ASEE's 123rd Annual • Conference & Exposition • New Orleans, LA • June 26-29, 2016

Paper ID #15515

Develop a New Mobile-Optimized Remote Experiment Application for Mobile Learning

Mr. Qianlong Lan, Texas Southern University

Dpt. of Computer Science Graduate Student

Mr. Ning Wang

Dr. Xuemin Chen, Texas Southern University

Dr. Xuemin Chen is the founding Director of Virtual and Remote Laboratory and an Associate Professor of Electrical and Computer Engineering at the Texas Southern University.

Dr. Gangbing Song, University of Houston (CoE)

Dr. Hamid R. Parsaei, Texas A&M University at Qatar

Hamid R. Parsaei is a Professor of Mechanical Engineering and Director of Academic Outreach Office at Texas A&M University at Qatar. Dr. Parsaei is also a Professor of Industrial and Systems Engineering and Mechanical Engineering in Texas A&M University in College Station. Dr. Parsaei is a Fellow of American Society for Engineering Education and Fellow of Institute of Industrial Engineers. He has published over 250 articles in the refereed archival journals and conference proceedings. Dr. Parsaei is a registered professional engineer in Texas.

Develop a New Mobile-Optimized Remote Experiment Application for Mobile Learning

Abstract

As Mobile Learning (M-Learning) has had a great impact on online education, more and more mobile applications are designed and developed for the M-Learning. In this paper, a novel mobile-optimized application architecture is proposed to integrate the remote laboratory into mobile environment for the M-Learning. With this mobile optimized application architecture, the remote experiment applications can use a common codebase to deploy native-like applications on many different mobile platforms (such as, iOS, Android, Window Mobile, etc.). To demonstrate the effectiveness of proposed new architecture for M-Learning, an innovative remote networked Smart Vibration Platform (SVP) experiment is successfully implemented based on this new application architecture. This remote SVP experiment has been used in several engineering courses.

Introduction

During recent decades, the rapid development of communications and wireless technologies have resulted in mobile devices (e.g., PDAs, smartphones) becoming widely available, more convenient, and less expensive[1]. Mobile learning (M-Learning)[2] which is the delivery of learning, education or learning support on mobile phones, PDAs or tablets has played an important role in E-Learning. According to research by Global Industry Analysts (GIA) published in 2014[3], it states that the "E-Learning market is one of the most rapidly growing sectors in the global education industry." E-Learning, which was instituted on desktops, in the beginning, has gradually shifted its base to portable tablets and smartphones[4]. Thus, more and more learning approaches, learning systems are configured and integrated for Mobile learning[5]. For example, a Microlecture Mobile Learning System at Guangdong University of Technology[6], a smart learning mobile system for collaborative M-learning at BaekSeok Culture University[7], etc.

Nowadays, M-Learning is a growing trend in classroom. This trend is creating a more flexible, collaborative and interactive learning experience in schools and districts, and according to a report released by Simba Information, over 75% of districts are using mobile technology for educational purposes[8]. More and more mobile devices like tablets, and smartphones being used by students and educators in schools, districts and individual classrooms. According to a recent report by EdNET Insight[9], 82.2% of districts surveyed said they would consider implementing laptop computers, 68.7% said they would consider implementing tablets, and 56.7% said they would consider implementing iPod or iPod touch to deliver digital instructional materials. According to App Store Metrics, the iTunes App StoreSM currently has over 90,000 education apps[10]. Consequently, M-Learning continues to be a major technology trend as we move in future.

Currently, for the remote laboratory applications development, more and more remote laboratory software systems have selected web services technology and Service Oriented Architecture (SOA) to implement the Browser-Server (B/S) architecture remote laboratory[11,12]. To integrate the remote laboratory application to mobile devices (e.g., PDAs, smartphones) for M-Learning, there are two common approaches to achieve the goal. 1) Web application for mobile devices: web applications normally based on the web browsers can implement cross-platform interfaces to perform the remote experiments, but it is difficult to achieve better user experience. 2) Native application for mobile devices: Although the native remote laboratory applications developed for different mobile platforms (such as Apple iOS, Android, Window Mobile, etc.) can achieve better interactive user experience, it is hard to implement the cross-platform interface[13,14]. How to design and implement a mobile optimized and easy-to-use application for M-learning already becomes a hot topic[15,16]. Consequently, it is an essential issue how to integrate the remote laboratory application for the M-learning.

To resolve this essential issue and improve the unified framework[17,18,19] to better support the M-Learning, an advanced HTML5 (Hyper Text Markup Language) Ionic framework, is integrated into the novel easy-to-use unified framework for remote laboratory development. With the Ionic framework, we successfully upgraded the unified framework to support M-Learning. In this paper, a mobile optimized application architecture based on the unified framework is proposed to support M-Learning. Meanwhile, a new mobile-optimized remote SVP experiment application as a case study is successfully implemented for M-Learning. With the new remote experiment mobile-optimized applications enabled by the improved unified framework, students can operate or view the real-time remote experiments via the portable devices and smart phones anywhere at any time. This improved unified framework will significantly benefit future remote laboratory developments.

Methodology

As the unified framework has solved several critical issues to improve the remote experiment performance and user experience, the remote laboratory based on the unified framework can provide the students real-time video and real-time data transmission without software plugins and the firewall issue[18,19]. To integrate the advantages of unified framework to mobile environment for better supporting M-Learning, a good and stable mobile framework must be selected as the foundational platform. With the HTML5 technology, more and more different application platforms and frameworks for building mobile applications emerge in endlessly. As the Table 1 shows, compare with other platforms and frameworks, the Ionic framework has the unique advantage. Ionic platform enables high native-like performance, around 70%. Meanwhile, it can be supported by both of two development environments, Apache Cordova platform and AngularJS framework. Based on Apache Cordova and AngularJS, the mobile optimized applications, which are developed by JavaScript language, can be built without any native code (e.g., Java, Objective-C) in mobile devices[20]. In this way, Ionic framework can be used to implement interactive mobile applications with the unified framework smoothly on different mobile operation systems (e.g., iOS, Android, Blackberry, Windows Phone, Palm, Web OS, Bada, and Symbian.). According to these benefits, the Ionic is the best candidate for the foundational platform of mobile optimized application architecture.

Table 1: Comparison Between Different Frameworks

Framework	Native-Like Performance	Support Cordova/AngularJS
Ionic platform	7/10	Yes/Yes
Onsen UI	6/10	No/Yes
Framework 7	8/10	No/No
React Native	8/10	No/No
jQuery Mobile	3/10	No/No
Native Script	8/10	No/No

The Mobile Optimized Application Architecture for Remote Experiment

The system architecture shows in Figure 1. We divide the architecture into two parts: the unified remote experiment framework layer, and Ionic platform layer. This unified framework provides the LtoN (LabVIEW to Node.js) technology and real-time video technology.

- LabVIEW to Node.js Technology. To achieve the experiment data transmission and experiment control commands transmission, LabVIEW to Node.js (LtoN) technology was implemented on server side. The LtoN is one of the technologies used for developing web-based remote experiment with LabVIEW. With this new technology, students can conduct the experiment, and save the experiment data in a real-time manner. To integrate the LtoN protocol to the optimized mobile application architecture, some improvement need to be implemented. 1) The new application communication protocol includes two parts, client part runs in Crosswalk rendering engine and server part runs in web server. 2) To implement real time experiment control commands and experiment data transmission by defined special communication instruction set. With the new communication instruction set, we can secure the data communication when user is conducting the remote experiment. 3) Some brief instructions to control experiment progress are designed to improve the experiment data transmission performance.

- Real-Time Video Technology. The purpose of this new approach is to improve the remote experiment video transmission function. The optimized remote experiment application's performance are supported by real-time video technology. In order to achieve these goals, a new video transmission approach via HTTP Live Streaming (HLS) protocol with FFMPEG, which is a powerful cross-platform command line video trans-code/encoding software package. The real-time video segments are transferred via HSL protocol, and will be reassembled in the WebView rendering engine and presented to end users.

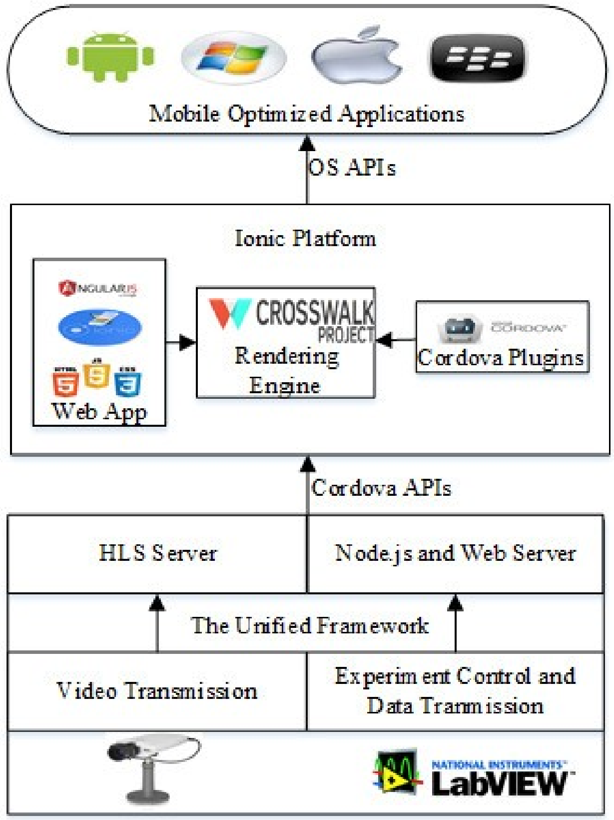


Figure 1: System Architecture

The unified framework layer is directly built on the top of a novel assembled server engine scheme. It includes two server engines, Apache HTTP server engine and Node.js server engine. With the server-based Mashup technology, the Apache HTTP server engine is used to combine the UI widgets and web content (such as, the real-time experiment data, the real-time experiment video, etc) together. Meanwhile, the Node.js web engine handles the real-time experiment data transmission.

The Ionic platform includes the Crosswalk Project, HTML5 (HTML, CSS, JavaScript) and Apache Cordova plugins.

- The Crosswalk Project. To enhance the mobile optimized application performance, Crosswalk project, as the rendering engine, is integrated into the mobile optimized application. To solve the fragmentation mobile Operating System (OS) like Android, the Crosswalk Project can automatically update the rendering engine.
- HTML5 Technology. Ionic platform emerges as the first full-stack service platform to build and scale the mobile applications with HTML5 technology. It also offers a library of mobile optimized HTML, CSS and JS components, gestures, and tools to build highly interactive applications.
- Apache Cordova. Apache Cordova is an open-source mobile development framework. To avoid each mobile platforms' native development language, it allows developer to use

standard web technologies (such as, HTML5, CSS3, and JavaScript, etc.) for cross-platform development. Mobile applications are executed within wrappers on different platforms, and they rely on the standard Application Program Interfaces (APIs) to access the different device's sensors, data, and network status. To maximize the native mobile devices hardware capabilities the Apache Cordova framework, as a plugin, is integrated into the new application architecture.

In application implementation process, the common codebase is wrapped into the Cordova framework, and rendered in the UI layer with the Webview rendering engine. Normally, the common codebase is reusable to deploy the mobile optimized application to the different mobile platforms. The Webviev works as a middleware to connect the Ionic platform with the unified framework. With these key technologies, all of the source code and software plugins are packaged with a new mobile optimized remote experiment application. To package the new mobile optimized application, the Eclipse, a popular Integrated Development Environment is needed. Table 2 shows the characteristic of the mobile optimized application for remote experiment.

Table 2: Characteristic of the Mobile Optimized Application for Remote Experiment

Characteristic	Technology
Cross-Platform	Apache Cordova
Code Reusable	HTML/JavaScipt/CSS/WV (Web View)
Fragmented OS Compatible	Crosswalk Runtime Engine
High Performance	Ionic Framework/AngularJS
Real-Time Experiment	The Unified Framework

The flowchart of the mobile optimized application architecture shows in Figure 2. Optimized application communicate with server via HTTP protocol. The user interface (UI) presented by Webview rendering engine, experiment control command will transfer to server via Web server protocol and processed by server. Finally users can conduct the remote experiment or view the video in a real-time manner. Experiment data and control command directly available to end user provides the flexibility and convenience of M-Learning.

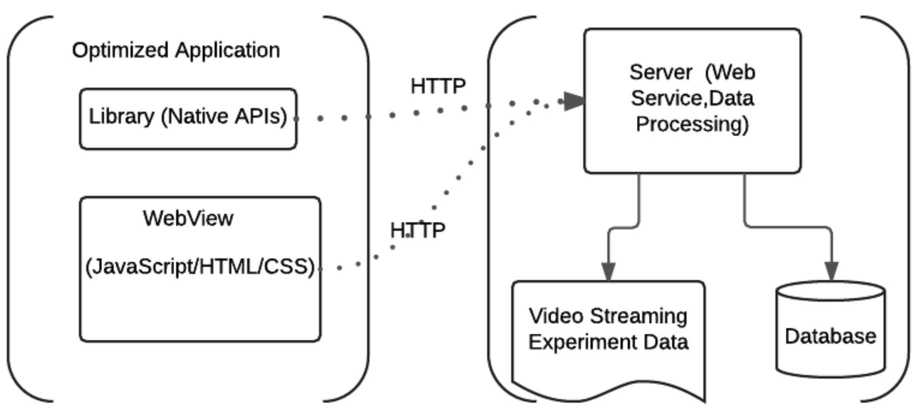


Figure 2: Flowchart of the Framework

Innovative Improvements for M-Learning

By explored higher educational learning practices with mobile technologies and focused on the interactions among technologies, contents and pedagogies. K-12 student access only few devices and search for usage of mobile technologies that create mobile-optimized version of their website or build stand-alone applications are the obstacles limit engaging in M-Learning. The learners need more access to academic-friendly devices and additional support to integrate mobile technologies for learning. The mobile-optimized application solutions in this paper provide both the cross-platform friendly devices and integrate mobile technologies for M-Learning. Convenience, flexibility, engagement, and interactivity are greatly improved to make M-Learning more attractive to students. Particularly, it shortens the gap between formal learning in classroom and informal learning outside the classroom. Learners can engage learning effectively at any time with any devices. Remote experiment can be seen as training courses for students contributes to students' engagement and learning. Teaching in K-12 curriculums, instructor can greatly improve the student's engage learning. Finally, technology use is further influenced by the modality of courses in which it is used. Understanding students' mobile practices more deeply can guide instructor development in the future.

The Implementation of Optimized Remote SVP Experiment Application

Initially, remote SVP Mechanical Engineering experiment, was incorporated as part of the remote laboratory series of the Intelligent Structural Systems (ISS) course. Block diagram of the LabVIEW program for SVP remote experiment shows in Figure 3. Smart materials in structural applications with a strong emphasis on vibration control was introduced to graduate students with this course. To successfully develop the optimized remote SVP experiment application, there are four parts need to be implemented.

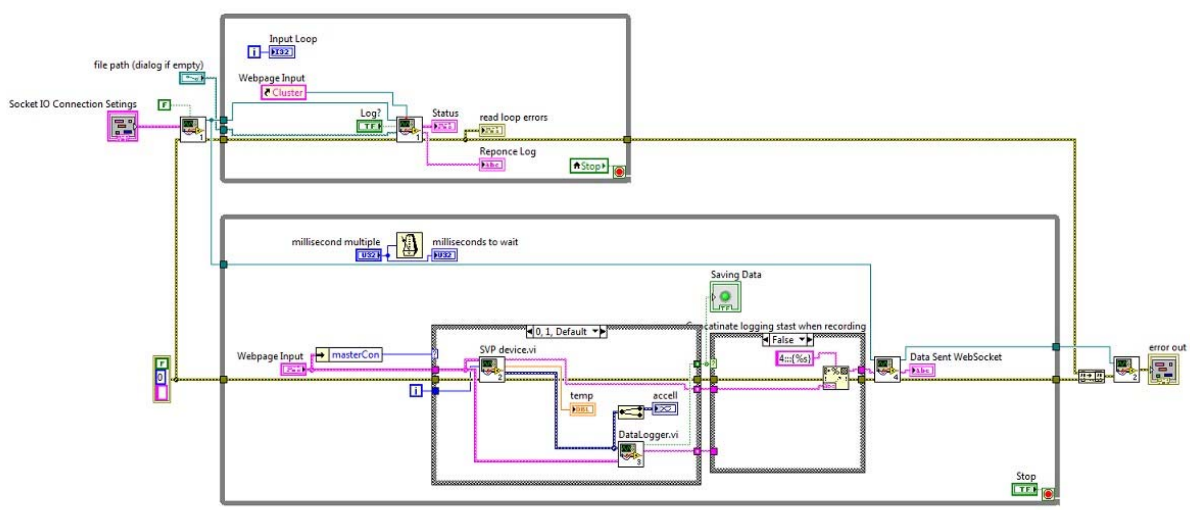
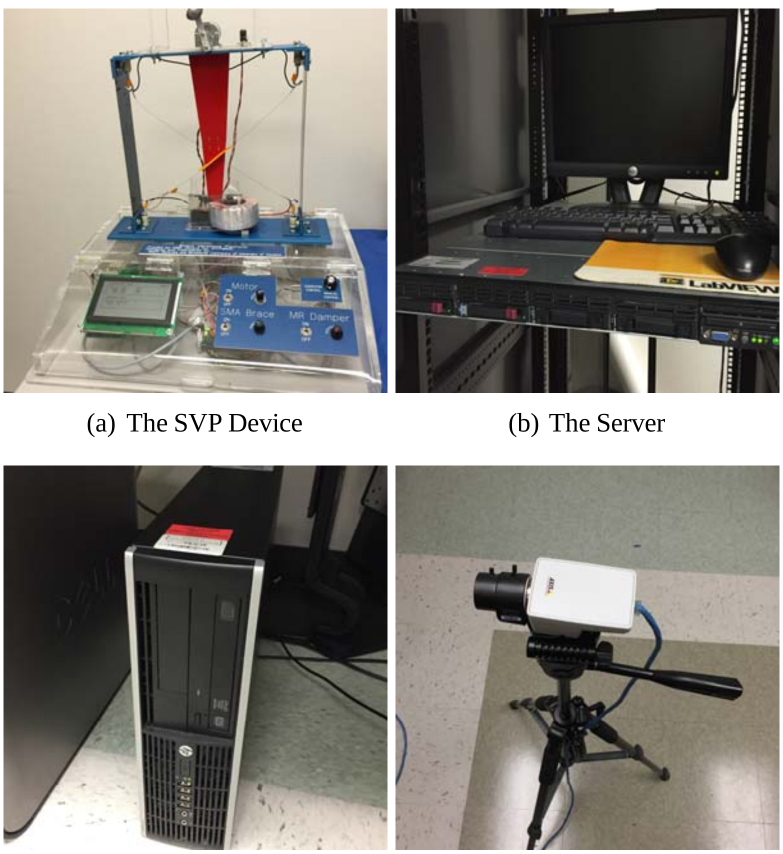


Figure 3: Block Diagram of the LabVIEW Program for SVP Remote Experiment

- The SVP Device. The SVP as shown in Figure 4 (a), has a two-story flexible steel frame fixed on top of a plexi-glass box. In the plexi-glass box, there are electric circuit boards made to control the experiment. It is designed and built by students in the Smart Materials and Structures Laboratory at the University of Houston. Other than the flexible steel frame, the SVP has a motor, SMA wires and a purchased magnetic iron clamped on a container of Magneto-Rheological (MR) fluid. The motor with a weight is mounted on the top of the frame and connected to the driver from the box on the bottom. When the user controls the current going through the electrical circuit, the speed of the motor can be adjusted. The rotation of the motor leads the flexible frame to vibrate. Two SMA wires are hung across the frame. When the current goes through the wires, the temperature will increase. At a certain point of rising temperature, the SMA wires will shrink in length to reduce the vibration of the frame; this is called a SMA brace. A red steel tongue is placed downwards into the container of MR fluid. The magnetic iron clamped on the container can generate magnetic field when it is turned on. That increases the viscosity of MR fluid,because MR fluid changes from fluid state to semi-fluid state under the magnetic field; this is called MR damper.



(a) The SVP Device (b) The Server

(c) Workstation (d) Axis Camera

Figure 4: Physical Devices

- Server Setup. There are three server engines that installed at HP ProLiant DL380e Gen8 which shows in Figure 4 (b): The HLS server engine, Node.js server engine and HTTP server engine. HLS server engine includes video trans-code/encoding software package which ensures the real-time video transmission. The Node.js server engine, a LtoN module that supports real time experiment control commands and experiment data transmission. Another is the Node-HTTP-Proxy software package installed on the server side, to achieve the experiment data across the network firewalls via network port 80. The HTTP server engine, provides web services to end users throughout the Internet.

- Client Side Setup. LabVIEW 2012 (32-bit) is installed on workstation shows in Figure 4 (c).Three DAQ 6008 USBs are connected to the workstation, and their voltage outputs and voltage inputs are controlled and sensed by programed code in LabVIEW running on workstation. The workstation is also connected to our web server via a Network Switch.One Axis Camera connected with workstation to record the video streaming which shows in Figure 4 (d). All data generated by LabVIEW is sent to the server. Control commands for the experiment from the Internet are also sent from the server to the workstations.
- Application Development. Application development commonly based on the Application Programming Interface (APIs) from Ionic, Apache Cordova and the Crosswalk Project. The common codebase (HTML, CSS, JavaScript) finally extended and wrapped into the Crosswalk rendering engine then presented to end user interface. The integrated development environment (IDE) is Eclipse which used for packaging the applications.

Sample Usage of the Mobile-Optimized Application

The new optimized mobile application is successfully implement for M-Learning on different smart phone platforms. Figure 5 depicts the SVP experiments optimized applications. The

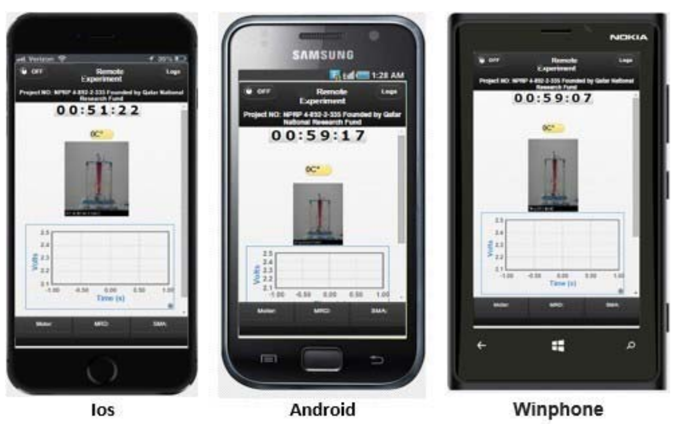


Figure 5: Remote SVP Optimized Applications for M-Learning

platform uses shape memory alloy (SMA) springs, whose stiffness can be adjusted by electrical hearting, and a magneto-rheological (MR) fluids damper, whose damping ratio can be controlled by an electromagnet. Students can remotely control the SVP experiment at any platform with better performance, which also implemented with real time video and data transmission. So the development of the SVP experiment into portable optimized applications, which utilize the mobility, flexibility and convenience of M-Learning. SVP was incorporated as part of the remote laboratory series used in the Intelligent Structural System (IIS) course. This course, which included both senior undergraduates and graduate students, introduced students to the applications of smart materials in structural applications with a strong emphasis on vibration materials, as well as the mechanical principles governing the control of such materials. After completion of the experiment, the students needed to submit a report which included analysis of the experimental data. Convenience, flexibility, engagement, and interactivity of remote experiment are greatly improved for M-Learning. So this optimized mobile application architecture can significantly enhance student learning involving dynamic systems, mechanical and structural vibrations, and vibration controls in multiple departments, including mechanical engineering, civil engineering, and engineering technology.

Conclusion

In this paper, an optimized mobile application architecture was presented. One optimized remote experiment application for M-Learning, SVP was successfully implemented under this new architecture. With remote experiment optimized application solution enabled by the optimized mobile application architecture, users can perform or view the real-time remote experiment on most popular mobile devices with better performance. Remote experiment optimized application improves the usability, flexibility and convenience of M-Learning. This research will improve the M-Learning activities as well as significantly benefit the remote experiment development.

References

[1] George Papagiannakis, Gurminder Singh, and Nadia Magnenat-Thalmann. A survey of mobile and wireless technologies for augmented reality systems. *Computer Animation and Virtual Worlds*, 19(1):3–22, 2008.

[2] Wikipedia. M-learning — wikipedia, the free encyclopedia, 2016. URL `https://en.wikipedia.org/w/index.php?title=M-learning&oldid=699450856`. [Online; accessed 27-January-2016].

[3] Global Industry Analysts. A global strategic business report. global industry analysts. *Inc., USA*, 2011.

[4] Agnes Kukulska-Hulme. Smart devices or people? a mobile learning quandary. *International Journal of Learning*, 4(3-4):73–77, 2012.

[5] Mohamed Ally and Josep Prieto-Blázquez. What is the future of mobile learning in education? *RUSC. Universities and Knowledge Society Journal*, 11(1):142–151, 2014.

[6] Cheng Wen and Juyong Zhang. Design of a microlecture mobile learning system based on smartphone and web platforms.

[7] Daedeok-gu Ojeong-Dong. International journal of software engineering and its applications. 2003.

[8] Simba Information. Going mobile in the prek-12 school market, 2011. URL `http://www.simbainformation.com/Going-Mobile-PreK-6055405/`. [Online; accessed 18-March-2016].

[9] EdNET Insight. How personal technology is transforming the classroom, 2012. URL `www.ednetinsight.com/insight/reports?page=2`. [Online; accessed 18-March-2016].

[10] PocketGamer. App store metrics, 2012. URL `http://www.pocketgamer.biz/metrics/app-store/`. [Online; accessed 18-March-2016].

[11] Javier García-Zubia, Pablo Orduña, Diego López-de Ipiña, and Gustavo R Alves. Addressing software impact in the design of remote laboratories. *Industrial Electronics, IEEE Transactions on*, 56(12):4757–4767, 2009.

[12] Mohamed Tawfik, David Lowe, Christophe Salzmann, Denis Gillet, Elio Sancristobal, and Manuel Castro. Defining the critical factors in the architectural design of remote laboratories. *Tecnologias del Aprendizaje, IEEE Revista Iberoamericana de*, 10(4):269–279, 2015.

[13] Andre Charland and Brian Leroux. Mobile application development: web vs. native. *Communications of the ACM*, 54(5):49–53, 2011.

[14] Antero Juntunen, Eetu Jalonen, and Sakari Luukkainen. Html 5 in mobile devices–drivers and restraints. In *System Sciences (HICSS), 2013 46th Hawaii International Conference on*, pages 1053–1062. IEEE, 2013.

[15] Daniel Ivanc, Radu Vasiu, and Mihai Onita. Usability evaluation of a lms mobile web interface. In *Information and Software Technologies*, pages 348–361. Springer, 2012.

[16] Saskia Teri, Anita Acai, Douglas Griffith, Qusay Mahmoud, David WL Ma, and Genevieve Newton. Student use and pedagogical impact of a mobile learning application. *Biochemistry and Molecular Biology Education*, 42(2):121–135, 2014.

[17] Ning Wang, Xuemin Chen, Gangbing Song, and Hamid Parsaei. Remote experiment development using an improved unified framework. In *World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education*, volume 2014, pages 2003–2010, 2014.

[18] Ning Wang, Xuemin Chen, Gangbing Song, and Hamid Parsaei. A novel real-time video transmission approach for remote laboratory development. *International Journal of Online Engineering (iJOE)*, 11(1):pp–4, 2015.

[19] Ning Wang, Xuemin Chen, Gangbing Song, and Hamid Parsaei. Using node-http-proxy for remote experiment data transmission traversing firewall. *International Journal of Online Engineering (iJOE)*, 11(2):pp–60, 2015.

[20] R Mahesh Babu, M Balaji Kumar, Rakesh Manoharan, M Somasundaram, and S Prabhakar Karthikeyan. Portability of mobile applications using phonegap: A case study. 2012.