Systems analysis of webpage integration frameworks into mobile devices' applications

Nikolajs Bumanis

Latvia University of Agriculture, Faculty of Informational Technology , Liela Str. 2, Jelgava, LV-3001, Latvia
my.way@inbox.lv

Abstract: *Web technologies allow developing fully functional mobile devices' application with basic knowledge in web development technologies, such as HTML, CSS and JavaScript. The evolution trend of the web technologies in mobile application development led to accepting the cross-platform development approach as the main goal for almost any nowadays projects. Nowadays, developers are able to choose between multiple web technology based frameworks and libraries not only for building the application itself, but also for ensure full customization of application's content, UI styling and functionality. After completing the research was determined that every other real-life project will require a bit different approach at choosing the correct technologies, and it is not purely one-sided decision.*

Keywords: PhoneGap, RhoMobile, jQuery Mobile, Sencha Touch, cross-platform, mobile development frameworks.

Introduction

Nowadays while developing the mobile devices' applications (further in text – application) developers stand before choice problems, associated with selection of web technologies and combinations of them, as well as with selection of the development path itself. Existing web technologies and solutions based on them, such as frameworks and cloud builders, allow developing very specific mobile devices' applications, as well as mass market oriented applications. Often, every other project is able to use new web technologies to fulfil the needs of project, based on applications markets' objective audience, exploitation objectives and business collaboration elements. Business collaboration possibility and constraints can hardly influence the development process, as well as selection of development technologies.

One of the main aims of application development is to ensure applications exploitation availability for m-commerce (Alqahtani, 2012). Thanks to m-commerce product and service distributors can expand their activities, which are limited by client satisfaction factor, by providing new products and services to their clients, based on client deal history. It is commonly accepted, that client gets to know about new product or service from distributor, but the product or service itself could be externally provided and integrated from provider side. By business means exists a problem, when provider focusing on one individual service or product delivery or development often has relatively small client base, which leads to applications development being inconvenient, because expenses on development itself would be much higher than final products potential exploitation profit. Distributor, using their client base can supply providers' products and services to their clients, where chosen providers' products and services are strictly associated with client deal history. Formally, this type of business collaboration is highly attractive for providers, because of possibility to expand on the market, by physically just paying compensation for client base usage.

For m-commerce purpose and for collaboration between distributors and providers application must be developed, so that it could fully satisfy developers' and providers' needs by not only functionality requirements, but development and collaboration expenses' means as well. Application design must live up to client requirements from such view points as attractiveness, usage simplicity. Applications' security and accessibility are one of the main factors as well (Alqahtani, 2012).

Applications can be developed using existing development frameworks, or simply web technologies, which allow getting required results. As popular web development frameworks can be stated: xUI, PhoneGap, RhoMobile and ApplicationCraft (Corral, 2012b; Firtman, 2012). Frameworks are based on commonly used web technologies, such as HTML5, JavaScript and CSS. For applications user interface styling are used frameworks or libraries, such as jTouch, Jo, jQuery Mobile and Sencha Touch (Firtman, 2012).

The aim of the research is to decide, which web technologies or frameworks are more suitable for application development, where applications consist of multiple web pages integrated together, and every web page is developed by distributor or provider, to meet the needs of both sides. In the framework of paper, cross-platform development approach, two popular frameworks – PhoneGap and RhoMobile and two user interface visualization and configuration javascript libraries – jQuery Mobile and Sencha Touch will be analysed, and finally, the criteria for webpage integration technologies selection will be created.

Materials and methods

Concept of crossplatform frameworks

For application development three main development paths can be selected, each defines the final products' functionality, visual styling, security and accessibility. Application can be developed in three ways – as native mobile application, as web application or as hybrid application. Each of these ways has its own strong and weak points. Native application is the best in sense of functionality, precision and overall user experience, but development of such application needs a lot more time and skill, as well as mobile platform specific development skills. Web applications are very fast and easily developed, but have highly constrained functionality and security possibilities. Third option – hybrid application (Fig. 1), are objected to use strong features of two first ways, by combining native development containers with fast and easy web coding (Beckman, 2012).

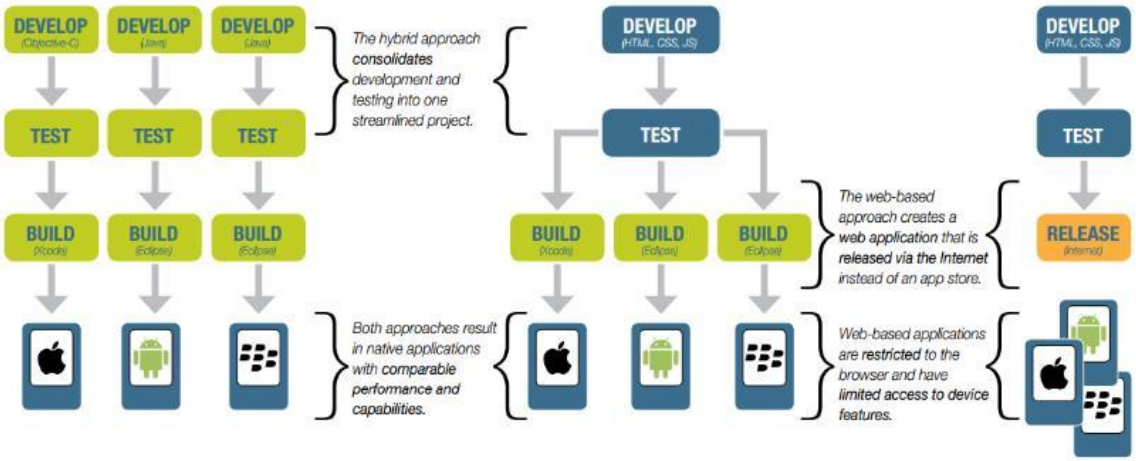

Fig. 1. **The continuum of mobile application development** (Christ, 2011)**.**

While developing application developers must take in mind mobile device's limitations (resources, input possibilities, screen features and etc.), must realize marketing requirements for applications' deployment and modelling, as well as distributors, providers, technologies providers and end user relations (Corral, 2012a).
Nowadays platform dependent applications are common, almost all Apple applications are natives. Developing application for specific platform allows developers to use full functional potential of mobile device, best user-system interaction, while lowers the possible user count because of clients, who use different platforms. It is possible to increase user count by developing the same application for another platform, or by porting the native application using objective platforms' API. Increment of user count using this method requires a lot of time and financial investments. Typical development process (Fig. 2.a) requires native code translation into every other platform's understandable programming language and low level deployment modification, so mobile device can recognize application as native. When coding is finished, application is rebuilt to accept required execution procedure way (Corral, 2012a).

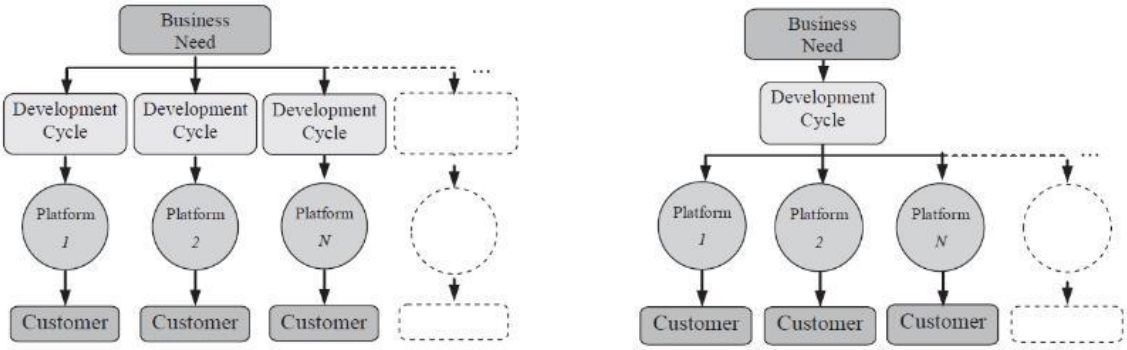

Fig. 2. (a) **Traditional development path**; (b) **Cross-platform development path** (Corral, 2012a).

Thanks to the latest development evolution trends development frameworks were created and they allow adapting existing application to, essentially, almost every popular mobile platforms (Mobile frameworks, n.d.) by going through only one development cycle (Fig. 2.b.). Essentially, using cross-platform development path developers are going through only one development cycle and result of that is deployed on required platforms by choosing objective specified adaptation (Corral, 2012a).

Idea of developing application only once and then deploy on every required platforms has great potential (Duarte, 2011). The only downside is that applications created with the use of frameworks are working on middle layer, what mobile platform is not used to do, thus applications' performance is relatively low compared to native application (Duarte, 2011). Despite of web technologies frameworks' drawbacks, using those it is possible to develop functionally better application then web application, because frameworks allow using mobile devices' features.

PhoneGap framework

PhoneGap framework is one of the hybrid application development tools, which uses before created HTML5 packages and expands HTML5 features using APIs, which grant access to mobile device's specific features, such as camera, geolocation and compass.

PhoneGap is free open code mobile framework which supports six most popular used mobile platforms - iOS, Android, BlackBerry, Windows Phone, Symbian and Bada (PhoneGap supported features, n.d.). PhoneGap allows to authorize native mobile application by using well known programming languages – HTML, CSS and JavaScript, allows connection to mobile device's specific features using platform independent JavaScript API and rebuilds application in the way, so it can be deployed and distributed using multiple different web stores and markets (Dingson, 2011). PhoneGap is package of libraries with the main purpose to "build bridge" between native mobile platform API and JavaScript "hook" – the same for all mobile platforms API is webpage's view – some way of representing webpage on the mobile device's screen(Building mobile apps, 2011). PhoneGap's approach is to use mobile device's web browser as an abstract intermediary layer, which allows implementation on JavaScript based logic layer and presentation layer, built on HTML and CSS. Same as desktop processing, this structure is easy to transfer on different web browsers; however this allows creating only script based applications, working web browser runtime, and thus, JavaScript cannot fully use mobile device's features (Corral, 2012b).

To solve this problem PhoneGap offers API package, which allows managing lower level components, such as telephone handling and hardware features, by using native engine (Corral, 2012b). API becomes available to JavaScript after it is loaded into web browser through PhoneGap's JavaScript engine. Thus, developers must rely on using web technologies (HTML5, CSS, JS) and web programming, because logic layer will count every excessive extension and interface, which influence the overall application resource consumption and mobile device's performance (Fig. 3).

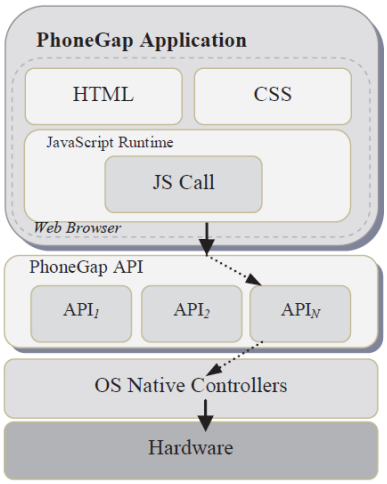

Fig. 3. **PhoneGap application architecture** (Corral, 2012b).

PhoneGap advantages:

- Free (whole framework is open source project);
- Ability to deploy a single codebase on multiple platforms;
- Interfaces with mobile device's hardware such as camera and GPS by using JavaScript;
- Integrated with native first code (application based web browser's object);
- Support all major mobile platforms;
- Applications are built using common web technologies such as HTML and CSS;
- Cloud based compiler;
- Purchased by Adobe (great support and large community).

PhoneGap disadvantages:

- Application is rendered using platform's web browser engine, not individual native UI objects;
- Requires plug-ins for some mobile device's hardware functionality and data security (Keychain plug-in);
- Each mobile platform requires installing appropriate SDK
- Different installation process depending on mobile device's platform (iOS and Androis, especially (Christ, 2011));
- Purchased by Adobe (Adobe doesn't have any open-source products, however PhoneGap code is published under GPL licence).

Best suitable for:

- Products, which require higher customization options' and development's control;
- Products, which require high number of supported mobile devices' platforms.

RhoMobile Rhodes framework

Rhodes is locally executed, device-optimized mobile applications' framework. Applications developed with Rhodes aim to collaborate with the support of enterprise deal application and is developed (Fig. 4) with the ability to synchronize data in local database, for example, SQLite or HSQLDB, or using overall synchronization support framework – RhoConnect, which is another RhoMobile product with partially free open-source. Rhodes supports such mobile platforms as BlackBerry, iOS, Android and Windows Phone (Rhomobile doc.., 2013; Mobile frameworks.., 2012). Rhodes uses Ruby programming language as base for its projects; combined with MVC (Model – View – Controller) pattern it allows creating very well structured projects. Rhodes also has imbued code generation tool, which allows creating applications without deep knowledge of Ruby language (however, high level projects still need a team of professional Ruby programmers). Same as PhoneGap, excluding Ruby, Rhodes is based on HTML, CSS and JavaScript.

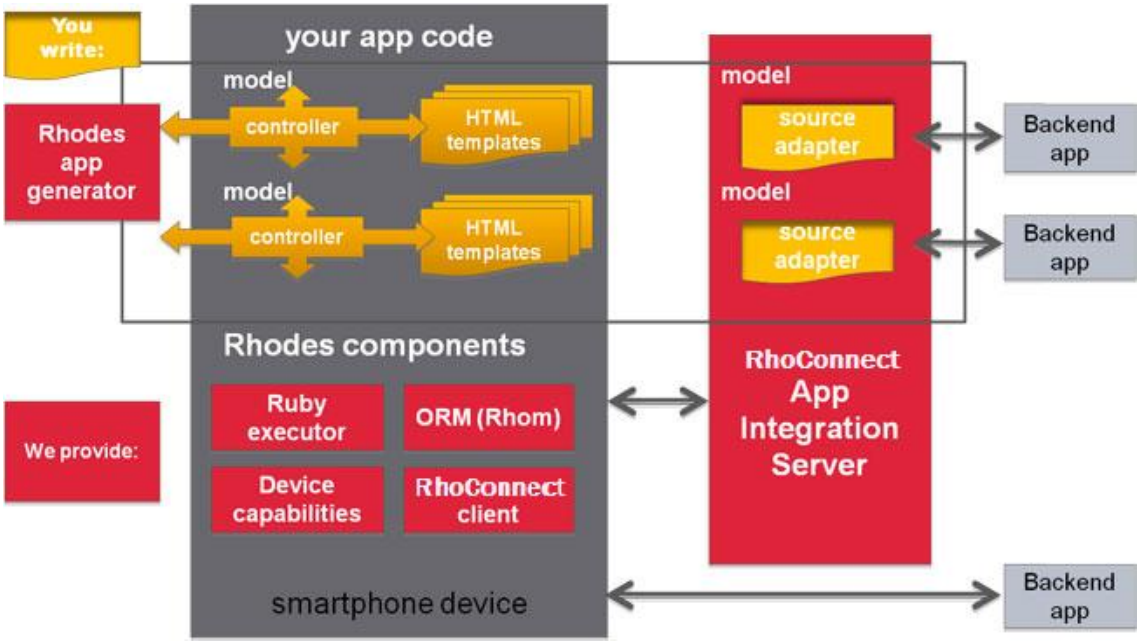

Fig. 4. **Rhomobile Development Environment** (Purdy, 2011.).

RhoConnect is the first framework kind of so called "mobile application integration" server category. Using RhoConnect developers can highly increase quality of compatibility with enterprise deal supporting applications (Rhomobile documentation, 2013). RhoConnect server and RhoConnect built-in client makes all the work associated with data transfer to the device. It is assumed (Rhomobile docucemntation, 2013) that this strategy lowers development efforts for 50 to 80 per cents, by integrating supporting application.

RhoConnect is mainly responsible for going on synchronization (Rhomobile dococumentation, 2013; Christ, 2011; Purdy, 2011): it saves all the data of supporting application into local device, and when supporting application' data is modified it is synchronized with all devices and is available even when device is offline.

RhoMobile Rhodos advantages:

- Partially free (open-source under commerce licence);
- Ability to deploy single codebase on multiple platforms;
- Compiles native application, no web application or hybrid or HTML5 in a mobile browser;
- Interfaces with mobile device's hardware such as camera and GPS by using JavaScript;
- Easy multi-device and cloud synchronizing using RhoConnect;

- Built on Ruby (strength for Ruby programmers);
- Does not require any external SDK installations (Prudy, 2011);
- The only framework with its own security solutions (Prudy, 2011): data encryption, secure authentication;
- Uses MVC model.

RhoMobile Rhodes disadvantages:

- Outputs only native package, not native source code;
- Executes application through RubyVM interpretation package;
- Build on Ruby (weakness for other than Ruby programmers);
- Additional RhoMobile products requires monthly fee (Christ, 2011);

Best suitable for:

- Products, which requires excellent structured project;
- Products, which requires advanced security and data synchronization.

User Interface styling

PhoneGap and RhoMobile very well work with such user interface markup technologies or libraries as Sencha Touch and jQuery Mobile.

Sencha Touch 2 – Upgraded version of Sencha Touch, which was the truly first HTML5 mobile framework created with the aim to fully exploit HTML5, CSS3 and JavaScript abilities to get most flexibility, performance and optimization level (Sencha Touch, 2013).

Sencha Touch (further in text – ST) is developed on JavaScript language, thus JavaScript knowledge is essential to use this framework. ST is the first HTML5 framework created with the aim to expand UI possibilities and compile application for multiple mobile devices' platforms. ST uses MVC (Model – View- Controller) pattern, which grants excellent projects' structuration and overview. Using ST is possible to create fully functional applications for optimized dynamic content and data. ST uses library structure thus is very dependent on global variables. This framework is not the greatest if developers need to provide large amount of themes for their application, because ST really focuses on not only JavaScript code, but also on advanced CSS language – SCSS ("Sassy CSS"), which adds another pack of variables, thus making project very complicated. Mostly, ST is aimed for web kit type browser support, which can be very beneficial with PhoneGap, which uses web kit. The downside on Sencha Touch is that it does not support all major mobile platforms (Mobile frameworks, n.d.), and requires OEM commercial licence purchase.

jQuery Mobile – Based on jQuery and jQuery UI developed mobile framework (physically, just JavaScript library), with the aim to reduce resource consumptions by lowering patch note size and easy-to-build low weight code, progressive user interface enhancement and usability skill wise (jQuery Mobile, 2013).

jQuery Mobile provides great theme creating possibilities by applying on-air build theme using free online ThemeRoller tool. jQuery Mobile supports all major mobile platforms, and essentially, is platform independent JavaScript library (Mobile frameworks, n.d.; jQuery Mobile, 2013). The downside of jQuery Mobile is lower performance compared to ST, and jQuery Mobile is not using MVC pattern, thus can make it harder to structure the project.

Results and discussion

For the analysis purposes following criteria were defined:

- Source code accessibility – Open Source Code, Closed Source Code, Partially Open Source Code;
- Supported platforms – iOS, Android, Blackberry, Windows Mobile, Windows Phone, Symbian, Bada;
- Supported development paths – Native app, Web app, Hybrid app, App generator;
- Supported commonly used web technologies – HTML5, JavaScript, CSS, PHP;
- Use of Model-View-Controller (assuming all technologies support MVC, but only few are based around it);
- Required skill set – HTML, CSS, JavaScript, Ruby, java, etc.;
- Required learning time – low (used only common web technologies), medium (used common web technologies with the help of advanced programming language), high (based around specific programming language, such as Ruby);
- Data security ;
- UI creation and modification – basic (buttons, textboxes, lists), advanced (animation, interactivity, advanced navigation solutions);
- Cloud computing;
- Technical support;
- Technology evolution trend – low activity project (doesn't have recent updates, low ticket response time, bugs are not being repaired), high activity project (have recent updates, fast (1-2 days) ticket

response time, large active community with plug-in ideas and solutions, count of projects, which use this technology), medium activity project;

- Additional features.

Table 1

Comparison of web frameworks

Criteria:		**PhoneGap**	**RhoMobile**
Source Code		Open	Open
Supported platforms	iOS	+	+
	Android	+	+
	Blackberry	+	+
	Windows Mobile	+	+
	Windows Phone	+	+
	Symbian	+	+
	Bada	+	+
Supported development paths		Hybrid	Native
Supported web technologies		HTML, CSS, JS	Ruby, HTML, CSS, JS
Use of MVC		-	+
Required skill set		HTML, CSS, JS	Ruby, HTML, CSS, JS
Required learning time		Medium	High
Data security		Plug-in: Keychain	RhoConnect
UI creation and modification		Basic	Basic
Cloud computing		Compilation	Data Synchronization
Technical Support		5 packages, developer count dependent	Forum, Blog
Technology evolution trend		High activity project	Medium activity project
Additional features		-	Data encryption, secure authentication

As a result of analysing two major web development frameworks and two user interface styling solutions two comparison tables were created (Table 1; Table 2). Even though the same criteria were used for both tables, elements cannot be integrated into one table, because they aim to show different purposes. Frameworks, such as PhoneGap and RhoMobile are the main building technologies, which allow to compile and deploy applications on many mobile platforms, but styling solutions, such as Sencha Touch and jQuery Mobile are used only after main application is already complied, modifying content and layout itself by applying JavaScript based functions and queries.

Web technologies are used, mostly, as a set, not an individual solution, for example, PhoneGap can use full potential of jQuery Mobile or Sencha Touch, and developers will choose ui styling solution only after analysing the main web development technology, set of technologies or framework, which is based, essentially, on set of web technologies. RhoMobile Rhodos framework is aimed more towards high quality, enterprise solutions with advanced security possibilities and better data synchronization, even though many projects, essentially, want to include data synchronization as one of the main criteria, and RhoMobile easily completes this task by implementing RhoConnect framework.

Table 2

Comparison of user interface styling solutions

Criteria:		jQuery Mobile	Sencha Touch
Source Code		Open	Partially Open
Supported platforms	iOS	+	+
	Android	+	+
	Blackberry	+	+
	Windows Mobile	+	-
	Windows Phone	+	-
	Symbian	+	-
	Bada	+	-
Supported development paths		Mobile, Web	Mobile, Web
Supported web technologies		HTML, CSS, JS	JS
Use of MVC		-	+
Required skill set		HTML, CSS, JS, MVC	JS, MVC, SCSS
Required learning time		Low	Medium
Data security		-	-
UI creation and modification		Advanced	Advanced
Cloud computing		CMS	Sencha.io
Technical Support		Forum, Blog, Rich Community	Charged ST support, Free Forum and Documentation
Technology evolution trend		High activity project	High activity project
Additional features		ThemeRoller	Dynamic data support

Table data shows that PhoneGap framework is better suitable with jQuery Mobile, because both of them use platform independent source web technologies, such as HTML5, CSS and JavaScript. Even though, PhoneGap can work with Sencha Touch, and the criteria for choosing it may be programmer skill set or license availability. RhoMobile Rhodos framework is built around Ruby, thus requires addition programming skills. However, Ruby language is not the worst disadvantage of RhoMobile Rhodos – it is licensing, which makes mass market products very expensive. RhoConnect synchronization and data security solutions compensate downside. If RhoMobile Rhodos framework was chosen as the main building technology, then Sencha Touch will suit better with it, because of MVC pattern usage, which will grant perfect project's structure and overall overview.

Conclusion

Analysis of only four web technologies and solution touches m-commerce partly, but main web technology evolution trends are well visible:

- Web technology frameworks are developed and modelled using common web technologies, such as HTML5, CSS and JavaScript;
- Cross-platform development path is the leading trend in nowadays mobile application projects;
- Project cannot be truly realized using only individual web technology and set of technologies must be used in almost any cases.

After accomplished analysis, following proposals can be stated:

- Before making analysis and selection of any existing web technology developers must understand the developing applications objectives in the aspect of usage, audience and skill package;
- Each project requires different mobile device's functionality, which can be achieved using different technologies, and technology selection must be appropriated with developers' skills, technology preferences and available licences.

Analysed web technology development frameworks and user interface styling solutions can be used to develop fully functional mobile devices' application with user friendly interface and handling. Analysed technology usage requires only basic web technology knowledge and experience from developers.

Acknowledgements

Research has been developed with support of European Social Fund activity "Support to Master studies" project "Support to the implementation of LLU masters". Contract No. 2011/0020/1DP/1.1.2.1.1/11/IPIA/VIAA/011.
Research has been developed in the framework of IT centre of competences branch research Nr. 1.19 "Systems model and prototype development of cross-branch mobile applications and smart card web services' management". Research is being developed in the framework of the first project activity, under contract without compensation.

References

Alqahtani, A., Goodwin, R., 2012. E-commerce Smartphone Application. *IJACSA, Vol. 3, No.8m* pp.54-59.

Beckman, M., 2012. Native, Web, or hybrid: How to choose your mobile development path. Available at: http://www.infoworld.com/d/application-development/native-web-or-hybrid-how-choose-your-mobile-development-path-204079?page=0,0, 26.01.2013

Building mobile apps with Sencha Touch and Phonegap, 2011. Available: http://www.lonelycode.com/2011/05/02/building-mobile-apps-with-sencha-touch-and-phonegap/, 26.01.2013.

Christ, A. M, 2011. Bridging the Mobile App Gap. *Sigma Noblis 11(1) (2011)*, pp. 27-32.

Corral, L., Janes, A., Remencius, T., 2012a. Potential advantages and disadvantages of multiplatform development frameworks – A vision on mobile environments. *Procedia Computer Science 10(2012)*, pp. 1202-1207.

Corral, L., Silitti, A., Succi, G., 2012b. Mobile multiplatform development: An experiment for performance analysis. *Procedia Computer Science 10 (2012)*, pp. 736-743.

Dingson, A., 2011. Writing a Hybrid Mobile Application with PhoneGap and the Dojo Toolkit.

Duarte, C., Paulo Afonso, A., 2011. Developing once, deploying everywhere: A case study using JIL. *Procedia Computer Science 5 (2011)*, pp. 641–644.

Firtman, M., 2012. jQuery Mobile: Up and Running, Sebastopol: O'Reilly Media, Inc.

jQuery Mobile Official Documentation. Available: http://jquerymobile.com/demos/1.2.0/docs/about/intro.html, 27.01.2013.

Mobile Frameworks Comparison Chart. Available: http://www.markus-falk.com/mobile-frameworks-comparison-chart/, 26.01.2013.

PhoneGap supported features. Available: http://phonegap.com/about/feature/, 26.01.2013.

Purdy, G. J., 2011, Rhomobile: Automating App Development across Multiple Mobile Platforms. Available: http://www.mobiletrax.com/Newsletters/tabid/115/EntryId/99/Rhomobile-Automating-App-Development-across-Multiple-Mobile-Platforms.aspx, 26.01.2013.

Rhomobile documentation. Available: http://docs.rhomobile.com/home, 26.01.2013.

Sencha Touch Official Documentation. Available: http://docs.sencha.com/touch/2-1/#, 26.01.2013.