

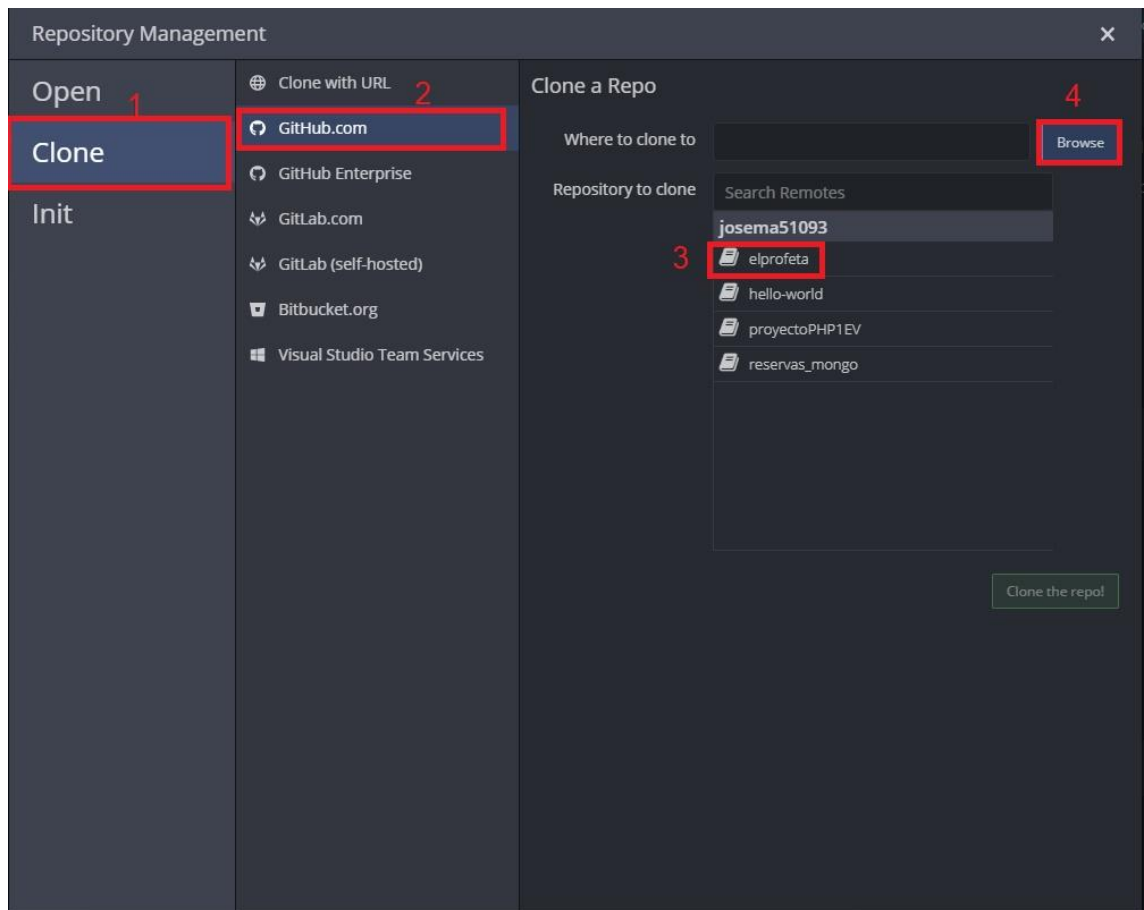
Manual para los DEMENTORES

1. Tareas asignadas a cada uno

- Yeison: migas de pan (breadcrumbs), paginadores, cards, noticias (buscar texto), una página de noticias
- Adrián: elementos sass (herencia, mixins, interpolación, variables...), estilos para tablas, títulos, formularios... y otra página de noticias
- Dani: aside, navbar, footer, elemento sass (herencia, mixins, interpolación, variables...)
- Cristian: sliders, galerías / carrusel, videos y fuentes

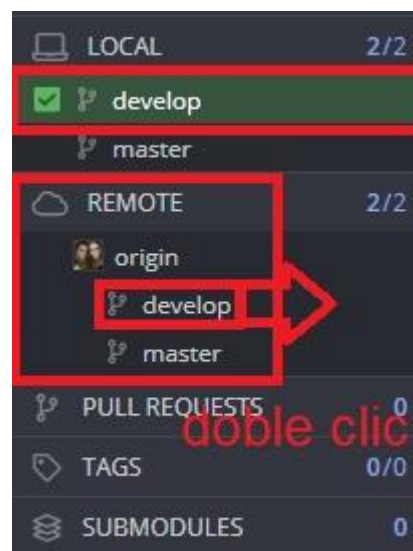
2. Comenzar bajando el proyecto a vuestro pc

Este paso supongo que lo tendréis más que mascado, pero lo pondré igualmente. Para empezar, abrís gitkraken (y conectáis con vuestra cuenta de github desde la aplicación, si no lo habéis hecho) y le dais a abrir proyecto (open a project). Os saldrá una carpeta como esta:



Como veis en la captura, os he marcado el proceso que tenéis que hacer. Simplemente le dais a “clone”, “github”, marcáis el proyecto “elprofeta” y por último indicáis donde lo vais a guardar.

Una vez se os sincronice, le dais a “open now” (saldrá arriba) y os cargará el proyecto. Seguramente os salga solo la rama master cargada, nosotros vamos a trabajar sobre la rama develop. Si os fijáis, a la izquierda aparece un apartado que pone “REMOTE”, y aparecerán 3 cosas: origin, develop y master. Pinchad sobre “develop” y de esta forma os carga en local la rama develop, para poder trabajar ahí:



Una vez hagáis eso, el proyecto se os habrá cargado correctamente en la rama develop y en la ruta que habéis indicado para ello. Cuando trabajéis, aseguraros de estar siempre en la rama develop, para evitar desastres.

3. Instalar Koala y configurarlo para compilar

No creo que sea necesario explicaros como se instala (como ya sabréis, se instala primero ruby y luego koala), así que paso directamente a como vincular el main.scss para que lo compile a main.css.

Para configurarlo, habrá que arrastrar la carpeta sass del proyecto a la interfaz del programa. Automáticamente el programa reconocerá los archivos que puede manipular. Os tenéis que fijar en que aparezca uno que ponga “main.scss”, y en teoría, debería salir ya debajo la ruta que tiene hasta el fichero css, como veréis en esta captura:



Si os aparece así, no tenéis que tocar nada, solo darle a botón derecho y “compilar o compile”. Si todo va bien, saldrá un mensaje que pondrá “success”, de ser así, ya tenéis el compilador funcionando y lo tendréis que tener abierto todo el rato que estéis trabajando.

En caso que no os aparezca la ruta anterior (la que enlaza al fichero css), debéis hacer lo siguiente: como os aparecerá el archivo main.scss sí o sí, pulsáis botón derecho sobre él y pincháis sobre “establecer la ruta de salida”. Se abrirá un cuadro para elegir la ruta, os vais a la carpeta css del proyecto y marcáis el fichero **main.css**, que se debe de encontrar ahí junto a los archivos de bootstrap y junto a otro main.map.

4. Estructura del proyecto

Tras tener todo configurado según visto lo anterior, procedo a explicaros como he estructurado el proyecto:

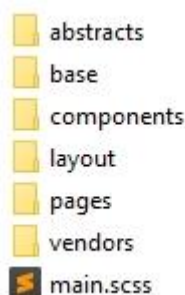


Como veis, la estructura os sonará bastante. Por ahora la carpeta “awesome” la ignoraremos.

- **Fonts:** irán las fuentes que Cristian elija a su gusto (te recomiendo que busques unas 3-4 más o menos, porque títulos y texto llevan fuentes distintas).
- **Images:** en esta carpeta se alojarán todas las imágenes y vídeos que vamos a ir utilizando para la web. Dentro ya se encuentran varios ejemplos de cómo va a ser el diseño, junto al logo que vamos a poner y un título auxiliar por si no conseguimos sacarlo con la fuente.
- **Js:** en esta carpeta se encuentran todos los archivos js que requiere bootstrap para funcionar, y si decimos de meter alguno más para controlar alguna función.
- **Pages:** aquí irán alojadas todas las páginas que tenga nuestra web (home, formulario, artículos...)
- **Sass:** aquí viene el culmen del proyecto y la carpeta tocha, la cual voy a explicaros de la mejor manera cómo la he estructurado.
- **Index.html:** no hace falta explicar que hace... aquí se encontrará seguramente el home, es decir, este index va a ser muy similar a alguna de las fotos de ejemplo que tenemos

5. Estructura y funcionamiento carpeta sass

Vamos al lío. Para comenzar, este es el contenido de la carpeta:

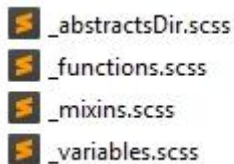


Por ahora no se ve complicación no? Pues voy a explicaros como funciona por dentro (supongo que el nombre de cada carpeta ya os va diciendo algo).

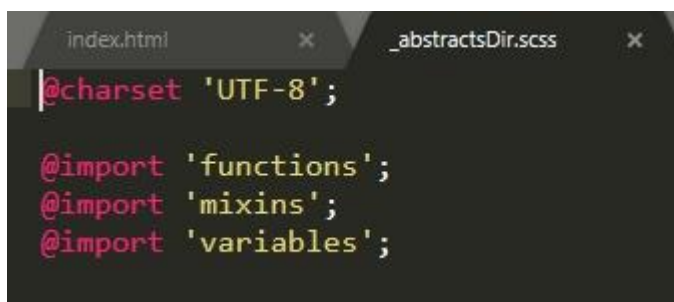
5.1. Carpeta abstracts

En esta carpeta se encuentran todos los partials referentes a los elementos de sass, tanto funciones, como mixins, como variables...

El contenido de esta carpeta es:



Vale, comenzando por el principio, tenemos el archivo _abstractsDir.scss. Este archivo lo único que tendrá en su interior será los @import de los otros tres archivos:

A screenshot of a code editor showing the content of the _abstractsDir.scss file. The code is as follows:

```
index.html x _abstractsDir.scss x
@charset 'UTF-8';

@import 'functions';
@import 'mixins';
@import 'variables';
```

NOTA: como sass reconoce los archivos partials (recordad, los que empiezan con _), no hace falta añadirles la barra baja ni la extensión. Luego veréis la importancia de este archivo en el main.scss.

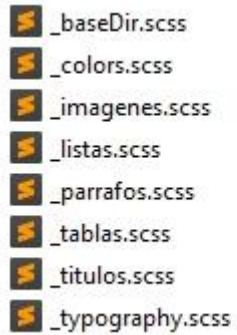
Los otros tres archivos (_functions.scss, _mixins.scss, _variables.scss) serán los que contendrán las variables que vamos a usar en todo el proyecto, así como los mixins y funciones que creemos.

En resumen, los únicos archivos que vamos a modificar y manipular, van a ser los que no lleven la palabra “Dir”. Ya he dejado preparado todo para que con guardar cualquier archivo se compile solo.

5.2. Carpeta base

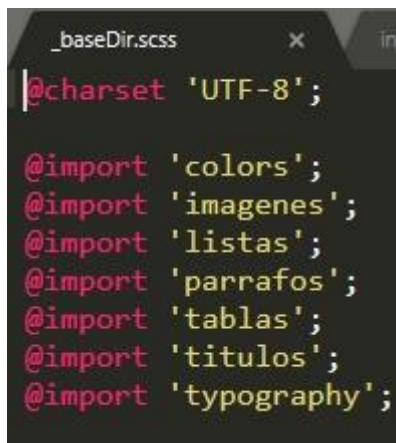
En esta carpeta se van a encontrar todos los partials referentes al estilo de cada componente que pongamos.

Su contenido es:



- _baseDir.scss
- _colors.scss
- _imagenes.scss
- _listas.scss
- _parrafos.scss
- _tablas.scss
- _titulos.scss
- _typography.scss

De nuevo (y así será en el resto de carpetas), el archivo “_baseDir.scss” será el que tenga el @import de todos los demás partials:



```
_baseDir.scss
@charset 'UTF-8';

@import 'colors';
@import 'imagenes';
@import 'listas';
@import 'parrafos';
@import 'tablas';
@import 'titulos';
@import 'typography';
```

Un ejemplo del partials “_colors.scss” sería:



```

_charset 'UTF-8';

//VARIABLES PARA LOS COLORES
//Color de fondo del container
$fondoContainer: black;

//Color de fondo header
$fondoHeader: pink;

//Color de fondo del main
$fondoMain: red;

//Color de fondo de .section1
$fondoSec1: yellow;









//Color de fondo de .section2
$fondoSec2: green;

```

5.3. Carpeta components

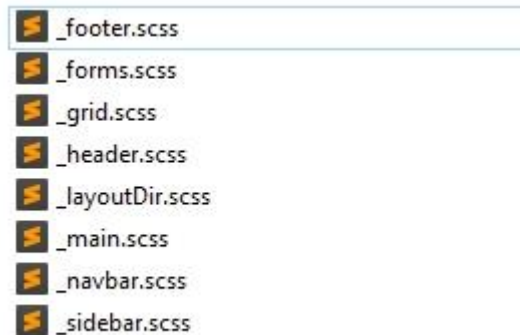
En esta carpeta se encuentran todos los partials referentes a los componentes a usar en el proyecto.

A partir de aquí creo que ha quedado bastante claro cómo va el funcionamiento de cada carpeta y cada archivo, así que pondré solo capturas del contenido de cada carpeta:

-  _aside.scss
-  _breadcrumbs.scss
-  _cards.scss
-  _carousel.scss
-  _componentsDir.scss
-  _paginadores.scss
-  _sliders.scss
-  _videos.scss

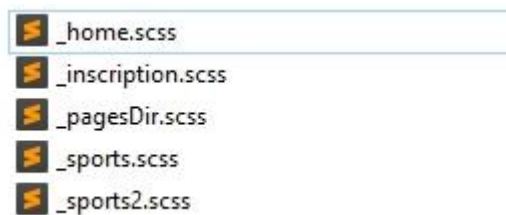
5.4. Carpeta layout

En esta carpeta se encuentran todos los partials referentes al diseño de toda la web.



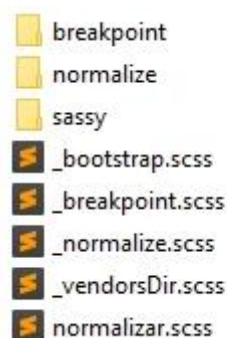
5.5. Carpeta pages

En esta carpeta se encuentran todos los partials referentes al diseño de cada página de artículos (seguramente aquí hagamos uso de herencia, ya que tendrán un aspecto muy similar a la página principal).



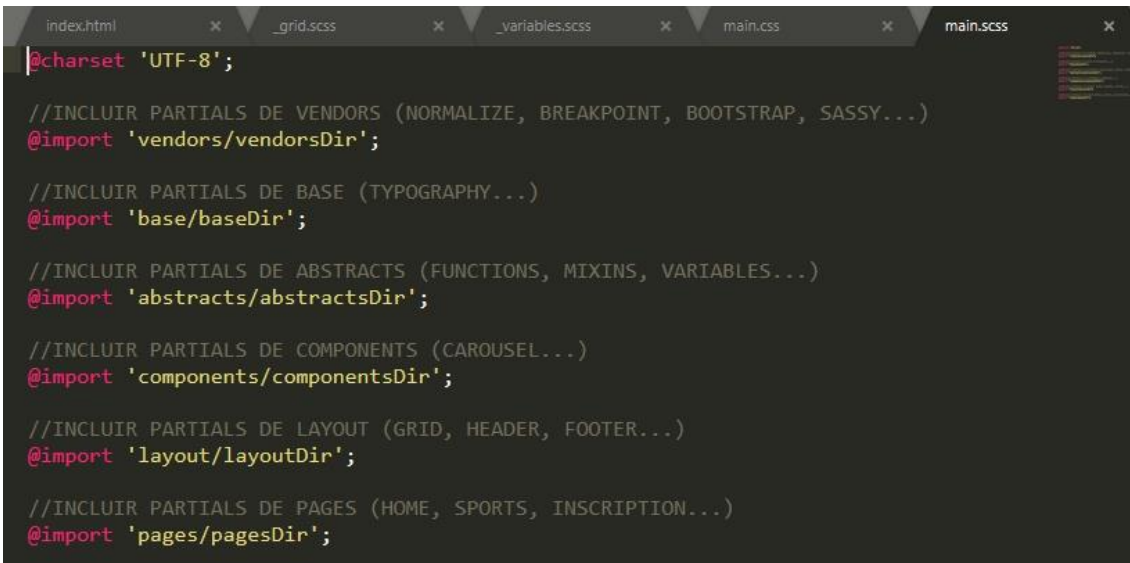
5.6. Carpeta vendors

En esta carpeta se van a encontrar todos los helpers externos, es decir, breakpoints (para hacer el diseño responsive), el normalize para normalizar la web, sassy para formularios y diversos archivos que nos enlazan esas carpetas. Esta carpeta seguramente no se tocará, ya que todo lo que hay ahí no necesita cambio alguno, de todas formas si veo que hay que meterles mano, os aviso.



5.7. Archivo main.scss

Por último y más importante, tenemos el archivo main.scss, cuya estructura es la siguiente:



```
@charset 'UTF-8';

//INCLUIR PARTIALS DE VENDORS (NORMALIZE, BREAKPOINT, BOOTSTRAP, SASSY...)
@import 'vendors/vendorsDir';

//INCLUIR PARTIALS DE BASE (TYPOGRAPHY...)
@import 'base/baseDir';

//INCLUIR PARTIALS DE ABSTRACTS (FUNCTIONS, MIXINS, VARIABLES...)
@import 'abstracts/abstractsDir';

//INCLUIR PARTIALS DE COMPONENTS (CAROUSEL...)
@import 'components/componentsDir';

//INCLUIR PARTIALS DE LAYOUT (GRID, HEADER, FOOTER...)
@import 'layout/layoutDir';

//INCLUIR PARTIALS DE PAGES (HOME, SPORTS, INSCRIPTION...)
@import 'pages/pagesDir';
```

Como veis, este archivo sirve para aunar o juntar todos los ficheros que os he mencionado antes, los cuales englobaban todos los partials de cada carpeta. De esta forma, se consigue tenerlo todo más ordenado y el código más limpio, que es de agradecer.

El funcionamiento de este archivo es sencillo, si os acordáis, antes en koala os dije que es el fichero al que había que indicarle una salida css. Pues bien, al estar vinculados todos los partials en este documento de una forma sencilla, cada vez que guardemos en el archivo que sea, se compilará automáticamente y nos generará el css completo con todo lo que hayamos ido haciendo en cada partial.

Hasta aquí el tutorial gente, espero que os haya servido y no haya sido muy pesado.

Vamos a darle duro que lo tenemos y lo vamos a sacar!!

¡¡No olvides suscribirte si no lo has hecho y dar un buen like para apoyar este tutorial, me harías muy feliz!!