Smartphone Application Development using HTML5-based Cross-Platform Framework

Si-Ho Cha[1] and Yeomun Yun[2,*]

[1]Dept. of Multimedia Science, Chungwoon University 113, Sukgol-ro, Nam-gu, Incheon, South Korea
[2]Dept. of Applied Music, Chungwoon University Daehakgil-25, Hongseong, Chungnam, South Korea
[1]shcha@chungwoon.ac.kr, [2]hippie740@chungwoon.ac.kr

Abstract

A smartphone is a mobile phone built on a mobile operating system, with more advanced computing capability and connectivity than a feature phone. The mobile operating systems (OS) used by modern smartphones are too diverse such as Google's Android, Apple's iOS, Microsoft's Windows Phone, and so on. Such operating systems can be installed on many different phone models. In these circumstances, smartphone application development is done using native platform such as iPhone using Objective-C, Android using Java, Windows Mobile using C# and so on. Therefore, a cross-platform framework which supports 'Write once and deploy everywhere' is required to support the development of smartphone applications that operate on various modern smartphones. The cross-platform framework should be able to support native device features as well as use new HTML5 Web technologies. This paper presents the HTML5-based cross platform framework which uses PhoneGap and Webkit to support the development of smartphone applications that are written as Web applications, run locally on the smartphone and can leverage native phone capabilities. This paper also shows the development of a sample smartphone application using the cross platform framework.

Keywords: *Smartphone Application, Cross-Plarform Framework, HTML5, PhoneGap, Webkit*

1. Introduction

A smartphone is a mobile phone built on a mobile operating system, with more advanced computing capability and connectivity than a feature phone. The mobile operating systems (OS) used by modern smartphones include Google's Android, Apple's iOS, Nokia's Symbian, RIM's BlackBerry OS, Samsung's Bada, Microsoft's Windows Phone, Hewlett-Packard's webOS, and embedded Linux distributions such as Maemo and MeeGo. Such operating systems can be installed on many different phone models, and typically each device can receive multiple OS software updates over its lifetime [1]. In general, there are three different solutions for smartphone development: 1) native, 2) Web, and 3) hybrid. Native application is the application that works natively. Native application code is written specifically for a particular phone’s operating system. In native development, smartphone application development is basically done using native platform say iPhone using Objective-C, Android

[*] Corresponding author

using Java, Windows Mobile using C# and so on [2]. Web application is the application that renders via a Web browser using Web application solutions including HTML, CSS, and JavaScript. There are three specific techniques in Web application development that are borrowed for these non-Web frameworks: 1) layout with mark-up (HTML5/CSS); 2) using URLs to identify screen layouts and visual state; and 3) incorporating dynamic languages, such as Javscript and Ruby [3]. Hybrid application is the combination of a native application and a Web application. Smartphone frameworks are influenced by the rapid application development techniques we are seeing in Web development today [4].

Application developers have been hindered by the need to develop an application from scratch for each smartphone platform. Because applications developed in the native language of one operating system aren't compatible with others, it's tough for users to have a consistent experience with an application. To address this, several cross-platform development tools have come out in the last year or two. By cross-platform development, it means the developer writes the code for the app once, and then can port it to other smartphone platforms relatively easily. One way to do this is to write the main code in Web standards, such as JavaScript, HTML, and CSS (Cascading Style Sheets). Many programmers are already familiar with these standards, and they were designed to be interpreted by many platforms and browsers.

Therefore, there is need of a framework which supports ‘Write once and deploy everywhere’. Different Smartphone operating systems come up with their own App store for distributing native application. There are some issues associated with the native application development such as, Out-of-sync data, No two mobile platforms share a mobile application, and there are too many mobile operating systems exist in the market such as iPhone, Android, Blackberry, Symbian *etc*. Same application needs to be developed for different platform differently. This increases development cost [2]. The best approach to create truly cross-platform app is to use HTML5 [5] and JavaScript based on cross-platform frameworks. One interesting framework for creating HTML5/JavaScript based apps is PhoneGap [6] and Webkit [7]. It has quite many API's available and wide support for different platforms.

The rest of the paper is organized as follows. Section 2 discusses the related work, and Section 3 introduces the proposed HTML5-based cross-platform framework for smartphone application development. Section 4 describes an example on smartphone application development. Finally, in Section 5, conclusions are made including the future research.

2. Related Works

Cross-platform development allows a single smartphone application to work across multiple OSs. Currently there are limited solutions and most of the cross-platform solutions do not support each of the main platforms and only a subset. In addition there are constraints to be considered in selecting a native platform compared to the cross-platform.

Table 1 shows the advantages and disadvantages of using native platform versus cross-platform environments [3]. HTML5 is a potential candidate for cross-platform mobile applications. Many features of HTML5 have been built with the consideration of being able to run on low-powered devices such as smartphones and tablets. Web technologies (*e.g.* browsers, caching systems, protocols) and languages (*e.g.* HTML5, JavaScript, mashups) provide accessible from anywhere, at any time, easy to install and upgrade, simple to use, comprehensive, lightweight, non-restrictive, and flexible way to deploy applications efficiently. PhoneGap [6] is an HTML5 application platform that allows you to author native applications with Web technologies and get access to APIs and app stores. Webkit [7] is an open source web browser engine. Webkit powers Google Chrome and Safari. WebKit was originally derived by Apple Inc. from the Konqueror browser's KHTML software library for use as the engine of Safari web browser, and has now been further developed by individuals from the KDE project, Apple Inc., Nokia, Google, Bitstream, Torch Mobile, Samsung and others.

Table 1. Native Platform vs. Cross-platform Development Solutions [3]

	Native	Cross-Platform
Advantages	• Library update • Direct technical support • Stable • App store and device portal solution • Better UI design result, can take full advantage of display	• Write once, run a lot of places • Open Source solution • One programming language family for all • Common UI design could be implemented for multiple OSs • Fast development • Reduced long term maintenance cost
Disadvantages	• Not all have Open Source solution • Different programming language • Different UI design pattern • Slow development time • Requires many budget and experience	• Limited direct technical support • Unstable • UI design depends on the platform and is limited

3. Cross-Platform Framework

A framework is a set of libraries, software components and architecture guidelines that provides the developer with a comprehensive toolkit to build a complete mobile application, from top to bottom. It is commonly called full-stack development frameworks; normally more complex to use than a single library [8]. We introduce the cross-platform for smartphone applications as showed in Figure 1. Our cross-platform framework uses HTML5, PhoneGap, Webkit, and so on.

PhoneGap [6] is an open-source tool that lets you use JavaScript, HTML, and CSS to code an application once, then deploy it to the iPhone, Android, and BlackBerry. It is support to give Web developers JavaScript access to popular mobile device features, like the camera, GPS, the accelerometer, local SQLite databases and more, without having to write full applications. The PhoneGap framework acts as a bridge between Web applications and mobile devices. It lets developers wrap Web applications inside a native application, making development easier for those who aren't familiar with Objective-C and Cocoa.

jQTouch [9] is a jQuery [10] plug-in for mobile Web development with native animations, automatic navigation, and themes for mobile WebKit browsers on the iPhone, Adnroid, iPod Touch, and other forward-thinking devices.

Webkit [7] is an open source library that renders HTML on the page. The iPhone and Google Android both use Webkit directly and now Torch Mobile [6] is making a Webkit-based browser for Windows Mobile. Eliminating the incompatibilities among mobile browsers makes it a bit easier to develop cross-platform tools. Android may run Java and the iPhone may be built on Objective C, but the same application can reach both via Webkit. Every new mobile device and every new Web browser support Web standards including

HTML5, CSS3, and JavaScript. These Web standards are open, reliable, highly secure, and efficient.

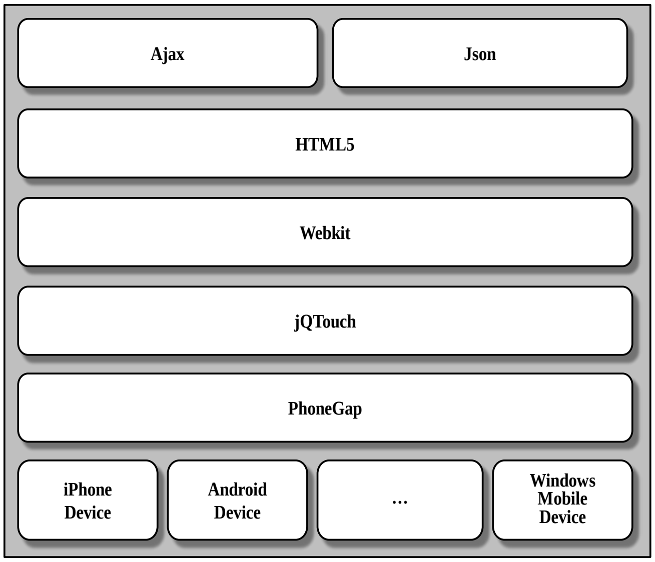


Figure 1. Cross-platform for Smartphone Applications

Ajax (Asynchronous JavaScript and XML) [11] is a group of interrelated web development techniques used on the client-side to create asynchronous web applications. With Ajax, web applications can send data to, and retrieve data from, a server asynchronously (in the background) without interfering with the display and behavior of the existing page. Data can be retrieved using the XMLHttpRequest object. Despite the name, the use of XML is not required (JSON is often used instead), and the requests do not need to be asynchronous [12].

JSON (JavaScript Object Notation) [13] is a text-based open standard designed for human-readable data interchange. It is derived from the JavaScript scripting language for representing simple data structures and associative arrays, called objects. Despite its relationship to JavaScript, it is language-independent, with parsers available for many languages.

The Web is continually growing more open and more social. Because of this, many websites have opened up programming interfaces (API's) that allow developers to get at their core information. A prime example of this is Google Maps, which is a very popular interface to use in mashups. Google allows developers to access their maps through API's. The developer can then combine these maps with another stream of data to create something new and unique. A Web mashup is a Web application that takes information from one or more sources and presents it in a new way or with a unique layout [14]. In the development of hydride smartphone applications, these Web meshups are able to use to develop new creative smartphone applications.

4. Implementation

HTML5 and JavaScript have become incredibly interesting (and fun) with the rise of frameworks and the mobile landscape. PhoneGap is an open source framework for quickly building cross-platform mobile applications using HTML5, JavaScript and CSS. The procedure of cross-platform application development is as follows: 1) Wrap your app with

PhoneGap; 2) Deploy to mobile platforms. Building applications for each device requires different frameworks and languages. PhoneGap solves this by using standards-based Web technologies to bridge Web applications and mobile devices.

A traditional Web application simply doesn't work without a network. One solution to this problem is to use two features of the HTML5 Standard: 1) Offline Web applications; 2) Client-side database storage. The user can use cloud functions on a mobile device, work offline with a locally deployed application on a local database, and share data with the rest of the cloud when going online again.

Figure 2 shows an overview of the major components of the sample application architecture. Our sample application is a simple book inventory management application. It lets you keep track of book lists you own. To do this, our sample application uses the Apache HTTP Server (httpd) version 2.2.23 [15]. The user can use cloud functions on a mobile device, work offline with a locally deployed application on a local database, and share data with the rest of the cloud when going online again.

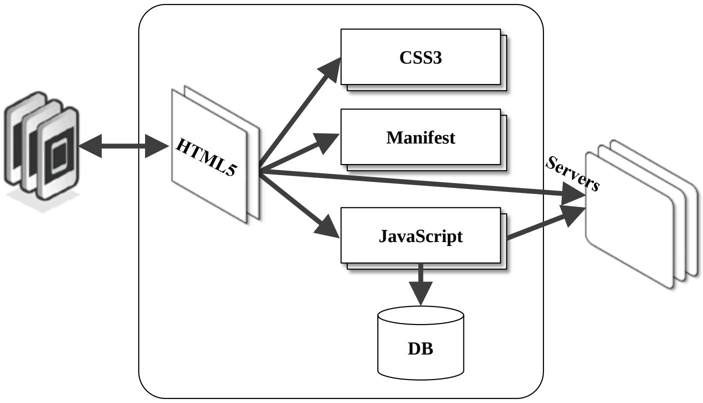
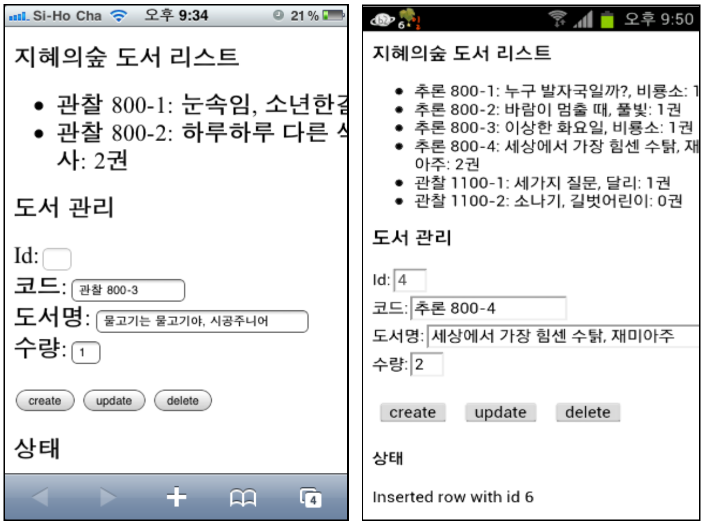


Figure 2. Core Elements of a Sample Application

The HTML5 page, the core of the application, has the model role. It contains the displayed data and the (default) render information. The JavaScript element contains the controller functions of the application. HTML elements are bound via event handlers to JavaScript functions. The HTML5 standard introduced local database storage. It is implemented in current versions of the Safari browser and Chrome browser. The browsers provide an embedded database, with SQLite, that can be accessed from the JavaScript by processing SQL queries. The business data of the application model is stored here. Our sample application uses a cache manifest file to support offline applications work. The manifest file is the mandatory deployment descriptor component for an offline Web application. It simply lists all the files that need to be loaded. The manifest file is a list of all of the resources that our sample application might need to access while it’s disconnected from the network. In order to bootstrap the process of downloading and caching these resources, we need to point to the manifest file, using a manifest attribute on HTML5’s <html> element. Our manifest file lists two resources: a CSS3 file and a JavaScript file.

An iPhone 3GS and a Galaxy S-II were used to run our sample application on real smartphones. Figure 3 shows our sample book inventory management application in both iPhone 3GS (iOS) and Samsung Galaxy S-II (Android).



지혜의숲 도서 리스트

- 관찰 800-1: 눈속임, 소년한
- 관찰 800-2: 하루하루 다른 사: 2권

도서 관리

Id:
코드: 관찰 800-3
도서명: 물고기는 물고기야, 시공주니어
수량: 1

상태

지혜의숲 도서 리스트

- 추론 800-1: 누구 발자국일까?, 비룡소: 1
- 추론 800-2: 바람이 멈출 때, 풀빛: 1권
- 추론 800-3: 이상한 화요일, 비룡소: 1권
- 추론 800-4: 세상에서 가장 힘센 수탉, 재아주: 2권
- 관찰 1100-1: 세가지 질문, 달리: 1권
- 관찰 1100-2: 소나기, 길벗어린이: 0권

도서 관리

Id: 4
코드: 추론 800-4
도서명: 세상에서 가장 힘센 수탉, 재미아주
수량: 2

상태

Inserted row with id 6

Figure 3. Sample Application Deployed on iPhone 3GS and Galaxy S-II

The list in the upper part of the screen gives an overview of all entered books. When a user selects a book in the list, the details (id, code, title, quantity) of the book are shown in the middle of the form. The details of the selected book can be changed using the Update button. The selected book can also be deleted from the application using the Delete button. New books can be created by entering the item's code, title, and quantity in the form and selecting the Create button. The application status is displayed in the lower part of the screen.

5. Conclusion

Native applications can provide a really nice responsive user experience. However, application developers who have to create smartphone applications for many different smartphones may not have the budget or experience to develop natively. A hybrid approach using JavaScript/CSS3/HTML5 with consistent application architecture provides a simple solution for modern smartphone application development. In this paper, we described a cross-platform framework for smartphone application which supports the hybrid approach and a sample smartphone application based on the cross-platform framework. A prototype of a simple book management application demonstrated the HTML5 technology with a locally deployed application and a local database. Writing once and deploying a lot of smartphones will help businesses to release fast applications development and reduce long term maintenance cost. Cross-platform frameworks will be best suitable options for real business benefits with modern smartphones.

References

[1] Wikipedia: Smartphone. http://en.wikipedia.org/wiki/Smartphone.

[2] R. Gangundi, “Smartphone Application Development using Cross Platform Frameworks”, Proceedings of the National Conference on Information and Communication Technology, NMIMS University, Mumbai, India, **(2010)**.

[3] S. Allen, V. Graupera and L. Lundrigan, Pro Smartphone Cross-Platform Development: iPhone, BlackBerry, Windows Mobile and Android Development and Distribution. Apress, **(2010)**.

[4] P. Douangboupha, “Smart Phone Platform Comparison”, Whitepaper Series, R2integrated, **(2009)**.

[5] W3C Draft: HTML5 - A vocabulary and associated APIs for HTML and XHTML. http://dev.w3.org/html5/spec/Overview.html.

[6] PhoneGap. http://www.phonegap.com/.

[7] Webkit, http://www.webkit.org/.
[8] G. Hartmann, G. Stead and A. DeGani, "Cross-Platform Mobile Development", Tribal, **(2011)**.
[9] jQTouch. http://www.jqtouch.com/.
[10] jQuery. http://docs.jquery.com/Main_Page.
[11] J. J. Garrett, Ajax: A New Approach to Web Applications. http://www.adaptivepath.com/ideas/ajax-new-approach-web-applications, **(2005)**.
[12] C. Ullman, Beginning Ajax. wrox. **(2007)**.
[13] JSON. http://www.json.org/.
[14] D. Nations, What is a Mashup? About.com.
[15] http://webtrends.about.com/od/webmashups/a/what-is-mashup.htm.
[16] Apache HTTP Server Project, http://httpd.apache.org/.

Authors

Si-Ho Cha received his B.S. degree in Computer Science from Sunchon National University, Sunchon, Korea, in 1995, and his M.S. and Ph.D. degrees in Computer Science from Kwangwoon University, Seoul, Korea, in 1997 and 2004, respectively. From 1997 to 2000, he worked as a senior researcher at R&D Center of Daewoo Telecom. He is now an Assistant Professor in the Department of Multimedia Science, Chungwoon University, Incheon, Korea. His research interests include network management, vehicular ad hoc networks, wireless sensor networks, and web of things.

Yeomun Yun received his B.S. degree in Music Production & Engineering from Berklee College of Music, Boston, MA, USA in 2004, and his M.S. degree in Music Technology from New York University, NY, USA in 2006. He is now an Assistant Professor in the Department of Applied Music, Chungwoon University, Chungnam, Korea. His research interests include music engineering, audio sound design, analog-to-digital audio conversion, electronic music, and computer music composition.