

Hybrid App Development

...

with Ionic

Ionic: Underneath it all

Ionic is a front-end UI framework that handles all of the look and feel and UI interactions in your app.

To install an app on a mobile, it has to be native, or it can be wrapped in native wrapper.

Cordova is a tool that wraps our code in WebView.

Ionic also uses AngularJS for a lot of the core functionality of the framework.

And the whole thing used NodeJS, which is a command-line tool that can be run as a regular web server and lets one run JavaScript programs.

Preparing The Environment

First, we need to install NodeJS: <https://nodejs.org/en/>

This will allow us to install any plugin and package from Ionic and Cordova.

Once NodeJS is properly installed. We use the terminal to install any plugin and package from Ionic and Cordova.

```
sudo npm install -g cordova ionic
```

First Ionic App

Start a new app by typing only 4 words.

```
ionic start myApp sidemenu
```

Ionic provides 3 templates:

- blank
- tabs - includes a title header bar and a footer with links to tabs
- sidemenu - includes a collapsible lefthand sidebar

```
Github — -bash — 113x44
Mhamads-MacBook-Pro:github mhamadk$ ionic start myApp sidemenu

One awesome Ionic app coming right up...

Creating Ionic app in folder /Users/mhamadk/Github/myApp based on sidemenu project
Downloading: https://github.com/driftco/ionic-app-base/archive/master.zip
[=====] 100% 0.0s
Downloading: https://github.com/driftco/ionic-starter-sidemenu/archive/master.zip
[=====] 100% 0.0s
Installing npm packages...

Adding initial native plugins
[=====] 100% 0.0s

Adding in iOS application by default

add to body class: platform-ios
Downloading: https://github.com/driftco/ionic-default-resources/archive/master.zip
[=====] 100% 0.0s
Saving your Ionic app state of platforms and plugins
Saved platform
Saved plugins
Saved package.json

🎉 🎉 🎉 Your Ionic app is ready to go! 🎉 🎉 🎉

Make sure to cd into your new app directory:
  cd myApp

To run your app in the browser (great for initial development):
  ionic serve

To run on iOS:
  ionic run ios

To run on Android:
  ionic run android

To test your app on a device easily, try Ionic View:
  http://view.ionic.io

New! Add push notifications, update your app remotely, and package iOS and Android apps with the Ionic Platform!
https://apps.ionic.io/signup
Mhamads-MacBook-Pro:github mhamadk$
```

Ionic Platforms

Preview the list of platforms that Ionic support:

`ionic platform list`

Add a new platform

`ionic platform add android`

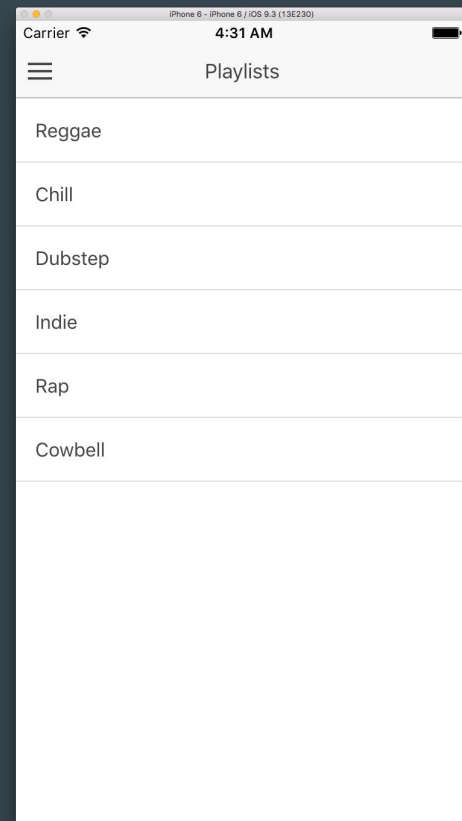
Remove a platform

`ionic platform rm android`

```
Mhamads-MacBook-Pro:myApp mhamadk$ ionic platform list
Installed platforms: ios 3.9.2
Available platforms: amazon-fireos, android, blackberry10, browser,
firefoxos, osx, webos
```

Running Your New App

ionic run ios



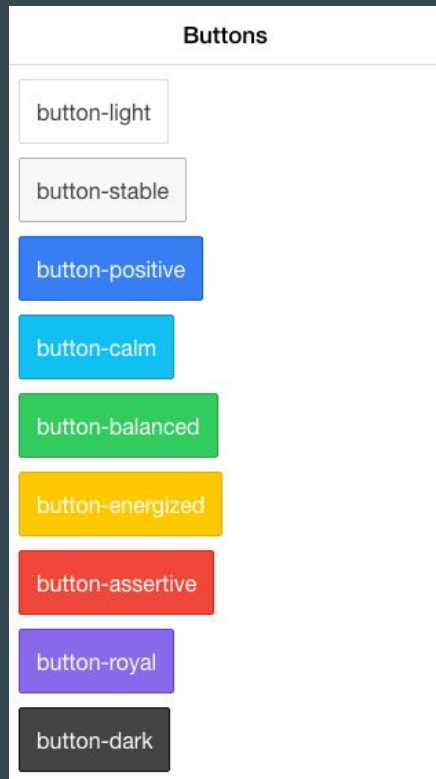
Features & Components

The beauty of Ionic is in it's slick UI.

<http://ionicframework.com/docs/>

Ionic provides CSS Components like Header, Buttons, Cards, Tabs, Grid, etc..

```
<button class="button button-stable">  
  button-stable  
</button>  
  
<button class="button button-positive">  
  button-positive  
</button>
```



Features & Components

Ionic provides JavaScript components like:

- Lists
- Loading
- Modal
- Popover
- Side Menus
- etc..

Feature: Popover

Most features are composed of a mix of HTML and JS.

The popover feature needs:

1. A button that will trigger the click event.
2. The code that will be triggered once we click the button.
3. The HTML to display once the code work.

Feature: Popover

1. A button that will trigger the click event.

```
<p>  
  <button ng-click="openPopover($event)">Open Popover</button>  
</p>
```

Feature: Popover

2. The code that will be triggered once we click the button.


```
angular.module('myApp', ['ionic'])
.controller('MyController', function($scope, $ionicPopover) {
  $ionicPopover.fromTemplateUrl('my-popover.html', {
    scope: $scope
  }).then(function(popover) {
    $scope.popover = popover;
  });

  $scope.openPopover = function($event) {
    $scope.popover.show($event);
  };

  $scope.closePopover = function() {
    $scope.popover.hide();
  };

  //Cleanup the popover when we're done with it!
  $scope.$on('$destroy', function() {
    $scope.popover.remove();
  });
});
```

This is what runs when you click the button.



Feature: Popover

3. The HTML to display once the code work.

```
<ion-popover-view>
  <ion-header-bar>
    <h1 class="title">My Popover Title</h1>
  </ion-header-bar>
  <ion-content>
    Hello!
  </ion-content>
</ion-popover-view>
```

Feature: Popover

There is nothing like trying it for yourself.

Check this codepen for a working example of the Ionic Popover.

<http://codepen.io/ionic/pen/GpCst>

Ionic is strong with You

If you have a problem, or if you wish to talk to other Ionic enthusiasts, check these:

Ionic Forums: <https://forum.ionicframework.com/>

Ionic Slack Channel: <http://ionicworldwide.herokuapp.com/>

Ionic Subreddit: <https://www.reddit.com/r/ionic/>

Ionic Twitter: <https://twitter.com/IonicFramework>