ISSN 2278 - 3091
International Journal on Advanced Trends in Computer Science and Engineering (IJATCSE), Vol. 4 No.1, Pages : 14 – 18 (2015)
Special Issue of ICACE 2015 - Held on March 02, 2015,Mysore, India
http://warse.org/pdfs/2015/icace2015sp04.pdf

A Novel approach of creating cross-platform mobile applications using Apache Cordova

Ananth G S[+] Dr. K Raghuveer[*]
+ Asst. Professor, Dept of PGSCEA, NIE, Mysore (ananth.gouri@gmail.com)
*Professor & Head, Dept of ISE, NIE, Mysore (raghunie@yahoo.com)

Abstract – This paper aims at demonstrating the approach of creating cross-platform mobile applications using the framework called Apache Cordova. To accomplish this we create a simple application, build it and demonstrate how simple it is to run the same applicaion on 2 platforms Android and iOS, though other platforms like palm OS, firefox, Blackberry are supported by Apache Cordova.

Keywords: Android, Blackberry, Cross-platform, Cordova, HTML, iOS

INTRODUCTION - Apache Cordova is a platform for building native mobile applications using HTML, CSS and JavaScript. Apache Cordova is a set of device APIs that allow a mobile app developer to access native device function such as the camera or accelerometer from JavaScript. When using the Cordova APIs, an app can be built without any native code (Java, Objective-C, etc) from the app developer. Instead, web technologies are used, and they are hosted in the app itself locally (generally not on a remote http server).

And because these JavaScript APIs are consistent across multiple device platforms and built on web standards, the app should be portable to other device platforms with minimal to no changes.

Apps using Cordova are still packaged as apps using the platform SDKs, and can be made available for installation from each device's app store.

Cordova provides a set of uniform JavaScript libraries that can be invoked, with device-specific native backing code for those JavaScript libraries. Cordova is available for the following platforms: iOS, Android, Blackberry, Windows Phone, Palm WebOS, Bada, and Symbian.

PhoneGap or Apache Cordova ? What's in a name?

In the year 2011, Adobe or Nitobi the creators of PhoneGap, donated the PhoneGap codebase to the Apache Software Foundation. It was done to ensure proper stewardship of the source of PhoneGap, and to maintain a transparent and open governance that was well documented and understood. Also they wanted to make it easier for the larger orgranizations towards contribution.

As a result of this contribution, they were required to ensure that the intellectual property was unfettered by trademark ambiguity. This presented them with the hard requirement to rename PhoneGap in its open source form as a project incubating at Apache. The name first chosen was Apache Callback, which was dumb for a host of reasons and quickly the community renamed it to Cordova. While genesis stories of PhoneGap often vary with the teller, most committers can agree the project was born at Nitobi, when the office was on Cordova Street in Vancouver and hence the name Apache Cordova[2].

Apache Cordova graduated in October 2012 as a top level project within the Apache Software Foundation. Through the ASF, future Cordova development will ensure open stewardship of the project. It will always remain free and open source under the Apache License, Version 2.0.[1]

INSTALLING APACHE CORDOVA

The Work Environment - Our work environment is a internet connected LINUX based Linux Mint OS with version 17.0. The steps of installation could change as per your developer perspective respectively. Please follow proper documentation of your environment before proceeding from here.

ISSN 2278 - 3091
International Journal on Advanced Trends in Computer Science and Engineering (IJATCSE), Vol. 4 No.1, Pages : 14 – 18 (2015)
Special Issue of ICACE 2015 - Held on March 02, 2015,Mysore, India
http://warse.org/pdfs/2015/icace2015sp04.pdf

In our environment Apache Cordova was installed as a Command Line Interface (CLI)

Installation of SDKs - The prerequisite before installing Apache Cordova is to install the appropriate SDKs for running the application, i.e, if we wish to deploy the application onto an Android and an iOS environment, it is mandatory to install the Android SDK and the iOS SDK before

proceeding. Installation of Android SDK is as in [3] and installation of iOS SDK is as in [4]

Installation of Apache Cordova

First we need to download and install Node.js. Following installation, we should be able to invoke *node* and *npm* on our command line.

Next we need to download and install a git client. Following installation we were able to invoke *git* on our command line.

Install the Cordova module using npm utility of Node.js by issuing the command

$ sudo npm install -g cordova

On Windows the same can be called from the cmd line using the command

C:\>npm install -g cordova

CREATING THE APP

From the directory where the source code is maintained we issued the command

cordova create hello com.example.hello HelloWorld

This may take some time for the command to complete. Running the command with the -d option displays information about its progress.

The first argument hello specifies a directory to be generated for your project. This directory should not already exist, Cordova will create it. Its www subdirectory houses our application's home page, along with various resources under css, js and img, which follow common web development file-naming conventions. These assets will be stored on the device's local filesystem, not served remotely. The config.xml file contains important metadata needed to generate and distribute the application.

The second argument com.example.hello provides the project with a reverse domain-style identifier. This argument is optional, but only if we also omit the third argument, since the arguments are positional. We can edit this value later in the config.xml file, but do be aware that there may be code generated outside of config.xml using this value, such as Java package names. The default value is io.cordova.hellocordova, but it is recommended that we select an appropriate value.

The third argument HelloWorld provides the application's display title. This argument is optional. We can edit this value later in the config.xml file, but do be aware that there may be code generated outside ofconfig.xml using this value, such as Java class names. The default value is however Hello Cordova

ADDING PLATFORMS FOR CROSS-PLATFORM MOBILE APPLICATIONS

All subsequent commands from now on should be run within the project's directory or any an subdirectories within Cordova's scope.

$ cd hello

Before we can build the project, we need to specify a set of target platforms. The ability to run these commands depends on whether our machine / the environment supports each SDK and also whether we have already installed each SDK.

ISSN 2278 - 3091
International Journal on Advanced Trends in Computer Science and Engineering (IJATCSE), Vol. 4 No.1, Pages : 14 – 18 (2015)
Special Issue of ICACE 2015 - Held on March 02, 2015,Mysore, India
http://warse.org/pdfs/2015/icace2015sp04.pdf

$ cordova platform add ios

$ cordova platform add amazon-fireos

$ cordova platform add android

$ cordova platform add blackberry10

To check the current set of platforms

$ cordova platforms ls

To remove a platform

$ cordova platform remove android

Running commands to add or remove platforms affects the contents of the project's *platforms* directory, where each specified platform appears as a subdirectory. The *www* source directory is reproduced within each platform's subdirectory, appearing for example in platforms/ios/www or platforms/android/assets/www. Because the CLI constantly copies over files from the source *www* folder, we should only edit these files and not the ones located under the *platforms* subdirectories. If version control software is used, we should add this source *www* folder, along with the *merges* folder, to the version control system.

SOURCE SNIPPET OF THE APPLICATION

The www folder contains an index.html page that is like a launcher for all the applications across all platforms. We created another file

1.html apart from the index.html file to demonstrate our application.

Content of index.html

```
<!DOCTYPE html>
<html>
<body>
<p>Link to the app <a href="1.html">
Our App</a></p>
</body>
</html>
```

Content of 1.html

```
<!DOCTYPE html>
<html>
<body>
<p> This is our demo app. </p>
<script type=”text/javascript”>
var a=window.prompt(“Enter number 1”);
var b=window.prompt(“Enter number 2”);
var sum=+a + +b;
document.write(“The sum of a and b is” + sum);
</script>
</body>
</html>
```

As the code can be analysed – its a pretty simple application written in complete JavaScript embedded in HTML for reading 2 numbers and performing addition of 2 numbers.

Let's see how this small program can be deployed across multiple platforms.

ISSN 2278 - 3091
International Journal on Advanced Trends in Computer Science and Engineering (IJATCSE), Vol. 4 No.1, Pages : 14 – 18 (2015)
Special Issue of ICACE 2015 - Held on March 02, 2015,Mysore, India
http://warse.org/pdfs/2015/icace2015sp04.pdf

BUILDING THE APP ACROSS CROSS-PLATFORMS

Before an application can be built – it is mandate that the platform's SDK path is set as an environment variable.

The command *$ cordova build* - will iteratively build the project.

The output is as shown in the below diagram:

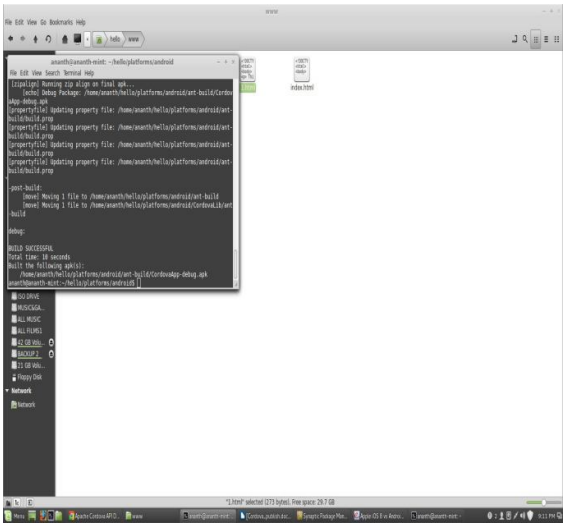


Fig 1 shows cordova build ouput

Building an app for the Android platform alone

Setting environment variable value – *export ANDROID_HOME=.:/home/ananth/android-sdk-linux*

$ cordova build android – will build the project that could be deployed onto a Android environment.

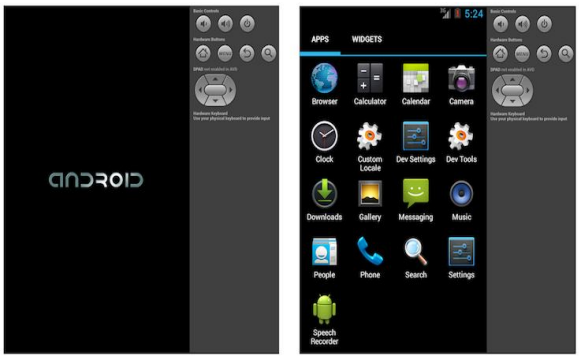
Building an app for the iOS platform alone

$ cordova build ios – will build the project that can be deployed onto an iOS environment.

The command *cordova build* is equivalent of 2 commands – *cordova prepare* and *cordova compile*.

DEPLOYING THE APP ON CROSS-PLATFORM USING EMULATORS

Assuming the respective emulators are up and running, for example if the Android emulator is up and running – the app can be tested on its emulator using the below command:



$ cordova emulate android

***Fig 2** shows* Android Emulator up and running

7.1 Deploying the app on cross-platform using hand-held devices

A hand-held device wrt Android is a smartphone based on Android OS.

Copy the .apk file from the path – *hello/platforms/android/ant-build* to the phone's sd-card or device storage[5][6]. Upon installation of the package, the below image should be seen and note that source installations from unconfirmed sources should be enabled in the android smart phones:

ISSN 2278 - 3091
International Journal on Advanced Trends in Computer Science and Engineering (IJATCSE), Vol. 4 No.1, Pages : 14 – 18 (2015)
Special Issue of ICACE 2015 - Held on March 02, 2015,Mysore, India
http://warse.org/pdfs/2015/icace2015sp04.pdf

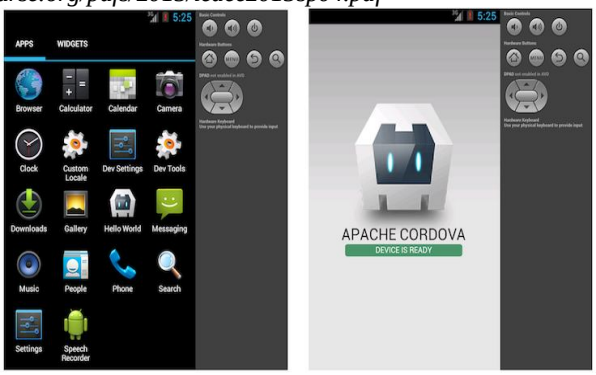


Fig 3 shows Apache Cordova instance running

When the apk is run or the application is run – the below figure 4 is got (with an input of a=4 and b=5, the output is 9).

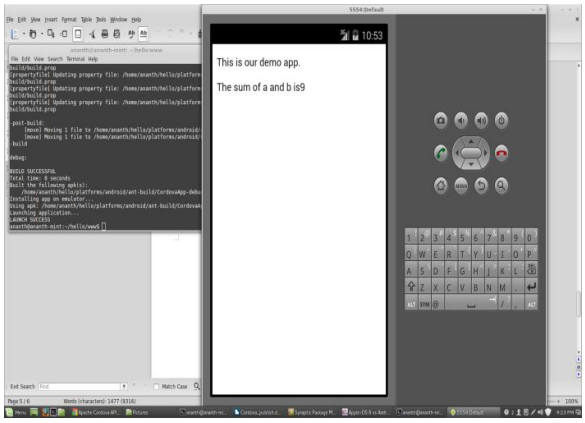


Fig 4 shows the output of the application – addition of 2 numbers on Android environment

CROSS PLATFORM DEPLOYMENT OF THE APP - RUNNING THE APP ON AN IOS ENVIRONMENT

The same application when built on iOS environment, the below output is achieved.

Fig 5 shows the application running on an iOS based smartphone

CONCLUSION

This paper demonstrated the easy usage of deploying a simple application onto various platforms with a small change of few commands. Apache Cordova and PhoneGap are both advantageous in building applications using HTML, CSS and JavaScript. There are numerous applications built using Cordova platform. To name a few – wikipedia app, facebook app, salesforce app etc[7] This paper laid a foundation to build applications using Apache Cordova.

ACKNOWLEDGMENT

This work was initiated by the Department of PGSCEA. We also thank the Principal and the Management of NIE for extending support to this work.

REFERENCES

[1] Phonegap API
[2] Apache Cordova API
[3]http://www.androidcentral.com/installing-android-sdk-windows-mac-and-linux-tutorial
[4]http://www.lynda.com/iOS-tutorials/Installing-Xcode-iOS-SDK/159179/159282-4.html
[5] Kerri Shots, Phonegap 3.x Mobile Application Development
[6] John M Wargo, Apache Cordova 3 Programming
[7] Inputs from www.tricedesigns.com