



# Tema 7. Desarrollo de aplicaciones móviles híbridas con Ionic

Programación web

Boni García  
Curso 2017/2018

# Índice

---

1. Introducción
2. Ionic
3. Ionic CLI
4. Componentes
5. Ionic View

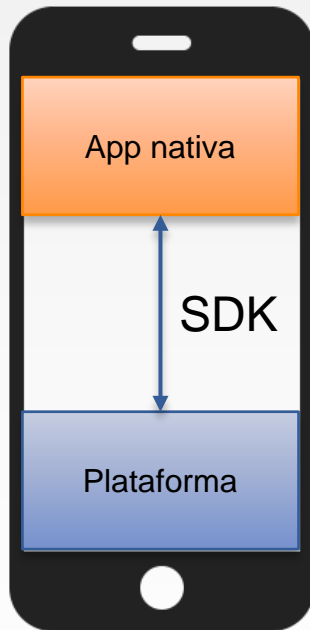
# Índice

---

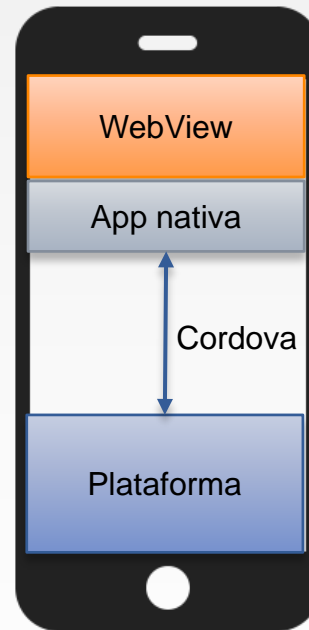
1. Introducción
  - Tipos de aplicaciones móviles
  - Aplicaciones móvil híbridas
  - Frameworks de desarrollo de aplicaciones móvil híbridas
2. Ionic
3. Ionic CLI
4. Componentes
5. Ionic View

# Introducción

## Tipos de aplicaciones móviles



Aplicación móvil nativa



Aplicación móvil híbrida



Aplicación móvil progresiva  
(PWA, *Progressive Web Apps*)

# Introducción

---

## Aplicaciones móvil híbridas

- Las aplicaciones móviles **híbridas** permiten el uso de tecnologías web de lado cliente (HTML, CSS y JavaScript) para el diseño de interfaces de usuario
- Reducen el esfuerzo de desarrollo, ya que la misma aplicación se empaqueta para diferentes plataformas

# Introducción

---

## Frameworks de desarrollo de aplicaciones móvil híbridas

- En la actualidad existen muchos frameworks para la creación de apps híbridas:



<http://ionicframework.com/>



<https://onsen.io/>



<http://mobileangularui.com/>



<https://software.intel.com/en-us/intel-xdk>



<https://www.meteor.com/>

# Índice

---

1. Introducción
2. **Ionic**
3. Ionic CLI
4. Componentes
5. Ionic View

# Ionic

---

- Ionic es un framework para el desarrollo de aplicaciones web híbridas
- Ionic se basa en frameworks existentes:
  - **Cordova**: para empaquetar la aplicación web como aplicación nativa
  - **Angular**: para implementar la lógica de la aplicación
- Ionic proporciona un conjunto de componentes y estilos que permiten implementar aplicaciones híbridas con una apariencia similar a las aplicaciones nativas



ionic

<http://ionicframework.com/>



# Ionic

---

- Ionic utiliza **Sass** (*Syntactically Awesome Stylesheets*), que es un metalenguaje de CSS que enriquece la forma de dar estilos
- Las versiones de Ionic son
  - **Ionic 1**
    - Basado en Angular 1
  - **Ionic 2**
    - Basado en Angular 2
  - **Ionic 3**
    - Basado en Angular 4
    - Los cambios de Ionic 2 a Ionic 3 son mínimos (equivalente al paso de Angular 2 a Angular 4)

# Índice

---

1. Introducción
2. Ionic
3. **Ionic CLI**
4. Componentes
5. Ionic View

## Ionic CLI

---

- Ionic CLI (*command line interface*) es una herramienta que facilita el desarrollo de aplicaciones Ionic
- Ha sido desarrollado en Node.js, con lo que la instalamos en nuestro sistema a través de NPM

```
> npm install -g ionic cordova
+ cordova@8.0.0
+ ionic@3.20.0
added 233 packages from 175
contributors, removed 967 packages,
updated 248 packages and moved 14
packages in 127.339s

> ionic -v
3.20.0
```

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

# Ionic CLI

Fork me on GitHub

```
> ionic start ionic-hello-world

? What starter would you like to use:
  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

? Would you like to integrate your new app with Cordova to target native iOS and Android? Yes

? Install the free Ionic Pro SDK and connect your app? Yes

-----

> npm i --save -E @ionic/pro
✓ Running command - done!
> ionic link
✓ Looking up your apps - done!
```

Para crear una aplicación Ionic mediante Ionic CLI usamos el comando `ionic start`

Para usar Ionic Pro necesitamos crear una cuenta en <https://apps.ionic.io>

# Ionic CLI

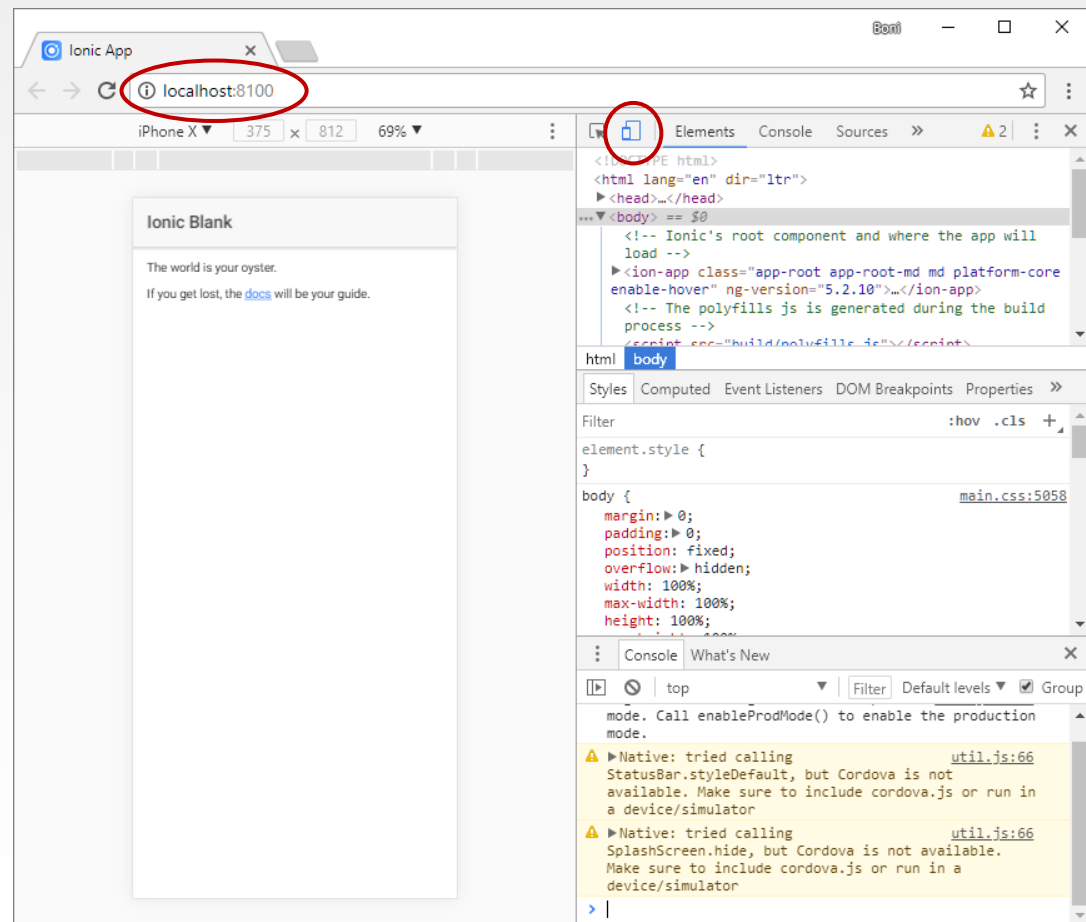
## > ionic serve

```
Starting app-scripts server: --address 0.0.0.0 --port 8100 --livereload-port 35729 --dev-logger-port 53703 --nobrowser -  
Ctrl+C to cancel  
[23:56:51] watch started ...  
[23:56:51] build dev started ...  
[23:56:51] clean started ...  
[23:56:51] clean finished in 2 ms  
[23:56:51] copy started ...  
[23:56:51] deeplinks started ...  
[23:56:51] deeplinks finished in 12 ms  
[23:56:51] transpile started ...  
[23:56:54] transpile finished in 3.20 s  
[23:56:54] preprocess started ...  
[23:56:54] preprocess finished in 9 ms  
[23:56:54] webpack started ...  
[23:56:54] copy finished in 3.42 s  
[23:56:59] webpack finished in 5.33 s  
[23:56:59] sass started ...  
Without `from` option PostCSS could generate wrong source map and will not find Browserslist config. Set it to CSS file path  
or to `undefined` to prevent this warning.  
[23:57:01] sass finished in 1.69 s  
[23:57:01] postprocess started ...  
[23:57:01] postprocess finished in 17 ms  
[23:57:01] lint started ...  
[23:57:01] build dev finished in 10.44 s  
[23:57:01] watch ready in 10.52 s  
[23:57:01] dev server running: http://localhost:8100/  
  
[OK] Development server running!  
Local: http://localhost:8100  
External: http://192.168.99.1:8100, http://192.168.154.1:8100, http://192.168.150.1:8100, http://192.168.1.128:8100  
DevApp: ionic-hello-world@8100 on LAPTOP-T904060I  
  
[23:57:04] lint finished in 3.18 s
```

Mediante el comando `ionic serve` desplegamos la aplicación en un servidor web local (<http://localhost:8100/>)

# Ionic CLI

Como primera aproximación, podemos visualizar nuestra aplicación usando un navegador como Chrome, que dispone de un modo de visualización que simula la pantalla de un móvil



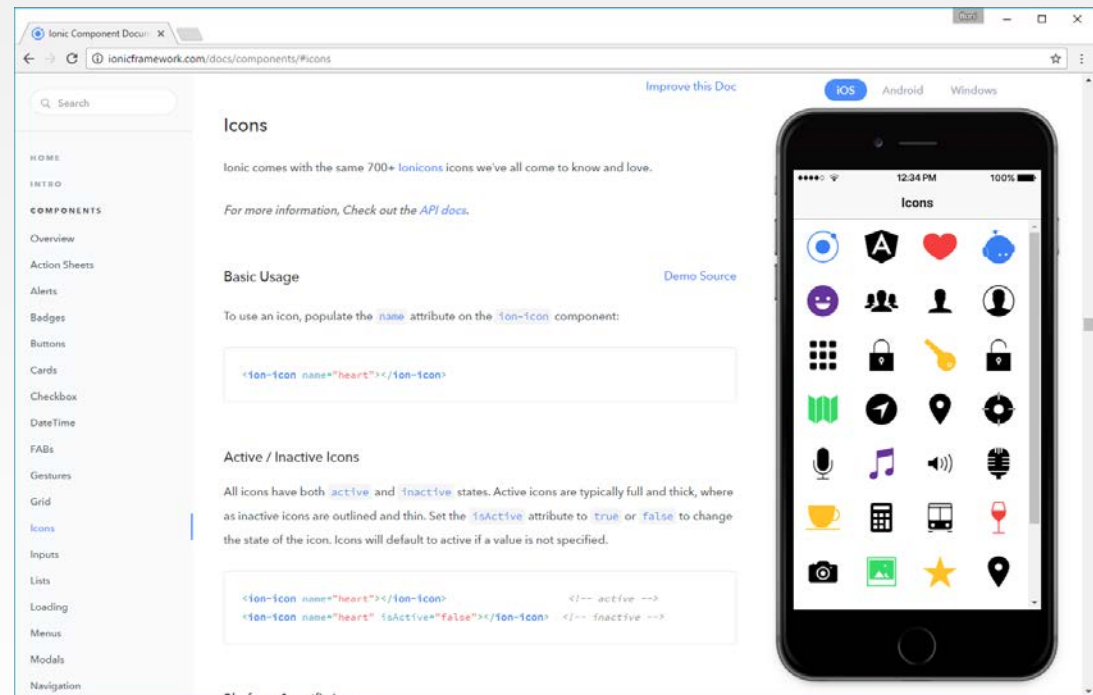
# Índice

---

1. Introducción
2. Ionic
3. Ionic CLI
4. **Componentes**
5. Ionic View

# Componentes

- La implementación de un aplicación Ionic supone la creación de una aplicación Angular usando **componentes** propios de Ionic
- La documentación oficial es muy buena (dispone de ejemplos interactivos)



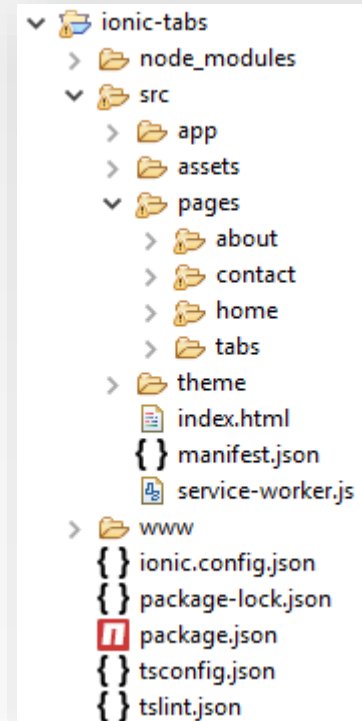


# Componentes

- Vamos a ver como ejemplo la aplicación “tabs” que se puede crear con Ionic CLI

```
> ionic start ionic-tabs

? What starter would you like to use:
> 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
```



```

v ionic-tabs
  > node_modules
  v src
    > app
    > assets
    v pages
      > about
      > contact
      > home
      > tabs
    > theme
    index.html
    manifest.json
    service-worker.js
  > www
    ionic.config.json
    package-lock.json
    package.json
    tsconfig.json
    tslint.json
  
```

Fork me on GitHub

# Componentes

- La navegación con pestañas se hace a través de elementos `<ion-nav>`

app.component.ts

```
import { Component } from '@angular/core';
import { Platform } from 'ionic-angular';
import { StatusBar } from '@ionic-native/status-bar';
import { SplashScreen } from '@ionic-native/splash-screen';
import { TabsPage } from '../pages/tabs/tabs';

@Component({
  templateUrl: 'app.html'
})
export class MyApp {
  rootPage:any = TabsPage;

  constructor(platform: Platform, statusBar: StatusBar, splashScreen: SplashScreen) {
    platform.ready().then(() => {
      statusBar.styleDefault();
      splashScreen.hide();
    });
  }
}
```

app.html

```
<ion-nav [root]="rootPage"></ion-nav>
```

De esta forma  
establecemos que en la  
página principal habrá  
una estructura de  
navegación cuya página  
principal estará definido  
en el componente  
TabsPage

<http://ionicframework.com/docs/components/#navigation>

# Componentes

## tabs.ts

```
import { Component } from '@angular/core';

import { AboutPage } from '../about/about';
import { ContactPage } from '../contact/contact';
import { HomePage } from '../home/home';

@Component({
  templateUrl: 'tabs.html'
})
export class TabsPage {

  tab1Root = HomePage;
  tab2Root = AboutPage;
  tab3Root = ContactPage;

  constructor() {

  }
}
```

## tabs.html

```
<ion-tabs>
  <ion-tab [root]="tab1Root" tabTitle="Home"
  tabIcon="home"></ion-tab>
  <ion-tab [root]="tab2Root" tabTitle="About"
  tabIcon="information-circle"></ion-tab>
  <ion-tab [root]="tab3Root" tabTitle="Contact"
  tabIcon="contacts"></ion-tab>
</ion-tabs>
```

Las diferentes pestañas  
se definen dentro de  
elementos `<ion-tab>`

# Componentes

## home.ts

```
import { Component } from '@angular/core';
import { NavController } from 'ionic-angular';

@Component({
  selector: 'page-home',
  templateUrl: 'home.html'
})
export class HomePage {

  constructor(public navCtrl: NavController) {

  }

}
```

## home.html

```
<ion-header>
  <ion-navbar>
    <ion-title>Home</ion-title>
  </ion-navbar>
</ion-header>

<ion-content padding>
  <h2>Welcome to Ionic!</h2>
  <p>
    This starter project comes with simple tabs-based layout for apps
    that are going to primarily use a Tabbed UI.
  </p>
  <p>
    Take a look at the <code>src/pages/</code> directory to add or
    change tabs,
    update any existing page or create new pages.
  </p>
</ion-content>
```

En cada pestaña deberá existir una estructura de elementos `ion-header`, `ion-navbar`, `ion-title`, e `ion-content`

# Componentes

---

- Vamos a añadir nuevos elementos a la aplicación
- Por ejemplo, para renderizar botones en aplicaciones Ionic usamos la etiqueta HTML `button` junto con la directiva `ion-button`

```
<button ion-button color="light">Light</button>  
<button ion-button>Default</button>  
<button ion-button color="secondary">Secondary</button>  
<button ion-button color="danger">Danger</button>  
<button ion-button color="dark">Dark</button>
```

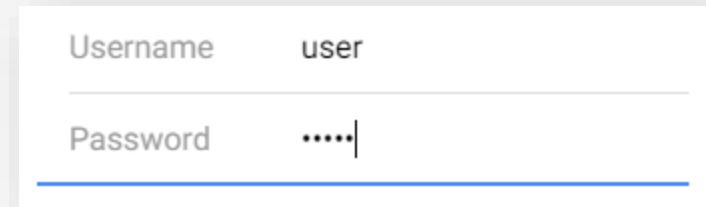


<http://ionicframework.com/docs/components/#buttons>

# Componentes

- La etiqueta HTML `input` se sustituye por `ion-input`:

```
<ion-list>
  <ion-item>
    <ion-label fixed>Username</ion-label>
    <ion-input type="text" value=""></ion-input>
  </ion-item>
  <ion-item>
    <ion-label fixed>Password</ion-label>
    <ion-input type="password"></ion-input>
  </ion-item>
</ion-list>
```



A screenshot of a mobile application form. It features two input fields stacked vertically. The first field is labeled 'Username' and contains the text 'user'. The second field is labeled 'Password' and contains five dots, indicating a password input. The form has a clean, modern design with a white background and a blue underline at the bottom.

<http://ionicframework.com/docs/components/#inputs>

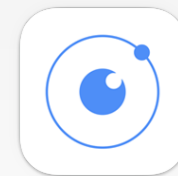
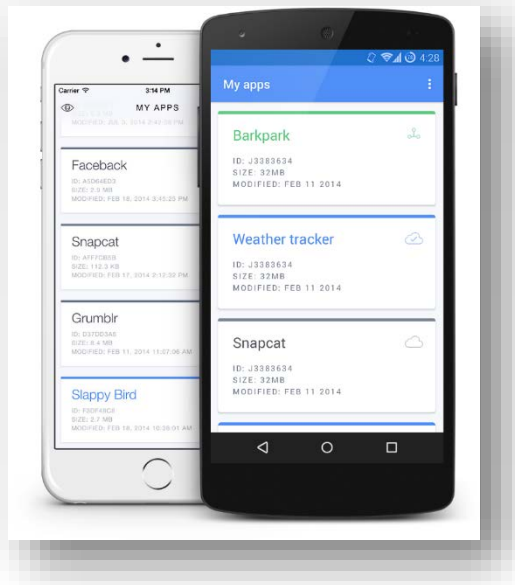
# Índice

---

1. Introducción
2. Ionic
3. Ionic CLI
4. Componentes
5. Ionic View

## Ionic View

- Ionic View es una aplicación móvil que nos permite compartir aplicaciones Ionic durante el desarrollo de las mismas, sin necesidad de desplegarla de forma manual o definitiva (en el *market* de Apple o Google)



ionicview



App Store

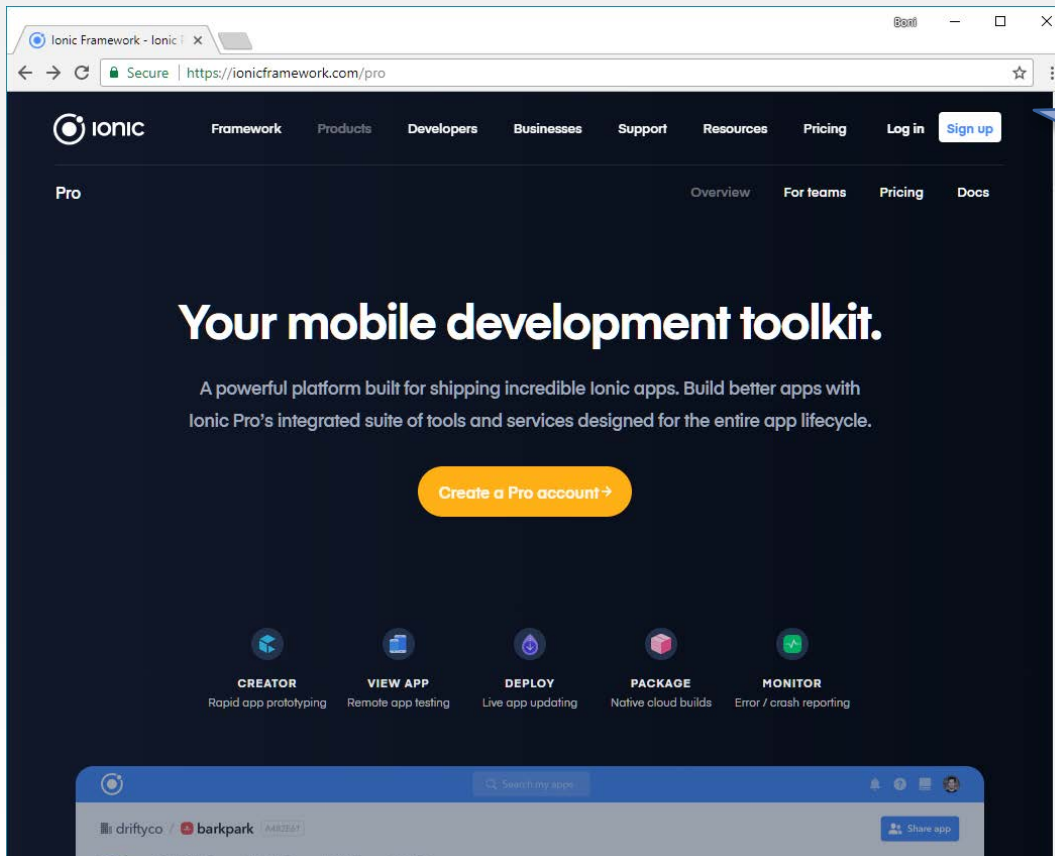


Google play

<http://view.ionic.io/>



# Ionic View



Como requisito, habrá que crearse una cuenta en <https://ionicframework.com/pro>

Problema: desde abril de 2018 se ha eliminado la app Ionic View del app store de Apple por supuestos problemas de seguridad

<https://blog.ionicframework.com/update-on-ionic-view-for-ios/>

# Ionic View

Mediante el comando `ionic link` subimos nuestra aplicación a Ionic Pro. Será necesario configurar una clave pública SSH

```
> ionic link
✓ Looking up your apps - done!

? Which app would you like to link (Use arrow keys)
> Create a new app
  Nevermind
D:\tmp\new\ionic-hello-world>ionic link
✓ Looking up your apps - done!

? Which app would you like to link Create a new app

? Please enter a name for your new app: ionic-hello-world

Ionic Pro uses a git-based workflow to manage app updates.
You will be prompted to set up the git host and repository for this new app. See
the docs[1] for more information.

[1]: https://ionicframework.com/docs/pro/basics/git/

? Which git host would you like to use? Ionic Pro
> ionic config set app_id "f448394e" --json
[OK] app_id set to "f448394e" in .\ionic.config.json!
> ionic git remote
> git remote add ionic git@git.ionicjs.com:bgarciautad/ionic-hello-world.git
[OK] Added remote ionic.
[OK] Project linked with app f448394e!
```

# Ionic View

Para poder publicar la aplicación en Ionic View habrá que hacer:

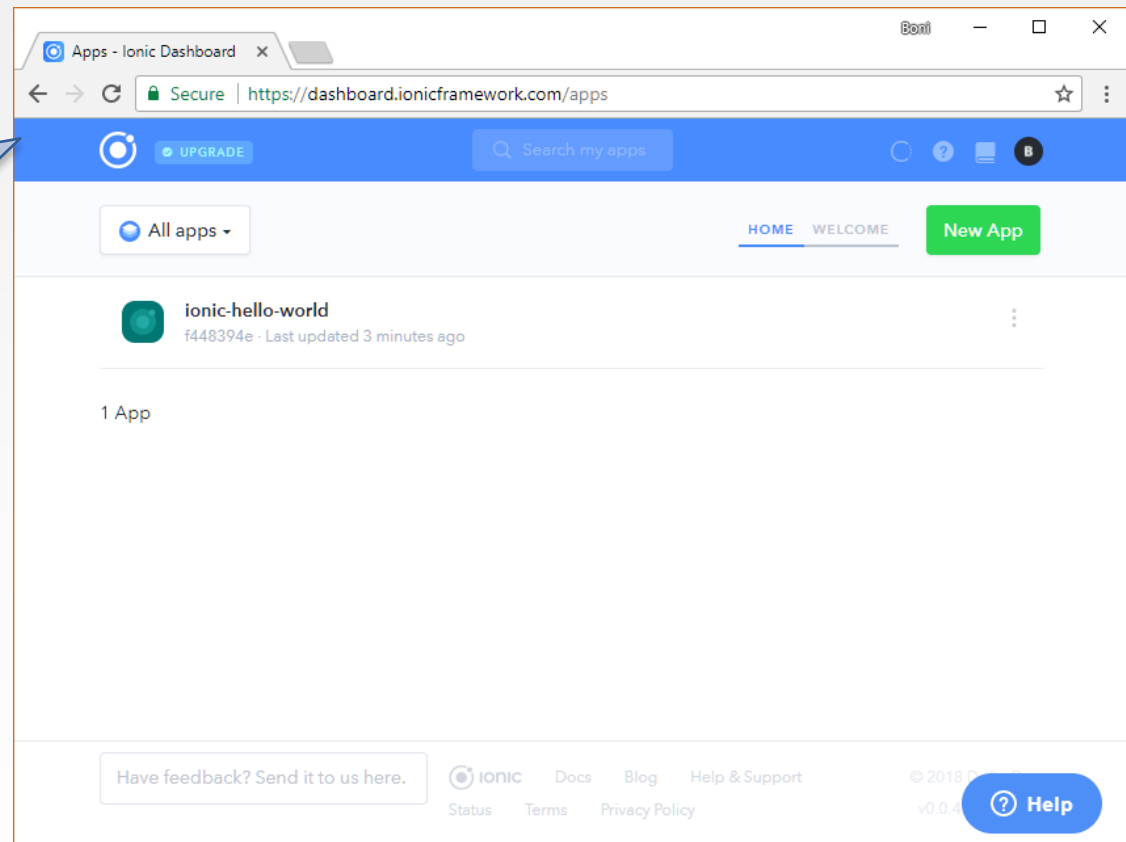
1. Subir el código fuente a un repositorio Git
2. Desplegarla en un canal público (configurado en Ionic Pro)

```
> git push ionic master
The authenticity of host 'git.ionicjs.com (35.166.119.60)' can't be established.
ECDSA key fingerprint is SHA256:Wv6rYBk1T3XSPfybwccdQJ+lhmR+FC6F61d1i46VPFI.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'git.ionicjs.com,35.166.119.60' (ECDSA) to the list of
known hosts.
Enter passphrase for key 'C:\Users\boni\.ssh\id_rsa':
Counting objects: 85, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (79/79), done.
Writing objects: 100% (85/85), 1.20 MiB | 2.15 MiB/s, done.
Total 85 (delta 0), reused 0 (delta 0)
remote: Received:
remote:      Branch:  master
remote:      Commit Sha:  87ecfa8a72248c6e118154bb9df60b496260954a
remote:      Commit Msg:  Initial commit
remote:  Checking for previous builds...
remote:  New commit or branch detected triggering new build view build log here:
remote:  https://dashboard.ionicjs.com/app/f448394e/code/builds/3410c01e-42fb-
4a31-aa81-6629f5318c2c
To git.ionicjs.com:bgarciautad/ionic-hello-world.git
* [new branch]      master -> master

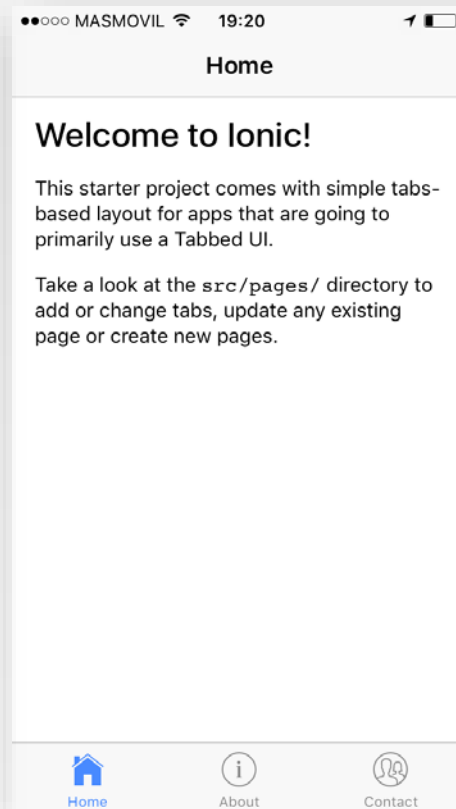
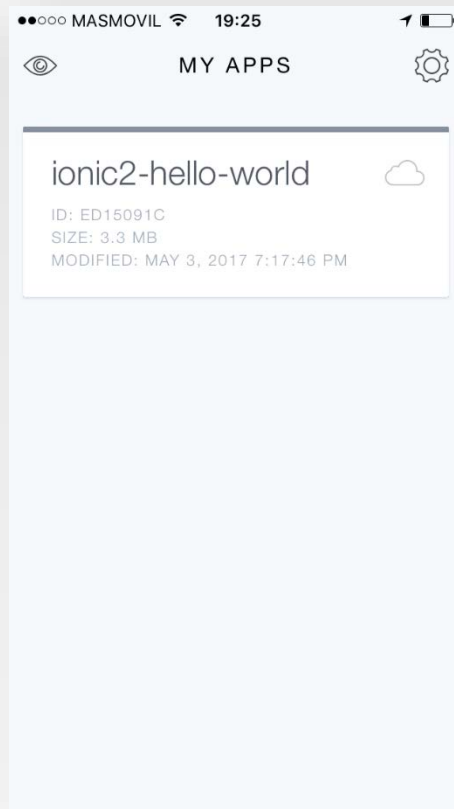
> cordova plugin add cordova-plugin-ionic --save --variable APP_ID="f448394e" --
variable CHANNEL_NAME="Master" --variable UPDATE_METHOD="background"
Adding cordova-plugin-ionic to package.json
Saved plugin info for "cordova-plugin-ionic" to config.xml
```

# Ionic View

En este momento deberíamos ver nuestra app en el tablero (*dashboard*)



# Ionic View



Podemos compartir nuestra aplicación usando su identificador único (id) en la app Ionic View (actualmente sólo es visible en dispositivos Android pero se supone que la compatibilidad con Apple volverá)