

IONIC FRAMEWORK

Priyanka Chaudhary

Student, Department of computer science, ABESIT, Ghaziabad

Abstract-Through this assignment on worthwhile framework that is IONIC, I did not only gain comprehension of how ionic works, but also had a great chance to explore my skills to develop hybrid environment application for varied mobile operating systems such as iOS and Android. Ionic Framework offers set of rules for using various predefined functions to work up on User Interface of app, without worrying about the several work surfaces of mobile application. Using ionic, application comes into existence by implementing the application only once, can be viewed on diverse platforms such as iOS, Android. Ionic grants the heterogeneity in developing application.

An Ionic application can build up by means of the major two technologies: Angular and Apache Cordova. Ionic framework is an SDK i.e., software development kit. It is available as open-source tool. Ionic provides numerous functional facilities and tools already built-in to cultivate mobile application for heterogenous platforms.

It has made the job effortless for technocrat to build an application where carrying out app is liberated of platform.

Key Words: Ionic framework, mobile app development, native application, hybrid application, platforms

1. INTRODUCTION

Ionic framework is existing open source allows usage to generate applications for mobile devices. It offers numerous tools and facilities for building Mobile interface with native appearance and impression. Ionic framework needs a packaging, which is known as native wrapper. This enables ionic application run on varied mobile devices.

Ionic allows to work out a mobile application for diverse platform using HTML (Hypertext Markup Language), CSS (Cascading Stylesheets) and JavaScript. Ionic is the HTML5 founded Mobile Application Development Framework. It handles the appearance in addition of impression of UI of Hybrid mobile application. (Wasserman, 2010)

The kind of development of application for heterogeneous mobile environment can be majorly partitioned into two segments, first is Native Application and another is Hybrid Application.

Native app is created for specified mobile operating system. These application programs are build using the platform-based SDK, pre-defined functionalities and languages for a specific platform such as Objective-C for iOS and java for Android.

Hybrid applications are those applications which uses web language for implementation. These applications then run in a native container, which allows them to run on heterogeneous mobile operating system.

Hybrid mobile applications are built in a likewise manner as any websites. But the hybrid applications target a WebView hosted inside a native container, instead of mobile browser. This enables the application to use numerous services of mobile like permission to way in the hardware capabilities of the mobile device. Most hybrid mobile applications leverage the technology- Apache Cordova, a platform which provides a consistent set of JavaScript application programming interfaces. It provides several plug-ins to access device capabilities, which are built with native code.

Ionic provides all the functionality which can be found in SDKs of native mobile environment. Ionic application enables inventor to build their applications and customize them according for different platforms such as for iOS, for Android, and deploy through Cordova. It includes various mobile components, extensible base themes and typography. Ionic uses framework of JS i.e., Angular, which offers custom components and method to interact.

1.1 OBJECTIVE

Ionic is a front-end HTML framework manages appearance and impression as well as User interface interactions of the hybrid app. Ionic apps are mainly created and obtained through the Ionic CLI (command line utility), the plugins of Cordova are used to develop and deploy as a native app. (Malavolta, 2015)

Ionic application has objective to build a UI for mobile application without implementing that app in the native language for the platform-based application. With ionic, an application can be packaged for IOS, Android, Windows Phone and Firefox OS. This will save a lot of time as it would take more time to separately develop application for iOS, Android or Windows.

Ionic application development starts with its installation, where the pre-generated app setup will be available. The starting of app development becomes easier.

Ionic supports the app development in modules, so that maintainability & updating is easier.

Ionic offers cross-platform application production for various mobile OS with the easy-going web technologies.

2. Features of Ionic

Ionic framework gets accompanied with various worthwhile features. The most salient attributes are:

Ionic CLI

It is utility of NodeJs having commands to perform operations such as booting up, executing, running, reproducing app via emulation of the ionic applications.

```
npm install -g ionic@latest
```

Fig 2.1 Command to install ionic

```
ionic start myNewProject
```

Fig 2.2 Command to start ionic app project

Ionic View

A platform convenient for transmitting, sharing, testing the applications on the native mobile devices.

```
ionic serve
```

Fig 2.3 Command to run ionic app

Cordova Plugins

Apache Cordova offers plugins to utilize application program interface(API) which enables native device utilities usage with javascript code. (Camden, 2015)

```
npm install -g cordova
```

Fig 2.4 Command to integrate cordova in app

MVC Architecture

Ionic practices MVC architecture through angular for forming single-page application which is usable for varied mobile devices. (Resig, 2015)

It segregates the whole application into three parts, those are Model, View and Controller.

Model is responsible for maintaining data.

View is responsible for displaying data to the user i.e., HTML and CSS part.

Controller controls the interactions between the Model and View.

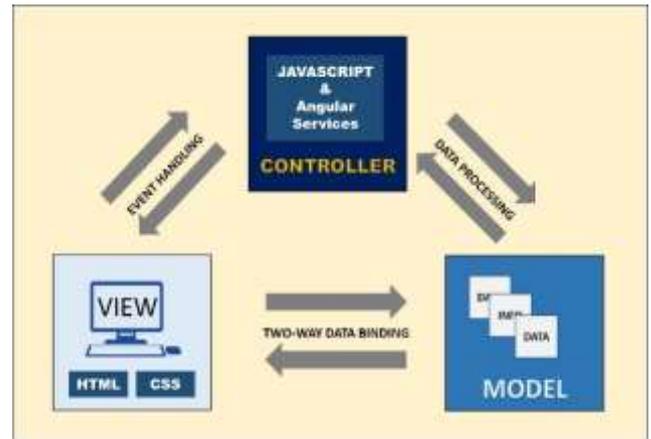


Fig 2.5 MVC Architecture

CSS components

Ionic makes it easier to bring native appearance and impression with backing of css components. Components which propose elements needed on mobile device applications. Styling of components ease modifying the default styles.

```
<div class="bar bar-header bar-calm">
  <h1 class="title">bar-calm</h1>
</div>
```

Fig 2.6 CSS component for heading bar

JavaScript components

JavaScript component adds extension to UI components with javascript services to case all the elements on varied mobile devices, which is not achievable by use of HTML and CSS.

Ionic Themes

Defined Series of styles used in decorating an app are themes. Ionic offers light theme as default theme. Themes can be customized according to the requirement of app.

3. Execution of Ionic Application

Ionic applications start with the pre-generated framework when installed. The project setup consists of folders such as-

node_modules for nodejs dependency, platforms for varied mobile OS, plugins for APIs, resources for splash screen assets, src and files to run application.

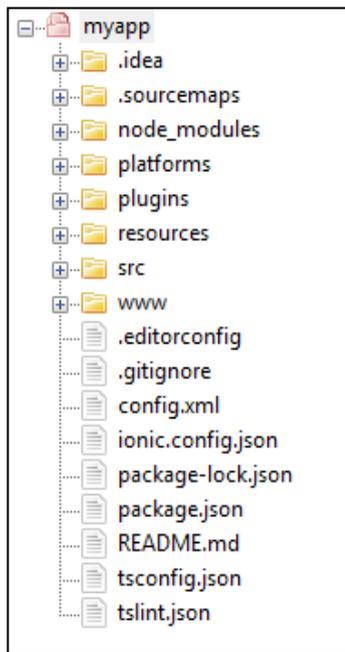


Fig 3.1 Folder Structure of ionic app

The ionic framework setup can be described as-

3.1 src folder

This folder comprises of source code for application. This will be implementation code for several mobile OS. The elements for respective component is defined in this folder.

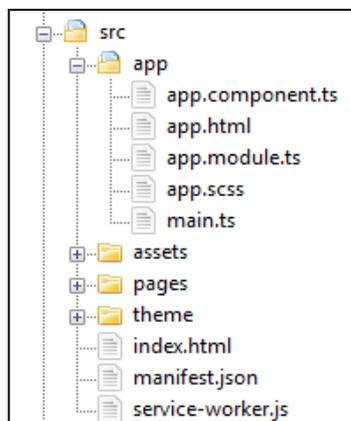


Fig 3.2 Expanded view of src folder

In src folder, app component is default main component. The pages folder comprises of all the sections of the template. The theme folder has the styling files of the application. There is assets folder stores supplies such as images and icons for application.

3.1.1 index.html

This is the HTML file which runs when the project loads. This file will run by default to load the HTML DOM. (Graham, 1995)

3.1.2 main.ts

This is the typescript file which is the entry point of the application. It bootstraps the application.

3.2 Configuration files

The two files config.xml and package.json are the most important ones.

config.xml

This is used by Cordova, which includes settings relevant for constructing application for the iOS and Android platform.

package.json

The dependencies of the npm packages are stated in this file.

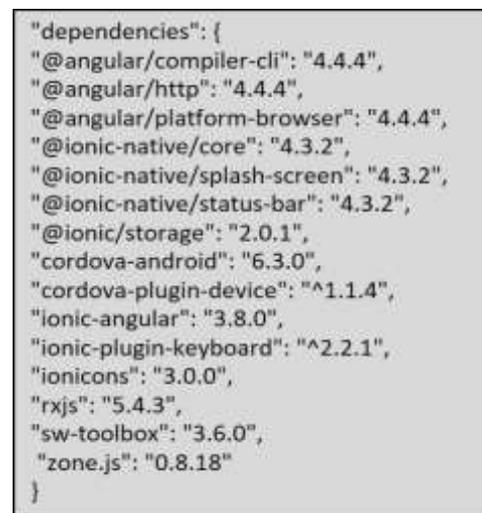


Fig 3.3 Dependencies in package.json

3.2 www folder

The web root folder which contains the compiled code of the ionic application for various platforms. Inside the www folder, the build subfolder has the compiled code.

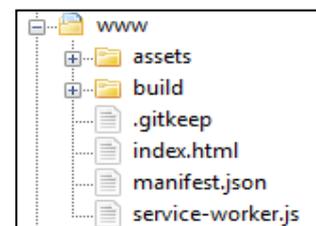


Fig 3.4 Expanded view of www folder

3.3 Resource folder

The resources folder contains images and icons which are used for the splash screen app when building for different platforms such as iOS and Android.



Fig 3.5 Expanded view of resources folder

3.4 node_modules folder

This is the storage folder where all the installed NPM dependencies can be found. This means any dependency which is defined in the package.json file will be downloaded and stored in here (Tilkov, 2010)

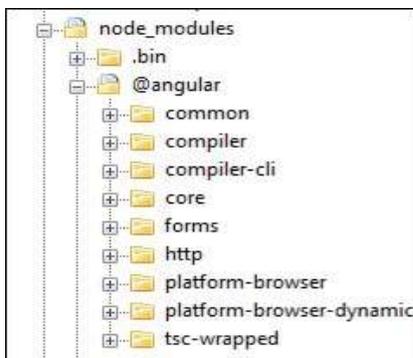


Fig 3.6 Expanded view of node_modules folder

Templates in Ionic app

Ionic framework provides majorly three types of template which are: Blank template, Side menu template, Tabs template.

Blank Template has a blank type starter project

Tabs Template has a simple tabbed interface.

Side menu template has navigation in the content area i.e., side menu.

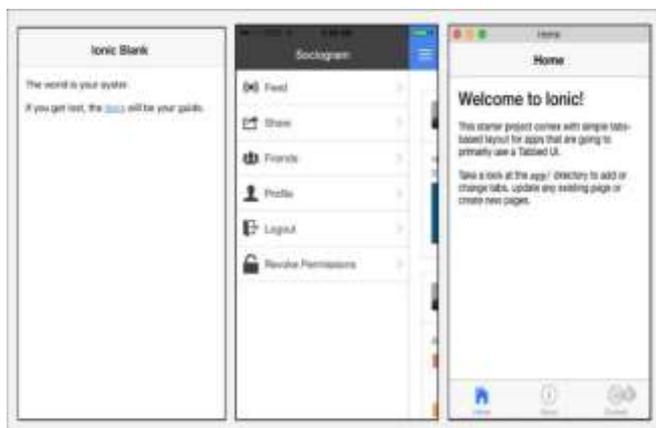


Fig 3.7 Different types of Templates

Ionic Themes

Ionic themes are set of styles for the application over various platform. Ionic framework works on top of the sass technology. With these, customization of application can be done. (Prabhu, 2015)

```
Colors: {
  primary: #488aff,
  secondary: #32db64,
  danger: #f53d3d,
  light: #f4f4f4,
  dark: #222
};
```

Fig 3.7 Customizing color theme

Ionic Components

Ionic framework has the defined components which are the constituents of cross-platform interface of a mobile application.

These components are of two types: Template Components and Dynamic Components.

Template Component

The components that are composed of just HTML and CSS are referred as template component. Also known as CSS component.

Example:

```
<ion-card>
  <ion-card-header>
    My Title
  </ion-card-header>
  <ion-card-content>
    My Content is Here
  </ion-card-content>
</ion-card>
```

Dynamic Component

The components that also include the dynamic functionalities to make the app interactive are the dynamic components.

Example:

```
<button ion-button (click)="open()">
  Open
</button>
```

Here in ionic, button element will be represented by ion-button to define the ionic button.

4. CONCLUSION

Ionic is a backdrop support which operate on top of Cordova to build hybrid apps. It allows to develop an application interface which are comparable as they were web pages i.e., using HTML, CSS, JavaScript. And these apps run inside WebView of native application. Virtue of ionic framework is – it is an open source framework in which code needs to be written once and execute on numerous mobile devices (Wasserman, 2010). The use of only one programming language enables the app on all the different mobile OS. Availability of various plugins to use hardware and other extra services for mobile app. Ionic apps has the well build graphics. This is the developer's favorable framework where simple technologies are used to build the entire interface of the application.

ACKNOWLEDGEMENT

I owe a special debt of gratitude to Prof. Ashutosh Pandey, Department of Computer Science & Engineering, ABES Institute of Technology, Ghaziabad for his constant support and guidance throughout the course of my work. His sincerity, thoroughness and perseverance have been a constant source of inspiration for me. I also do not like to miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind assistance and cooperation during this assignment. Last but not the least, I acknowledge my friends for their contribution throughout the assignment.

REFERENCES

- [1] Camden, R. K. (2015). Apache Cordova in Action. Greenwich: Manning Publications Co.
- [2] Graham, I. S. (1995). The HTML SourceBook. New York: John Wiley & Sons, Inc.
- [3] Malavolta, I. (2015). Hybrid mobile apps in the Google play store: an exploratory investigation. Florence, Italy: MOBILESoft '15.
- [4] Prabhu, A. (2015). Introduction to Preprocessors. Mumbai: Apress, Berkeley, CA.
- [5] Resig, J. (2015). AngularJS and Testing. NJ: Apress, Berkeley, CA.
- [6] Tilkov, S. (2010). Node.js: Using JavaScript to Build High-Performance Network Programs. IEEE Internet Computing, 80 - 83.
- [7] Wasserman, A. I. (2010). Software engineering issues for mobile application development. New Mexico: FoSER '10.