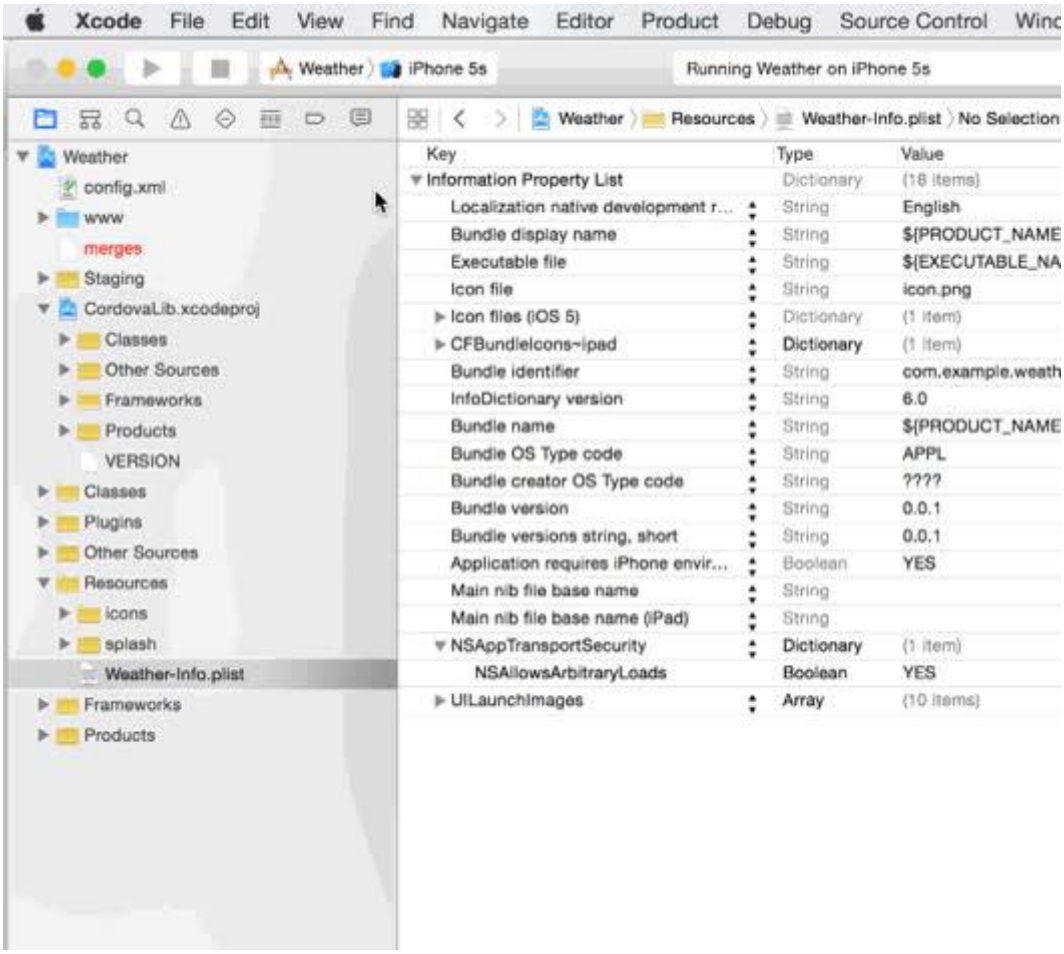
iOS App Development Using Cordova

Created by Todd Treece



https://learn.adafruit.com/ios-app-development-using-cordova

Last updated on 2023-08-29 02:57:31 PM EDT

Table of Contents

Overview 3

Installing Dependencies 3

Creating a New App 4

- index.html
- index.css
- index.js

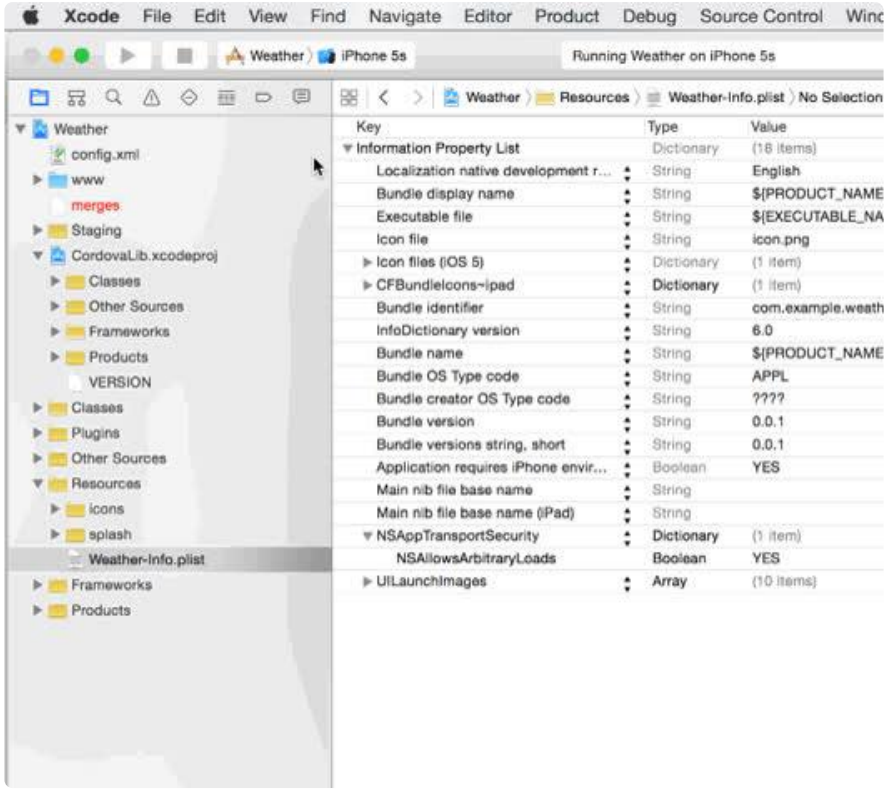
Testing the App 8

- iOS Simulator
- iOS Device

Customizing the App 9

- Modifying the config.xml

Overview



This tutorial will take you through the steps necessary to get started with using Apache Cordova () to write iOS applications. The benefit of using Cordova is that you do not have to learn Objective-C or Swift to write an app. If you can write HTML and Javascript, you can build a native iOS app with Cordova!

We are going to make a very simple weather application for our first project. It will use data from openweathermap.org ()and display it on the screen with an icon. Let's get started.

Installing Dependencies

You will need a Mac running OS X 10.10 to run the latest version of XCode, which is required for uploading to iOS devices. Thanks to changes in XCode 7, you no longer need an Apple Developer account to test applications on iOS devices. You can also use XCode 6 for this tutorial, but you will need an Apple Developer account to upload to your iOS device. XCode 7 is still in beta and not available in the App Store yet, so you will need to download it from Apple's Developer site and install.

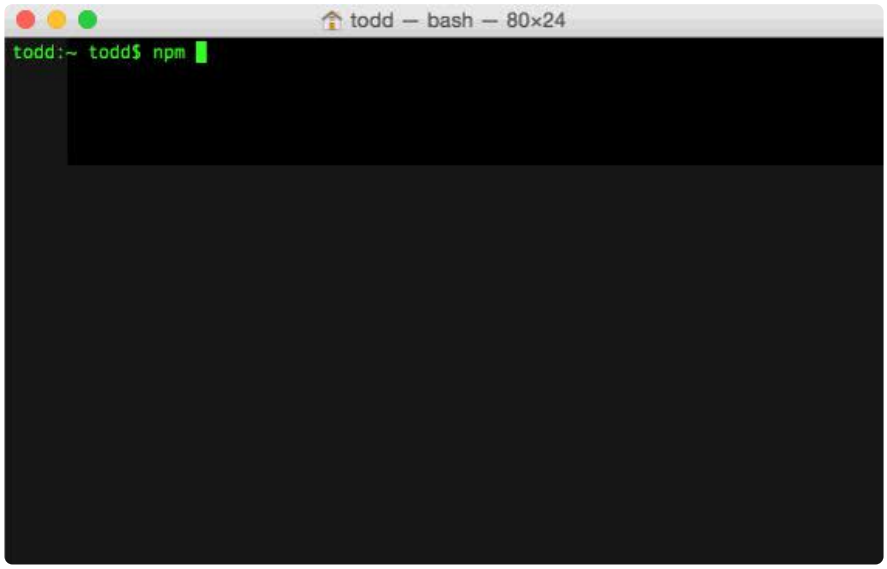
XCode 7 Download

You will also need the latest version of node.js () installed on your computer. Download and install the latest version using the link below.

Node.js Download

Next, you will need to install Cordova using npm from the terminal.

```
npm install -g cordova ios-sim
```

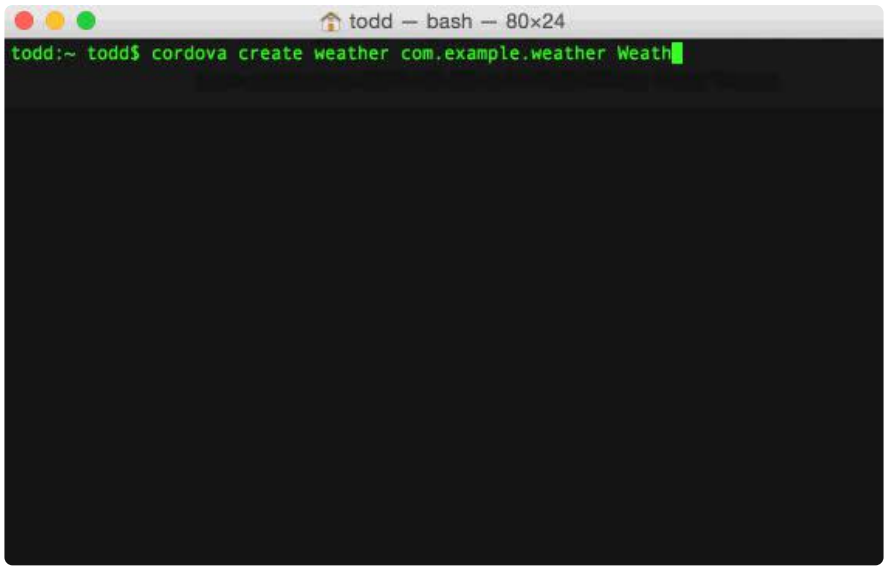


Now we are ready to look at creating a basic application.

Creating a New App

First, we will need to create the project from the terminal using the following command.

```
cordova create weather com.example.weather Weather
```



The cordova create command bootstraps a basic app using the supplied folder, app ID, and name. The new app has the following directory structure.

```
├── config.xml
├── hooks
│   └── README.md
├── platforms
├── plugins
└── www
    ├── css
    │   └── index.css
    ├── img
    │   └── logo.png
    ├── index.html
    └── js
        └── index.js

7 directories, 6 files
```

Most of our work will be focused on the files in the www folder. Since a Cordova application is a web app, most of the development can be done in a browser before testing on a phone. Let's get started with writing our basic weather app.

index.html

First, open up www/index.html in your favorite text editor, and replace the default HTML with the following:

```
&lt;!DOCTYPE html&gt;
&lt;html&gt;
  &lt;head&gt;
    &lt;meta name="viewport" content="user-scalable=no, initial-scale=1, maximum-
scale=1, minimum-scale=1, width=device-width"&gt;
    &lt;link rel="stylesheet" type="text/css" href="css/index.css"&gt;
    &lt;title&gt;Weather&lt;/title&gt;
  &lt;/head&gt;
  &lt;body&gt;
    &lt;div class="app"&gt;
```

```
      &lt;img src="" id="icon"&gt;
      &lt;div id="info"&gt;Loading...&lt;/div&gt;
    &lt;/div&gt;
    &lt;script src="https://ajax.googleapis.com/ajax/libs/jquery/2.1.4/
jquery.min.js"&gt;&lt;/script&gt;
    &lt;script type="text/javascript" src="cordova.js"&gt;&lt;/script&gt;
    &lt;script type="text/javascript" src="js/index.js"&gt;&lt;/script&gt;
  &lt;/body&gt;
&lt;/html&gt;
```

The HTML above sets the title of the page to 'Weather', adds a couple div elements to hold our weather info, and adds the latest version of jQuery to the page.

index.css

Next, open up www/css/index.css in your text editor, and replace the existing CSS with the following:

```
* {
  -webkit-tap-highlight-color: rgba(0,0,0,0); /* make transparent link selection,
adjust last value opacity 0 to 1.0 */
}

body {
  -webkit-touch-callout: none;                /* prevent callout to copy image, etc
when tap to hold */
  -webkit-text-size-adjust: none;             /* prevent webkit from resizing text
to fit */
  -webkit-user-select: none;                  /* prevent copy paste, to allow,
change 'none' to 'text' */
  background-color:#000;
  font-family:'HelveticaNeue-Light', 'HelveticaNeue', Helvetica, Arial, sans-serif;
  font-size:12px;
  height:100%;
  margin:0px;
  padding:0px;
  text-transform:uppercase;
  width:100%;
  color:#fff;
}

/* Portrait layout (default) */
.app {
  position:absolute;             /* position in the center of the screen */
  left:50%;
  top:16%;
  height:50px;                   /* text area height */
  width:225px;                   /* text area width */
  text-align:center;
  padding:180px 0px 0px 0px;     /* image height is 200px (bottom 20px are
overlapped with text) */
  margin:-115px 0px 0px -112px;  /* offset vertical: half of image height and text
area height */
  /* offset horizontal: half of text area width */
}

/* Landscape layout (with min-width) */
@media screen and (min-aspect-ratio: 1/1) and (min-width:400px) {
  .app {
    padding:75px 0px 75px 170px;  /* padding-top + padding-bottom + text area =
image height */
    margin:-90px 0px 0px -198px;  /* offset vertical: half of image height */
```

```
    /* offset horizontal: half of image width and text area width */
  }
}

#icon, #info {
  text-align: center;
}

#icon {
  width: 50%
}

h1 {
  padding: 0;
  margin: 0;
}
```

index.js

The final modification in the www folder we will need to make is to the www/js/index.js file.

```
var app = {

  zip: 10013, // nyc. feel free to change

  initialize: function() {
    this.bindEvents();
  },

  bindEvents: function() {
    document.addEventListener('deviceready', this.onDeviceReady, false);
  },

  onDeviceReady: function() {

    // grap weather info
    $.getJSON('http://api.openweathermap.org/data/2.5/find?q=' + app.zip +
',us&amp;units=imperial', function(data) {

      var info = $('#info'),
          icon = $('#icon'),
          result;

      // no results. bail
      if(data.count &lt; 1) {
        info.html('could not load weather');
        return;
      }

      // set result var to first result
      result = data.list[0];

      // show icon if we have a result
      if(result.weather.length &gt; 0) {
        icon.attr('src', 'http://openweathermap.org/img/w/' +
result.weather[0].icon + '.png');
        icon.show();
      }

      // add info to page
      info.html('&lt;h1&gt;' + result.name + '&lt;/h1&gt;&lt;br&gt;');
      info.append(result.main.temp + '° F&lt;br&gt;');
      info.append(result.main.humidity + '% Humidity&lt;br&gt;');
```

```
      info.append(result.wind.speed + ' MPH');
    });
  }
};
app.initialize();
```

Now that we have added all of our code to the project, we can test our app.

Testing the App

Before testing our app, we will need to add the Cordova whitelist plugin to our app, and to add the iOS platform. First add the plugin using the following command. This will only need to be done once for any plugin you use.

```
cordova plugin add cordova-plugin-whitelist@1.0.0
```

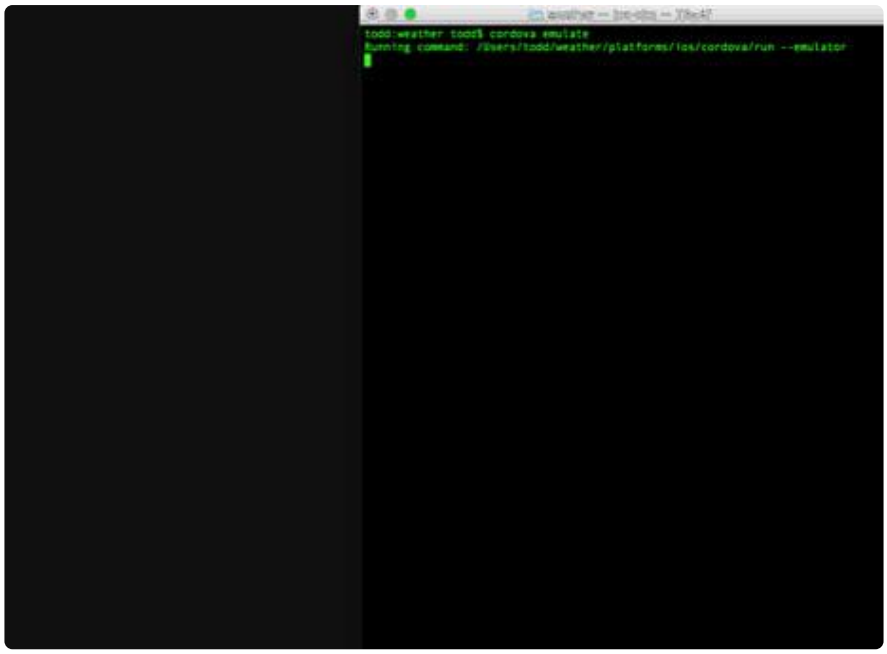
Next, add the iOS platform to the project. This will only need to be done once.

```
cordova platform add ios
```

iOS Simulator

Now we are ready to run the app in the iOS simulator. Run the following command to build the app and launch the simulator.

```
cordova emulate
```



iOS Device

Now we are ready to test the app on an actual iOS device. Plug your device in using a USB cable, and open XCode 7. Once XCode has finished opening, we will open the project using the File->Open menu. Navigate to the weather/platforms/ios folder and open Weather.xcodeproj.

Once the project has opened, click the play button at the top of the window to build and install the app on your device. Make sure the device is unlocked before clicking the play button.

The app should now launch on your device and display the weather! If you modify any of the files in the project, you will need to run the following command before uploading to your device.

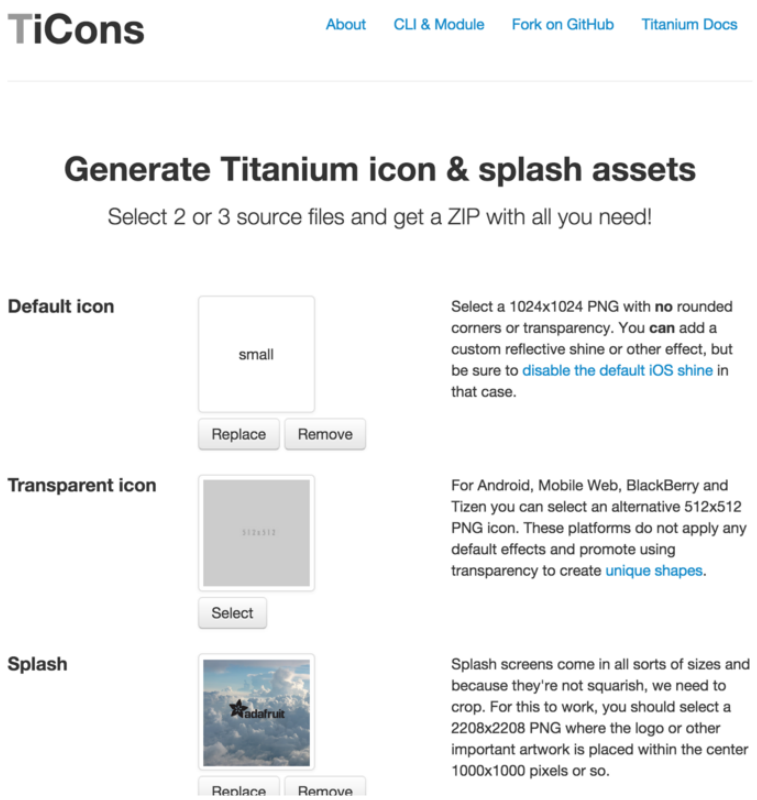
```
cordova build
```

Next, we will look at how to customize the app with a new icon and splash screen.

Customizing the App

Now that you know the basics of building and uploading an app, let's customize it a bit by adding a splash screen. We will be using TiCon () to make new splash and icon

images. Upload your images with the suggested sizes to the site and generate new images for your app.



Download the images and add them to your project under a new res/ios folder. The folder structure of your app should now look like the structure below.

```
├── config.xml
├── hooks
│   └── README.md
├── platforms
│   └── platforms.json
├── plugins
├── res
│   └── ios
│       ├── Default-568h@2x.png
│       ├── Default-667h@2x.png
│       ├── Default-Landscape-736h@3x.png
│       ├── Default-Landscape.png
│       ├── Default-Landscape@2x.png
│       ├── Default-Portrait-736h@3x.png
│       ├── Default-Portrait.png
│       ├── Default-Portrait@2x.png
│       ├── Default.png
│       ├── Default@2x.png
│       ├── appicon-60.png
│       ├── appicon-60@2x.png
│       ├── appicon-60@3x.png
│       ├── appicon-72.png
│       ├── appicon-72@2x.png
│       ├── appicon-76.png
│       ├── appicon-76@2x.png
│       ├── appicon-Small-40.png
│       ├── appicon-Small-40@2x.png
│       ├── appicon-Small-50.png
│       ├── appicon-Small-50@2x.png
│       ├── appicon-Small.png
```

```
│       ├── appicon-Small@2x.png
│       ├── appicon-Small@3x.png
│       ├── appicon.png
│       ├── appicon@2x.png
│       ├── iTunesArtwork
│       └── iTunesArtwork@2x
└── www
    ├── css
    │   └── index.css
    ├── img
    │   └── logo.png
    ├── index.html
    └── js
        └── index.js

9 directories, 35 files
```

Modifying the config.xml

Next, you will need to add the locations of your images to the config.xml found in your app. The file should look like this after you have finished.

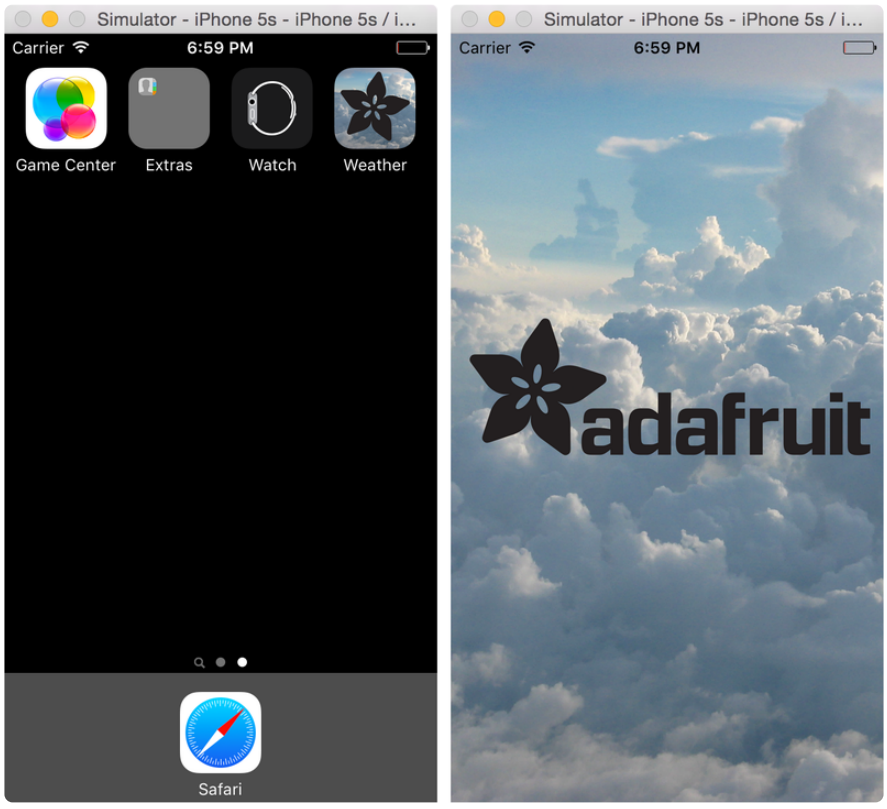
```
&lt;?xml version='1.0' encoding='utf-8'?&gt;
&lt;widget id="com.example.weather" version="0.0.1" xmlns="http://www.w3.org/ns/
widgets" xmlns:cdv="http://cordova.apache.org/ns/1.0"&gt;
    &lt;name&gt;Weather&lt;/name&gt;
    &lt;description&gt;
      A simple weather app.
    &lt;/description&gt;
    &lt;author email="dev@cordova.apache.org" href="http://cordova.io"&gt;
      Todd Treece
    &lt;/author&gt;
    &lt;content src="index.html" /&gt;
    &lt;plugin name="cordova-plugin-whitelist" version="1.0.0" /&gt;
    &lt;access origin="*" /&gt;
    &lt;allow-intent href="http://*/*" /&gt;
    &lt;allow-intent href="https://*/*" /&gt;
    &lt;allow-intent href="tel:*" /&gt;
    &lt;allow-intent href="sms:*" /&gt;
    &lt;allow-intent href="mailto:*" /&gt;
    &lt;allow-intent href="geo:*" /&gt;
    &lt;platform name="ios"&gt;
        &lt;allow-intent href="itms:*" /&gt;
        &lt;allow-intent href="itms-apps:*" /&gt;
        &lt;icon src="res/ios/appicon-60@3x.png" width="180" height="180" /&gt;
        &lt;icon src="res/ios/appicon-60.png" width="60" height="60" /&gt;
        &lt;icon src="res/ios/appicon-60@2x.png" width="120" height="120" /&gt;
        &lt;icon src="res/ios/appicon-76.png" width="76" height="76" /&gt;
        &lt;icon src="res/ios/appicon-76@2x.png" width="152" height="152" /&gt;
        &lt;icon src="res/ios/appicon-Small-40.png" width="40" height="40" /&gt;
        &lt;icon src="res/ios/appicon-Small-40@2x.png" width="80" height="80" /&gt;
        &lt;icon src="res/ios/appicon-Small-40@3x.png" width="120" height="120" /
&gt;
        &lt;icon src="res/ios/appicon.png" width="57" height="57" /&gt;
        &lt;icon src="res/ios/appicon@2x.png" width="114" height="114" /&gt;
        &lt;icon src="res/ios/appicon-72.png" width="72" height="72" /&gt;
        &lt;icon src="res/ios/appicon-72@2x.png" width="144" height="144" /&gt;
        &lt;icon src="res/ios/appicon-Small.png" width="29" height="29" /&gt;
        &lt;icon src="res/ios/appicon-Small@2x.png" width="58" height="58" /&gt;
        &lt;icon src="res/ios/appicon-Small@3x.png" wdth="87" height="87" /&gt;
        &lt;icon src="res/ios/appicon-Small-50.png" width="50" height="50" /&gt;
        &lt;icon src="res/ios/appicon-Small-50@2x.png" width="100" height="100" /
&gt;
        &lt;splash src="res/ios/Default.png" width="320" height="480"/&gt;
```

```
        &lt;splash src="res/ios/Default@2x.png" width="640" height="960"/&gt;
        &lt;splash src="res/ios/Default-Portrait.png" width="768" height="1024"/&gt;
        &lt;splash src="res/ios/Default-Portrait@2x.png" width="1536" height="2048"/
&gt;
        &lt;splash src="res/ios/Default-Landscape.png" width="1024" height="768"/
&gt;
        &lt;splash src="res/ios/Default-Landscape@2x.png" width="2048"
height="1334"/&gt;
        &lt;splash src="res/ios/Default-568h@2x.png" width="640" height="1136"/&gt;
        &lt;splash src="res/ios/Default-667h@2x.png" width="640" height="1136"/&gt;
        &lt;splash src="res/ios/Default-Landscape-736h@3x.png" width="750"
height="1242"/&gt;
        &lt;splash src="res/ios/Default-Portrait-736h@3x.png" width="1242"
height="2208"/&gt;
        &lt;splash src="res/ios/640x1136.png" width="640" height="1136"/&gt;
    &lt;/platform&gt;
&lt;/widget&gt;
```

After modifying the config, you will need to rebuild the app by running the following command.

```
cordova build
```

Now if you upload the app to a device or test it in the simulator, you will see the new icon and splash image.



That's it! We will be exploring interating with hardware from iOS in future tutorials, so stay tuned!