

# TableGames con Ionic

Rodriguez Perez, Juan Manuel



## Contenidos

1. Introducción .....	2
2. Que necesitamos .....	3
3. Comandos a recordar .....	4
3.1 Comandos.....	4
4. Creando nuestra aplicación .....	7
5. Estructura del proyecto creado.....	8
6. Añadiendo plataformas (platforms).....	10
<b>6.1 Listado de plataformas</b> .....	10
<b>6.2 Añadir plataformas:</b> .....	11
7. Ejecutando y Debuggeando.....	12
8. Ciclo de construcción .....	16
9. Iniciando Desarrollo.....	17
<b>9.1 Generando Elementos</b> .....	17
<b>9.2 Adaptando Estilos</b> .....	17
<b>9.2.1 Distribución archivos scss en la App</b> .....	17
10. Bibliografía .....	1
11. Control de versiones .....	2

## 1. Introducción

Ejercicio para realizar en Ionic juegos de mesa como el tres en línea, conecta 4, etc.

## 2. Que necesitamos

Ante de empezar recordar que necesitamos tener instalado para crear nuestra aplicación.

- a. Tener instalado Node y NPM
- b. Tener instalado Cordova
- c. Tener instalado Ionic
- d. Tener instalado JDK

(<http://www.oracle.com/technetwork/pt/java/javase/downloads/jdk8-downloads-2133151.html> )

- e. Tener instalado SDKs a usar  
(<http://webostv.developer.lge.com/sdk/download/download-sdk/> ,  
<https://developer.android.com/studio/index.html> )

### 3. Comandos a recordar

Partiendo del punto 2 estos son algunos de los comandos a recordar.

- a. Instalar node (<https://nodejs.org/es/download/> )
  - b. Comprobar versión de Node: **node -v**
  - c. Instalar cordova: **npm install -g cordova**
  - d. Comprobar versión de cordova: **cordova --version**
  - e. Instalar Ionic, última versión: **npm install -g ionic** también **npm install -g cordova**
- ionic**
- f. Comprobar versión ionic: **ionic --versión**
  - g. Comprobar requerimientos: **ionic cordova requirements**
  - h. Al instalar ionic tendremos a nuestra disposición todos los comandos de ionic CLI, ejemplo obtenido de la web oficial (<https://ionicframework.com/docs/cli/>):

#### 3.1 Comandos

Comandos obtenidos de la web oficial de ionic:

<https://ionicframework.com/docs/cli/commands.html>

Command	Description
<a href="#">build</a>	Build web assets and prepare your app for any platform targets
<a href="#">docs</a>	Open the Ionic documentation website
<a href="#">generate</a>	Generate pipes, components, pages, directives, providers, and tabs (ionic-angular >= 3.0.0)
<a href="#">info</a>	Print system/environment info
<a href="#">link</a>	Connect your local app to Ionic
<a href="#">login</a>	Login with your Ionic ID
<a href="#">serve</a>	Start a local dev server for app dev/testing
<a href="#">signup</a>	Create an Ionic account
<a href="#">start</a>	Create a new project
<a href="#">telemetry</a>	(deprecated) Opt in and out of telemetry
<a href="#">upload</a>	(deprecated) Upload a new snapshot of your app
<a href="#">config get</a>	Print config values

Command	Description
<a href="#">config set</a>	Set config values
<a href="#">cordova build</a>	Build (prepare + compile) an Ionic project for a given platform
<a href="#">cordova compile</a>	Compile native platform code
<a href="#">cordova emulate</a>	Emulate an Ionic project on a simulator or emulator
<a href="#">cordova platform</a>	Manage Cordova platform targets
<a href="#">cordova plugin</a>	Manage Cordova plugins
<a href="#">cordova prepare</a>	Copies assets to Cordova platforms, preparing them for native builds
<a href="#">cordova requirements</a>	Checks and print out all the requirements for platforms
<a href="#">cordova resources</a>	Automatically create icon and splash screen resources
<a href="#">cordova run</a>	Run an Ionic project on a connected device
<a href="#">doctor check</a>	Check the health of your Ionic project
<a href="#">doctor ignore</a>	Ignore a particular issue
<a href="#">doctor list</a>	List all issue identifiers
<a href="#">git remote</a>	Adds/updates the Ionic git remote to your local Ionic app repository
<a href="#">integrations disable</a>	Disable an integration
<a href="#">integrations enable</a>	Add various integrations to your app
<a href="#">monitoring syncmaps</a>	Sync Source Maps to Ionic Pro Error Monitoring service
<a href="#">package build</a>	(deprecated) Start a package build
<a href="#">package download</a>	(deprecated) Download your packaged app
<a href="#">package info</a>	(deprecated) Get info about a build
<a href="#">package list</a>	(deprecated) List your cloud builds
<a href="#">ssh add</a>	Add an SSH public key to Ionic

Command	Description
<a href="#">ssh delete</a>	Delete an SSH public key from Ionic
<a href="#">ssh generate</a>	Generates a private and public SSH key pair
<a href="#">ssh list</a>	List your SSH public keys on Ionic
<a href="#">ssh setup</a>	Setup your Ionic SSH keys automatically
<a href="#">ssh use</a>	Set your active Ionic SSH key

## 4. Creando nuestra aplicación

Una vez revisados los anteriores puntos, crearemos nuestra aplicación; para ello usaremos una terminal de comandos y nuestro IDE favorito.

En mi caso usare VSCode que tiene la opción de incluir la terminal en el propio IDE.

Nos situaremos en la carpeta raíz donde queramos crear nuestro proyecto y una vez que sepamos el nombre de nuestro usaremos el siguiente comando:

- **ionic start TableGames**

Este comando te mostrara las siguientes opciones:

```
F:\Data\PROYECTOS\Ionic>ionic start TableGames

? What starter would you like to use: (Use arrow keys)
> tabs ..... ionic-angular A starting project with a simple tabbed interface
blank ..... ionic-angular A blank starter project
sidemenu ..... ionic-angular A starting project with a side menu with navigation in the content area
super ..... ionic-angular A starting project complete with pre-built pages, providers and best practices for Ionic
development.
conference ..... ionic-angular A project that demonstrates a realworld application
tutorial ..... ionic-angular A tutorial based project that goes along with the Ionic documentation
aws ..... ionic-angular AWS Mobile Hub Starter
```

- Yo he elegido la opción **sidemenu**

Una vez creado el proyecto te preguntara si quieres incluir cordova para usar Android y iOS:

```
✓ Creating directory .\TableGames - done!
✓ Downloading and extracting sidemenu starter - done!

? Would you like to integrate your new app with Cordova to target native iOS and Android? (y/N) |
```

- En este caso le he dicho que si ('y')

Una vez que termine la instalación te pregunta si quieres instalar la opción pro (tienen diferentes versiones entre la free hasta la Business).

```
Learn more about Ionic Pro: https://ionicframework.com/products

? Install the free Ionic Pro SDK and connect your app? (Y/n) |
```

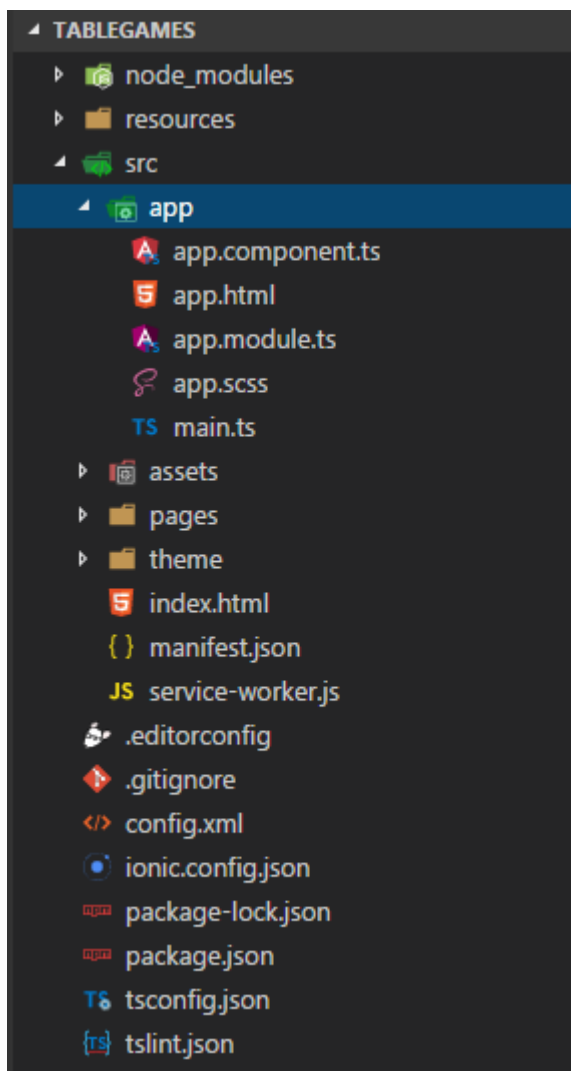
- En este caso le he dicho que no ('n')

Y con esto terminaría la creación del proyecto, ahora solo nos tenemos que situar dentro de la carpeta del proyecto para comenzar a trabajar y usar los comandos de ionic CLI.



## 5. Estructura del proyecto creado

A continuación os pongo una imagen de cómo quedaría la estructura del proyecto una vez creado.



Sin entrar a detalle de la estructura creada (se basa en Angular):

- **Node\_modules:** Carpeta que contiene todas las dependencias que va usar ionic en el proyecto entre ellas las de angular.
- **Resources:** Carpeta con los recursos que usaran nuestros diferentes targets construidas en el proyecto (Android, iOS).
- **Src:** Carpeta que contendrá todo nuestro código fuente desarrollado y será el que se transforme a JavaScript cuando sea compilado.
- **App:** Carpeta interna de src, que contendrá los elementos de la aplicación.

- **Assets:** Carpeta donde podremos alojar imágenes, iconos, etc.
- **Pages:** Carpeta con diferentes páginas o pantallas de la aplicación
- **Theme:** Carpeta del tema de la aplicación
- **Index.html:** Entrada a la aplicación contiene el root de ionic `<ion-app></ion-app>`

## 6. Añadiendo plataformas (platforms)

Según hemos creado la aplicación aún no está preparada para ser distribuida a ninguna plataforma como Android o iOS para ello debemos añadir las “platforms” que queramos distribuir.

### 6.1 Listado de plataformas

Estas son las plataformas descritas en la web oficial (<https://ionicframework.com/docs/api/platform/Platform/> )

Platform Name	Description
android	on a device running Android.
cordova	on a device running Cordova.
core	on a desktop device.
ios	on a device running iOS.
ipad	on an iPad device.
iphone	on an iPhone device.
mobile	on a mobile device.
mobileweb	in a browser on a mobile device.
phablet	on a phablet device.
tablet	on a tablet device.
windows	on a device running Windows.

## 6.2 Añadir plataformas:

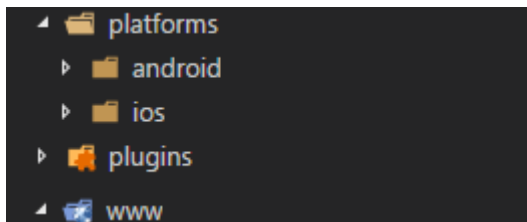
Para añadir plataformas desde la consola de comandos o terminal usaremos:

- **ionic cordova platform add <plataforma>**

Ejemplos:

- **ionic cordova platform add android**
- **ionic cordova platform add ios**

Una vez ejecutados veremos en el proyecto se han creado las siguientes carpetas:



- **Platforms:** Carpeta que contendrá una carpeta por cada plataforma añadida
- **Plugins:** Carpeta donde se añaden plugins de cordova
- **www:** Carpeta que contendrá los recursos web y almacenará nuestro proyecto construido.

## 7. Ejecutando y Debuggeando

Antes de comenzar a desarrollar podemos tener en cuenta las siguientes anotaciones que iremos usando para crear nuestra aplicación.

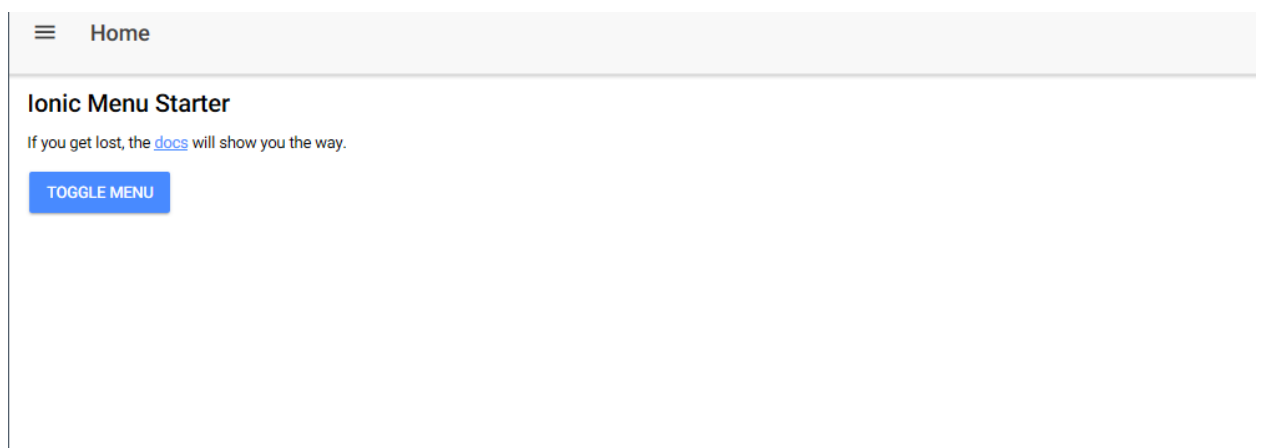
### 7.1 Debuggeando en el navegador

Ionic nos permite debuggear nuestra aplicación directamente en el navegador incluso aunque no tengamos plataformas instaladas.

Para ello usaremos el comando:

- **ionic serve**

Con lo que hemos creado hasta este punto al ejecutar este comando veríamos la aplicación de la siguiente manera:



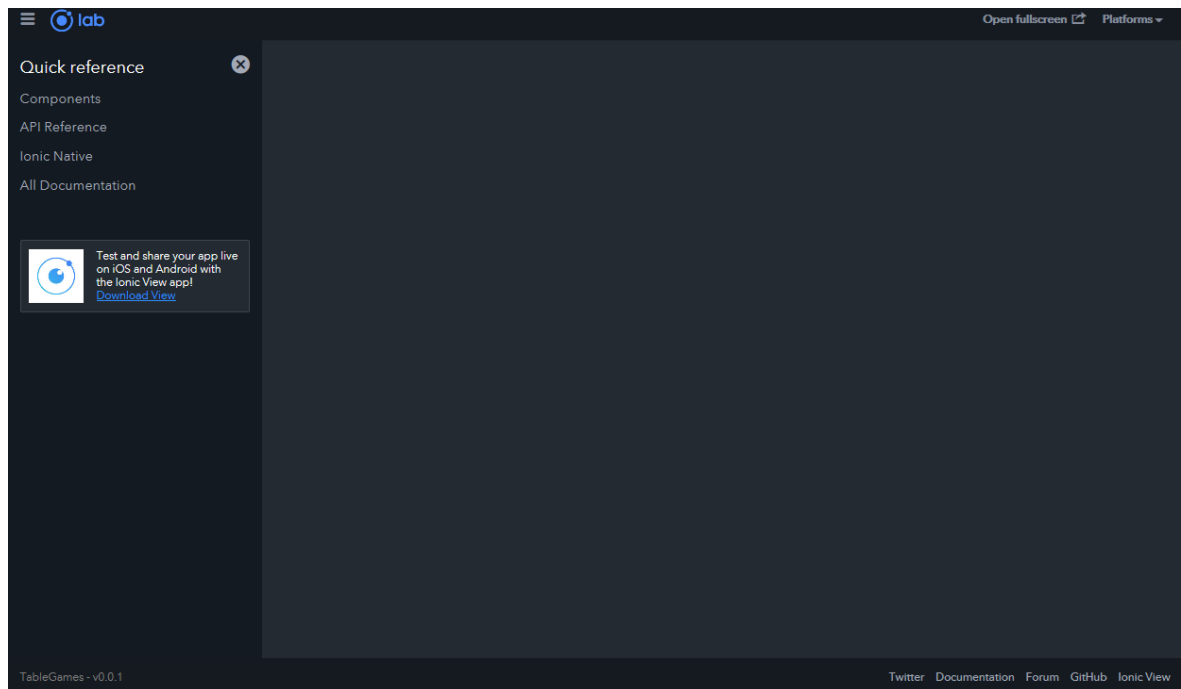
## 7.2 Ionic Lab

Es una funcionalidad muy interesante que nos permite depurar la aplicación en las plataformas Android, iOS y Windows directamente en el navegador.

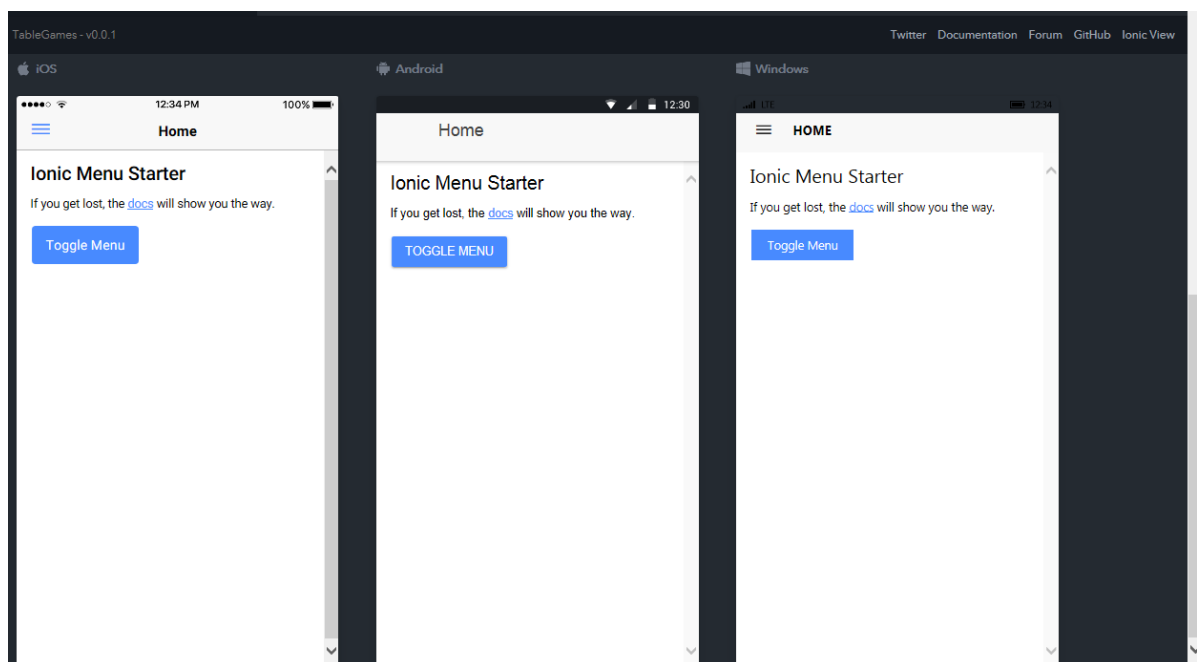
Para ello usaremos el comando:

- **ionic serve --lab**

Lo veríamos así en el navegador:



Haciendo scroll nos encontramos:



### 7.3 Ejecutar en terminales

Además de ejecutar la aplicación en el navegador podemos ejecutarla en diferentes terminales conectados o AVDs Android mediante el comando:

Para terminales reales Android, se deberá tener la opción de “desarrollador” activa junto la “depuración USB”.

- **ionic cordova run <plataforma>** (se deberá tener en cuenta que tengamos las SDK y configuradas. Si no se ha instalado Android Studio y se ha instalado el SDK manualmente se deberá instalar Gradle - <https://gradle.org/install/#manually> )

Ejemplos:

**ionic cordova run android**

**ionic cordova run ios**

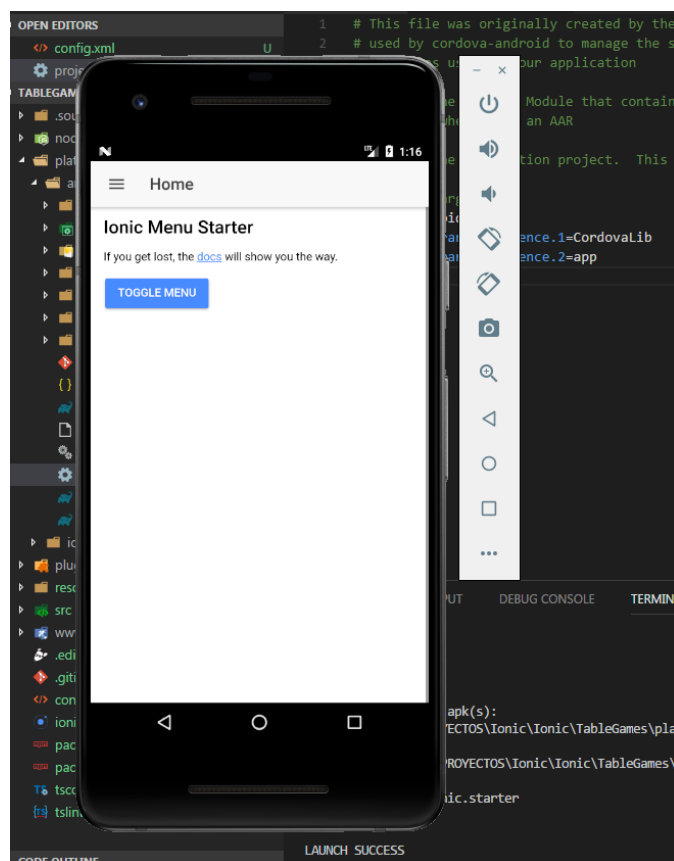
**ionic cordova run android --livereload**

- Para ver dispositivos disponibles:

**cordova run android --list --device**

- Dentro del proyecto en el fichero **package.json** tenemos las siguientes líneas donde podemos cambiar la versión del S.O. y que será la usada cuando generemos las platforms:

```
"cordova-android": "7.0.0",  
"cordova-ios": "4.5.4",
```



Posibles errores:

ERROR: TypeError: Cannot read property 'semver' of null

An unexpected error has occurred while running list-emulator-images with code 2: Error: cmd: Command failed with exit code 2

O podéis usar el comando (o desde el AVD de AndroidStudio dejar solo el que os interese):  
**ionic cordova run Android --livereload --target <emulator\_name>**

#### 7.4 Construir la aplicación

Para construir nuestra aplicación (recordar que hay que tenerla añadida) usaremos el siguiente comando:

- **ionic cordova build <plataforma>** (se deberá tener en cuenta que tengamos las SDK y configuradas.)

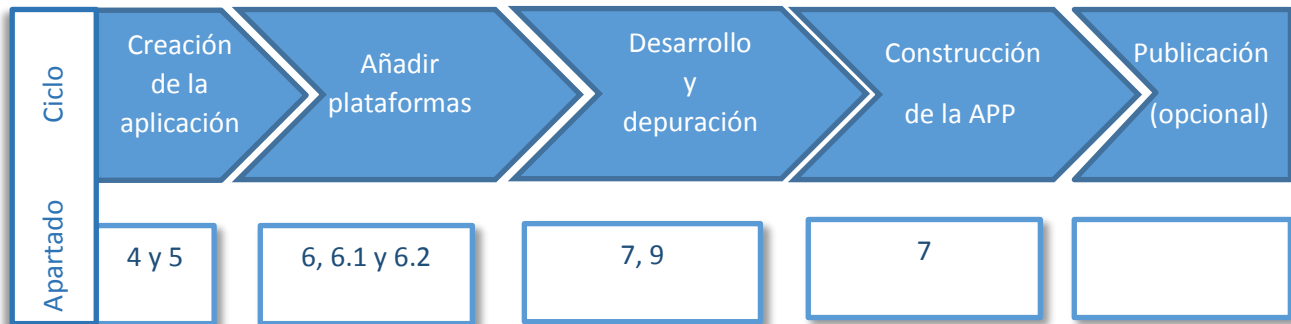
Ejemplos:

- **ionic cordova build Android**



## 8. Ciclo de construcción

Podríamos resumir el ciclo de construcción visto hasta ahora en el siguiente diagrama:



## 9. Iniciando Desarrollo

En este apartado iremos viendo diferentes temas según avancemos en el desarrollo de la aplicación.

### 9.1 Generando Elementos

En Ionic podemos generar elementos de forma parecida a Angular cuando se usa el comando 'ng' en este caso usaremos el comando:

- **ionic generate <generator> <name>**: Donde name será el nombre que le queremos dar a nuestro elemento y como generator serán los distintos elementos disponibles de ionic (component, directive, pipe, page, provider, tabs).

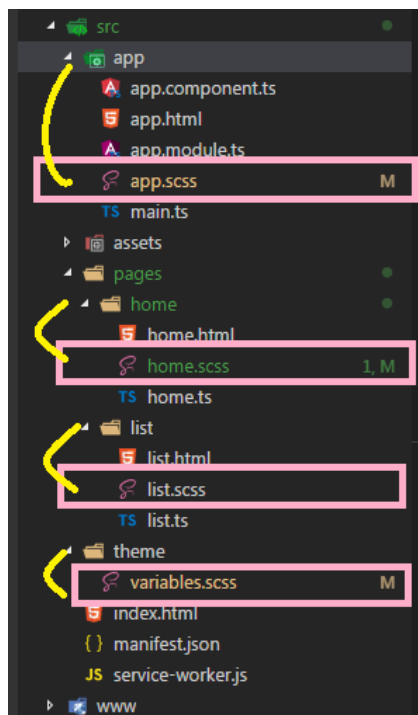
### 9.2 Adaptando Estilos

En Ionic por defecto nos instala el uso de SASS (Syntactically Awesome StyleSheets) pero no pasa nada si no lo conoces ya que es totalmente compatible con CSS.

¿Por qué empiezo con los estilos?, no es por nada en particular; simplemente así podemos ver los archivos creado por defecto por ionic al escoger nos cree la barra de menú y por ver la App más chula desde el principio.

#### 9.2.1 Distribución archivos scss en la App

Si lo comparamos con Angular que nos creaba el fichero **styles.css** a la altura del fichero **index** aquí vemos como nos crea una carpeta **'theme'** donde tenemos el fichero **variables.scss** que será para albergar como su nombre indica las variables que declaremos de scss.



Dentro de la carpeta **'app'** tenemos el fichero **app.scss** que albergara los estilos generales luego a diferencia de Angular que creábamos nuestros componentes donde queríamos en ionic nos crea una carpeta llamada **'pages'** donde iremos creando las diferentes pantallas (components) de nuestra aplicación.

Por cada pantalla creada tendremos un fichero scss del mismo nombre que la pantalla y que albergara los estilos propios de esa pantalla.

En cada fichero particular de cada página podemos ver cómo nos crea un elemento vacío por defecto, en el caso de la página home:

```
1 page-home {  
2  
3 }
```

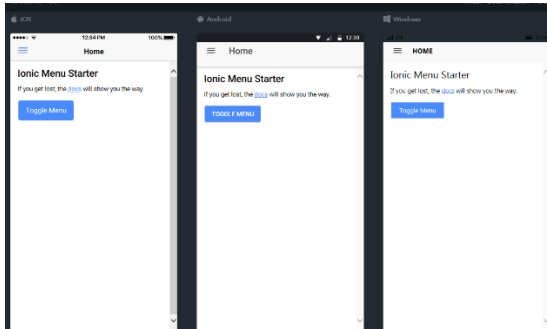
Este coincide con el nombre asignado al componente como **'selector'** en la clase **ts**.

Además de customizar los estilos también hay disponible en el **market** de Ionic temas completos:

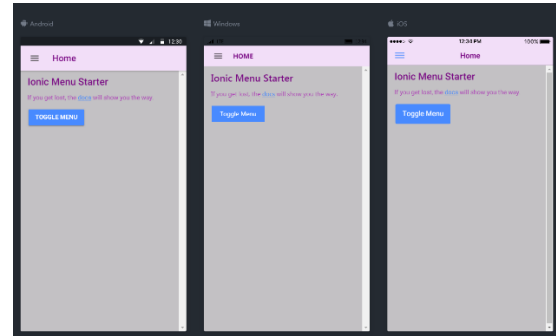
<https://market.ionicframework.com/themes>

Mis cambios:

Antes



Después



Para hacer estos pequeños cambios he modificado los siguientes ficheros:

- variables.scss, donde simplemente he añadido más colores:

```
$colors: (
  primary:    #488aff,
  secondary:  #32db64,
  danger:     #f53d3d,
  light:      #f4f4f4,
  dark:       #212121,
  violet:     #ae41b8,
  darkViolet: #780483,
  backgroundColor: #c3c1c4,
  toolbarColor: #f2def8
);
```

- app.scss, he modificado el general de la aplicación ya que los cambios han sido genéricos para toda la aplicación en este caso he añadido:

```
h1,h2,h3,h4,h5,h6{
  color: color($colors, darkViolet, base);
}

p{
  color: color($colors, violet, base);
}

//Clases
.toolbar-background{
  //también se puede acceder con --> map-get($map, $color-name); => background-color: map-get($colors, backgroundColor);
  background-color: color($colors, toolbarColor, base);
}

.toolbar-title{
  color: color($colors, darkViolet, base);
}

.content{
  background-color: color($colors, backgroundColor, base);
}
```

## 10. Bibliografía

Enlaces interesantes a tener en cuenta o donde se ha obtenido información.

- Ionic: <https://ionicframework.com/>
- Node: <https://nodejs.org/es/>
- Typescript: <https://www.typescriptlang.org/>
- VSCode: <https://code.visualstudio.com/>
- Sublime Text: <https://www.sublimetext.com/3>
- Atom: <https://atom.io/>
- AndroidStudio: <https://developer.android.com/studio/index.html?hl=es-419>
- Java JDK: <http://www.oracle.com/technetwork/pt/java/javase/downloads/index.html>
- iOS Develop: <https://developer.apple.com/develop/>
- Angular: <https://angular.io/>
- NativeScript: <https://www.nativescript.org/>
- Sass: <https://sass-lang.com/>

## 11. Control de versiones

Versión	Descripción	Fecha
1.0	Versión Inicial	23/02/2018
1.1	Cambiando Estilos	26/02/2018