Low Power Computer for Schools in Rural Developing Communities
2nd Year Group Project Report Group 40

Project Manager: Yuchen Wang
Members: Abdulaziz Azman, Ikenna Okoli
Matthew Douthwaite, Michael Chung
Taoda Hong, Xu Zhang

Table of Contents

1 Abstract

This report investigates the feasibility of implementing a low power, low cost computing device in rural Africa for educational purposes. We propose a solution based around the Raspberry Pi following extensive research into the possible hardware and software components. This report also details the prototype built based on this proposal, which includes preloaded applications written by both third party developers and by our group that is relevant to the Rwandan curriculum.

1.1 Background

E.quinox is a student-led organisation aiming to find the blueprint solution to rural electrification; for which one of the main ideas currently being explored is the Energy Kiosk concept. An energy kiosk has solar panels mounted on its roof and uses the energy harvested to charge battery boxes. These boxes are then rented to local people for a small fee so that they can have access to electricity.

Last summer, in partnership with Dartmouth Humanitarian Engineering, e.quinox's first hydroelectric kiosk was implemented in the district of Rugote, south-west Rwanda. The $40,000 USD project currently uses a 1 kW stream engine, and there are plans to expand the capacity of the system in the summer of 2013 to produce a maximum output of 5.5kW [1] .

Hoping to make a wider impact, our solution will be designed around the latest battery box. This generation was manufactured in Rwanda with a 12V 7Ah battery, two 12V DC jacks and a 5V 500mA USB output (Figure 1).

Figure 1 e.quinox Battery Box

In January 2013, e.quinox members returned to Rwanda on an information gathering trip. A meeting with the headmaster of Rugaragara primary school took place during this trip to help us gather more specific requirements which we can feed back into our product. The headmaster specifically requested an office application suite to help prepare older students for secondary school and office jobs. He would also like applications to help the students with English and Maths.

1.2 Design Specification

This project aims to design a low power computer along with exemplar software that would serve as educational aid for schools in rural Rwanda.

The remoteness and temperate climate of rural Rwanda requires the computer to be as robust and low-maintenance as possible. A lack of rural electrification denies many people access to an electrical power grid. Hence given the scarce power supplies, the computer needs to be power efficient. A low power central processing unit (CPU) should be used along with a very basic operating system (OS) and power-efficient hardware peripherals to satisfy this constraint.

Much of the existing teaching aids and learning avenues for schools in rural communities are limited to classes and textbooks. Educational software is to be developed for adults and children in the community, with a view to create an interactive and enjoyable way of learning.

1.3 Project aims

- Research into different approaches to design and build a low power computer prototype that costs below £100, and can run on the latest e.quinox battery box for at least 6 hours.
- Implement a demonstrator for proof of concept and testing.
- Provide an initial exposure of computers and technology to school children.
- Demonstrate simple computer programs to teachers and students in rural schools.
- Improve productivity of teachers and provide teaching aids.

2 Proposed Solution

Our proposed solution is based around a Raspberry Pi Model B. An 8” TFT LCD supplied by BBOXX, a USB mouse and keyboard is also included. The Raspberry Pi will be loaded with Raspbian OS, LibreOffice, Google Chrome, VLC media player and two demonstration games which our group developed using Python and Pygame.

3 Hardware Overview

In this section, we discuss pre-existing solutions and possible hardware that our group has considered.

3.1 One Laptop Per Child (OLPC)

OLPC aims to provide each child a laptop that is rugged, low-cost and low-powered[2]. Their flagship model, xo-1.75, is a fully functional water and dust proofed laptop using Linux, with a 7.5” rotatable screen, wireless connectivity and camera. The device costs $165 USD (£110) when ordered in excess of 100,000 units, which is well over our budget. However, after reading the datasheet, we think the specification of xo-1.75 is too high for our purposes. For example, our solution will not need a camera, microphone input or in-built batteries. But we think it will be beneficial if we use this specification as a benchmark for other solutions.

3.2 ARM-based boards

As electronic components are getting cheaper and smaller, there are quite a few different low cost computer modules on the market. We have chosen a few that are suited to our needs and compared the hardware and software in these products. Specification of each device is compared in Table 1.

Raspberry Pi was given a lot of media coverage in the past year. The Model B version which our group is using is a £25 ARM development board that is capable of booting Linux. It is sized similar of a credit card and can be plugged into any devices with RCA/HDMI input. Beaglebone is a similar device and is the entry edition of the Beagleboard range. It supports similar specification but it is more than double the cost and uses significantly more power.

The other two devices we compared could only run Android. With around 70% market share in the smartphone OS market[3], this attracts many developers to design applications and improve the platform. The APC 8750 has an 800MHz processor and 2.5GB of RAM, which should ensure applications run smoothly. However, the power consumption is the highest out of all the devices we compared, and its 9V input may present technical challenges when implemented with existing e.quinox battery boxes. Gooseberry, on the other hand, has a faster CPU and supports Wifi connectivity, which can be beneficial in a long term, however it costs almost twice as much as the Raspberry Pi.

We decided to use Raspberry Pi as the basis for our project because of its low power consumption and price. Despite a faster CPU, we felt that BeagleBone was too expensive for our application, and APC 8750 will require a redesign of our battery box because of the 9V input. We see Gooseberry is a good alternative but over our requirements; hence the price difference cannot be justified at this stage.

Table 1. Comparison of different hardware alternatives

	Primary supported OS	Storage	CPU	RAM	Output	Power	Connectivity	Cost (£)
XO - 1.75 (One Laptop Per Child) [2]	Fedora (Linux)	4 – 8 GB NAND Flash memory	Marvell Armada 610 with an ARM CPU (800 MHz)	1 GB	3 USB 2.0 ports	DC Jack, 11-18V, 0.83-1.36A, 15W	802.11b/g/s wireless LAN	110 at mass order
Raspberry Pi [4]	Linux	SD/MMC 2GB+	700 MHz Low Power ARM1176JZ-F Applications Processor	512MB	HDMI, Composite RCA, 2xUSB	USB, 5V, 700mA, 3.5W	Ethernet	25
APC 8750 [5]	Android	MicroSD	VIA 800MHz Processor	2.5GB	HDMI VGA, 4x USB	DC Jack, 13.5W, 9V/1.5A	Ethernet	32
BeagleBone [6]	Linux	microSD 4GB+	700-MHz superscalar ARM Cortex™-A8	256MB	USB	USB, 5V, 2A, 10W	Ethernet	58
Gooseberry [7]	Android	Internal 4GB NAND + microSD	A10 1 Ghz	512MB	1xminiUSB, 1x HDMI	AC/USB, 4W, 5V, 800mA	Wireless LAN	40

3.3 Displays

Part of our solution is to implement a desktop setup and the choice of display is an important consideration. It is the component with the greatest power consumption and can also make up a significant proportion of the cost. To choose an appropriate display, we needed to consider the trade-off between power consumption and price. The main technologies considered were cathode ray tube (CRT) monitors and liquid crystal displays (LCD). Plasma and organic light emitting diode (OLED) screens were ruled out due to power consumption and manufacturing costs respectively. [8]

3.3.1 Cathode Ray Tube (CRT) Monitors

As an aging technology, CRT monitors can be purchased for a very low cost over the internet at present, however they are very power hungry [9]. They are also large and heavy and so it is likely that they would be unsuitable for transportation in a country with a less developed infrastructure.

3.3.2 Liquid Crystal Displays (LCDs)

Despite a higher cost, more modern LCD monitors are a lot more energy efficient. There are several types of LCDs on the market and the most common is the thin-film transistor (TFT) LCD, which is widely used in current mobile devices and the general display market.[10]

The TFT LCD is an active matrix that operates by blocking photons from a backlight through different colour and polarisation filters[11]**.** Twisted Nematic (TN) and Super Twisted Nematic (STN) LCDs were also considered. These are both older, passive matrix techniques, which were used in early generation mobile phones[12]. Although these have lower power consumption, these technologies are inherently inferior with limited information content, colour range and viewing angle compared to TFTs.[13]

The latter is a serious drawback as the application of this system is in school classrooms where as many children as possible should be able to see the display. TFT LCDs also suffer from a limited viewing angle, but it is much greater (70° or more), and its other advantages make it a suitable choice.[14]

Another way of reducing both power consumption and cost, is to reduce the size of the screen. Approaching the problem from the perspective of someone who has never owned or even seen a computer before, we feel that it is acceptable to sacrifice certain features that a user in a developed country would take for granted.

For these reasons, we have opted for a smaller screen than is normal for a desktop computer. There were several options considered. We searched through a number of suppliers, and considered a number of options including a 7.0” integrated TFT module supplied by RS Components.[15] This was ruled out however as it uses an 8080 serial input connection. The Raspberry Pi does have a display serial interface, but the software driver for it to run with the graphics processing unit has not been created yet.

An alternative is a TFT LCD screen which is supplied by BBOXX, a spin-off company of e.quinox.
These LCD screens are designed to be powered by 12V batteries and used as TVs along with BBOXX products in Africa. They have an RCA connection and so can be used as a display for our system. The screen has an 8.0” diagonal display size, a resolution of 800 x 600 and is rated at 12V 400mA. The power connection is a 12V DC jack which will connect to the e.quinox battery box which we intend to use. These TVs would be well suited to our design as this is their intended setup, and at 5W, they should allow an acceptable battery life for the whole system. The screen cost $60 USD (~£40), but this cost could be reduced in bulk.

3.3.3 Charitable donations and second-hand screens

It is possible that large companies may be willing to charitably donate LCD screens when they replace their office stock. This option would become more feasible if this project were to grow to a larger scale and gain more credibility. Companies would also want to see a thorough recycling plan, to ensure no damage to their reputation. Another potential source of cheap or free screens is internet sources such as eBay or recycling websites. However, the quality and the power consumption of these screens cannot be guaranteed in comparison to the BBOXX screen mentioned above.

3.3.4 Reducing the power consumption of TFT LCDs

Although they generally consume less power than CRTs, TFT LCDs are actually quite inefficient.[10] Choi [16] documents a number of techniques to reduce the power consumption of these screens. Some are easily implemented, such as choosing the colours you display carefully. Black is the most power intensive as the most light must be blocked, whereas white is the least. Other techniques require more complicated operations. Choi shows that the refresh rate of the display can be reduced to 33% before a minor flicker becomes observable, saving 346.2mW (for a 6.4" TFT LCD)[16]. Backlight brightness can also be reduced, although changes must be made to prevent image degradation.[16] These techniques could all be applied to our system to optimise it for use with a battery.

3.4 Power Testing

In order to determine the battery life while running the system, we carried out measurements under different operational conditions. A multimeter and the e.quinox battery box printed circuit board (PCB) are connected in series to a power supply. The multimeter is used to measure the current draw and the voltage supplied, while the PCB provides protection for the battery in the form of a low voltage disconnect and provides a stable 12V and 5V output for the system (Figure 2). We recorded values during booting, an idle state and running applications. The data is shown in Table 2.

While the system needs to be tested for continuous use over a period of several hours, the observed current during this testing never rose above 750mA. Therefore the maximum power we could draw with the battery charged at 13V (the expected voltage after the initial charging voltage has settled [17]) is 9.75W (13V*0.75A). A fully charged 12V 7Ah battery stores up to 84 Watt-hours of energy (12V*7Ah). This means it can supply 84W for one hour, or power our system for 8.62 hours (84Wh/9.75W).

Figure 2 Prototype and Test Setup

Assuming a peak current draw of 750mA, we obtain a projected battery life of almost 9 hours on a single charge, which more than satisfies the project aims. It should be noted that battery capacity can vary, so the actual battery life could be slightly different.

Table 2. Experimental power data

Components connected/ System State	Voltage (V)	Current (mA)	Power (W)
Battery Box circuit only	14.38	24.54	0.353
Battery Box circuit connected to Raspberry Pi / Booting up	14.35	210	3.014
Battery Box circuit connected to Raspberry Pi / Idle	14.31	180	2.576
Battery Box circuit connect to BBOXX screen	14.21	422	5.997
Battery Circuit, Raspberry Pi, screen / Booting up	14.18	600	8.508
Battery Circuit, Raspberry Pi, screen / Idle	14.35	575	8.154
Full system with mouse and keyboard / First plugged in.	14.2	600	8.520
Full system, with mouse still	14.2	595	8.449
Full system, moving the mouse	14.2	616	8.747
Full system, opening python.	14.2	635 (max)	9.230
Full system, maths game opened/ Booting up	14.2	650	9.230
Full system, maths game open/ idle	14.2	645	9.159

4 Software Overview

As an open source project, the Raspberry Pi can run different versions of Linux operating systems. This also means a wide range of software is available online. Here we evaluated some of these options we think is essential for a successful implementation.

4.1 Operating System (OS)

The ARM1176JZ-F processor used in the Raspberry Pi only supports ARM instruction sets, so we immediately had to rule out Windows and Mac OS. It is presently not possible to obtain copies which allow these OSs to be run on either non-Apple or ARM hardware. We decided that we will use Linux-based OSs with the Raspberry Pi, as it is more likely that students will encounter Linux than any of the other compatible OSs in the future. With that in mind, we drew up a shortlist of Linux OSs that we feel appropriate for the project:

Raspbian [18]	An operating system based on one of the most popular Linux distributions, the Debian distribution. This is also the official distribution supported by the Raspberry Pi Foundation. Recommended as one of the most beginner friendly distributions by the online forums.
Android OS [19]	An operating system designed for mobile devices. This was considered because it supports keyboard and mouse inputs, and has the biggest app ecosystem out of all the Linux distributions making this one of the most versatile OSs.
Arch Linux (ARM) [20]	Another Linux distribution, but one focuses on simplicity for the developer. The main advantage is that it can be customised to be more system resource friendlier than the other OSs.

We were at first in favour of Android OS due to the high number of applications available for download and it is perceived to be one of the easier environments to develop applications for. However, the lack of a publicly available graphics driver made the responsiveness of the system very slow. We trialled the OS on the Raspberry Pi and there are noticeable long delays between user's keystrokes and the letters appearing on screen. As we aim to implement the solution on the ground in summer 2013, we do not see this as usable; though as developers overcome the issues in the future, this decision may be revised. Also, due to the lack of internet connection where we are trialling the solution, it will not be possible to install any applications in situ.

Arch Linux was then briefly considered because of its high customisability and requiring the lowest computing resources to run.[19] However, from an application development and usability point of view, we found Raspbian to be more user-friendly. With the latest Raspbian release, the Raspberry Pi now supports overclocking without voiding the warranty, which allows the processor to have a higher clock frequency than the factory default, which could potentially reduce the battery life, but gain a slightly faster performance. We were also able to install and build many third party applications such as VLC media player with working audio output and remotely access the device with a desktop client. Recently, the foundation has also set up an App Store exclusively for the device; this will be beneficial when internet is available in the area.

4.2 Educational Software

The Rwandan curriculum for schools was researched on the website of the Ministry of Education [21]. The ministry has devised a wide ranging curriculum. In the English syllabus, a lot of emphasis has been put into learning vocabulary, as well as grammar and reading. [22] The Maths syllabus is similar to that of Cambridge Primary, one of the examination boards in the UK. [23,24]

From our group's own experience, games or interactivity makes the lesson a lot more interesting and hence why it was decided to create games for the Raspberry Pi. The idea of these games also means that students can also make self-learning a lot more interesting. When designing these games, we have to keep in mind that the contents of the games must be easily configurable. This is done so that teachers can create more content themselves, and in the long term, they can create games to better suit their needs.

4.2.1 Game Engine

Due to the limited specifications of Raspberry Pi and our groups' limited experience, 3D games would not be a feasible option, so we decided to look for game engines which made 2D games. The Raspberry Pi website states that any software which can be compiled with ARMv6 will be compatible with the device [3]. That was the direction we took and we found three suitable game engines.

Pygame [25]	One of the more popular game engines, Pygame was suggested online in many forums. Designed primarily for game development, Pygame is an additional library of the programming language Python, which is one that we are already familiar with.
Marshmallow [26]	Marshmallow is a C++ game engine still in development and was originally built for Marshmallow Entertainment System, an open-source project for DIY game console. This game engine supports 8 bit or 16 bit graphics and is used for 2D games. It is compatible with Linux and been tested on a Raspberry Pi.
Slick2D [27]	Slick2D provides tools and libraries for 2D Java game development. Although easy to program, it is likely to have compatibility issues with the device since the language is Java. At the time of research, we have yet to find any evidence of Slick2D being implemented on a Raspberry Pi.

Based on the above analysis, we concluded that Slick2D will be inappropriate for our project. Further research shows the Raspberry Pi Foundation announced Python is the official educational language the device uses [3]. Hence it seems logical that our games use Pygame as the engine instead of experimenting with Marshmallow.

4.2.2 Maths Game

This is a simple drag and drop game designed to help improve a child's basic arithmetic abilities. It involves basic addition, subtraction, multiplication and division with numbers no greater than 16.

The interface is simple; there are two boxes called the "Drag Box" and "Answer Box", and two buttons named "Check" and "Next". The "Drag Box" contains 16 giraffe icons. A question appears at the top of the screen e.g. "what's 3+2?". To answer the question, 5 giraffes are dragged into the "Answer Box". The user can then check the answer by clicking the "Check" button and a text message will appear informing the user whether the answer was correct. If the user chooses to skip the question, he/she can click on the "Next" button and another question will be randomly generated, with all the giraffe icons reset to their initial positions.

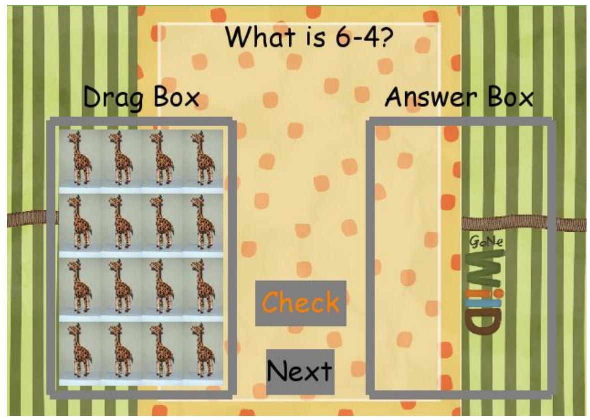

Figure 3 Maths Game

4.2.3 English game

The user interface consists of spaces for questions and answers, a button and a counter for the score. The questions and answers can be entered into the fields indicated in the code. This is then put into a dictionary in Python. By clicking the button, it checks whether the answer is the same as the input and indicates the result, the score is also updated in realtime.

Although this program is primarily developed as an English game, it can be easily adapted as a general quiz game. The user only needs to change the questions and answers for use in other subjects.

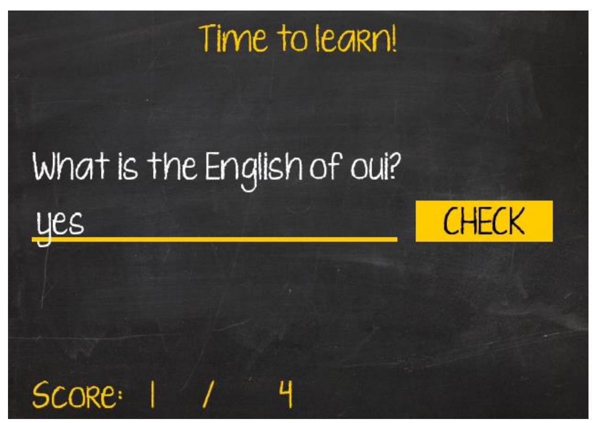

Figure 4 English Game

5 Cost of the solution

With the proposed solution, the estimated cost is shown in Table 3. Since all software used is open source and therefore free, our cost only includes hardware. At the time of writing, the cost of our solution is below £100 as aimed. This cost is only for building a single prototype device, however if the solution is to be done in bulk, the cost can definitely come down further.

Item	Cost
Raspberry Pi [28]	£26.78
Bboxx Display [29]	£38.86
Composite Video Cable [30]	£2.99
Keyboard [31]	£5.00
Mouse [32]	£2.21
Memory Card 8GB [33]	£6.99
Micro USB to USB Charging cable [34]	£4.00
Case for Pi [35]	£4.79
Total	£91.62

Table 3. Cost breakdown of the solution

6 Future Developments

In terms of hardware, the system needs to be tested over longer periods of time. At present, the battery life figures are very much theoretical. The battery boxes are all currently in Rwanda, so it is not possible for the team to do repeated run-down tests to obtain a typical battery life on a single battery charge. Another main challenge presently is making it robust enough to run in a warm, dusty environment. The screen and other peripherals should stand up to this with care, but the electronics of the Raspberry Pi are very much exposed so a case will need to be created before distribution.

In terms of software, the main improvements which could be made to the prototype will need to be established through user trials. The accessibility of the software to a user completely new to computers is an important factor to consider and can only be tested with candidates of the correct level of experience. We envisage the software content to expand and adapt to the needs of the school. The Raspberry Pi has built-in networking abilities, and so it should be possible to have the games updated over the internet from the UK if a mobile internet module is added to the Raspberry Pi. Further applications could be added to cover a range of subjects to suit a wider variety of age groups. Our aspiration is for the teachers or students to use the system to be able to write code. We feel that this would actually be the most useful skill that they cannot teach themselves at the moment, and would greatly increase their employability in a country fast approaching the internet age.

The scope of our research is limited when it comes to long term impacts of the project. For example, we would have to consider the environmental impact and disposal method for electronic equipment.

Furthermore, we need to take into consideration a business model that would allow us to distribute the system; the product needs to be financially sustainable as well as technically feasible. Even if the system is not distributed for profit, money will be needed for maintenance and to replace parts when necessary.

7 Conclusion

It will be necessary to evaluate the product's performance and to determine whether it meets the requirements of our target audience. Our group feels that by July 2013, our prototype will be ready to be trialled in a school in Rwanda.

Overall, our project forms a good foundation for the development of low power computers for schools in rural developing countries. We have shown that a system can be built which encompasses the primary features of a desktop computer; including input devices, a user interface and word processing software. Our research shows that it is theoretically feasible to power our system for at least 6 hours using an e.quinox battery box.

8 References

[1] e.quinox. *Hydro Kiosk Project.* [Online] Available from: http://www.e.quinox.org/index.php/projects/hydro-kiosk [Accessed 4 Dec 2012].
[2] One Laptop Per Child. *Specifications.* [Online] Available from: http://one.laptop.org/about/specs [Accessed 21 Dec 2012].
[3] Strategy Analytics. ***Android and Apple iOS Capture a Record 92 Percent Share of Global Smartphone Shipments in Q4 2012***. ; 2013.
[4] Raspberry Pi. [Online] Available from: http://www.raspberrypi.org/ [Accessed 21 Dec 2012].
[5] APC. *8750.* [Online] Available from: http://apc.io/products/8750a/ [Accessed 21 Dec 2012].
[6] BeagleBoard. *BeagleBone.* [Online] Available from: http://beagleboard.org/bone [Accessed 21 Dec 2012].
[7] Gooseberry. [Online] Available from: http://gooseberry.atspace.co.uk/ [Accessed 21 Dec 2012].
[8] Liu DN. ***Plasma Display Panels***.
[9] Socolof ML, Overly JG, Kincaid LE, Geibig JR. *Desktop Computer Displays: A Life-Cycle Assessment.* Environmental Protection Agency, US. Report number: EPA-774-R-01-004a; 2001.
[10] Berkoff B. ***LCDs, Growth and Market Coverage***.
[11] Flewitt A. J. **Hydrogenated Amorphous Silicon Thin Film Transistors (a Si:H TFTs)**. In: Chen J, Cranton W, Fihn M. (eds.) *Handbook of Visual Displays Technology*; 2012. pp. 627.
[12] Raynes P. ***Twisted Nematic and Supertwisted Nematic LCDs***. ; 2012.
[13] C-More. *C-more STN vs TFT.* [Online] Available from: http://goo.gl/7xxIB [Accessed 2 Feb 2013].
[14] Blankenbach K. Viewing Angle. In: Chen J, Cranton W, Fihn M. (eds.) *Handbook of Visual Displays Technology
* : Springer-Verlag Berlin Heidelberg. pp. 2367.
[15] RS Components. ***TFT LCD 7.0" integrated module 800x480***. [Online] Available from: http://uk.rs-online.com/web/p/lcd-colour-displays/7502588/ [Accessed 30 Nov 2012].
[16] Choi I, Shim Hojun, Chang N. Low-Power Color TFT LCD Display for Hand-Held Embedded Systems. 2002 112.
[17] Yasua. *NP SERIES - NP7-12 Data Sheet.*
[18] Raspbian. [Online] Available from: http://www.raspbian.org/ [Accessed 21 Dec 2012].
[19] Android. [Online] Available from: http://www.android.com/ [Accessed 21 Dec 2013].
[20] Arch Linux ARM. [Online] Available from: http://archlinuxarm.org/ [Accessed 21 Dec 2012].
[21] Ministry of Education, Republic of Rwanda. [Online] Available from: http://www.mineduc.gov.rw/ [Accessed 2 Jan 2013].
[22] Ministry of Education, Republic of Rwanda. *Curriculum Of English Language For Primary Schools in Rwanda Grade 1-6.* ; 2010.
[23] Cambridge Primary. *Cambridge Primary - Mathematics Curriculum Framework.* ; 2011.
[24] Ministry of Education, Republic of Rwanda. *Mathematics Curriculum For Upper Primary Education (P4, P5, P6).* ; 2005.
[25] Pygame. [Online] Available from: http://www.pygame.org/news.html [Accessed 30 Nov 2012].
[26] Gamaral. *Marshmallow Game Engine.* [Online] Available from: http://guillermoamaral.com/marshmallow_h/ [Accessed 30 Nov 2012].
[27] Slick2D. *Open Source 2D Java Game Engine.* [Online] Available from: http://www.slick2d.org/ [Accessed 30 Nov 2012].
[28] Farnell United Kingdom. *Raspberry Pi.* [Online] Available from: http://uk.farnell.com/raspberry-pi [Accessed 25 Feb 2013].
[29] Baker-Brian C. *BBOXX Display Pricing.* Personal communication.
[30] RS Components. ***RCA to RCA video cable,1.5m***. [Online] Available from: http://goo.gl/NGJGm [Accessed 3 Jan 2013].
[31] OneCall Farnell. ***HK2026 USB - KEYBOARD, BASIC USB BLACK, UK***. [Online] Available from: http://uk.farnell.com/jsp/displayProduct.jsp?sku=1848107&CMP=e-2072-00001000&gross_price=true&mckv={ifsearch:s}{ifcontent:c}YKMwxwd8|pcrid|productlistings [Accessed 4 Dec 2012].
[32] Amazon UK. ***DIGIFLEX Small Black USB Wired Optical Light Scroll Wheel Mice Mouse for PC Laptop Computer***. [Online] Available from: http://www.amazon.co.uk/DIGIFLEX-Optical-Scroll-Laptop-Computer/dp/B002MTXL26/ref=tag_stp_s2_edpp_url [Accessed 4 Dec 2012].
[33] Currys. ***VERBATIM Class 4 SDHC Memory Card - 8GB***. [Online] Available from: http://www.currys.co.uk/gbuk/verbatim-class-4-sdhc-memory-card-8gb-10938441-pdt.html [Accessed 4 Dec 2012].
[34] Cables 2 u. ***Micro USB Cable A to Micro B***. [Online] Available from: http://www.cables2u.co.uk/micro-cable-micro-p-709.html?gclid=CLTz_ZuEw7UCFXDKtAodkVwA_Q [Accessed 4 Dec 2012].
[35] RS Components. ***Raspberry Pi Type B Case - Clear***. [Online] Available from: http://raspberrypi.rsdelivers.com/product/rs/raspberry-pi-type-b-case-clear/caseclr.aspx [Accessed 4 Dec 2012].