

ASHESI UNIVERSITY COLLEGE
An Introduction to Mozilla's Firefox OS:
Customization and Application Development

Mawuena Senam Ahadzie

2014

Applied Project

ASHESI UNIVERSITY COLLEGE

**An Introduction to Mozilla's Firefox OS:
Customization and Application Development**

By

MAWUENA SENAM AHADZIE

Dissertation submitted to the Department of Computer Science

Ashesi University College

In partial fulfillment of the requirements for the award of Bachelor of Science

Degree in Management Information Systems

APRIL 2014

I hereby declare that this dissertation is the result of my own original Work and that no part of it has been presented for another degree in this university or elsewhere.

Candidate's Signature:.....

Candidate's Name:.....

Date:.....

I hereby declare that the preparation and presentation of the dissertation were supervised in accordance with the guidelines on supervision of dissertation laid down by Ashesi University College.

Supervisor's Signature:.....

Supervisor's Name:.....

Date:.....

Acknowledgements

I would like to thank my parents Rev Ben Ahadzie and Mrs. Dorothy Ahadzie for their immense support and prayers.

Special thanks to my supervisor, Mr. Kwadwo Gyamfi Osafo-Mafo for guiding me through this project. You are awesome.

To Nina Chachu, Head Librarian at Ashesi University, I am grateful.

To my friends and family, I did this for you guys.

Abstract

The smartphone market is steadily increasing all over the world, as such, most mobile device users are gradually shifting from feature phones to smartphones, tablets and portable computing devices. However, in emerging markets, majority of the mobile device users still use their feature phones, presenting an opportunity for OEM'S and operating system manufacturers to produce a phone that would suit this market.

Mozilla, the company that popularized the open source browser Firefox, has developed a mobile operating system. Hopefully, this operating system would provide smartphones at affordable prices to consumers who cannot afford the high-end devices that are currently sold on the market. This paper seeks to explore Firefox operating system to give an in-depth understanding on how the operating system works.

Table of Contents

Acknowledgements.....	iv
Abstract.....	v
Chapter 1: INTRODUCTION	1
1.1 Introduction and Background to the Project	1
1.2 The architecture of Firefox Operating System.....	3
1.3 Relevance of this project.....	6
1.4 Motivation.....	6
1.5 Related projects	8
2 Chapter 2: OBJECTIVE	10
3 Chapter 3: APPROACH.....	11
3.1 Explore the background for the Firefox OS.....	11
3.2 Searching for versions of source code and simulators	11
3.3 Compile source code by getting tools and software necessary.	12
3.4 Explore how to modify underlying core system	12
3.5 Explore the API for building applications and system components	12
4 Chapter 4: IMPLEMENTATION	13
4.1 Introduction	13
4.2 Exploring the background of the operating system.....	13
4.3 Downloading the Source Code and compiling it.....	13
4.4 Modifying the underlying core system	15
4.5 Evaluation of the API structure and App Development.....	19
4.6 Comparison between Android and Firefox OS	22
4.7 Installation and Testing on a phone.....	26
5 Chapter 5: Evaluation.....	28
6 Chapter 6: Conclusion	30
Bibliography	31
Appendices.....	36

Chapter 1: INTRODUCTION

1.1 Introduction and Background to the Project

Firefox operating system is a new mobile operating system that is based on the open web standards: HTML5, JavaScript and CSS. Considering the number of HTML and JavaScript developers available, Firefox OS presents an opportunity for them to exhibit their creativity and show the skills they have, by developing applications that would run on the operating system or by customizing parts of the operating system to include features that suit particular users.

Since the launch of the operating system in 2013 a number of OEM's (Original Equipment Manufacturers) have expressed interest in having the operating system run on their phones. A month after the operating system was launched, a total of nine hundred and eighty phones were sold on eBay's website in America, whilst in the United Kingdom, the number of sold out units on eBay totaled nine hundred and ninety. [1] To manufacturers of mobile devices this is an opportunity to attract more customers.

Firefox operating system is open source, so the source code is readily available. In my opinion, this presents telecom operators with a viable opportunity because they would be able to customize the operating system so that it would have features, color schemes and other properties that would match their brands.

The rapid growth and development of smartphones in the world has led to competition among the manufacturers of mobile operating systems to come

up with software that is capable of providing customers with the full functionality that their phones can offer. In 2013, research by the Business Insider team revealed that on average, for every nine people on earth, two of them have smartphones. The research also showed that the number of smartphones in the world amounts to about 1.4 billion. This means that as the years go by, more people would use smartphones and this would make feature phones less prevalent. [2]

The mobile operating system market in the world has been dominated by Apple's IOS and Google's Android. Together, these firms hold about 92.5 % of the market share in the mobile operating system market. [3] Despite their market share, it is very difficult for users to switch apps between the two operating systems. For instance, if a smartphone user switched between an android phone and an iPhone, it would be impossible to have his apps from android on IOS, especially if they are not available in the app store. The reason for this disparity is that IOS apps are built using objective-C and Android apps with Java so a developer would need to have knowledge of these languages before he/she would be able to build apps to suit both markets. Mozilla realizing the impact of the duopoly which currently exists in the market for mobile operating systems decided to come up with Firefox OS. Mozilla believes that this operating system comes in at the right time to help ease the burden of developers when they have to develop for other mobile operating system platforms. They also believe that this operating system would help customers because, they can have access to their favorite apps so far as there is access to the internet. [4]

Firefox OS also known by its project name, boot to gecko, is an open source mobile operating system developed by Mozilla. It is a Linux based operating system that is set to run on low end smartphone devices as well as tablets. The first commercial mobile device running Firefox OS was launched on July 2013 in Spain by Telefonica. [5] The operating system has a core which is built on the open web standards employed by the web development community. HTML5, JavaScript, CSS and anything else that can be used for web development can be used to develop apps for Firefox OS. [6]

The image below shows the first phone released by Telefonica.



1.2 The architecture of Firefox Operating System

The architecture of Firefox OS is based on three main layers. Gaia, Gecko and Gonk. Being the topmost layer, Gaia forms the user interface of the Firefox operating system. Anything that is displayed on the screen when the phone loads is implemented by Gaia. For example the lock screen, the home screen and all the standard applications available on a modern smartphone are implemented in this layer. HTML5, CSS and JavaScript are the main technologies used in the implementation of the Gaia layer. In order for this

layer to effectively interact with the operating system, it makes use of channels known as Web API's.

The second layer is Gecko. It provides all of the support for the layer above it. Gecko also ensures that the APIs work well on every operating system that it supports. In order to ensure that the interaction between Gaia and Gonk is efficiently performed, the Gecko layer has a networking stack, graphics stack, a layout engine and a JavaScript virtual machine.

The last layer is known as Gonk and it forms the operating system of the Firefox OS platform. This layer consists of a Linux kernel, user space and a hardware abstraction layer. Gonk is described as a porting target of Gecko; meaning that there is a port of Gecko to Gonk allowing Gecko to have direct access to the telephony stack and display frame buffer on Gonk. The diagram below gives a clearer understanding of the structure of the operating system.

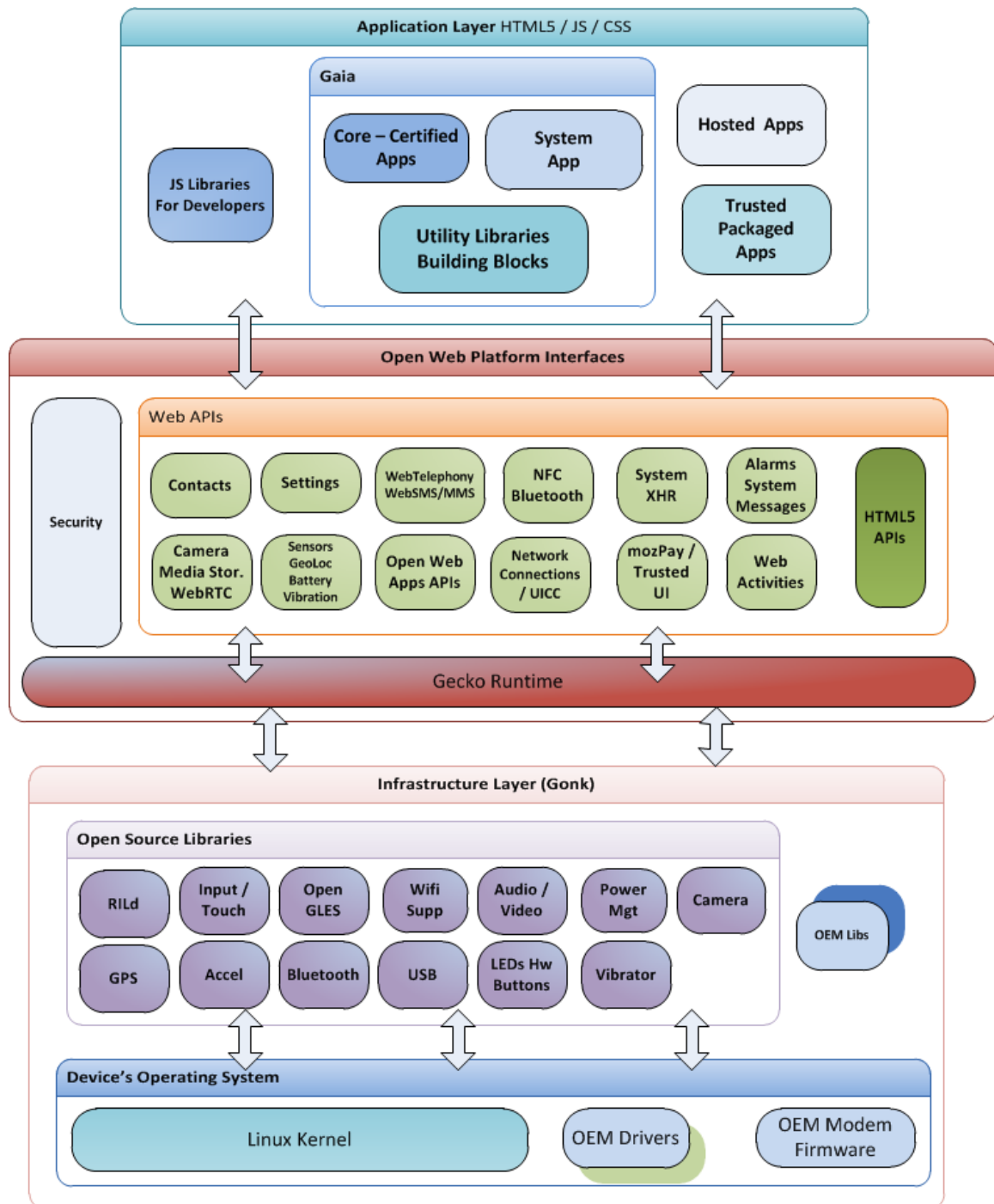


Figure 1.1 – The Architectural Structure of Firefox OS [7]

1.3 Relevance of this project

Elance an online freelance marketplace has about 116,549 registered web developers as compared to 57,114 software developers. These numbers are an estimate of the increased number of web developers as compared to software developers. Since Firefox OS is web based, it presents all these people with an opportunity to develop apps that would work on the operating system and also on the web.

Firefox OS is relatively new, and people have written quite a number of articles about it. I believe that by undertaking this project I would be able to come up with a paper that would serve as a reference for anyone who wishes to undertake a project related to Firefox operating system.

1.4 Motivation

This project is motivated by the author's interest in developments in the mobile industry where phone manufacturers are gradually shifting to the production of phones that are capable of performing transactions similar to what computers can do. To ensure that these phones perform as expected, they require operating systems that are complex enough to be able to provide the needed performance. In the advanced countries, the market for smartphone devices has been saturated with high tech devices that have a lot of fancy features, however there is still room for growth of these devices in a number of countries within Africa and Asia.

Research shows that in the fourth quarter of 2012, the estimated total number of feature phones shipped totaled 264.4 million units. [8] However,

an article suggests that the smartphone market is growing most especially in countries with emerging markets. In Africa, the rate of growth of smartphones has been 19% year over year. [9] This goes to show that a firm that is able to provide a smartphone at a very affordable price would experience a significant increase in its market share, and that is what Firefox OS hopes to achieve by giving OEMs'(Original Equipment Manufacturers) an opportunity to provide smartphones at the cheapest price possible. Telefónica, one of the largest telecom companies and provider of communication, information and entertainment solutions, with over 320 million customers has proven its strong support and backing for the operating system. The Director of Open Web Devices at Telefónica Digital is quoted as saying, "There is a strong momentum building behind Firefox OS as we bring more devices to more of our markets. But this is just the beginning as the platform, content ecosystem and range of devices continues to expand. There is an inherent imbalance in the smartphone market at the moment and we believe that Firefox OS represents the best opportunity to restore balance and provide a more open smartphone experience to customers." [10].

Finally, because Firefox OS is open source and developed by an open community of contributors I am greatly motivated because I am confident that in my own small way I would have contributed to the development of a project that would soon be changing the face of smartphone development and encouraging the growth of smartphones in emerging markets.

1.5 Related projects

From my research I realized that there has been an attempt at implementing an operating system which is JavaScript based: the Skylight project. This operating system is similar to Firefox OS in the sense that it is a web operating system that is written using only JavaScript. The fascinating thing about the Skylight project is that it provides operating system-like features using cloud computing concepts, making it possible to run on most of the modern internet browsers that are available. Anyone who wants to use the Skylight project can simply navigate to the webpage, create an account and then have an operating system that has fully functional word processor, a spreadsheet, paint, file browser and an application development studio. [11]

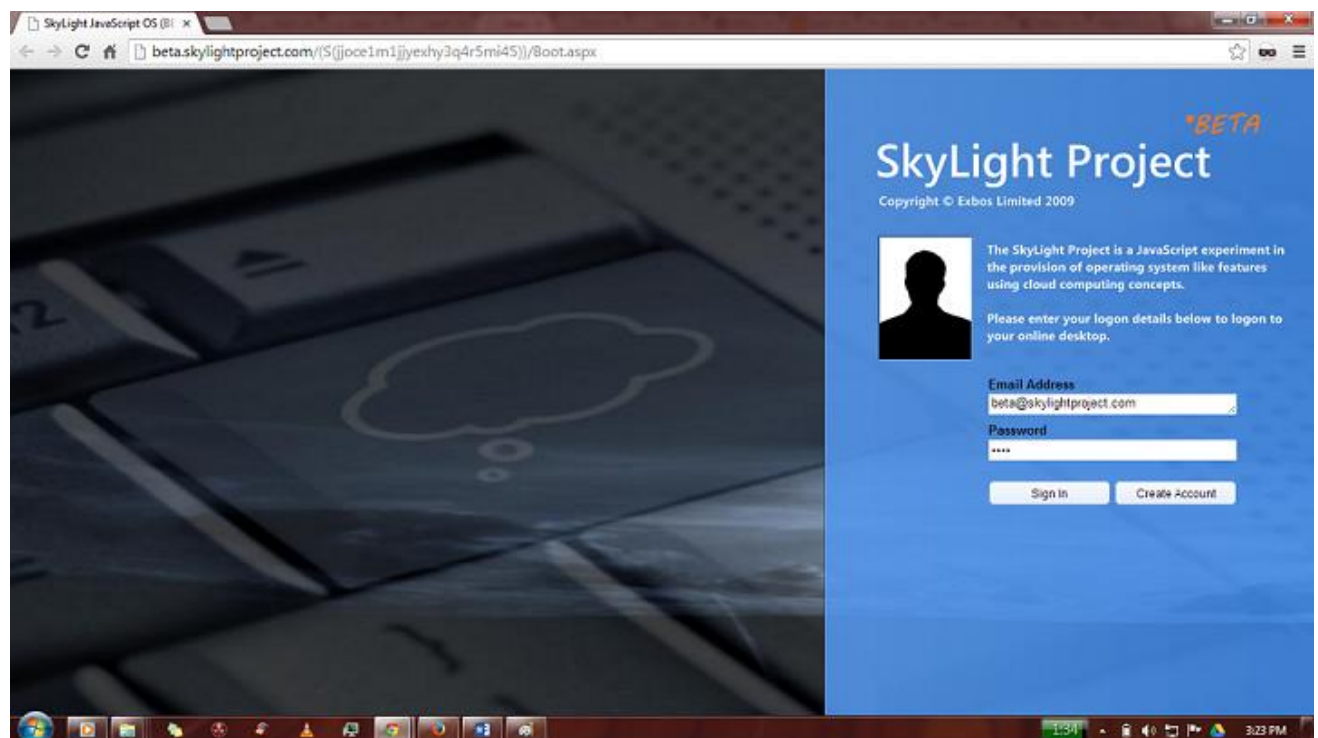


Figure 1.2- The login page of Skylight Project

Another project that can be described as similar to Firefox OS is Google's Chromium OS. This operating system is also open source and aims to provide computer users who spend most of their time online, with an operating system that is fast, simple and secure. In Chrome OS every activity takes place within the browser, meaning, there are no conventional desktop applications. Without conventional desktop applications, the operating system takes away the stress users face with installing, managing and updating programs. [12] All the packaged apps that run on Chrome OS are written in HTML, JavaScript and CSS. Despite the fact that the operating system is web based, it is designed such that users still have apps comparable to native apps even when there is no network connection available. [13] Google claims that by running all the apps within the browser, each app is contained in a sandbox which makes it difficult for malware and viruses to infect your computer. [12]

Jason Evangelho in his article on why Microsoft is worried about Google's Chrome books suggests that in 2012, the percentage of Chrome books that were purchased in America made up 0.2% of total computer purchases. He states further that in 2013, the Chrome book penetration has skyrocketed to a 9.6% share indicating an increase in the number of computer users who prefer to be online most of the time. Since the world is becoming more connected via the internet, the chrome book could become the preferred choice for people looking to buy computers. [14]

2 Chapter 2: OBJECTIVE

The General objective of this project is to explore the Firefox mobile operating system and then provide guidelines on the best steps to take if you would want to build or customize the system. This project will explore how to compile source code for Firefox OS and also customize certain components of the operating system as well as create installable applications. This exploration would hopefully create the gateway for people to realize the impact that would be made by Firefox with the release of the operating system.

3 Chapter 3: APPROACH

In order to efficiently understand the structure and development model of the operating system, this project will explore how to compile and customize the operating system. Secondly, an evaluation of the API structure would be done so as to understand how HTML5 and JavaScript are used in building an operating system. Thirdly, the project would do a comparison between Firefox OS and another operating system. Possibly, an analysis of the tools and IDE's for development to be able to understand why Firefox OS differs from other mobile operating systems.

3.1 Explore the background for the Firefox OS.

An understanding of the history of Firefox OS would be key in determining the essence of the operating system and probably what the developers of this operating system hoped to achieve. Research to identify the release date of the operating system would help in estimating the impact that the operating system has made, and if it is on the path to achieving its goal.

3.2 Searching for versions of source code and simulators as well as the necessary tools will also be very handy considering the nature of this project. Firefox OS is open source and has its source code available. It would be important to identify the best place to download the source code. Since the project is still under development, identifying the version of the code to download that would suit the objective of the project is essential.

3.3 Compile source code by getting tools and software necessary.

After identifying and downloading the code, the next step would be identifying the tools necessary for development. It would be important to look at the best operating system to run the build effectively and tools that would enable a developer read, modify the code and makes appropriate changes to suit the market for which he/she may be developing.

3.4 Explore how to modify underlying core system

One of the goals of the project is to be able to explore the constituents of the operating system and then possibly customize some key system components like the keyboard, home screen and Bluetooth. In most operating systems, the foundational structures assist people to interact with the hardware on which the operating system is running.

3.5 Explore the API for building applications and system components

An API specifies how software components interact with each other. Knowing how the API works in this application, would allow developers to create applications that are dependent on each other. This means that it should be possible for a user to send an SMS or make a phone call from within the app.

4 Chapter 4: IMPLEMENTATION

4.1 Introduction

The implementation section of this project details with how to compile the operating system, how to customize the application layer and core components. It explores the use of the API and how to build applications. Compilation of the code involved downloading and compiling the source code. For customization I looked at changing the source code and then changed the wallpaper on the home screen using a script

4.2 Exploring the background of the operating system

Firefox OS is built using HTML5 and other open web standards, making it free from the rules and restrictions of existing proprietary platforms. In 2013, the first commercial device running this operating system was released in Spain. Firefox claims that they have been instrumental in innovating and shaping the web to maintain its openness and accessibility. As a result, one fifth of all web activity happens through Firefox products. Influenced by the desire to see the internet accessible to all, Mozilla's goal for this operating system is to provide users with smartphones that understand and deliver what they need. [15]

4.3 Downloading the Source Code and compiling it

Mozilla offers developers three main methods by which the operating system can be deployed to encourage development. First and foremost the source code for the operating system can be downloaded and then installed onto a compatible mobile device. The second option for developers, is to configure

an emulator and then deploy the source code on it. This provides them an opportunity to make modifications and see the impact of the modifications that are made. The last option is to do a build on a Linux distribution. After having consulted my supervisor, we identified that it would be best to do the build and exploration on a computer running Ubuntu since Firefox OS is Linux based.

In order to effectively to do a desktop build, Ubuntu 12.04 was installed on a virtual machine. Mozilla requires that there must be some prerequisite software installed on the build computer, and this involved downloading and installing all the software necessary for the development. First and foremost, the Linux command “sudo apt-get update” would update the system. After the update is complete the next step would be to run the command “sudo apt-get install autoconf2.13 bison bzip2 ccache curl flex gawk gcc g++ g++-multilib gcc-4.6 g++-4.6 g++-4.6-multilib git ia32-libs lib32ncurses5-dev lib32z1-dev libgl1-mesa-dev libx11-dev make zip” which installs all the software that is needed to build the entire operating system from scratch. There are also hardware prerequisites that Mozilla suggests for the computers on which the operating system would be built. For example, the computer should have at least a 2G RAM with lots of available swap space. They also require that for debug builds, there should be at least 8GB free disk space and 1GB free disk space for optimized builds. [6]

After setting up the computer to start building, I then downloaded the source code for Firefox OS and Gaia into a source directory. After downloading the files, I navigated to the Mozilla directory from terminal and then issued the

command; `./mach build`. The command builds the profile of Firefox OS. The next step is to run `DESKTOP_SHIMS=1 NOFTU=1 DEBUG=1 make` in the Gaia directory; which creates a profile directory within the Gaia directory. The profile contains a customized extension and other necessary files needed to make B2G run properly. The command, `.../b2g-bin -profile Gaia/profile-debug` is used to run the compiled code. Unfortunately, I was not able to use the compiled source code to understand the structure of the operating system because the computer could not load the required startup files. After updating the code several times and discussing the situation with my supervisor, he suggested that an option to explore, would be to use the simulator plugin that can be downloaded and installed in the Firefox browser. This plugin has a similar code structure as compared to the one that I downloaded and compiled.

[4.4 Modifying the underlying core system](#)

After downloading the plugin that simulates the operating system in the Mozilla browser, I ran it a few times to make sure that it was working as expected.

I then decided to look at how the structure of the keyboard implemented, as the first step in understanding how to modify the structures of the operating system. Initially, I had planned to implement a full keyboard with the characters from my native language Ewe. However, after doing research I realized that most of the characters in the Ewe alphabet are similar to the English one with some slight variations. With that, I decided it would be most appropriate to incorporate the Ewe characters as part of the English

keyboard so that users can simply long press buttons on the keyboard to make use of the letters.

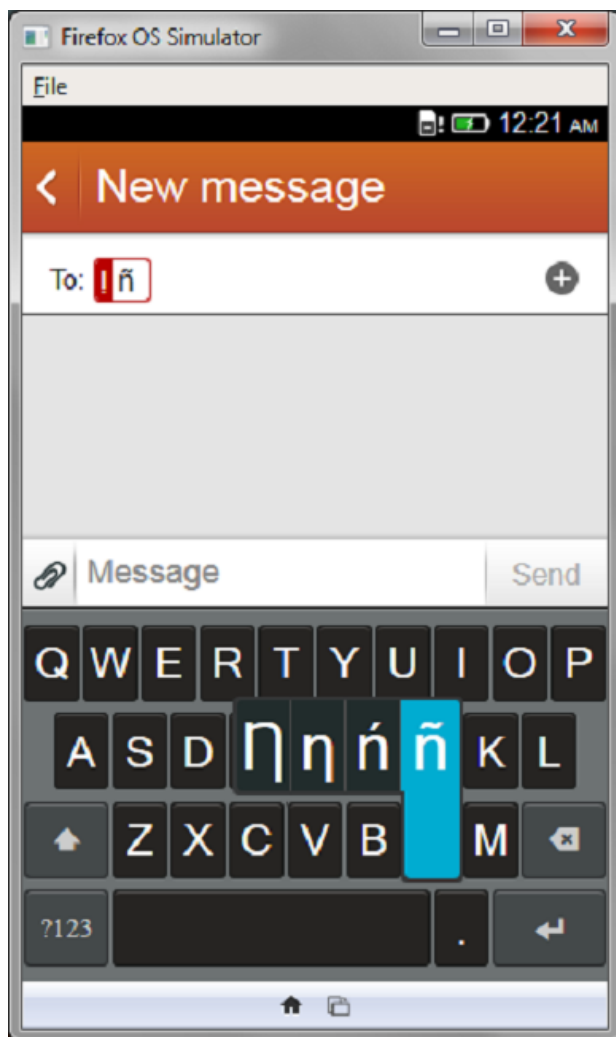
IV. Phonemes

The alphabet for writing Ewe is generally written thus:

a b d d e e f f g h y x i k l m n ŋ o o p r s t u v
u w y z.

Figure 4.1 shows the characters that form the Ewe alphabet

The image below shows a Snapshot of the customized keyboard on the simulator.



In addition to customizing the keyboard, I was able to customize the wallpaper that shows on the home screen. The reason for considering this is that when a developer is able to change the wallpaper based on the time of the day, it would not be difficult to relay real time information to the phone's home screen. Using JavaScript, I wrote a script that changes the wallpaper on the screen after every ten seconds. Unfortunately, one challenge I faced with this part of the customization was that when the images on the screen are changing the transition is not smooth. The screen shows a jittery image before it changes completely. The images below show the screen when the wallpapers have changed.



4.5 Evaluation of the API structure and App Development

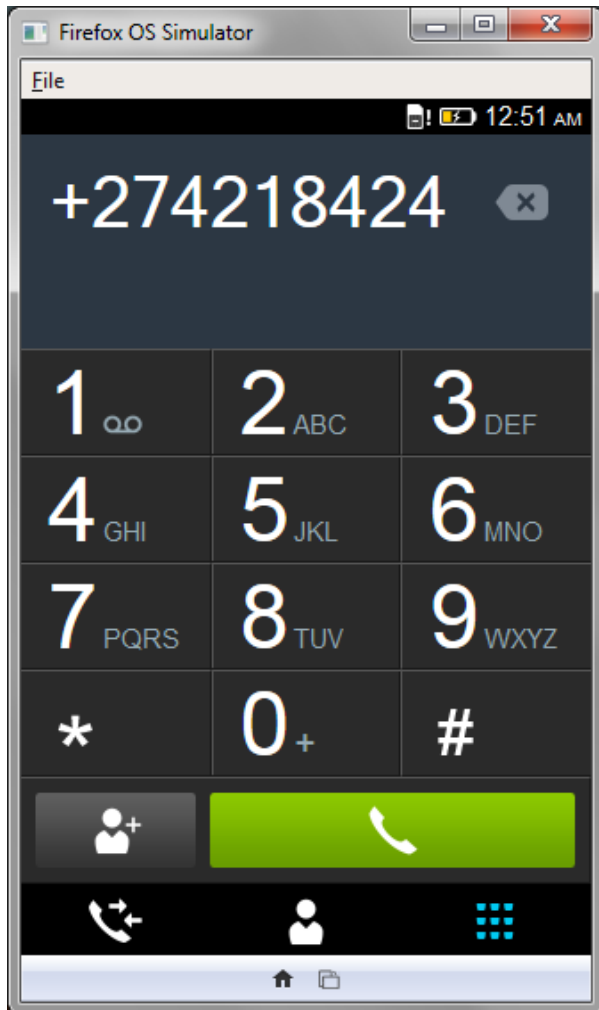
The fourth thing I did was to explore how JavaScript and HTML5 are integrated with the phone API's to create the functionality of the phone. JavaScript is commonly used on the World Wide Web as a client side scripting language. Essentially, the JavaScript component of most applications allow the developer to create functions that foster interactions with the user. HTML on the other hand handles the layout and structure of elements that make up the user interface of the system. In Firefox OS, HTML is used to create layouts for tables, forms etc. However, HTML needs style sheets to help add color, font types and other design elements to the designed HTML pages, so as to improve the design of the user interface. I decided to develop an application using jQuery mobile. This application has four tabs, the home tab or the first tab allows the users to read RSS news feeds on Ashesi, which I generate from the Ashesi Foundation website. The next tab loads videos from Ashesi's YouTube channel so that users can easily stream and watch videos that are related to Ashesi. The third tab gives a user directions to Ashesi with a map generated with google maps so that users of the app would be able to find their way to Ashesi's campus. The last tab of the application would allow users to make calls using the phones' dialer. In other words there are buttons that can be clicked by the user and then it outputs dedicated numbers to the dialer screen enabling the user to make a phone call. The development of this app helped me understand how API's work in the operating system. In Firefox OS, the API's are referred to as mozActivities. To be able to get a user to make calls from the app using buttons, I add the mozActivity that activates phone calls to a file in the app

folder. The figure below shows the JavaScript code that enables phone calls from the phones' dialer app.

```
1  $(document).ready(function() {  
2  
3      //Shown below are just a few Intents (Web Activities)  
4      $("#btnCall").click(function(){  
5          var call = new MozActivity({  
6              name: "dial",  
7              data: {  
8                  number: "+274218424"  
9              }  
10         });  
11     });  
12     $("#btnCall1").click(function(){  
13         var call = new MozActivity({  
14             name: "dial",  
15             data: {  
16                 number: "+29348943"  
17             }  
18         });  
19     });  
20     $("#btnCall2").click(function(){  
21         var call = new MozActivity({  
22             name: "dial",  
23             data: {  
24                 number: "+274534334"  
25             }  
26         });  
27     });  
28  
29  
30  
31  
32 });
```

Figure 4.3 shows code used to make phone calls from within an app

The image below is a snapshot of how the dialer looks when it is called on the emulator.



Apart from the `mozActivity` that enables phone calls, there are other Firefox OS `mozActivities` that developers can make use of when building apps. The structure employed in activating these activities are similar, however, a few modifications must be made to implement the necessary features. The image below shows code for the implementation of a `mozActivity` for sending SMS. The difference between the SMS activity and the dial activity is that for the SMS block a keyword type is added and then using SMS or WEBSMS as the value type would engage the SMS function. There is a second `mozActivity` in the image, and this activity allows users to select pictures. By specifying the

image types within the code, users would be able to pick an image from the gallery or use the camera to capture an image.

```
13     $("#btnSMS").click(function(){
14         var newSMS = new MozActivity({
15             name: "new",
16             data: {
17                 type : "websms/sms",
18                 number: "+274218424"
19             }
20         });
21     });
22     $("#btnPickImage").click(function(){
23         var pickImageActivity = new MozActivity({
24             name: "pick",
25             data: {
26                 type: ["image/png", "image/jpg", "image/jpeg"]
27             }
28         });
29
30         pickImageActivity.onsuccess = function() {
31             //Create the element via jQuery
32             var imageElement = $('<img id="image1" height="400" width="300">');
33             imageElement.attr('src', window.URL.createObjectURL(this.result.blob));
34             $("#pickImage").html(imageElement);
35
36             //Switch to the #pageid = showImage
37             $.mobile.changePage ($("#showImage"));
38         };
39
40         pickImageActivity.onerror = function() {
41             alert("Cannot pick the image");
42             console.log(this.result);
43         };
44     });
```

Figure 4.4 shows the code for other mozActivities and how they are implemented

4.6 Comparison between Android and Firefox OS

The final stage of my implementation process was to compare the Firefox operating system to those that are already available on the market. My aim is to do a comparison between Firefox OS and Android. The reason for doing the comparison to Android is because of its increased popularity over the past few years. Also, both android and Firefox OS are open source meaning they believe in the value that the open source community has, and the role it

plays. The table below highlights the comparison between the two mobile operating systems.

Factors for Comparison	Android	Firefox
Level of expertise	Requires knowledge in Java	Requires knowledge of HTML5, JavaScript
Tools required for development.	An IDE for android development. The android SDK	A text editor, and a browser
Level of popularity	Large market share	Emerging product in mobile operating system Market
Availability of Apps	Available through the play store	Aside from having apps on the play store, users can also download apps from the internet.
Cost of handset	Average price as at 2013 is \$276[16]	Starting price for first phones manufactured is \$25
Availability of source code For testing and deployment.	Android has its source code available	Mozilla also has its source code available.

The table above shows a few factors that are used as a basis for comparison between android and Firefox OS. The first factor for comparison is the assumed level of expertise that is required by android developers. Before developing an android application it is expected that the developer has some fore knowledge of java programming. On the other hand, for a developer to produce an application for Firefox OS, he or she would need to understand the use of web technologies like HTML5, JavaScript and CSS. The entire operating system and applications that run on it are developed using these technologies so it means that developers would be able to develop apps and make system modifications at a quick rate. It is estimated that a novice would be able to understand the basics of HTML and JavaScript within 20-40 minutes. [17] This goes to show the ease with which anyone with a Firefox OS phone and a need for a computerized solution could easily develop an application without having to wait for an experienced developer to develop the app he or she needs.

For development tools, Android developers need to download an IDE and SDK as well as configure it appropriately before they can start developing applications. Developing for Firefox OS on the other hand, requires a text editor and a browser. With HTML and JavaScript, developers do not need to install an IDE or an SDK to build apps they can simply use the notepad and browser applications that come with all windows computers.

Considering the size of the mobile market owned by Android it is quite obvious that the number of android phones on the market is more as

compared to that of Firefox. Firefox OS is relatively new, and compatible phones to suit market prices are still being manufactured. So it would take a while for Firefox OS to compete with other mobile operating system manufacturers.

As at November 2012, the total estimated number of apps on Google Play; the app store for Android, totaled about 700, 000.[18] Just four months after the release of the first phone running Firefox OS, Mozilla claimed they had about a thousand apps in their app store.[19] In addition to the app store, because Firefox OS is web based, users have the option of downloading other apps that are hosted on the internet. It has been argued that engaging this style, makes users susceptible to people with malicious intents who may lace apps with viruses that may destroy the functionality of the phone. In my opinion this is subjective because the users can decide not to download from any website online.

The price at which these phones are being sold is one of the most important factors to consider.[20] Aside the performance and distinct features that smartphones offer, most consumers usually consider the price of the phone before buying it. Despite the fact that most people would assume a \$25 dollar phone may not have the best of features, Mozilla's vision of offering users a new level of choice and innovation online, can best be achieved by providing phones at the cheapest cost possible. This would mean that people in emerging economies who would spend very little on purchasing a phone

can still have access to the internet and possibly take advantage of the opportunities that it presents.

The last factor for comparison is the availability of the source code for both mobile operating systems. These operating systems believe in having their source code available to the general public so that changes can be made to it to suit the needs of various users. Unfortunately, Google employs a different strategy in the Android open source project. According to Rosenblatt, Google developers work on the Android project in their labs until they are ready. The source code for Android operating system changes are released to the public after it has been launched, limiting public input on the Android code before it is released.[21] On the other hand, Mozilla has its developers working hand in hand with members of the open source community to develop and produce Firefox OS.

[4.7 Installation and Testing on a phone.](#)

After using the simulator to test and modify system functionality I was able to get a Firefox OS compatible device with the help of my supervisor. While I was looking for the phones compatible with the operating system, I realized that there were a number of devices that developers could use for testing and debugging. On their website, Mozilla classifies these devices into three tiers. Devices in the first tier represent a primary target for development. These devices are usually the first to receive bug fixes and feature updates. The second tier represents devices that are generally functional and being used by many developers so they tend to pick up changes secondarily. The final tier of devices represent those that allow for Firefox OS to be built on,

but are not used regularly by the core developers. [22] Taking into consideration the resources available, I settled on using the Samsung Galaxy S2 device. The next step for me was to research on how to transfer the changes I had made to the phone successfully.

Firefox has a tutorial on their web page that assists developers to build for particular devices. I first had to flash the phone and install Android's Ice cream Sandwich as well as a Siyah Kernel. Then run a build for the device. However, running an entire build was becoming challenging so I researched for other alternatives. The best solution I found was to install Cyanogenmod 10.2 and then I flashed a pre-built image that I found online.[23] The image below shows the actual phone running the operating system.



5 Chapter 5: Evaluation

Firefox OS has been released and as compared to other smart phone mobile operating system developers it is performing well. The devices running this operating system are affordable and targeted at emerging markets, providing an opportunity for people to be online at a very affordable price.

Firefox OS is also simple and does not require any expertise or specific tools. HTML5 and JavaScript are the platform tools used when developing for Firefox OS. This makes it easy for developers who have knowledge of web technologies to contribute to the growth of the operating system. The tools required for development are a browser and a text editor so it is very easy to write a few lines of code and if it renders properly in the browser it would display on the operating system. However, developers must integrate the code they write with the API of the system so that the functionality of the phone is fully engaged.

With this project I was able to customize the application layer core. A task which is otherwise difficult with other operating systems. Also other operating systems do not easily allow users to make changes to the system's core but with Firefox OS I was able to change the keyboard layout.

Another thing about this operating system is that it makes use of web activities called mozActivities and these activities allow developers to develop applications that can interact with the main system cores.

This operating system like other operating systems has many API's that ensure full phone functionality. I explored some of them but the one that stands out for me is how local storage is implemented. The size of local storage associated with this operating system is between 5MB – 10MB. With local storage there are six methods that developers need to be familiar with. The methods are: `getItem(<key>)`- which returns back the value for the key, the second is `setItem(<key>,<value>)`: Which sets a value for a key. The third is `removeItem(<key>)`- This deletes the entry for the particular key. The fourth is `clear()`- It deletes the entries in the Local Storage. The fifth is `length` which returns a count of the number of keys in the local storage. The Final method is `key(<Index>)` which returns the value for a key as per the index.

6 Chapter 6: Conclusion

My motivation for undertaking this project is that I believe that in the near future, Firefox OS would be the preferred operating system for smartphone manufacturers. I was able to explore and document the structure of the operating system. Customization of the operating system is important for users and service provider. This project has proven that it would be easy for a developer to customize the operating system.

Also, I learnt how to develop applications that run on this operating system. I am strongly convinced that anyone who hopes to undertake a project related to Firefox OS would be able to use this document as a first hand guide.

There is more room for research to be done with respect to this operating system. It is new and was released last year. Throughout my research I did not find a scholarly article or scholarly research related to the project. If research is done on this operating system, it would serve as a reference for anyone who does some work on Mozilla's Firefox OS.

I would suggest that Universities adopt the study of this operating system as a foundational course for programming or operating systems. From my exploration and exposure to Firefox OS I realized that the ease of deployment, the structure of the operating system and the language that is used for development would make it simple to explain to students how operating systems work and how applications that run on such systems are developed

Bibliography

[1] Seth Rosenblatt. (2013, August) CNET Mobile [Online].

"<http://www.cnet.com/news/first-batch-of-firefox-phones-sell-out-in-u-s-u-k/>"

<http://www.cnet.com/news/first-batch-of-firefox-phones-sell-out-in-u-s-u-k/>

[2] John Heggestuen (2013, December) Business Insider [Online].

"<http://www.businessinsider.com/smartphone-and-tablet-penetration-2013-10>"

<http://www.businessinsider.com/smartphone-and-tablet-penetration-2013-10>

[3] Cameron Graham (2013, August) Technology Advice [Online].

"<http://www.technologyadvice.com/small-business/blog/changes-in-smartphone-marketshare/>"

<http://www.technologyadvice.com/small-business/blog/changes-in-smartphone-marketshare/>

[4] Jan Varga (2013, April) Vargajan [Online].

"<https://vargajan.wordpress.com/2013/04/08/firefox-os-the-motivations-and-the-goals/>"

<https://vargajan.wordpress.com/2013/04/08/firefox-os-the-motivations-and-the-goals/>

[5] Aditya Dey (2013, July) TechStake [Online].

"<http://www.techstake.org/2013/07/telefonica-launches-first-firefox-os-phones-in-spain.html>"

<http://www.techstake.org/2013/07/telefonica-launches-first-firefox-os-phones-in-spain.html>

[6] Mozilla Developer Network (n.d) Mozilla Developer Network [Online].

"https://developer.mozilla.org/en-US/Firefox_OS"

https://developer.mozilla.org/en-US/Firefox_OS

[7] Mozilla Developer Network (n.d) Mozilla Developer Network [Online].

"https://developer.mozilla.org/en-US/Firefox_OS/Platform/Architecture"

https://developer.mozilla.org/en-US/Firefox_OS/Platform/Architecture

[8] Michael Millar (2013, May) BBC News [Online].

"<http://www.bbc.com/news/business-22390444>"

<http://www.bbc.com/news/business-22390444>

[9] Nmachi Jidenma (2013, November) CNN Opinion [Online].

"<http://www.cnn.com/2013/11/07/opinion/real-mobile-revolution-africa-smartphone/>"

<http://www.cnn.com/2013/11/07/opinion/real-mobile-revolution-africa-smartphone/>

- [10] Angela Moscaritolo (2013, August) Pc Mag [Online].
"http://www.pcmag.com/article2/0,2817,2422667,00.asp"
<http://www.pcmag.com/article2/0,2817,2422667,00.asp>
- [11] SkyLight Project (n.d) Skylight Project [Online].
"http://www.skylightproject.co.uk/pages/javascript-os-392.html"
<http://www.skylightproject.co.uk/pages/javascript-os-392.html>
- [12] Google Official Blog (2009, November) Google Official Blog [Online].
"http://googleblog.blogspot.com/2009/11/releasing-chromium-os-open-source.html"
<http://googleblog.blogspot.com/2009/11/releasing-chromium-os-open-source.html>
- [13] Ted Samson (2013, May) InfoWorld Web site [Online].
"http://www.infoworld.com/t/application-development/google-entices-chrome-os-developers-prospect-of-native-apps-218773"
<http://www.infoworld.com/t/application-development/google-entices-chrome-os-developers-prospect-of-native-apps-218773>
- [14] Jason Evangelho (2013, December) Forbes [Online].
"http://www.forbes.com/sites/jasonevangelho/2013/12/29/heres-why-microsoft-is-worried-about-google-chromebooks/"
<http://www.forbes.com/sites/jasonevangelho/2013/12/29/heres-why-microsoft-is-worried-about-google-chromebooks/>

- [15] Mozilla Web site (n.d) Mozilla Web site [Online].
"http://www.mozilla.org/en-US/firefox/os/"
<http://www.mozilla.org/en-US/firefox/os/>
- [16] Joshua Sherman (2014, February) Digital trends [Online].
"http://www.digitaltrends.com/mobile/android-way-cheaper-than-ios/#!DgSz6"
<http://www.digitaltrends.com/mobile/android-way-cheaper-than-ios/#!DgSz6>
- [17] Media College (n.d) Media College [Online].
"http://www.mediacollege.com/internet/html/why-learn.html"
<http://www.mediacollege.com/internet/html/why-learn.html>
- [18] Mashable (2012, November) Mashable [Online].
"http://mashable.com/2012/11/01/google-apps-tie-apple/"
<http://mashable.com/2012/11/01/google-apps-tie-apple/>
- [19] Liam Tung (2013, November) ZDNET [Online].
"http://www.zdnet.com/mozilla-chalks-up-thousands-of-firefox-os-apps-7000023422/"
<http://www.zdnet.com/mozilla-chalks-up-thousands-of-firefox-os-apps-7000023422/>

[20] Verge Staff (2013, November) THE VERGE [Online].
"http://www.theverge.com/2011/11/16/2565102/smartphone-buyers-guide"
<http://www.theverge.com/2011/11/16/2565102/smartphone-buyers-guide>

[21] Seth Rosenblatt (2014, February) CNET [Online].
"http://www.cnet.com/news/what-google-really-means-when-it-calls-android-open/"
<http://www.cnet.com/news/what-google-really-means-when-it-calls-android-open/>

[22] Mozilla Developer Network (n.d) Mozilla Developer Network [Online].
"https://developer.mozilla.org/en-US/Firefox_OS/Firefox_OS_build_prerequisites"
[https://developer.mozilla.org/en-US/Firefox OS/Firefox OS build prerequisites](https://developer.mozilla.org/en-US/Firefox_OS/Firefox_OS_build_prerequisites)

[23] XDA Developers (n.d) XDA Developers [Online].
"http://forum.xda-developers.com/galaxy-s2/development-derivatives/rom-firefoxos-dualboot-cyanogenmod-10-2-t2612940"
<http://forum.xda-developers.com/galaxy-s2/development-derivatives/rom-firefoxos-dualboot-cyanogenmod-10-2-t2612940>

Appendices

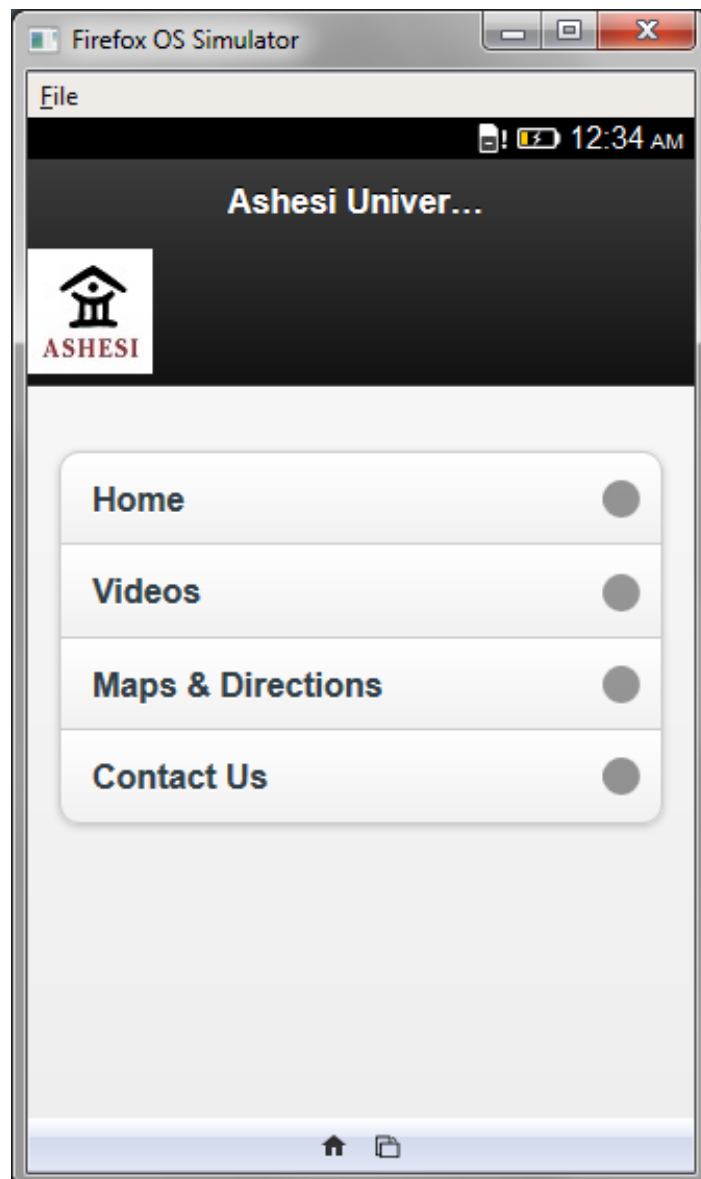
Appendix A

The image below shows the home screen that loads when the simulator is loaded. Like other operating systems it has a dock were apps are saved.



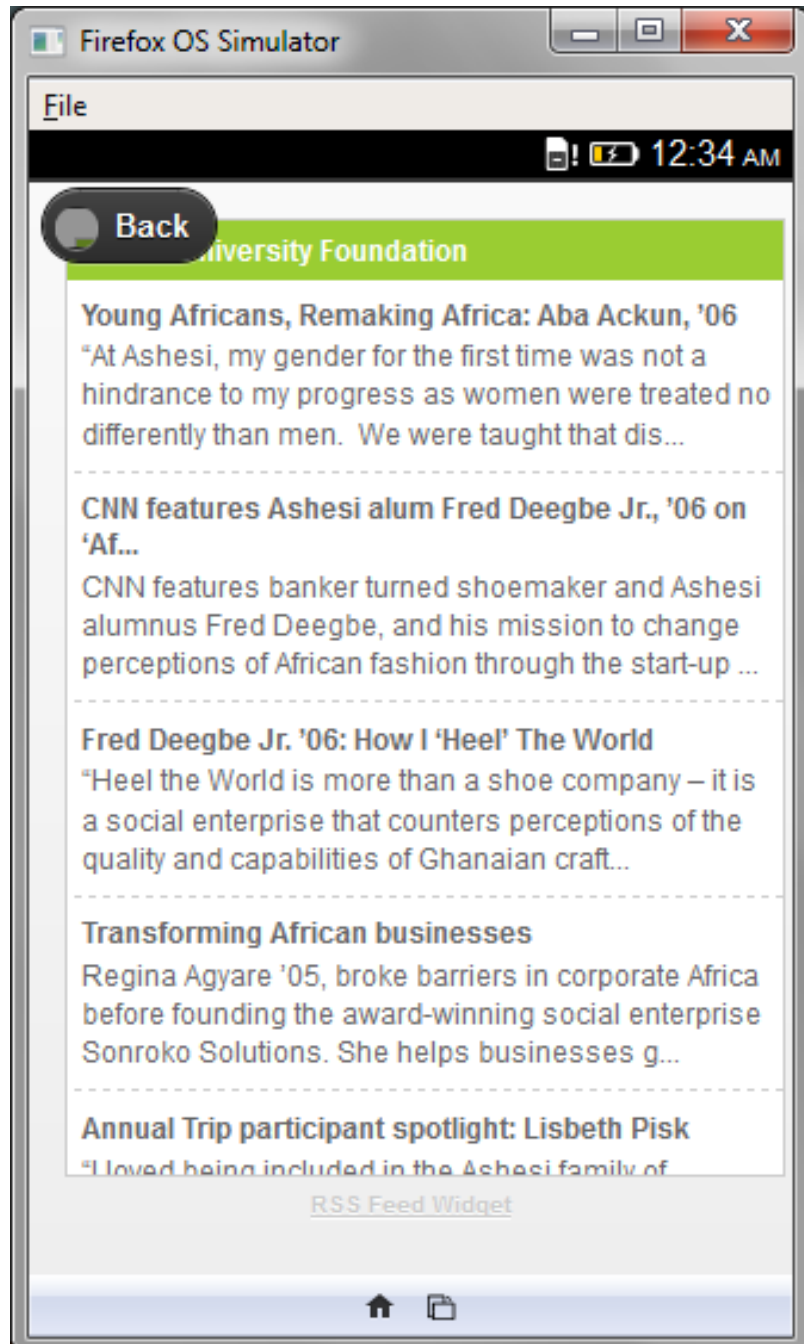
Appendix B

This image represents the Ashesi University application that I wrote to run on the Firefox Operating System. The Home tab loads RSS feeds on news related to Ashesi. The videos tab loads videos from Ashesi's YouTube channel. With the help of google maps, the map and directions tab helps a user find his way from wherever they are to the school. The contact us tab allows a user to activate the dialer of the phone to make phone calls.



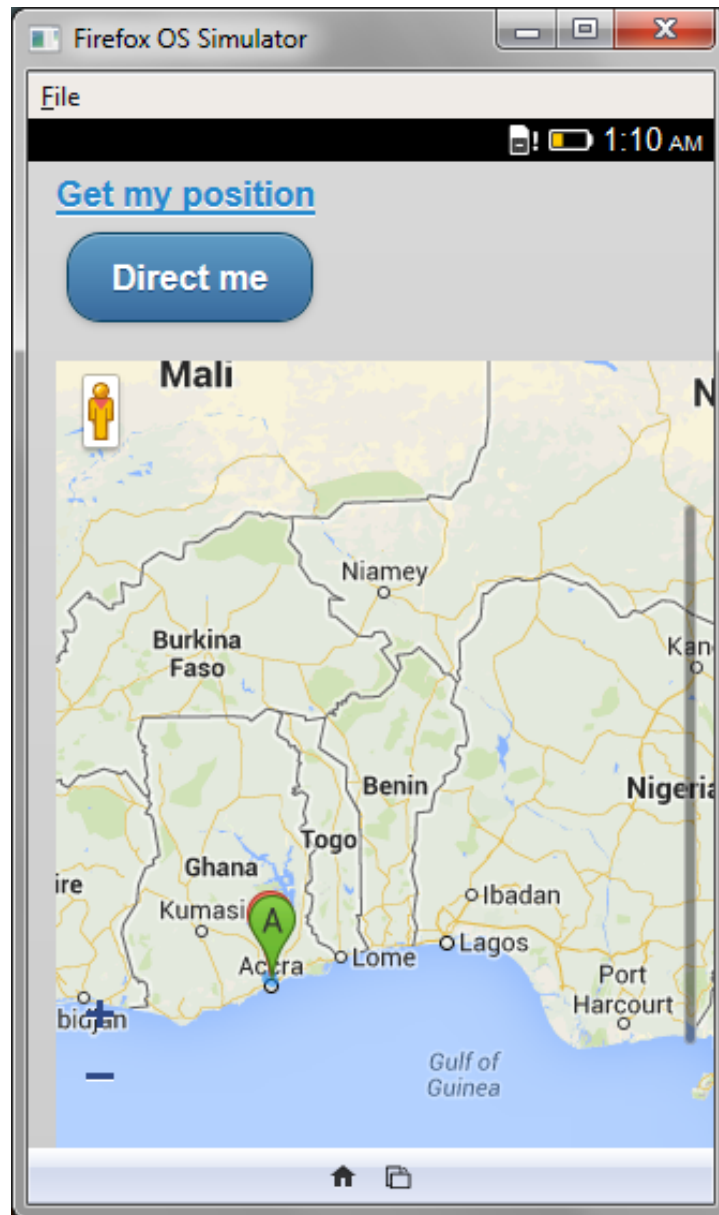
Appendix C

This image represent the RSS feeds that are loaded from the Website of Ashesi Foundation.



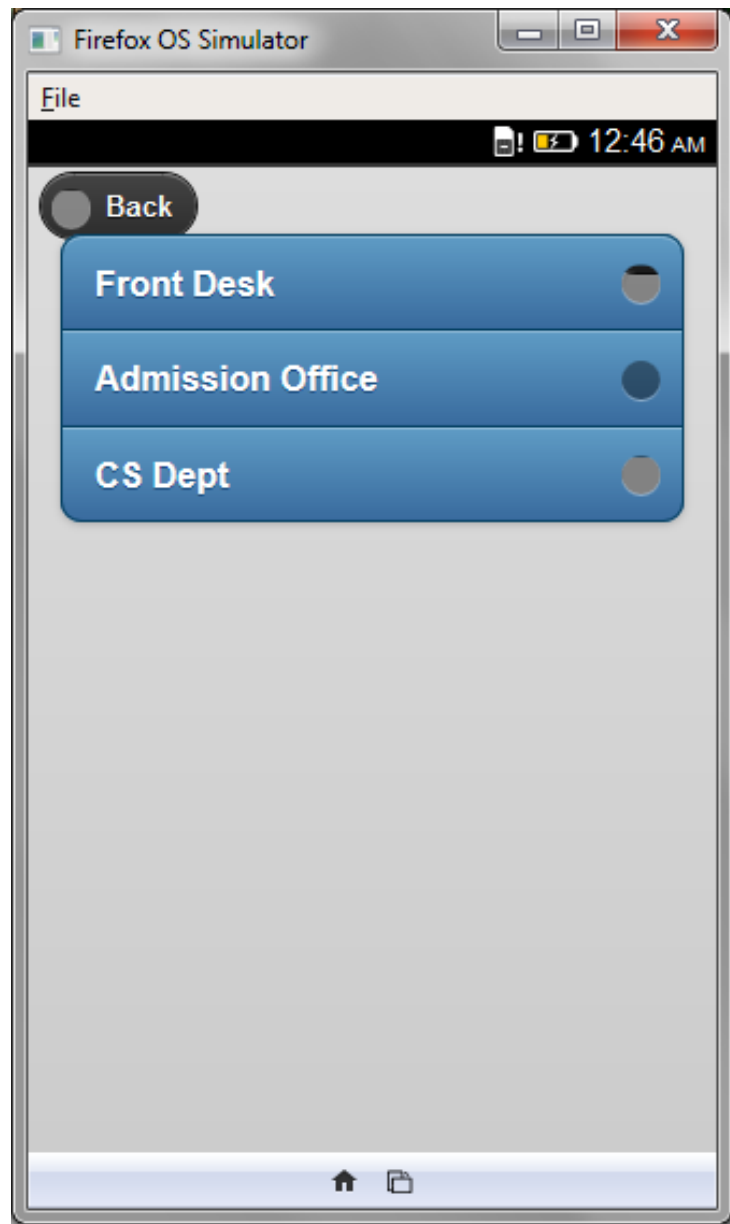
Appendix D

The map and directions tab of the application is shown below. On this page the user enters his location or can click Get my position to get his current position and then select Ashesi University to get directions from his location to Ashesi



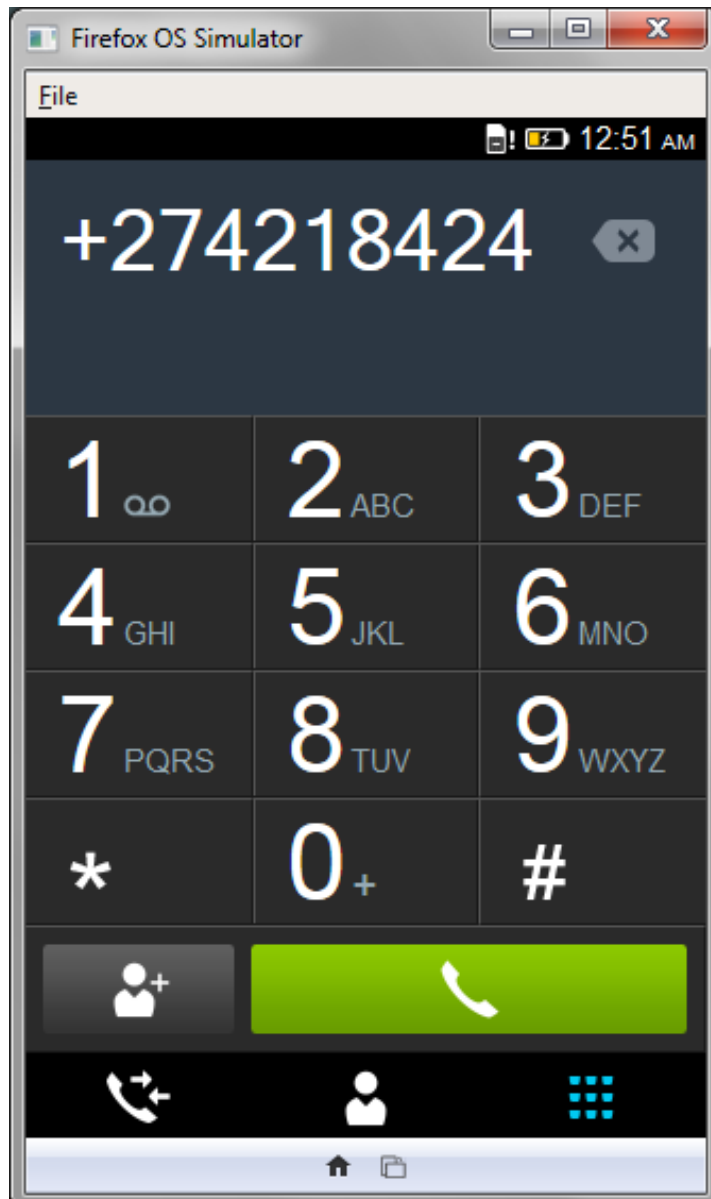
AppendixE

This page allows the application to interact with the phone's dialer app. When the user hits a particular button it would show a dedicated number to the screen that allows you to place a phone call.



Appendix F

This image shows exactly what happens when the user hits the front desk button. It sends the number to the dialer allowing the user to place a call immediately.



Appendix G

The image shows the key options available when a user long presses the letter E. Demonstrating how the keyboard modification works.

