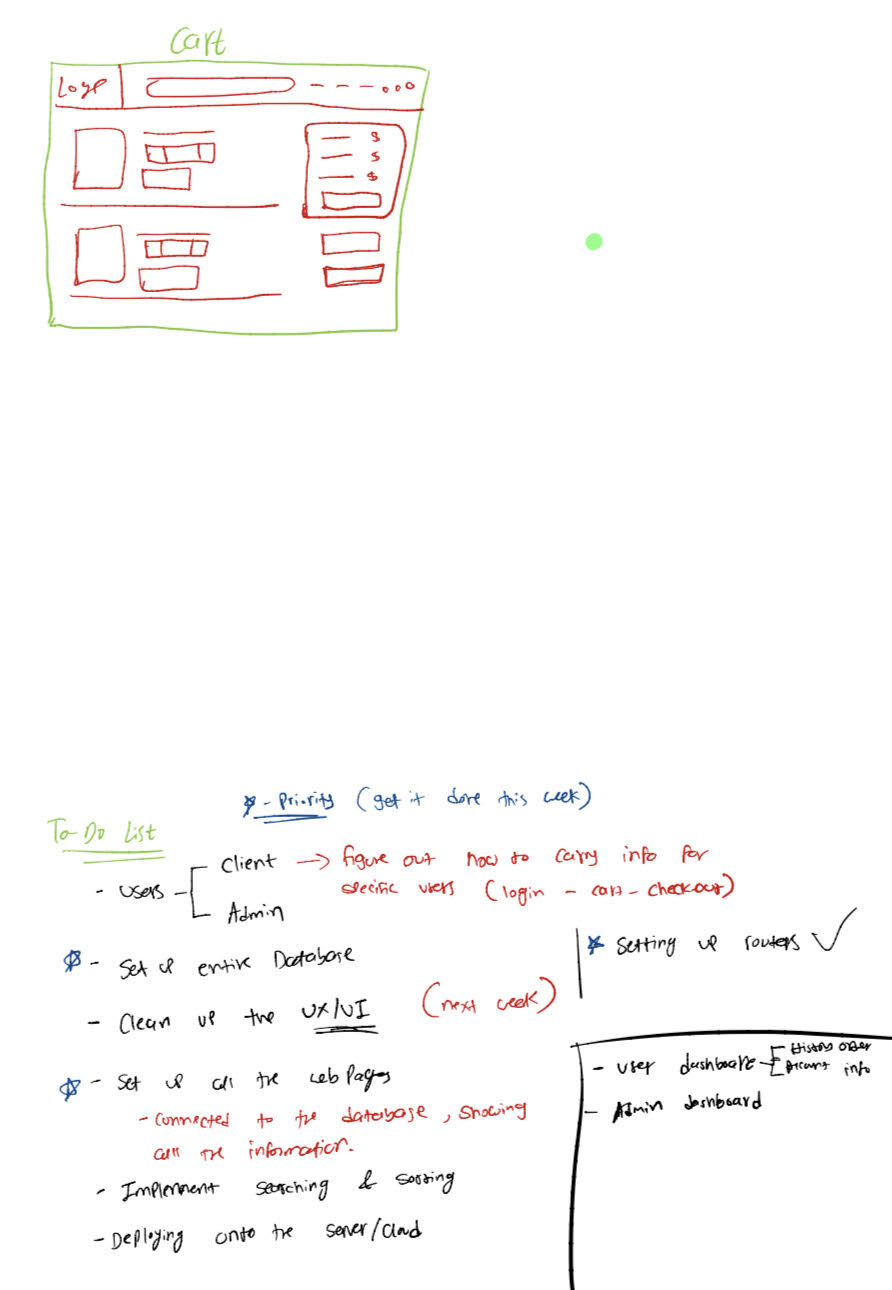
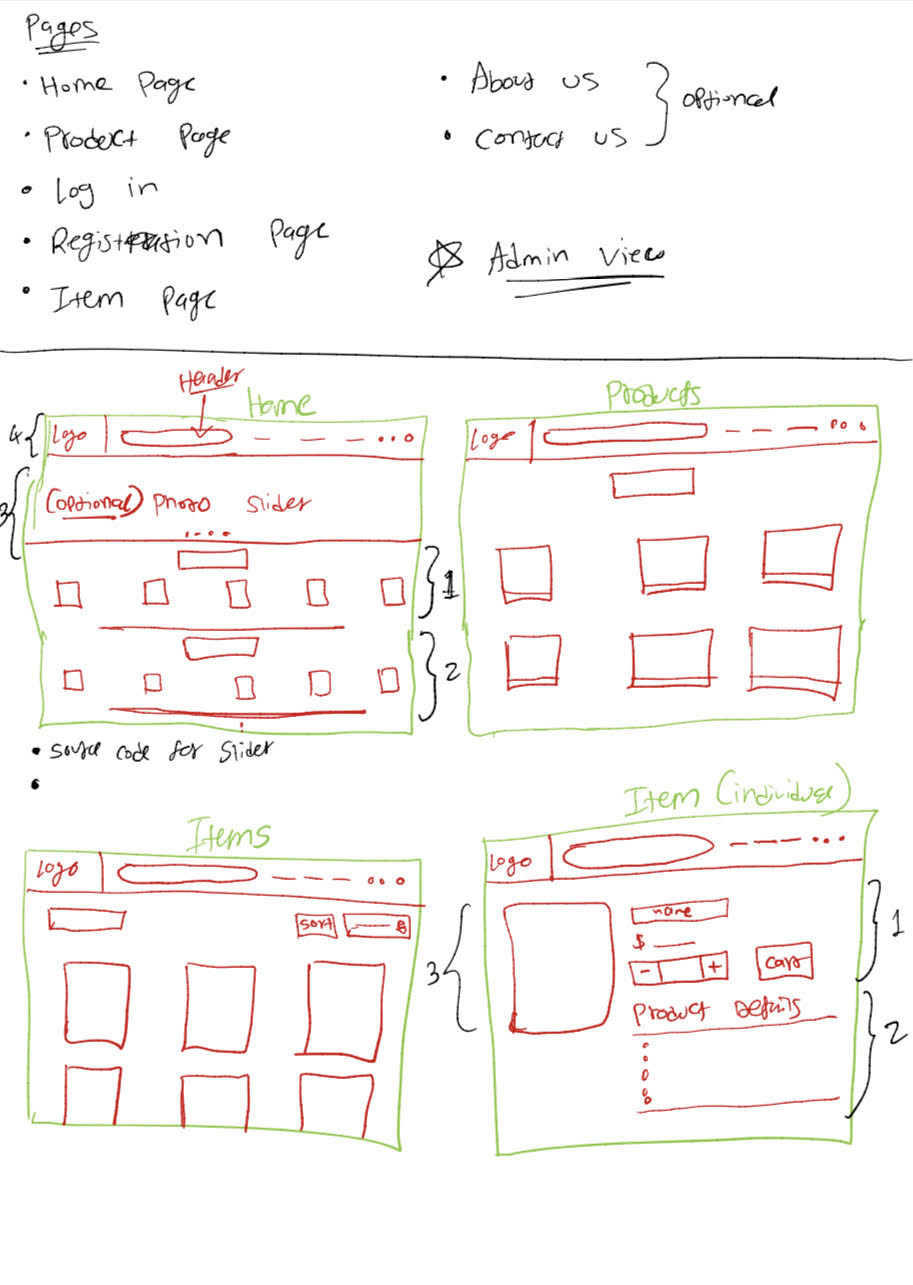
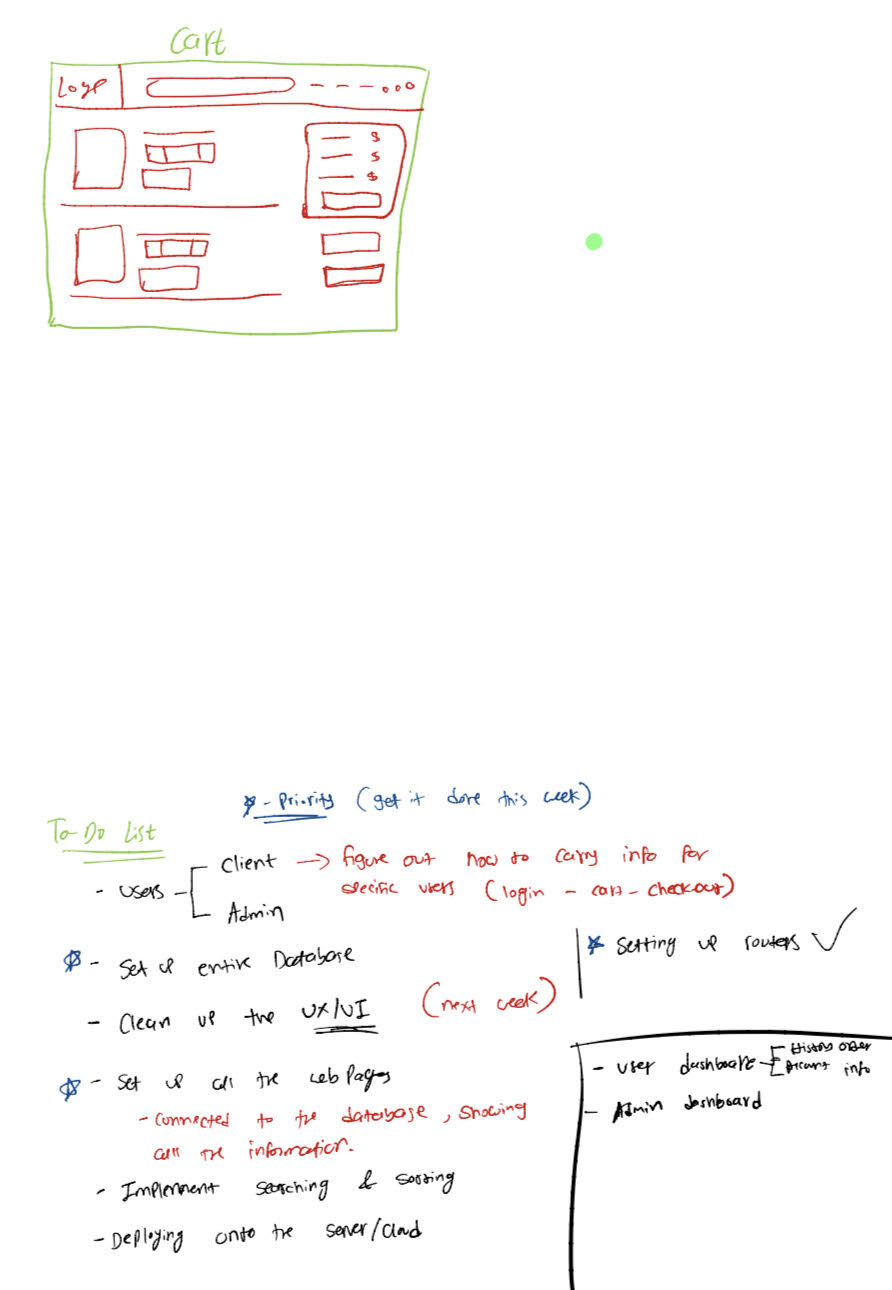
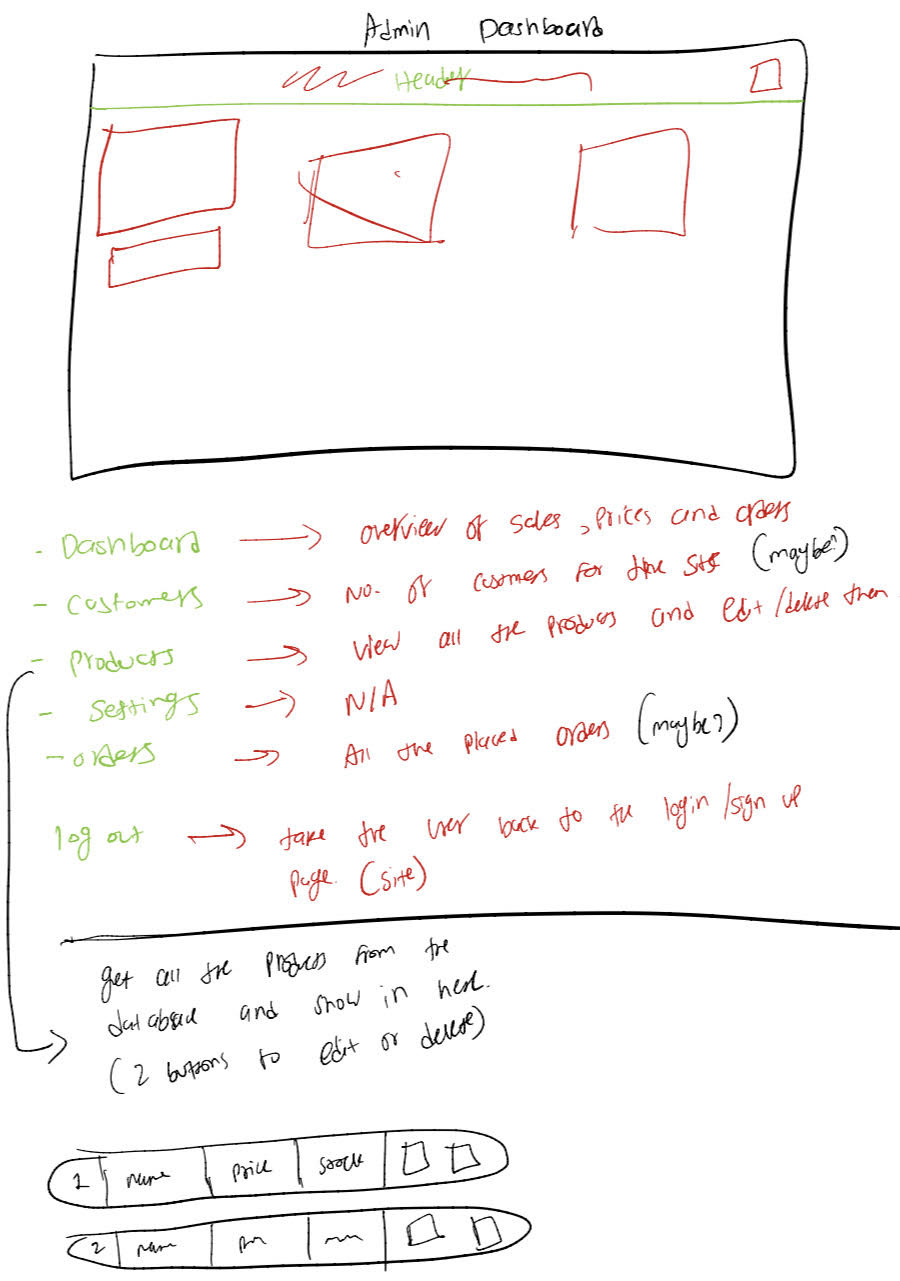
We also created mock data for our database to showcase how the data is shown and can be deleted/added onto. This is important to show how our website could work in a real life scenario, carrying all the fashion products you'd need for an actual online shop.

---------------------------------------------------------------------------------------------------------------------

**FRONTEND DEVELOPMENT:**

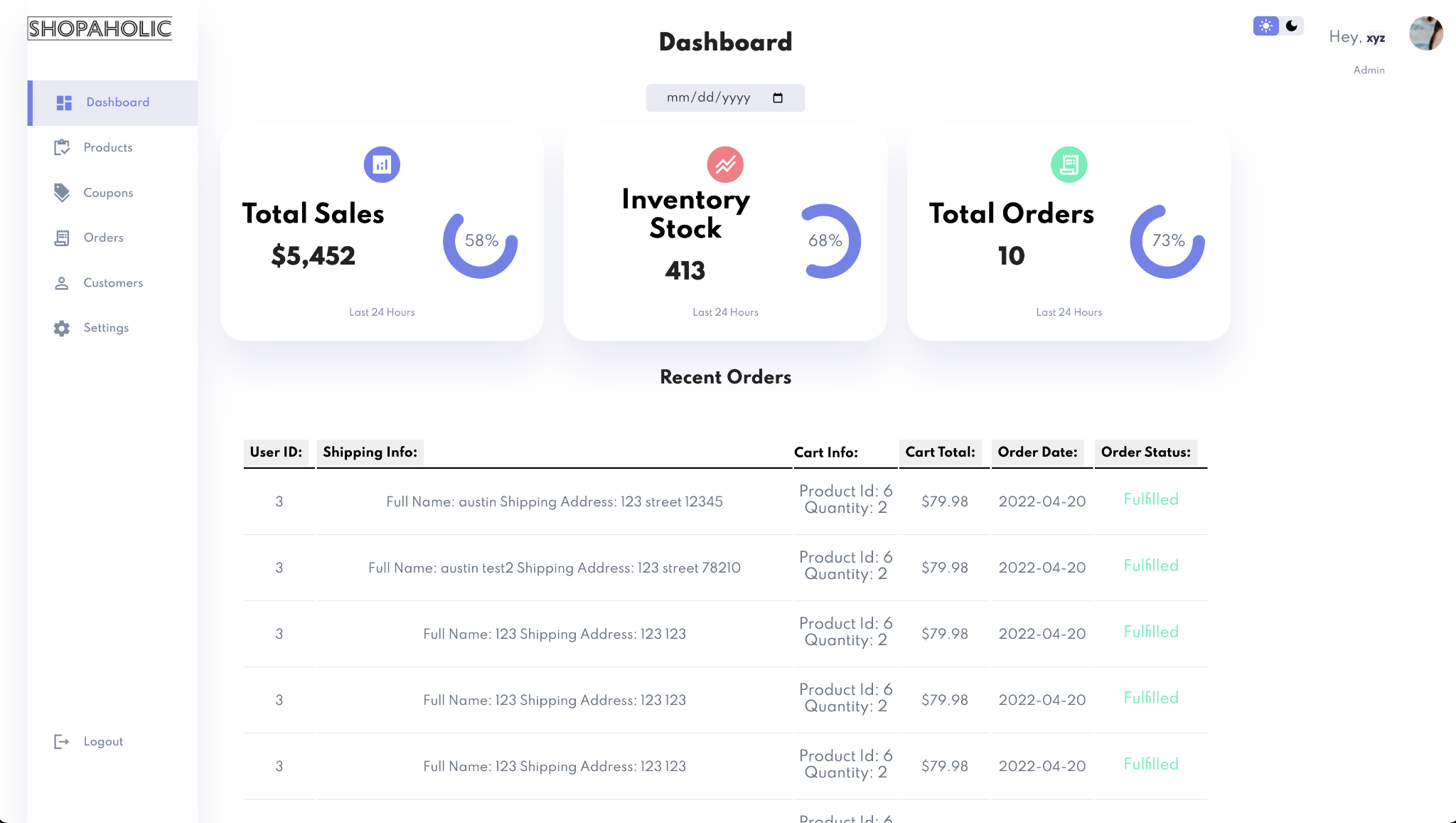
The front end development was relevantly smooth. Mayank, our front end developer, did have to recreate the front end from scratch when he lost the data for it, however the second version came out a lot more refined and professional than the first version. This front end was for the customer persona including a cart, login, products page, and about us page. Our admin view came later, which we created as a one page, containing current orders, income, completed orders and a way to edit current products. We also eventually added a User dashboard view, so that users can view their currently placed orders as well.

Planning of the view of the webpage :

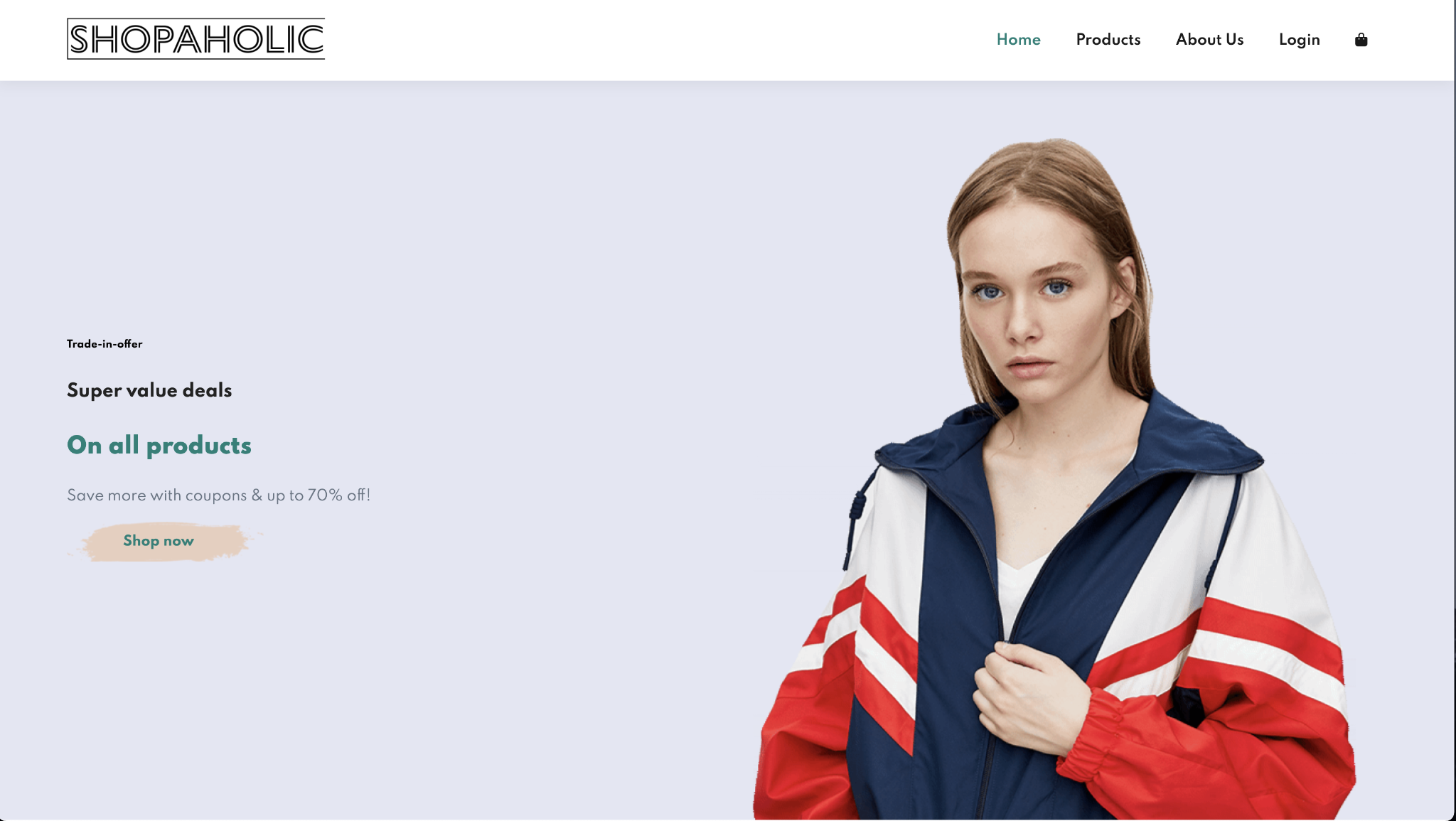


While connecting the front end to the server, a lot of the HTML files had to be converted to JSX, we used an online converter for this which worked efficiently and got the web pages correctly converted. To save time, most of the web pages used the same CSS to avoid having to style each page individually. By the end of the project, we were able to get every webpage to work cohesively with one another and having graphically pleasing UI’s and pages :

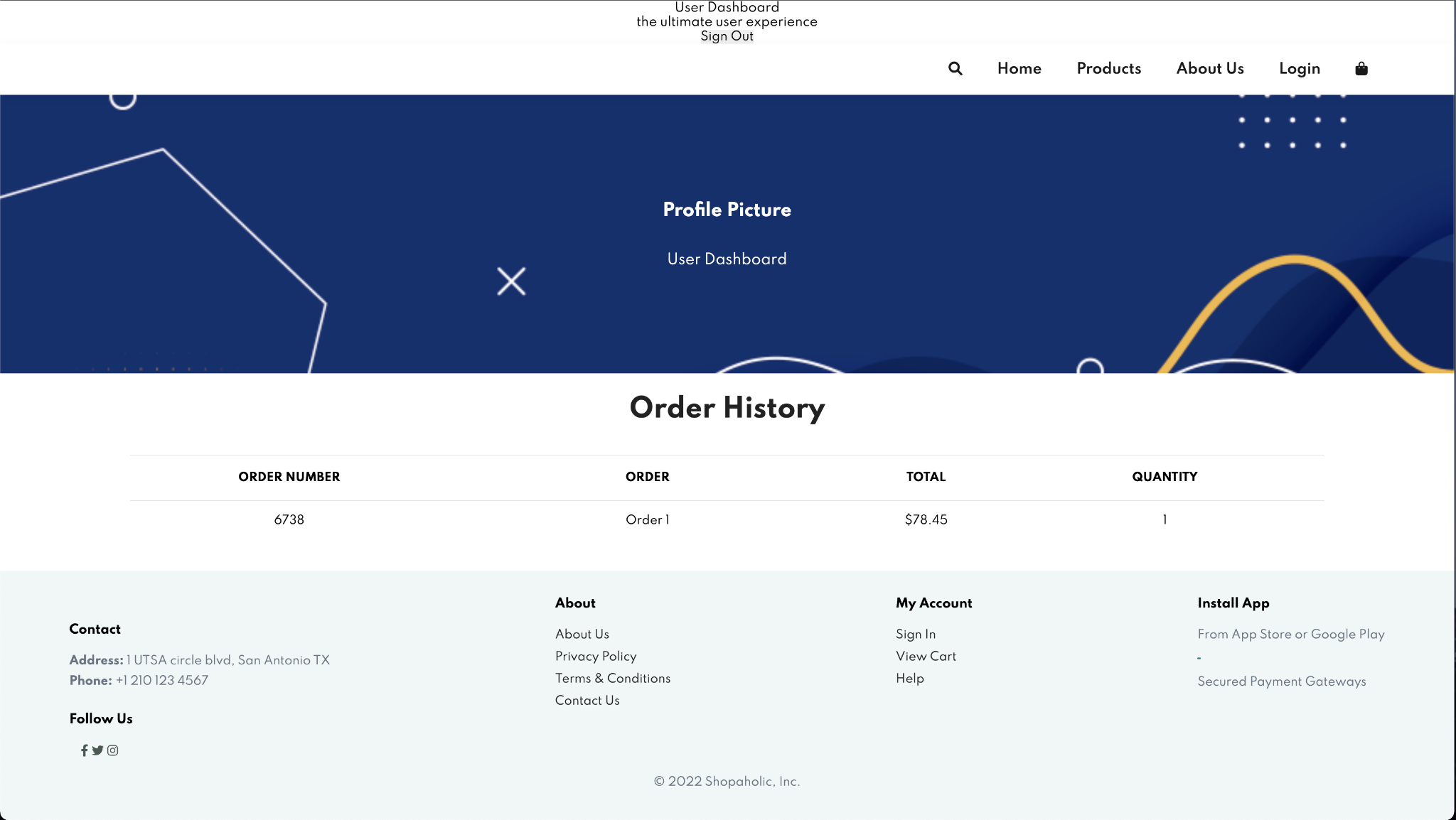
ADMIN VIEW:



HOME VIEW:



USER DASH:



The final touches were adding a search bar and sort to the products page as well as touching up on our stylization due to the translation, Mahnoor helped work on this and fix up all the visual errors and bugs we had with the front end of the site.

---------------------------------------------------------------------------------------------------------------------

**HOSTING AND SERVER SETUP:**

The creation of the server, and connection between each of the working parts of the webpage was mostly done by Austin. To host the majority of the webpage we used an AWS EC2 Free Tier server.

FRONTEND / BACKEND :

The server on the port listens for POST and GET requests so it can pull and push onto the database. In simple terms, the server looks for an update made on the webpage, then changes the database according to this change.

Austin then worked on a router/route that can go through the pages of the website. This helps carry the products and login info throughout the pages when a user would switch between them and effectively connects our Frontend and Backend as the webpage is able to dynamically show data straight from the database.

During development, upon successful login, the backend would create a session cookie that would carry the user session from page to page. This cookie was used to do a number of things such as assign userId when adding a product to cart or load the correct user dashboard.

ISSUES :

Multiple issues arose when deploying the site. First issue was that the front end and back end were hosted on different services. This was a problem because heroku, the service used for hosting the backend, did not allow setting cross site cookies. So to solve this, Austin created another EC2 to host the front end and back end together. Our setup now consists of one EC2 instance to host the front and back end, and another EC2 instance to host the MySQL database.

The second issue arose when the express-session extension on the back end would not communicate user data to the front end. This is a big issue because although the cookie was created in the session, it contained no user data to use for later backend requests. The solution for this was creating a json web token in the backend and sending its information to the frontend. The frontend will then store the token in the session storage for later use/validation.

------------------------------------------------------------------------------------------------------------------—

**TESTING:**

Our first tests were just visual tests, ensuring that the frontend looked as expected to what we had wanted. A few issues had to be changed, as mentioned in the Frontend section of the write up. An in depth explanation of this is, due to having to translate the front end from HTML to JSX and using an online converter to do this, not all text converted properly and it caused for some portions of the site to look incoherent or bugged. Mahnoor and Mayank manually reviewed each page within the site and fixed any formatting issues it had (one of the issues was having a 1 product per line, causing just a long vertical line of products which was visually unappealing and hard to look at).

During the creation of the site we also realized several issues, as listed in the Issues tab under the Hosting and Server Setup section. We also discovered issues with dynamically displaying the products, and then also when creating the user cart.

Our final day was going to be spent bug testing (4/24/2022) however the project write up is due the morning of this day so I am unable to write any bugs we further discover and fix on sunday.

---------------------------------------------------------------------------------------------------------------------