**Documentation:**

Design model : **Desktop first.** The reason is that I feel more comfortable starting from a wider viewport and using media queries afterwards.

For the responsiveness, I decided to use a combination of flexbox and media queries . Fonts, margins and paddings as well as image sizes were positioned by using relative values(%;em;rem) instead of static ones(px).

JS feature: for now, it just says “hello world” but will be improved in future versions.

**Dependencies:**

**PROD:**

BABEL: will enable me to use ECMAScript 6 for future developments.

POST CSS : to parse CSS and used Plugins like auto-prefixer.

**DEV**

Font-awesome: for icons

Google fonts

Parcel

**STYLES:**

I like to use <https://meyerweb.com/eric/tools/css/reset/> to reset styles for most of my projects.

/\* RESET \*/

html{

  margin: 0;

  padding: 0;

  border: 0;

  font-size: 100%;

  font: inherit;

  vertical-align: baseline;

  }

Font is Roboto and has been installed via NPM. No margin and font-size is not too small.

body{

  margin:0;

  font-family: 'Roboto', sans-serif;

  font-size: 1.8rem;

  line-height: 1;

}

**INDEX:**

I went for a simple navigation bar with a logo. The logo is positioned relatively to the left of the body and the subtitle has been positioned absolutely to the top.

nav{

   color:white;

  height:10vh;

  font-size:2rem;

  background: #0f70e6;

  padding:1em 0.5em;

  margin-bottom:1em;

  position: relative;

      }

#nav-title{

  position: relative;

  left:0.5em;

  top:0.2em;

}

#subtitle{

color:rgb(213, 226, 238);

font-size: 1rem;

position: absolute;

left:8.5em;

top:3em;

}

The content is a simple list of links to the main categories.

**CATEGORY1(Front End) + CATEGORY2(Module Bundlers)**

Main content has been positioned in a single row with the following properties:

.row{

  display: flex;

  justify-content: space-around;

  align-items: center;

  flex-wrap: wrap;

}

Flexbox allows for easy, responsive positioning and smart wrapping when shrinking the viewport. I was happy with the result and decided not to make any further change.

Logos have the following properties

.mb-imgs > p >a > img{

  height:120px;

  width:120px;

  padding:1em;

  opacity:0.3;

 }

.fe-imgs > p >a > img{

  height:120px;

  width:120px;

  padding:1em;

  opacity:0.3;

 }

*NB: For this project I used regular CSS but with SaaS those two lines could have been reduced to one. I might use SaaS in the next iteration.*

Captions are in italic, smaller and positioned relatively to the left by 50px

 figcaption{

  font-style: italic;

  font-size: smaller;

  position: relative;

  left:50px;

 }

Opacity has been reduced for non-active links.

 #python-logo{

  opacity:1;

 }

 #parcel-logo{

  opacity:1;

 }

**Details1(Parcel)+ Details 2(Python)**

This section includes an image of the selected category followed by 3 descriptive paragraphs.

I preferred using paragraphs <p> to using lists of definitions <td> as the content is not truly a definition per se.

The main content has a width of 90% of the viewport and a high enough height so that the text can collapse to the bottom without covering the footer.

 .main-details {

  width: 90vw;

  height:230vh;

  display: flex;

  flex-direction: column;

  margin: 0 auto;

}

Line height has been increased so that the text is pleasurable to read.

.details-text {

  border: 10px solid #95bdee  ;

  line-height: 2em;

  padding:0.8em;

}

Images´container (« details-img ») has been converted in a flex container and their content is centred to the middle. Images are 70% and 50% of the size of their parent container.

.details-img{

  display: flex;

  justify-content: center;

     }

#parcel-img{

  height:auto;

  width:70%;

   }

 #python-img{

  margin-bottom:3em;

  height:auto;

  width:50%;

 }

**MEDIA QUERIES**

Media queries for row margins(categories) + footer and to adjust font-sizes according to the viewport width.

@media only screen and (max-width: 600px) {

  .row{

    margin-top: -80px;

  }

  footer{

    margin-top: 600px;

  }

}

@media only screen and (max-width: 768px) {

  .title{

    font-size:2rem;

  }

  body{

    font-size: 1.1rem;

  }

}