

Group Project Final Proposal

E-Commerce Application

Chris McCorkle, Elian Zamora

1. Detailed Plan Documentation

List of React Components

- **React** - Front end dev and handling the logic behind the products displaying and search functionality. Also allowing users to add and delete new products as well as edit products as needed.
- **BootStrap** - Used for creating a seamless front end project development to have more control over the appearance and components in the app.
- **SQL Database** - This will keep track of users and their listings, allowing for other users to track their listings and post their own. Listings will be bound to individual accounts
 - Database will have 'uuid', 'field', 'product name', 'product description', and possibly 'image'

Functions and Logic

- **Adding** - Function will add a new product listing and display to the user and other users of the site
- **Deleting** - Function will delete a listing when completed as marked by the user or when the product has been 'sold'
- **Editing** - Function will allow a product listing to be edited by the user if a certain listing is wanted to be replaced by a different one
- **Single state-lifted master list** - This will allow the site to keep track of the user's specific products and act as their list of products. By using a single state-lifted 'database' we can easily add/remove and update listings as deemed by the end user.
- **Login** - This will allow for users to sign into accounts and login to theirs to view their listings
- **Search** - This will allow for users to search for specific product listings. If no item is available a message will be displayed.

API Details

- **GET** - Used for updating / retrieving the current data the user will need in regards to their listings
- **PUT / POST** - Used for adding a new product to the list, or updating a product that is already inside of the list
- **DELETE** - Used for when a user would like to remove an item from the list
- Along with the API requests, our system will respond to these using the status codes, (200, 400, etc...) to provide feedback to the user when they make changes or modify the list in any way, or if something goes wrong an error message will be displayed.

Endpoints

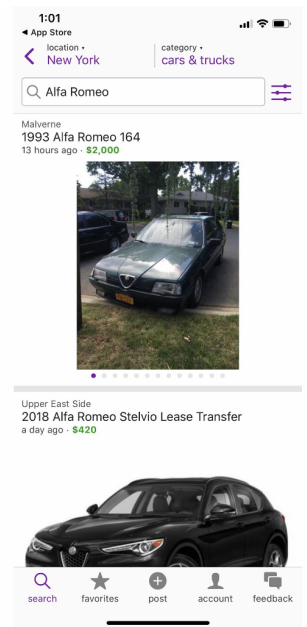
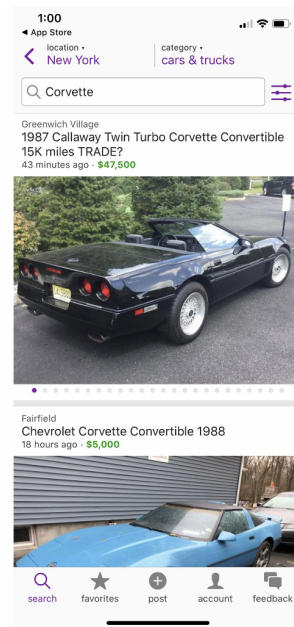
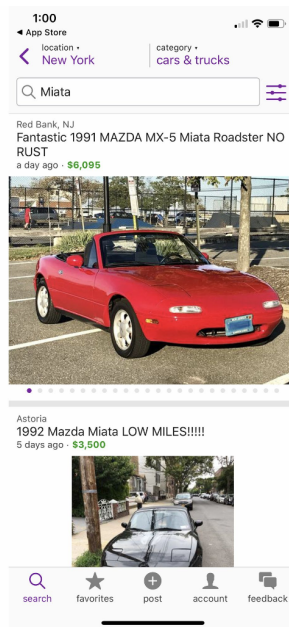
- **‘/deleteProduct’** - This endpoint will be used when a user wants to delete a specific listing. Data passed to this endpoint will be the unique identifier (key) uuid4 value that is associated with the product
- **‘/createProduct’** - This endpoint will be used when a user wants to add a new product to their list. Data passed will be the user's product name, a description, any data associated with the product; and the function will automatically assign a unique ID to the product and update the list.
- **‘/updateProduct’** - This endpoint will be utilized when a user wants to edit an existing product. Data passed here will be the changes they want to make to the product (such as changing the name). As the product is already created, the key ID and the unchanged data will not need to be passed
- **‘/index’** - This homepage will serve as the core of our application. The homepage will include the Login functions and Search functions as well as displaying a random set of listings placed by other users.
- **‘/uuid/listingID’** - Pages like these will be accessed when a user clicks on a specific product listing, which will direct them

to the product page and show images of the product, the price, who posted it, the date it was posted, and the product's description.

- **'/uuid'** - This endpoint will be a specific user's ID and will display all of their current listings. It will function as a 'homepage' for a specific user.
- **'/category'** - This will display specific listings for a certain category, allowing users to browse a category for their needs.

Design

- Inspiration



○



- The above picture is inspiration for our homepage, wherein users can go to listings they would like to view, and the second image contains UI elements we would like to incorporate like the search and login functionality.
- A design like this would allow users to be able to clearly see their products in a neat and concise way. This much detail is out of our scope, however, the general format and display is our goal. We would want to create a user friendly design that allows the user to easily view, add, delete, and edit the products in a way that is quick and efficient.