

Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.

Maquetando con SASS y HTML5

En esta ocasión me dispuse a trabajar en un proyecto para una aplicación web utilizando el famoso pre-procesador **SASS** (para Hojas Estilo de Cascada) y algo de **HTML5** (Hypertext Markup Language) para su maquetación.

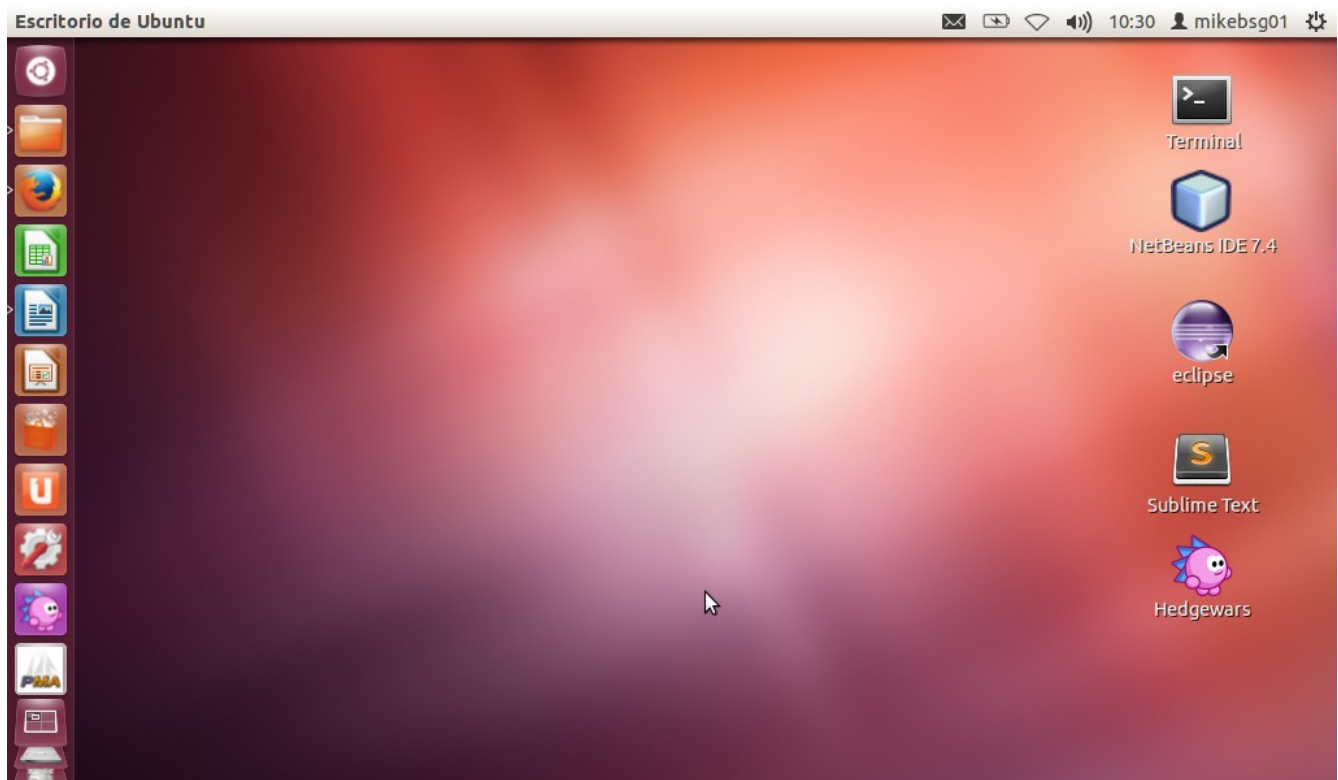
En sí el propósito de este post, es dar a conocer una excelente forma de aplicar SASS en nuestros proyectos posteriores y comprender más acerca de la funcionalidad del mismo.

REQUISITOS PARA ESTE PROYECTO:

- Conocimientos de HTML5 y CSS3 - **AMPLIOS**
- Conocimientos de la Estructura Semántica de HTML5 - **BÁSICOS**
- Conocimientos sobre Diseño de aplicaciones (en especial **Responsive Design**) - **BÁSICOS**
- Conocimientos del pre-procesador SASS – **BÁSICOS O CASI NULOS** (*No se requieren*).

NOTA: Si no cuentas con SASS instalado en tu ordenador no te preocupes!!, en html5facil.com ya contamos con una guía de instalación muy completa para cualquiera de los distintos Sistemas Operativos (Windows, Linux y Mac).

Bueno, sin más que decir comencemos nuestro proyecto :)

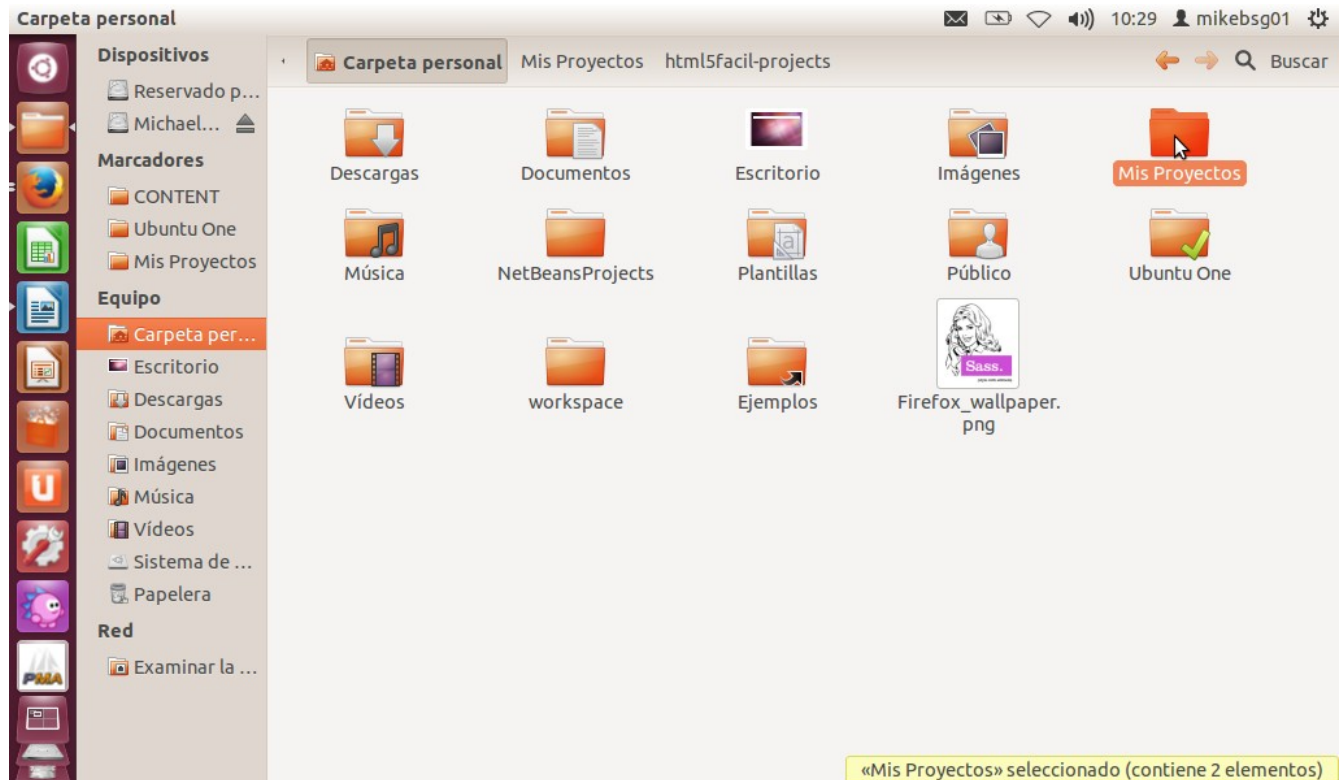


Autor: Michael Serrato. Twitter: @mikebsg01

Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.

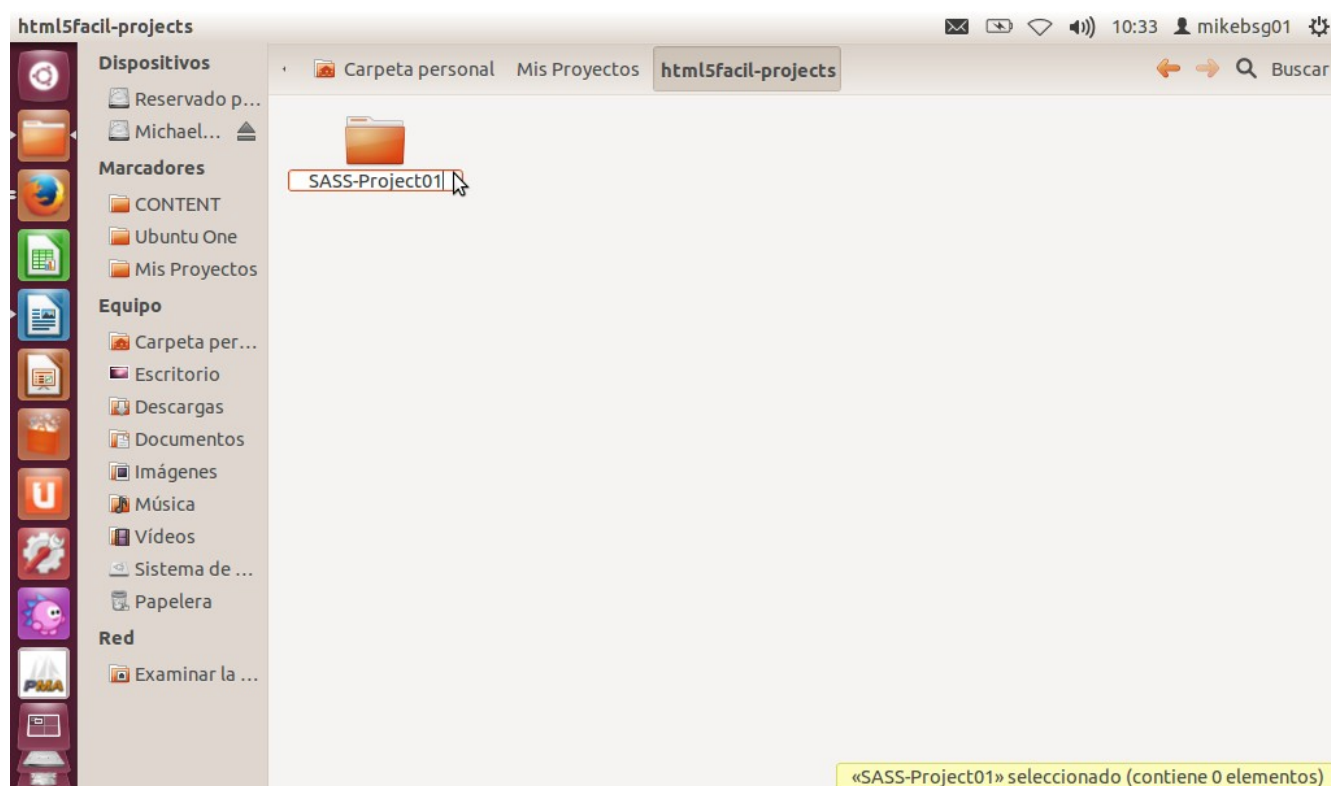
En este momento me encuentro en mi escritorio.

Para iniciar vamos a crear una carpeta que contendrá todos los archivos de nuestro proyecto. En mi caso yo tengo una carpeta donde guardo todos mis proyectos así que me voy a posicionar en ese directorio.



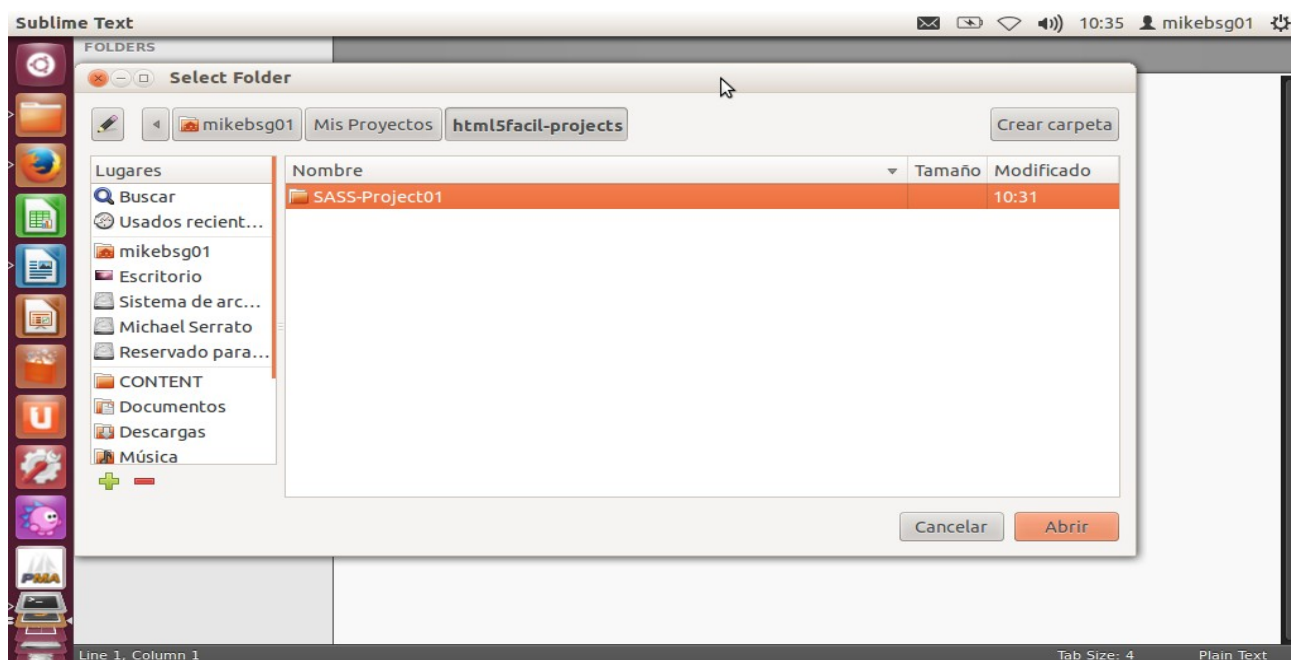
Una vez posicionado creamos la carpeta para el proyecto, yo la nombraré “**SASS-Project01**” (ustedes pueden nombrarla de cualquier forma, no hay ningún problema).

Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.

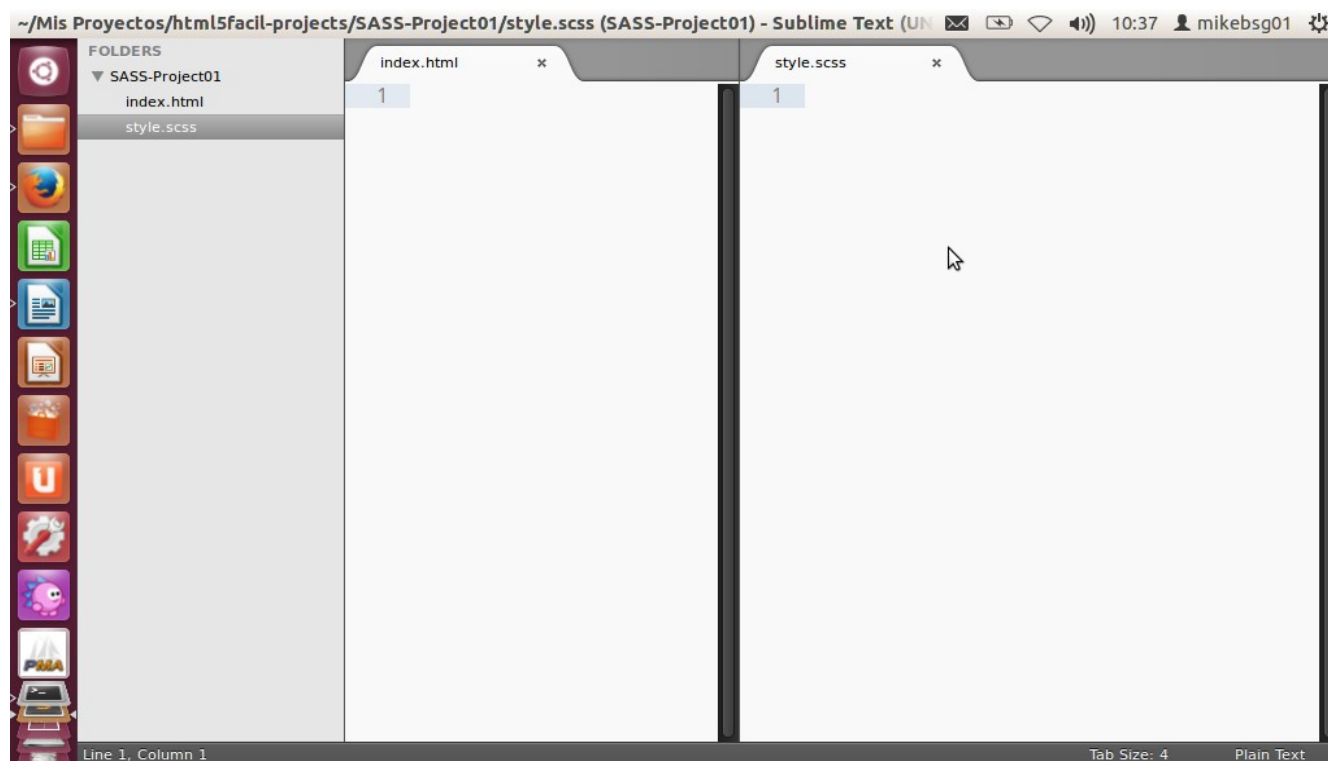


Hecho esto, lo siguiente es abrir la carpeta con nuestro IDE favorito, en mi caso yo utilizaré **Sublime Text** ya que es muy rápido y práctico a la hora de escribir código en él, aunque sin duda alguna ustedes pueden utilizar el Entorno que más les agrade.

Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.



En este proyecto únicamente vamos a crear 2 archivos con el IDE, uno para el HTML (Archivo: **index.html**), y otro para generar los estilos CSS con SASS el cual tendrá una extensión “.scss” (Archivo: **style.scss**). Hasta el momento los archivos de nuestro proyecto y del entorno se verá de la siguiente forma:



Autor: Michael Serrato. Twitter: @mikebsg01

TRABAJANDO CON SASS DESDE LA TERMINAL

Como ya sabrán aquellos que ya han trabajado con SASS se necesita de la terminal de comandos para poder generar los archivos “.css” que contendrán los estilos CSS en un determinado formato y orden.

En SASS se trabaja de la siguiente forma:

```
sass -watch entrada.scss:salida.css
```

¿Pero... Qué es lo que hace este comando?

Sass – indica que haremos uso del pre-procesador SASS

--watch – este comando en particular hará que SASS procese el código en tiempo de desarrollo.

Entrada.scss – indica cual será el archivo de entrada.

: - sirve para indentificar cual será el archivo de salida.

Salida.css – y finalmente éste, indicará cual será el archivo de salida, o mejor dicho el archivo generado con los estilos CSS procesados por SASS.

De igual forma, para dar mejor a entender el comando, se puede ejemplificar de la siguiente manera:

```
sass -watch archivo_entrada.scss:archivo_salida.css
```

NOTA: Para conocer todos los parametros que recibe SASS, ejecuta en la terminal el comando `sass -help`.

IMPORTANTE: No es obligatorio trabajar con SASS desde la terminal, una de las opciones es utilizar SCOUT que permite gestionar los archivos .scss y .css desde una fácil y sencilla interfaz gráfica.

Vease en el artículo: [Guía básica de instalación de SASS – html5facil.com](http://html5facil.com)

FORMATOS

Un tema muy importante al desarrollar grandes aplicaciones es el formato en el que se obtendrán los archivos .css, esto debido a que cada uno le dará distinta utilidad a su código.

Existen 4 tipos de formatos:

nested (formato *por defecto*)

compact (compacto)

compressed (comprimido)

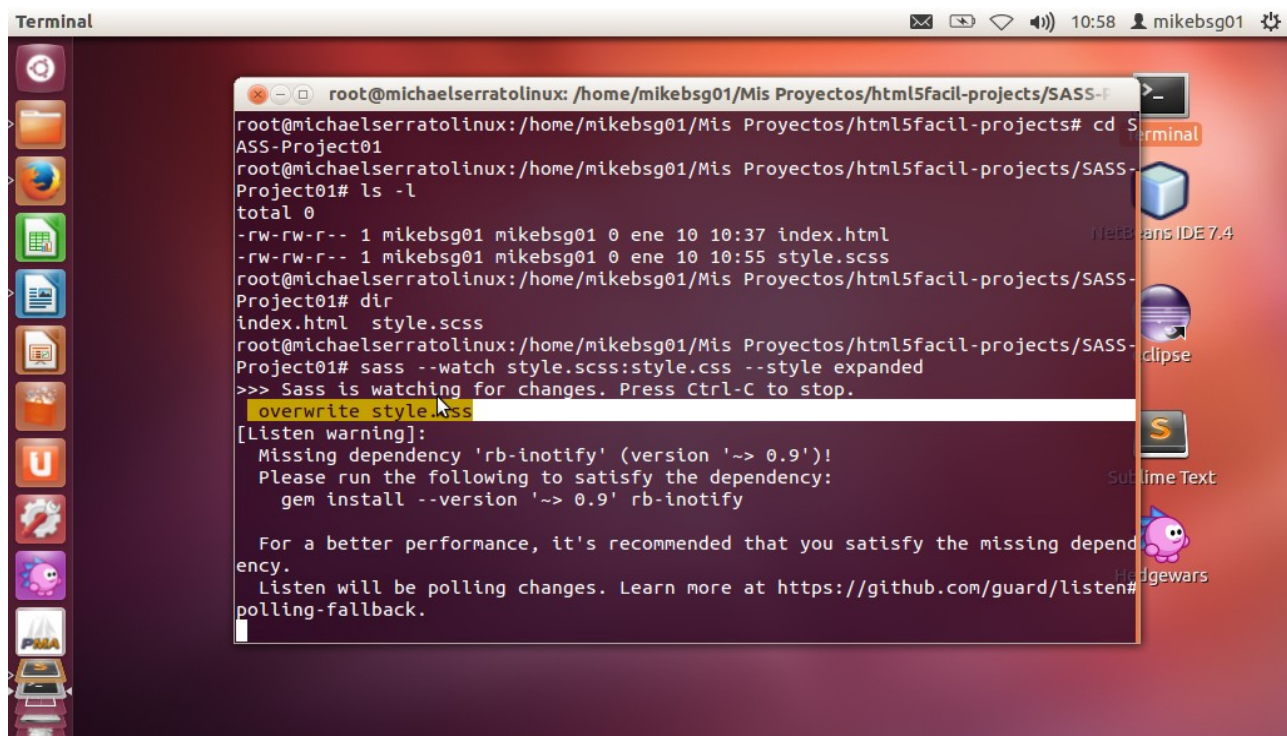
expanded (expandido)

Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.

Para este proyecto decidí utilizar el formato “expanded”, ya que este sencillo proyecto será de código abierto y al finalizarlo la lectura del código generado será mucho más fácil para aquellos desarrolladores que quieran hechar un vistazo.

Un mensaje así, tendrá que salir en la terminal:

>>> Sass is watching for changes. Press Ctrl-C to stop.



```
Terminal
root@michaelserratolinux: /home/mikebsg01/Mis Proyectos/html5facil-projects/SASS-Project01
root@michaelserratolinux: /home/mikebsg01/Mis Proyectos/html5facil-projects/SASS-Project01# cd SASS-Project01
root@michaelserratolinux: /home/mikebsg01/Mis Proyectos/html5facil-projects/SASS-Project01# ls -l
total 0
-rw-rw-r-- 1 mikebsg01 mikebsg01 0 ene 10 10:37 index.html
-rw-rw-r-- 1 mikebsg01 mikebsg01 0 ene 10 10:55 style.scss
root@michaelserratolinux: /home/mikebsg01/Mis Proyectos/html5facil-projects/SASS-Project01# dir
index.html style.scss
root@michaelserratolinux: /home/mikebsg01/Mis Proyectos/html5facil-projects/SASS-Project01# sass --watch style.scss:style.css --style expanded
>>> Sass is watching for changes. Press Ctrl-C to stop.
overwrite style.scss
[Listen warning]:
Missing dependency 'rb-inotify' (version '~> 0.9')!
Please run the following to satisfy the dependency:
gem install --version '~> 0.9' rb-inotify

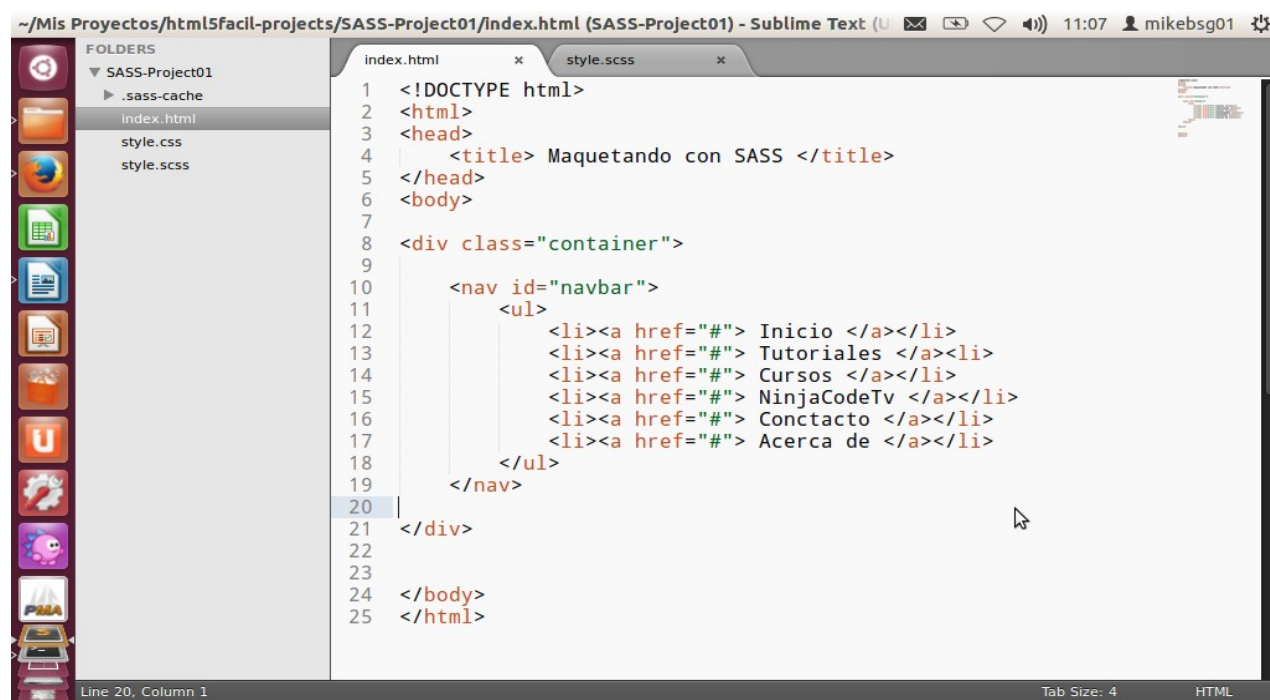
For a better performance, it's recommended that you satisfy the missing dependency.
Listen will be polling changes. Learn more at https://github.com/guard/listen#polling-fallback.
```

Bien, una vez explicado esto es hora de codear :D

¿Qué les parece si iniciamos con un navbar para el diseño? Algo parecido a esto:

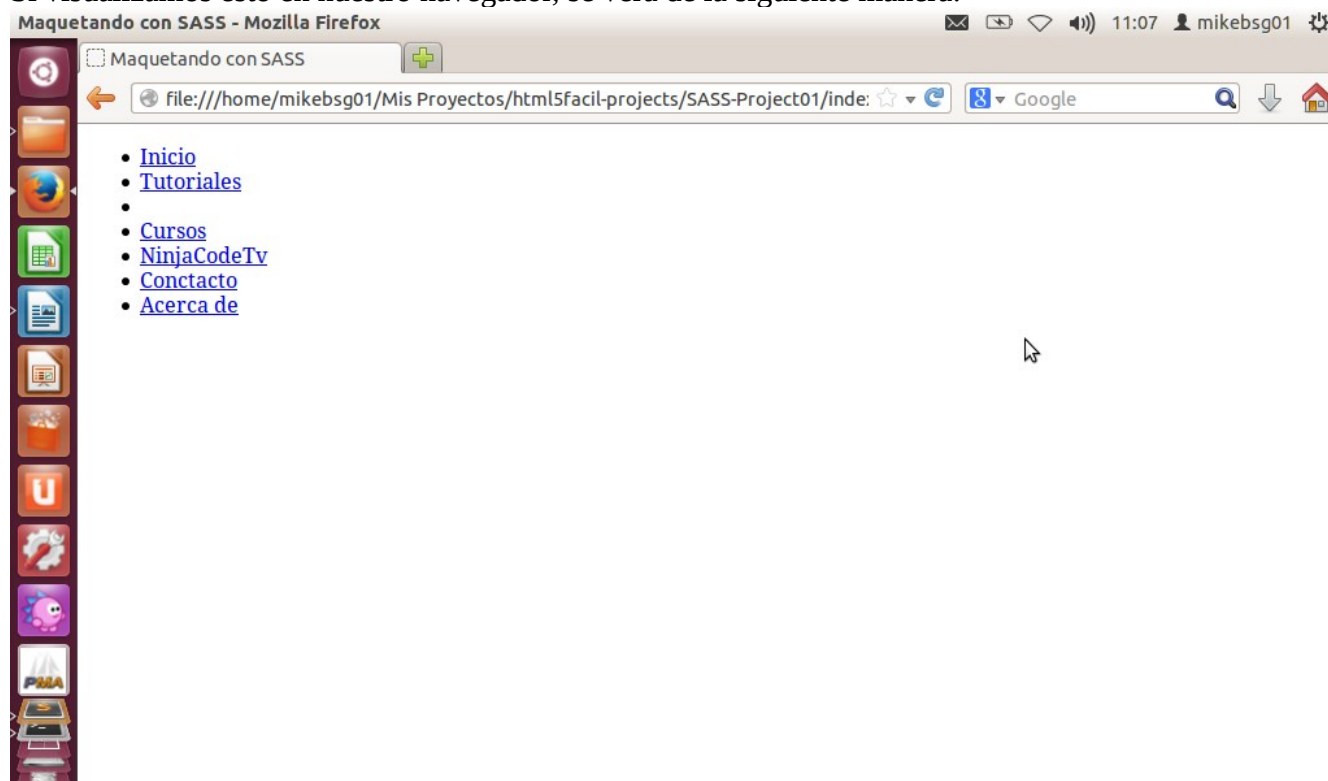
Autor: Michael Serrato. Twitter: @mikebsg01

Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title> Maquetando con SASS </title>
5 </head>
6 <body>
7
8 <div class="container">
9
10   <nav id="navbar">
11     <ul>
12       <li><a href="#"> Inicio </a></li>
13       <li><a href="#"> Tutoriales </a></li>
14       <li><a href="#"> Cursos </a></li>
15       <li><a href="#"> NinjaCodeTv </a></li>
16       <li><a href="#"> Contacto </a></li>
17       <li><a href="#"> Acerca de </a></li>
18     </ul>
19   </nav>
20
21 </div>
22
23
24 </body>
25 </html>
```

Si visualizamos esto en nuestro navegador, se verá de la siguiente manera:



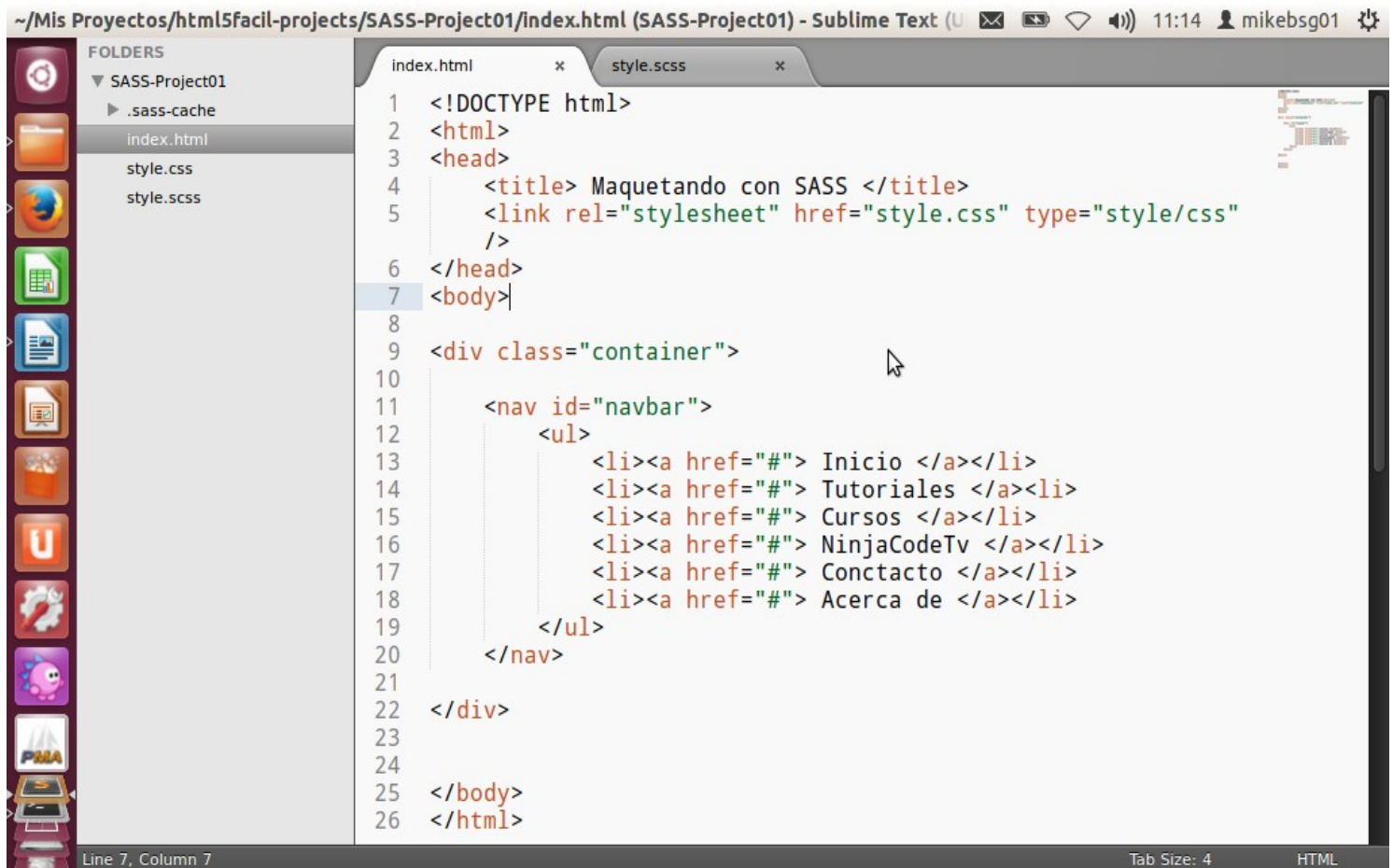
Autor: Michael Serrato. Twitter: @mikebsg01

Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.

Nota importante: Cometí un pequeño error al cerrar la etiqueta “li” como se puede ver en la pantalla entre la lista “Tutoriales” y “Cursos”, por motivos de tiempo y organización del post no pude corregir esa impresión de pantalla. Espero puedan notar ese error en el código html y corregirlo, es algo sencillo.

Bien, ahora regresemos a lo importante, como ya saben una vez que ejecutamos el comando desde la terminal SASS estará esperando en tiempo real cuando se realice algún cambio en el archivo “style.scss”, por lo que cualquier nuevo cambio actualizará el archivo generado “style.css”.

Sabiendo esto, es necesario agregar la hoja estilo de cascada generada por SASS a nuestro documento de HTML:



```
~/Mis Proyectos/html5facil-projects/SASS-Project01/index.html (SASS-Project01) - Sublime Text (U) 11:14 mikebsg01
FOLDERS
▼ SASS-Project01
  ► .sass-cache
  index.html
  style.css
  style.scss

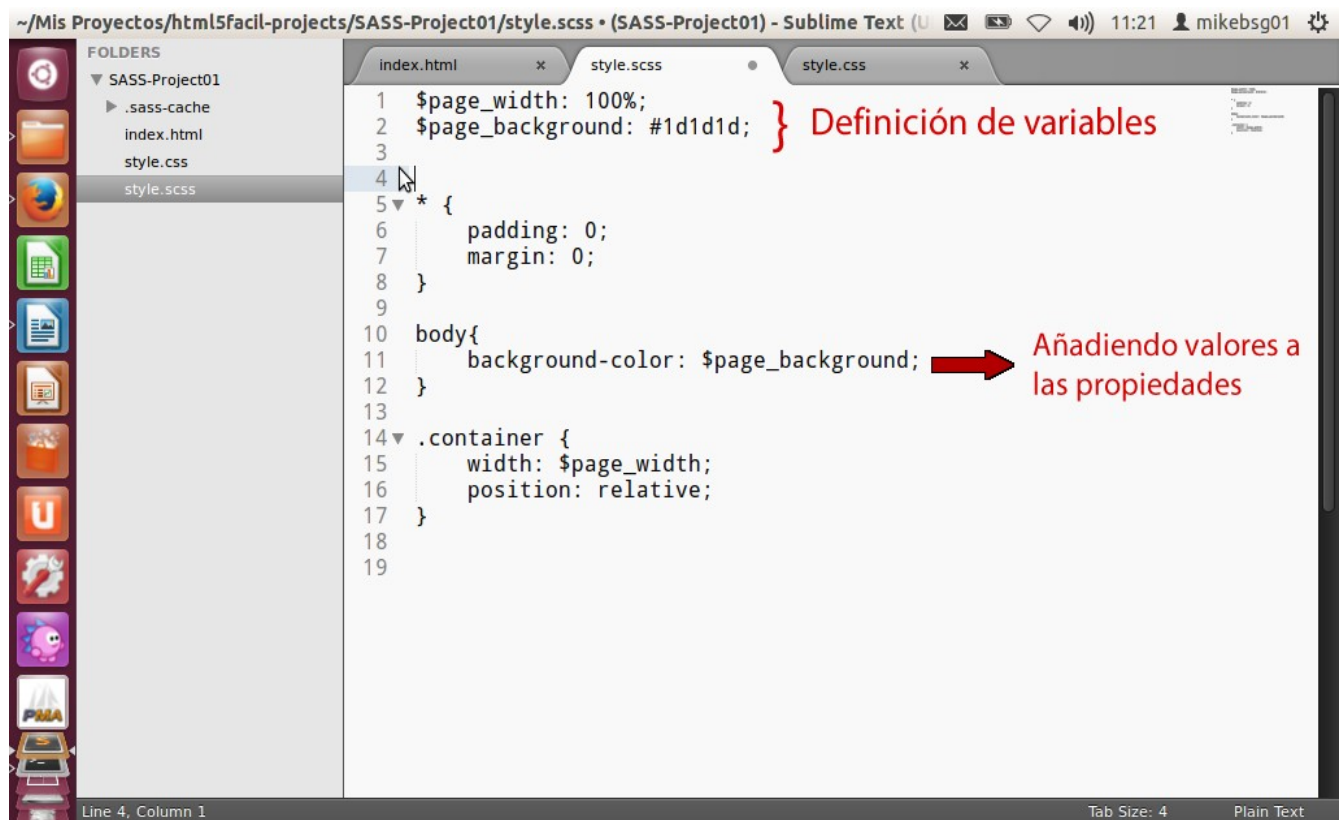
index.html
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title> Maquetando con SASS </title>
5   <link rel="stylesheet" href="style.css" type="style/css"
6   />
7 </head>
8 <body>
9   <div class="container">
10
11     <nav id="navbar">
12       <ul>
13         <li><a href="#"> Inicio </a></li>
14         <li><a href="#"> Tutoriales </a></li>
15         <li><a href="#"> Cursos </a></li>
16         <li><a href="#"> NinjaCodeTv </a></li>
17         <li><a href="#"> Conctacto </a></li>
18         <li><a href="#"> Acerca de </a></li>
19       </ul>
20     </nav>
21
22   </div>
23
24
25 </body>
26 </html>
```

Listo, aunque el archivo “style.css” aun no se ha generado debemos tenerlo agregado de una vez en nuestro documento puesto que es el archivo de salida que nos servirá para maquetar nuestra aplicación.

Ahora veamos algo de código en SASS:

Autor: Michael Serrato. Twitter: @mikebsg01

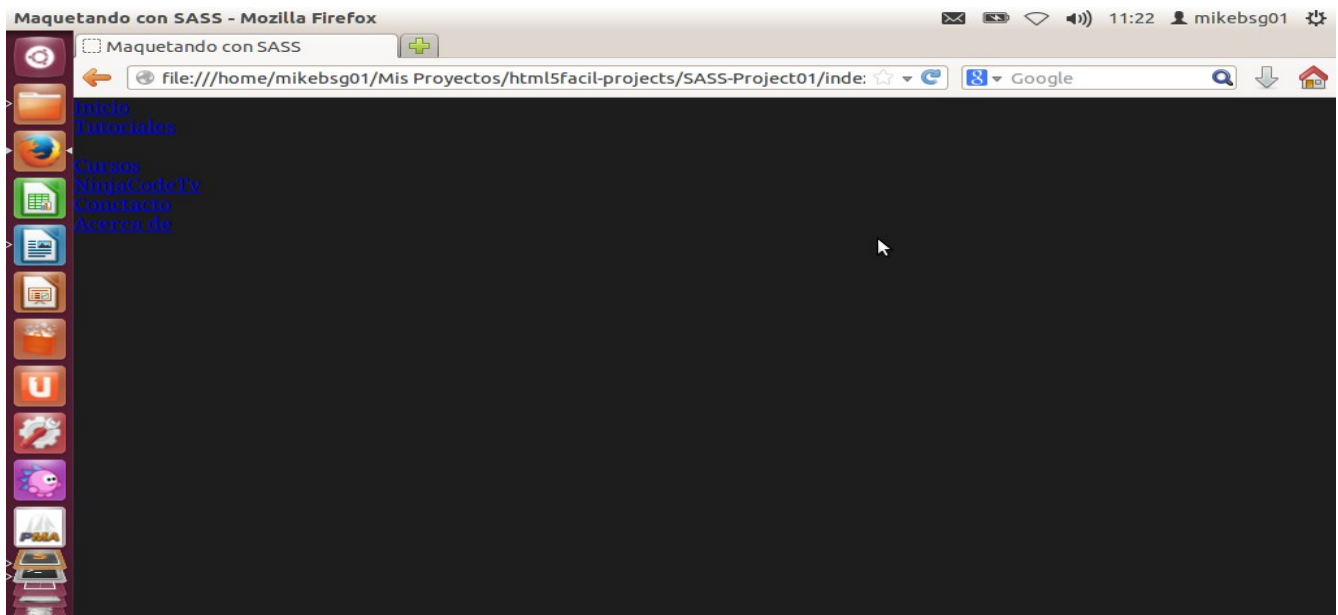
Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.



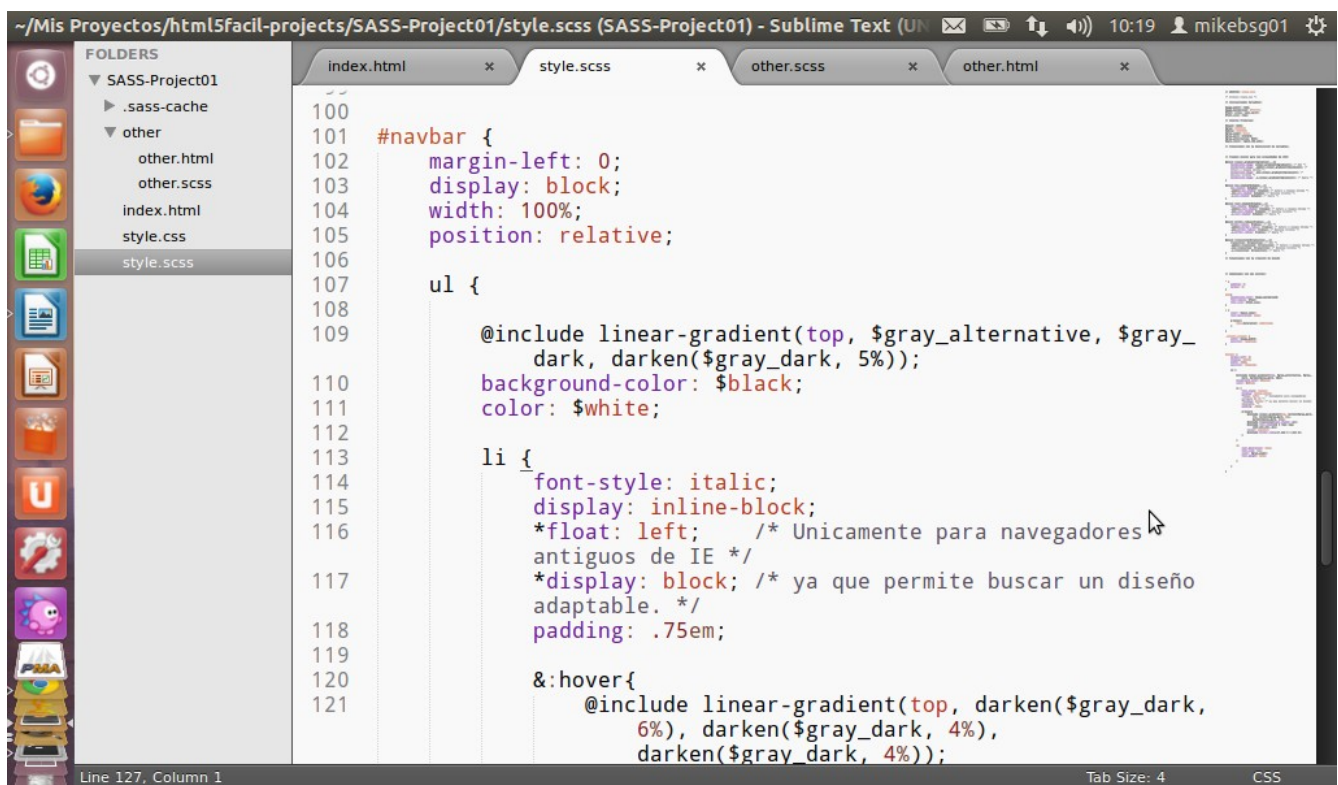
Cómo podrán ver el código en SASS es un poco diferente a como se trabaja directamente en CSS, sin embargo, este es un sencillo ejemplo, puesto que SASS nos permite hacer bastantes cosas que CSS no puede lograr, entre ellas: evitar escribir varias veces el mismo código, darle uniformidad y buena presentación a nuestras aplicaciones, editar propiedades fácilmente desde las variables, la creación de mixins, el uso de directivas del pre-procesador SASS pero principalmente ahorrar horas de trabajo en el desarrollo de aplicaciones web.

Luego de esta importante explicación vamos a visualizar cómo quedó este sencillo ejemplo en nuestro navegador:

Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.

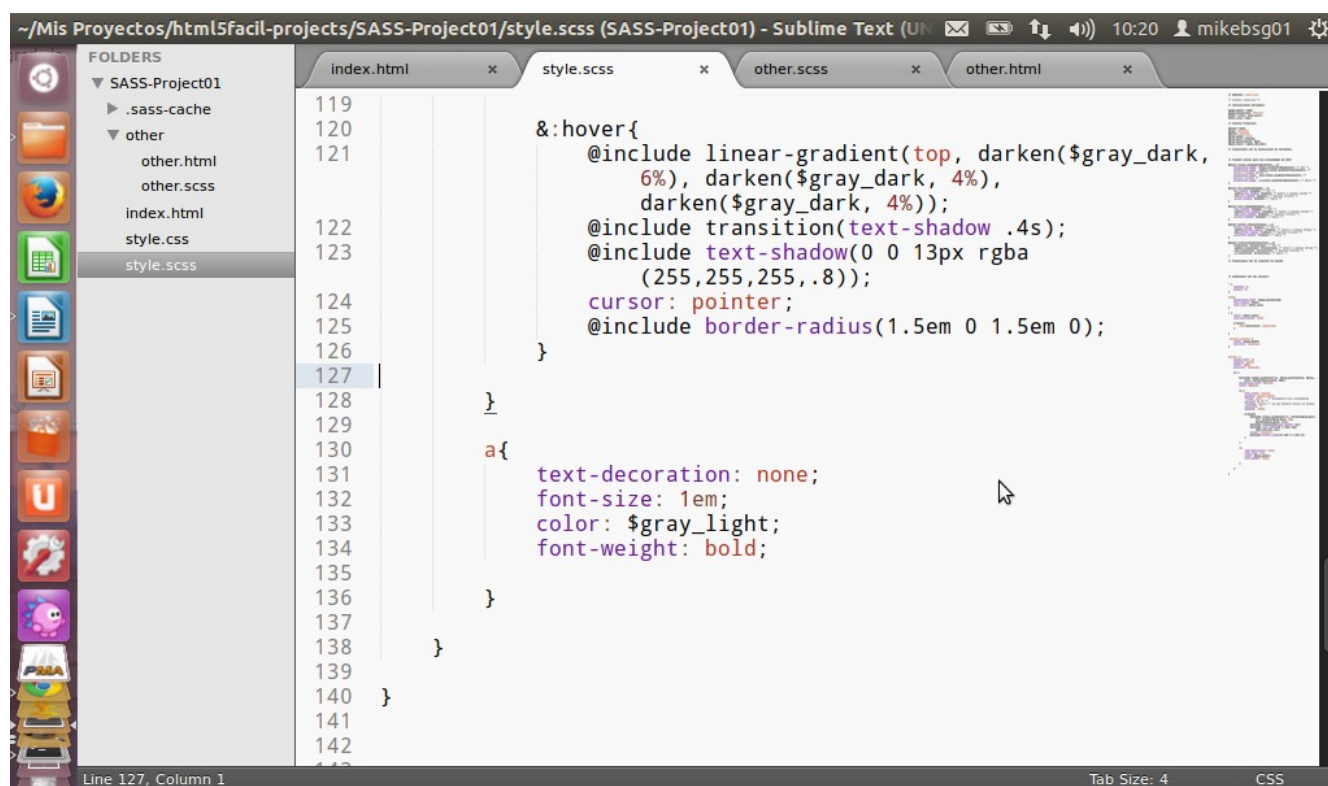


Correcto, los estilos se han agregado en el documento HTML de la manera esperada, así que continuemos con los estilos del **navbar**:



Autor: Michael Serrato. Twitter: @mikebsg01

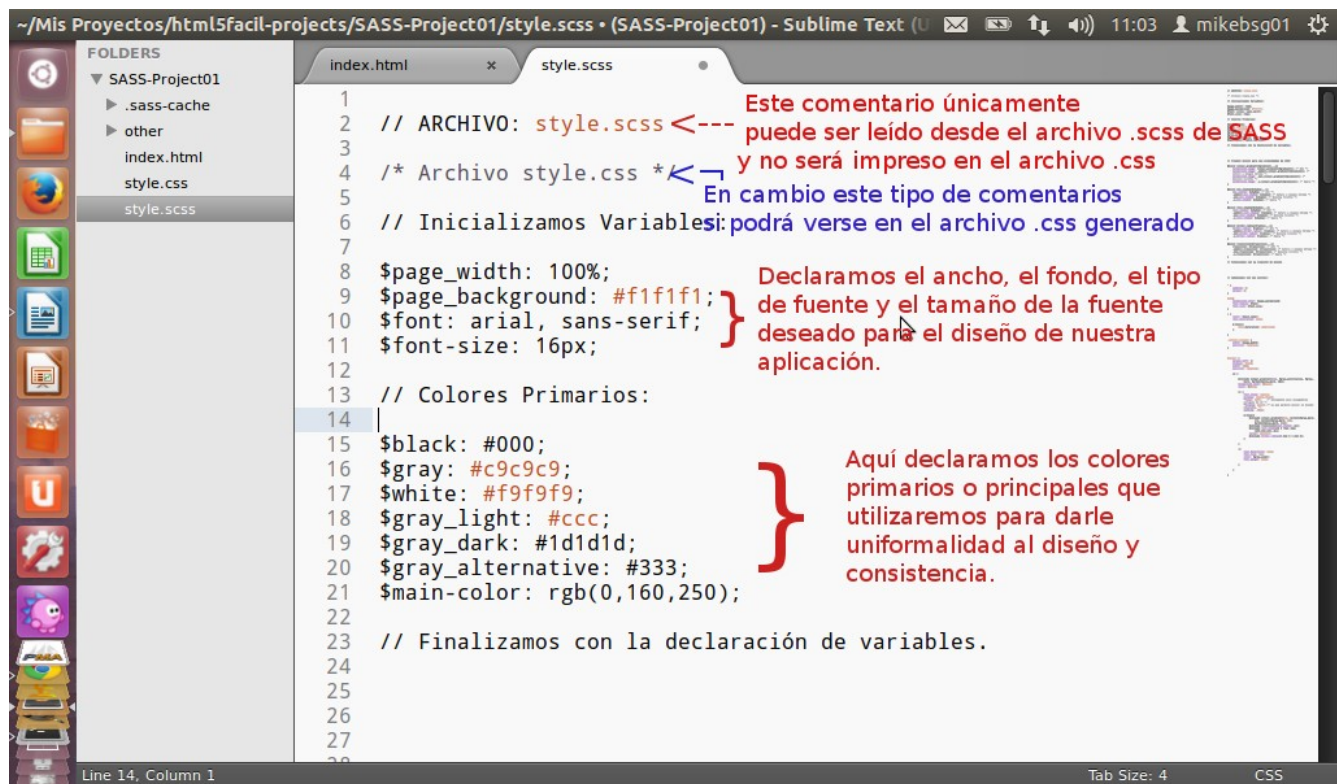
Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.



VARIABLES

Como habrán notado estoy haciendo uso de “funciones” y “variables” distintas a las establecidas en un principio al crear el archivo “style.scss”, es por eso que muestro la siguiente captura de pantalla en donde ubiqué las variables principales que estaremos utilizando durante este proyecto:

Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.



NOTA: La declaración de variables se encuentra en el mismo archivo “style.scss” justo en la parte del inicio.

Ahora podremos darnos cuenta la estructura de nuestros estilos y una de las formas en las que podemos utilizar sass logrando así tener un código ordenado y limpio.

IMPORTANTE: En el uso de pre-procesadores siempre es una excelente idea declarar los colores principales para el diseño ya que en un momento futuro si es necesario cambiar estructura de colores en el diseño de la aplicación únicamente será necesario cambiar el valor de las variables evitando así tener que cambiar las propiedades de cada clase (o elemento que se desee cambiar) una por una.

Excelente, ya hemos visualizado la declaración de las variables, pero como vimos en la imagen de nuestro código en SASS aún falta ver eso que la mayoría conoce en un lenguaje de programación como “funciones”.

MIXINS

Los mixins no son más que una sección de código. Permiten reutilizar estilos las veces necesarias con la finalidad de no repetir un mismo código.

Pero cómo se crea un **mixin**?

Autor: Michael Serrato. Twitter: @mikebsg01

Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.

Los mixins se crean con la necesidad de no tener que repetir más código en nuestro archivo logrando así ahorrar tiempo y trabajo en la creación de los estilos CSS para el diseño de la aplicación. En este ejemplo podrán ver una manera sencilla de declarar un **mixin**:

Así se vería nuestro código sin el uso de mixins, ahora si pueden observar se darán cuenta que repetimos las mismas propiedades en ambos casos a excepción del ancho (la propiedad width) en cada selector.

```
.button-small {  
    width: 120px;  
    border-radius: 3px 3px;  
    color: #ffffff;  
    background: #1d1d1d;  
    height: 60px;  
}  
  
.button-large {  
    width: 260px;  
    border-radius: 3px 3px;  
    color: #ffffff;  
    background: #1d1d1d;  
    height: 60px;  
}
```

Con el uso de **mixins** esta sería la forma correcta de realizar el código en SASS:

```
@mixin button(){  
    border-radius: 3px 3px;  
    color: #ffffff;  
    background: #1d1d1d;  
    height: 60px;  
}  
  
.button-small {  
    @include button();  
    width: 120px;  
}  
  
.button-large {  
    @include button();  
    width: 260px;  
}
```

NOTA IMPORTANTE: La directiva mixin **@include** seguido del nombre permite mandar a llamar

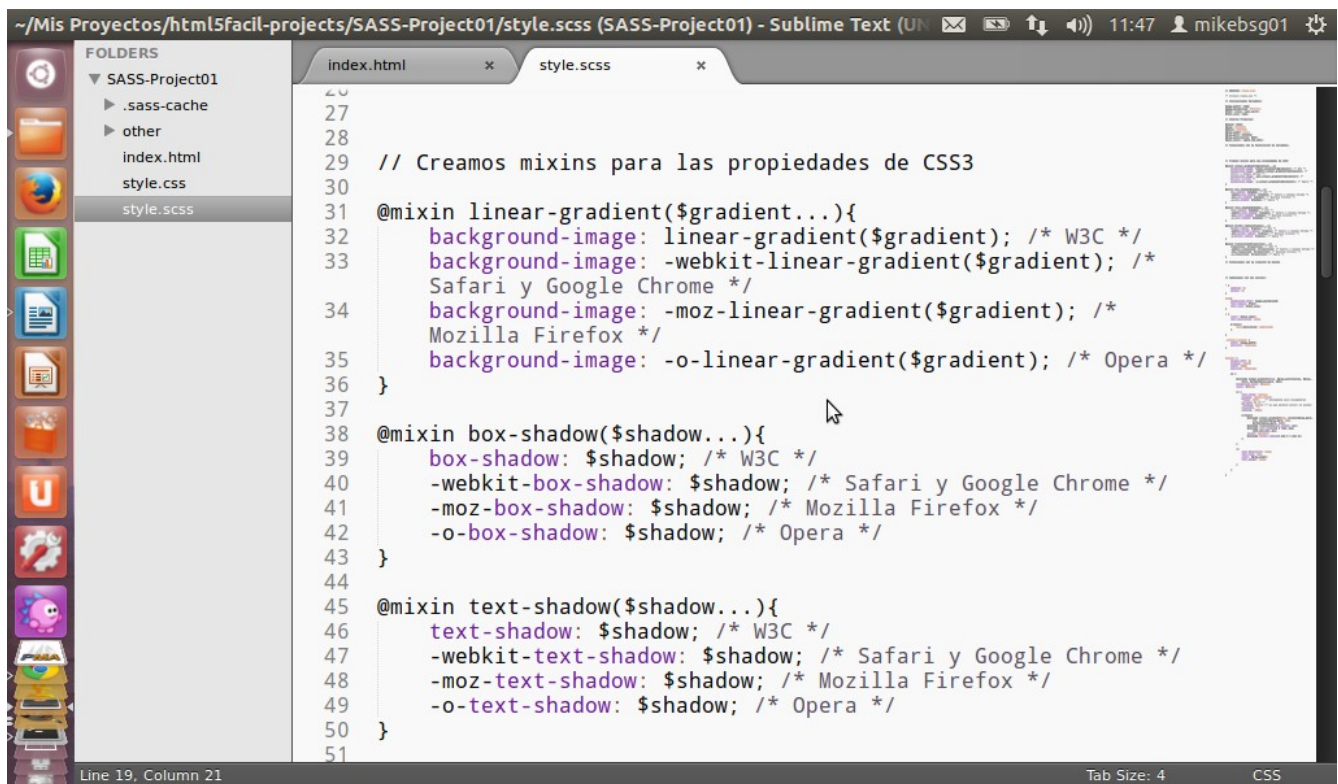
Autor: Michael Serrato. Twitter: @mikebsg01

Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.

una sección de código **mixin**.

Después de haber entendido y estudiado la función y utilidad de los mixins, es hora de mostrar los **mixins añadidos** a nuestro archivo “style.scss”

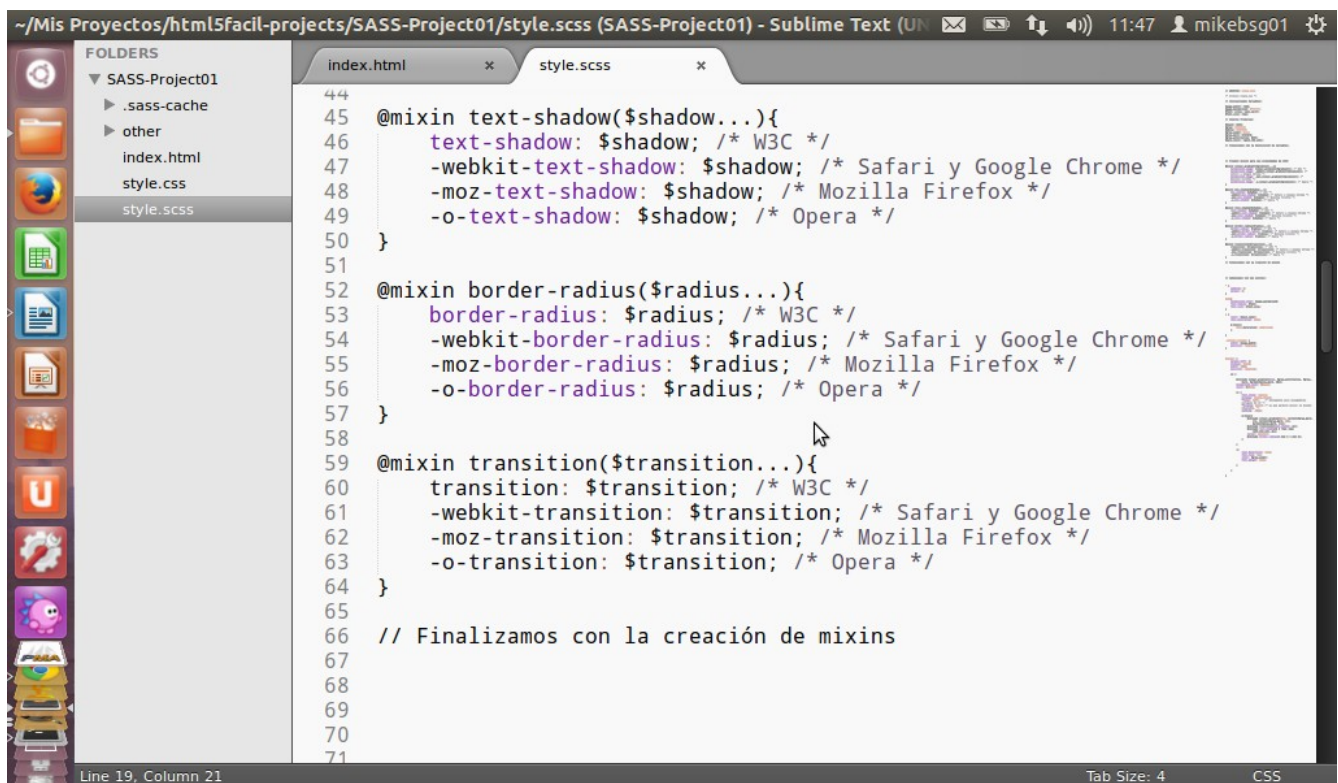
Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.



```
27
28
29 // Creamos mixins para las propiedades de CSS3
30
31 @mixin linear-gradient($gradient...){
32     background-image: linear-gradient($gradient); /* W3C */
33     background-image: -webkit-linear-gradient($gradient); /*
34     Safari y Google Chrome */
35     background-image: -moz-linear-gradient($gradient); /*
36     Mozilla Firefox */
37     background-image: -o-linear-gradient($gradient); /* Opera */
38 }
39
40 @mixin box-shadow($shadow...){
41     box-shadow: $shadow; /* W3C */
42     -webkit-box-shadow: $shadow; /* Safari y Google Chrome */
43     -moz-box-shadow: $shadow; /* Mozilla Firefox */
44     -o-box-shadow: $shadow; /* Opera */
45 }
46
47 @mixin text-shadow($shadow...){
48     text-shadow: $shadow; /* W3C */
49     -webkit-text-shadow: $shadow; /* Safari y Google Chrome */
50     -moz-text-shadow: $shadow; /* Mozilla Firefox */
51     -o-text-shadow: $shadow; /* Opera */
52 }
```

Line 19, Column 21

Tab Size: 4 CSS



```
44
45 @mixin text-shadow($shadow...){
46     text-shadow: $shadow; /* W3C */
47     -webkit-text-shadow: $shadow; /* Safari y Google Chrome */
48     -moz-text-shadow: $shadow; /* Mozilla Firefox */
49     -o-text-shadow: $shadow; /* Opera */
50 }
51
52 @mixin border-radius($radius...){
53     border-radius: $radius; /* W3C */
54     -webkit-border-radius: $radius; /* Safari y Google Chrome */
55     -moz-border-radius: $radius; /* Mozilla Firefox */
56     -o-border-radius: $radius; /* Opera */
57 }
58
59 @mixin transition($transition...){
60     transition: $transition; /* W3C */
61     -webkit-transition: $transition; /* Safari y Google Chrome */
62     -moz-transition: $transition; /* Mozilla Firefox */
63     -o-transition: $transition; /* Opera */
64 }
65
66 // Finalizamos con la creación de mixins
67
68
69
70
71
```

Line 19, Column 21

Tab Size: 4 CSS

Autor: Michael Serrato. Twitter: @mikebsg01

Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.

Si ya se dieron cuenta, entonces notaron que estos mixins únicamente permiten agregar código de los estilos CSS3 más comunes, esto parece no ser muy útil al principio pero es muy importante para los proyectos grandes en donde las aplicaciones diseñadas con estos estilos pretenden ser 100% compatibles y adaptables en casi todos los navegadores (véase en los post de **responsive design**), en lugar de escribir todos y cada uno de los prefijos que hay para cada navegador únicamente mandamos los parametros al mixin seleccionado con nuestros valores o propiedades y automáticamente se agregarán los prefijos para así lograr la perfecta visualización de la aplicación en cualquier navegador deseado.

NOTA IMPORTANTE: Actualmente ya existen librerías que ya hacen este trabajo por ti, incluso gemas para **Rails** que permiten hacer uso de **SASS** con los mixins de CSS3 ya incluidos, un ejemplo sería el de la librería **Bourbon** que se puede encontrar en la siguiente página: <http://bourbon.io/>.

NESTING (ANIDAMIENTO)

Hasta este punto ya hemos visto todo el código necesario para los próximos estilos que faltan por agregar. Sin embargo, aún falta un último punto por estudiar y es el de “Anidamiento”. Ahora es momento de echar un vistazo a la imagen donde se ubica el código de nuestro **navbar**, para aquellos que ya han trabajado mucho con **CSS** se les hará un poco raro que se encuentren “**selectores** dentro de otros **selectores**”, bien aquí va la explicación, en SASS es posible realizar algo conocido como **anidamiento** que es la posibilidad de incorporar una estructura de código dentro de otra y que en la mayor parte de las ocasiones es requerido utilizar una sangría para poder separar y distinguir estos bloques de código (Ok, perfectamente entendido para los que han practicado con lenguajes de programación como en C/C++ o en JAVA, pero para quienes no, específico) esto quiere decir que podemos agregar un bloque de código dentro de otro, por ejemplo, si quisiéramos añadir ciertas propiedades a un elemento que se encuentra dentro de otro en nuestro documento **HTML** entonces haríamos lo siguiente:

```
.elemento-padre {  
    // Código del padre. . .  
    .elemento-hijo {  
        // Código del hijo  
    }  
}
```

Esto sin duda al ser procesador por SASS quedaría de esta forma:

```
.elemento-padre{  
    // Código del padre  
}
```

Autor: Michael Serrato. Twitter: @mikebsg01

Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.

```
.elemento-padre .elemento-hijo {  
    // Código del hijo  
}
```

En un caso en particular podría ser cuando deseamos añadir código a la **pseudo-clase** **:hover** de un botón, por ejemplo:

```
button {  
    background-color: gray;  
    color: white;  
  
    &:hover {  
        background-color: blue;  
    }  
}
```

Lo cual en CSS quedaría de la siguiente forma:

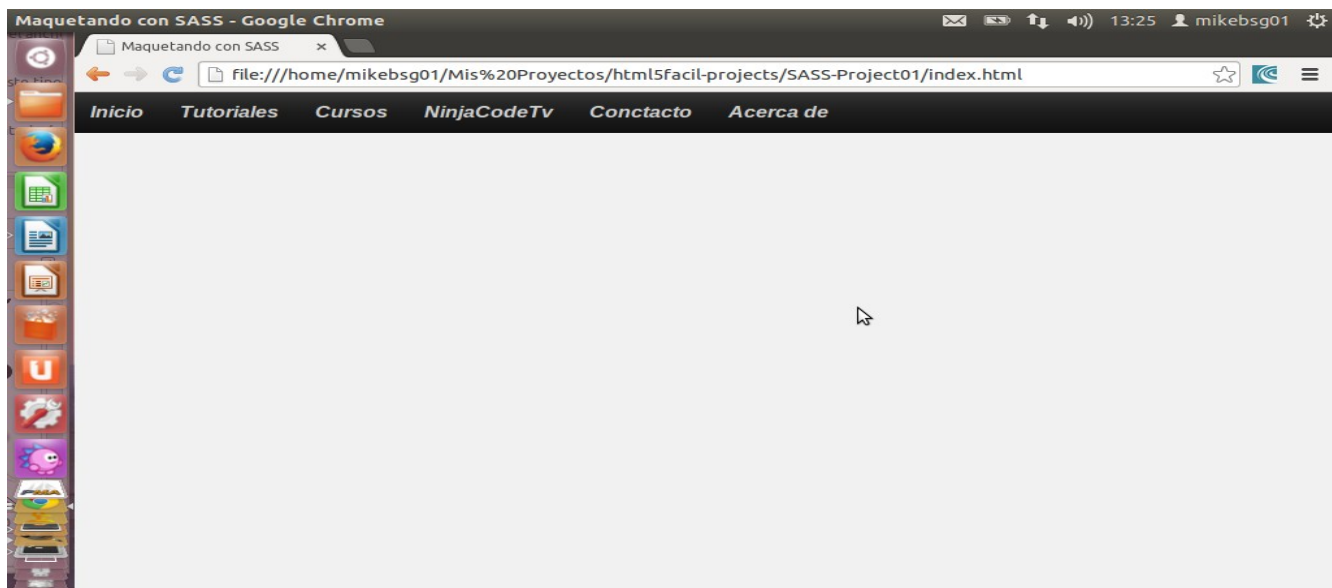
```
button {  
    background-color: gray;  
    color: white;  
}  
  
button:hover {  
    background-color: blue;  
}
```

NOTA IMPORTANTE: El carácter **&** permite hacer referencia al padre, por lo que al hacer uso de él dentro de un selector estaremos agregando código a la pseudo-clase de dicho elemento.

Perfecto, luego de esta amplia explicación estarán gustosos por saber que ya han aprendido lo más elemental de SASS. Lo único que nos queda es finalizar con la maquetación de nuestro proyecto.

DISEÑO DEL NAVBAR

Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.



Hasta el momento con el código anteriormente agregado tenemos ya completado el diseño de nuestro menu de navegación o barra de navegación (**navbar**).

DISEÑO DEL HEADER

Para el diseño de la etiqueta header sólo es necesario agregar una etiqueta div con el id welcome de esta manera:

```
<header>
  <div id="welcome">
  </div>
</header>
```

Aquí el código de los estilos:

```
#welcome {

    width: $page_width;
    display: block;
    margin: auto;
    height: 256px;
    background-image: url("images/html5facil_logo.png"); // PARA ESTE PROYECTO YO
                                                         // UTILICÉ EL LOGO DE HTML5FACIL
                                                         // COMO FONDO

    background-color: $gray_dark;
    background-size: 80% 160%;
    background-position: center center;
    background-repeat: no-repeat;
```

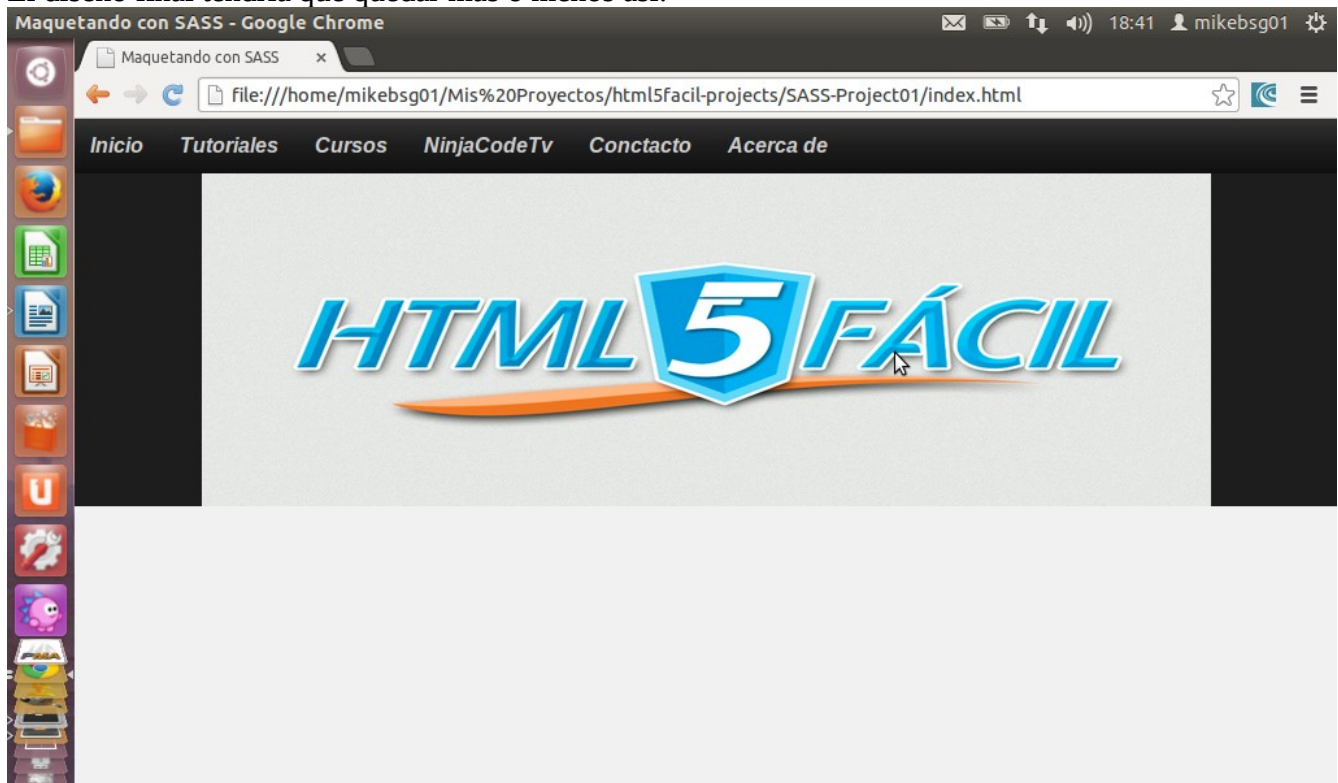
Autor: Michael Serrato. Twitter: @mikebsg01

Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.

}

IMPORTANTE: La imagen utilizada para el fondo la he ubicado dentro de una carpeta nombrada “images”.

El diseño final tendría que quedar más o menos así:



DISEÑO DEL SECTION, ARTICLE y ASIDE

En esta parte de nuestro proyecto se preguntarán por qué unimos el diseño del **section** con el **article** y el **aside**, bien la explicación es sencilla, tomando en cuenta que la **Estructura Semántica de HTML5** es nuestra base, tenemos que saber que al diseñar de una forma cada uno de estos elementos dependerá mucho de como se verá el otro, por ejemplo, si hacemos un **section** más pequeño a su tamaño acostumbrado el margen que habrá entre el article y el aside deberá cambiar ya que si no se realiza esto el diseño se estropeará por completo, es por eso que es necesario saber las medidas exactas con las cuales estaremos trabajando para cada uno de estos elementos.

Código HTML:

Autor: Michael Serrato. Twitter: @mikebsg01

Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.

```
<section>
```

```
<div class="content">
```

```
<article>
```

```
<hgroup>
```

```
<h1> Maquetando con SASS y HTML5 </h1>
```

```
</hgroup>
```

```
<div class="date">Ultima modificación: 10/01/2014 </div>
```

```

```

```
<p>
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vivamus iaculis erat a ultrices pharetra. Nullam bibendum, felis a faucibus pellentesque, est sapien feugiat est, ac bibendum justo massa ac enim. Maecenas lectus felis, elementum et arcu sit amet, bibendum sollicitudin mauris. Interdum et malesuada fames ac ante ipsum primis in faucibus. Vestibulum ligula mauris, vulputate sed neque vitae, suscipit venenatis diam. Sed molestie lorem sed mi rhoncus malesuada. Etiam hendrerit nisl et eros ultrices pharetra. Vestibulum non ante blandit, pulvinar augue sagittis, pulvinar magna. Aliquam bibendum quis libero eu sollicitudin. Pellentesque eget tincidunt enim, id tincidunt turpis. Nullam vitae urna et elit semper sagittis at sit amet metus.

```
</p>
```

```
<p>
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vivamus iaculis erat a ultrices pharetra. Nullam bibendum, felis a faucibus pellentesque, est sapien feugiat est, ac bibendum justo massa ac enim. Maecenas lectus felis, elementum et arcu sit amet, bibendum sollicitudin mauris. Interdum et malesuada fames ac ante ipsum primis in faucibus. Vestibulum ligula mauris, vulputate sed neque vitae, suscipit venenatis diam. Sed molestie lorem sed mi rhoncus malesuada. Etiam hendrerit nisl et eros ultrices pharetra. Vestibulum non ante blandit, pulvinar augue sagittis, pulvinar magna. Aliquam bibendum quis libero eu sollicitudin. Pellentesque eget tincidunt enim, id tincidunt turpis. Nullam vitae urna et elit semper sagittis at sit amet metus.

```
</p>
```

```
</article>
```

```
<div class="comments-form">
```

```
<form action="#" method="post">
```

```
<p class="input">Nombre: <input type="text" name="nombre"
```

```
/>*</p>
```

Autor: Michael Serrato. Twitter: @mikebsg01

Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.

```

        <p class="input">Email: <input type="text" name="correo"
/>*</p>

        <p class="input">Comentario: <p>
        <textarea id="comentario" class="input"
name="comentario"></textarea>
        <input name="submit" value="Enviar" type="submit" />
        </form>
    </div>

    <div class="clear-50px">
    </div>

</div>

<aside>
    <div class="item-aside">
        <hgroup>
            <h3> Aside </h3>
        </hgroup>

        <p>
            Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vivamus
iaculis erat a ultrices pharetra. Nullam bibendum, felis a faucibus pellentesque, est sapien feugiat est, ac
bibendum justo massa ac enim. Maecenas lectus felis, elementum et arcu sit amet, bibendum
sollicitudin mauris. Interdum et malesuada fames ac ante ipsum primis in faucibus.
        </p>

    </div>
    <div class="item-aside">
        <hgroup>
            <h3> Aside </h3>
        </hgroup>

        <p>
            Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vivamus
iaculis erat a ultrices pharetra. Nullam bibendum, felis a faucibus pellentesque, est sapien feugiat est, ac
bibendum justo massa ac enim. Maecenas lectus felis, elementum et arcu sit amet, bibendum
sollicitudin mauris. Interdum et malesuada fames ac ante ipsum primis in faucibus.
        </p>

    </div>
    <div class="item-aside">
        <hgroup>
            <h3> Aside </h3>

```

Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.

</hgroup>

<p>

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vivamus iaculis erat a ultrices pharetra. Nullam bibendum, felis a faucibus pellentesque, est sapien feugiat est, ac bibendum justo massa ac enim. Maecenas lectus felis, elementum et arcu sit amet, bibendum sollicitudin mauris. Interdum et malesuada fames ac ante ipsum primis in faucibus.

</p>

</div>

</aside>

</section>

Código SASS:

```
.content {
  width: 62%;
  min-height: 256px;
  vertical-align: top;
  float: left;
  margin-left: 0;
  clear: both;
  display: block;

  article {

    width: 96%;
    padding: 1em 0;
    margin: auto;
    display: block;

    hgroup h1, h2, h3 {
      text-align: center;
      margin: .5em 0;
      padding: .25em 0;
      background-color: $gray_dark;
      @include border-radius(1.5em 0 1.5em 0);
      @include linear-gradient(top, $gray_alternative, $gray_dark, darken($gray_dark,
5%));
      color: $main-color;
    }
  }
}
```

Autor: Michael Serrato. Twitter: @mikebsg01

Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.

```
        font-style: italic;
        font-size: 1.5em;
    }

    .date {
        display: block;
        margin-right: auto;
        width: 52%;
        padding: 2px 8px;
        text-align: center;
        background-color: $gray_dark;
        @include border-radius(1.5em 0 1.5em 0);
        @include linear-gradient(top, $gray_alternative, $gray_dark, darken($gray_dark,
5%));

        color: $gray_light;
        font-size: .85em;
        font-style: italic;
    }

    img {
        width: 144px;
        max-width: 144px;
        height: 120px;
        max-height: 120px;
        display: block;
        border: 1px solid $gray_light;
        @include border-radius(2px 2px);
        padding: 5px 5px;
        float: right;
    }

    p {

        margin-top: 26px;
        margin-bottom: 10px;
        word-wrap: break-word;

    }

}
```


Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.

```
.comments-form {  
  
    width: 92%;  
    display: block;  
    margin: auto;  
    min-height: 120px;  
    border: 1px solid $gray_light;  
    @include border-radius(2px 2px);  
    padding: 1em 1em;  
  
    input[type="text"],  
    input[type="email"]{  
        width: 66%;  
        height: 1.75em;  
        display: inline-block;  
        *float: left;  
        *display: block;  
        @include border-radius(1.5em 0 1.5em 0);  
        border: 1px solid $gray_light;  
    }  
  
    textarea{  
        width: 96%;  
        height: 122px;  
        @include border-radius(1.5em 0 1.5em 0);  
        border: 1px solid $gray_light;  
    }  
  
    .input {  
        margin-bottom: 1.25em;  
    }  
  
    input[type="submit"]{  
        @include border-radius(1.5em 0 1.5em 0);  
        border: 1px solid $gray_light;  
        padding: 4px 22px;  
        font-size: 1em;  
        color: $main-color;  
        background-color: inherit;  
  
        &:hover{
```

Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.

```
        cursor: pointer;
        @include transition(all .8s);
        background-color: $white;
    }

}

}

}
```

Luego de agregar este código en nuestro archivo “style.scss” falta una cosa muy importante que debemos poner en la parte de arriba y es lo siguiente:

```
section, header, footer {
    width: $page_width;
    position: relative;
    display: block;
}
```

Básicamente este bloque de código lo que hará es agregar propiedades a los 3 elementos de la Estructura Semántica de HTML5 más similares en su diseño. Con un ancho del 100% y un Display en estilo de Bloque estos elementos lograrán darle mejor forma a nuestra maqueta.

DISEÑO DEL FOOTER

Y por último pero no menos importante el elemento **footer** que es el pie de página de nuestro documento de HTML5.

Código HTML:

```
<footer>
    <div class="copyright">
        <p> Copyright &copy; 2014 | Creado por: Michael Serrato - Correo electrónico:
        <a href="mailto:mikebsg01@gmail.com">mikebsg01@gmail.com</a> - Twitter: @mikebsg01 -
        Comunidades: <a href="http://www.html5facil.com/" title="El sitio web de la comunidad.">HTML5
        Facil</a> y <a href="http://www.ninjacode.tv" title="Stream en vivo todos los Miércoles!">Ninja
        Code Tv</a> </p>
    </div>
```

Autor: Michael Serrato. Twitter: @mikebsg01

Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.

```
</footer>
```

IMPORTANTE: Esta parte del código fuente deberá colocarse justo después de la etiqueta **Section**.

Código SASS:

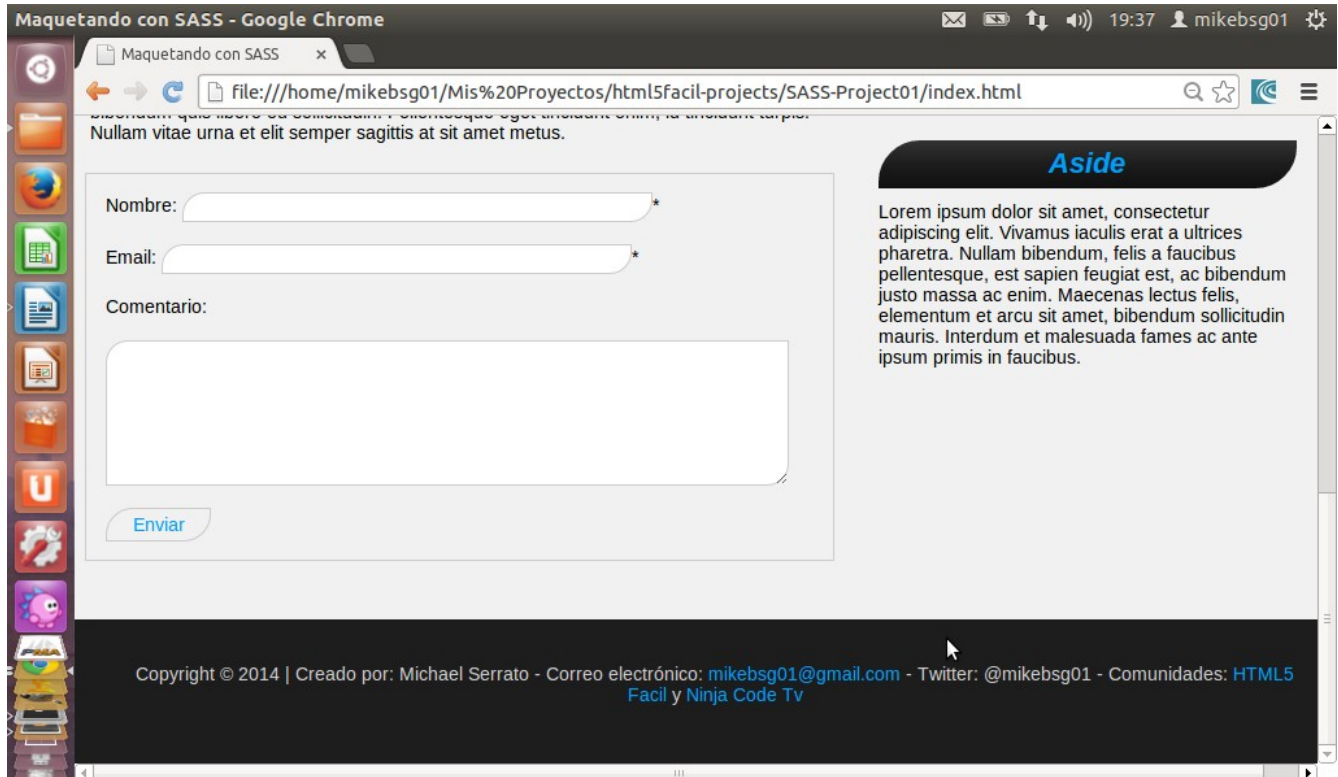
```
footer {  
  
    min-height: 92px;  
    padding: 1em 1em;  
    vertical-align: bottom;  
    display: block;  
    width: 100%;  
    position: relative;  
    clear: both;  
    background-color: $gray_dark;  
    color: $gray_light;  
    margin-bottom: 0;  
    bottom: 0;  
  
    .copyright{  
        width: 96%;  
        margin-top: 22px;  
        margin-left: auto;  
        margin-right: auto;  
        margin-bottom: 22px;  
        text-align: center;  
    }  
}
```

DISEÑO FINAL

Una vez concretados todos y cada uno de los elementos que conforman el diseño de este proyecto nos debería quedar algo así:

Autor: Michael Serrato. Twitter: @mikebsg01

Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.



Autor: Michael Serrato. Twitter: @mikebsg01

Guía de Maquetación con SASS y HTML5 - Por: Michael Serrato.

Si haz llegado hasta esta sección

**Felicitaciones haz concretado la Guía de Maquetación con SASS
y HTML5 !!! :D**

Autor: Michael Serrato

**El proyecto completo con todos los archivos puedes
descargarlos en el siguiente link:**

<https://github.com/mikebsg01/Maquetando-con-SASS-y-HTML5>

Autor: Michael Serrato. Twitter: @mikebsg01