

## Scenario

Gousto's technical infrastructure includes an API Gateway. The gateway offers a number of product operations. There is a call to get products on current menu and another call to list all available categories.

Your task is to design a page to showcase the existing products and categories.

## Functional Requirements

- Shows all the available category names in one row (can wrap into multiple lines if screen is not wide enough)
- Shows a list of product titles under each other
- Once a category is clicked only products belonging to the clicked category will be shown and clicked category will be bold to indicate it is active
- When the user navigates with the back and forward button, the previously selected category should be shown.
- An input field should be placed above the list of products and below the categories.
- The input field should be used as a free text search filter for products. Once input is entered only products which feature the given text in their title or description should be shown . (eg: input is "win", products with title "Red Wine" should be shown)
- When a product name is clicked it becomes bold and toggles the visibility of the description, ie. when clicked it shows the product description below the product name. When clicked again the description should not be shown. Multiple descriptions can be shown in the same time

### Api endpoints:

To get categories: <https://api.gousto.co.uk/products/v2.0/categories>

To get

products: [https://api.gousto.co.uk/products/v2.0/products?includes\[\]=categories&includes\[\]=attributes&sort=position&image\\_sizes\[\]=365&image\\_sizes\[\]=400&period\\_id=120](https://api.gousto.co.uk/products/v2.0/products?includes[]=categories&includes[]=attributes&sort=position&image_sizes[]=365&image_sizes[]=400&period_id=120)

For cross origin accessibility please disable cross origin restrictions in your browser. Safari:

Develop/Disable Cross Origin Restrictions, Chrome: install this

plugin <https://chrome.google.com/webstore/detail/allow-control-allow-origi/nlfbmbojpeacfhkpbjhddihlkkiljbi>

## Non-Functional Requirements

- UI should be built with React, but you are allowed to use additional other libraries if needed
- Use your preferred library for state management
- Don't include too much CSS but use your preferred CSS solution / library, you will be judged on the structure and set up of styling. It should only have basic styling to facilitate positioning and some colours.
- The code should be production ready (ie include tests)
- It should work in every major browsers (Firefox, Safari, Chrome, IE 11+, Mobile Safari & Chrome)

## Please also ensure the following

- Finished task should be submitted via email as a ZIP or GZIP attachment or alternatively via a public GitHub Repository
- Submission should include a README file explaining
  - How to use your solution
  - Explain how your solution uses the components you built
  - Anything else you think is relevant to your solution

## Visuals

Find the visuals below which indicates how the solution should look and behave.

Store Cupboard Wine Gin Desserts Homeware

Cabernet  
Merlot  
Shiraz  
Sauvignon Blanc  
Chenin Blanc  
Chardonnay  
Gavi  
Champagne

Store Cupboard Wine Gin Desserts Homeware

Cabernet  
Merlot  
**Shiraz**  
*Qui facit eorum claritatem Investigationes  
demonstraverunt lectores legere me.*  
Sauvignon Blanc  
Chenin Blanc  
Chardonnay  
Gavi  
Champagne

Store Cupboard Wine Gin Desserts Homeware

Sauvignon Blanc  
Chenin Blanc