International Journal of Computer Science
and Engineering (IJCSE)
ISSN(P): 2278-9960; ISSN(E): 2278-9979
Vol. 3, Issue 3, May 2014, 23-30
© IASET

International Academy of Science, Engineering and Technology
IASET
Connecting Researchers; Nurturing Innovations

IMPLEMENTATION OF CROSS-PLATFORM MOBILE APPLICATION USING PHONEGAP FRAMEWORK

AVINASH SHRIVAS[1] & ANANDKUMAR PARDESHI[2]

[1]Associate Professor, Vidyalankar Institute of Technology, University of Mumbai, Mumbai, Maharashtra, India

[2]Assistant Professor, Fr. C Rodrigues Institute of Technology, University of Mumbai, Mumbai, Maharashtra, India

ABSTRACT

Mobiles are an important part of daily life. Nowadays, users are expecting handy applications in less time. It is big challenge to develop high performance mobile applications that would meet the expectation of customers. Several mobile platforms available in the market, developing same type of applications for different mobile platforms are time consuming. Mobile operating systems companies are giving their best available resources for creating application in much convenient way. Mobile application development tools have solved this cross-platform problem to a great level. These tools provide the development of application in much shorter time than to develop by using specific software development kit of mobile operating systems. The applications created by the tools are cross-platform which avoids developing same type of application separately. So these tools could be really useful for the developers to build applications with a wide scope and a reduced amount of time.

This paper aims to develop Cross-Platform Mobile Application using Phone Gap Framework. Resulting application is a cross-platform mobile application for student information system. In Institution of higher education, the number of student owning mobile phones is more than the one owning computers. With mobile device, "Information at our finger tips anywhere at any time", so that users can have a better experience to access information and enjoy many applications. There is a chance to handover academic affairs management from computer to mobile phones to improve service quality and management efficiency. This project is to sharing information between students, staffs and college administration. The application will help students to check their results, attendance, view personal details, check for announcements etc. which are easy to use directly from mobile phones.

KEYWORDS: Cross, Platform, Information System, Mobile Application, Phone Gap

1. INTRODUCTION

In recent years, with the rapid development of mobile communication technology, mobile phone not only are used to call or send short messages, but also a smart tools, through which we can browse the web pages, send and receive e-mails, view word or excel documents, play games, study, etc. A smartphone is a mobile phone that offers a more advanced computing ability and connectivity than a phone with up to date features [1]. They are much more efficient in form factor, chip type, internal storage capacity, battery lifetime and operating system. Modern mobile platforms include Symbian, iPhone, Android, Windows Phone 7, Palm, Blackberry, etc. from the developer standpoint [2].

Application development for different mobile and tablet platforms is very costly and time intensive. The same application has to be developed multiple times in several programming languages using platform specific development tools to cover all the platforms. Similarly, the cost to use development tools and application revenue taken by the operating systems for selling the applications through their stores results in a lot of wasted revenues for companies. Hence, there is a need to develop cross-platform applications capable of running across all mobile and tablet platforms, which is more cost effective and less labor-intensive [3]. Through cross-platforming, developers can create applications for multiple platforms using the same code base. Cross-platform mobile development tools deliver native application code which can be deployed on all supported mobile OS. These types of tools are gaining popularity in the globe. Such tools are mainly depending on web development languages like HyperText Markup Language (HTML5), JavaScript and Cascading Style Sheets (CSS) with some native wrapper code for accessing hardware features like Camera, Contacts, etc. The application development is very easy and time saving in these tools. There are plenty of such tools available now in the market, which create confusion among developers on which one to embrace and which one to skip [4]. This paper will try to put some light on PhoneGap tool. These tools have cross-platform compatibility and the architecture which is mostly based on web services. The objective of this paper is to creation and management of accurate and up-to-date information for staff, students and college authorities. It will ensure data integrity and validation and support for strong error-handling system. This system is expected to increase efficiency; users can access as well as share the information from mobile phones. The system utilizes user authentication, displaying only information necessary for an individual.

In this paper, after Introduction, section 2 gives some idea about the mobile application development. Then, section 3 gives the development tools which are required to develop and implement mobile app. Section 4 and 5 gives design and implementation of the system and next is test cases and screenshots of the system which is followed by conclusion and future scope presented in section 6.

2. MOBILE APPLICATION DEVELOPMENT

There are various types of mobile phone applications: native applications, applications based on web technologies and hybrid applications.

2.1 Native Applications: A native application is one that was designed to be installed on a specific OS (iOS, Android, Symbian etc.) and it is also depending of the device's firmware. In order to have an application running in different devices of models small or big changes will be needed. The Table 1 shows major mobile application development along with the native packaging format [5]. This can be developed using C, C++, and Java etc. But, these applications will only support their respective operating systems and will not provide Cross-Platform support.

Table 1: Major Mobile Platforms

	Apple iOS	Android	Blackberry OS	Windows Phone
Languages	Objective-C, C, C++	Java	Java	C#, VB.NET and more
Tools	Xcode	Android SDK	BB, Java Eclipse Plug-in	Visual Studio, Windows Phone development tools
Packaging format	.app	.apk	.cod	.xap
App stores	Apple app store	Google Play	Blackberry app world	Windows Phone Marketplace

2.2 Applications Based on Web Technologies: Web based applications are developed using web technologies such as HTML, JavaScript and should be interpreted by the web browsers of the different operating systems. Table 2 gives the comparison between pure mobile web apps and pure mobile websites [6].

Table 2: Comparison between Pure Mobile Web Apps and Pure Mobile Websites

Feature	Pure Mobile Web Apps	Pure Mobile Websites
Tools and knowledge	Written entirely in HTML, CSS and JavaScript	Written entirely in HTML, CSS and JavaScript
Execution	"installed" shortcut, launched like a native app	Reached by navigating to a website by way of a URL
User experience	Touch-friendly, interactive UI	Navigational UI between pages displaying static data
Performance	UI logic resides locally, making the app responsive and accessible offline	All code executed from a server, resulting in network-dependent performance

2.3 Hybrid Applications: The hybrid approach combines native development with web technology. Using this approach, developers write significant portions of their application in cross-platform web technologies, while maintaining direct access to native APIs when required. The native portion of the application uses the operating system APIs to create an embedded HTML rendering engine that serves as a bridge between the browser and the device APIs. This bridge enables the hybrid app to take full advantage of all the features that modern devices have to offer.

Table 3: Comparison between Native App vs Hybrid App vs Web App

Feature	Native App	Hybrid App	Web App
Development languages	Native only	Native and web or web only	Web only
Code portability and optimization	None	High	High
Access device-specific features	High	Medium	Low
Leverage existing knowledge	Low	High	High
Advanced graphics	High	Moderate	Moderate
Upgrade flexibility	Low	Moderate	High
Installation experience	High	High	Medium

3. DEVELOPMENT TOOLS

For developing the cross-platform app, different development tools and technologies were used. For the Android app development, the Android SDK combined with the Eclipse IDE and its corresponding plug-in was used.

Eclipse is a multi-language software development environment containing an IDE and a plug-in system. Eclipse is written in Java and the IDE is a particular favorite with Java developers. The IDE can also be used to write applications in other programming languages, i.e. Ada, C and C++. The Eclipse SDK is meant for Java developers and users can extend its capabilities by installing plug-ins for the Eclipse platform. For Android development in Eclipse the plug-in Android Development Tools (ADT) is installed. Eclipse SDK is free and open source software under the terms of Eclipse Public License.

Android is an open source platform based on the Linux kernel. Developing Android applications or games do not require a license. Therefore Android recommends open source tools for developers as the Eclipse IDE. The Android web browser is an open source engine running WebKit with some properties from the Chrome browser. The programming language in Android is mostly Java and XML [7]. The Android SDK contains many tools and some of them are the Android Virtual Device (AVD) with its virtual mobile device emulator and the debugger called Dalvik Debug Monitor Server (DDMS). Smartphones running Android implement full multitouch since version 2.0.

Phone Gap is an open source mobile development framework originally produced by Nitobi, but later purchased by Adobe Systems. It enables mobile app developers to build apps for mobile devices using HTML5, CSS3 and Java Script. This makes it easier for the developer, because they do not have to be skilled in every lower-level language corresponding to the desired platforms, such as Objective-C for iOS or Java for Android. After using PhoneGap the resulting applications are hybrid applications, meaning that they are neither truly native (all the layout is rendered via the webview instead of the platform's native UI framework) nor purely web-based (they are not ordinary web apps but they are packed for marketing distribution, and have access to part of the device API) [8].

HTML, CSS and Java Script The Hyper Text Markup Language (HTML5) is the common programming language among web browsers when displaying web pages. HTML combines the content of a page, such as images, links, text and scripts. The markup language uses CSS to give a web page a more suitable layout than just plain text. CSS adds color, font, text size and disposition to a web page. JavaScript can be included to a HTML document to give a page some interactivity, e.g. changing the color of a button when the mouse pointer is hovering over it.

jQuery is constructed to ease developers when creating scripts for a web page. Some features are selection and manipulation of elements, CSS manipulation and HTML event functions. jQuery is an open source software licensed under MIT and GNU General Public License.

JQuery Mobile Building cross-platform web applications with jQuery Mobile allow developers to access a JS specifically targeted for smartphones. Developing with jQuery Mobile gives the application a unified look and feel across different OSes for smartphones. The jQuery Mobile package comes with pre-defined themes and icons. jQuery Mobile is optimized for touch screens and comes with built in themes which simplifies the design of an application.

Phone Gap Build Phone Gap Build is a service that lets developers compile or build their PhoneGap based apps in the cloud service. The service’s goal is to eliminate the need to download and configure mobile platform SDKs required to build native/installable applications [9]. The principle is to write the application in HTML, CSS and JavaScript and upload it to the PhoneGap build service and get back app-store ready application binaries for Apple iOS, Google Android, Palm, Symbian, BlackBerry, and later for Windows Phone 7, Meego and Bada.

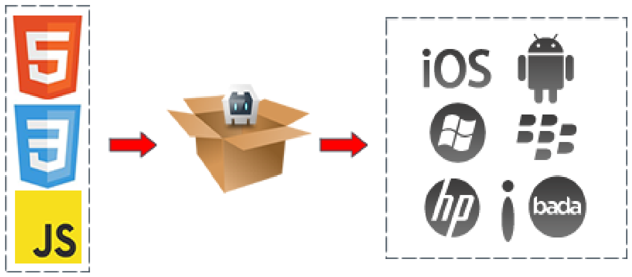


Figure 1: Phone Gap Application Packaging and Distribution [10]

4. DESIGN AND IMPLEMENTATION OF THE SYSTEM

Design of SIS Phone Gap is used to make the transition from the native mobile languages to web based programming languages in all three operating systems. The backend code of every platform is written in native language. As seen in figure 2, the PhoneGap framework bridges the gap between the native language and HTML by working as a wrapper for the app and generating JavaScript used in the app for accessing the native API. Due to jQuery and jQuery Mobile the application is written in one HTML file thus allowing every page to be within a div and identifying them with different ID's when used to navigate through the app. The content shown in the app differs depending on what page the user has navigated to. The application is constructed with different views that the user navigates through. The home screen lets the user see an overview of all the available pages the user can navigate to, see figure 3. The paper aimed for a familiar look and feel of the application so that a user easily can understand the navigation through the app. With the help of jQuery Mobile the design of the application is similar in icons and toolbars with most common applications. Within a page, a header was created that contains a navigation bar, allowing the user to return back to previous page or go directly to the home screen.

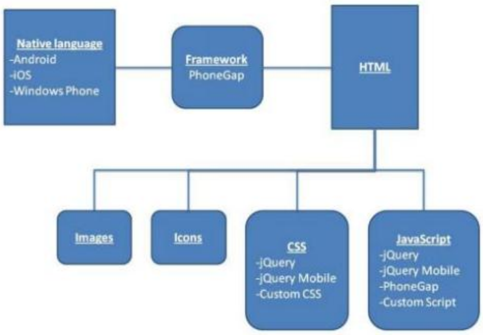


Figure 2: Structure of a Phone Gap Produced App

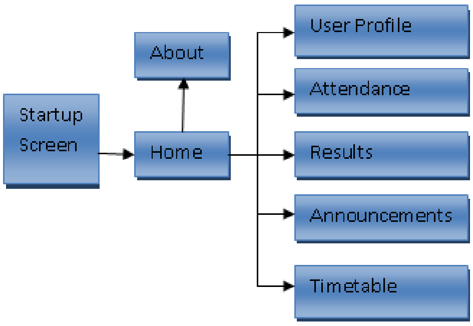


Figure 3: Overview of the Navigation through the App

Implementation of the System

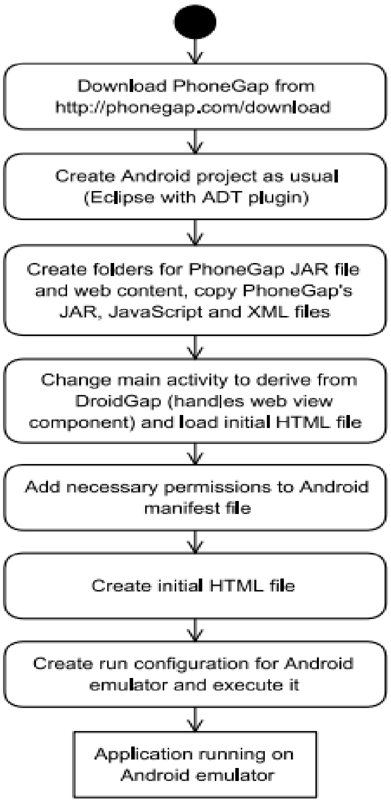

Download PhoneGap from http://phonegap.com/download
Create Android project as usual (Eclipse with ADT plugin)
Create folders for PhoneGap JAR file and web content, copy PhoneGap's JAR, JavaScript and XML files
Change main activity to derive from DroidGap (handles web view component) and load initial HTML file
Add necessary permissions to Android manifest file
Create initial HTML file
Create run configuration for Android emulator and execute it
Application running on Android emulator

Figure 4: Procedure of Installation and First Application Run Using Phone Gap [11]

The figure 4 shows the installation procedure for PhoneGap. During development the build process using the Eclipse IDE since this setup was a one-time effort, it is not part of the setup/development time measurement. The development cycle is slightly different between tests in a browser and on a device/emulator. Browser testing is of course not possible if device features such as the camera are used, so in order to enable testing in the browser, the application should check if those features are available at all. In order to view the application in a browser, only a local web server must be started, serving the HTML, JavaScript and other files. Any changes in the code thus only require a page reload. For device or emulator-based development, the web part of the application and the native project must be combined. With Android, for instance, the web part must be copied to the assets/www folder from which PhoneGap's web view implementation will serve the web page.

Working of the System

The project was developed using the MVC model design plan shows in figure 5. The MVC approach improves development by breaking the application into three distinct components: the model, the view and the controller. The business logic is separated from the presentation, and the controller is placed between them to communicate. With MVC, each component can be developed and maintained in isolation. Using MVC, the business logic can be reused, and thus time is saved when rewriting the same code. The code will also be much easier to read and understand.

Similarly, the model holds the real business logic and the state. It defines the process an application is intended to represent. In addition to this, the model communicates with the database if needed and sends the appropriate information back to the controller. The controller determines the response of the application by communicating with both the model and view. It uses the user input from the request, which it gets from the view, and sends it to the appropriate model. It is also used to route all requests to the appropriate controller and to return the response [3]. Finally, the view is used to format the data returned by the model and to present it to the users. The view receives the state of the model from the controller.

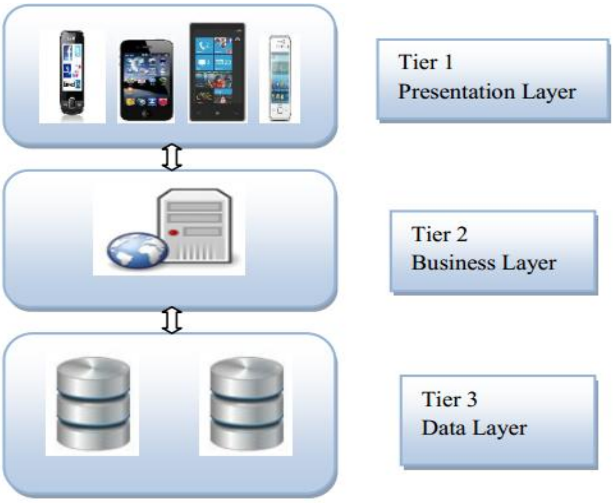


Figure 5: MVC Model Used for Application Development

When the users request for any page or click any link, the request directs the appropriate controllers through the routing system. Any values or parameters sent by the users are sent along with the request to the controller. The controller then uses the appropriate model with any values provided by the user if applicable. The model will connect to the database with the stored credential on the web server if necessary. When the model connects to the database, the necessary Structured Query Language (SQL) commands are executed and values from the database are passed to the model. The model then sends all the data to the controller which then displays the appropriate view page along with the output data.

5. TEST CASES AND SCREENSHOTS OF THE SYSTEM

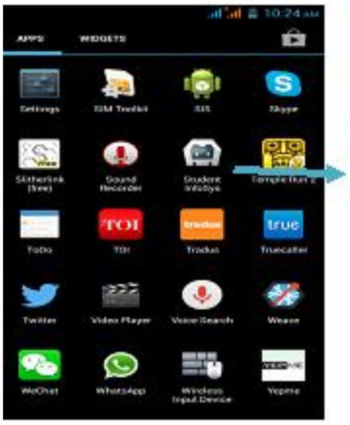
5.1 Test Cases

Table 4: Validation Table

Test Case No.	Test Case Description	Test Input	Expected Output	Actual Output	Result
1	To check valid User Name & Pass Word	User Name: anand Pass Word: anand	Home Page should be displayed	Window should be displayed	Success
2	To check empty User Name & Pass Word field & click on login button	User Name: Pass Word:	Alert message should be displayed	Alert message is displayed "Please enter username and password"	Success
3	To check valid User Name & empty Pass Word field	User Name: ntalele Pass Word:	Alert message should be displayed	Alert message is displayed "Please enter username and password"	Success

The above table 4 gives few of the test cases of the system. Application is tested on Android mobile (i Ball and i 4.5 K6) and fire fox OS simultor [12]. Following section gives few of the snapshots of a system.

5.2 Snapshots of the System



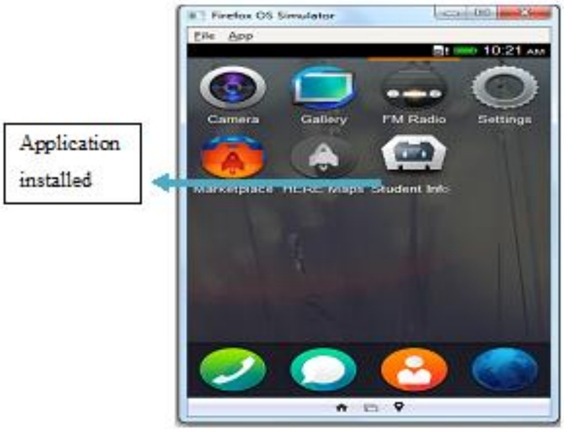
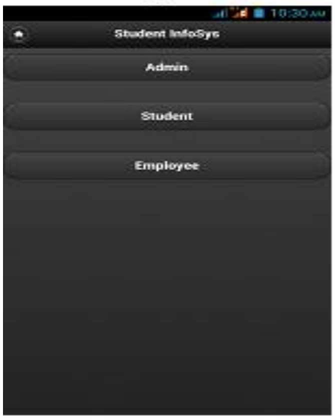


Figure 6: Snapshot from I Ball and i 4.5K6 Figure 7: Snapshot from Firefox OS Simulator



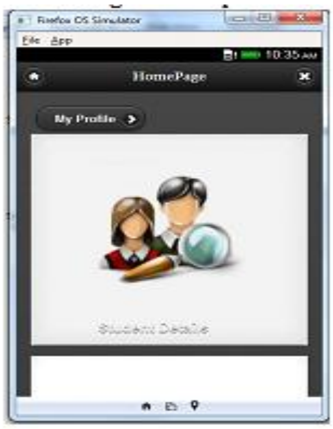


Figure 8: Individual Login Page for Users Snapshot from I Ball and i 4.5K6 **Figure 9: Employee Home Page from Firefox OS Simulator** **Figure 10: Student Can View Timetable**

6. CONCLUSIONS AND FUTURE SCOPE

The paper focused on building a cross-platform mobile application for mobiles and tablets, and then studying how to make the application cost-effective as well as saves time by using PhoneGap framework. The technology selected was the jQuery Mobile framework, HTML5, JavaScript, and CSS. Also, PhoneGap was analyzed to offer native features to the mobile application. If budget is not a constraint, native apps provide the best features and performance. "Develop once and run anywhere" is an important concept, but there are different approaches for achieving it. A mobile web app provides almost worldwide coverage for a relatively modest development effort. A cross-platform solution allows the similar code to be run on various platforms, dropping development and maintenance cost. Cross-platform frameworks can provide a richer experience while reaching the most strategically important mobile devices.

Future scope of the system is to improve look and feel of the user interface of the system. The look and feel of the UI is slightly inferior to that developed on a native mobile development framework. Cross-platform mobile applications are often criticized for this reason. However, it can be improved by experimenting with several front end frameworks to determine which is most suitable for the app. In addition to that, an online payment feature can also be added to the application. A verification method could be used as a web service in the application. Additionally, it could be made possible for the students to pay their fees through the application itself which would save the extra time.

REFERENCES

1. Avinash S, Anandkumar P, “To Study and Design a Cross-Platform Mobile Application for Student Information System using PhoneGap Framework”, www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 3, Issue 9, September 2013).
2. Weifeng S, Maofa W, “A Study on Educational Administration Inquire System based on Mobile Device”, CSAE, 2012 IEEE International Conference, Volume 3: Page(s): 392-395.
3. http://www.theseus.fi/bitstream/handle/10024/38060/Pradhan_Dipesh.pdf?sequence=1.
4. Manuel P, Inderjeet Singh, Antonio Cicchetti, “Comparison of Cross-Platform Mobile Development Tools”, 2012 16th International Conference on Intelligence in Next Generation Networks.
5. Andre R., Alberto R., “Survey on Cross-Platforms and Languages for Mobile Apps”, 2012 Eighth International Conference on the Quality of Information and Communication Technology.
6. http://public.dhe.ibm.com/common/ssi/ecm/en/wsw14182usen/WSW14182USEN.PDF.
7. http://publications.lib.chalmers.se/records/fulltext/167502.pdf.
8. http://phonegap.com/
9. Mahesh Babu, Manoharan, Kumar M.B., “Portability of mobile applications using PhoneGap: A case study”, IEEE.
10. http://phonegap.com/2012/05/02/phonegap-explained-visually/
11. Andreas Sommer, “Comparison and Evaluation of Cross-Platform frameworks for the development of mobile bussiness applications”, Master’s Thesis in Informatik 2012.
12. https://developer.mozilla.org/en-US/Firefox_OS/Using_the_App_Manager