

JavaScript Engineer Test Tasks

This post is open to engineers that can complete a number of the tasks in this test.

- Download the [source files](#)
- Follow the "Getting Started" part of the README to set up the project
- Ignore the "What we expect from you" part
- Please firstly complete task 1, then pick whatever tests you feel comfortable working on. Let us know which tests you tackled along with your source code.
- You must have written some tests
- We recommend you spend 30-35 minutes on a task
- *hint:* We want to see a good balance of JavaScript & CSS

1. Styling the menu

The styling can be either be done using SASS or CSS, if using CSS you will need to add a CSS file to index.html

Apply the style as per [frontend-skills-test--1.png](#)

You will need copy the meal-card__curve.svg from the design/assets folder to app/images folder.

You can ignore:

the tag indicators (circles under the meal name)

the collapse and expand of long description text

the number of options available (you can hard code that to 3).

2. Add the tag indicators

Each meal in the data/menu.json file contains an array of tags, these tags have indicators:-

charcoal

cheese

chicken

grilled

high-protein

lamb

pasta

peanut

pork

seafood

snack

spicy

starter
sweet
vegetarian

The .svg files for these tags are in the design/assets folder you should copy them to the app/images folder

3. Add a basket service

Display the count and the total in the header if there is anything in the basket

All meals can be considered as 'main'

'Add To Your Order' button adds the meal to the basket

The total and the count are updated every time you press the 'Add To Your Order' button.

4. Expand the shopping basket

When you click on the total price in the main screen header, you should show the basket as expanded. This will list all of the items in the basket.

5. Add '+' and '-' buttons in the expanded basket

This will increment and decrement the quantity and total value.

You get extra points if clicking "-" on a meal with a quantity of "1" will remove it from the basket

6. A click on the 'Confirm Order' will clear the basket and close it.

7. Styling the basket

The styling should be as per [frontend-skills-test--2.png](#)

8. Add animation (with CSS)

For opening and closing of the basket, the animation is shown across

[frontend-skills-test--3a.png](#)

[frontend-skills-test--3b.png](#)

and

[frontend-skills-test--3c.png](#)

9. Detect Main Meals

Consider meals that have a tag of "#course:main_courses" as a main meal and meals that do not have this tag as "other"

Update the collapsed basket display [frontend-skills-test--2.png](#) and the expanded basket display [frontend-skills-test--3c.png](#) to show quantity and totals for 'main' and 'other'

10. Add local storage of the basket

After a full refresh of the page the contents of the basket will remain.