Project Phase 2

Chris McCorkle, Elian Zamora

Phase 2 -

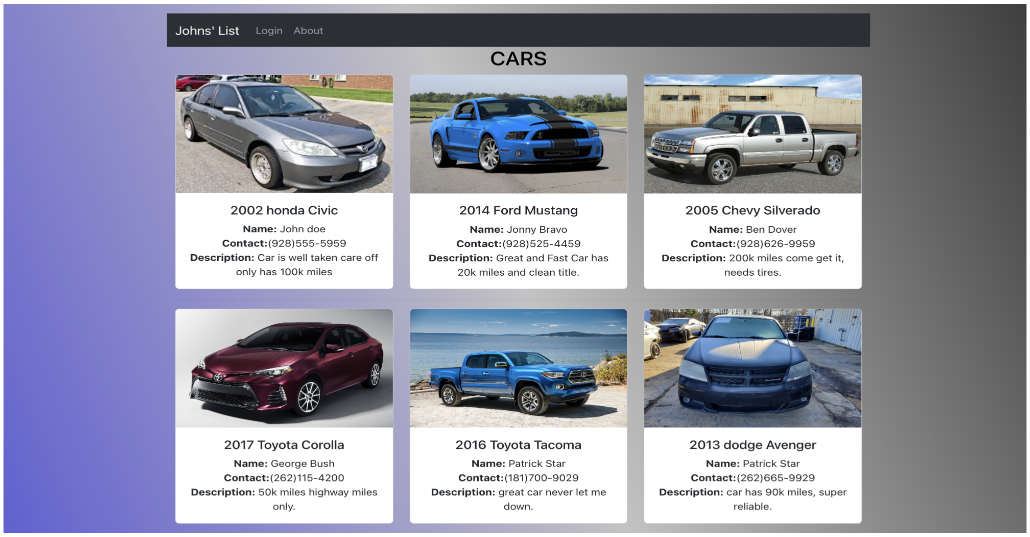
For phase two's deliverable, we decided to implement the back end portion of the application. By doing this, It allows for us to connect all of the different pieces created, to make the application work as a whole and have a better user experience. In this report we'll be going over our implementation of the backend design as outlined in the Phase two section of the deliverables report. Our team overcame these challenges by using effective communication and making sure that our goal was met and understood to the best of our ability, while trying to effectively achieve our backend design in a manner that makes sense to both us as developers and users who will interact with the site.

Tasks Accomplished

1. **Linking Pages:**
    - When it came to linking all of the pages together, we wanted to make sure everything worked as one. We were able to link the login page from the main page for the user to log in. Once logged in the user would be able to go back to the main page and create a listing while on their account.
    - We were able to Link the about page to the main page, that way the user can just click on the about link and be able to see the description of the page and how it works.
    - When the user hits the “Create Listing” button the user is able to create a listing under a specific category and that listing will show up under the selected category

2. Listings:

- When clicking the explore button under each one of the categories, the user will be redirected to a page with dummy listings that match the category that was selected.
- Each Category has about 6 different dummy listings that display a picture of the item for sale. the name of the seller, a contact phone number and a short description of the item. Assuming that these listings were real. If the user was interested in an item, the user would be able to call or text the seller and arrange a way to buy the item.



CARS

2002 honda Civic
Name: John doe
Contact:(928)555-5959
Description: Car is well taken care off only has 100k miles

2014 Ford Mustang
Name: Jonny Bravo
Contact:(928)525-4459
Description: Great and Fast Car has 20k miles and clean title.

2005 Chevy Silverado
Name: Ben Dover
Contact:(928)626-9959
Description: 200k miles come get it, needs tires.

2017 Toyota Corolla
Name: George Bush
Contact:(262)115-4200
Description: 50k miles highway miles only.

2016 Toyota Tacoma
Name: Patrick Star
Contact:(181)700-9029
Description: great car never let me down.

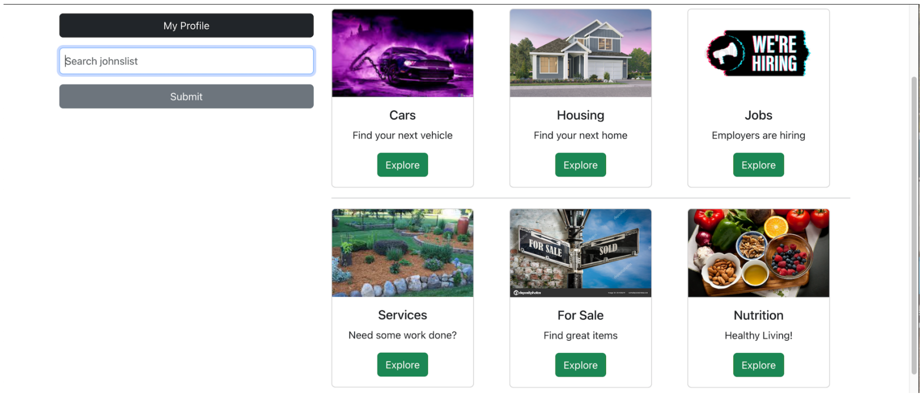
2013 dodge Avenger
Name: Patrick Star
Contact:(262)665-9929
Description: car has 90k miles, super reliable.

3. Functional Components

- **Adding the Listing -** When the user clicks on the create Listing Button the user will have a prompt that requires the user to type in their full name, a contact phone number, the name of the item, and a brief description of the item. The user will have to select a category out of the 6 categories provided so that the listing can be added under the respective category. This means that if the user is selling a vehicle the category that's chosen would be the car category or if the user is providing a listing for a service they offer such as landscaping the user would have to choose the "service" category.
- **Removing the Listing -** When an item is sold the user is able to click on the remove button in order to remove the listing completely. Once the listing is removed there is no way of getting it back. If the user removes their own listing by accident, the only way to get the certain

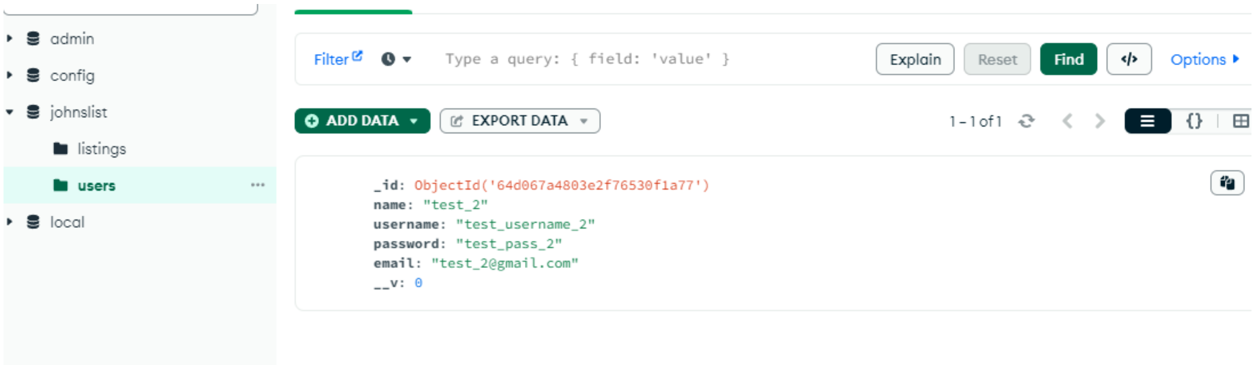
listing to reappear would be to create a new one from scratch by clicking on the “Create listing” button.

- **Displaying the Listing -** When a listing is added under a category it will be displayed in a specific way (something like a baseball Card). The first thing the user will see is the picture of the item. Right under the picture. The name of the item will be displayed as a title. Under that title the user information will be displayed such as the user name and last name, the contact phone number and the description of the item.
- **LogIn/Register Components -** When signing in to the app the user will have to sign in using a username and a password. The account will allow for the user to see their listings as well as add more listings or delete them. If this is the first time the user has entered the app and does not have an account the user is able to create a brand new account that requires the user's full name, a unique username and a unique password. Once the account is created or the user is able to sign in. they will be redirected to the home page and the user can start selling their items.
- **Search Function -** Under the create Listing button, there is a search bar that allows any user to look up an item. This allows for the user to not waste time and find what they are looking for in a fast manner.
- **View User Profile -** On the right hand side of the main page there is a button that says “My profile. When this button is clicked the user is able to see their information. Such as name, user name contact phone number, along with their posted listings.

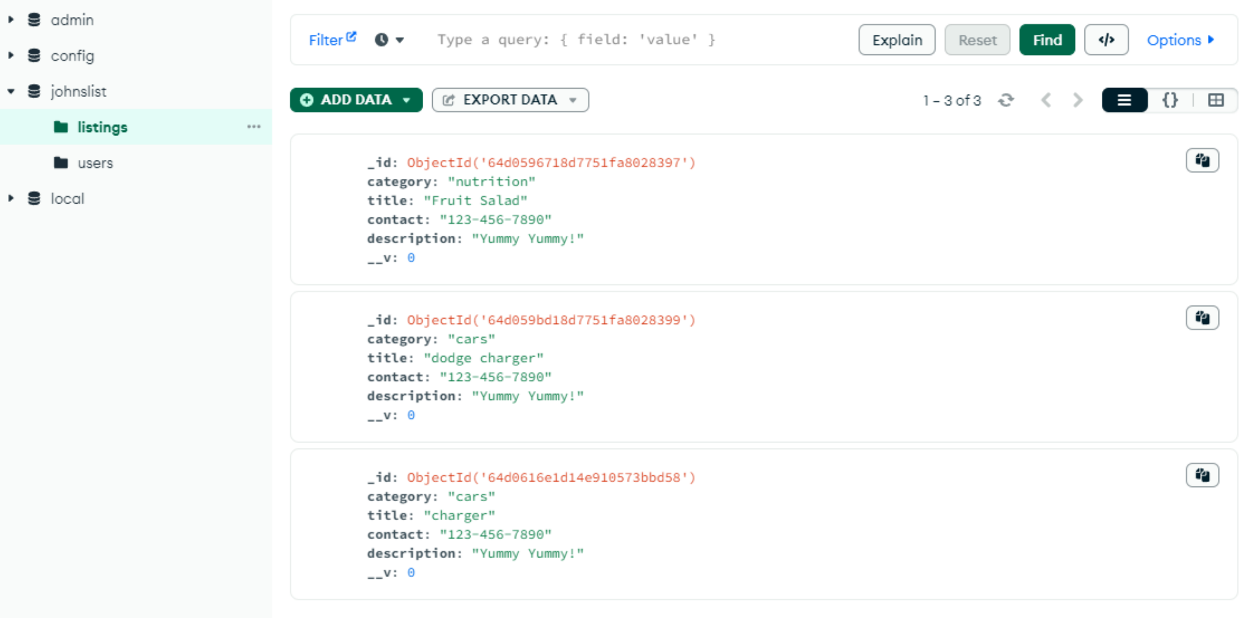


4. **Creating a Database:**

   - **MongoDB -** The mongo database was created using the collections "users" to store the user information, and "listings" to store the data of the listing, which included the category, title, who made it, description and the user's personal contact so they could be reached in order to acquire the product. For the user's collection, only the email and password are stored for their account, but a name and username could be implemented as the DB is set up to receive those as well.



```
_id: ObjectId('64d067a4803e2f76530f1a77')
name: "test_2"
username: "test_username_2"
password: "test_pass_2"
email: "test_2@gmail.com"
__v: 0
```



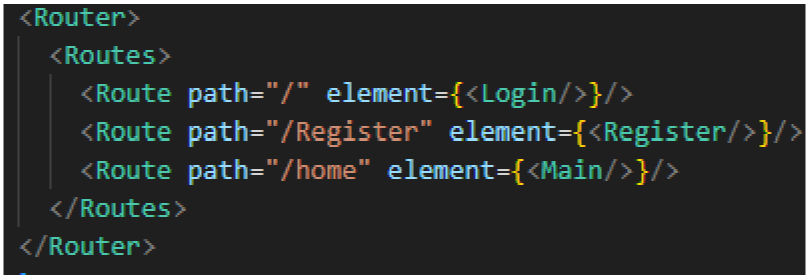
```
_id: ObjectId('64d0596718d7751fa8028397')
category: "nutrition"
title: "Fruit Salad"
contact: "123-456-7890"
description: "Yummy Yummy!"
__v: 0
```

```
_id: ObjectId('64d059bd18d7751fa8028399')
category: "cars"
title: "dodge charger"
contact: "123-456-7890"
description: "Yummy Yummy!"
__v: 0
```

```
_id: ObjectId('64d0616e1d14e910573bbd58')
category: "cars"
title: "charger"
contact: "123-456-7890"
description: "Yummy Yummy!"
__v: 0
```

5. **Linking API endpoints:**

   - **API's -** By themselves, the API endpoints all work as intended, wherein a user will see the category of products that are congruent with the current category they are exploring, however, the endpoints were our biggest difficulty and were not able to be linked together. We plan to implement react routing as it was working in testing for the login section, and setting up the endpoints using a simple link reference following a route that is defined in the App.js section (pictured below)



```
<Router>
  <Routes>
    <Route path="/" element={<Login/>}/>
    <Route path="/Register" element={<Register/>}/>
    <Route path="/home" element={<Main/>}/>
  </Routes>
</Router>
```

Along with routing, this will allow us to fully implement all of the nav buttons that bootstrap has provided, thus allowing the site to be fully traversable with just the nav buttons.

- For the login section specifically, the endpoints are currently connected and working as intended when a user clicks on either the login redirect or the register redirect, the home redirect works as well, but the POST functionality on the frontend is not working as intended, and has been one of our biggest challenges as well.