Cross-platform development of mobile applications

-Evaluation of the PhoneGap framework

José Calla Guzmán

Victor Cernea

Bachelor thesis

Computer Engineering

CHALMERS UNIVERSITY OF TECHNOLOGY
Department of Computer Science and Engineering
Gothenburg 2012

Abstract

This thesis investigates the benefits and drawbacks of cross-platform mobile app development by developing a smartphone app using the PhoneGap framework. A web based thesis app is created for use on Android, iPhone and Windows Phone smartphones. The implementation of the app is based on a single codebase of the standard web development technologies - HTML, CSS and JavaScript. The application supports Android version 2.2 or later and for iOS version 4.2.1 or later is accepted. On Windows Phone the application is designed to be operational from version 7. The compatibility of the application on any tablets is not tested. The result will reveal problems that may encounter and notable issues that one must take in consideration when using the software development method of cross-platform.

Sammanfattning

Denna rapport utreder för- och nackdelar med plattformsoberoende mobilapputveckling genom att utveckla en smartphone app med PhoneGap-ramverket. En webbaserad konferensapp skapas för användning på Android, iPhone och Windows Phone smartphones. Implementationen av applikationen är byggd på en enda kodbas skriven i de vanliga webbutvecklingsspråken - HTML, CSS och JavaScript. Programmet stöder Android version 2.2 eller senare och för iOS gäller version 4.2.1 eller senare. I Windows Phone är applikationen fungerande från och med version 7. Programmets kompatibilitet på surfplattor är ej testad. Resultatet kommer att åskådliggöra problem som kan inträffa samt svårigheter som man måste ta i beaktande när man använder sig utav plattformsoberoende mjukvaruutveckling.

Keywords: cross-platform, smartphone, app, PhoneGap, web based

Contents

Abbreviations and concepts

3G Third generation mobile telecommunications standards

4G Fourth generation mobile telecommunications standards

AMOLED Active Matrix Organic Light Emitting Diode

API Application Programming Interface

app Mobile application

CPU Central Processing Unit

CSS Cascading Style Sheets (CSS) is a programming language used for describing the look and formatting used in a document written in a markup language. CSS is designed to separate the content of a document (usually written in HTML) from the presentation of the text. The CSS code is usually written in a separate file and defines the layout, colors and fonts of the document.

cross-platform, cross-platforming
Method of software development for multiple hardware platforms.

div Division in HTML

DOM The Document Object Model is a cross-platform and language-independent convention of representing and interacting with HTML, XHTML and XML documents.

emulator In computing, an emulator is hardware or software or both that duplicates (or emulates) the functions of a first computer system (the guest) in a different second computer system (the host), so that the emulated behavior closely resembles the behavior of the real system. Historically, the term emulator was used more for describing mostly hardware emulation and the term simulator was used for describing software simulation, but now the two terms are often interchanged. [64]

feature phone
A feature phone is a mobile phone with additional functions above standard mobile services. These additional functions could be e.g. camera and GPS navigation. Feature phones are at manufacturing considered low-end devices in contrast to smartphones that are considered high-end devices.

GPU Graphics Processing Unit

GUI Graphical User Interface

HTML HyperText Markup Language (HTML) is the most used markup language for web pages throughout the Internet. Web browsers are used to display the content of the pages written in HTML. The HTML code consists of tags enclosed in angle brackets (like <body>). These tags are not shown when displaying the web page, but are used by the web browsers to interpret the content of the page.

ID Identification, Identifier

IDE Integrated Development Environment

JS JavaScript

LCD Liquid Crystal Display

meta-tag Provides metadata about a HTML document. For browsers, it indicates how a page is reloaded.

multitouch Using gestures to communicate with the OS by touching the screen. Elements a user can control could be sliders, switches and buttons. Some common features are pinch, reverse pinch, tap and swipe.

native language, native mobile app

A native mobile app is a concept that defines a smartphone application coded in a specific native programming language. For an iPhone the native language is Objective-C and for Android the native language is Java. [68]

OEM Original Equipment Manufacturer

OS Operating System

ppi points per inch

SDK Software Development Kit

simulator see emulator

smartphone A smartphone is a mobile phone that is built on a mobile computing platform, with more powerful computing ability and connectivity than a feature phone. The differences between smartphones and feature phones are vague and some of the capabilities are overlapping. Some of the features that are normally coupled with smartphones are large screens (2 inches to over 4 inches), high resolution screens, user interface control with multi-touch gestures, QWERTY-keyboard (either physical or on-screen), portable media player, compact digital camera, pocket video camera, and GPS navigation unit. One of the most significant differences between smartphones and feature phones, is that the advanced APIs on smartphones allows third party applications to have better integration with the OS and hardware than what is typical with feature phones.

thesis app The thesis app for this thesis is a conference app assigned by the consultant company Squeed AB and developed by the authors. This app is supposed to give information about the dev:mobile conference, on the 12th of June 2012 in Gothenburg, Sweden.

TFT Thin Film Transistor

web-based application
An application in which all or some parts are downloaded from the Web each time it is run. The application is written in the standard web technologies, such as HTML, CSS and JavaScript.

WP Windows Phone

XNA XNA’s Not Acronymed (first known as XBOX New Architecture)

XHTML eXtensible HyperText Markup Language

XML eXtensible Markup Language

List of tables and figures

1 Introduction

Applications for smartphones, mobile apps, are in a stage where they have to exist on multiple platforms in order to reach out to as many users as possible. Developing in native languages for multiple platforms requires more resources for software development companies which will also result in higher expenses. Through cross-platforming, developers can create applications for multiple platforms using the same code base. Cross-platform development uses common programming languages that smartphones implement - HTML, CSS and Javascript. [73]

1.1 Problem

The authors intend to develop an application for smartphones by using the cross-platform method with the PhoneGap framework and from there investigate what differences there are between each platform and compare benefits and drawbacks of using cross-platforming.

1.2 Purpose

The purpose of this thesis is to investigate the feasibility of cross-platform software development for mobile devices by implementing an app with the PhoneGap framework. The app was requested by the consultant company Squeed AB, that hosts the conference dev:mobile for mobile developers. For this conference there was a need for a mobile application for Android, iPhone and Windows Phone smartphones. By accepting the assignment to develop the thesis app, the authors aimed to become more competent in cross-platform mobile app development and app market distribution.

1.3 Thesis disposition

This thesis is divided into certain sections where different topics are presented and discussed. The thesis starts with an introduction of the problem and purpose of the entire study as mentioned earlier. In section two, the method of how the authors intend to solve the thesis problem is introduced. The third section covers the background of the thesis by starting with a brief description of the smartphone market, and then continuing with an explanation of the technical environment of the project, by addressing the different areas vital to the subject. In the fourth section, are the application implementation concepts and design models outlined. The section also features a description of the testing process of the app. After the theoretical sections the authors continue with the achieved practical results, section five, where the author’s choice of GUI design is depicted. In section six the authors elaborate, with help of the theoretical and practical work within the study, the conclusions and limitations of developing mobile apps in the cross-platform PhoneGap framework. In this last section the authors also suggest what future improvements and additions can be made to the app, but also what other apps could be implemented based on the constructed app.

2 Method

A good way to investigate the feasibility of developing an app with PhoneGap, is to actually develop an app in the framework for a real-life assignment. This will answer the question if the PhoneGap generated app is of such good quality that it can be used in a commercial environment. The development of the thesis app will start with research of the cross-platform framework PhoneGap, and then continuing with the design of the different components of the app, for example the navigation in the app and of the GUI. The design of the app is decided by the authors. The coding will mostly be done in web development languages, such as HTML5, CSS, XML and JavaScript. For Android, the work will be done on the development tool Eclipse with Android Development Tools. For the development of the iOS app, the work will be done on a Mac OS X computer together with Xcode. The Windows Phone development will be done with Visual Studio and with Windows Phone Toolkit. All three platforms have specific system development kits that will be used for each platform. Testing of the app will be done on IDE simulators for each platform and also on physical smartphones.

2.1 Scope

The scope for this thesis is to a large extent dependent of the initial app requirements introduced by the consultant company Squeed AB. These requirements decide which platforms are to be targeted (Android, iOS and Windows Phone), which cross-platform framework is to be used (PhoneGap), and which functionality is to be expected from the app. Additionally, the time limit for the development of the thesis app is set to a month before the conference, due to time needed for the distribution on the respective app markets.

3 Background

The background of the thesis is covered in this section by firstly describing the mobile market focusing on the thesis platforms, and secondly by presenting the technical environment for the thesis.

3.1 Smartphone Market

The three mobile platforms of interest in this thesis are as mentioned before Android, iOS and Windows Phone. However, there are other important platforms on the market that should be mentioned when discussing the smartphone market as a whole, and they are primarily Symbian (by Nokia) and Blackberry (by RIM), as seen in Figure 3.1. These two platforms have historically had large market shares, but were dropping in market shares lately due to strong sales for iOS and Android [20]. It could also be mentioned, that the fifth most popular platform for smartphones is at present Bada manufactured by Samsung.

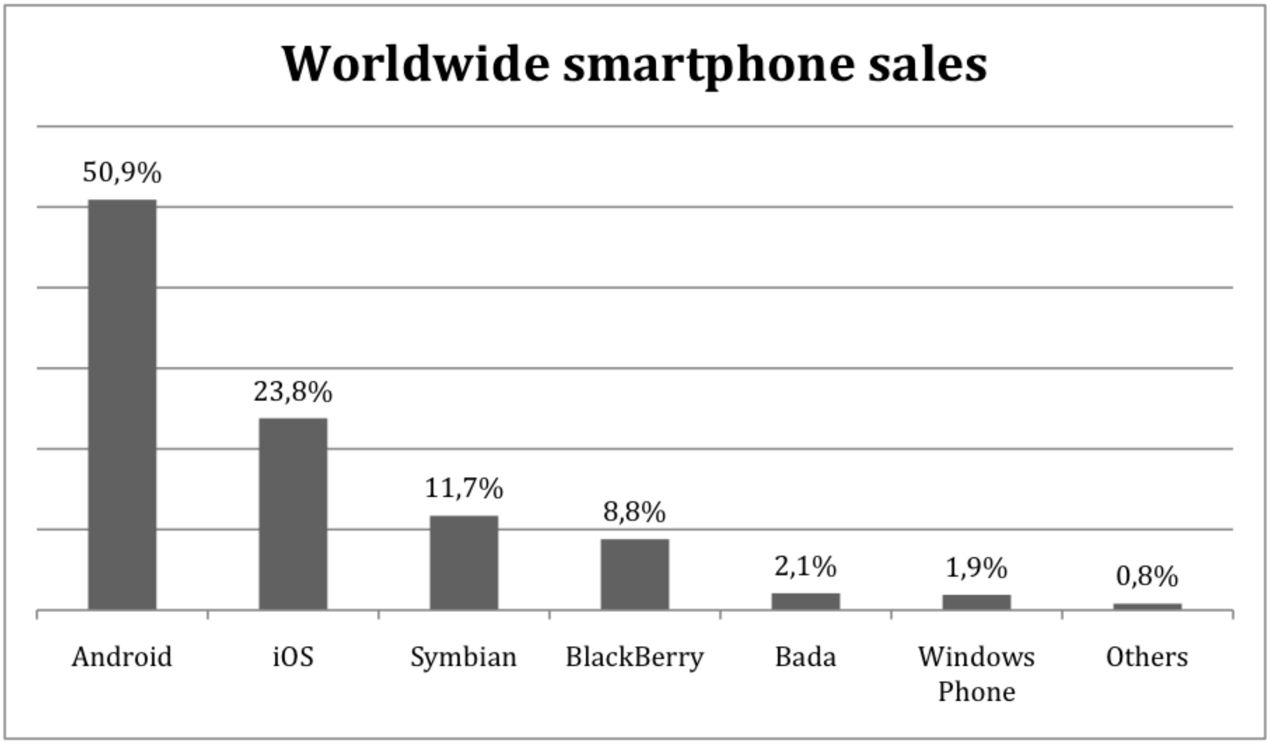


Figure3.1 Worldwide smartphone sales grouped by OS, fourth quarter 2011. [39]

3.1.1 Android

Google purchased Android Inc in 2005 and since then developers have been introduced to a numerous of new versions [9]. The open source platform has steadily increased since 2008 and is now the largest smartphone platform worldwide [20]. The Linux based platform has become popular among developers because of the simplicity in allowing third party application to be downloaded and installed from a large variety of app markets [45]. The Android platform is backwards compatible with older versions, which allows developers to create applications for smartphones not running the latest versions and thus so reaching out to more users. Android version 2.2, Froyo, is one of the older versions and 23.1% of the Android users are running it as their smartphone OS. The mostly used Android version is 2.3, Gingerbread, with 63.7% of the Android users [44]. In March 2012, Google launched their new app market, expanding it with music and video downloads, by merging the two predecessors Android Market and Google Music services [34].

3.1.2 iOS

iOS is the mobile operating system developed for Apple’s mobile and tablet devices. The OS was originally released 2007 under the name iPhone OS, but was renamed 2010 to reflect the extended support for other Apple devices such as IPad and Apple TV [35]. Some of the competitors to iOS on the mobile OS scene are Google’s Android OS, Windows Phone 7 and BlackBerry OS. In comparison, the iOS is leading in the area of total app downloads, with over 25 billion downloads [11], compared to its closest contender Android with over 10 billion downloads [7]. Also when it comes to global mobile application stores revenue, the iOS App Store has the highest revenue [40], but that may be due to it only having 28% of free apps, in comparison to Google Play's over 57% of free apps [37]. iOS’s closed and proprietary nature has been criticized from developers to competitors [42] [15]. Some of the issues that have been in focus are the development model (requiring a yearly subscription costing US$99) and the thoroughly review process for distributing apps on App store. However there are other developers that argue that a more closed platform could provide added security [17].

3.1.3 Windows Phone

Microsoft OS for smartphones, formerly known as Windows Mobile, is the sixth largest operating system, after Android, iOS, Symbian, Blackberry and Bada as seen in Figure 3.1. When distributing an app for WP, every application must be approved by Microsoft and has to be available from the Windows Phone Marketplace [31]. The merely two-year-old WP platform has not yet shown impressive sales figures, and it is hard to predict the future for Windows Phone. With Nokia as manufacturer and Windows Phone “Tango” as new OS, some predict the WP devices can target the emerging markets as a Symbian replacement. [34]

3.2 Technical environment

The important technical areas of the thesis are covered in this section. Firstly by describing the different platforms that are at interest for this thesis, and secondly by presenting the framework used in the app development, and thirdly by introducing the development tools used during the implementation process. In the end of this section a comparison of the different platforms is presented.

3.2.1 Android

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 [45]. The programming language in Android is mostly Java and XML. 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) [44]. Smartphones running Android implement full multitouch since version 2.0. [38]

3.2.2 iOS

An UNIX based operating system developed by Apple for iPhone, iPad and iPod touch. iPhone apps, like iOS, are written in Objective-C, an object-oriented programming language based on C programming language. As mentioned in section 2.1.2 developers must have a license in order to develop iOS applications. Multitouch is standard and implemented on all iPhones. iOS uses Mobile Safari, developed by Apple, as the default web browser. [74]

3.2.3 Windows Phone

The launching of Windows Phone introduced the users to a new user interface that completely differed from Windows Mobile, and was based on a design language named Metro [18]. The intentions with the Metro design are to enable the user for fast interaction with the installed applications on the phone [26]. Windows Phones uses Internet Explorer Mobile as web browser combined with Internet Explorer 9 [31]. Third party applications for WP must be based on XNA, Silverlight and .NET. Windows Phone SDK comes with Visual Studio 2010 Express in order to allow developers getting started more quickly. [60]

3.2.4 PhoneGap

PhoneGap is an open source mobile development framework originally produced by Nitobi, but later purchased by Adobe Systems [38]. It enables mobile app developers to build apps for mobile devices using HTML5, CSS3 and JavaScript. 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). [66]

3.2.5 Development tools

For developing the cross-platform app described in this thesis, 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. The iPhone app was developed using iOS SDK together with the Xcode IDE. The Windows Phone app was implemented using the Visual Studio 2010 Express IDE that contained the Windows Phone SDK. The actual programming was done utilizing different web development technologies.

3.2.5.1 Eclipse

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 favourite with Java developers [52]. 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. [51]

3.2.5.2 Xcode

Xcode is Apple’s IDE for developing software for OS X and iOS with a built in compiler and debugger that supports Objective-C, Java, C++ and many more languages. Xcode together with iOS SDK is used when developing for iPhone [47]. Worth noting is that, like OS X, it requires an Intel based processor and at least Mac OS Leopard 10.5, in order to run version 3.1 of Xcode. Downloading the software requires a registered Apple ID. With the iOS SDK you can upload your own developed application or game directly to your phone if you have a developer’s license and a development registered iPhone [48]. There is also a simulator available for testing in Xcode, which does not require a license. [54]

3.2.5.3 Visual Studio 2010

Visual Studio 2010 is an IDE for developing software for Windows based systems. The IDE has built-in language support for C/C++, C# and .NET among others [59]. Microsoft delivers the Windows Phone SDK with Visual Studio 2010 Express with a 30 day trial which facilitates developers when starting to work with WP [58]. Like the other two mentioned IDE’s, the WP SDK also comes with an emulator, so the need for a Windows phone is not required in order to test run the software. Visual Studio has a built in marketplace test kit allowing developers to perform a series of tests before an eventual upload to the marketplace for Windows Phone. [63]

3.2.5.4 HTML, CSS and JavaScript

The HyperText Markup Language (HTML) 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 [10]. The markup language uses CSS to give a web page a more suitable layout than just plain text. CSS adds colour, font, text size and disposition to a web page [69]. JavaScript can be included to a HTML document to give a page some interactivity, e.g. changing the colour of a button when the mouse pointer is hovering over it. [70]

3.2.5.5 jQuery

The most used JS library [50], 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 [71]. jQuery is an open source software licensed under MIT and GNU General Public License.[56]

3.2.5.6 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. The user interface that jQuery offers support 86% of the web browsers in smartphones [4]. When using jQuery Mobile to create applications, the app is constructed by div's each assigned a different *data-role*. The syntax is written inside the div. [57]

3.2.6 Comparison of platforms

It is a complex task to compare the Android, iOS and Windows Phone platforms, because of the wide variety of smartphones available, specially within the Android platform. The general differences and similarities between the latest smartphones (as of early 2012) will be discussed in this section.

3.2.6.1 Hardware

When comparing the different smartphones hardware, it is easier to point out what the phones lack compared to the others. For instances, most Android and Windows Phones displays are larger than the iPhone screen. The iPhone 4/4S's standard is the 3.5 inch screen, but the Android and Windows Phone standards seems to be moving towards 4.5 inch and above [33]. When the iPhone 4 was released in 2010, it had the best screen on any smartphone, with a 960x640 pixel-resolution and a pixel density of 326 ppi [14]. But in 2012, the competition has produced smartphone displays with higher resolutions and equally high pixel densities, like e.g. the LG Nitro HD smartphone with a 1280x720 pixel-resolution and a pixel density of 326 ppi [25].

Another important hardware comparison is the processor. The iPhone 4S has the 800Mhz Apple A5 processor with dual core CPU and dual core GPU. When it comes to Android, there is large variety of processors from different manufacturers [72], but many of the latest Android and Windows Phone smartphones are running Qualcomm Snapdragon processors [67]. The latest Snapdragon is the 1.2 - 1.4GHz S3 processor with dual core CPU and single core GPU. In the comparison between the A5 processor and the S3 processor, the result is that they have comparable features and they have similar CPU architecture but the A5 has better GPU mainly due to its dual core. [32]

There are some other extra hardware features on some Android and Windows Phone smartphones that don’t exists on iPhone 4S, such as micro HDMI ports [1], FM Radio and microSD cards.

Finally, the battery for the iPhone is technically rated lower, but does good in comparison with other platforms [23]. More important for the battery lifetime than platform, seems to be if the phone has single core processor or dual core processor, the brightness level of the screen and if the phone’s 3G or 4G services are turned on.[3]

3.2.6.2 Software

When it comes to mobile platforms there are two general types of software, platform OSes and platform dependent apps. This software comparison only covers the OSes and their features. To simplify the OS comparison and make it more fair, the discussion is only about OSes installed on stock versions of smartphones right of the box. This because, when it comes to Android, OEMs tend to skin the phones and often adding functionality to the stock version of Android. Below, in table 3.1, a selection of features of the three mentioned OSes are listed. Furthermore, there are plenty of third-party apps that may fill the gaps found in the base OSes. In regards to most features in this comparison, Android and iOS ties for the first place. The Windows Phone platform has not yet reached the same maturity level as the other two platforms [19]. For Apple iOS 5 compatibility, see Appendix 1.

Table 3.1 Platform characteristics of cross-platform development. [12], [14], [24], [53]

	Android 4.0	iOS 5.1	Windows Phone 7.5
Native language	Java	Objective-C	C#
Web browser	Webkit	Mobile Safari	Internet Explorer Mobile
Widget Support	Yes, with the option to resize widget	Yes, Stock and Weather widget	Live Tiles
Multi-tasking	Yes	Yes	Yes (limited)[61]
Closing apps within task manager	Yes	Yes	No
Unified Notifications	Pull down Notification Center	Pull down Notification Center	Toasts and Live Tiles
Folders	Yes	Yes	No
IDE	Eclipse	Xcode	Visual Studio
Storage Security	Sandbox	App Sandbox	Isolated Storage
Flash support	Yes	No	Yes
Developing licens	Open Source	Required	Required
Registered hardware	No	Required	Required
Tethering	Yes	Yes	Yes (subjected to device compatibility)
Supported development OSes	Windows, Mac OS X, Linux(i386)	Mac OS X	Windows
Integrated Voice Assistant	GoogleTalk	Siri	TellMe
Data Monitor and Control	Yes	No	No
Multitouch	Yes	Yes	Yes
Markets	Google Play	App Store	Windows Phone Marketplace
Apps	400.000+[28]	550.000+[30]	80.000+[2]

3.2.6.3 Similarities

Even though all platform OSes bring something unique to the market, there are a lot of features that are similar between platforms. The competition between the different manufacturers is furious and in some cases they flat out copy from each other. One such example is the unified notification center implemented in iOS 5 that, in some developers mind, in large extent is copied from Android. [29]

Another similarity, which is found on every platform, is their support for standards-based Web technologies like HTML, CSS and JavaScript. This is crucial for cross-platform development and some developers mean that the future may lie in these technologies. [6]

3.2.6.2 Differences

Android, iOS and Windows Phone share many features but looking at some of the differences worth of interest is how they store data locally on the device and licenses issues. When it comes to WP the data is stored using Isolated Storage compared to local storage for Android and iOS. For Android there is no need for licensing the development devices but for iOS and WP it is a requirement. Another difference is the native web browsers of the different platforms as seen in table 3.1.

4 Implementation

This chapter covers how the application is constructed from the different programming modules to the user interface. The application design section describes how we utilize cross-platforming by using the PhoneGap framework. In the user interface section the authors portray a complete schema of the different pages a user can navigate through. Testing is the last part in this section that describes how the application was tested.

4.1 Application Design

PhoneGap is used to make the transition from the native mobile languages to web-based programming languages in all three OSes. The backend code of every platform is written in native language. As seen in figure 4.1, the PhoneGap framework bridges the gap between the native language and HTML by working as a wrapper for the app and generating Javascripts 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. When the user has navigated to the *Agenda* page, figure 4.2, there is a possibility to save the desired subjects locally on the phone. In order to save these subjects locally on the phone, the authors implemented a script handling these requests. This locally saved agenda will be shown in the *My Agenda* page.

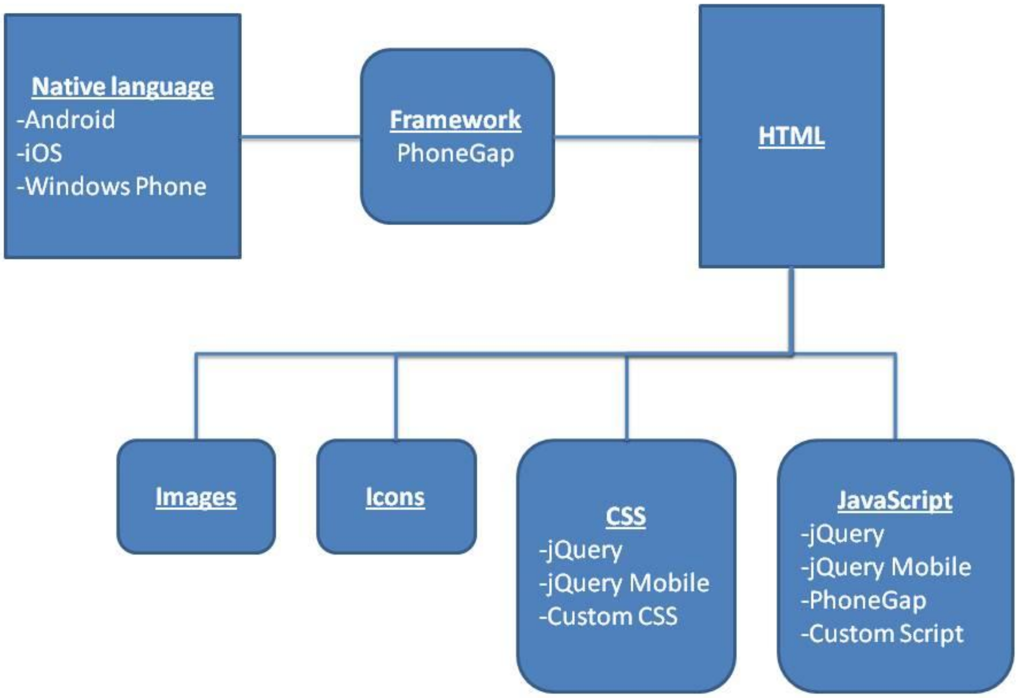


Figure 4.1 Structure of a PhoneGap produced app

4.1.1 Application Scope

The application shall run on Android version 2.2 or later and for iOS version 4.2.1 or later. On Windows Phone the application shall be operational from version 7. The authors will not investigate the compatibility on any tablets of the platforms mentioned above.

4.2 User Interface Design

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 4.2. The authors 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.

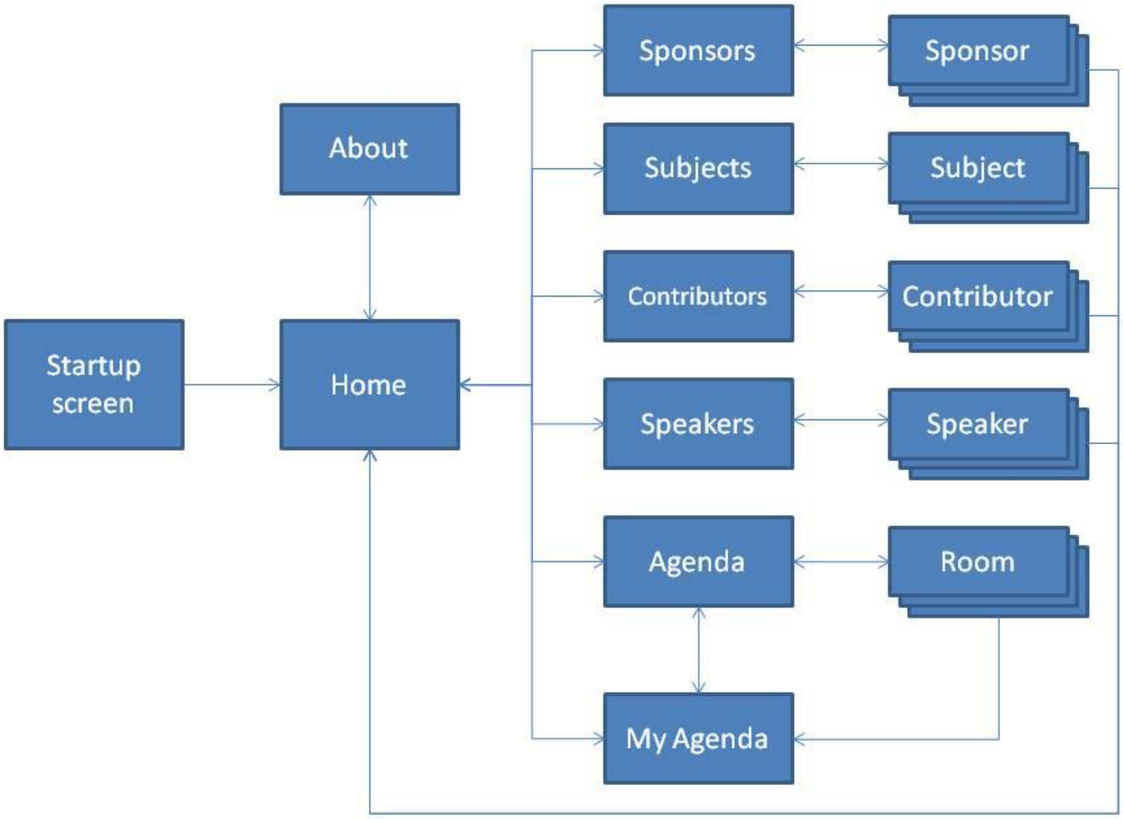


Figure 4.2 Overview of the navigation through the thesis app

4.3 Testing

During the process of developing an app, the functionality and operation of the app is tested in two ways. Either by using the simulators integrated into the IDEs, or by using real mobile devices, but usually by combining the two methods. The advantages of simulators are that they are cheap (in most cases completely free) and they can sometimes be used to test different hard to reproduce scenarios (no network, location, etc.). Another benefit of simulators is that it is easy to get started with the testing process, by just downloading the simulator and start using it. The main drawback of simulators is the fact that the testing is not performed on the real platform, and it is not sure that the app will actually work on the mobile devices it is intended for. There are also some performance differences between the simulators (using the hardware of the computer) and the physical mobile devices. The app most likely performs differently in a simulator, on a computer with more CPU power and more memory, than on a smartphone. The advantage of testing on real mobile devices is of course that the testing is done on the devices the app is supposed to run on. If the app works correctly on the mobile devices the app is fully functional. The drawback is that it is more expensive and time consuming to acquire and manage the different physical devices, and especially if the app is developed for multiple platforms and hardware specifications. [65]

4.3.1 Simulators

The following section lists the simulators used during the development and testing of the thesis app. In figures 4.3, 4.4, 4.5, the home screen of the thesis app is shown, while running and testing the app in the different simulators.

4.3.1.1 Android Virtual Device (AVD)

The AVD Manager and emulator are tools for developing Android apps, and they are included in the Android SDK. This emulator runs on all OSes supported by Android SDK, such as Windows and Mac. Figure 4.3 shows the emulator displayed on the computer screen. The emulator program displays to the left the simulated mobile device screen, and to the right simulated keyboard and hardware buttons that exists on some Android devices. [46]

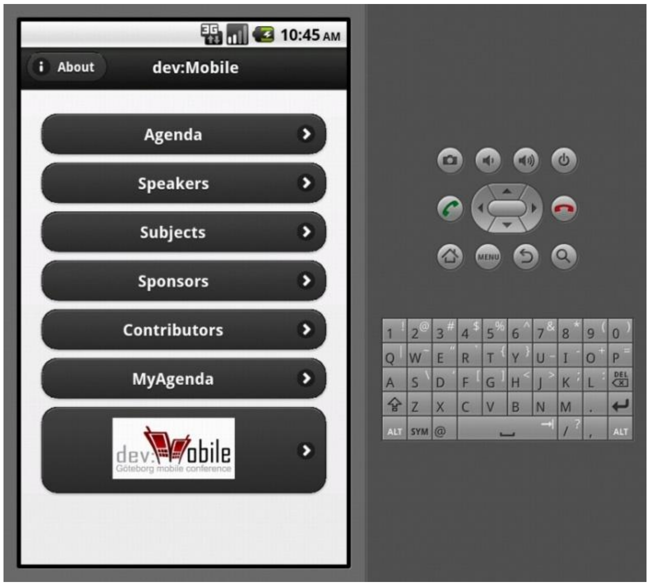


Figure 4.3 Android Virtual Device (AVD)

4.3.1.2 iOS Simulator

For developing and testing iOS apps on a Mac computer, the iOS simulator is used. This simulator can be used for simulating three types of devices; iPhone, iPhone with retina displays and iPad. It also supports different versions of iOS. The simulator can mimic gestures and hardware manipulations, i.e. rotate the device and shake the device. Figure 4.4 depicts the simulator program running on a Mac computer. The simulator program has the look of a real iPhone 4S device. [55]

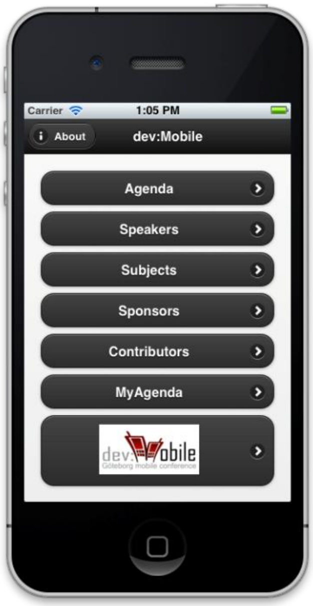


Figure 4.4 iOS Simulator

4.3.1.3 Windows Phone Emulator

Windows Phone Emulator is an application that emulates Windows Phone 7 devices. It provides a virtualized environment for developing and testing apps for Windows Phone. The emulator is installed with the Windows Phone SDK, and supported OSes are Windows 7 and Windows Vista or later. Figure 4.5 displays the emulator program running on a Windows computer. The emulator program mimics the look of a generic Windows Phone device. [62]

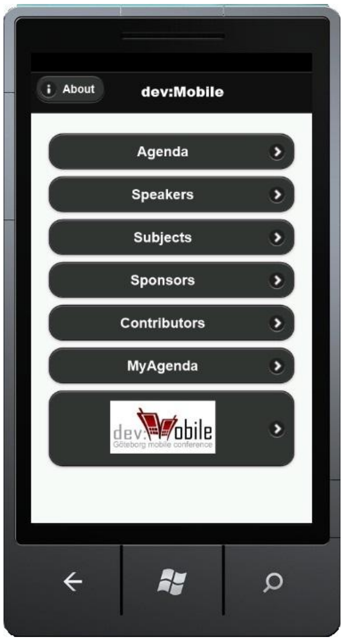


Figure 4.5 Windows Phone Emulator

4.3.2 Mobile devices

This section lists the physical devices used during the development of the thesis app. Different Android devices were tested during the development, to cover some of the different Android screen sizes available. The app was also tested on two different iPhones to test it on both retina displays and non-retina displays. Finally the app was tested on a physical Windows Phone smartphone. In Table 4.1, the devices that were used during the testing of the app are listed with some general hardware information.

Table 4.1 Tested Mobile Devices [72]

	Samsung Galaxy S II	HTC Legend	LG P500 Optimus One	Apple iPhone 4S	Apple iPhone 3GS	Samsung GT-i8700 Omnia 7
OS	Android 4.0.3	Android 2.2	Android 2.3.3	iOS 5.1.1	iOS 5.1.1	Windows Phone 7.5
CPU	1.2 GHz dual-core ARM Cortex-A9 Samsung Exynos	600 MHz Qualcomm MSM7227	600 MHz Qualcomm MSM7227	800 Mhz dual-core Apple A5	600 Mhz Samsung S5PC100 ARM Cortex-A8	1 GHz Qualcomm QSD8250 Scorpion (Snapdragon)
GPU	ARM Mali-400MP or GeForce ULP	Adreno 200 (AMD z430)	Adreno 200 (AMD z430)	Imagination Technologies PowerVR SGX543 dual-core[49]	PowerVR SGX535	Adreno 200
Memory	1 GB	512 MB	150 MB	512 MB	256 MB	512 MB
Display Size	4.27 inches (108 mm)	3.2 inches (81 mm)	3.2 inches (81 mm)	3.5 inches (89 mm)	3.565 inches (91 mm)	4.0 inches (102 mm)
Display Technology	Super AMOLED Plus	AMOLED	TFT LCD	Retina LED backlit IPS TFT LCD	24-bit LCD	Super AMOLED
Display Resolution	800x480	480x320	480x320	960x640	480x320	800x480
Display Pixel Density	218 ppi	180 ppi	180 ppi	326 ppi	163 ppi	233 ppi

The testing of the application on respective smartphone was conducted by uploading the software through the IDE’s of the different platforms. For the two iPhones, the IDE Xcode was used to upload the application onto the phone, and from there test that the interface and functionality was up to standard. Eclipse was used for the Android phones and Visual Studio for the Windows Phone. One notable difference when uploading the software through the IDE´s was that Xcode and Visual Studio required that the devices were registered as developer phones, unlike Eclipse where it was merely plug and play.

5 Result

This section depicts the design choices made for the thesis app implemented in this work. The overall design is constructed using jQuery Mobile components, such as pages, themes, dialogs, buttons, etc. The home screen is also the main menu and contains buttons with choices to navigate to different pages. In these pages, the user can read about subjects and sponsors and other conference information. In addition, the agenda page shows which subjects that are scheduled and from here the user can add subjects to his/her own agenda in the *My Agenda* page. The three platforms does not show much difference when displaying the different pages, except for differences in showing resolution depending on what physical device the app is running on and what display size the device has. In figure 5.1, the home screen is displayed on different testing devices to demonstrate these displaying differences. The figure shows the devices with the biggest and smallest screen sizes during testing, and there is some unused space under the buttons on the big screen compared to the smaller.

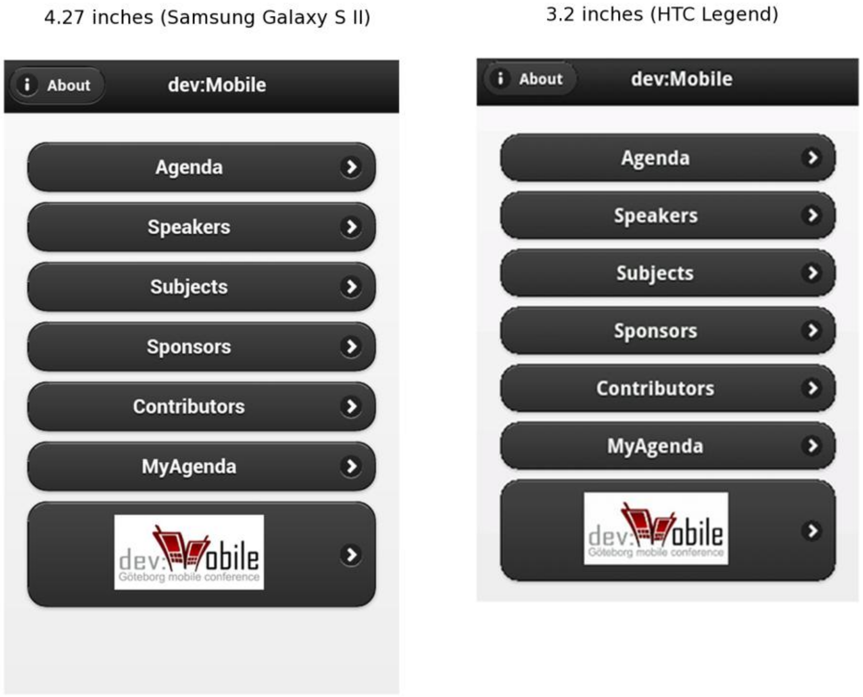
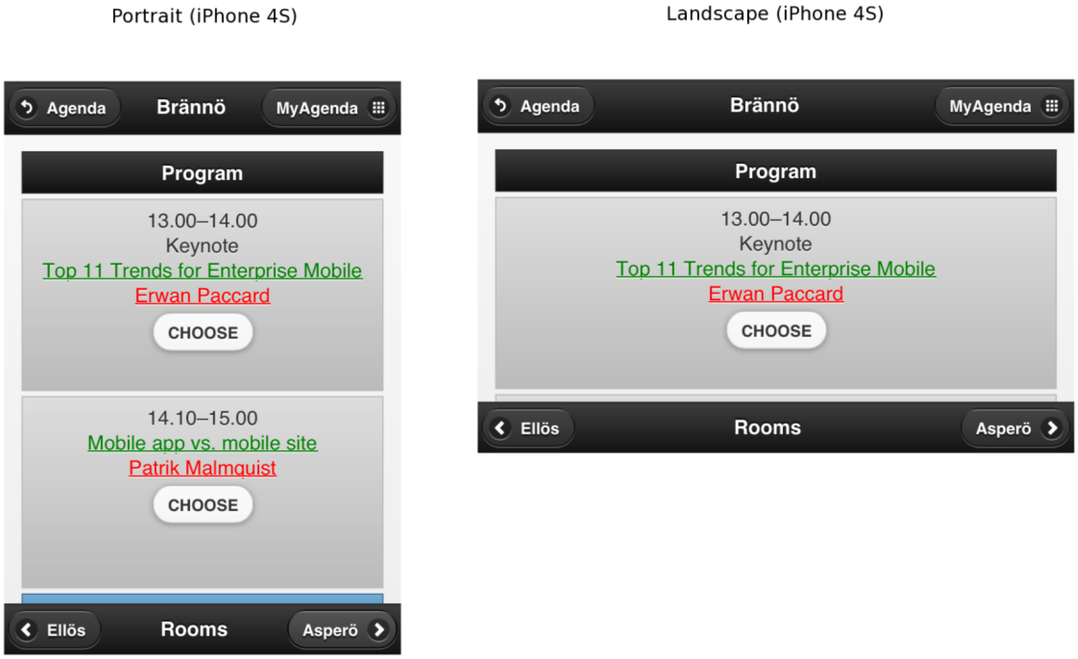


Figure 5.1 Display size comparisons

In figure 5.2, it is seen that the thesis app supports both portrait orientation and landscape orientation. The app rescales the width and height of the viewed page (including header and footer) and displays it correctly. When implementing the orientation support, the authors encountered some issues that will be discussed in section 6, Discussion.



Portrait (iPhone 4S)

13.00–14.00
Keynote
Top 11 Trends for Enterprise Mobile
Erwan Paccard

14.10–15.00
Mobile app vs. mobile site
Patrik Malmquist

Landscape (iPhone 4S)

13.00–14.00
Keynote
Top 11 Trends for Enterprise Mobile
Erwan Paccard

Figure 5.2 Orientation support

The Windows Phone showed some limitations both graphical and in development usability. The graphical limitations where that the color themes where not displaying as intended and that the page transition animations where not supported. This can be seen in figure 5.3 where the buttons and headers/footers coloring differ with no color gradients on the Windows Phone device. Another graphical limitation of the WP platform was that fixed footers where not supported. Fixed footers are when the footers are always shown on the bottom of the screen and on the WP device it is only possible to have the footer showing at the bottom of the page, as seen in the figure 5.3. The development usability limitation is that the WP OS has no built-in support for taking screenshots on the devices (contrary to Android and iOS), so the WP capture in the figure was taken within the emulator.

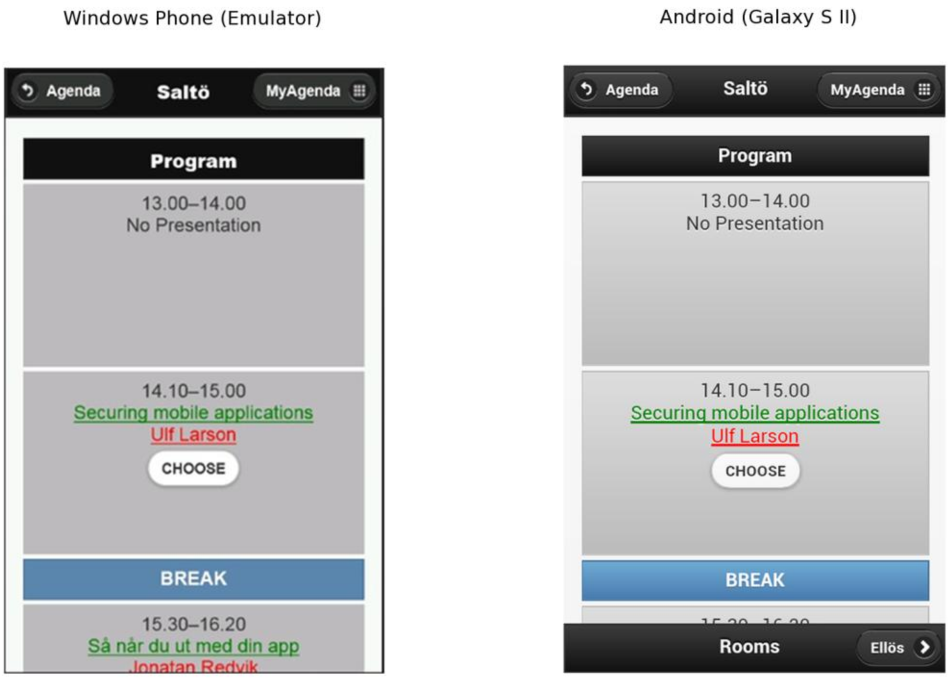


Figure 5.3 Limitations of Windows Phone

During the development of the thesis app some problems arose that had to be dealt with in order to make the functionality work on all the platforms. These problems are discussed in section 6.1. In some cases the authors were content to make GUI compromises to make the app work acceptably on all the platforms. Some of the compromises are seen here in this section, such as in figure 5.1, with the unused space in the bottom of the Home screen page. This was purely a design choice to make the app functional even on smaller device screens. Also, as mentioned before, the colorings of the Windows Phone components rendered by the jQuery Mobile script are uniform and not having the same rich-looking style as on Android and iOS, as seen in figure 5.3. This was considered merely an appearance detail and was not affecting the functionality of the app.

6 Discussion

In this section, the conclusion the authors reached when developing a conference app using cross-platforming is presented. Some of the main problems are brought up and a discussion of the app requirements and if they were completed is presented. A discussion about the benefits and drawbacks of working in the PhoneGap framework is also included.

6.1 Problems

Some of the problems of developing with cross-platforming is of course to make the codebase functioning on all the platforms. In this particular project, the authors chose an approach to make the app code base as general as possible to work on all platforms. Some graphical issues are mentioned in section 5, Result, which did not interfere with the functionalities of the application. A different graphical issue that arose was how WP handled the landscape view. When switching to landscape view on Android and iOS the resolution was kept and the content was stretched out to fill the wider screen, while on WP the content was enlarged. The solution was to add a new meta-tag, different width and height scale, that was only launched if the application was run on a Windows Phone.

Another issue with the landscape view was on iOS when using the fixed header and footer. When switching to landscape view, the content of the page was not stretched out. The header and footer were displayed as demanded but the content remained at the same width as in portrait mode. The CSS file of jQuery mobile was the issue and the solution was to modify how the header and footer were displayed on an iPhone by creating a new CSS file specifically for iOS.

Another problem that was discovered during development was the way local storage is implemented on the different platforms. The thesis app could not save data locally as intended on a Windows Phone device. WP is using isolated storage for security measures, see table 3.1, which was encountered when data had to be stored locally from the application. To solve this problem a separate DOM JavaScript was used for Windows Phone to handle the writing process for the isolated storage.

6.2 Conclusion

The thesis app developed in PhoneGap was finished and all the primary requirements were completed, except for releasing the app on the different markets. As of this writing, the thesis app is uploaded and awaiting approval at the different markets.

The app was functioning satisfactorily on all the test devices, which in addition were running different OS versions. The secondary requirements were not implemented due to lack of time and left for future development of the app.

The total time spent on this project was around ten weeks. The first week was spent on planning the project and the second week the authors studied and learned about PhoneGap. Week three through four was spent developing the app prototype and testing it on simulators and physical devices. The work during this phase went better than anticipated and the authors could start write this thesis report earlier than originally planned. The following three weeks were spent writing the report, holding on the programming work of further enhancement of the app. The last three weeks were used to finalize the app and starting to commit the app on the different markets but also to complete writing the thesis.

The conclusion of the authors was that they learned about developing an app in PhoneGap. Even though the mobile programming background for the authors was only native programming in Java for the Android platform, they found it quite easy to get started with this cross-platform framework and quickly produce a prototype of the thesis app. When learning about PhoneGap, the authors consulted *PhoneGap Beginner's Guide* [21] which had constructive hands-on examples to try out. The authors also became more competent in the app distribution process for the different platforms even though the app is not yet available at the different markets.

The benefit of using PhoneGap for cross-platform development of mobile apps is, as mentioned before, the possibility of developing apps for multiple platforms by using standard web development technologies. Other benefits of using PhoneGap are that it is fast to develop, test and deploy the app to the chosen platforms and that the resulting apps are allowed to be distributed on the different app markets.

A drawback of using PhoneGap and other cross-platform frameworks, is that not every feature of the platforms is possible to access in the framework, just because of the fact that all platforms have different features. When cross-platforming, the developer should always try to make general solutions for the code base to work on all platforms and not include platform dependent specialties. The result might, in comparison to a native app, be an app with a generic feel and look. The performances of native apps are for the most part faster and smoother than mobile web apps, such as the apps developed in PhoneGap. Therefore, PhoneGap is not suitable for making graphic intense games.

To summarize, if you want to quickly develop an app for multiple platforms and are willing to compromise in the design and performance of the app you might find PhoneGap very useful and economical to use.

6.3 Future work

Features that could be added to the thesis app are for instance the secondary requirements that never were added, i.e. registration of conference participants, Twitter-feeds and full support of web browsers. The app is developed as an open-source project and the source-code is free to download, use and modify. The thesis app can, with small modifications, be fitted to any other conference or event. [5]

References

1. Beckett, C., *sellmymobile.com.* Windows Phone 7.5 (Mango) to support micro hdmi. http://www.sellmymobile.com/blog/windows-phone-7-5-mango-to-support-micro-hdmi/. Retrieved May 2012.

2. Blandford, R., *All About Windows Phone.* Windows Phone marketplace pass 80 000 apps. http://allaboutwindowsphone.com/news/item/14554_Windows_Phone_Marketplace_pass.php. Retrieved May 2012.

3. Boles, D., *Sooper Articles.* Battery comparisons for Blackberry Android Apple and Windows smartphones. http://www.sooperarticles.com/technology-articles/gadgets-gizmos-articles/battery-comparisons-blackberry-android-apple-windows-smartphones-897041.html. Retrieved May 2012.

4. Broulik, B., (2011) *Pro jQuery Mobile*, Apress

5. Calla Guzmán, J., Cernea, V., *Github.* devmobile. https://github.com/josecallaguzman/devmobile/blob/master/DevM/www/index.html. Retrieved May 2012.

6. Clark, S., *HTML Goodies.* Web-based mobile apps of the future using HTML 5, CSS and JavaScript. http://www.htmlgoodies.com/beyond/article.php/3893911/Web-based-Mobile-Apps-of-the-Future-Using-HTML-5-CSS-and-JavaScript.htm. Retrieved May 2012.

7. Dilger, D., E., *Apple Insider.* Apple's rivals battle for iOS scraps as app market sales grow to $2.2 billion. http://www.appleinsider.com/articles/11/02/18/rim_nokia_and_googles_android_battle_for_apples_ios_scraps_as_app_market_sales_grow_to_2_2_billion.html. Retrieved May 2012.

8. Edmonds, R., *WPCentral.* Windows Phone still lagging but should hit 8.6% marketshare says Gartner. http://www.wpcentral.com/windows-phone-still-lagging-should-hit-86-marketshare-says-gartner. Retrieved May 2012.

9. Elgin, B., *Bloomberg businessweek.* Google buys Android for its mobile arsenal. http://www.businessweek.com/technology/content/aug2005/tc20050817_0949_tc024.htm. Retrieved May 2012.

10. Hammond, D., *Webdevout.* Web browser standard support. http://www.webdevout.net/browser-support. Retrieved May 2012.

11. Haselton, T., *BGR.* Android market surpases 10 billion app downloads; Google kicks off $0.10 app sale. http://www.bgr.com/2011/12/06/android-market-surpasses-10-billion-app-downloads-google-kicks-off-0-10-app-sale/. Retrieved May 2012.

12. Hee, S., T., *Hardware.Zone.com.* Android 4.0, iOS 5 and Windows Phone 7.5 in comparison. http://www.hardwarezone.com.sg/feature-android-40-ios-5-and-windows-phone-75-comparison?utm_source=hwzsgforums&utm_campaign=Autopost%2BEdt%2Bin%2BForums&utm_contentcontent=Android%2B4.0%2C%2BiOS%2B5%2Band%2BWindows%2BPhone%2B7.5%2Bin%2BComparison&utm_medium=referral. Retrieved May 2012.

13. Honan, M., *Macworld.* Apple unveils iPhone. http://www.macworld.com/article/1054769/iphone.html. Retrieved May 2012.

14. Imran, A., *Redmond Pie.* iOS 5 vs. Android 4.0 Ice Cream vs. Windows Phone 7.5 Mango - Comparison. http://www.redmondpie.com/ios-5-vs.-android-4.0-ice-cream-sandwich-vs.-windows-phone-7.5-mango-comparison/. Retrieved May 2012.

15. Jacobowitz, PJ, *PCMAG.* iPhone 4 display quality beats Samsung Galaxy S. http://www.pcmag.com/article2/0,2817,2369761,00.asp. Retrieved May 2012.

16. Johnson, B., *the Guardian.* Apple iPad will choke innovation, say open internet advocates. http://www.guardian.co.uk/technology/2010/feb/01/apple-ipad-choke-innovation. Retrieved May 2012.

17. Kahney, L., *Cult of Mac.* Pundits on the iPad's closed system: it's doom for PCs, no it's great. http://www.cultofmac.com/28440/pundits-on-the-ipads-closed-system-its-doom-for-pcs-no-its-great/. Retrieved May 2012.

18. Kindel, C., *The Windows blog.* Windows Phone 7 series UI design & interaction guide. http://windowsteamblog.com/windows_phone/b/wpdev/archive/2010/03/18/windows-phone-7-series-ui-design-amp-interaction-guide.aspx. Retrieved May 2012.

19. Kweazy., *The Verge.* Windows Phone: What it needs to succeed. http://www.theverge.com/2012/4/5/2927704/windows-phone-what-it-needs-to-succeed. Retrieved May 2012.

20. Goasduff, L., Pettey, C., *Gartner.* Gartner Says Worldwide Smartphone Sales Soared in Fourth Quarter of 2011 With 47 Percent Growth. http://www.gartner.com/it/page.jsp?id=1924314. Retrieved May 2012.

21. Lunny A. (2011) *PhoneGap Beginner's Guide*, Packt Publishing

22. Massey, S., *Portfolio of Stéphane Massey.* Metro UI design principles. http://www.stephanemassey.com/metro-design-principles/. Retrieved May 2012.

23. Mies, G., *PCWorld.* iPhone 4S battery life: How does it compare to other smartphones. http://www.pcworld.com/article/243242/iphone_4s_battery_life_how_does_it_compare_to_other_smartphones.html. Retrieved May 2012.

24. Mitchell, J., *Read Write Web.* Google Play: Movies, Books, Music & Apps from Google. http://www.readwriteweb.com/archives/google_play_movies_books_music_apps_from_google.php. Retrieved May 2012.

25. Mitosh., *MobileSpect.* Top Retina display alternative phones you would love to buy. http://www.mobilespect.com/top-retina-display-alternative-phones-you-would-love-to-buy.html. Retrieved May 2012.

26. Molen, B., *Aol Engadget.* Window Phone 7.5 Mango in-depth preview. http://www.engadget.com/2011/06/27/windows-phone-7-5-mango-in-depth-preview-video/. Retrieved May 2012.

27. Panda, K., *EzineMark.com.* The announcment of iOS 5 and Android 4.0. http://android.ezinemark.com/the-announcement-of-the-ios-5-and-android-40-7d355f0c4b69.html. Retrieved May 2012.

28. Paul, I., *PCWorld.* Android market tops 400 000 apps. http://www.pcworld.com/article/247247/android_market_tops_400000_apps.html. Retrieved May 2012.

29. Paul, I., *PCWorld.* Five ways Apple rips-off Android with iOS 5. http://www.pcworld.com/article/229611/five_ways_apple_ripsoff_android_with_ios_5.html. Retrieved May 2012.

30. Peak, K., *Vator News.* Amazon app store makes 89 cents for every iTunes dollar. http://vator.tv/news/2012-03-30-amazon-app-store-makes-89-cents-for-every-itune-dollar. Retrieved May 2012.

31. Polimenov, A., *Silverlight show.* What is Windows Phone 7. http://www.silverlightshow.net/items/WP7-What-is-Windows-Phone-7.aspx. Retrieved May 2012.

32. Ragel, R., *DifferenceBetween.com.* Difference between Apple A5 and Qualcomm Snapdragon S3. http://www.differencebetween.com/difference-between-apple-a5-and-vs-qualcomm-snapdragon-s3/. Retrieved May 2012.

33. Ricknäs, M., *PCWorld.* Bis is not always better with Smartphones screens. http://www.pcworld.com/article/239521/big_is_not_always_better_with_smartphone_screens.html. Retrieved May 2012.

34. Rosenberg, J., *Google Official Blog.* Introducing Google Play: All your entertainment, anywhere you go. http://googleblog.blogspot.se/2012/03/introducing-google-play-all-your.html. Retrieved May 2012.

35. Slivka, E., *Macrumors.* Apple’s App store reaches 25 billion downloads, now on pace for 15 billion apps per year. http://www.macrumors.com/2012/03/03/apples-app-store-reaches-25-billion-downloads-now-on-pace-for-15-billion-apps-per-year/. Retrieved May 2012.

36. Smith, J., *9to5 Google.* Google announces Google music, features heavy Android market and Google plus integration. http://9to5google.com/2011/11/16/google-announces-google-music-with-heavy-android-integration/. Retrieved May 2012.

37. Soulskill., *Slashdot.* Apple’s trend away from tinkering. http://apple.slashdot.org/story/10/01/31/1657233/Apples-Trend-Away-From-Tinkering. Retrieved May 2012.

38. Taylor, W., *Android and me.* Android 2.0 does indeed support multitouch. http://androidandme.com/2009/10/news/android-2-0-does-indeed-feature-multitouch/. Retrieved May 2012.

39. Toor, A., *Gartner*. Gartner: Apple leads the way among smartphone vendors, Android sees slight decline. http://www.engadget.com/2012/02/15/gartner-q4-2011-apple-android-smartphone/. Retrieved May 2012.

40. Wauters, R., *TechCrunch.* Distimo: 57% of Android apps are free vs. 28% of iPhone apps. http://techcrunch.com/2010/07/05/distimo-june-2010/. Retrieved May 2012.

41. Wilson, J., L., Fenton, W., *PCMAG.* Smartphone survey: iOS 5 vs. Android 2.3 Gingerbread vs. Windows Phone 7.5 Mango vs. Blackberry 7 OS. http://www.pcmag.com/article2/0,2817,2386579,00.asp. Retrieved May 2012.

42. Worthington, D., *Technologizer*. Microsoft: iPad’s Closed platform is humorous. http://technologizer.com/2010/01/28/microsoft-ipads-closed-platfor-is-humorous/. Retrieved May 2012.

43. Android Developers. *Google.* Platform versions. http://developer.android.com/resources/dashboard/platform-versions.html. Retrieved May 2012.

44. Android Developers. *Google.* Tools. http://developer.android.com/guide/developing/tools/index.html. Retrieved May 2012.

45. Android Developers. *Google.* What is Android? http://developer.android.com/guide/basics/what-is-android.html. Retrieved May 2012.

46. Android Developers. *Google.* Android emulators. http://developer.android.com/guide/developing/tools/emulator.html. Retrieved May 2012.

47. Apple support centre. *Apple.* iOS developer library. About xCode. https://developer.apple.com/library/ios/#documentation/ToolsLanguages/Conceptual/Xcode4UserGuide/Introduction/Introduction.html. Retrieved May 2012.

48. Apple support centre. *Apple.* Developer. xCode. https://developer.apple.com/support/xcode/. Retrieved May 2012.

49. Apple Wiki. iPhone 4S. http://apple.wikia.com/wiki/IPhone_4S. Retrieved May 2012.

50. BuiltWith technology usage statistics. jQuery usage statistics. http://trends.builtwith.com/javascript/JQuery. Retrieved May 2012.

51. Eclipse. Eclipse Public license – v 1.0. http://www.eclipse.org/legal/epl-v10.html. Retrieved May 2012.

52. Eclipse community survey. *Eclipse.* The open source developer report. http://www.eclipse.org/org/community_survey/Eclipse_Survey_2011_Report.pdf. Retrieved May 2012.

53. Google Play. *Google.* Presentation av Google Play. https://play.google.com/about/. Retrieved May 2012.

54. iOS developer library. *Apple.* About the tools workflow for iOS. http://developer.apple.com/library/ios/#documentation/Xcode/Conceptual/ios_development_workflow/00-About_the_iOS_Application_Development_Workflow/introduction.html#//apple_ref/doc/uid/TP40007959. Retrieved May 2012.

55. iOS developer library. *Apple.* Using iOS simulator. http://developer.apple.com/library/ios/#documentation/Xcode/Conceptual/ios_development_workflow/25-Using_iOS_Simulator/ios_simulator_application.html#//apple_ref/doc/uid/TP40007959-CH9-SW1. Retrieved May 2012.

56. jQuery. *jQuery project.* License. http://jquery.org/license. Retrieved May 2012.

57. jQuery Mobile. *The jQuery foundation.* Quick start guide. http://jquerymobile.com/demos/1.1.0/#/demos/1.1.0/docs/about/getting-started.html. Retrieved May 2012.

58. Microsoft download Center. *Microsoft.* Windows Phone SDK 7.1. http://www.microsoft.com/download/en/details.aspx?id=27570. Retrieved May 2012.

59. Microsoft help and support. *Microsoft.* Description of Visual studio 2010 service pack 1. http://support.microsoft.com/kb/983509. Retrieved May 2012.

60. MSDN library. *Microsoft.* Developing a Windows Phone application from start to finish. http://msdn.microsoft.com/en-us/library/gg680270(v=PandP.11).aspx. Retrieved May 2012.

61. MSDN. *Microsoft.* Multitasking for Windows Phone. http://msdn.microsoft.com/en-us/library/hh202866(v=vs.92).aspx. Retrieved May 2012.

62. MSDN. *Microsoft.* Windows Phone emulator. http://msdn.microsoft.com/en-us/library/ff402563(v=vs.92).aspx. Retrieved May 2012.

63. MSDN. *Microsoft.* Windows Phone marketplace test kit. http://msdn.microsoft.com/en-us/library/hh394032(v=vs.92).aspx. Retrieved May 2012.

64. National Library of the Netherlands. What is Emulation? http://www.kb.nl/hrd/dd/dd_projecten/projecten_emulatiewatis-en.html. Retrieved May 2012.

65. Perfecto Mobile Ltd. Real handsets vs. simulators. http://51degrees.mobi/Portals/0/Documentation/Real%20handsets%20vs%20simulators.pdf. Retrieved May 2012.

66. PhoneGap. http://phonegap.com/about/features. Retrieved May 2012.

67. Qualcomm. Devices featuring Snapdragon processors. http://www.qualcomm.com/snapdragon/devices. Retrieved May 2012.

68. Techopedia. Native Mobile App. http://www.techopedia.com/definition/27568/native-mobile-app. Retrieved May 2012.

69. w3schools.com. CSS introduction. http://www.w3schools.com/css/css_intro.asp. Retrieved May 2012.

70. w3schools.com. JavaScript introduction. http://www.w3schools.com/js/js_intro.asp. Retrieved May 2012.

71. w3schools.com. jQuery introduction. http://www.w3schools.com/jquery/jquery_intro.asp. Retrieved May 2012.

72. Wikipedia. Comparison of smartphones. http://en.wikipedia.org/wiki/Comparison_of_smartphones. Retrieved May 2012.

73. Wikipedia. Mobile application development. http://en.wikipedia.org/wiki/Mobile_application_development. Retrieved May 2012.

74. Wikipedia. iOS. http://en.wikipedia.org/wiki/IOS. Retrieved May 2012.

Appendix

Appendix A1: Thesis application requirements

The primary requirements are the requested requirements from the client and were therefore highly prioritized. The secondary requirements were optional and supposed to be implemented in case of early prototype completion.

A1.1 Primary requirements

- The application shall work on three platforms, Android, iOS and Windows Phone
- The application shall be published on the corresponding markets
- The application shall contain a presentation of the conference
- The application shall contain an outline of the schedule
- The application shall contain a function to combine your own schedule
- The application shall contain a view of all the speakers
- The application shall contain a view of all the subjects
- The application shall contain a view of all the sponsors
- The application shall contain a view of all the contributors

A1.2 Secondary requirements

- The application should have registration of the conference participants
- The application should support Twitter-feeds
- The application should work on web browsers

Appendix A2: Apple iOS 5 device compatibility

Apple iOS 5 is compatible with the following iOS devices: iPhone 3GS, iPhone 4, iPhone 4S, iPad, iPad 2, and iPod touch (third and fourth generation).
Apple iOS 5 is not compatible with the following iOS devices: iPhone, iPhone 3G, and iPod touch (first and second generation).