Ionic App Development with Joomla REST API

Mr. Dashrath Mane\(^1\) & Mr. Kaushik Shirodkar\(^2\)
\(^1\)Department of MCA, V.E.S.Institute of Technology, Mumbai, India
\(^2\)MCA Final year Student, V.E.S. Institute of Technology, Mumbai, India

Abstract: As there is a lot of development in the mobile application development, approach to develop these apps of the mobiles has become the one of major aspects. Hybrid apps are essentially small websites running in a browser shell in an app that have access to the native platform layer.

Keywords: Ionic, Joomla, REST

1. INTRODUCTION
WHAT IONIC IS:
Ionic is front-end framework. Hybrid mobile applications or apps are created using this technology. It is MIT licensed. The process of creation, building and package of hybrid mobile apps are performed with the help of Ionic which is also known as Node module. Drifty company supports both Ionic framework and module. Developer’s skills in AngularJS and HTML5 are empowered by Ionic to create high-end apps.

2. Setting Up Ionic

The main requirement for Ionic is Node.js. Node.js can be installed from [http://nodejs.org/](http://nodejs.org/). Then we can easily install Ionic using the package manager of Node which is also known as npm by entering the following command:

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

2.1 Creating an Ionic Project

Creating a project in Ionic is very easy. We have to write following command in console window:

```
ionic start myApp blank
```

The argument “start” specifies Ionic to start a new project in Ionic. “myApp” parameter specifies the name of the path and the fourth parameter determines the template. The following are templates available to us at this time:

a) Blank Template
b) Sidemenu Template
c) Tabs Template
3. Joomla Framework

The Joomla! Framework is a new PHP framework (a collection of software libraries/packages) for writing web and command line applications in PHP, without the features and corresponding overhead found in the Joomla! Content Management System (CMS). It provides a structurally sound foundation, which is easy to adapt and easy to extend. The Joomla! Framework is free and open source software, distributed under the GNU General Public License version 2 or later; and is comprised of code originally developed for the Joomla! CMS.

3.1 Advantages of Joomla Framework
a) Building a RESTful web services platform
b) Building both simple and complex command line tools
c) Building next generation web applications

4. REST API:

RESTful APIs (Application Programming Interfaces) or REST APIs are web services using REST architecture. REST is often used in mobile applications, social networking Web sites, mashup tools, and automated business processes. The client and services interaction is emphasized by REST by providing limited number of verbs. Resources or nouns are provided with additional flexibility with their own unique Universal Resource Identifiers (URIs). Because each verb has a specific meaning (GET, POST, PUT and DELETE), REST avoids ambiguity.

4.1 Data received from API:

REST api returns the data in the json format. Most Internet services will serve “send” you the data you requested “using HTTP GET/POST request” in JSON format.

5. Starting Ionic App:

Index.html file in our Ionic app is the first page. Basically every app in Ionic is a web page.

<!-- Needed for Cordova/PhoneGap (will be a 404 during development) -->
<script src="cordova.js"></script>
</head>
<body>
</body>
</html>

We need to include Ionic CSS and Ionic JS with AngularJS extension in the ionic.bundle.js file. To be able to use angular modules like ngAnimate or ngSanitize etc, we need to include them in lib/js/angular directory. Files like cordova.js or phonegap.js are required to be specified in the last script so that it will be automatically included while running or emulating the app.

5.1 Initializing the App:

We first need to create AngularJS to turn custom tags like <ion-side-menu> into functional. Otherwise when we run the code we wouldn’t see anything because there is not content. First, we need to create a new AngularJS module and tell Angular to initialize it. Create a file www/js/app.js and write the following code in it:

angular.module('todo', ['ionic'])

In index.html and right before the <script src="cordova.js"></script> line, add:

<script src="js/app.js"></script>

And to make our new app run, we need to add the ng-app attribute to the body tag:

<body ng-app="todo">

5.2 AngularJS – Controllers:

The flow of data in the AngularJS application relies on the controllers. “ng-controller” is used to define a controller. A controller is a JavaScript object containing attributes/properties and functions. $scope is accepted as a parameter by each controller which refers to the application that controller is to control.

<ng-app="" ng-controller = "SlideCtrl">
... 
</div>

5.3 ng directives used:

a) ngApp: To auto-bootstrap an AngularJS application this directive is used. The ngApp directive describes the root element of the application and it is placed near the root element of the page - e.g. on the <body> or <html> tags.
b) ngHref: Using Angular markup like {{hash}} in an href attribute will make the link go to the wrong URL if the user
clicks it before Angular has a chance to replace the {{hash}} markup with its value. Until Angular replaces the markup the link will be broken and will most likely return a 404 error. The ngHref directive solves this problem.

c) ngSrc: Using Angular markup like {{hash}} in a src attribute doesn't work right: The browser will fetch from the URL with the literal text {{hash}} until Angular replaces the expression inside {{hash}}. The ngSrc directive solves this problem.

d) ngController: This directive attaches a controller class to the view. This is how angular supports the principles behind the Model-View-Controller design pattern.

e) ngClick: The ngClick directive allows you to specify custom behavior when an element is clicked.

f) ngSubmit: Enables binding angular expressions to onsubmit events.

g) ngRepeat: The ngRepeat directive instantiates a template once per item from a collection. Each template instance gets its own scope, where the given loop variable is set to the current collection item, and $index is set to the item index or key.

6. Running an Ionic Project:
The myApp directory contains a lot of stuff that is generated for you. In all 137 files (~44MB) are created. We have 2 options with us.
To run the app in the browser.
To run the app on a native platform

6.1 Running in the Browser:

Ionic apps are built with web technologies. Webkit browser allows us to develop debug and test our Ionic apps. Apple Safari, Google Chrome, and the Opera browser supports Webkit. We first need to host our app in a web server if we want to open Ionic app in one of these browsers. Ionic comes with LiveReload, it allows you to run app locally. LiveReload is a server that gets installed as a Node module. Following command is used to run the app:

ionic serve

6.2 Running on a Native Platform:

Apache Cordova is required to run apps built with Ionic framework on native platform. Cordova is an open source library that makes device-level APIs accessible to JavaScript. This bridge helps you cross the divide between physical hardware and more malleable software.

Ionic module simplifies target platforms when app is ready for testing and deploying on specific platforms. To prepare app for a specific environment, enter ionic platform add <PLATFORMNAME>. For example, if you want your app to run on the three most popular mobile platforms (Android, iOS, and Windows Phone), enter the following:

ionic platform add android
ionic platform add ios  (will only work on a Mac OS X machine)
ionic platform add wp8
ionic build android
ionic build ios  (will only work on a Mac OS X machine)
ionic build wp8

Building your Ionic app creates a package using the SDKs installed on your local machine. You have the option of using the package task instead. That task uses the Ionic Build service to create app-store ready bundles without using SDKs on your local machine.

6.3 Running on a Physical Device:

To run ionic app on an actual physical device, enter the following into the command line:

ionic run --device
7. CONCLUSION

We only have scratched the surface of building an Ionic App. This article showed how to start a new hybrid app with Ionic. We also learned how to run the project in the browser and natively on a device. To build a feature-rich app, we need to turn our focus to the Ionic framework itself.

REFERENCES


[5] Native mobile apps: The wrong choice for business?- A white paper from michaels, ross & cole, ltd. (mrc)

[7] Hybrid Mobile Application Development Approaches-Sankar Srini, Sathish Venkatraman