



Products Industries Research About Login Sign up



Financial Services



Physical Traders



Sourcing and Procurement



Consulting



Products Industries Research About Login Sign up



Gro web app

Jump straight into analysis through intuitive visualizations that can be monitored on live dashboards

[Learn More >](#)



Gro analytics

Gain a competitive edge by using Gro's proprietary machine-learning algorithms and predictive models

[US Corn Yield Forecast >](#)  
[US Soy Yield Forecast >](#)  
[Argentina Soy Yield Forecast >](#)

[Learn More >](#)



Gro API

Gain direct access to Gro's entire database, including supply, demand, weather, and environment data

[Learn More >](#)



Gro dictionary

Explore data from our collection of over 8.828M datasets.

[Learn More >](#)

Hello!

My name is June and I applied for the web developer position at Gro Intelligence.

This was a great exercise for me to practice react and so I actually had a fun time coding this navigation bar.

I put much weight on scalability aspect for this assignment because it seems to me that the items for the products and industries may change in the future when I saw the difference from the gif and the mockup images of the navigation you guys sent me.

## Start

Please download the folder here.

<https://github.com/deepseafishing/simplenavbar>

Then in Terminal, you can type this line to run the server.

```
yarn start
```

Open

<http://localhost:3000>

to view it in the browser.

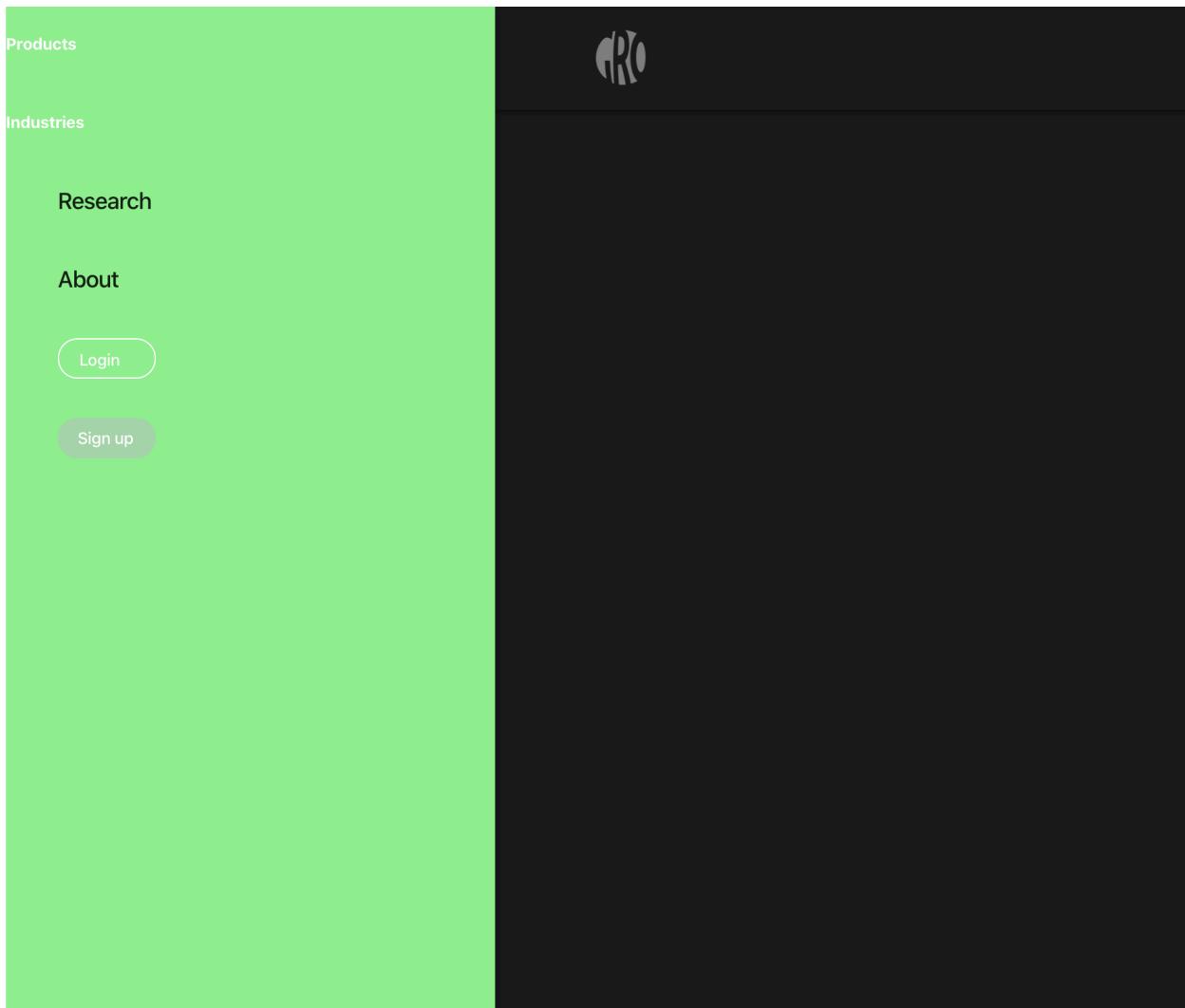
## Tech Stack

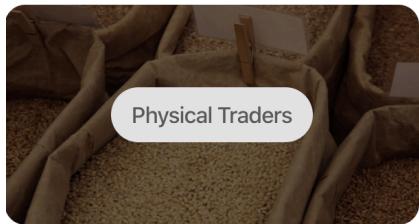
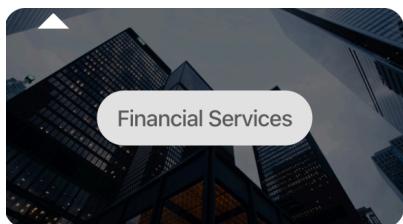
Here is the tech stack I used for this project.

- React
- Create React App (<https://github.com/facebook/create-react-app>)
- React Materialize (<http://react-materialize.github.io/react-materialize/?path=/story/react-materialize--welcome>)
- @material-ui/styles (<https://material-ui.com/styles/basics/>)
- react-use-media-query-hook (<https://www.npmjs.com/package/react-use-media-query-hook>)

I used react because I thought this would be a good way to make this navigation bar as scalable as possible. I wanted to use Material Design because it provides better and more tactile user interfaces. I wanted to have styled react components without separate css files because it supports Sass, it makes the component more modular, and also has many other benefits. (<https://blog.logrocket.com/8-reasons-to-use-styled-components-cf3788f0bb4d>)

For bonus points, I made a side navbar and a hamburger icon when the size becomes less than 1024px. It was not perfect due to of limit of time but I used react-use-media-query-hook to change Col component to Row component so that the items would line up vertically instead of horizontally.



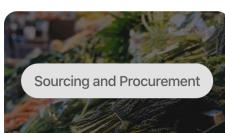
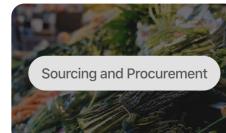
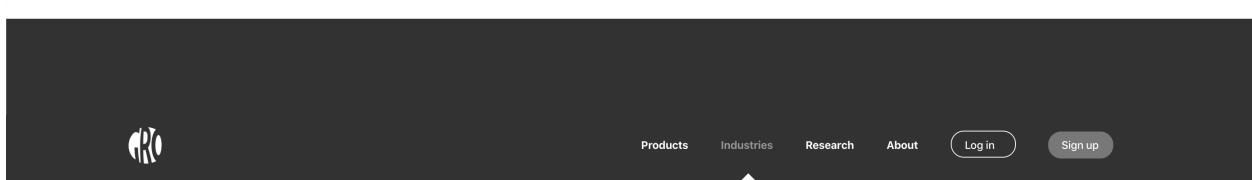
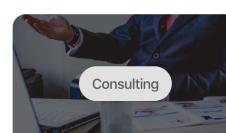
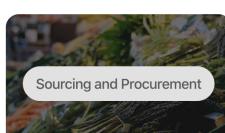
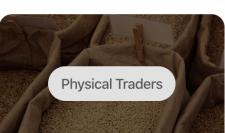
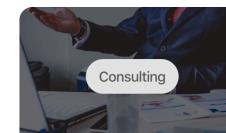
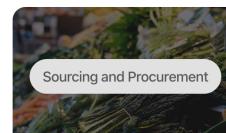
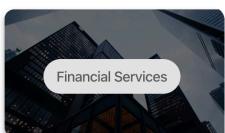
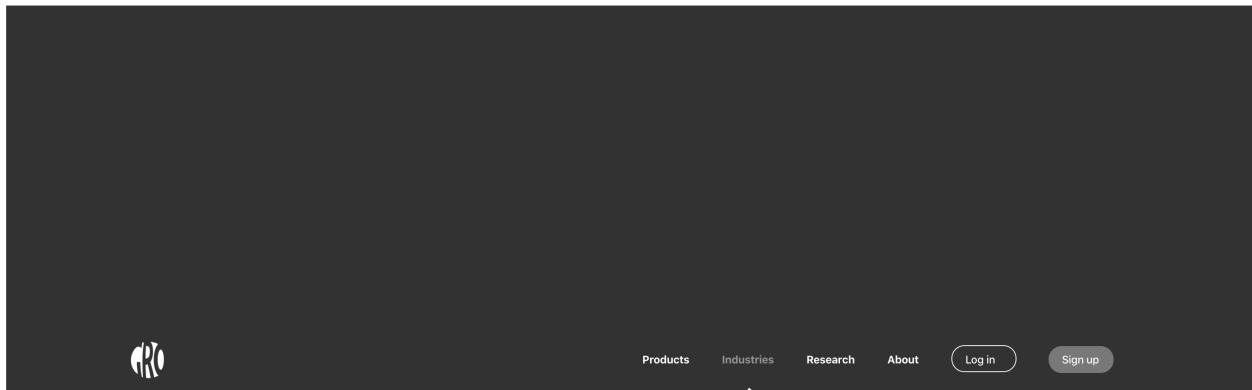
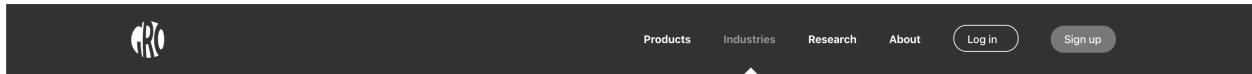


## Architecture

Like I said, my ultimate goal was scalability and good maintenance. When users want to give changes to the product items or industry items or other navigation items, all they have to do is just change values and structure of content.json file. This will automatically generate new navigation items, navigation dropdown menu items, product items and industry items. I will explain it more in detail in the code explanation section.

For React Components, I made it all as functional components because I thought they wouldn't need any states thus should not become stateful components. I first put functions that constructing children react components or html tags inside each related components file such as constructNavBar function, constructItems function and constructLinks function which can be more readable since they are in the same file as the components they are being used. However, I put them all into constructUtils.js file instead in case where other react components may use these functions elsewhere.

These are the pictures when you either remove or add Product Items in the content.json file.



These are the pictures when you either remove or add Industry Items in the content.json file.  
Try adding or removing Product Items or Industry Items or Navigation Menus yourself!

The screenshot shows the top navigation bar with the Gro logo, Products, Industries (highlighted), Research, About, Log in, and Sign up. Below the navigation are three cards:

- Gro web app**: An icon of a dashboard with charts. Description: "Jump straight into analysis through intuitive visualizations that can be monitored on live dashboards". Call-to-action: "Learn More >".
- Gro analytics**: An icon of a line graph. Description: "Gain a competitive edge by using Gro's proprietary machine-learning algorithms and predictive models". Call-to-action: "US Corn Yield Forecast > US Soy Yield Forecast > Argentina Soy Yield Forecast > Learn More >".
- Gro dictionary**: An icon of a book. Description: "Explore data from our collection of over 8.828M datasets.". Call-to-action: "Learn More >".

The screenshot shows the top navigation bar with the Gro logo, Products, Industries (highlighted), Research, About, Log in, and Sign up. Below the navigation are four cards:

- Gro web app**: An icon of a dashboard with charts. Description: "Jump straight into analysis through intuitive visualizations that can be monitored on live dashboards". Call-to-action: "Learn More >".
- Gro analytics**: An icon of a line graph. Description: "Gain a competitive edge by using Gro's proprietary machine-learning algorithms and predictive models". Call-to-action: "US Corn Yield Forecast > US Soy Yield Forecast > Argentina Soy Yield Forecast > Learn More >".
- Gro API**: An icon of a globe with nodes. Description: "Gain direct access to Gro's entire database, including supply, demand, weather, and environment data". Call-to-action: "Learn More >".
- Gro dictionary**: An icon of a book. Description: "Explore data from our collection of over 8.828M datasets.". Call-to-action: "Learn More >".

The screenshot shows the top navigation bar with the Gro logo, Products, Industries (highlighted), Research, About, Log in, and Sign up. Below the navigation are three cards:

- Gro web app**: An icon of a dashboard with charts. Description: "Jump straight into analysis through intuitive visualizations that can be monitored on live dashboards". Call-to-action: "Learn More >".
- Gro analytics**: An icon of a line graph. Description: "Gain a competitive edge by using Gro's proprietary machine-learning algorithms and predictive models". Call-to-action: "US Corn Yield Forecast > US Soy Yield Forecast > Argentina Soy Yield Forecast > Learn More >".
- A fourth card slot is present but empty.

# Code Explanation

## React Components

### - NavBar

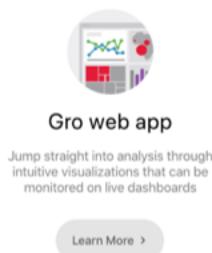
Here, we use constructNavBar function to generate navigation items. We pass styles and data from content.json.

### - NavDropDown

This component is like a container for product items or industry items and using constructItems function from Util functions, we generate these items according to the data given to us.

I used a Row component from react-materialize library with Column components so that it would give us 12 grid system for the items. For each item, I gave space of 3 and 4 items can be inserted in one row. When it exceeds more than 4 items, they would put them in the next row.

### - ProductItem

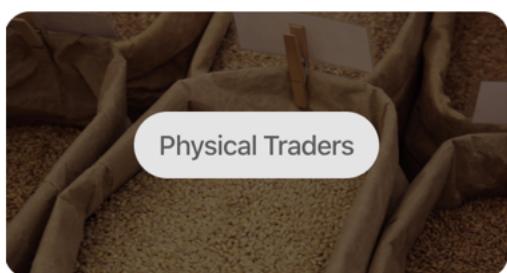


Each product item component, represents the box of product item like the one on the left. It has chip component and an image div that gets the path of the image from the code from content.json file. In order to do that, I added each image files' names to the content.json file like this.

When there is other links, the link array, we use constructLinks function to generate clickable span tags inside li tag.

```
"subnavList": [
  {
    "label": "Financial Services",
    "img": "financial-services.jpg"
  },
]
```

### - IndustryItem



Each industry item component, represents the box of industry item like the one on the left. It has chip component and an image div that gets the path of the image from the code from content.json file.

## Util functions

### - constructNavBar

We classify nav items into drop menus, login, signin and just normal nav items. If we see there is a subnavList, we make that into dropdown menu. If the label matches with login or signup, we create a react component accordingly. For the rest, we just make them into normal navigation menu like ‘about’.

### - constructItems

Depending on the label of data, it generates either product items or industry items.

To always align items to the left, we generate empty Col divs when there is a shortage from 4 items. So if there is only one item in products dropdown menu, it will attach empty col divs to the end of that row.

### - constructLinks

Here, we generate ul tag with li tags using the link array data.