Front-End Documentation

Login:

By: Michael Alaniz

The login page is simple. It uses Google Authentication in order to add security and a login feature to the site. The user signs in with their Google account, then using the Google OAuth npm library, it grants access to the user in order for them to access the site.

This is a simple React component that is added to the project and can be adjusted by just adding a new client ID key to the App.js page.

This React page/component will be the first that is accessed within the website. It will then redirect the user to the homepage.

Location: SlugShop < slugshop < src < pages < Login.jsx

Home-Page:

By: Danny Nguyen, Jodh Khubbar

This page is used as the main home page for the web application. It consists of a navbar that has options to take the user to many different pages including the home page itself, a profile page showcasing the user's information and all their listings that they posted, as well as a listings page, and a search bar.

The home page features many random listings that the user can scroll through. To do that, the program fetches a request from the back end to retrieve the information of the listing, which includes the main picture, price, and name. It then formats the listings neatly onto our home page.

Clicking on each listing takes the user to a different page, where the main listing is shown. This is done by adding the address of the listing page to the listing on the home page. (All from json)

The home page is built off of React and styled with CSS.

This page is the first page that is accessed within the page after the user signs in.

Location: SlugShop < slugshop < src < pages < HomePage.jsx

Profile-Page:

By: Danny Nguyen, Jodh Khubbar

This page is used for the user to keep track of all their listings, as well as have their information stored. Just like the home page, the profile page fetches a request from the back end to retrieve the information of the listings and formats it neatly onto the page. The profile page will include all their prior and current listings, as well as their school email, name, and profile picture.

The profile page is built off of React and styled with CSS.

Location: SlugShop < slugshop < src < pages < Shop.jsx

Individual Listings:

By: Michael Alaniz

This page is used to display the user listings made by the individual people. It shows the name of the listing, the price, the condition of the item, the size of the item, and at least one picture.

This page does not specifically take anything from the user, instead it fetches a request from the backend and displays the correct information about the listing.

This component is used within the home-page. It can be used, also when a user searches for specific listings within categories.

Using Material UI and TailwindCSS, we built the overall presentation of the webpage.

This page sends a ‘GET’ request to the backend and retrieves it from the database. This will output the given user assigned ID’s information.

This page will update and be different depending on the user ID. The listing in the front page will redirect to the individual listing that is stored within the Database.

This uses the Navbar component within SlugShop < slugshop < src < components < Navbar.jsx
and uses the footer that can be found in SlugShop < slugshop < src < components < Footer.jsx

Location: SlugShop < slugshop < src < pages < UserLisitngs.jsx

Create Listings:

By: Michael Alaniz

This is that page where the user goes in order to create different listings websites. The user clicks on the ‘Create Post’ button, then inserts the information that is listed.

Once the user posts the listing, it sends a ‘POST’ request to the backend server. This then updates the general listings of the home page and search results.

Using Material UI we created a page that prompts the user with a form that uploads the information to the backend server. The user can use this in order to add their specific listing to the webpage.

It gives options for categories, price, description, and the name of the listing. Using JavaScript, it then sends a request to the backend, which takes care of taking to the Database, then storing the information. This can then later be retrieved by the front-end.

This uses the Navbar component within **SlugShop < slugshop < src < components < Navbar.jsx**
and uses the footer that can be found in **SlugShop < slugshop < src < components < Footer.jsx**

Location: SlugShop < slugshop < src < pages < CreateLisitngs.jsx