



Mobile app development

Part 1: Rapid Prototyping with Ionic



Muzaffar

Technical trainer.

Worked with mobile industry since 2011

Different industry: M-Commerce, Newsfeed, Media Broadcasting, Food Delivery , Airline and Loyalty.



Positionnez le code
barre du produit dans
le cadre ci-dessous

Recherche Détails du produit



Ducros
Curry en poudre

1.86 €
le flacon de 42g

DESCRIPTION INGREDIENTS INFO NUTRI.

Curcuma, graine de coriandre, fenugrec, amidon de blé, graine de moutarde, oignon déshydraté, cardamome, sel (4%), gingembre, anis, fenouil, girofille, ail, épices, plantes aromatiques.

Quantité 1

+ Ajouter à la liste

Annuler

Magasins

Recherche Carte Caméra



A map of Lille, France, highlighting the locations of several Carrefour supermarkets. The map shows major roads like the A25, E42, and N358, as well as local streets and landmarks such as the Vieux Lille, Lille Centre, and Lille Fives. Carrefour logos are placed on the map at their respective store locations.

-  PKR, PAS capai kesepakatan isu pertindihan kerusi
-  Masa untuk Kedah kembali kepada BN - Muhyiddin 40 saats lalu
-  M. United juara, #GGMU trending di Twitter
-  "Ramai rakyat Selangor O.K.U." 34 minits lalu
-  Harapan penduduk di Batu Sapi 5 jams lalu
-  Umno pecat 58, termasuk Kamilia, Mohd Shariff 18 jams lalu

Bahasa Malaysia

Terkini

Trending

Pilihan Tertinggi

Nasional

Bisnes

Sukan

Hiburan

Dunia

Blogs

Cuaca

Tonton sekarang

News

Kolumnis

Video

Foto

Program



NICOLAS MADURO
President Venezuela

Natasha Hudson selamat melahirkan bayi perempuan

2:17 -0:33

Langsung Awani Live

Jadual

 Buletin Awani 12:45

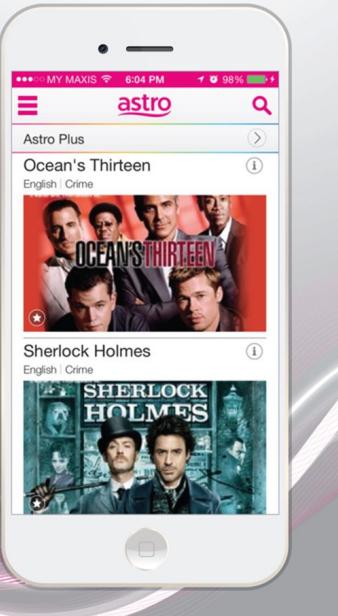
 Nadi Arena 13:00

4

First few projects in Astro

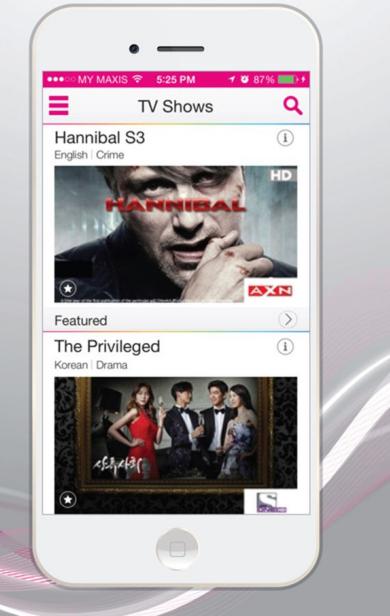
Sign up for FREE!

For more features, link your Astro account. You can also buy a subscription/ pay-per-view.



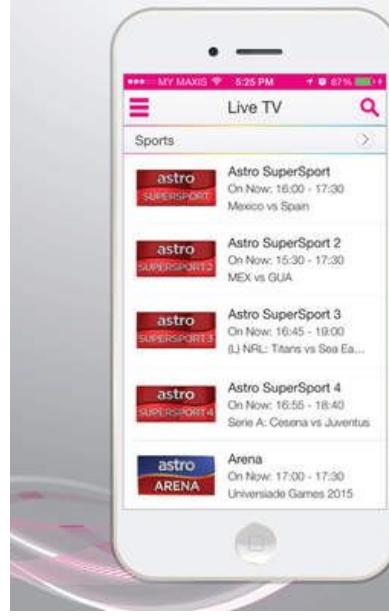
TV SHOWS

The latest episode is just a tap away. You can even binge-watch entire seasons in one seating – from the first episode to the finale!



LIVE TV

From sports to current affairs and more, catch every history-making moment as it happens.



Main project in Astro

Product management work.

Carrier ⌂ 6:47 PM

Enter Postal Code

 ROOM SERVICE
a food runner company

Order Food Now

Postal Code, e.g., 199597

Find Restaurants

 Use My Location

Restaurants Cart My Account Call Us

Carrier ⌂ 6:47 PM

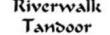
Location Restaurants Filter

non-vegetarian cuisine. Brinda's is Indian...

 California Pizza Kitchen
American
They're known for their thin crust pizzas & innovative toppings, need we say more?

 Tony Roma's
American
Renowned for their succulent ribs, tony roma promises the best slow smoked ribs the city can offer.

 Hard Rock Café
American
Famous for their classic american burgers, you can't go wrong with this one.

 Riverwalk Tandoor
Indian
Riverwalk tandoor started out as a catering business and has expanded over the years. We have been serving authen...

 Al Qasr
Lebanese and Middle Eastern Cuisine
Be one with exotic middle eastern cuisine as you bite in to juicy shawarmas.

Kinara

Restaurants Cart My Account Call Us

Carrier ⌂ 6:51 PM

My Order Pre-order

California Pizza Kitchen

Spicy Chicken Ti... - 1 + 14.20

Two In A Bowl - 1 + 8.70

Total : 22.90
Service Fee : 2.29
Delivery Fee : 9.90
Tax : 2.56

Coupon Code Enter

Grand Total : \$39.07

extra chili! please call before knocking

Go To Checkout

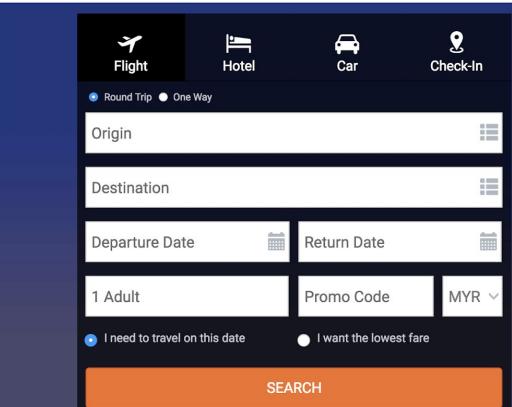
Restaurants Cart 2 My Account Call Us

Development Manager

The image displays three screenshots of the Firefly mobile application, illustrating its user interface and features:

- New and elegant user interface:** Shows the home screen with a promotional banner for a flight deal from RM 55, followed by navigation icons for Book Flight, Manage Flight, Mobile Check-in, and Boarding Pass.
- Simplified booking flow:** Shows the "BOOK FLIGHT" screen where users can select their departure and arrival locations (Subang and Penang), travel dates (01/06/2016 and 05/06/2016), and the number of passengers (Adult: 1, Infant: 0).
- Access your boarding pass on the go:** Shows the "BOARDING PASS" screen displaying a QR code and flight details: MR Lal Nordin, Depart SUBANG, Arrive PENANG, Flight Date 1 JUN 2016, Boarding Time 06:30 AM, Flight No. FY2990.

Another work as Development Manager



The screenshot shows the Firefly Beyond Convenience website's flight search interface. It includes tabs for Flight, Hotel, Car, and Check-In. The flight search form is set to 'Round Trip' and shows fields for Origin, Destination, Departure Date, Return Date, number of adults (1 Adult), and a Promo Code dropdown set to MYR. There are two radio buttons: 'I need to travel on this date' (selected) and 'I want the lowest fare'. A large orange 'SEARCH' button is at the bottom.



A promotional banner for 'Daily Flight' effective from 25 March 2018. The banner features a sunset over a tropical beach with palm trees. Text on the banner reads: 'Daily Flight', 'EFFECTIVE 25 MARCH 2018', 'It's better to see something once than to hear about it a thousand times.', and a table of flight details:

| DESTINATION | DEPART | ARRIVE |
|-----------------|--------|--------|
| Penang - Phuket | *14:05 | *14:20 |
| Phuket - Penang | *15:20 | *17:35 |

*Local time



Firefly Holiday
Great value with Firefly Holiday!
[Find Out More](#)



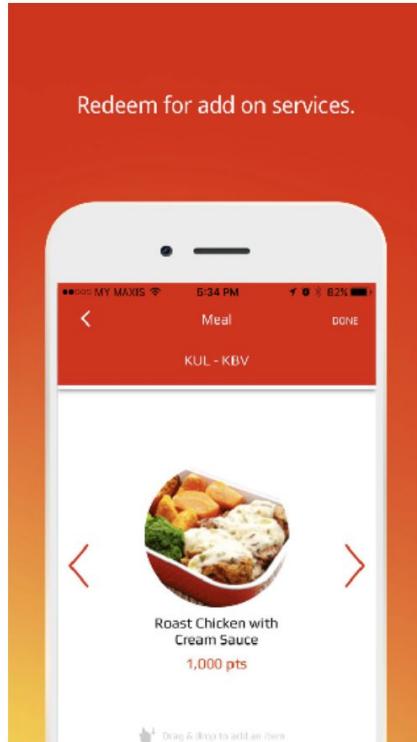
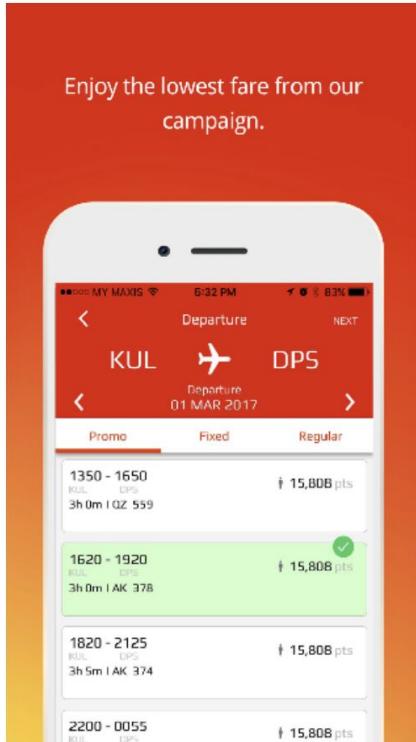
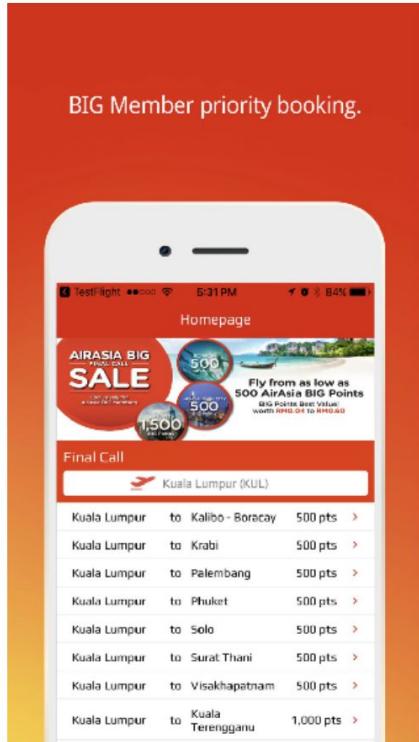


Car insurance renewal online
Get 10% rebate and enjoy FREE road tax delivery.
[Find Out More](#)

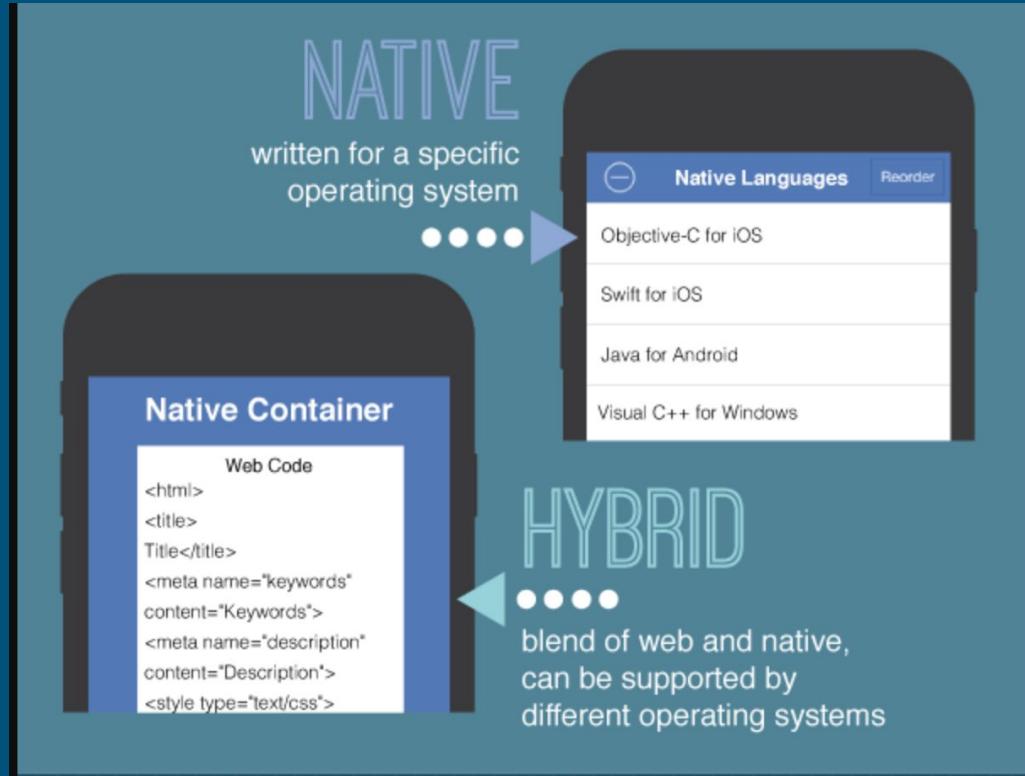




Last project as Development Manager



Hybrid vs Native



Mobile App Types

- **Native**
 - Programmed using Swift on the iPhone or using Java/Kotlin on Android devices.
- **Hybrid**
 - Mix between these two types of mobile applications.
 - Normally based on web programming language, eg: HTML, CSS, Javascript.
 - Built once to be run on Android and iOS.
- **Web Apps / Progressive Web Apps.**
 - Web based.
Runs in the phone's browser.
 - Can have native features based on HTML5

Why Native is Better?

Make use of all the phone's features such as the camera, geolocation, user's address book, etc.

Do not need to be connected to the Internet to be used.

Specific to the mobile device it is run on since it uses the features of that specific device.

Big support of the community and distributor.



This repository

Search

Pull requests Issues Marketplace Explore



GO ▾

Ionic Team / cordova-plugin-ios-keychain

forked from shazron/KeychainPlugin

Watch ▾ 10

Star 40

Fork 88

Code

Issues 4

Pull requests 1

Projects 0

Wiki

Insights

Apache Cordova (PhoneGap) plugin

57 commits

2 branches

2 releases

11 contributors

Branch: master ▾

New pull request

Create new file

Upload files

Find file

Clone or download ▾

This branch is 26 commits ahead, 4 commits behind shazron:master.

Pull request Compare

mlynch Merge pull request #25 from danielsogl/patch-1 ...

Latest commit 036aa60 on 6 Mar

example

Update index.js

a year ago

src/ios

Fixing security issue

2 months ago

www

fix(set): make useTouchID truly optional

a year ago

.gitignore

Updated to use Cordova 2.2.0 and made it plugininstallable

6 years ago

LICENSE

Updated to use Cordova 2.2.0 and made it plugininstallable

6 years ago

README.md

refactor(): rename driftyco

11 months ago

package.json

Adding package.json for Cordova 7 compat

7 months ago

plugin.xml

Moving files

2 years ago

README.md

Why Hybrid is Better?

One code for both platform, iOS and Android.

Support from big company's technology:
Eg: Google and Facebook.

Shorten developer time and initial cost.
(Might be high on maintenance cost)

Suitable for prototyping, MVP or startup development.



is all you'll need to build
POWERFUL HYBRID APPS
like these



LinkedIn



Sworkit
Built with Ionic



ChefSteps
Built with Ionic



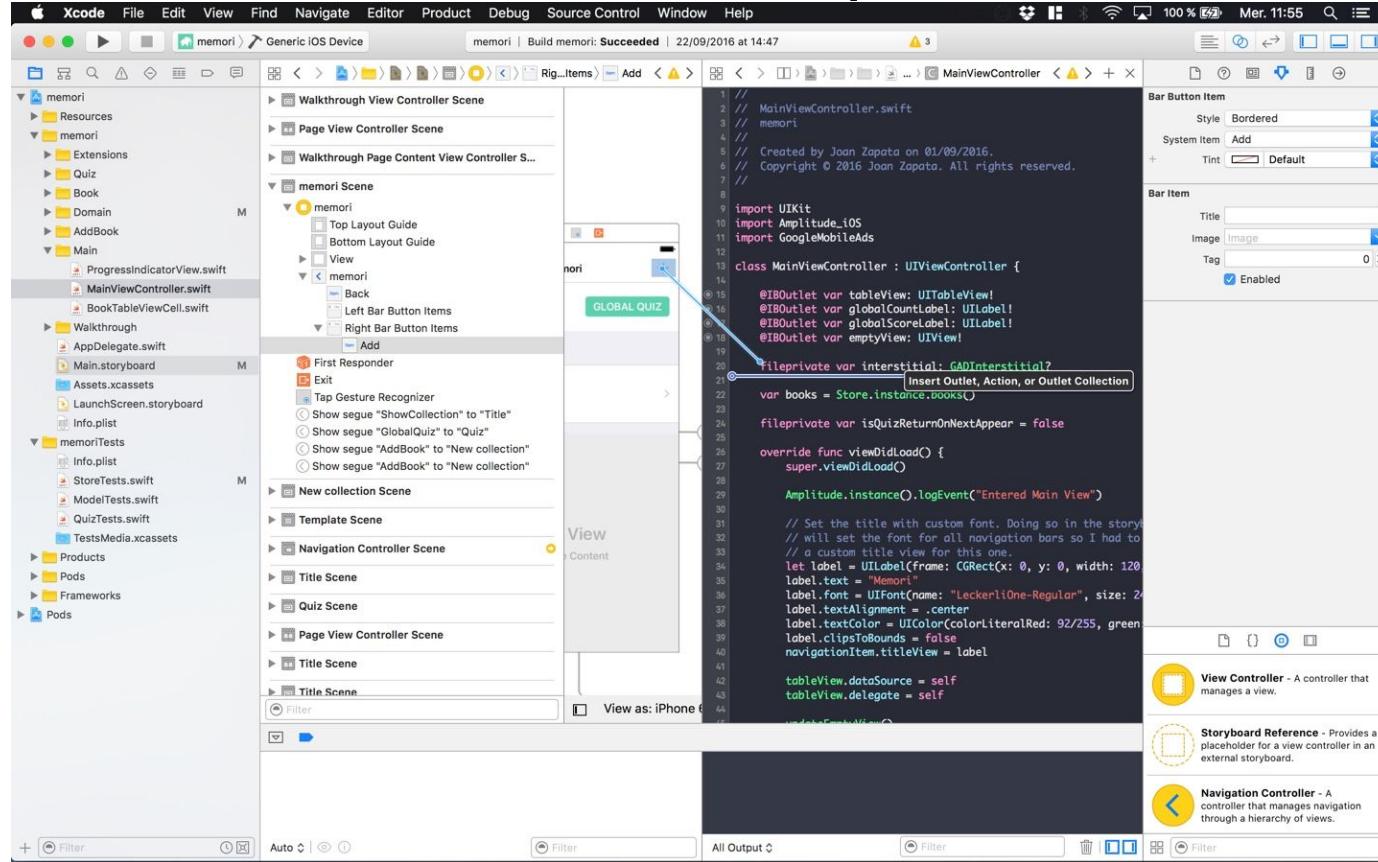
Netflix

If you're concerned about the performance of
Hybrid Apps, make sure to find the
most suitable framework for your project

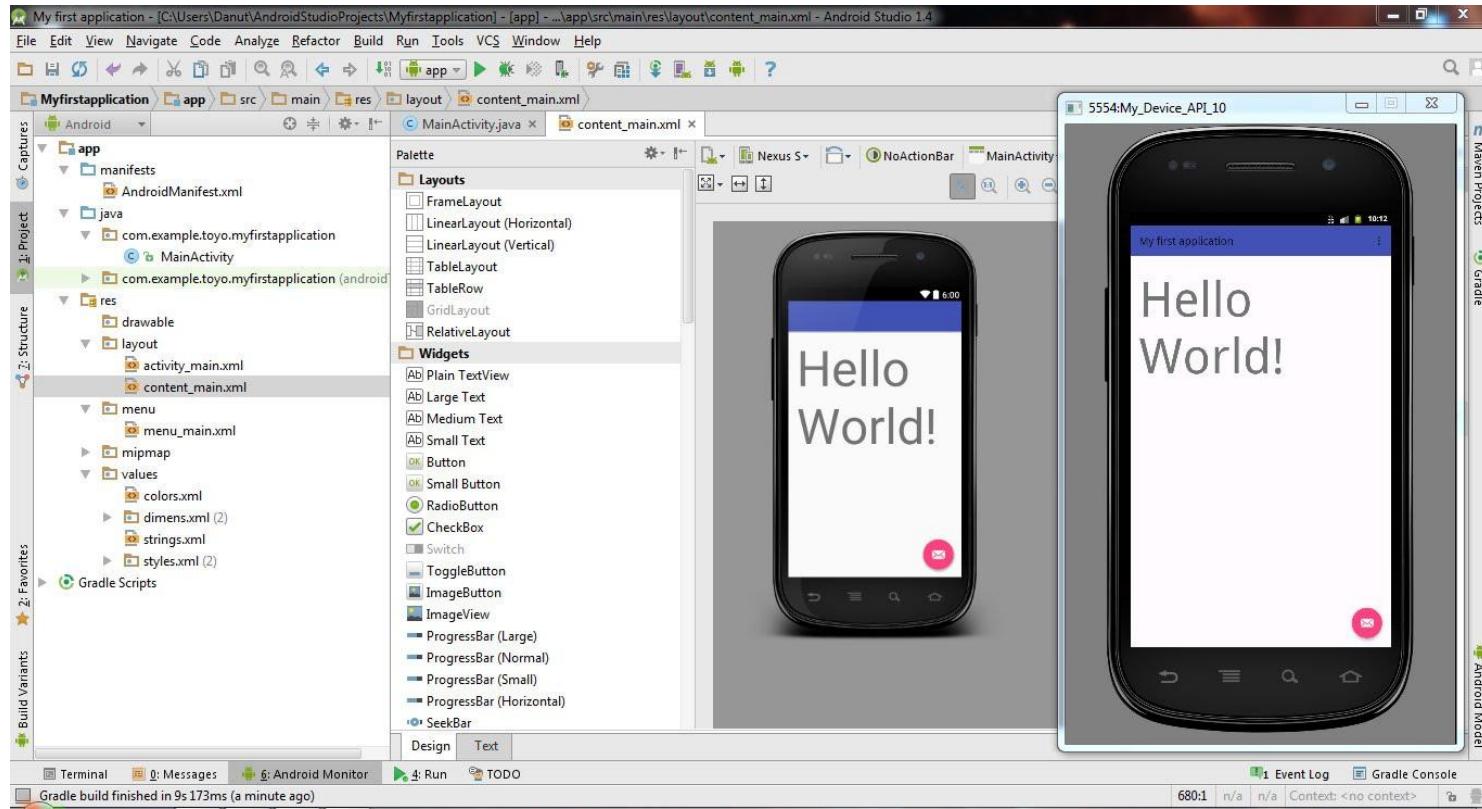
Mobile App Development: Environments

- Most platforms have an **SDK** that can be downloaded and built against.
- Every platform has an **emulator** that can be used to test the apps.
- Most emulators are configurable to match a variety of mobile devices.
 - Various screen sizes, memory limitations, tablets, etc.
 - In practice, emulators are quite limited.
- **IDE: Integrated Development Environment** that provides tools to allow a developer to write, test and deploy apps into the target platform environment.

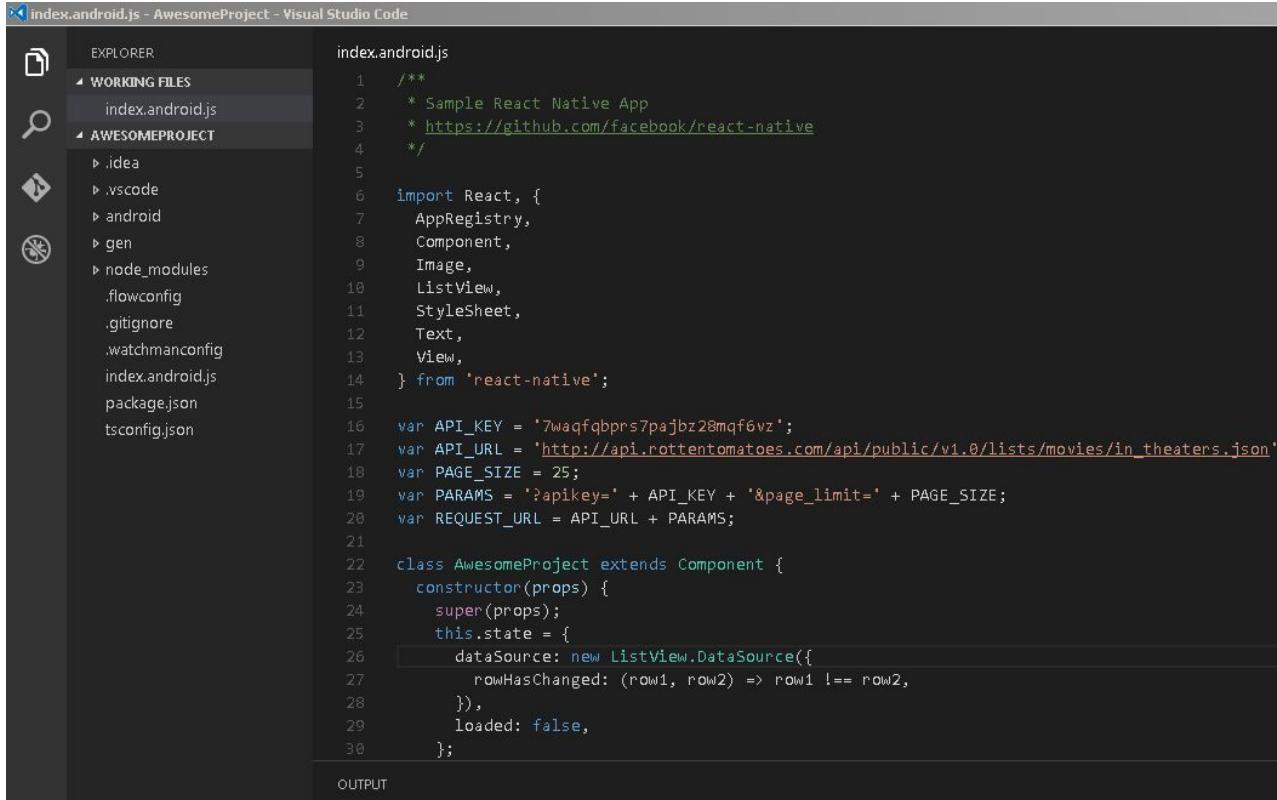
XCode IDE for iOS Development



Android Studio and Android Emulator



Visual Code for Hybrid Development



The screenshot shows the Visual Studio Code interface with the following details:

- Title Bar:** index.android.js - AwesomeProject - Visual Studio Code
- Left Sidebar (EXPLORER):** Shows the project structure:
 - WORKING FILES: index.android.js
 - AWESOMEPROJECT: .idea, .vscode, android, gen, node_modules, .flowconfig, .gitignore, .watchmanconfig, index.android.js, package.json, tsconfig.json
- Code Editor:** The file index.android.js is open, displaying code for a React Native application. The code imports various components from 'react-native' and defines a class named AwesomeProject extending Component. It includes logic for fetching movie data from an API and setting up a ListView.DataSource.
- Bottom Status Bar:** OUTPUT

```
index.android.js
1  /**
2   * Sample React Native App
3   * https://github.com/facebook/react-native
4   */
5
6  import React, {
7    AppRegistry,
8    Component,
9    Image,
10   ListView,
11   StyleSheet,
12   Text,
13   View,
14 } from 'react-native';
15
16 var API_KEY = '7waqfqbpns7pajbz28mqf6vz';
17 var API_URL = 'http://apirottentomatoes.com/api/public/v1.0/lists/movies/in_theaters.json';
18 var PAGE_SIZE = 25;
19 var PARAMS = '?apikey=' + API_KEY + '&page_limit=' + PAGE_SIZE;
20 var REQUEST_URL = API_URL + PARAMS;
21
22 class AwesomeProject extends Component {
23   constructor(props) {
24     super(props);
25     this.state = {
26       dataSource: new ListView.DataSource({
27         rowHasChanged: (row1, row2) => row1 !== row2,
28       }),
29       loaded: false,
30     };
31
32   }
33
34   render() {
35     return (
36       <View style={styles.container}>
37         <Text>Awesome Project</Text>
38         <ListView
39           dataSource={this.state.dataSource}
40           renderRow={this._renderRow}
41         />
42       </View>
43     );
44   }
45
46   _renderRow(rowData) {
47     return (
48       <View style={styles.item}>
49         <Image source={{uri: rowData.image}} style={styles.image}/>
50         <Text>{rowData.title}</Text>
51       </View>
52     );
53   }
54
55   componentDidMount() {
56     AppRegistry.registerComponent('AwesomeProject', () => AwesomeProject);
57   }
58
59   static navigationOptions = {
60     title: 'Awesome Project',
61   };
62
63   render() {
64     return null;
65   }
66 }
67
68 AppRegistry.registerComponent('AwesomeProject', () => AwesomeProject);
```

Benefits of Ionic 3

- Smaller learning curve, easy to learn compared to Native development.
- Have a development flow that is closer to native app development.
- Get “iOS”, “Android” and “Windows” looks and feel for free.
- Easier to customize.

What is Ionic?

- Twitter Bootstrap for hybrid mobile apps.
- Provide directives/components that mimic native features.
- Provide access to native device features via Cordova.
- Created in 2013.
- Over 1,000,000 apps built with it.
- Have apps that have top rating in App Store.

Companies hiring developer with Ionic expertise



ALPS kiple



Ionic Dependencies



Installing Ionic

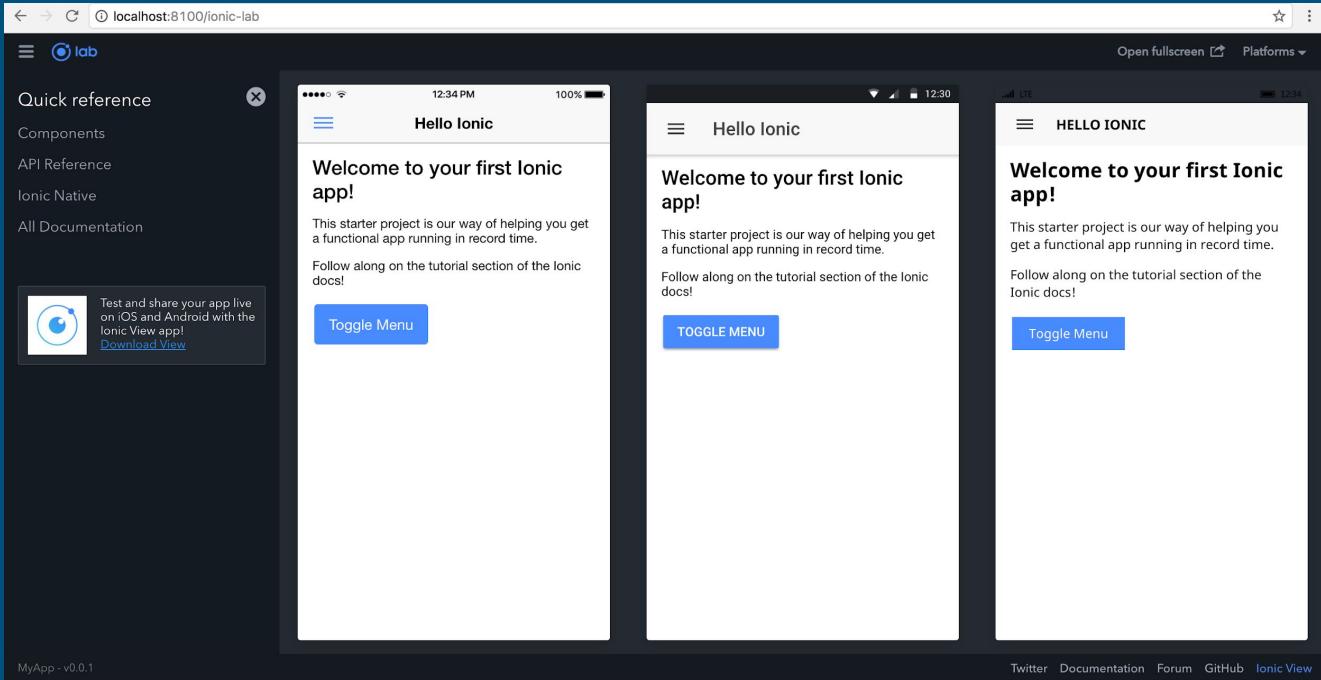
Ionic require Node.JS for installation. You should have installed Node.JS already. Open your terminal console, enter the following command:

npm install -g cordova ionic

Ionic lab

- Ionic Lab allows you to view the created application into your browser and how it is .
- You will just need to insert the following URL: <http://localhost:8100/ionic-lab>.
- Add the different device Operating System to see how the app would appear on different devices.

Ionic Lab



<http://localhost:8100/ionic-lab>

Hello Ionic!

- 1) Run ionic start command as listed on the previous page, start to run :

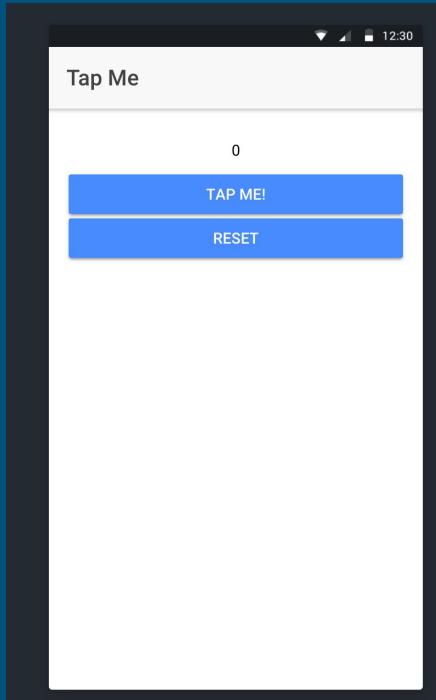
ionic start helloworld blank

- 2) Type **ionic serve** in Ionic CLI.
- 3) Your browser will be opened automatically and created and the file appears.

Ionic templates

| Template name | Purpose | Command |
|---------------|--|--------------------------------|
| blank | A blank starter project for Ionic | ionic start MyApp blank |
| complex-list | A complex list starter template | ionic start MyApp complex-list |
| maps | An Ionic starter project using Google Maps and a side menu | ionic start MyApp maps |
| salesforce | A starter project for Ionic and Salesforce | ionic start MyApp salesforce |
| sidemenu | A starting project for Ionic using a side menu with navigation in the content area | ionic start MyApp sidemenu |
| tabs | A starting project for Ionic using a simple tabbed interface | ionic start MyApp tabs |

Codelab 1: Counter app



You will create a counter app that will have a variable

1) Create new variable in home.ts

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

@Component({
  selector: 'page-home',
  templateUrl: 'home.html'
})
export class HomePage {
|public tapCounter: number = 0;
  constructor(public navCtrl: NavController) {

  }

}
```

2) Change the ion-title and add variable in home.html.

```
<ion-header>
  <ion-navbar>
    <ion-title>
      Tap Me
    </ion-title>
  </ion-navbar>
</ion-header>

<ion-content padding>
  <div>
    <p align="center">{{tapCounter}}</p>
  </div>
</ion-content>
```

3) Add button components, click action : buttonTapped and reset.

```
<ion-header>
  <ion-navbar>
    <ion-title>
      Tap Me
    </ion-title>
  </ion-navbar>
</ion-header>

<ion-content padding>
  <div>
    <p align="center">{{tapCounter}}</p>
  </div>
  <ion-buttons>
    <button ion-button block (click)="buttonTapped()">Tap me!</button>
    <button ion-button block (click)="reset()">Reset</button>
  </ion-buttons>
</ion-content>
```

4) Create and implement buttonTapped method.

```
1 import { Component } from '@angular/core';
2 import { NavController } from 'ionic-angular';
3
4 @Component({
5   selector: 'page-home',
6   templateUrl: 'home.html'
7 })
8 export class HomePage {
9   public tapCounter: number = 0;
10  constructor(public navCtrl: NavController) {
11  }
12  }
13  buttonTapped() {
14    this.tapCounter++;
15  }
16  |
17
18 }
19
```

5) Try It: Implement Reset Button

- Try to implement the “Reset” Button.
- It should reset the counter to 0 once it is pressed.

Try It: More features

Replace the “Tap It” button with “Increase” and “Decrease” function.

- “Increase” will increase the counter
- “Decrease” the counter.
- “Reset” - reset the number to 0

Ionic Components

- Ionic apps are made of high-level building blocks called components.
- Components allow you to quickly construct an interface for your app. Ionic comes with a number of components, including modals, popups, and cards.
- Have a look at Ionic Documentation to see all available components that can be used inside your application.

HOME

INTRO

COMPONENTS

Overview

Action Sheets

Alerts

Badges

Buttons

Cards

Checkbox

DateTime

FABs

Gestures

Grid

Icons

Inputs

Lists

Components

Ionic apps are made of high-level building blocks called components. Components allow you to quickly construct an interface for your app. Ionic comes with a number of components, including modals, popups, and cards. Check out the examples below to see what each component looks like and to learn how to use each one. Once you're familiar with the basics, head over to the [API docs](#) for ideas on how to customize each component.

[Improve this Doc](#)

Action Sheets

Action Sheets slide up from the bottom edge of the device screen, and display a set of options with the ability to confirm or cancel an action. Action Sheets can sometimes be used as an alternative to menus, however, they **should not** be used for navigation.



Important Components

- ActionSheet
- Button
- SearchBar
- Alert
- Modal
- IonRefresher
- Keyboard

Codelab 2: Creating an Alert

From the previous exercise, add a new button on the new page.

Create a “Click” handler on the button.

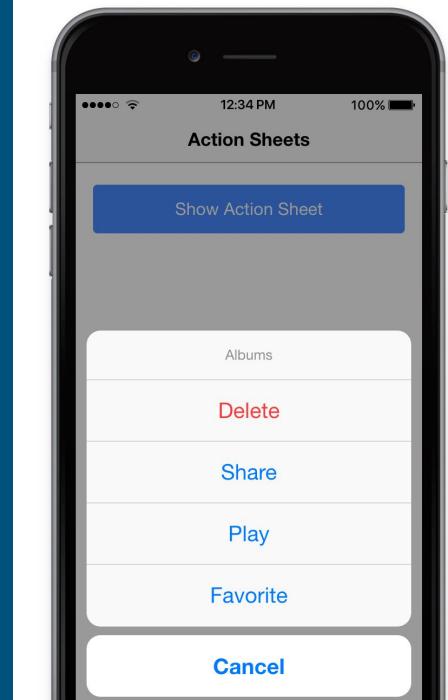
We will add Alert into the application, three steps to remember:

- Import alertController into the file.
- Inject alertController inside the constructor.
- Implement the code (get from Documentation)

Identify and implement the three steps mentioned previously.

```
1 import { Component } from '@angular/core';
2 import { AlertController, NavController, NavParams } from 'ionic-angular';
3
4 /**
5  * Generated class for the NewPage page.
6  *
7  * See http://ionicframework.com/docs/components/#navigation for more info
8  * on Ionic pages and navigation.
9  */
10
11 @Component({
12   selector: 'page-new',
13   templateUrl: 'new.html',
14 })
15 export class NewPage {
16
17   constructor(public navCtrl: NavController, public navParams: NavParams, public alertController: AlertController) {
18   }
19
20   ionViewDidLoad() {
21     console.log('ionViewDidLoad NewPage');
22   }
23   buttonPressed() {
24     let alert = this.alertController.create({
25       title: 'Example',
26       subTitle: 'Example subtitle',
27       buttons: ['OK']
28     });
29     alert.present();
30   }
31 }
32
33
```

Try It: Implement an ActionSheet



Recap : Ionic CLI command

- Create a new project using blank template
- Run it on browser
- Run it on browser while showing different OS (Android and iOS)
- Open your project in IDE (eg: Sublime) to continue with the following exercise.

Ionic generate command

Ionic CLI provides an “ionic generate” command that easily helps to move a basic app to full-featured app.

It saves time to generate app artifacts while keeping your app structure under control.

Use **ionic generate --list** command to see all the available options.

```
[wans-MacBook-Pro:WeatherApp wanmuz$ ionic generate --list
? What would you like to generate: (Use arrow keys)
> component
directive
page
pipe
provider
tabs
```

Try It: Ionic Generate

You will be using Ionic generate command to create two pages, user-list and user-detail.

Use the command **ionic generate page <page name>**

Verify in Sublime on the newly created folder in app/pages that the new folders are created.

The screenshot shows a Mac OS X application window with a dark theme. On the left is a file tree view, and on the right is the code editor area.

File Tree (Left):

- resources
 - android
 - ios
 - icon.png
 - splash.png
- src
 - app
 - assets
 - pages
 - home
 - user-detail
 - user-detail.html
 - user-detail.module.ts
 - user-detail.scss
 - user-detail.ts
 - users-list
 - users-list.html
 - users-list.module.ts
 - users-list.scss
 - users-list.ts
 - theme
 - index.html
 - manifest.json
 - service-worker.js
 - www

Code Editor (Right):

```
app.module.ts
```

```
1 import { BrowserModule } from '@angular/platform-browser';
2 import { ErrorHandler, NgModule } from '@angular/core';
3 import { IonicApp, IonicErrorHandler, IonicModule } from 'ionic-angular';
4 import { SplashScreen } from '@ionic-native/splash-screen';
5 import { StatusBar } from '@ionic-native/status-bar';
6
7 import { MyApp } from './app.component';
8 import { HomePage } from '../pages/home/home';
9
10 @NgModule({
11   declarations: [
12     MyApp,
13     HomePage
14   ],
15   imports: [
16     BrowserModule,
17     IonicModule.forRoot(MyApp)
18   ],
19   bootstrap: [IonicApp],
20   entryComponents: [
21     MyApp,
22     HomePage
23   ],
24   providers: [
25     StatusBar,
26     SplashScreen,
27     {provide: ErrorHandler, useClass: IonicErrorHandler}
28   ]
29 })
30 export class AppModule {}
```

Modules

Like any modern development system (eg: NodeJS), Angular is modular.

You may segregate each function or logical units by module. This increases reusability of your code

You will export the module using following syntax:

```
import {HomePage} from './home/home'
```

Import List and Detail Modules.

- Every time a new page/component is added, you need to import it inside app.module.ts.
- Open **app.module.ts** and import the newly added Component into it. You should follow how it was done with HomePage, under entryComponents, declarations and import.

```
3 import { IonicApp, IonicErrorHandler, IonicModule } from 'ionic-angular';
4 import { SplashScreen } from '@ionic-native/splash-screen';
5 import { StatusBar } from '@ionic-native/status-bar';
6
7 import { MyApp } from './app.component';
8 import { UserDetailPage } from '../pages/user-detail/user-detail';
9 import { UsersListPage } from '../pages/users-list/users-list';
10
11 @NgModule({
12   declarations: [
13     MyApp,
14     UserDetailPage,
15     UsersListPage
16   ],
17   imports: [
18     BrowserModule,
19     IonicModule.forRoot(MyApp)
20   ],
21   bootstrap: [IonicApp],
22   entryComponents: [
23     MyApp,
24     UserDetailPage,
25     UsersListPage
26   ],
27   providers: [
28     StatusBar,
29     SplashScreen,
30     {provide: ErrorHandler, useClass: IonicErrorHandler}
31 ]
32 })
33 export class AppModule {}
```



Changing the root page.

You will change the root page to go directly into List Page instead “Home” page.
This is done on **app.component.ts** file.

```
1 import { Component } from '@angular/core';
2 import { Platform } from 'ionic-angular';
3 import { StatusBar } from '@ionic-native/status-bar';
4 import { SplashScreen } from '@ionic-native/splash-screen';
5
6 import { HomePage } from '../pages/home/home';
7 import { UsersListPage } from '../pages/users-list/users-list';
8
9 @Component({
10   templateUrl: 'app.html'
11 })
12 export class MyApp {
13   rootPage:any = UsersListPage;
14
15   constructor(platform: Platform, statusBar: StatusBar, splashScreen: SplashScreen) {
16     platform.ready().then(() => {
17       // Okay, so the platform is ready and our plugins are available.
18       // Here you can do any higher level native things you might need.
19       statusBar.styleDefault();
20       splashScreen.hide();
21     });
22   }
23 }
24
25 }
```

Ionic UI Revision

Using Ionic Documentation come out with the UI on following page.

Identify which elements need to be used.

Random User



Najib Razak
najibrazak@malaysia.com



Mahathir Mohamad
mahathir@malaysia.com

Revision: Ionic TypeScript Revision.

In Typescript file, create a variable `peoples` as an array that will contains user object with following info: name, email, phone number, image_url.

You may enter random information first.

Directives

Directives are the fundamental building blocks of AngularJS.

They either modify the layout structure or modify aspects of DOM (Document Object Model) elements.

There are three types of directives:

- structural directives
- attribute directives
- components.

Structural Directives : *ngFor and *ngIf

*ngFor is a repeat directive. It repeats a template once per item in a collection.

For example: `<div *ngFor="let item of items"></div>`

*ngIf adds or remove an element and its element based on a condition:

`<item-detail *ngIf="atruestmt"></item-detail>` .

In the following example, the item will only be shown if the atruestmt is true.

Interpolation

Interpolation is the form of data binding to show the parameter defined in TS file into html.

You will use {{ }} to show the data, eg:

```
<div>My name is {{name}}</div>
```

Here, AngularJS will replace the name with the corresponding property.

Try It: Using *ngFor and interpolation.

Using *ngFor, you will repeat the ion-item to show it depending on the number of user inside userList.

For each item, using Interpolation, show the value from the array into the **ion-item**.

Try It : Using *ngFor with Ion Item

```
<ion-content padding>
<ion-list>
  <ion-item *ngFor="let user of usersList" >
    <ion-thumbnail item-start>
      <img src="{{user.image}}>
    </ion-thumbnail>
    <h2>{{user.name}}</h2>
    <p>{{user.email}}</p>
  </ion-item>

</ion-list>
</ion-content>
```

Ionic Navigation

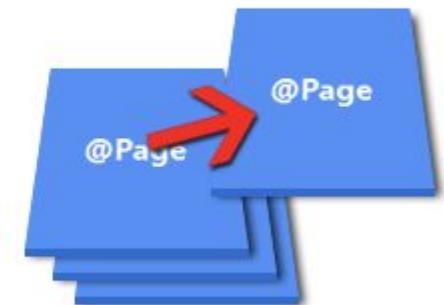
Navigation Stack



Push



Pop



Ionic Navigation

Each pages in Ionic is managed inside a Navigation Stack, via NavController.

Every time we add new page, it will remain inside a stack, and we push it to a new page.

Whenever we want to remove the page from the stack, we “pop” the stack to remove it from navigation controller.

We use NavParams, to pass data between two view/page.

Try It: Ionic Navigation

- 1) Create a new method to open a new page, call it itemTapped. The declaration can be found on next page.
- 2) Call this method when the ion-item is pressed (Revision)
- 3) On detail page, retrieve the user/item passed from the first page (refer to code on next two pages)
- 4) Show the item passed inside the new page using Interpolation.

```
21    }
22
23    ionViewDidLoad() {
24        console.log('ionViewDidLoad UsersListPage');
25    }
26    itemTapped(user){
27        this.navCtrl.push(UserDetailPage, {
28            user: user
29        });
30
31    }
32 }
```

```
@IonicPage()
@Component({
  selector: 'page-user-detail',
  templateUrl: 'user-detail.html',
})
export class UserDetailPage {

  userDetail = {}

  constructor(public navCtrl: NavController, public navParams: NavParams) {
    this.userDetail = navParams.get('user');
  }

  ionViewDidLoad() {
    console.log('ionViewDidLoad UserDetailPage');
  }
}
```

Expectation on Detail Page.

