**manual para realizar el deploy de la app**

**PREPARANDO EL ENTORNO.**

Para poder subir una página web al hosting de la nube de Firebase, primero tenemos que instalar las herramientas de Firebase (firebase tools), una utilidad que instalas para la terminal, a través de la cual puedes ejecutar comandos para administrar tus proyectos de Firebase, entre otras cosas, para subir una carpeta con un sitio web al hosting.

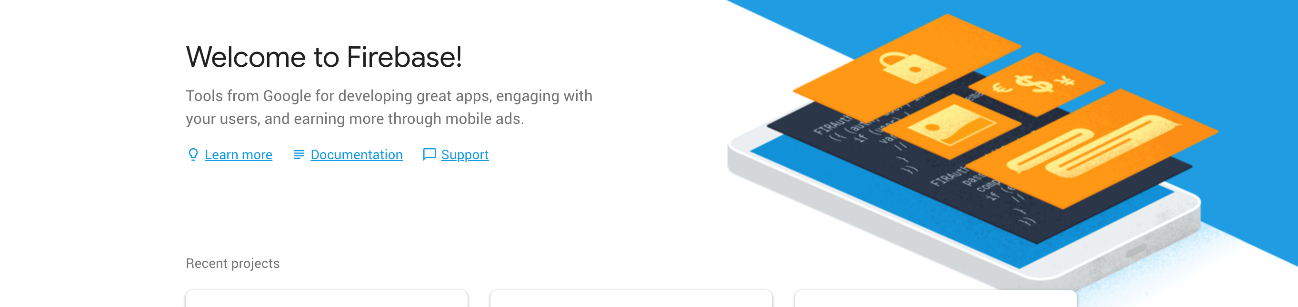
Para instalar las Firebase Tools, necesitas primero tener instalado NodeJS y NPM, una vez que hayas instalado estas dependencias, debes ejecutar el siguiente comando dentro de el CMD o la Terminal, dependiendo de si estás en Windows, MacOS o Linux:

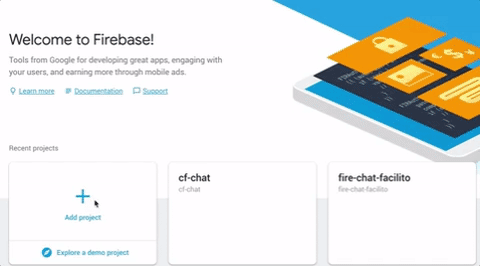
npm install -g firebase-tools

Este comando instalará el paquete, que a su vez expondrá un comando firebase en la terminal, que como decía, te permite administrar tus proyectos.

**FIREBASE CONSOLE.**

Para poder administrar tu cuenta Firebase, primero necesitas haber creado una, ingresa a la dirección [http://console.firebase.google.com](http://console.firebase.google.com/) para poder iniciar una cuenta Firebase, para hacerlo necesitas tener una cuenta de Google (como un correo de Gmail), al ingresar se creará una cuenta de Firebase para tu cuenta activa de Google, si no habías iniciado sesión en este navegador con una cuenta de Google, se te pedirá que ingreses una.

Una vez en la pantalla que aparece como imagen justo arriba de este texto, vamos a seleccionar la opción **Add project** para crear un nuevo proyecto. Aquí se te pedirá que coloques el nombre del proyecto con el que lo identificarás, el nombre es deliberado y puedes colocar el que mejor describa tu proyecto.



**LOGIN EN FIREBASE TOOLS**

Ya que tienes una cuenta creada y un proyecto configurado, necesitas enlazar tu cuenta de Firebase con las firebase-tools que previamente instalaste. Para eso debes ejecutar el siguiente comando en la terminal:

firebase login

Inicialmente se te preguntará si quieres que tu información de uso de la herramienta sea compartida con Firebase o no, si decidas compartir tus datos escribe Y, si no, escribe n, luego presiona Enter.

Este comando deberá automáticamente abrir en tu navegador web una página para que selecciones la cuenta con la que enlazarás tu herramienta de la terminal, adicionalmente puedes copiar la URL que aparece en negritas en la terminal, en la barra de direcciones de tu navegador para ver la misma página.

Ahí, selecciona la cuenta con la que quieres enlazar la terminal, debe ser la misma que creaste en el paso anterior, para usar el proyecto que creamos. Una vez que autorices los permisos para la app de Firebase, automáticamente la terminal detectará que ya autorizaste y enlazaste tu aplicación mostrando un mensaje como el siguiente:

✔ Success! Logged in as urielcodigofacilito.com

**SUBIR TU PÁGINA A FIREBASE HOSTING**

Para subir tu proyecto al hosting cloud de Firebase, debes primero posicionar la terminal en la carpeta que contiene el proyecto. Como este ejemplo está orientado a una aplicación de angular, el primer paso para la subida será genera el *build* de Angular.

**ANGULAR BUILD**

El buid es una versión optimizada de nuestra aplicación, en la que el código fue comprimido, reducido y optimizado para que cargue más rápido, sea de menor tamaño y en general mejore su rendimiento. Para generar el build coloca el siguiente comando en la Terminal o el CMD:

ng build

El resultado de este comando debe verse como a continuación te muestro:

Date: 2018-07-15T06:18:49.945Z

Hash: 8ee114542ed8a659f7e1

Time: 7756ms

chunk {main} main.js, main.js.map (main) 55 kB [initial] [rendered]

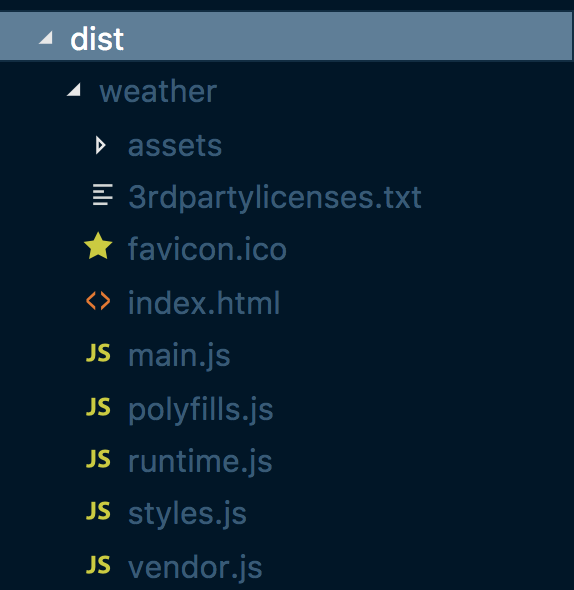
chunk {polyfills} polyfills.js, polyfills.js.map (polyfills) 227 kB [initial] [rendered]

chunk {runtime} runtime.js, runtime.js.map (runtime) 5.22 kB [entry] [rendered]

chunk {styles} styles.js, styles.js.map (styles) 16.1 kB [initial] [rendered]

chunk {vendor} vendor.js, vendor.js.map (vendor) 3.11 MB [initial] [rendered]

Esto quiere decir que el build de tu aplicación fue generado con éxito, para comprobarlo debes verificar que se añadió una carpeta dist en tu proyecto de Angular:



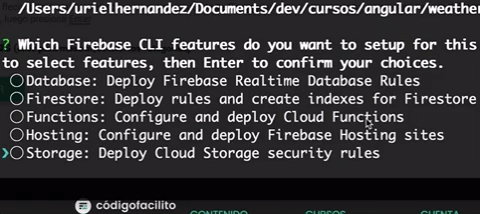
**SUBIR EL BUILD A FIREBASE**

Para comenzar con la subida ejecuta el comando init de Firebase Tools para que la herramienta configure el proyecto como un proyecto de Angular, y además, enlaza el proyecto que previamente creamos en la consola de Firebase con la carpeta actual, para eso coloca el siguiente comando:

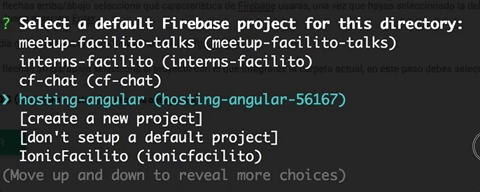
firebase init

Este comando iniciará un asistente que te guiará por la configuración del proyecto, las opciones que debes seleccionar son:

1. Usando las flechas arriba/abajo selecciona qué característica de Firebase usarás, una vez que hayas seleccionado la del Hosting, presiona la barra espaciadora para activar el uso de esta característica, luego presiona Enter



1. Usando las flechas arriba/abajo selecciona el proyecto con el que integrarás la carpeta actual, en este paso debes seleccionar el proyecto que creamos con anterioridad en este mismo artículo, presiona Enter cuando hayas seleccionado el proyecto adecuado.



1. El siguiente punto es muy importante cuando trabajamos con un proyecto de Angular, en este paso el asistente de Firebase nos pregunta qué directorio contiene nuestra aplicación, es muy importante que indiquemos que es el directorio **dist/nombre-proyecto** sustituye la parte de nombre-proyecto por la carpeta que veas dentro de dist

? What do you want to use as your public directory? (public) dist/nombre-proyecto

1. El siguiente paso, está relacionado con las Single Page Applications, como Angular suele contener su propio router (si no lo contiene, no importa no es necesario), escribiremos y y luego presionaremos Enter. Luego de eso, Firebase notará que ya existe un archivo index.html y te preguntará si quieres que lo sobre escriba, o no, escribe ny luego Enter para indicarle que no debe sobre escribirlo y se conserve el archivo index.html de tu proyecto Angular.

Esto debe completar el asistente de Firebase y habremos configurado con éxito nuestro directorio como una app de Angular que podemos subir al Firebase Hosting. Para terminar, vamos a ejecutar el siguiente comando que se encargará de subir nuestros archivos a Firebase para que podamos visualizar nuestra página:

firebase deploy

Si todo sale bien, deberás ver un mensaje como el siguiente:

=== Deploying to 'hosting-angular-56167'...

i deploying hosting

i hosting: preparing dist directory for upload...

✔ hosting: 43 files uploaded successfully

✔ Deploy complete!

Project Console: https://console.firebase.google.com/project/hosting-angular-56167/overview

Hosting URL: https://hosting-angular-56167.firebaseapp.com

Como parte del mensaje podrás notar que se imprime la URL en la que podrás ubicar tu sitio web, en mi caso es [https://hosting-angular-56167.firebaseapp.com](https://hosting-angular-56167.firebaseapp.com/) si ingresas ahí, deberías ver tu página web funcionando y en línea

**FUTUROS CAMBIOS.**

Lo interesante de Firebase Hosting es que con una configuración muy simple, ahora tienes un flujo de desarrollo en el que cada modificación a tu aplicación puede ser publicada con un solo comando en la terminal.

Si hablamos de un proyecto de Angular, una vez que quieras subir cambios a tu página web, debes ejecutar los siguientes comandos:

ng build

firebase deploy