Design a Cross Platform Mobile Application for Student Information System

Dr. G. Shankar Lingam1
Professor, Dept. of CSE, Chaitanya Institute of Technology & Science, Warangal, Telangana, India

ABSTRACT: In Institutions of better schooling, using mobile smartphone is greater than computer systems. With mobile devices, we will have “Information at our finger guidelines everywhere at any time”, so that users could have a higher experience to get right of entry to facts and revel in many applications. There is a threat to transplant academic affairs and facts from computer to mobile phones to improve service exceptional and management efficiency. The applications created via PhoneGap mobile improvement framework are cross-platform which enables avoid growing equal sort of application one at a time. This paper aims at sharing records between college students, team of workers, participants, departments and college management. The utility will help college students to check their effects, attendance, view private information, test for bulletins and many others. Staff members and college students can view their time table of lessons and additionally, offer more value-introduced services that are clean to apply without delay from mobile phones. This paper objectives to expand a Cross-Platform Mobile Application for Student Information System (SIS).

KEYWORDS: Cross-Platform, Information System, Mobile Application, SIS, PhoneGap, Web Service

I. INTRODUCTION

Mobile improvement nowadays is developing exponentially. Today each and every individual in this worldwide has a smart-phone in his pocket. Smartphone’s integrate more than a few functions including mediagamers, camera and GPS with advanced computing capabilities and touch panels are enjoying ever increasing popularity. Smartphone’s help us to attain a variety of tasks via something called packages or Apps to brief. According to Gartner [3], Google’s Android, Apple’s iOS and RIM’s Blackberry all have at least a 10% marketplace share. For completing this assessment paper and observe about this subject matter a total of 4 studies papers had been used which helped to recognize conceptual and contemporary scenario of Cross-platform cell software development.

CIDER[1-2] is essentially a working device compatibility architecture that can run packages constructed for different mobile ecosystems preferably iOS and Android collectively on the identical Smartphone or tablet. Basically in less complicated phrases CIDER had the capacity to run unmodified iOS binaries at the Android subsystem without any sort of modification. CIDER achieves the venture of increasing the capability of home Android kernel through concurrently the use of the space kernel and the slave kernel that is the utility binary interface in our case.

User area of the slave kernel gets in touch with the Cider enabled kernel in exactly the same approaches as the slave kernel. That is, the iOS programs get in to Linux based totally kernel approach as if they’re operating on at home kernel of iOS subsystem that is going for walks on a regular iOS based tool. Instance of a foreign kernel, and reuse and run unmodified foreign user space library code. Now coming to the structure of those two operating systems. iOS runs on ARM CPUs like Android, however has a very distinct software program ecosystem. IOS is built at the XNU [8] kernel, a hybrid aggregate of a monolithic BSD kernel and a Mach microkernel [5] walking in asingle kernel handle with space. When we communicate about Android, Each Android app is compiled into Dalvik [4] byte code (dex) layout, and runs in a separate Dalvik VM instance.

Now Comparison of latest cross-platform mobile software improvement strategies which might be currently to be had in the market. Some of the cross-platform mobile application development techniques are PhoneGap [3], Titanium and so forth. The distinguish between strategies that rent a run-time environment and those who generate platform-precise apps from a commonplace code base at compile time. The latter, generator based category consists of version driven
The improvement now takes area inside the Apache Cordova task of the Apache Foundation, of which Phone Gap is a distribution. Phone Gap a completely popular cross-platform mobile software improvement device is loosely primarily based on jQuery which is a totally fast library conversion tool. This research paper helped to analyse how different apparatus which can be currently to be had in the market function and interact with every different. Also the technology on which cross-platform cellular software development tools which might be presently to be had inside the marketplace are currently based on a number of those technologies which care currently very popular are HTML5, Javascript and open supply libraries such as jQuery[7] and jQtouch[6]. Thus developers can use an outstanding a part of their abilities to increase mobile programs. Mobile-internet applications are programs whichever is an example of mobile internet browser to run the utility. These are perfect for mobile websites liken.Yahoo.Com, m.Fb.Com. These applications are developed the usage of pass-platform SDK’s and open supply libraries including jQuery, jQtouch, and so forth. The user interface (UI) is developed in HTML5 and common sense is described by means of JavaScript. The very last deliverable is a fixed of files that can be hosted on an internet server and the application can be accessed the usage of any net browser which might be from a PC, Android device, iOS tool or a Blackberry device.

Hybrid cellular applications are a mixture of the preceding utility kinds. These applications are advanced the usage of open supply libraries however additionally have access to a number of the native competencies of a tool which includes Camera, GPS and so on. So in easy phrases, pass-platform mobile improvement with the assistance of taking instance of HTML5 primarily based web software which can be accessed from any form of Mobile Browser.

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.

II. RELATED WORK

Little research is available comparing CPDTs. Some comparison of features has been done in [9] and [11] although they lack some depth. The CPDT evaluators used a 13 item chart in [9] to allow comparison of tool features. Many important items like storage and camera access are covered in the survey. However, neither of these reports includes performance evaluation or discussion of development practices and detailed costs.

In [10], the authors discuss many CPDTs and provide partial comparison. Their work includes discussion of native versus web-based user interface elements and the importance of well performing applications. However, the authors state that they are not concerned with the internal workings of the tools and only if the applications will be approved for the application stores as the main development need. The article discusses the lack of debugging tools in many current CPDTs and provides an 8 point feature comparison. The authors develop a simple application that creates a screen with a text label and measured the RAM usage and start time for nine CPDTs.

The results show the quick launch time for the application built using the native Android SDK but a much slower start for other tools such as Titanium and PhoneGap. Runtime based CPDTs seem to fare the worst and have large RAM footprints. This information provides useful understanding of the possible performance differences with developing applications on the Android platform using CPDTs but does not include other platforms or more extensive tests beyond initialization. It is possible with further testing that an application may perform poorly for initialization but run remarkably well afterwards but this cannot be determined by this test.

The PhoneGap architecture is composed mainly of three layers: Web Application, PhoneGap, and OS and native API’s. In Fig. 2 the top layer represents the applicationsource code. The central layer is composed by JavaScript and native API’s. Mainly, this layer is responsible for the interfacing between web application and PhoneGap layers[12].
Furthermore, it also takes care of the interfacing between JavaScript API’s that are used by the application with native API’s that are used by mobile OS’s. The functionality of this layer is to maintain the relationship between JavaScript API’s and native API’s of each mobile OS. PhoneGap supports most of the mobile operating systems like iPhone, Windows Mobile, BlackBerry, Android Symbian, and WebOS which can be worked along with other operating systems like Linux, Mac and Windows. It also supports scripting languages like HTML, JavaScript, and CSS which makes it an easy tool to work with. PhoneGap provides JavaScript API’s to developers that allow the access to advanced device functionality, such as Accelerometer, Barcode, Bluetooth, Calendar, Camera, Compass, Connection, Contacts, File, GPS, Menu, NFC, etc. [13].

![Figure 2: interfacing layers of the phonegap architecture](image)

In Fig. 3 is shown a more detailed architecture schema provided by IBM. It represents all components about the web application, HTML rendering engine, PhoneGap API’s and OS layers. Moreover, some different interfaces are shown in detail, such as the interfacing between PhoneGap API’s and native API’s layers.

![Figure 3: complete schema of phonegap architecture and interfacing among components](image)

### III. SUGGESTED SCHEME

Our suggested system is based on cross-platform mobile application. Cross platform development targets on creating a single application which can be used across multiple platforms. This helps the application vendor to maintain the same code base for multiple platforms. Maintenance and release overheads for multiple platforms can be reduced by cross platform application development. Due to mobile devices, users can have their information accessible at anytime as well as anywhere. Our system has various features like user-friendly interface, fast access to database, more storage capacity,
look and feel environment etc. All the manual difficulties in managing the details of SIS have been rectified by implementing app on mobile device.

A. DESIGN OF SIS
PhoneGap 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 local language. As seen in Figure 4, the PhoneGap framework bridges the gap between the local language and HTML via running as a wrapper for the app and generating JavaScript used inside the app for gaining access to the native API. Due to jQuery and jQuery Mobile the utility is written in one HTML document thus allowing every page to be inside a div and figuring out them with different ID’s while used to navigate over the app. The content material proven within the app differs depending on what page the user has navigated to. The software is constructed with one-of-a-kind views that the person navigates through.

Figure 4: structure of a phonegap produced app [15].

The home screen lets the user see a top level view of all of the available pages the consumer can navigate to, see Figure 5. The paper intended for a familiar appearance and sense of the application in order that a user without problems can apprend the navigation through the app. With the assist of jQuery Mobile the design of the utility is comparable in icons and toolbars with most common packages. Within a web page, a header changed into created that includes a navigation bar, permitting the user to return lower back to preceding web page or cross at once to the home screen.

Figure 5: overview of the navigation through the app
As shown in Figure 6, SIS can be classified into three layers according to system functions. They are data layer, business layer and presentation layer.

**A. DATA LAYER**
This provides basic data for the upper layers and stores the data into the database system, including timetable, attendance, students and staff profiles, announcements etc.

**B. BUSINESS LAYER**
This provides business logic and functions for storing and retrieving data from the database, and works as a mediator between the Presentation layer and Data layer.

**C. PRESENTATION LAYER**
This provides the user interface that allows the user to interact with the application. PhoneGap application acts as a client for the user to interact with. PhoneGap client communicates with an application server to receive data. The application server handles business logic and communicates with a back-end data repository. The application server is normally a webserver (Apache, IIS, etc...) and has a server side scripting language such as ColdFusion, Java, .NET, PHP, etc. PhoneGap is agnostic of back-end technologies and can work with any application server using standard web protocols [25]. The application server performs business logic and calculations, and generally retrieves or persists data from a separate data repository - this is normally a relational database, but could be any structure or mechanism for data persistence. PhoneGap applications cannot communicate directly to a database; communication is routed through an application server. The client to application server communication can be based upon standard HTTP requests for HTML content, RESTful XML services, JSON services, or SOAP. These are the exact same techniques that can be used for a desktop-browser based AJAX application.

![Figure 6: Architecture of SIS.](image)

The client-side architecture generally uses the single-page application model, where the application logic is inside a single HTML page. This page is never unloaded from memory. All data will be displayed by updating the HTML DOM, data is retrieved from the application server using AJAX techniques, and variables are kept in memory within JavaScript.

**IV. CONCLUSION**

In this paper, we propose a go-platform utility development using PhoneGap framework. This paper introduces the study and layout of Student Information System based totally on mobile tools so that we can enhance the control efficiency and service quality. The device is easy to install, secure and convenient to apply. The practical cost of mobility in teaching...
can be extra in the future because mobile devices are flexible, easy to apply in actual time. PhoneGap is used for cross-platform development of mobile apps for multiple structures by using the usage of general web development technology.

REFERENCES

[5]. D Keuper, (s1019775, XNU a security evaluation), University of Twente and Certified Secure, December 13, 2012.
[7]. http://source.googlecode.com
[8]. https://developer.apple.com/scode

BIOGRAPHY

Dr. G. Shankar Lingam completed his MCA in Chaitanya Degree & P.G College and M.Tech in CSE from Ramappa Engineering College respectively. He is having teaching experience of more than 20 years in various Under Graduate and Post Graduate courses. He has guided lots of students in various Under Graduate and Post Graduate Research Projects. At Present, he is working Professor, Dept. of CSE, Chaitanya Institute of Technology & Science, Warangal, Telangana, India.