

Frontend Challenge

Introduction

The purpose of this challenge is to exhibit familiarity with frontend technologies, programming patterns and to provide a sample of what clean and reusable code means to you. Please put your comments about the decisions you have made inline and give a summary about the architectural choices that you considered.

We ask you to keep the following in mind:

- Pay attention to the structure of the implementation and defend if the dependencies are truly necessary
- During the review we'll focus on: proper testing, code splitting, documentation and the usefulness of artifacts

Assignment

Build a Compare Page as attached design, with the following features:

- Show and hide products
- Highlight differences between product features
- Features ordered alphabetically by feature name
- Badges should be placed before features

According to the API, the following keys are not product features to compare:

salePrice, manufacturerName, grossPrice, BUP_UOM, BUP_Value, uom, productImage, BUP_Conversion, minQuantity, manufacturerImage, name, sku, listPrice, channel, display, atp

API

The following URL provides 4 products:

<http://5c35e7f96fc11c0014d32fcd.mockapi.io/compare/products>

Architecture

There is no limitation. Design your application any way you want. Focus on clean, reusable code and frontend best practices. Show us how to produce high quality modern web applications.

Libraries

You can use any libraries you think are best but we would love to see your skills with **React**.

Backend

No Backend required.

Extra points

- Webpack
- Sass/Less or Styled components
- Responsiveness
- Browser backward compatibility (IE11)
- Delivery the exercise with a readme file that explains what you have done and how to run your project.

Design

4 producten vergelijken