Introduction about hybrid & Ionic framework

Thinh VoXuan

Sprint 2016

GEEK UP

Agenda

- Native vs Hybrid Apps
- Intro Ionic
  - Apache Cordova
  - AngularJS
  - Eco system
- Ionic Basic
  - Installation
  - Ionic CLI
  - Directory Source code
  - Routing
  - View lifecycle
  - UI Component
  - Demo

Native vs Hybrid Apps

Introduction about hybrid & Ionic framework

Pros:
- Natives UI allows users to quickly learn the app
- Access to device hardware software (GPS, Location, shake, Calendar, etc…)
- Native applications have the best performance, highest security, and best user experience.

Cons:
- Entirely separate code bases
- Timely & expensive development
- More Platforms. More Problems.

Pros:
- Single code-base, multi platform
- Development Cost Reasonable
- All the content will be updated from web directly
- The in-app interaction has a look and feel consistent
- Web technology

Cons:
- Run on Webview
- Base on Plugins
- May require native coding in device-related case

GEEK UP

Part 1 - Intro Ionic

Intro Ionic

Introduction about hybrid & Ionic framework

- Website: http://ionicframework.com/
- Ionic is a powerful HTML5 SDK that helps you build native-feeling mobile apps using web technologies like HTML, CSS, and Javascript.
- top of the world's #1 mobile hybrid Framework

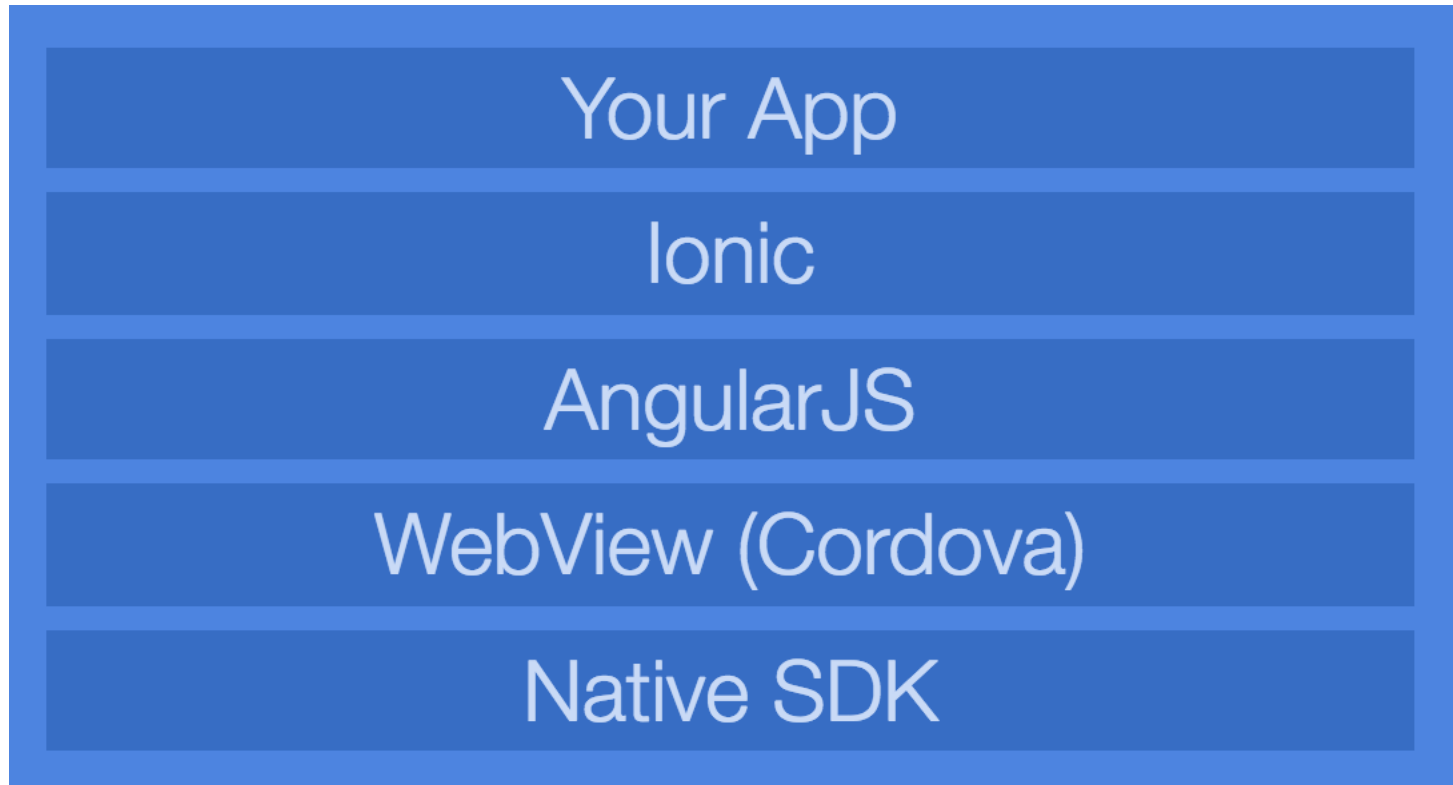- *Cordova + Twitter Bootstrap + AngularJS*

- http://blog.ionic.io/where-does-the-ionic-framework-fit-i

GEEK UP

Intro Ionic

Introduction about hybrid & Ionic framework

Intro Ionic

Introduction about hybrid & Ionic framework

Your App

Ionic

AngularJS
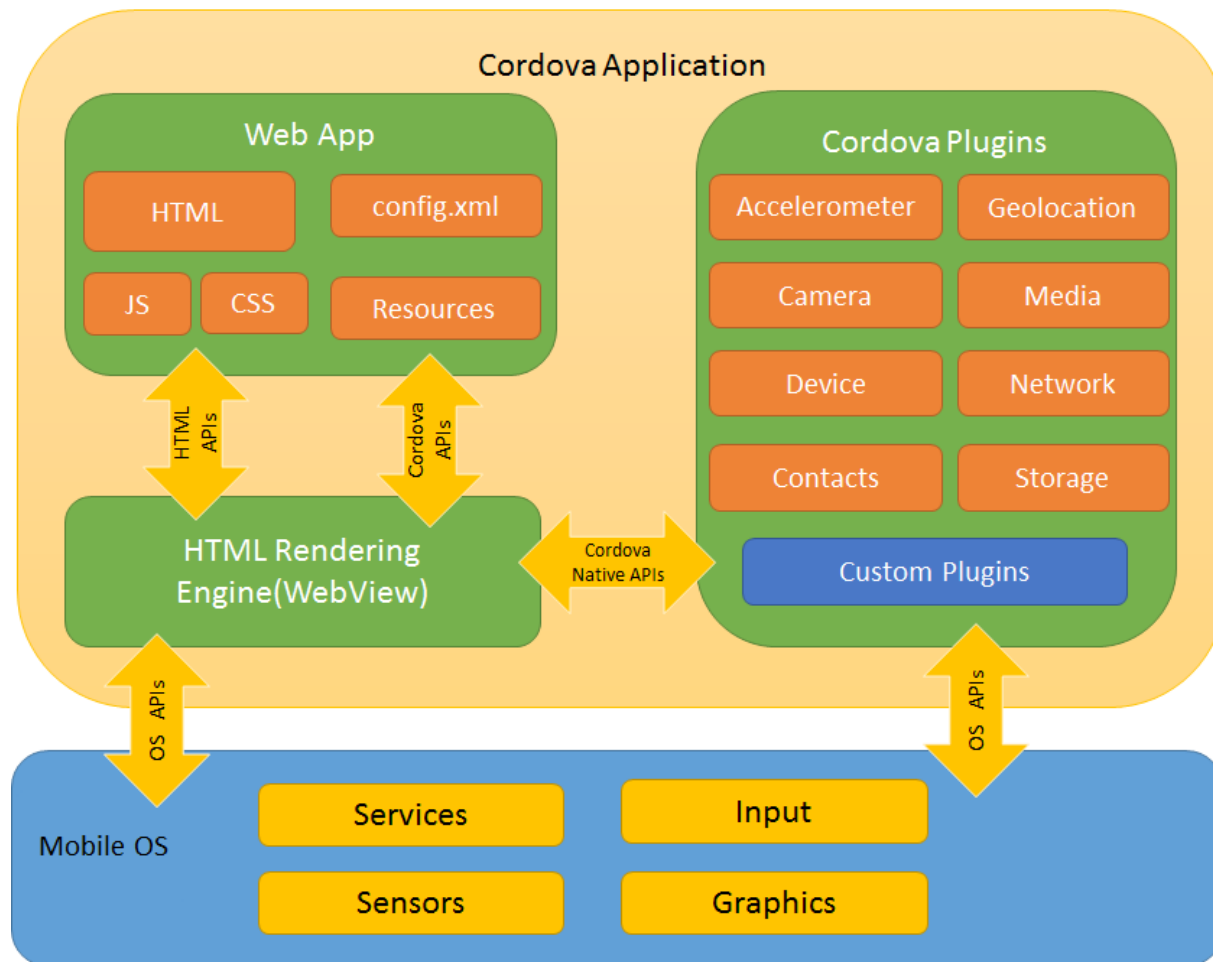
WebView (Cordova)

Native SDK

Cordova

- An open-source mobile development framework

- Use standard web technologies - HTML5, CSS3, and JavaScript for cross-platform development

- Applications execute within wrappers targeted to each platform

- Rely on standards-compliant API bindings to access each device's capabilities such as sensors, data, network status, etc.

GEEK^{UP}

Cordova

Introduction about hybrid & Ionic framework

PhoneGap vs Cordova

- **PhoneGap** is an HTML5 app platform that allows you to author native applications with web technologies **and** get access to APIs **and** app stores. **PhoneGap** is a distribution of Apache **Cordova**. You can think of Apache **Cordova** as the engine that powers **PhoneGap**, similar to how WebKit is the engine that powers Chrome **or** Safari.
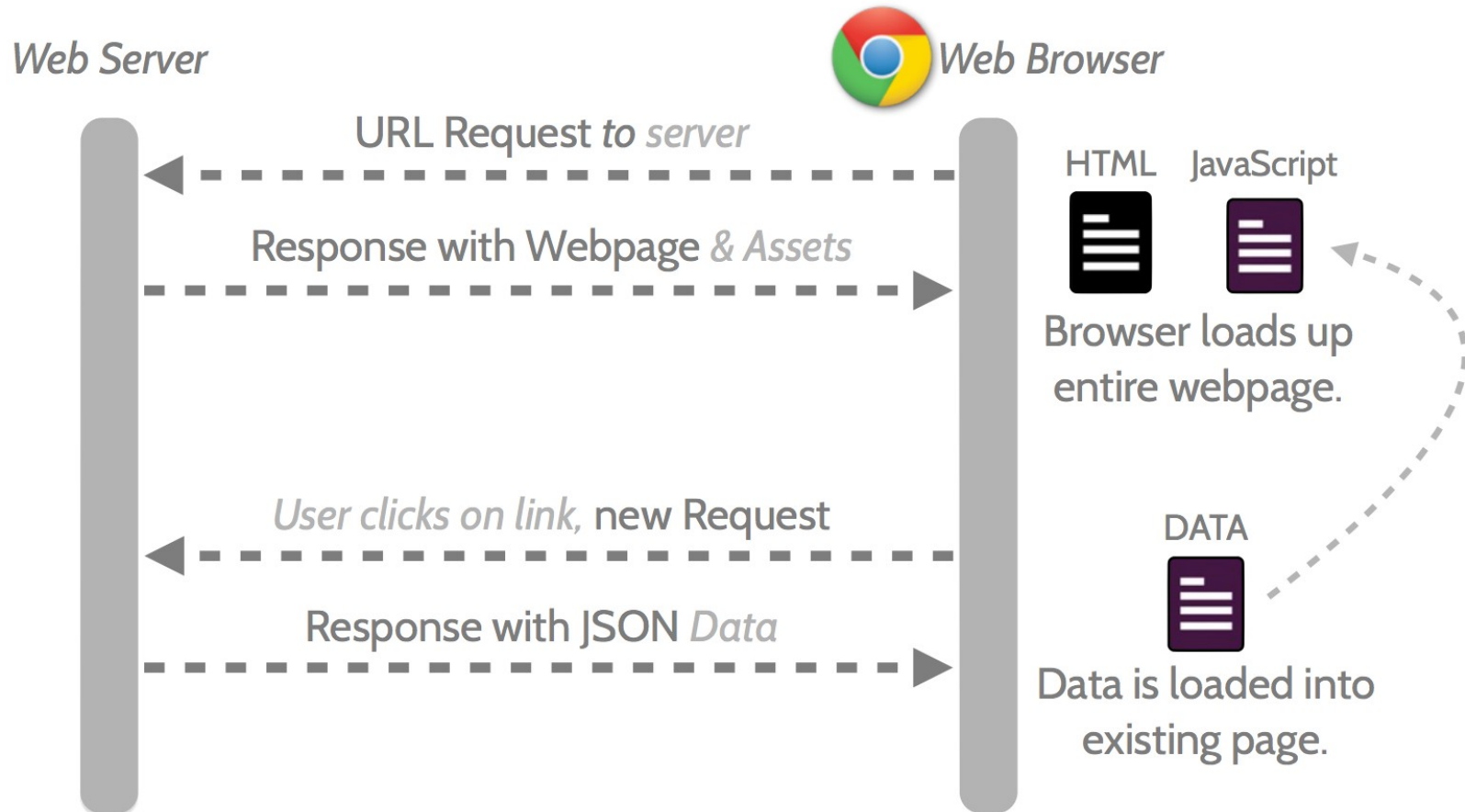
AngularJS

- MV*: Model-View-Whatever
- organize your JavaScript modular
- Interconnection with HTML at the root level
- Data Handling made simple
- Two-way Data Binding

GEEK UP

AngularJS

Introduction about hybrid & Ionic framework

AngularJS

- Controller
- Template & Filtering
- Two-way Data Binding
- Routing & Multiple Views
- Built-in Directives
- Forms
- Custom Directives
- Services & Factory.

Ionic Ecosystem

Introduction about hybrid & Ionic framework

- Ionic market: http://market.ionic.io/
  - Plugins
  - Themes
- Ionic platform: http://ionic.io/platform
  - Live Deploys
  - User Authentications
  - Push Notification
  - Native Builds

Ionic Ecosystem

Introduction about hybrid & Ionic framework

Notifications

Framework

Builds

Support

Updates

Users

GEEK UP
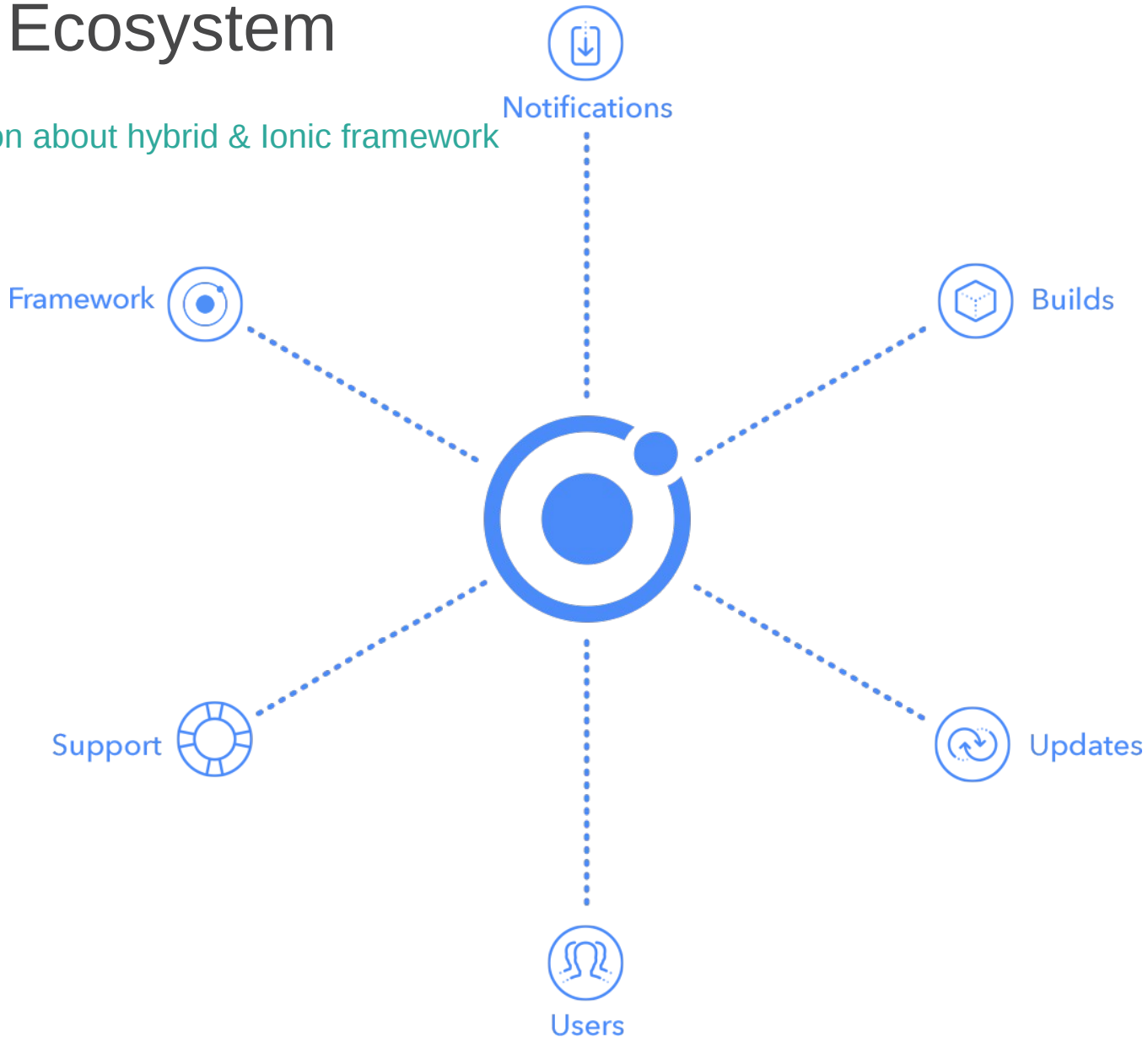
Ionic Ecosystem

- http://ionic.io/products/creator
  - Is a drag-&-drop prototyping tool for creating great apps using Ionic, with just a click of the mouse.
- http://lab.ionic.io/
  - Lab is a desktop app for Mac, Windows, and Linux, that makes it a joy to get started with Ionic.
- http://view.ionic.io/
  - Ionic View makes it easy to share your Ionic and Cordova apps with clients and testers around the world, all without ever going through the App Store.

Part 2 – Basic Ionic

Install and start first project

- ## What need to install:
  - Node-JS (prefer install via NVM)
  - Apache Cordova
    - *npm install -g cordova*
  - Ionic
    - *npm install -g ionic*
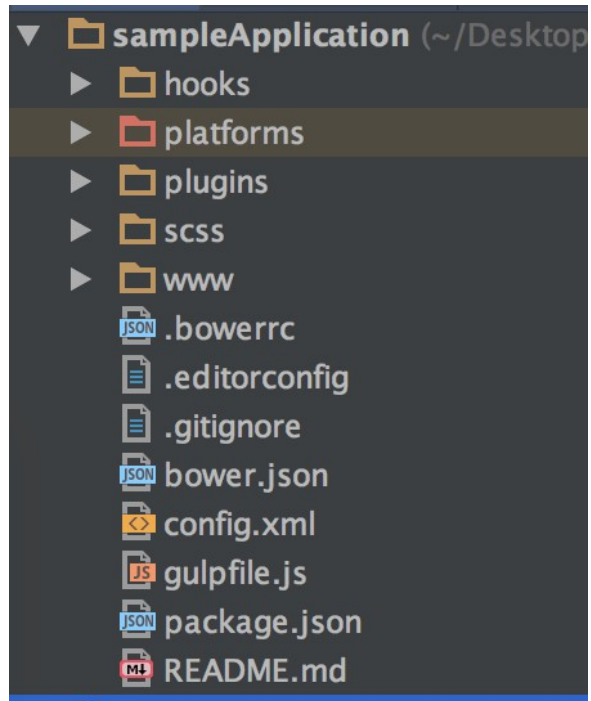
Ionic CLI

- Start project
  - *Ionic start –a <app_name> <newApp> <template>*
  - *<template>* can either come from a named template, a Github repo, a Codepen url, or a local directory.
- Run with local server
  - *ionic serve*
- Add platform iOs or Android:
  - *ionic platform add [ios/android]*
- Prepare iOs and Android
  - *ionic platform [ios/android] prepare*
- *Build iOS and Android*
  - *ionic platform [ios/android] build*
- *Run on device:*
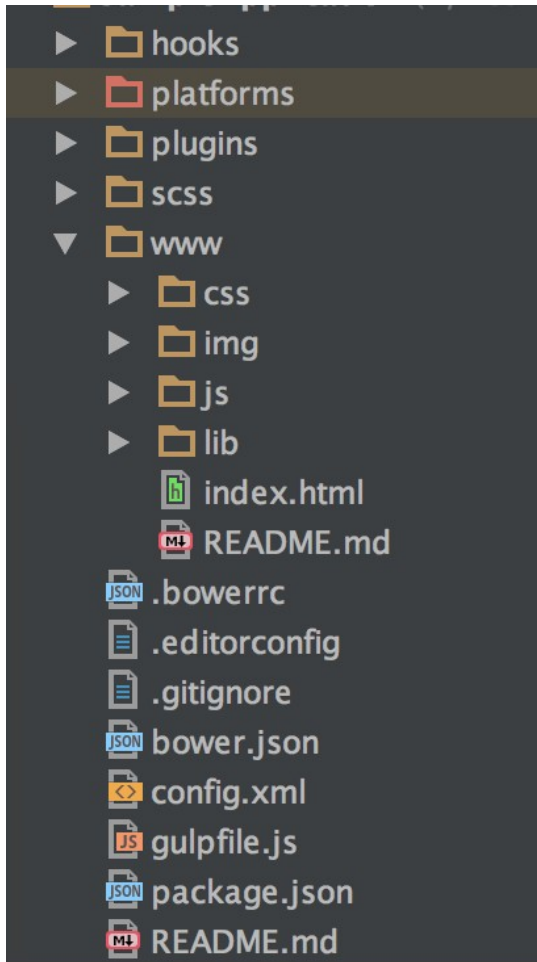 - *ionic run [ios/android] --device*

Directory Source code

```
▼  📁 sampleApplication (~/Desktop
  ▶  📁 hooks
  ▶  📁 platforms
  ▶  📁 plugins
  ▶  📁 scss
  ▶  📁 www
     📄 .bowerrc
     📄 .editorconfig
     📄 .gitignore
     📄 bower.json
     📄 config.xml
     📄 gulpfile.js
     📄 package.json
     📄 README.md
```

- platforms: directory contains your iOS and Android projects.
- plugins: are where Cordova stores the plugins that you add to your project
- scss: for your app's SASS file
- www: where your app is developed and where you'll spend most of your time when building an Ionic app.
- config.xml: config base to platform
- bower.json: bower config file
- gulpfile.js: build scss script.

GEEK UP

Directory Source code

Introduction about hybrid & Ionic framework

```
▶  📁 hooks
▶  📁 platforms
▶  📁 plugins
▶  📁 scss
▼  📁 www
   ▶  📁 css
   ▶  📁 img
   ▶  📁 js
   ▶  📁 lib
      📄 index.html
      📄 README.md
   📄 .bowerrc
   📄 .editorconfig
   📄 .gitignore
   📄 bower.json
   📄 config.xml
   📄 gulpfile.js
   📄 package.json
   📄 README.md
```

- css: all css file
- img: image file
- Js: angular source code
- lib: bower install file
- Index.html: starting file

GEEK UP

Run app

Introduction about hybrid & Ionic framework

ionic.bundle.js

- ionic.js
- angular.js
- angular-animate.js
- angular-sanitize.js
- angular-ui-router.js
- ionic-angular.js

GEEK UP

View life cycle

- $ionicView.loaded

- $ionicView.beforeEnter
- $ionicView.enter
- $ionicView.afterEnter

- $ionicView.beforeLeave
- $ionicView.leave
- $ionicView.afterLeave

- $ionicView.unloaded

GEEK UP

View life cycle

LifeCycle Event Usage

Below is an example of how to listen to life cycle events and access state parameter data

```
$scope.$on("$ionicView.beforeEnter", function(event, data){
    // handle event
    console.log("State Params: ", data.stateParams);
});

$scope.$on("$ionicView.enter", function(event, data){
    // handle event
    console.log("State Params: ", data.stateParams);
});

$scope.$on("$ionicView.afterEnter", function(event, data){
    // handle event
    console.log("State Params: ", data.stateParams);
});
```

Routing

Introduction about hybrid & Ionic framework

```javascript
// Learn more here: https://github.com/angular-ui/ui-router
// Set up the various states which the app can be in.
// Each state's controller can be found in controllers.js
$stateProvider
// Each tab has its own nav history stack:
    .state('board', {
        url: '/board',
        controller: 'BoardCtrl',
        templateUrl: 'templates/board.html'
    })
    .state('login', {
        url: '/login',
        controller: 'LoginCtrl',
        templateUrl: 'templates/account/login.html'
    })
    // setup an abstract state for the tabs directive
    .state('tab', {
        url: '/tab',
        templateUrl: 'templates/tabs.html',
        controller: "TabCtrl"
    })
    // Each tab has its own nav history stack:
    .state('tab.schedule', {
        url: '/schedule',
        views: {
            'tab-schedule': {
                templateUrl: 'templates/schedule/tab-schedule.html',
                controller: 'ScheduleCtrl'
            }
        }
    })
    .state('tab.class-detail', {
        url: '/schedule/class/:centerId/:classId/:dateUnix',
        views: {
            'tab-schedule': {
                templateUrl: 'templates/schedule/class-detail.html',
                controller: 'ClassDetailCtrl'
            }
        }
    })
```
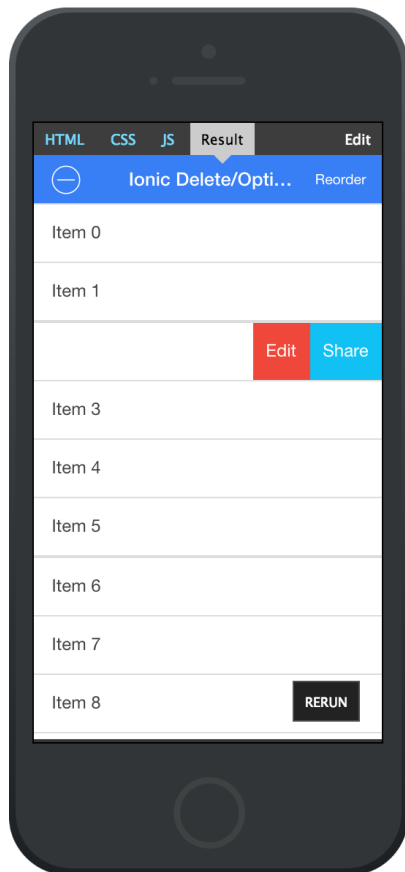
GEEK UP

UI Component

- List
- Navigation
- Tabs
- Side menu
- Slide box
- Action Sheet
- Pull-to-refresh & Infinite loading

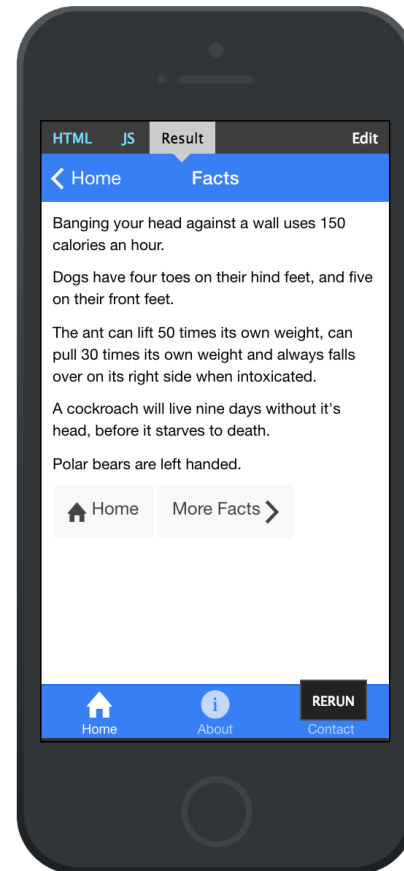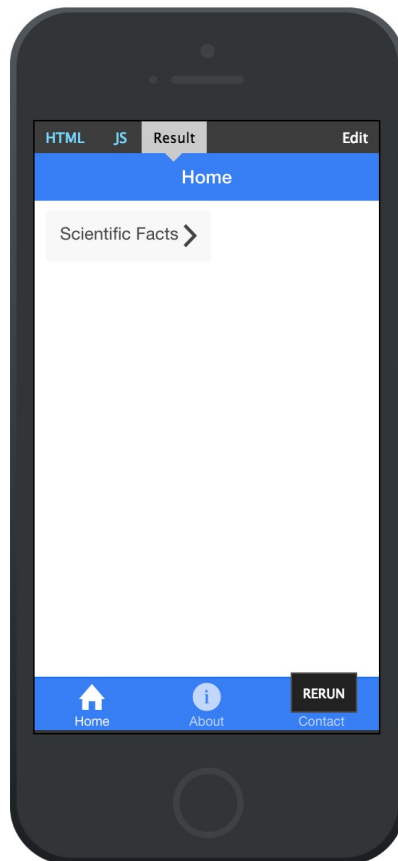# UI Component - List

Introduction about hybrid & Ionic framework

- http://ionicframework.com/docs/api/directive/ionList/

UI Component - Navigation

Introduction about hybrid & Ionic framework

- http://ionicframework.com/docs/api/directive/ionNavView/

UI Component - Tabs

Introduction about hybrid & Ionic framework

- http://ionicframework.com/docs/api/directive/ionNavView/
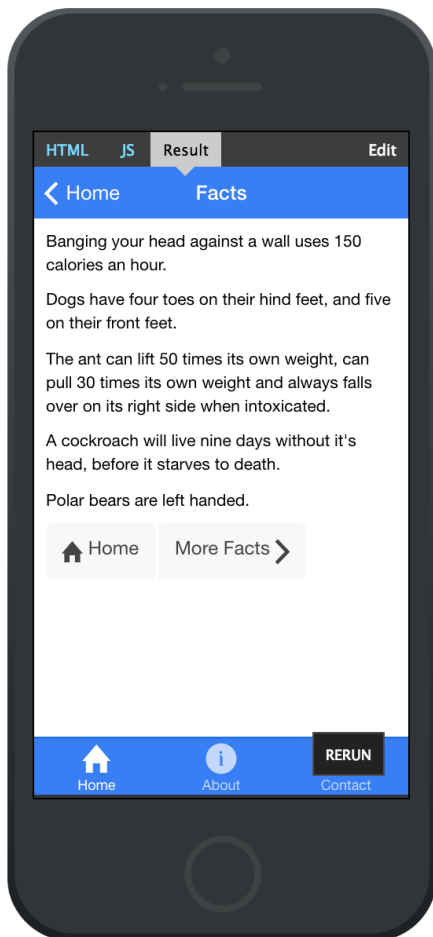
UI Component - Side menu

- http://ionicframework.com/docs/api/directive/ionSideMenus/
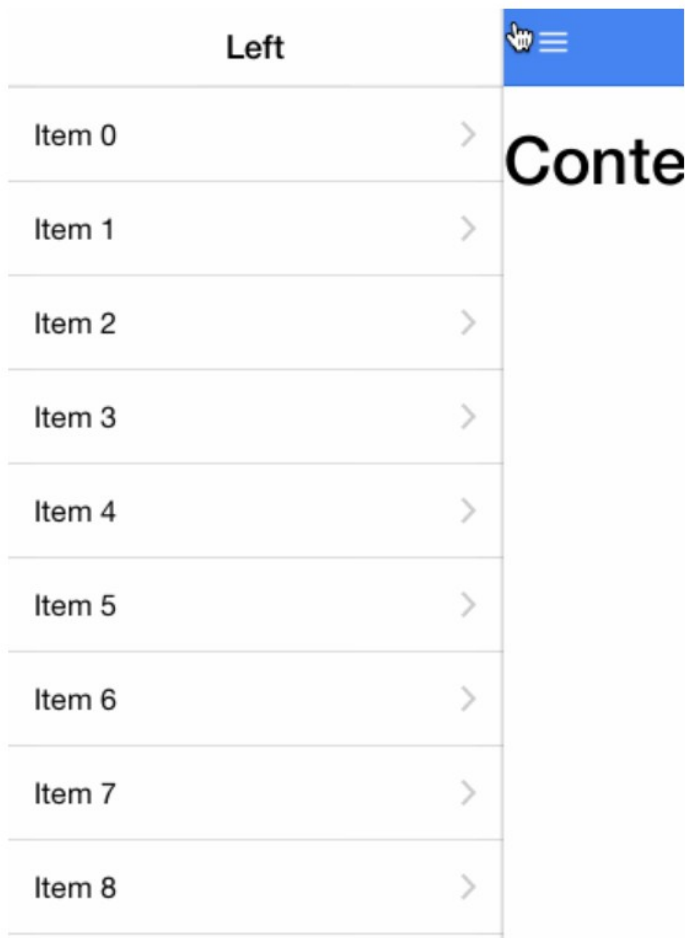
UI Component - Slide box

Introduction about hybrid & Ionic framework

- http://ionicframework.com/docs/api/directive/ionSlideBox/
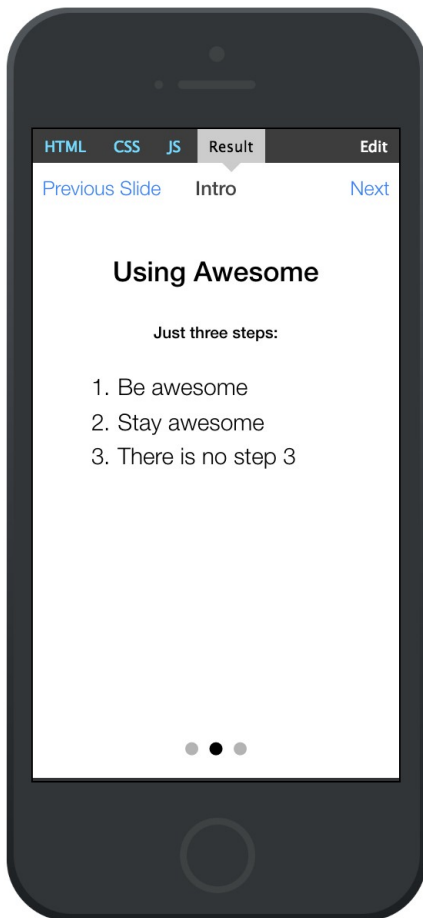
UI Component - Action Sheet

Introduction about hybrid & Ionic framework

- http://ionicframework.com/docs/api/service/$ionicActionSheet/
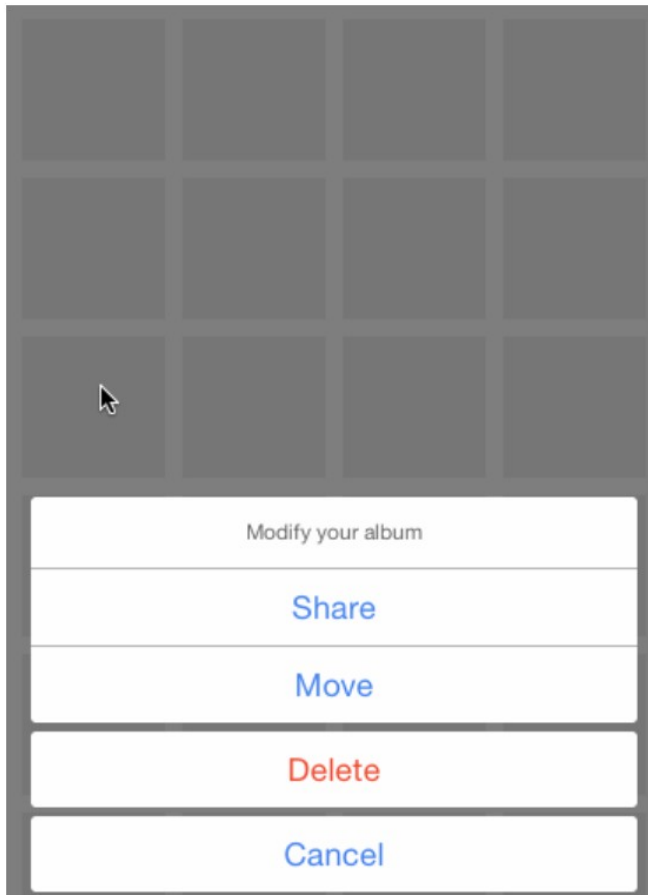
Pull-to-refresh

- http://ionicframework.com/docs/api/directive/ionRefresher/
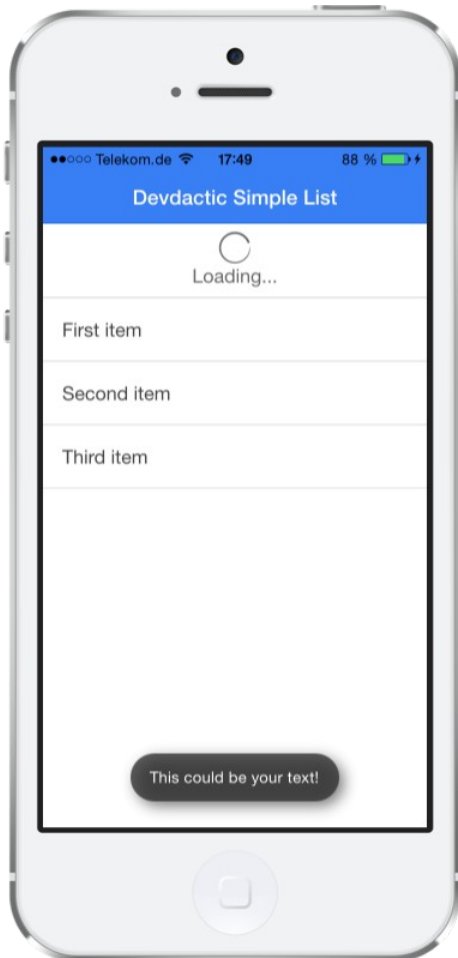
Demo

- ## TODO List application

- ## Source code: https://github.com/voxuanthinh/todolist-ionic
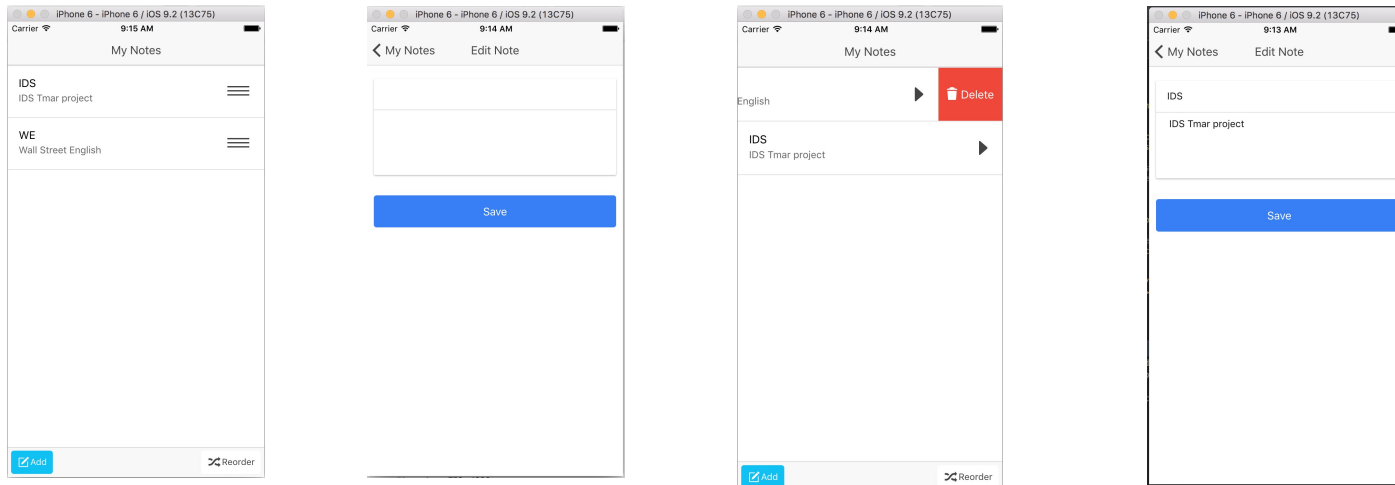
Question & Answers

Introduction about hybrid & Ionic framework

References

- Ionic present slide: http://ionicframework.com/present-ionic/slides/#/6
- Hybird vs native mobile: http://www.ymedialabs.com/hybrid-vs-native-mobile-apps-the-answer-is-clear/
- Ionic Framework: http://ionicframework.com/docs/

Thank you

Introduction about hybrid & Ionic framework