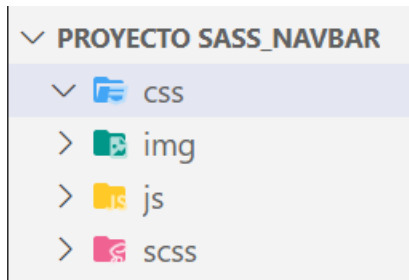


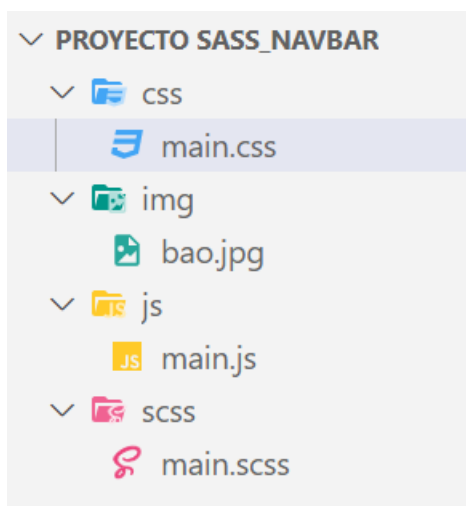
Guía de Aprendizaje: NavBar Responsive con HTML + CSS + SAAS + JS

El objetivo del ejercicio es diseñar un DashBoard que incluye un menú desplegable

1. Creamos un proyecto con la siguiente estructura



2. Creamos los archivos en sus respectivas carpetas: ***css/main.css***, ***js/main.js***, ***scss/main.scss***; incluir la foto del perfil en la carpeta img (***img/bao.jpg***).



3. El archivo index.html debe ser creado con la plantilla html:5; y lo modificamos para contener los objetos que se definen a continuación:

```

index.html > ...
1  <!DOCTYPE html>
2  <html lang="es">
3    <head>
4      <meta charset="UTF-8" />
5      <meta
6        name="viewport"
7        content="width=device-width, initial-scale=1.0"
8      />
9      <title>NavBar</title>
10     <!-- CSS -->
11     <link rel="stylesheet" href="css/main.css" />
12   </head>
13   <body>
14     <!-- NAV -->
15     <nav class="nav" id="nav"></nav>
16     <!-- MAIN -->
17     <main class="main" id="main"></main>
18     <!-- JS -->
19     <script src="js/main.js"></script>
20   </body>
21 </html>

```

4. Incluimos la librería de Google Fonts – Icons cuyo Link CDN esta en la página:
https://developers.google.com/fonts/docs/material_icons#icon_font_for_the_web

```

3  <head>
4    <meta charset="UTF-8" />
5    <meta
6      name="viewport"
7      content="width=device-width, initial-scale=1.0"
8    />
9    <title>NavBar</title>
10   <!-- CSS MAIN -->
11   <link rel="stylesheet" href="css/main.css" />
12   <!-- Libreria Google Fonts - Icons -->
13   <link
14     rel="stylesheet"
15     href="<link href="https://fonts.googleapis.com/icon?family=Material+Icons"
16     rel="stylesheet">"
17   />
18 </head>

```

5. Procedemos a crear un **div** dentro de la etiqueta **nav**, que llamaremos **nav-menu**;
 Para hacerlo digitamos el comando seguido de un <enter> **div.nav__menu.container#nav-menu <enter>**

```

24   <nav class="nav" id="nav">
25     <div class="nav__menu container" id="nav-menu">
26
27     </div>
28   </nav>

```

6. Dentro del **div** creado en el punto anterior, creamos tres (3) **div** adicionales ejecutando los comandos **div.nav__shape<enter>**, **div.nav__close#nav-close<enter>** y **div.nav__data<enter>**

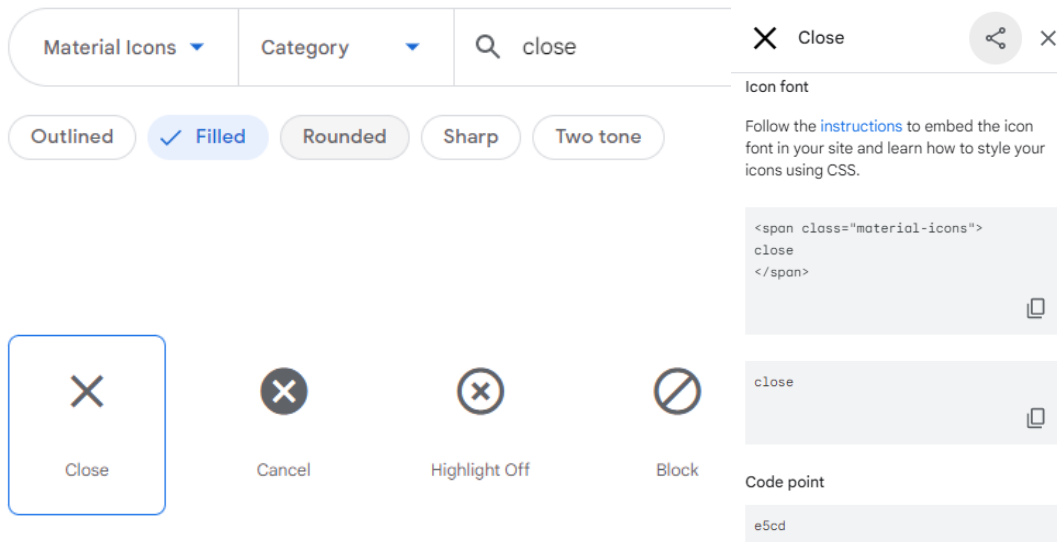
```

24   <nav class="nav" id="nav">
25     <div class="nav__menu container" id="nav-menu">
26       <div class="nav__shape"></div>
27
28       <div class="nav__close" id="nav-close"></div>
29
30       <div class="nav__data"></div>
31     </div>
32   </nav>

```

7. Insertamos el icono **Close** dentro del **nav** llamado **nav-close**

- a. Buscamos en Google Icons, <https://fonts.google.com/icons?icon.query=close&icon.style=Filled&icon.set=Material+Icons&icon.platform=web>, el icono CLOSE



- b. Copiamos el código de la etiqueta **span**

```
<nav class="nav" id="nav">
  <div class="nav__menu container" id="nav-menu">
    <div class="nav__shape"></div>

    <div class="nav__close" id="nav-close">
      <span class="material-icons"> close </span>
    </div>

    <div class="nav__data"></div>
  </div>
</nav>
```

8. En el **div** de la clase **nav__data**, creamos tres objetos:

```
<div class="nav__data">
  <div class="nav__mask">
    
  </div>
  <span class="nav__saludo">Hola, Yo soy</span>
  <h1 class="nav__name">Maui <br> Hawaii</h1>
</div>
```



Hola, Yo soy

Maui
Hawaii

9. Con la etiqueta **ul** creamos una lista del menú

a. Ejecutamos el comando **ul.nav__list**

```
<ul class="nav__list">
```

```
</ul>
```

b. Dentro de la etiqueta **ul**, ejecutamos el comando **li.nav__item**

```
<ul class="nav__list">
```

```
<li class="nav__item">
```

```
</li>
```

```
</ul>
```

c. Dentro de la etiqueta **li**, ejecutamos el comando **a.nav__link**

```
<ul class="nav__list">
```

```
<li class="nav__item">
```

```
<a href="" class="nav__link">
```

```
</a>
```

```
</li>
```

```
</ul>
```

d. Copiamos este código, cinco (5) veces, para personalizar las opciones del menú

```
<ul class="nav__list">
  <li class="nav__item">
    <a href="#inicio" class="nav__link">
      <span class="material-icons">home</span>
      Inicio
    </a>
  </li>

  <li class="nav__item">
    <a href="#acerca" class="nav__link">
      <span class="material-icons">
        >account_circle</span>
        Acerca
      </a>
    </li>

  <li class="nav__item">
    <a href="#portafolio" class="nav__link">
      <span class="material-icons">work</span>
      Hoja de Vida
    </a>
  </li>

  <li class="nav__item">
    <a href="#habilidades" class="nav__link">
      <span class="material-icons">bookmarks</span>
      Habilidades
    </a>
  </li>

  <li class="nav__item">
    <a href="#contacto" class="nav__link">
      <span class="material-icons">chat </span>
      Contacto
    </a>
  </li>
</ul>
```

X



Hola, Yo soy

Maui
Hawaii

-  [Inicio](#)
-  [Acerca](#)
-  [Hoja de Vida](#)
-  [Habilidades](#)
-  [Contacto](#)

10. Dentro de la etiqueta **main**, creamos seis (6) objetos:

```
Header.header#header
section.section.section__height.container#inicio
section.section.section__height.container#acerca
section.section.section__height.container#portafolio
section.section.section__height.container#habilidades
section.section.section__height.container#contacto
```

```
<main class="main" id="main">
  <!-- Encabezado -->
  <header class="header" id="header">...
  </header>

  <!-- Inicio -->
  <section class="section section__height container" id="inicio">
    <h1>Inicio</h1>
  </section>

  <!-- Acerca -->
  <section class="section section__height container" id="acerca">
    <h1>Acerca</h1>
  </section>

  <!-- Portafolio -->
  <section
    class="section section__height container"
    id="portafolio"
  >
    <h1>Portafolio</h1>
  </section>

  <!-- Habilidades -->
  <section
    class="section section__height container"
    id="habilidades"
  >
    <h1>Habilidades</h1>
  </section>

  <!-- Contacto -->
  <section class="section section__height container" id="contacto">
    <h1>Contacto</h1>
  </section>
</main>
```

11. Nos ubicamos en la etiqueta **header**

a. Ejecutamos el comando **nav.header__nav.container**

```
<!-- Encabezado -->
<header class="header" id="header">
  <nav class="header__nav container">

  </nav>
</header>
```

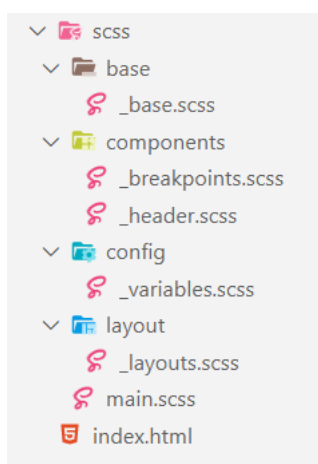
b. Creamos dos (2) objetos, dentro del **nav "header"**, ejecutando los comandos

a.header__Logo y **div.header__toggle#header-toggle**

```
<!-- Encabezado -->
<header class="header" id="header">
  <nav class="header__nav container">
    <a href="#" class="header__logo"> Hawaii </a>

    <div class="header__toggle" id="header-toggle">
      <span class="material-icons">view_headline</span>
    </div>
  </nav>
</header>
```

12. Para el diseño, utilizaremos el precompilador Sass; por lo anterior, debemos crear la estructura con archivos que están vacíos inicialmente.

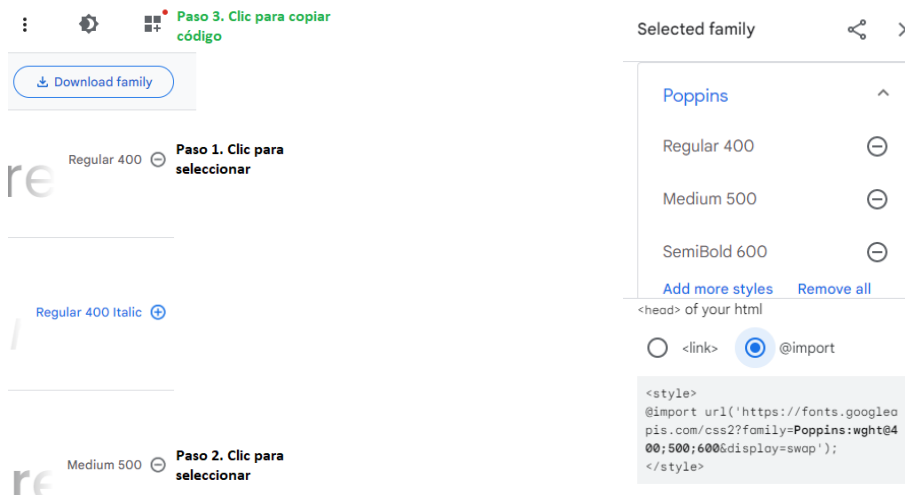


13. En el archivo **main.scss** importamos los otros archivos **scss** que definimos:

```
scss > main.scss
1  @import 'config/variables';
2  @import 'base/base';
3  @import 'layout/layouts';
4  @import 'components/header';
5  @import 'components/breakpoints';
-
```

14. Definimos las variables que vamos a utilizar en el archivo **_variables.scss**

- a. Para esta aplicación vamos a utilizar Google Fonts y nos interesa el tipo Poppins en los tamaños 400, 500 y 600



- b. El archivo **_variables.scss** queda.

```
scss > config > _variables.scss > ...
1  /* == Google Fonts == */
2  @import url('https://fonts.googleapis.com/css2?family=Poppins:wght@400;500;600&display=swap');
3
4  /* == Variables CSS == */
5  :root {
6      --header-height: 3.5rem;
7      /*== Colores: modo HSL ==*/
8      --first-color: hsl(273, 97%, 66%);
9      --first-color-shape: hsl(253, 33%, 32%);
10     --title-color: hsl(273, 8%, 15%);
11     --text-color: hsl(273, 4%, 45%);
12     --body-color: hsl(273, 100%, 99%);
13     --container-color: hsl(0, 0%, 100%);
14     --title-color-dark: hsl(273, 83%, 98%);
15     --text-color-dark: hsl(273, 21%, 56%);
16     --container-color-dark: hsl(273, 80%, 14%);
17
18     /*== Tipografía y Tipos de letra ==*/
19     --body-font: 'Poppins', sans-serif;
20
21     --big-font-size: 1.5rem;
22     --normal-font-size: .938rem;
23     --small-font-size: .813rem;
24
25     /*== Tipografía Responsive ==*/
26     @media screen and (min-width:968px){
27         --big-font-size: 2rem;
28         --normal-font-size: 1rem;
29         --small-font-size: .875rem;
30     }
31
32     /*== Z Index ==*/
33     --z-fixed: 100;
34 }
```

Nota: Una vez que esté funcionando podrá personalizar estos valores.

15. El diseño básico de nuestro **dashboard** estará definido en el archivo **_base.scss**, donde se establecen las propiedades **css** de cada objeto.

```
1  /*== Código diseño básico ==*/
2
3  *{
4      box-sizing: border-box;
5      padding: 0;
6      margin: 0;
7  }
8
9  html{
10     scroll-behavior: smooth;
11     overflow-x: hidden;
12 }
13
14 body {
15     font-family: var(--body-font);
16     font-size: var(--normal-font-size);
17     color: var(--text-color);
18     overflow-x: hidden;
19 }
20
21
22 h1, h2, h3 {
23     color: var(--title-color);
24 }
25
26 ul {
27     list-style: none;
28 }
29
30 a{
31     text-decoration: none;
32 }
33
34 img{
35     max-width: 100%;
36     height: auto;
37 }
```



16. En el archivo **_Layouts.scss** establecemos las clases **css** que reutilizamos.

```
1  /*== Clases CSS reutilizables ==*/
2
3  .container {
4      max-width: 1024px;
5      margin-left: 1.5rem;
6      margin-right: 1.5rem;
7  }
8
9  .section {
10     padding: 4.5rem 0 1rem ;
11
12     //Importante: la clase section_height debe ser eliminada o modificada de acuerdo
13     //al contenido de las secciones
14     &__height{
15         height: 100vh; //define que la seccion ocupa el 100% de la altura de la pantalla
16     }
17 }
18
19 ..
```

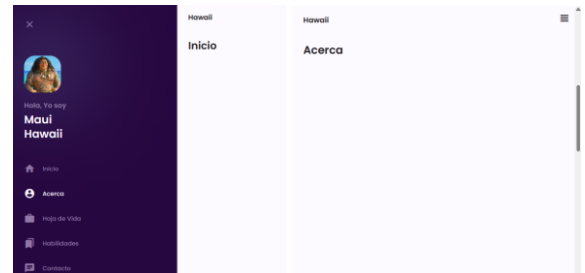
17. El diseño que daremos a las etiquetas **Header**, **Nav**, **Main** se realiza a través del archivo **_header.scss**

20. Para darle funcionalidad a cada opción del menú desplegado, diseñamos código **JS** en el archivo **main.js**

```

22
23 /*===== FUNCIONALIDAD MENU DESPLEGADO (OPCIONES) =====*/
24 const navLink = document.querySelectorAll('.nav__link')
25
26 const linkAction = () =>{
27   const main = document.getElementById('main')
28   // Cuando demos click en cada opción del menu, quitaremos la clase 'show-menu'
29   main.classList.remove('show-menu')
30 }
31 navLink.forEach(n => n.addEventListener('click', linkAction))

```



21. Cambiamos el encabezado del **header** para que despliegue una sombra en caso de utilizar el scroll; creamos la funcionalidad en el archivo **main.js** y definimos el diseño de la clase **'scroll-header'** en el archivo **_header.scss**

main.js

```

33 /*===== CAMBIO HEADER AL DAR SCROLL =====*/
34 const scrollHeader = () =>{
35   const header = document.getElementById('header')
36   // Cuando el desplazamiento es mayor que 50 de altura de ventana gráfica,
37   // agregue la clase 'scroll-header' a la etiqueta header
38   this.scrollY >= 50 ? header.classList.add('scroll-header') : header.classList.remove('scroll-header')
39 }
40
41 window.addEventListener('scroll', scrollHeader)

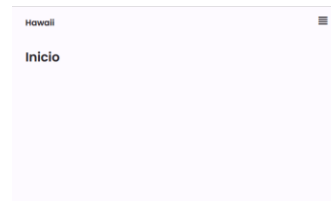
```

_header.scss

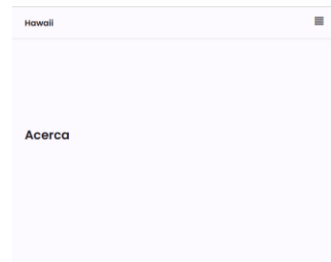
```

137 .scroll-header {
138   box-shadow: 0 1px 4px hsla(0, 4%, 15%, .1);
139 }

```



Al hacer scroll



22. Relacionamos la opción del dashboard que se esta visualizando con la opción del menu desplegable; esta funcionalidad se crea en el archivo **main.js**, se agrega la clase **'active-link'** al archivo **_header.scss** y a cada etiqueta **a** del archivo **index.html**.

main.js

```

43 /*===== SCROLL SECCIONES - OPCION ACTIVA MENU DESPLEGABLE=====*/
44 const sections = document.querySelectorAll('section[id]')
45
46 const scrollActive = () =>{
47   const scrollY = window.pageYOffset
48
49   sections.forEach(current =>{
50     const sectionHeight = current.offsetHeight,
51     sectionTop = current.offsetTop - 58,
52     sectionId = current.getAttribute('id'),
53     sectionsClass = document.querySelector('.nav__menu a[href="#" + sectionId + "']')
54
55     if(scrollY > sectionTop && scrollY <= sectionTop + sectionHeight){
56       sectionsClass.classList.add('active-link')
57     }else{
58       sectionsClass.classList.remove('active-link')
59     }
60   })
61 }
62 window.addEventListener('scroll', scrollActive)

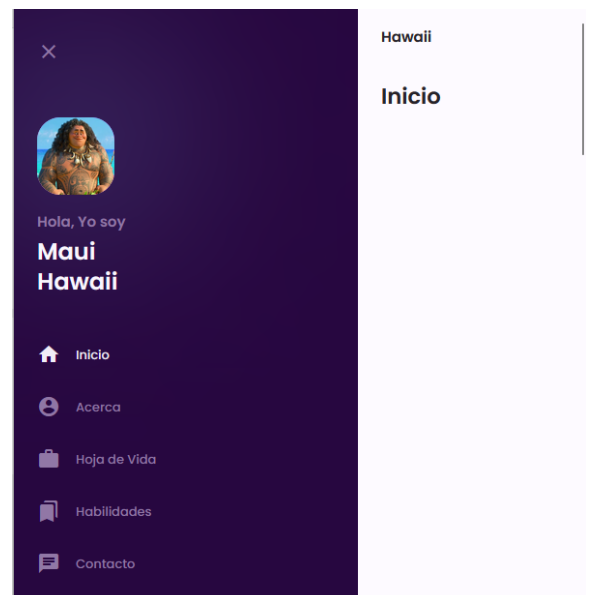
```

_header.scss

```

141 .active-link {
142   color: var(--title-color-dark);
143 }

```



index.html

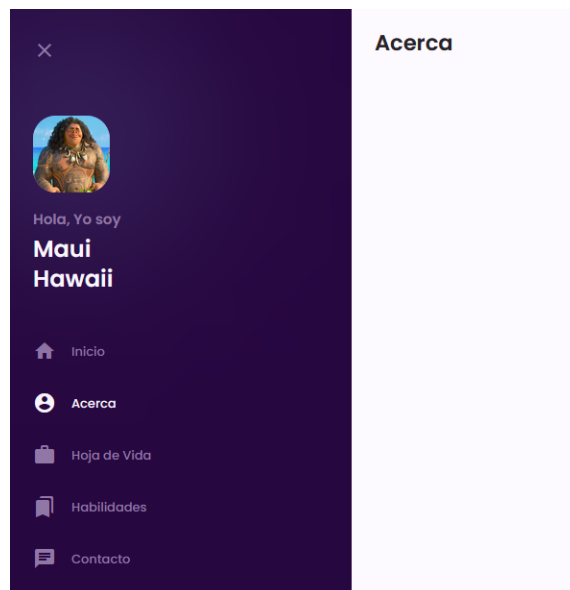
```
<ul class="nav_list">
  <li class="nav_item">
    <a href="#inicio" class="nav_link active-link">
      <span class="material-icons">home</span>
      Inicio
    </a>
  </li>

  <li class="nav_item">
    <a href="#acerca" class="nav_link active-link">
      <span class="material-icons">account_circle</span>
      >
      Acerca
    </a>
  </li>

  <li class="nav_item">
    <a href="#portafolio" class="nav_link active-link">
      <span class="material-icons">work</span>
      Hoja de Vida
    </a>
  </li>

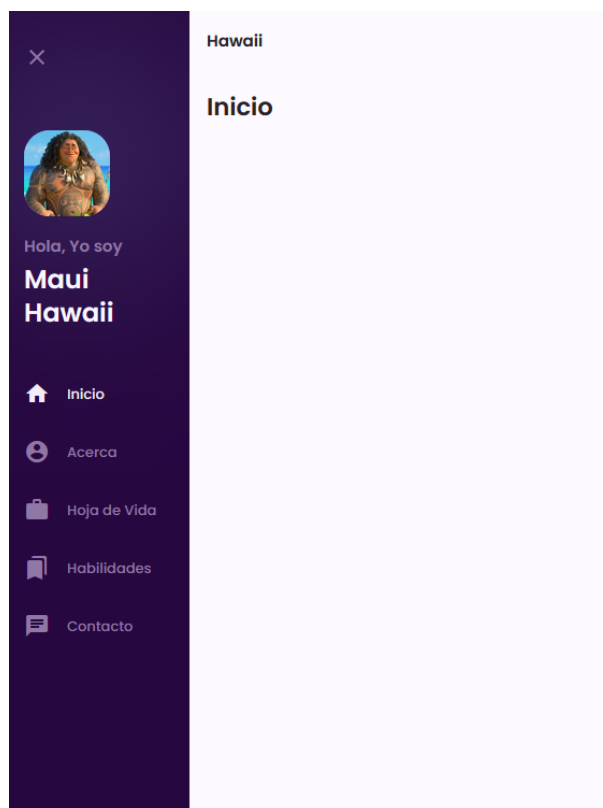
  <li class="nav_item">
    <a href="#habilidades" class="nav_link active-link">
      <span class="material-icons">bookmarks</span>
      Habilidades
    </a>
  </li>

  <li class="nav_item">
    <a href="#contacto" class="nav_link active-link">
      <span class="material-icons">chat</span>
      Contacto
    </a>
  </li>
</ul>
```



23. Con la funcionalidad completa, centremos la atención en el diseño responsive; para ello trabajamos en el archivo **breakpoints.scss**

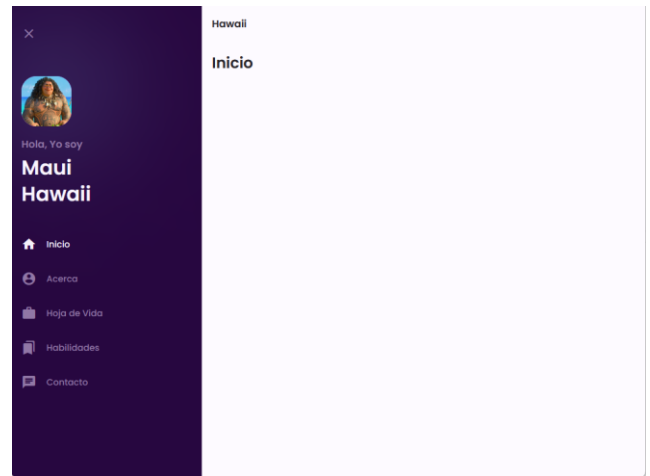
```
1  /*== Dispositivos Mobiles ==*/
2  @media screen and (max-width: 640px){
3    .container {
4      margin-left: 1rem;
5      margin-right: 1rem;
6    }
7
8    .show-menu {
9      transform: translate(40%);
10   }
11 }
```



```

12
13 /*== Tablet ==*/
14 @media screen and (max-width:1023px){
15     .container {
16         margin-left: 1rem;
17         margin-right: 1rem;
18     }
19
20     .show-menu {
21         transform: translate(30%);
22     }
23 }

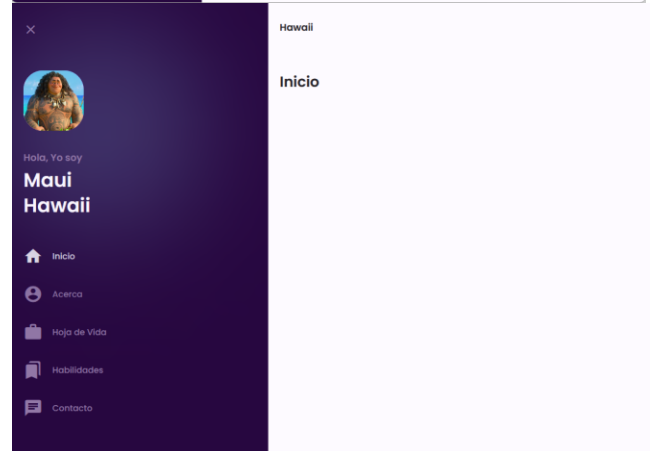
```



```

26 /*== Equipos de escritorio ==*/
27 @media screen and (min-width:1024px){
28     .container {
29         margin-left: auto;
30         margin-right: auto;
31     }
32
33     .section {
34         padding: 7rem 0 2rem;
35     }
36
37     .header__nav {
38         height: calc(var(--header-height) + 1.5rem);
39     }
40
41     .nav {
42         &__shape {
43             width: 350px;
44             height: 350px;
45         }
46
47         &__mask {
48             width: 100px;
49             height: 100px;
50             border-radius: 2rem;
51             margin-bottom: 2rem;
52         }
53
54         &__img {
55             width: 100px;
56         }
57
58         &__link {
59             font-size: --big-font-size;
60
61             & span {
62                 font-size: 2rem;
63             }
64         }
65     }
66
67     .show-menu {
68         transform: translate(40%);
69     }
70 }

```



Nota: Tenga en cuenta que estos valores pueden variar según su gusto; realice cambios y obtenga los que considere óptimos.

Proyecto: diseñe un modelo para dispositivos 4K (mas de 1921px).