Implementation of Web Content Management System for a charity group

By
Sambou Jadama

Mälardalen University
Department of Computer Engineering
Rikard Lindell, Federico Ciccozzi

January 18, 2012
Abstract

In the early days of the World Wide Web, web content management laid entirely in the hands of the webmasters. With the evolution of the web technology and demanding business requirements, the responsibility of web contents management has shifted from webmasters to anyone without any level of web programming knowledge.

A charity organization intends to have a website to reach their online audience. They have decided to deploy a Web-based system that will allow the administrative members with basic computer skills to manage the web contents. This thesis addresses the issue by developing Web Content Management System with simple user interfaces where the administrators can manage digital contents. The system should allow end-users to register on the website, upload photos, send emails and make donations.

This work evaluates three online payment systems: SMS payment, Google Checkout and PayPal. Eventually, the method that results to be most suitable for the organization is chosen and implemented for online donations.

During the work, information is gathered by requirement solicitation methods to facilitate the implementation of end-user interfaces. A database is created and a website is built using the ASP.NET technology. The result is a web-based Content Management System where the administrator can edit contents by using a simple text editor and publish them on the web.
Preface:
First of all I want to thank God for giving me the strength and power to complete my studies. Many thanks and sincere gratitude goes to my examiner Rikard Lindell and my supervisor Federico Ciccozzi for their continuous support and flexibility through the whole process of this thesis. I would also like to express my sincere gratitude to all my teachers at the Mälardalen University for their help. This work is dedicated to my late grandmother, AjaNandingDanpha, for her unconditional love and support throughout my life; your prayers and patience I received will always be remembered as the reason I was able to endure. I would also thank my parents and my paternal aunts for their support they gave me. Many thank to my wife, Isatou, my sons Sulayman and Sainey for their unconditional love and support. Many thanks to my brothers, sisters and my cousins and the entire extended for their love and support throughout my life. I would like to express my sincere appreciation to my friends and fellow students for their encouragement and support throughout the course of this thesis effort. Finally I would like to thank all the members of the BadibuKafo for their excellent work in Badibu, The Gambia: Your effort and work will benefit these societies for long time to come. I thank you all for giving me the opportunity for contributing to your great work. I would also like to thank the Africa Group website for using their images.
6.5.2 Implementing the Business Logic Layer

6.6 Result

6.7 Testing

7 Summary and conclusions

8 Future works

9 References

10 Appendix

Appendix 1
Appendix 2
Appendix 3
Appendix 4
Appendix 5
Appendix 6
Appendix 6
Appendix 8
Appendix 9
Appendix 10

30
32
33
34
34
35
36
36
36
37
37
37
38
39
40
40
41
42
1 Introduction

In today's modern world organizations that want to reach global audiences uses the power of internet technology to achieve their goal. An emerging non-profit organization also intends to reach their online audiences as well as potentials donors across the world to promote their course. The organization aims to increase its revenue by appealing to their online audiences. For the sake of simplicity and because of financial and technical constraints, the organization looks for a web solution that does not require programming knowledge for managing its contents. This report describes the work to develop the website for a charity organization.

1.1 Introducing the charity organization

This work is intended for the non-profit organization and their end-users. The organization, named BadibunkaKaffo is a registered charity organization that has one of its bases in Sweden. Its main purpose is to raise money in order to assist those in need of emergency aid in Badibu, North Bank Region in The Gambia [0]. The organization has also sister charity groups in France and Spain where they are responsible for running and raising their own revenues, though all sharing the same goal of helping less fortunate people in the aforementioned region.

About three year before this project started, I became interested in the organization and became a member. Over the years, I became impressed with enthusiasm and the dedication these members spend helping those people in my native country. The idea of the project then came during the discussion about the important and convenience of having a website for the organization. A website that would inform its members on their activities, and even possibly to enable them to pay their monthly contribution on the website. Although, the organization had earlier had thoughts about outsourcing the development of a simple website but, because of financial and technical constraints the idea was never realized. Few years later, at the end of my studies I got the idea of me developing a website as my bachelor thesis. A proposition for the project was formed and presented to the organization and to my university. The proposition was accepted and it was decided that the development of the website would take place in close collaboration with the owner.

2 Purpose

The main purpose of this work is the development of a WCMS for the administrators to be able to manage the web contents without necessarily being knowledgeable in web programming. The website should represent the organization and convey its message to the current members as well as to potential donors. Members should be able to contribute with donations and the administrators to manage all online donations. Registered users are allowed to perform several different actions on their registered profile (e.g. manage donation records, manage photo albums and personal information).

2.1 Delimitation

The website will be temporarily hosted by the Mälardalen’s student server (www3.idt.mdh.se/mdh/sja06001) to allow end users and the stakeholders to follow the development process online. However, the student server has some limitations which are beyond the developer’s control. Since the server consists of only one shared database for all the students, students are not allowed to create other databases on it, but rather create new tables and add them to the shared database. Once the student is logged on the server, he or she has full access to all tables in the database. Due these reasons no valuable data would be stored on the temporal website. In any case, such issues do not appear on the local development server.
Once the website is completed and the stakeholders satisfied with it, the process of hosting it on a permanent commercial server will begin, but process will not be covered in this work.

3 Background
In the early days of World Wide Web (WWW), websites were largely consisting of static HTML code. In those days, both the responsibility of web development and web content management laid entirely in the hands of the webmasters. Content writers were completely dependent on webmasters in order to deploy contents on the web pages [3]. Nowadays, a very suitable solution for such problem is the web application that separate presentation from content and also enable the user to deploy and interact with text, image and other data on a web browser. In his article [18] "Content Management and the Separation of Presentation and Content", Clark discusses the importance of separating design (HTML and CSS combined) from content as a baseline assumption of web publishing. A recent technology development, a WCMS offers a valuable solution to these problems and many more.

Content Management (CM) topics often cover a wide range of areas within digital technology. In fact, Miller states that CM encompasses a broad spectrum of areas including WCMS. Other areas also include document management (DM), knowledge management (KM), electronic content management (ECM), and financial content management (FCM) [19 - ]. This work specifically covers the web aspect CM.

Miller defines CM as a “process of collecting, organizing, categorizing, and structuring informational resources of any type and format so that they can be saved, retrieved, published, update, and repurposed or reused in any way desirable”, [19].

WCMS practices and goals vary from organizational to organizational. Cooperate business, non-profit organization, e-commerce websites, educational institutions and many medium-size to large-size organizations use WCMS, but in different ways. This leads to differences in terminology and in the names and number of steps in the process. For this reason we will provide different definition from various research papers.

Liduo and Yan define WCMS as CMS implemented as a web-based application designed for creating and managing HTML and also enabling non-technical users, with friendly user interfaces to create, edit, manage and control a dynamic web material [1].

Clark defines WCMS in [18] as CMS devoted software specifically designed to provide tools for creation, presentation, and maintenance of website content includes everything from simple blogging website to full WCMS features containing user-customizable portal facilities.

In case studies on Content management in [2] Grabill et al. define WCMS as a software tool which facilitates distribution of writing practices across the organization and also make it possible for the actual content writers to be directly responsible for editing and publishing contents on the web pages without relying on a webmaster for such services.
3.1 Evolving from Markup to Structured Content

The very first websites were simply static files that contained HTML. HTML is a markup language that is based on a set of markup tags that are used to describe HTML documents in web browsers [17]. The browser does not display the HTML tags, but uses them to interpret the content of the page. Tags are simple keywords within angle brackets like <html>, and they usually come in pairs, like <html>, as start tag, and </html>, as end tag. In [18] Clark identifies the problems web developers encountered when using HTML. This method required designers to meld presentation and content through the use of tags such as font-, color-, and layout-specific. Consequently, the process of updating and making changes to these static files became very difficult and time consuming particularly on a larger website. Consequently, these designers sought to create a clear separation that would free HTML-tagged content from a particular page structure and visual style. As a solution to the problem with the decentralized tagging, the W3C developed Cascading Style Sheets (CSS). With CSS, the style information need not appear in the same place, or even the same document. With CSS the process of styling or updating became less time consuming. The biggest contribution of CSS was that it enabled developers to change the appearance and layout of any amount of pages in a web site, only by editing a single CSS file. Nevertheless non-technical users were still depending on the webmasters for managing and publishing content on the web page(s) [3].

By combining HTML design and CSS layout of the web page and requesting data content from the database users can create pages by merging the HTML and data based on each individual page request. Using a simple text editors, similar to those found in common word processing software such as WYSIWYG (“What You See Is What You Get”), non-technical users can write content and then submit the content in a database, without regard to HTML and CSS. More importantly, a WCMS allows the non-technical users to edit the content of a single page with a simple text editor, removing the need of using HTML and CSS [18]. McKeever discusses in [3] the evolution from static HTML and CSS to WCMSs.

3.1.1 The choice of Web Content Management System

WCMSs are becoming more common for organizations due to business requirement and the need to reduce the dependency on webmasters for content management. A quick research shows that there are many available WCMS in the market ready to be deployed. Choosing the right WCMS that meets a particular organization’s need can be a long process, instead an assumption has been done that the most effective way in form of required functionality and money is to build the software. However, the goal is to take use of the existing techniques used in WCMS and select those techniques that are suitable for our system. Gupta et al. state in [20] that there is no industry consensus on standard CM features and single tool that resolves all CM issues. Choosing the best solution requires a clear understanding of corporate needs, which can be a daunting task. Nonetheless, a thoroughly implemented WCMS produces several benefits, i.e. cost-effectiveness and efficiency, for the organization [1][4]. However, one of the disadvantages of WCMS can be the cost of implementation, especially for low budget organizations. But once the system is setup, the overall cost of not having to hire full-time developers can lower the total costs.
3.1.2 Separation of Presentation and Content

WCMS allows content writing to be distributed across the organization by granting writing access to more individuals within the organization. The primary task of WCMS is the separation of presentation from content. Grabill et al. contend that separation of content management from the technical management of the website enables the web technicians to concentrate on improving the website while the content management team focuses on improving the quality of the contents [2].

In a case study described in [2] Grabill et al. review two example cases where they specifically looked at: (1) how contents are written and published on a website prior to introducing a CMS, and (2) how writers negotiated writing after the introduction of the CMS. Their article is focused on findings from 3-years workplace study project designed to understand the effects of the introduction of a CMS on the writing. In the first case, authors submit document to the webmaster who publishes them on the website. Their study identifies the problem: of the content publication becoming a bottleneck because the person approving and the one publishing were not the same. The study also reveals that the cost associated with making changes to the website charged by the webmaster made a dynamic site challenging both procedurally and financially.

In the other case study, after introducing the CMS, most of these problems were overcome but new ones introduced. With the CMS, the workflow was greatly simplified, and could be completed entirely within the CMS. Furthermore, all documents are stored in a single database which also facilitates the reuse of such documents. The CMS eliminated the reliant on the webmaster by allowing anyone with appropriate administrative right to update or edit pages on the Website. Grabill et al. concluded that CMS reduced the workload of the webmaster responsible for making all web updates.

Reducing the dependency on webmasters can generate the resources needed to accommodate more employees with the skills needed to manage the website and allowing technical professional to concentrate on improving the web page. In a similar research, Hallikainen et al. identify a third key issue, namely making the content creation and publication directly available to content writers with posting and editing access [2].

3.1.3 WCMS lifecycle

In order to understand the concept of web content management, it is also important to be aware of its lifecycle process. Content creation, review, approval and publication are performed in a continuous and iterative process that most often involves one or more people from the administrative team [3]. We categorize this process in two iterative phases:

- creation and collection of content
- delivery or publication on the web

The first phase is the creation and collecting of contents that can be assigned to a group of people (referred to as editors) within the management team. When the content is created it may be stored in the organization's database or repository. The editors may be non-technical users and therefore require highly usable user interfaces to accomplish their task.

The second phase is publishing the contents on the website. This task may be assigned to one member of the administrative team whose responsibility is to approve the content created by
the editors before publishing it online. This process continues along with the website’s lifetime.

To satisfy the need of a website that is fully maintained without any programmer or web designer inside the organization, much effort has been laid on developing the functions and design that satisfied the basic requirements of WCMS.

3.2 Online Payment
This section introduces the different payment technologies and briefly compares them.

Online donations have increasingly become an important source of revenue for non-profit organizations [8]. Therefore, most of current charity organizations provide multiple online payment options for their donors. Nevertheless, due to resource constraints, small organizations may not be able to provide the complete set of payment options to their members and donors. Therefore, an additional task of this work is the selection and implementation of the payment option that best meet the organization’s needs. Three payment options are evaluated and the selected one implemented.

According to Target Analytics [8], a company that works with non-profits organizations to maximize their profit, states that online donors contribute with larger sums than traditional mail donors and those non-profit organizations get about 10% of their income from online donations.

Online Payment or Electronic commerce (EC) is described as doing business online because it facilitates the acceptance of electronic payment for online transactions [9]. This technology allows the payment for goods, services and to make donations on the web. EC has become increasingly popular due to the wide spread use of the WWW. It also helps organizations to attract more traders and potential donors by operating on such a global channel.

Therefore, our non-profit website will use an online payment system for its members and potential donors to contribute with their donations. There are several ways for non-profit organizations to process an online payment transaction. Two of the most commonly used payment methods are Mobile Payment using Short Message Service (SMS) or by credit card via some third party. These third parties are called Payment Service Provider (PSP) or payment gateways that provide secure online transactions.

The online payment by credit card can be considered the most widespread. However in recent years and with emerging new technologies, Mobile Payment Services are gaining popularity and that makes them a powerful source of revenue for organizations. Such payment methods provide an alternative method to credit card payment for those users who are not willing to leave their credit card information on a website.

There are different methods available for mobile payments, each of which using different technologies and approaches, [10]:

- Premium SMS based transactional payments
- Direct Mobile Billing
- Mobile web payments (WAP)
- Contactless NFC (Near Field Communication)

The one interesting for our application is the Premium SMS base transactional payment.
3.3 Premium SMS based transactional payments

Premium SMS based transactional payments are phone-bill based, and it is the most commonly used model by mobile network operators [11]. This technology provides a mobile network operator or a third party to charge users for sending an SMS message to a premium number (short code). The customer sends the SMS message to the short code and the corresponding amount is debited to the phone-bill or top-up card. The interesting aspects of this method are security, usability and convenience for the user to use it for making payments.

The SMS payment process is very similar to sending a normal SMS text message except that the text is sent to a particular short code premium number. Most mobile phones are delivered with the SMS application without the need for any added software. The transaction period is usually very short and upon completion, a confirmation message, either success or failure, is sent back. It is a very convenient and attractive way for donors without a credit card as it does not require a bank account. It is also convenient for micro-payments (up to $2) or mini-payments ($2 - $20) for users with or without a credit card. On the other hand a possible drawback may be the limited network coverage in some areas since the related technology in use in Europe and parts of Asia is still not yet fully adopted in other parts of the world [10].

Most of the security issues concerning mobile payments are very much related to three other models of Mobile Payment; Direct Mobile Billing, Mobile web payments (WAP) and Contactless NFC (Near Field Communication). In fact, security issues regarding SMS payments are directly handled by the mobile network operators, thereby eliminating the need for organizations to expose and to store any personal information on their website. The transaction is encrypted and that makes it difficult for packet sniffers (a computer program that allows eavesdropping on traffic traveling between networked computers) to make use of the data. The security issue in the case of SMS payment concerns mostly the timely delivery of the SMS message and the fact that mobile phones could be lost and used by somebody different from the owner.

There is support for integrating an SMS premium payment system with most platforms such as ASP.NET, JSP, PHP and RUBY.

3.4 Credit card Payment Method

One of the most common forms of online payment methods is by credit card. Nonetheless, there are still a large number of people who are unwilling or hesitant to leave personal information on websites, particularly on smaller or less known ones. To overcome this problem, small organizations adopt another method that allows a third party called Payment Service Provider (PSP) to handle the security issues. This includes risk management, fraud protection, customer confidence and all technical connections with external networks. PSP are recognizable and trusted brands that generate trust and confident among users and donors.

Two known PSPs associated with non-profit organizations are PayPal and Google Checkout. The section below briefly explains the services offered by the two providers.
3.4.1 PayPal

PayPal provides services that help individuals and merchants to pay, accept and transfer money without having to reveal any financial information on host websites. It is similar to a digital wallet where all online transactions along with bank accounts and credit cards are securely stored. PayPal offers merchants and non-profit organizations to accept various types of payments and donations against a commission for handling the transaction.

PayPal is one of the most adopted forms of online transactions used on the web and available in 24 major currencies. It has a global reach across 190 countries and regions, making it a popular choice for many organizations. PayPal's charge rate for non-profit organization can be as low as 2.2% per transaction; it has helped raising $1.8 billion in 2010 for over 200,000 non-profit groups.

PayPal shields the customer's credit card and personal information from security vulnerabilities by offering SSL (Secure Sockets Layer) to encrypt personal information when transmitting over the internet.

PayPal provides a test account and a sandbox toll to simulate a real environment for testing transactions on your website without having to involve real credit card details. Upon successful testing, the migration to real service environment requires only minor changes and of course a valid account [12].

3.4.2 Google Checkout

Google is another PSP provided by the online search engine Google. Just like PayPal, Google Checkout also facilitates non-profitable organization to collect donations at fairly low rates. Moreover they provide Google Grant to selected charitable organizations to advertise free of charge on Google [13].

Also Google Checkout shields and protects the customer's credit card and personal information from security vulnerabilities by offering SSL protection when transmitting monetary transactions over the internet.
The difference between the two payments methods are summarized in Table 1.

<table>
<thead>
<tr>
<th></th>
<th>PayPal</th>
<th>Google Checkout</th>
</tr>
</thead>
<tbody>
<tr>
<td>Setup time</td>
<td>quick and easy for simplest implementation</td>
<td>quick and easy for simplest implementation</td>
</tr>
<tr>
<td>Accept payment methods</td>
<td>Credit Card, Debit Card, Direct Debit, eCheck</td>
<td>Credit Card, Debit Card</td>
</tr>
<tr>
<td>Cost</td>
<td>$0.30 + 1.9-2.9% per transaction</td>
<td>$0.30 + 1.9-2.9% per transaction</td>
</tr>
<tr>
<td>Non-Profit Discounts</td>
<td>Yes, 2.2% per transaction</td>
<td>only for Google Grants recipients</td>
</tr>
<tr>
<td>Branding</td>
<td>trusted, well known payment processor</td>
<td>Google Checkout logo appears by your site in search listings. Said to boost click through rates.</td>
</tr>
<tr>
<td>Customer service</td>
<td>phone prompt, can eventually speak to a human</td>
<td>Email only</td>
</tr>
<tr>
<td>Fraud protection</td>
<td>excellent, but does not handle complaints for orders under $50</td>
<td>Excellent</td>
</tr>
<tr>
<td>Record keeping</td>
<td>sophisticated invoice system suited to small businesses</td>
<td>rudimentary record keeping</td>
</tr>
<tr>
<td>Merchants/sellers</td>
<td>55 countries</td>
<td>US and UK only</td>
</tr>
<tr>
<td>Currencies supported</td>
<td>24</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 1: Site-by-site comparison

### 3.5 Payment choice

Both PayPal and Google Checkout are quite easy to set up and provide ad-hoc APIs (Application Programming Interfaces) for all major frameworks. Even Mobile payment methods have most often a set of standard APIs for various frameworks. All three methods have good security mechanisms for online transactions. Multiple payment options are excellent for any charitable organization to provide their donors with various options. An international website can benefit from any of these methods by providing the most effective option on a country-by-country preference.

After carefully considering the stakeholders’ needs, we concluded that although the SMS payment method is very much adopted in Europe and in parts of Asia, but not in other part of the world, it became the least favorable option for us. We also found that PayPal and Google Checkout match in many aspects but PayPal's customer support system, its popularity and payment options meet most of our needs.
3.6 Environmental scanning of similar websites

Environmental scanning consists in gathering information from multiple sources that provide a similar solution to ours. This information is then analyzed to help both stakeholders and developer understand the latest trends and important issues related to the specific application field. Performing such study often helps in avoiding costly mistakes [5].

The author studied and analyzed the current trends and strategies used in attracting new donors, and apply them in a suitable way for the organization.

There is a vast amount of non-profit websites dedicated to helping people in need all over the world. Some of these inspired this work and have simple and secure payment systems as well as effective methods for attracting new donors. In [6], the author recommends ten key characteristics for building efficient charity websites. Among those we selected the followings to characterize our solution:

- **Home page** should give clear description of the organization's purpose or mission. First time visitors or potential donors may not know about the organization and its purpose. Upon arrival, the home page should quickly give these visitors a uniform impression about the organization's mission.

- **About Us** page should describe the detailed background, mission and accomplished milestones of the organization.

- The website should give simple and clear messages to its audience, including members, volunteers and visitors.

- Information for donors about how to make donation and providing multiple payment options is crucial.

- Photo albums section that shows how the organization has helped and helps the targeted region(s).

- Contact information for donors and users to get in touch with the organization.

Similar non-profit organizations:

  - One of the world's largest humanitarian networks, their mission is to prevent and alleviate human suffering in all countries regardless of political affiliations. They are present in over 180 countries.
  - Offers membership
  - Payment method: SMS payment, Internet banking, credit card, PayEx, bankgiro system

- **Africa Groups** - [http://www.afrikagrupperna.se/english](http://www.afrikagrupperna.se/english)
  - Is a solidarity organization independent of any political or religious affiliations.
- Offers membership and even for potential volunteers
- Payment method: Direct internet bank, bank giro system

  - An international humanitarian organization working for a world free from poverty and injustice. They are working with more than 25 million people in than 40 countries.
  - Offers membership
  - Payment methods: Multiple methods in different countries

- Other non-profit organizations can be here:
  [http://www.bidra.nu/FattigdomoUtveckling.html](http://www.bidra.nu/FattigdomoUtveckling.html)

### 4 Problem formulation

In order to both increase their visibility and reach a broader range of potential members and donors the charity organization decided to rely on the Internet and more specifically on the development of an easy-to-manage website. Among all the available possibilities, according to the considerations given in Section 3, we decided to develop the website by means of a WCMS. During a meeting with the administration, a list of key functionalities for the website was presented as a foundation for the creation of a requirements document for the project.

Web owners like to keep people interested on their website and even entice them to return back by providing fresh and updated articles as often as possible. One way for our site to accomplish this is to design the WCMS to provide features that will allow more people to participate in content writing and allow them to dynamically edit and upload their content directly on the website. Content writers won't need to e-mail their article to the site administrator to upload it on the site database or to depend on webmasters for such services. These content writers can be even anywhere across the world where they can remotely manage their article with just an access to a computer and an internet connection to access the website. Once you have a source of article, a second problem arises: how can you add them to our website. The people who would be responsible for administering the website on a daily basis may not be knowledgeable in web programming. Additionally, the web content management should be made in such a way that the administrators do not have to depend on the webmasters for editing and uploading contents on the website. Non-technical users like to easily edit and publish the web contents without depending on webmasters. Additionally, the system has to provide features such as the capability to organize article into categories and show abstracts. The last problem is the system to provide access privileges to one or more administrators, editors and contributors to submit articles and allow other users to just read the articles. Finally, the system should provide features for registered users to upload pictures directly on the website.
4.1 Proposed Solution

Our solution aims at solving the issues described above by developing a web-based content management system with features such as separate administrative section for content management and general user section for displaying the contents. The main facility is to allow the administrator and content writers to manage their web content without any programming knowledge; other features to be implemented are the ability to categorize articles into appropriate categories and enabling registered members to upload and edit pictures and also edit their profile when the user is authenticated.

5 Method

When the concepts for the project were initiated a project plan was drafted and presented to the stakeholders in order to reach an agreement on the project plan and outcome. Different technologies are explored and compared in order to pick the more suitable for the project purposes. A software development process methodology is also chosen in order to follow an efficient software development work-flow and thus make sure that the requirements are fulfilled. Moreover, this methodology allows non-technical stakeholders to easily follow the development progress. Once technologies are chosen as well as a development process, a requirements specification is agreed with the stakeholders and modeled through state-of-the-practice techniques, such as scenario-driven and use-case-driven. The system is modeled from architectural and behavioral perspectives too. Before starting with the actual implementation, the modeled system is presented to the stakeholders for eventual modifications before starting with the actual implementation. Once the implementation phase is completed, the application is tested and evaluated by both the developers and the stakeholders in order to finalize and eventually tune features for achieving the final product. The following sections of this chapter describe the respective methods used in the different phases of the actual development process.

5.1 Software Development Life Cycle

Any software development project goes through a process known as a software life cycle. The International Standard, ISO/IEC 12207:2008 have established a common framework outlining the standard that defines the activities required for software life cycle processes [21]. These activities contain tasks that are to be applied during project life cycle. The methods used for implementing the software is based on the development process model described in the next section.

Somerville defines software process in [7] as a set of activities and processes that are involved in a production of a software product. Boehm states that the primary functions of a software process model are to determine the order of the stages involved in software development and the criteria for progressing from one stage to the next one [22]. Boehm further contends that software process models are important because the models guide the developer(s) on the order in which a project should be carried out. These days a wide variety of software process
models have evolved, each with its own strengths and weaknesses. Some of these recent models today are the rapid throwaway prototypes, incremental development, evolutionary prototypes, reusable software, and automated software synthesis. In [7], Somerville discusses various software process models and identifies that one process model is not necessarily suitable for use by all projects. Each process model is best suited to specific kinds of projects, based on various techniques and organizational needs. Nevertheless, some fundamental stages are common to all the software process models, example of such stages are the requirements phase, the design phase, the implementation phase and the testing phase.

In the same work Somerville discusses the following software development models:

**The Waterfall Model:** This model is represented as a sequential design process where the development process progresses downwards. The waterfall model was defined in the 1970s by Royce and later refined by Boehm in 1976. Because of this transition from one phase to another, the model is called the waterfall model, just like a waterfall. The sequential tasks involve in this model includes the requirement specification, software design, implementation, testing and maintenance. This model represents a linear model which makes it the most simple to be understood. One of the advantages of this model is that documentation is produced after every stage which makes the understanding of the development procedure simpler by all stakeholders. One of the disadvantages of this model is that if any of the previous stage has gone wrong, things can get very complicated in later stages. There are other modified versions of the waterfall model such as the V Waterfall model which aims to improve the shortcoming of the original model.

**The spiral model:** In order to overcome the disadvantages of waterfall model, spiral model was developed. This model is represented as a spiral where each phase incorporates the different stages of the waterfall model. The spiral model contains four phases which are: Planning, Evaluation, Risk Analysis and Engineering. As one move from one phase of the spiral to another, you repeat all the stages of the waterfall model. Some of the disadvantages of the spiral model are that it demands more time, it is complicated and requires highly skilled people in the area of planning, risk analysis and mitigation, development, customer relation etc.

**Evolutionary development:** This is model based on the idea of developing an initial implementation, presenting it to the stakeholders for feedback, development team responds to the feedback, often by refining the product until an adequate system has been developed. One of the advantages of this model is that the customers can see steady progress. One of the disadvantages is that it’s impossible to know at the outset of the project how long it will take. There are two types of Evolutionary development models:

1. **Exploratory development:** Objective is to work with the customers to evolve a final system from an initial outline specification. Process starts with the well-understood requirements.
2. **Throw-away prototyping:** Objective is to understand the system requirements. Process starts with the poorly understood requirements.

**Component-based software engineering:** This approach is based on already existing of reusable components that can be integrated into other components. After studying the different software development processes, the model chosen for this project will be a slightly modified version of the waterfall model. The pure waterfall model provides
very clear phases and considering that there is only one developer implementing the software, the developer can focus on each part of the model during stages and come back previous stages if needed to. The modified waterfall adopted for this thesis work uses the same phases of the pure waterfall, but on iterative basis. This enables the phases to overlap when needed.

5.2 The waterfall model for software development
The waterfall model recommends software development to go sequentially and follow certain rules and stages. The methods used for implementing the WCMS are categorized into different phases following the recommendations of the waterfall model.

**Planning phase:** During the planning phase, development of the product is initiated. Stakeholders and goals are identified. Various existing technologies are being explored and learnt and how useful they are in producing a well-organized software system.

**Requirements specification:** This phase is about gathering information about what the owner needs and defining it in the clearest possible terms and the problem that the software is expected to solve. Techniques used to obtain these specifications include Scenario-driven description. The results of the analysis form the requirement specifications then to serve as a contract between partners.

**Design:** This phase consists of designing the web layout interfaces to satisfy specified requirements. A mock-up layout is designed and presented to the stakeholders for approval. It also involves separating the administrative interfaces from all other users.

**Implementation:** This phase consists of actually constructing the software according to the design specification(s) developed in the previous phase.

**System Testing and Integration:** This phase is about integrating different components of the software and testing them in order to validate the requirement specifications.

**Maintenance:** This phase occurs after testing and deployment. It involves making modifications to the system to alter attributes or improve performance. However this phase is not covered in this work.

6 Developing the software
In development section, the WCMS is implemented as scheduled. The first phase is to prepare the necessary development tools and environment followed by subsequent phases as outlined in the previous section.

6.1 Planning phase
During the planning phase, various technologies are being explored and learnt and how useful they are in producing a well-organized, systematic and advance system.
The WCMS website has been developed using the ASP.NET framework because it integrates all the necessary tools we need for building our application. Such technology has been purely dictated by the developer’s programming knowledge in ASP.NET. The ASP.NET is a software framework developed by Microsoft for building web applications. ASP.NET has gained popularity among web developers for its good integration with the Microsoft IDE (Integration Development Environment) called Visual Studio [16]. Visual Studio provides various software programs for developing GUI (Graphical User Interface), websites, web applications, and so on. It also integrates AJAX (Asynchronous JavaScript and XML), LINQ (language-integrated query), HTML, CSS (cascading style sheets) and compliant compilers with debugging support for programming languages such as C#, C++, Python, VB and Script.

HTML is described as a markup language (not a programming language) that is based on a set of markup tags that are used to describe HTML documents in web pages. HTML was originally developed by Tim Berner-Lee in the early 1990s and it was part of the rapid growth of the World Wide Web. The Web depends on Web page authors and vendors sharing the same conventions for HTML. This has motivated joint work on specifications for HTML. HTML has been developed with the vision that all manner of devices should be able to use information on the Web [17].

6.2 Requirements Specification
Requirements specification is a document that describes the process of understanding and defining the services expected from the system and its constraints and limitations. This document establishes the contract between stakeholders and system development team [7].

Moreover, system requirements and user requirements specifications have been defined and described in the next sections. In the case of our WCMS website, there were number of basic expectations and therefore requirements for the system.

General questions
- How does the WCMS system support the work of the non-profit organization?
- How does a non-technical person manage the website that does not require further training?

Specification
- The website consists of a Web Content Management System that enables the administration team to make changes to the web site with very basic training. The web administrator should be able to add and delete contents of the web page
- It supports three groups of users; anonymous, registered and administrative user. The user requirements specify the different roles and permissions for the three groups
- The website should have a database to enable its members to register their profile.
- It should enable registered members to upload and edit pictures and also edit their profile when the user is authenticated.
- The WCMS website should allow the editors to add contents to the website.
- It should allow users to pay their membership and donations through an online payment system
6.2.1 User Requirements

The user requirements specification provides an abstract description for both functional and non-functional requirements, categorized by user group, which is easily understandable by non-technical end-users.

End-users have been analyzed and differentiated into three categories:

1. The anonymous user, who can only access the public pages.
2. The registered user, who has a limited access to internal contents.
3. The administrative user, who has unlimited access.

The anonymous web user is the user whose credentials are not being identified or authenticated by the system. This group of users has limited access to web contents and it encompasses all web users. The registered user has less restriction than the anonymous user but still less access possibilities than the administration team. The administrators have, in fact, unlimited access to all web pages and are responsible for the contents of the WCMS. The following subsections describe the high-level requirements and a summary of what is expected from the system.

6.2.1.1 Anonymous user requirement

Anyone having access to a computer and an internet connection should be able to browse the public pages (e.g. Home, Contact Us, Register, Photos and Donation pages). The anonymous user should be able to view all available photos. Anonymous web users will neither be able to create a photo album nor will they be able to upload pictures without being registered. Anonymous web users should be able to access the “Contact us” section to send messages to the administrators at any time. This group of users can even access the “Donation” web pages where users can make their donation to the organization. In order to do so, the user is required to be registered in order to proceed with their donation process. An anonymous web user can register to the website by providing user-name, password and e-mail address. On successful registration the user would get a confirmation by e-mail.

The anonymous user requirements are summarized in the following:

- Access to the sections:
  - Home
  - Donation
  - Photos
  - Contact Us
  - Register
- On the Register section, the user should have the possibility to create an account on the website by providing user-name, password and e-mail address.

The restrictions for an anonymous user include the following:

- Creation of a photo album
- Donation process without being registered
- Access to a registered user section
- Access to content management section
6.2.1.2 Registered user requirement
The registered user is the one whose credentials, usually in form of username (or e-mail) and password are authenticated by the system. The registered user can access information unavailable to the anonymous user. This group of users should be able to create photo albums and upload pictures on the “Photos” page. The registered user should have access to “Profile” page. This group of users should also be able to retrieve their password on the “Recovery” page by providing their e-mail address. The registered user should not have to access the content management.

The registered user requirements are summarized in the following:
- Access to pages available to the anonymous user
- Access to create photo album and upload pictures
- Access to profile page and ability to edit profile information
- Ability to change credentials, or recover lost password
- Ability to complete the donation

The restrictions for the registered user include the following:
- Access to content management section

6.2.1.3 Administrative user requirements
The administrative web user is responsible for the managing the website. The administrator should have access to any content and should be responsible for approving and publishing any item on the website. The administrator should be able to assign different roles to users, such as editor and member.

The administrative user requirements are summarized in the following:
- Access to the administrative section
- Ability to manage contents
- Ability to manage categories
- Ability to manage donations
- Ability to manage members and their profiles

6.2.2 Scenario-driven description
It is often easier for people to understand the system behavior when related to real-life examples than to an abstract description [7]. A scenario is an example-driven way of describing a system in which the system is used in practice or a user carries out some activity by interacting with it. It usually helps to describe an expected flow of events between the user and the system. Scenarios can also provide description of unwanted behaviors and possible related countermeasures. The scenario described in the following section describes the interaction between different users and the system.

6.2.2.1 Interaction users-system scenario
Sam (User 1) is a member of the “badibukafo” website and has an editorial role. Sam writes contents but does not have an administrative right to publish them on the website. Maria (User 2) is part of the administrative team and thus responsible for publishing contents or articles to the web. The scenario below describes the process of content creation and publication by the two end-users.

Content creation process:
1. Sam accesses the website and comes to the homepage.
2. He enters his username and password in the login interface.
3. The system validates his credentials which enable the editor user interface to appear.
4. On the administrative home page he clicks the ”manage articles and donations” link to get to the next page where he can choose to do any of the following:
   - add a new article
   - add a new category
   He clicks on the add new article link and comes to the ”Add article” page with a WYSIWYG (What You See Is What You Get) text editor.
5. He creates the content and finally inserts the data into the database waiting for an approval.

Content publication process:
1. Maria accesses the website and comes to the homepage.
2. She enters her username and password in the login interface.
3. The system validates her credentials which enable the administrative interface to appear.
4. On the administrative home page, she clicks the ”manage articles and donations” link to come to another page where she can choose to do any of the following:
   - add a new article
   - edit article
   - delete article
   - add a new category
   - publish article
   - manage donations
   - Maria reads the new content created by Sam; she approves it and finally publishes it on the web.
6.3 Designing the architecture

The first phase of the software design is defining the software architecture. Software architecture is defined as a high-level view of the system, the relationships among the subcomponents and their properties [14]. We chose the 3-tier architecture design model which divides presentation, business logic and data access into three separate tiers or layers. The system will comprise of user interfaces (or web pages) as Presentation Layer, a Business Logic Layer (BLL) that defines the functional part the system and acts as link between the presentation layer and the Data Access Layer (DAL) which is in charge of retrieving and storing the data. Figure 2 shows the 3-tier architecture while Figure 3 shows a high level view of the system and the user-system interaction.

![Figure 1: software architecture](image-url)
Figure 2: user-system interaction
6.4 Designing the website layout
We created a mock-up design to show how the final web pages would look like in a web browser. The website consists of two different sections with different layouts. Figure 4 shows the public section layout while Figure 5 shows the administrative section layout. The mock-up was then presented to the stakeholders that approved it.

Figure 3: mock-up public section layout
The administrative section layout consists of the administrative menu which is shown only when the user is authenticated as administrator. The interface has a link for managing categories, adding new articles and managing the donations.

On the bottom the articles extracted from the database are shown. For each of them related information is given: (i) article title, (ii) user who created it, (iii) location and (iv) abstract. The “approve” in the ellipse represent the button to approve the article for publication. The “edit” is the link for editing it. The “delete” represent the button to remove the article from the database.

Cascading Style Sheets (CSS) is applied to transform and style the appearance of the mock-up in the browser. CSS provides vast amount of options including fonts, colors, background colors, positioning of element in the in the web page, borders around elements and so on. CSS is compatible with all major browsers. Its compatibility with major development platforms makes it a very popular choice among web programmers.

Figure 4: mock-up administrative section layout
6.5 Implementation
The system behavior is designed using a Use Case technique. Using this technique makes implementation easier. It facilitates the developer to clearly identify the user functions and their relation. Use Case is a scenario-based technique for illustrating the interaction between actors and the system [7]. The actor represents an external entity (a person or an external device) that communicates with the system and the ellipses represent the action that the system performs (Figure 6).

![Use Case diagram](image-url)

*Figure 5: Use Case diagram*
Here is the Use Case related to the administrator’s attempt to login in the website, create contents and decide to postpone their online publication.

**Use Case description:** The administrator’s login in to the website.

**Name:** Login

**Initiator:** Administrator

**Goal:** Successful login to the website
1. Administrator types username and password.
2. The system accepts the login.
3. The system shows the administrative view.
4. The administrator performs some tasks and then logs out.

**What can go wrong?**

5. Login failed
   - Systems allows customer to try again
   - The system increments the login attempt count
   - The system checks if count is less than five then resume at 1
   - The system checks if count is equal to 5 then blocks the user

**Use Case description:** Content creation by the administrator.

**Name:** Content Creation

**Initiator:** Administrator

**Goal:** Create an article and insert it into the database.

**Initial assumption:** The administrator has logged in the website and has located the administrative page. The category drop-down-list is already filled with one or more categories to choose from.

1. The administrator clicks on the “Add new Article” link
2. The system displays the web page (addnewarticle.aspx) with WYSIWYG (What You See Is What You Get) text editor.
3. He writes the content in body of the editor. The text editor consists of optional and required fields. To categorize the article he selects a suitable category from the drop-down-list of categories. Gives the article a title and an abstract description in the in the title-field and abstract-field respectively. He leaves the “Approve article” checkbox unchecked so that the administrative team could have the time to review the article before publishing it. If any optional field is not filled then the default data will be applied to that field. The lists of fields are listed below. If he is satisfied with the content he chooses one of the buttons listed below. He chooses the insert button to insert the article into the database.
- Required fields:
  - Category, title and body
- Optional fields:
  - Abstract, Country, State, Release date, Expire date, Approve article, only for members
- Buttons options:
  - Insert article
  - Cancel article

4. The system inserts the article into the database and shows a new text editor
5. The administrator logs out

**What can go wrong?**

6. The content cannot be saved
   - The required fields are not filled
   - Go back to 3 and fill in the title, abstract and select a category from the list

**Use Case description:** Publishing the content on the website

**Name:** Content publication

**Initiator:** Administrator

**Goal:** Successfully publish the content onto the website.

**Initial assumption:** The administrator has logged in the website and has located the administrative page. The content to be published is created and stored in the database.

1. The system shows the manage article web page.
2. The administrator locates the article and clicks the “approve article”.
3. The system displays a dialog box asking whether the administrator wants to approve this article or not and wait for the respond. The dialog box has “Ok” and “Cancel” options.
4. The administrator clicks the “Ok” button to publish the article onto the web page.
5. The system displays the content onto the web page.
6. The administrator logs out.

**6.5.1 Implementing data access layer**

A database is defined as a collection of tables that stores data in a structured way so that it can be easily accessed, managed and updated. The database consists of a database Engine. This engine provides the core services for permanent storage of data. It also provides services for processing and securing data. The most popular database is the relational database which consists of one or more tables. Data in the tables are stored in rows and columns.

For the application, the relational database has been created with Visual Studio’s integrated Server Explorer to manage the data. In the database, two tables are created, *bk_Article* and *bk_Categories* for managing the contents. As shown is Figure 7, relationship is created between the primary key of the *bk_Categories* tables and a foreign key of the *bk_Article* table. The *bk_Article* table gets a relation to the *bk_Categories* table and only stores its CategoryId instead of the entire table.
Data Access Layer:
DAL is the layer which provides access to the database. Data Access Layer (DAL) is closest to the database thereby linking the overlying layer to the database. The DAL queries the database to insert, retrieve, update and delete data. The DAL includes a data access component called LINQ (Language-Integrated Query) that enables the developer to query data anywhere within C# or VB programming languages. LINQ is available for different frameworks. In our implementation we use the LINQ-to-SQL implementation which is part of Lenin the ADO.NET framework. LINQ-to-SQL enables the programmer to query the Microsoft SQL Server database. LINQ-to-SQL is actually LINQ to Microsoft SQL Server [15]. With LINQ-to-SQL implementations transform database objects into .NET objects that can be access anywhere in our code.

6.5.2 Implementing the Business Logic Layer

The presentation layer (or web pages) uses the classes in the Business Logic Layer (BLL) to provide access to data in the DAL. In another words, classes in the BLL are shared by all web pages. The three main classes in the BLL are: (i) Article, (ii) Category and (iii) Donation. They inherit from the super class called Base Article. Figure 8 shows an excerpt of these classes and their properties.

Article class:

The Article class provides properties and methods for creating and managing articles or contents on the website. It has an ID property which is unique for each article in the database. Other properties are: title, addedDate, addedBy, body, abstract and so on (Figure 7). The class also has categoryId and the categoryTitle properties to attach articles to the parent category. Moreover, methods that allow the caller to retrieve article, insert, delete article, approve article, and update articles are implemented.
The *Approve* method set the approve field of the specified article to true. This allows the administrative team to review the article before publicizing it on the website. The *insertArticle* method creates a new article and inserts it into the database. The *getArticleByArticleID* method returns all the fields of the specified article from the database. The *delete* method uses the *getArticleByArticleID* method to retrieve and delete the specified article from the database. The *updateArticle* method is used for updating all the field of the specified article in the database. The *update* method is used for updating a single field for the interested article. The caller can, for example invoke the *update* method for approving specified article for publication. The *articleCount* method returns the number of total articles stored in the database. The *getArticleFromArticleDetailList* method returns article object from the *articleDetail* class.

**Category class:**
The Category class provides properties that describe a category of an article such as the title, image and a description. Just like the Article class it also has methods to allow the caller to create, delete and update categories.

The *delete* and *deleteCategory* methods delete category items from the database. The delete method deletes a single category while the *deleteCategory* method deletes all the categories from the database. To retrieve a single article the caller invokes the *getCategoryByArticleID* method which returns a category object with a specified id. The caller can use the id to delete the given category from the database. The *updateCategory* method is used to update a particular category in the database. The *getCategory* method returns a list of all categories from the database.

**Donation class:**
The Donation class provides properties and methods for the caller to make a donation on the website. It provides properties for personal information such as name, surname, address, the amount and a message. The Donation class has methods for retrieving, inserting and updating donation records in the database.

The caller can invoke the *DeleteDonation* method to delete a specified donation record from the database. The caller creates a new donation record by invoking the *insertDonation* method. The *getPayPalPaymentUrl* method invokes the PayPal API (Application Programming Interface) which enables the caller to be redirected to PayPal’s website for payments transactions. The *getDonationID* retrieves the id for the specified donation record from the database system. The retrieved id is used by the *update* method to update the specified donation record.
The results are shown in the presentation layer. The main function of the presentation layer is to allow the interaction between end-users and the system. In the case of our web-based system, it displays HTML resources as web pages to enable such interaction. The presentation layer comprises of both public and administrative web pages. The core web pages are described below.

**Home page:**

The first public web page visitors see is the default “Home” page when the URL name of the website is entered in the web browser. Visitors can also register or retrieve forgotten password from this page. All users (including administrator) use the same login interface on the “Home” page to login. Upon authenticating, the information and the interface the user sees is limited by the ACL (Access Control List). The “Home” page is shown in appendix 1.
Register page:
To become a member users have to create an account on this page. User can create an account by providing a user-name, password and an e-mail address. A security question and answer is also required. On successful registration the user gets a confirmation by e-mail. See appendix 2.

Managing photo album page:
Once users are authenticated they can upload photos on this page. Users select a photo from their computer’s and insert it in the image gallery called Photo album. Users are not required to login to view photos. See appendix 3.

Contact page:
This page provides the interface where users can send messages to the administration. To send a message, the user must provide his or her name, e-mail address and the message body. If the message is successfully sent a confirmation message is displayed. See appendix 4.

Manage article:
This is a page of the administrative section. It comprises of links and buttons for managing the contents. The administrator can add, edit or delete articles. The interface provides a link to manage categories and donation records. The article title provides the link to review the article. The administrator can also create new article using a simple text editor. See appendix 6 and 7 for article management and article creation.

6.7 Testing
Before an application or website can be deployed it has to be tested to satisfy the requirement specifications. They are many types of test to carry out on a web application such as performance, functionality, database loading time, response time, server time handling, user’s actions and many others. During the implementation of the website owners and end-users were able to follow the development on the temporal website by the courtesy of MDH University. This facility enables end-users to test the functionality of the application online. A final presentation was shown to the owners and they were able to test system functionalities. The following tests below were noted during the testing.

Does the website represent the organization’s purpose?
✓ A web page describing the organization.

Browser check:
IE, Firefox and Google Chrome
✓ Load the website on above browsers.
✓ Registration on the website using the above browsers and registration confirmation by e-mail.

Functionality check:
✓ Members can create a photo album and upload pictures.
✓ Members can make donations via PayPal.
✓ User can send messages to the administration via the Contact page.

Administrative section
✓ Allow access to only the administrative members.
Administrator can create content.
Administrator can publish content on the web page.
Simple WYSIWYG editor for entering content.
Administrator can administer users’ profile and donations.

7 Summary and conclusions
The developer defined system specifications and selected the ASP.NET technology for the implementation of the WCMS together with the stakeholders and modeled the system following the state-of-the-practice techniques, e.g., scenario-driven description together with Use Case diagrams.
The modeled system was presented to the stakeholders and some modifications were done at modeling level before starting with the implementation.
Once the implementation phase was completed, the application was tested and evaluated by both the developers and the stakeholders who resulted satisfied by the final product hence not demanding any change in the running WCMS.

One of the main goals of the website was to represent the organization’s mission in helping less fortunate people on the WWW. To achieve this goal, the author performed an environmental scanning of similar websites for adopting the most suitable solutions among the current trends in WCMS development and online payment methods.

The website is developed with an administrative and a non-administrative section. Only authenticated administrative members and content writers can access the administrative area. The system provides features that facilitate users with no web-programming knowledge to manage web contents without any dependency on webmasters. Administrators can add, edit, delete and manage web contents. The system provides features that allow content writers to remotely access the website and upload their article on the database. Users can register and upload photos. Before the website is hosted on the permanent commercial server, a thorough and exhaustive testing has to be carried out. Finally, even if not representing a fully implemented enterprise WCMS, this work is an already working (though basic) version for a future more complete implementation.

8 Future works
This system is developed mainly to target a charity group and its end-users but with just few changes the system can also be utilized in many real world applications. For our stakeholders the system has delivered its requirements but there are lots of improvements that can be done. One of those improvements is to add multiple payment methods for the donors. The system can also provide more privileges for members to make comments on an article. Many non-profits websites advertise their web contents to other websites or clients using RSS (Really Simple Syndication). Our website can also be improved to implement RSS to provide newsfeed to those websites interested in promoting non-profit works. The system can also be improved to add forums and videos to attract more donors on the website.
9 References

[0] Last checked 2011-10-24
URL: http://resourcepage.gambia.dk/region3.htm


URL: http://www.ctg.albany.edu/publications/guides/and_justice_for_all?chapter=6&PrintVersion=2

URL: http://vandelaydesign.com/blog/design/non-profit-organization-websites/


[8] [blackbaud] last checked 2011-10-24

[9] Last checked 2011-10-24
URL: http://en.wikipedia.org/wiki/E-commerce_payment_systems

[10] Last checked 2011-10-24
URL: http://en.wikipedia.org/wiki/Mobile_payment


[12] Last checked 2011-10-24
URL: https://www.paypal.com/cgi-bin/webscr?cmd=xpt/Marketing/general/what-is-paypal-outside

[13] Last checked 2011-10-24
URL: https://checkout.google.com/seller/fees.html


[21] Last checked 2012-01-13
10 Appendix

Appendix 1 below shows the home page of the web site. It shows the abstract and a link of the article. Users can click the link to read the whole article.

Appendix 2 below is the page where users can register for membership.
Appendix 3 shows how registered users can create a photo album and upload photos on the website.

Appendix 4 shows how registered users can create a photo album and upload photos on the website.
Appendix 5: Contact page below enables site users to send messages to the administration.
Appendix 6: administrative interface below provides user interface for content management.
Appendix 6: administrative area for creating a new article.

Appendix 8: Interface for donors to make their donation.
Appendix 9: PayPal webpage. The user has to login with his or her PayPal password or create one if she does not have one to proceed.
Appendix 10: PayPal confirmation webpage.

This page is to confirm the payment. The user can click on the link to view her PayPal account or click that returns her to our website.
You just made a payment of
250.00 SEK

Print receipt
Paid to
badibukafo@gmail.com
Ship to
Test User
1 Main St
San Jose, CA 95121
United States

Thanks for your order

✓ Test, you just completed your payment.
Your transaction ID for this payment is: 59H61454 LX3383J
We'll send a confirmation email to justofo_1312 111918_pe@e.com.

Return to badibukafo@gmail.com
Got a PayPal account overview
Add funds from your bank