

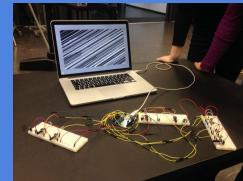
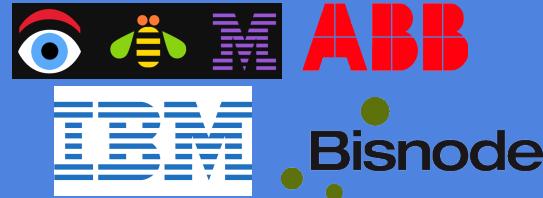


# ionic

Building Hybrid Apps with AngularJS and Ionic

# Roman Rast

- **What I do**
  - Mobile App Dev
  - UX / Interaction Design
  - Frontend
  - Zühlke
- **Slides and stuff**
  - <https://github.com/romanrast/fhnwionic>
- **Flipboard** my Ionic magazin +180 followers
  - <http://flip.it/q-lfz>
- **Twitter**
  - @romanrast
- **Contact**
  - [roman.rast@zuehlke.com](mailto:roman.rast@zuehlke.com)



# Use Web Technologies



# Ionic

- HTML5 Framework for hybrid mobile apps
- Build on AngularJS
- Uses Cordova
- First alpha in 2013, 1.0 Beta 2014, 1.0 2015
- High performance
- iOS > 7, Adnroid > 4.1
- Open source
- Many Frameworks built-in
  - AngualrJS, Cordova, Gulp, Sass, NodeJS, Bower
- Custom directives/components -> <ion-list>
- Drifty Co



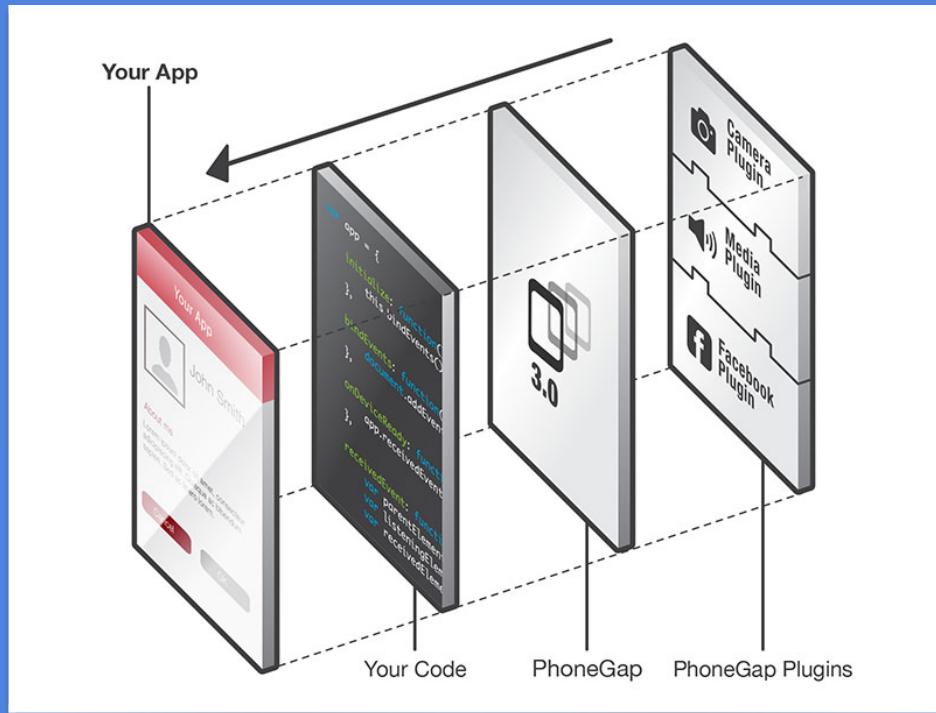
# Architecture





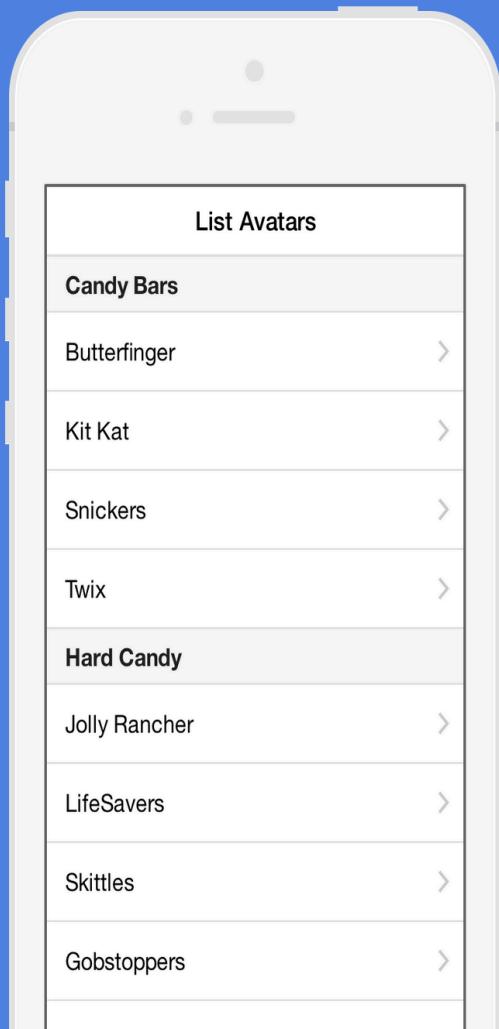
- Wraps app into a native app
- App uses WebView (browser of the phone without url bar and other stuff)
- Offers APIs to access camera, contacts, accelerometer finger print scanner etc.
- Develop a single code base for iOS and Android...
- PhoneGap/Cordova - Adobe





# Pro | Cons

Pro	Cons
Open-source and free	Rendering complex UI can be slow
AngularJS, HTML, CSS	Not the best choice for graphic intense UI
SASS support	Android can be a hard to show same performance on mid-range phones
Offers large number of 3rd party resources, plugins, themes, starter apps (ngCordova Plugins, ionic market, ionicons)	
Great from	
One code base for iOS and Android (almost)	



# Lists

```
<div class="list">

  <div class="item item-divider">
    Candy Bars
  </div>

  <a class="item" href="#">
    Butterfinger
  </a>

  <a class="item" href="#">
    Kit Kat
  </a>

  ...
</div>
```

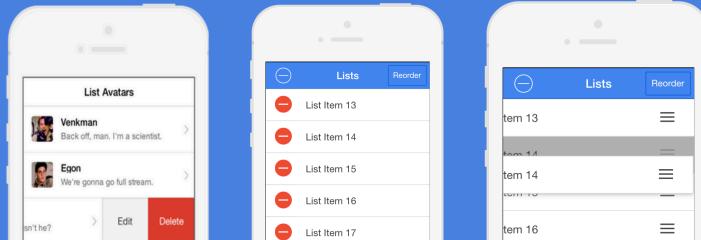
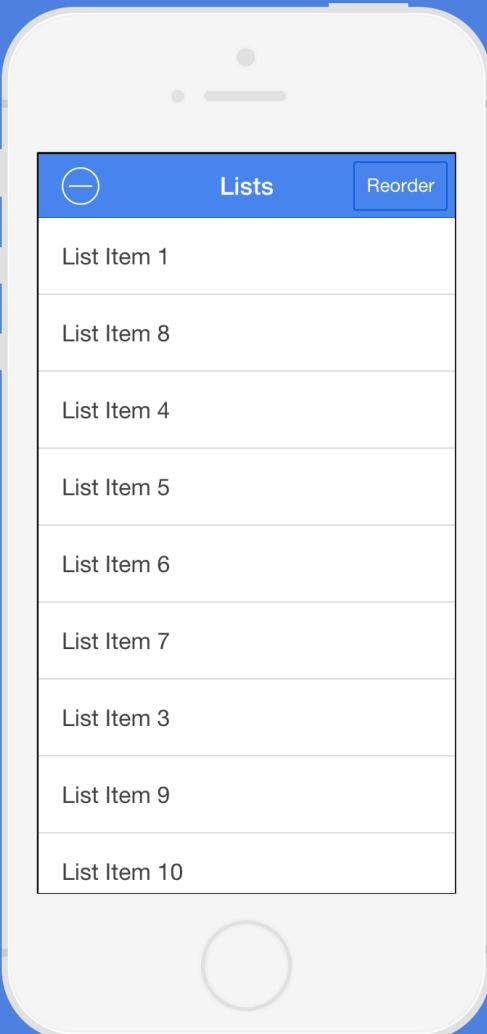
# Complex Lists

- AngularJS Directive
- Buttons exposed by swiping
- Reorder
- Delete

```
<ion-list>

  <ion-item ng-repeat="item in items" item="item">
    List Item {{ item.id }}
  </ion-item>

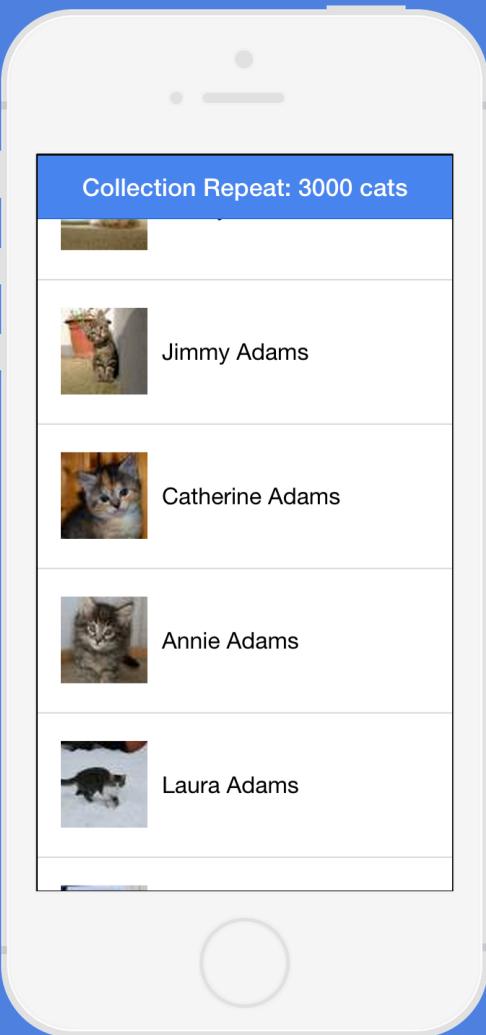
</ion-list>
```



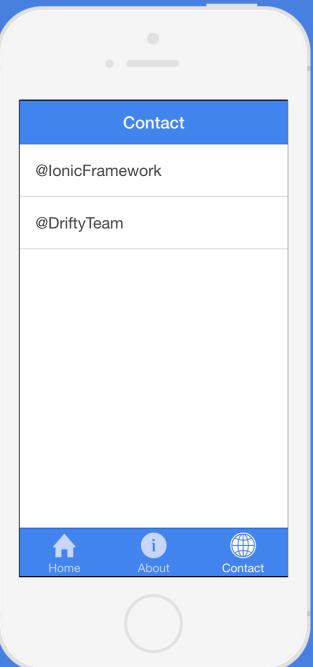
# Collection Repeat

- Similar to Angular's `ng-repeat`
- Inspired by iOS's UICollectionView
- Scroll through thousands of items
- Only renders the viewable items
- Smooth jank-free scrolling

```
<div class="list">
  <div collection-repeat="c in contacts">
    <h2>{{ c.name }}</h2>
    <p>{{ c.email }}</p>
  </div>
</div>
```



# Tabs



The image shows a white iPhone displaying a mobile application. The screen has a blue header bar with the word "Contact". Below it is a white content area containing two lines of text: "@IonicFramework" and "@DriftyTeam". At the bottom is a blue footer bar with three icons: a house (labeled "Home"), a person (labeled "About"), and a globe (labeled "Contact").

## Tabs

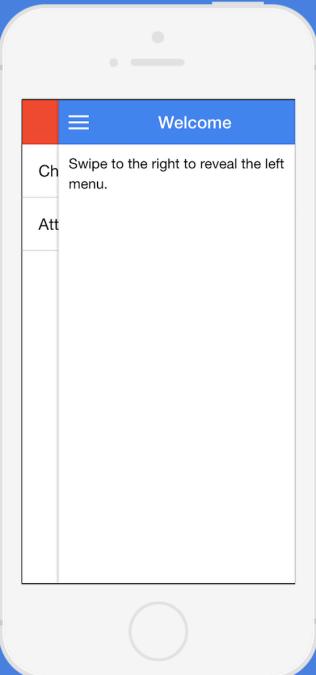
- Nested views
- Each tab has its own nav history
- Abstract states in AngularUI Router

```
<ion-tabs>
  <ion-tab title="Home" icon="ion-home">
    <ion-nav-view name="home"></ion-nav-view>
  </ion-tab>

  <ion-tab title="About" icon="ion-information">
    <ion-nav-view name="about"></ion-nav-view>
  </ion-tab>

  <ion-tab title="Contact" icon="ion-ios7-world">
    <ion-nav-view name="contact"></ion-nav-view>
  </ion-tab>
</ion-tabs>
```

# Side Menu



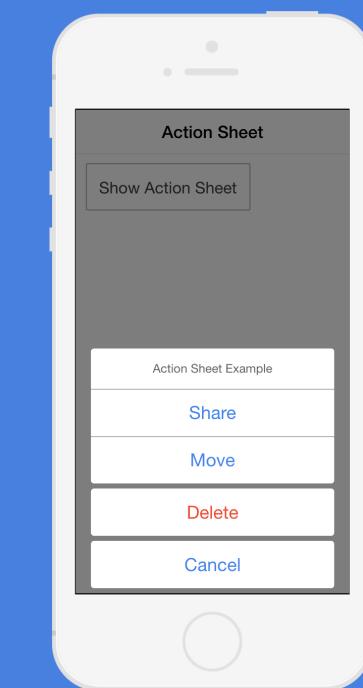
The image shows a white smartphone with a black screen. On the screen, there is a mobile application interface. At the top, there is a blue header bar with a red section on the left containing three horizontal lines (a menu icon) and the word "Welcome" on the right. Below the header, there is some text: "Ch" followed by "Swipe to the right to reveal the left menu." and "Att". The main content area is currently empty. To the right of the phone, the text "Side Menu" is displayed in large white letters. Below this, a block of code is shown:

```
<ion-side-menus>
  <ion-side-menu-content>
    <ion-nav-view name="menuContent"></ion-nav-view>
  </ion-side-menu-content>

  <ion-side-menu side="left">
    <ion-header-bar class="bar-assertive">
      <h1 class="title">Left Menu</h1>
    </ion-header-bar>

    <ion-content>
      ...
    </ion-content>
  </ion-side-menu>
</ion-side-menus>
```

# Action Sheet

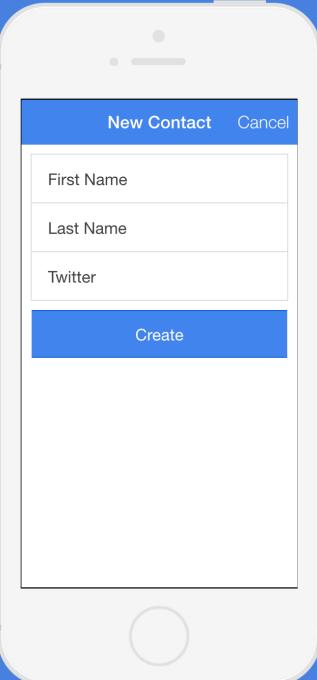


## Action Sheet

- AngularJS Service
- Inject into controllers

```
$ionicActionSheet.show({
  titleText: 'Action Sheet Example',
  buttons: [
    { text: 'Share' },
    { text: 'Move' },
  ],
  destructiveText: 'Delete',
  cancelText: 'Cancel',
  buttonClicked: function(index) {
    console.log('BUTTON CLICKED', index);
    return true;
  }
});
```

# Modal

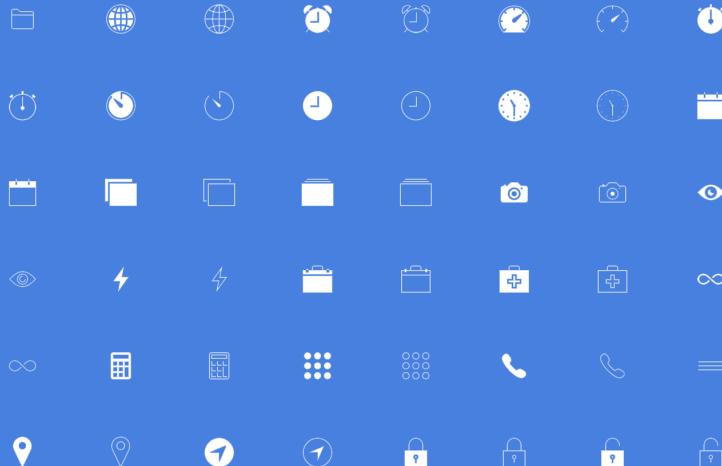


## Modal

- AngularJS Service
- Inline or external template

```
$ionicModal.fromTemplateUrl('modal.html', {  
  scope: $scope  
}).then(function(modal) {  
  $scope.modal = modal;  
});
```

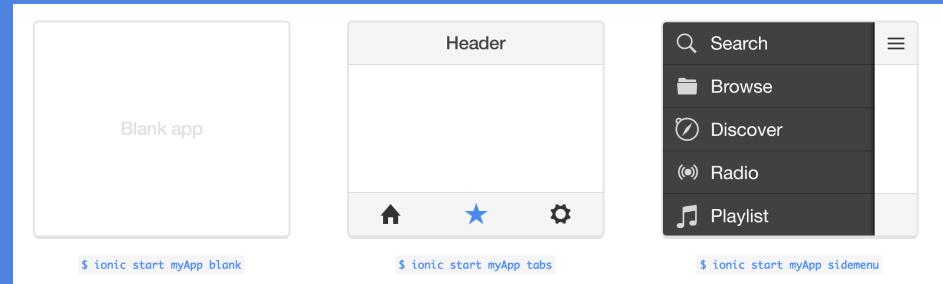
# Ionicons



Over 700 MIT licensed font-icons included  
[ionicons.com](http://ionicons.com)

# Installation

- Install Node.js (<https://nodejs.org/>)
- \$ npm install -g cordova ionic
- \$ sudo ionic start myApp tabs
- \$ cd myApp
- (change access right if necessary on mac)
- \$ sudo ionic platform add ios | android
- \$ sudo ionic build ios
- \$ sudo ionic setup sass
- \$ sudo ionic serve
- \$ sudo ionic run ios | ionic run android (when device is plugged in)



Keep Xcode and iOS versions always up to date!

# Plugins

- Out of the box
  - cordova-plugin-splashscreen
  - cordova-plugin-statusbar
  - ionic-plugin-keyboard
  - More on ngCordova
    - <http://ngcordova.com/>

```
angular.module('starter', ['ionic', 'starter.controllers', 'starter.services'])

.run(function($ionicPlatform) {
  $ionicPlatform.ready(function() {
    // Hide the accessory bar by default (remove this to show the accessory bar above the keyboard
    // for form inputs)
    if (window.cordova && window.cordova.plugins && window.cordova.plugins.Keyboard) {
      cordova.plugins.Keyboard.hideKeyboardAccessoryBar(true);
      cordova.plugins.Keyboard.disableScroll(true);

    }
    if (window.StatusBar) {
      // org.apache.cordova.statusbar required
      // StatusBar.styleDefault();
      $cordovaStatusBar.style(1) //Light
    }
  });
})
```

# CLI

```
$ cd myApp
```

---

```
# Add iOS and Android platform for cordova
```

```
$ ionic platform add ios | $ ionic platform add android
```

```
# Build Debug version
```

```
$ ionic build ios | $ionic build android
```

```
# Build Prod version
```

```
$ ionic build ios --release
```

```
# Run in Browser
```

```
$ ionic serve
```

```
$ ionic serve --lab
```

```
# Run in emulator
```

```
$ ionic emulate ios
```

```
# Run on real device if connected
```

```
$ ionic run ios
```

```
$ ionic run android
```

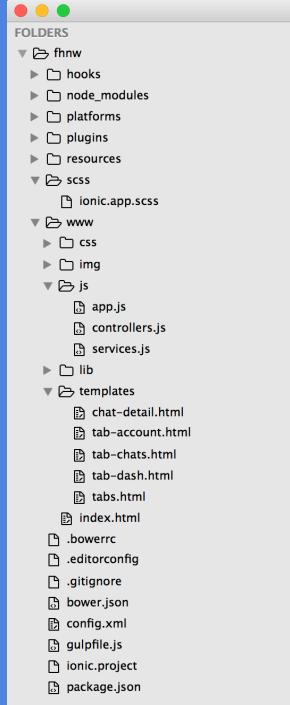
---

```
$ ionic package debug android (debug or release | android or ios)
```

# Prepare Your Device

- Android - activate developer mode
  - Settings/about phone tip on Android Version until message appears
  - Go back to Settings, then Developer options and check USB debugging
- iPhone
  - Just plug in an press play in xcode – yay!

# App Structure



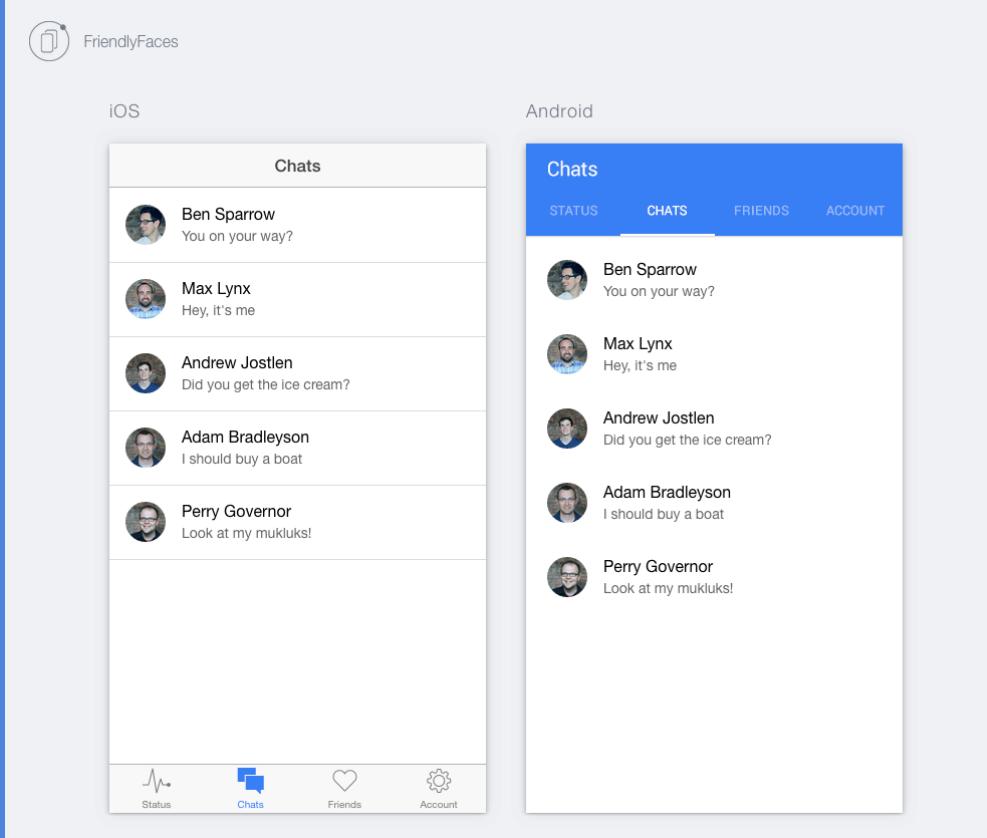
- Hooks
- Plugins
- **www**
  - bowerrc
  - gitignore
  - config.xml
  - gulpfile.js
  - package.json
  - **css**
  - **img**
  - **js**
  - **scss**

# App Structure

- Work in the **www** folder on the **root** of your app
- The code will be copied down to platforms/ios and android
- For iOS go to platforms/ios/xxx.xcodeproj to run app on device or Simulator

# iOS vs Android

iOS swipe back



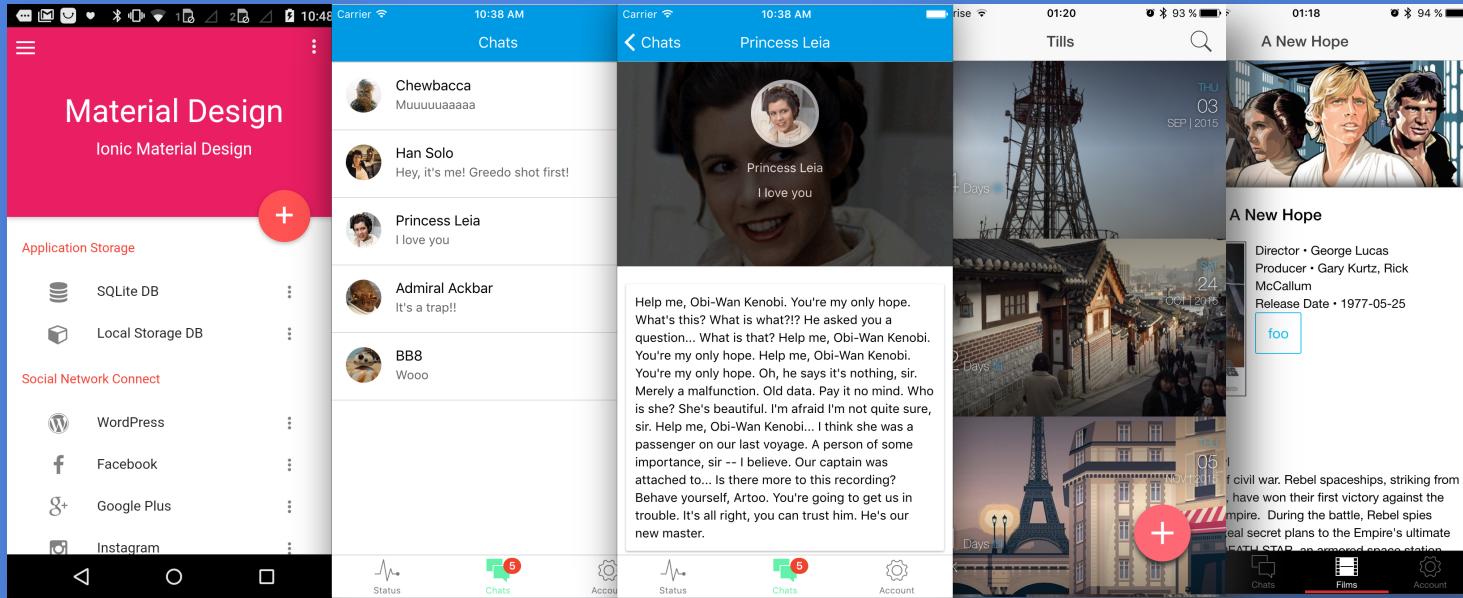
Android nav  
on top

# Debugging & Testing

- Use Google CHROME!
- Developer Tools



# Demo Time



# Android

## Android Prerequisite

- Android SDK
  - Ant
- 

### # MacOs

- brew install ant
- Having ANDROID\_HOME environment variable set (in your .bashrc or .bash\_profile file)

### # Path to your sdk in your file system

- export ANDROID\_HOME=~/Library/Android/sdk

# Android Splashscreen

- In config.xml
- FadeSplashScreenDuration only works on iOS (Android will crash)

```
<!-- for activating android splashscreen -->
<preference name="ShowSplashScreen" value="true" />
<preference name="SplashScreen" value="screen"/>
<preference name="SplashScreenDelay" value="3000"/>
<preference name="AutoHideSplashScreen" value="true" />
<preference name="SplashShowOnlyFirstTime" value="false" />
<preference name="FadeSplashScreen" value="false"/>
<!-- <preference name="FadeSplashScreenDuration" value="0.3"/> -->
<feature name="SplashScreen">
    <param name="android-package" value="org.apache.cordova.splashscreen.SplashScreen" />
</feature>
```

# Add plugins

---

Use ngCordova plugins (<http://ngcordova.com>)

---

Install

```
$ bower install ngCordova (in root of app)
```

Then in index.html add src to ng-cordova.js before cordova.js

```
<script src="lib/ngCordova/dist/ng-cordova.js"></script>
<script src="cordova.js"></script>
```

Then Inject as an Angular dependency

```
angular.module('myApp', ['ngCordova'])
```

Add the plugin to your project using the Cordova CLI

```
$ cordova plugin add cordova-plugin-media-capture
```

Wrap each plugin call with the deviceready event - important !

```
$ionicPlatform.ready(function() {
  $cordovaPlugin.someFunction().then(success, error);
});
```

```
module.controller('MyCtrl', function($scope,
$cordovaCapture) {
```

# Performance

- **Comes with latest version of Ionic!**
- On Android → Crosswalk
- Chromium
- Develop around device fragmentation
- Provide a feature rich experience on all Android 4.x devices
- Easily debug with Chrome DevTools
- Improve the performance of your HTML, CSS, and JavaScript
- Increases app size by about 20Mb

```
# add the Crosswalk browser to your Android project  
$ ionic browser add crosswalk
```

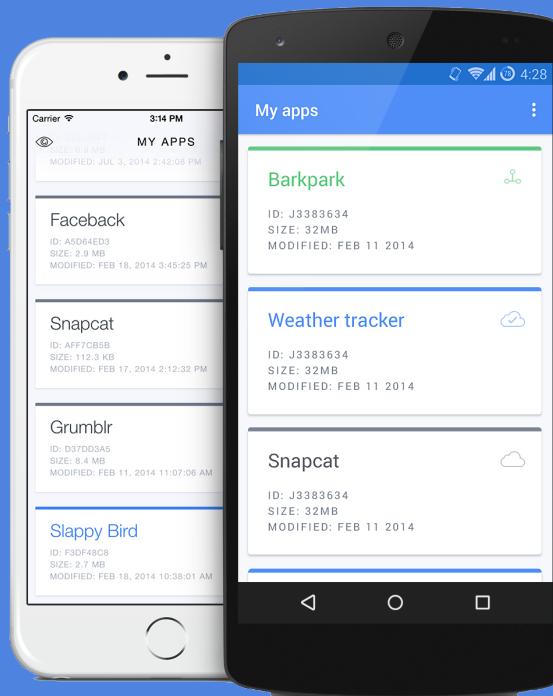
# Ionic View

- Send your app to users for testing

\$ ionic login

\$ ionic upload

\$ ionic share EMAIL

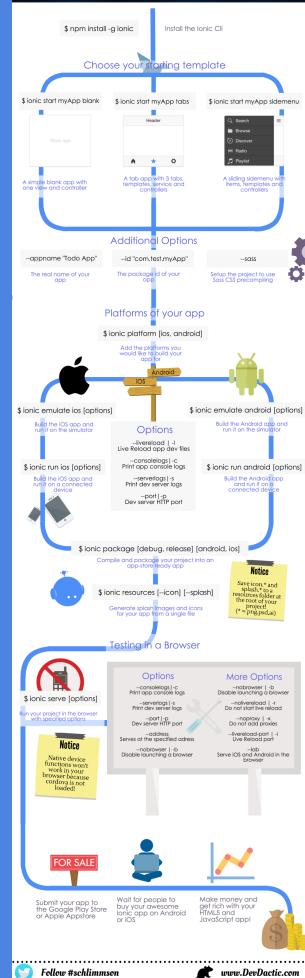


# Links

- <http://ionicframework.com/>
- <http://ngcordova.com/>
- <https://www.airpair.com/javascript/posts/a-year-using-ionic-to-build-hybrid-applications>
- Flipboard my Ionic magazin +180 followers
  - <http://flip.it/q-lfz>

# ionic Cheatsheet

by devDuctic  
from zero to app!



# ngCordova

## CORDOVA WITH THE POWER OF **ANGULARJS**

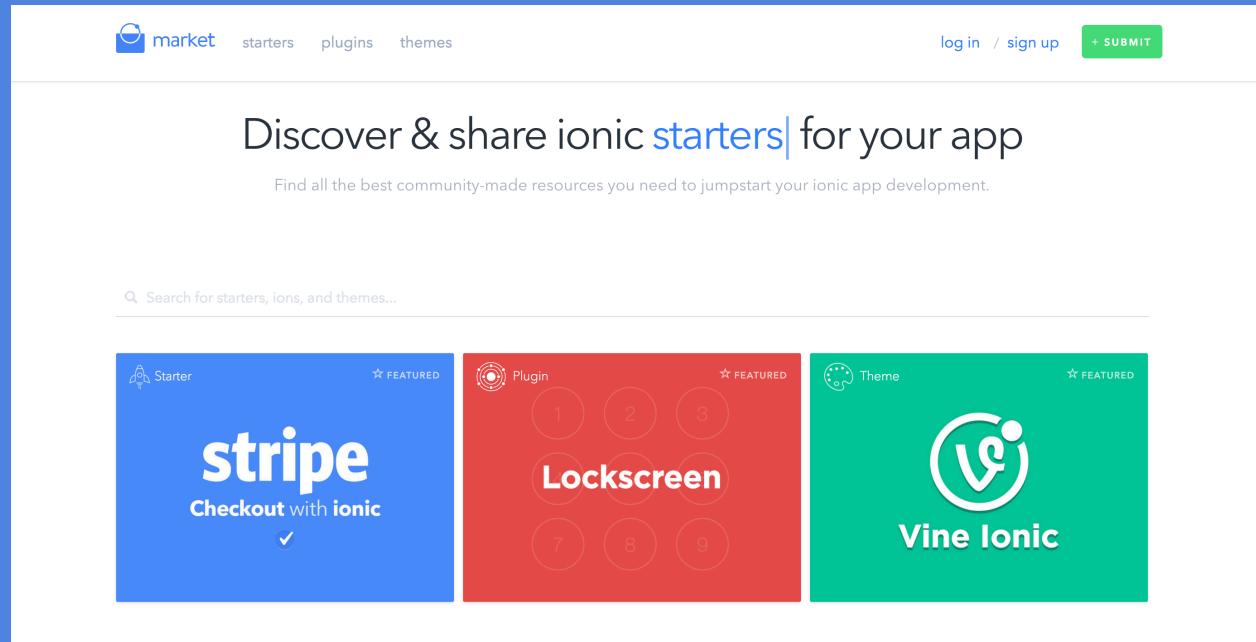
ngCordova is a collection of **70+** AngularJS extensions on top of the Cordova API that make it easy to build, test, and deploy Cordova mobile apps with AngularJS.



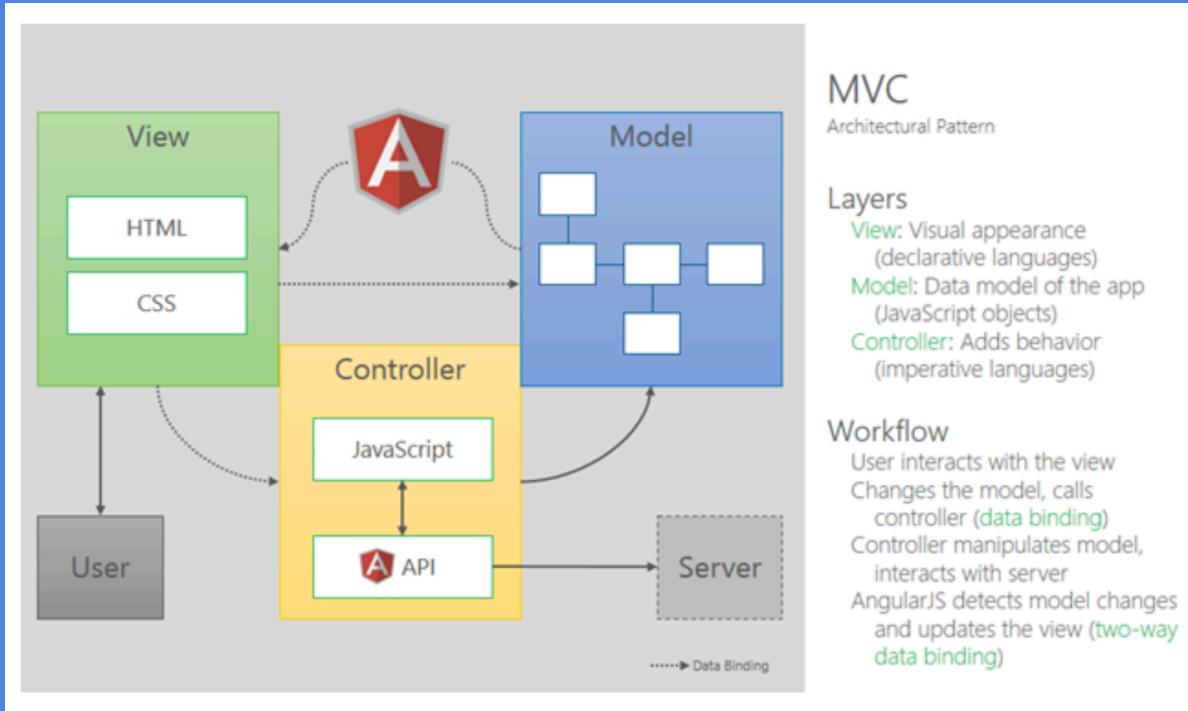
# Market.ionic.io

- <http://market.ionic.io/>

- Themes
- Starters
- Plugins



# MVC



# MVC

- View = Templates (chats.html)
  - www/templates/...
- Controllers = angular.module('starter.controllers', []).controller('DashCtrl', function(\$scope) {})
  - www/js/controllers.js
- Model = best in services.js, call it with REST and read/persist in db
  - www/js/services.js

# SPA (Single Page Application)

The diagram illustrates the SPA architecture through code snippets and corresponding DOM structures.

**Code Snippets:**

```
angular.module('helloWorldApp', [])
  .config(function ($routeProvider) {
    $routeProvider
      .when('/', {
        templateUrl: 'views/main.html',
        controller: 'MainCtrl'
      })
      .when('/about', {
        templateUrl: 'views/about.html',
        controller: 'AboutCtrl'
      })
      .otherwise({
        redirectTo: '/'
      });
  });

angular.module('helloWorldApp')
  .controller('MainCtrl', function ($scope) {
```

**DOM Structures:**

- Angular Controller:** A green box containing the MainCtrl controller code. It is connected by a blue arrow to the `controller: 'MainCtrl'` entry in the first code snippet.
- View Container:** An orange box containing the `<div class="container" ng-view=""></div>` directive. It is connected by a blue arrow to the `templateUrl: 'views/main.html'` entry in the first code snippet.
- Hero Unit View:** A blue box containing the `<div class="hero-unit"><h1>'Allo, 'Allo!</h1></div>` content. It is connected by a blue arrow to the `templateUrl: 'views/main.html'` entry in the first code snippet.

**Definitions:**

- SPA**: Single Page Apps
- Define routes with [`\$routeProvider`](#) service  
Placeholder with `..` (e.g. `/admin/users/:userId`)  
Access route parameter values with [`\$routeParams`](#) service
- Define where view should be included in index.html using [`ng-view`](#)
- URL Modes  
Hashbang and HTML5 mode  
See [`\$location`](#) service docs for details

