TABLE OF CONTENTS

CHAPTER ONE

1.1 INTRODUCTION

1.2 Background Information

Transmission of information has been an important problem which man has tried to solve over the ages. In the olden times, town criers were used to distribute useful information to members of communities and cities. Later on, more sophisticated methods such as the newspapers and books were used. With the invention of the Radio, transmission of information over long distances became infinitely easier. Also, the element of entertainment and other uses could be added to give the technology more appeal to the public.

Today, Radio technology is a mainstay in our society and it is common occurrence for there to be several FM (Frequency Modulation) and AM (Amplitude Modulation) radio stations in any given area. These stations act as a source of entertainment as well as information to the general public.

It is a common trend also, for new advancements in technology to be integrated into existing and already established technologies. It is therefore not surprising that the Internet technology can be integrated into several already existing technologies, one of which is the Radio.

This report outlines the methods taken to design and implement a Radio Transmitter using Internet Technologies.

1.3 MOTIVATION AND SIGNIFICANCE

A lot of Universities and other colleges of higher learning have radio stations which they use to fulfill different needs. However, conventional Radio stations require expensive FM or AM transmitters to deploy the signals. They also require Radio receivers which receive said signals and convert them into sound. This means that a sizeable investment has to be made for such a radio station to exist.

At Covenant University, there already exists a campus wide Intranet network. This network is robust enough to serve as a carrier for the signals of the radio station. Also, a large number of student and staff of Covenant University already possess laptop or desktop computers which they connect to the aforementioned Intranet network on a regular basis.

This means that, the infrastructure required to deploy a radio station utilizing the Internet technology is already in place at Covenant University.

A radio station is of great significance to an institution such as Covenant University because, it can serve to bridge the gap created by the ban on mobile phones within the campus. Information which typically would have been transmitted by word of mouth via mobile phones can be broadcast to all the students who tune in at the same time. Also, it can serve as a source of entertainment by streaming music directly to the computers of the recipients.

Using Internet technology to deploy the Radio station also creates several added advantages. For instance, other Internet technologies such as the HTTP (Hypertext Transfer Protocol) can be utilized to create a website where listeners to the radio station

can login to interact and discuss about the radio programs or any other topic of interest. Also, other service such as the FTP (File Transfer Protocol) can be used to enable file transfer between different users of the network. Thirdly, the Internet can be used to create a dynamic radio station where content can be added, removed or shuffled on the fly. These features cannot be realized using analogue radio technology.

There are several advantages to using Intranet Technologies to deploy a Radio station and the disadvantages are mostly negligible.

1.4 AIMS AND OBJECTIVES

The Aim of this project is to design a Radio Transmitter using Internet Technologies.

The objectives are

i. To provide a cheap means of radio transmission utilizing already existing technologies

ii. Provide an easy means of communicating important information within Covenant University

iii. Provide other services built around the Radio station also utilizing Web technologies serve as a new and exciting usage of the campus wide Intranet network already in existence in Covenant University

iv. Serve as a source of entertainment by streaming music via the radio network

1.5 METHODOLOGY

The design and implementation of this project is divided into various phases. This includes the Hardware phase, Configuration, Software Development, Testing and Deployment Phase.

1.5.1 HARDWARE PHASE

During the Hardware Phase of the project development, the Hardware required for the project has to be purchased. The required hardware include

i. A desktop computer to act as a server

ii. A Wireless access point to broadcast the Intranet network

iii. A microphone for recording Jingles and Announcement

A server computer has to be a powerful computer with the ability to host memory and processor intensive applications that are accessible by client computers over a network. It must be able to effectively stream audio files and has to have network ports for the network connections to be made. It also has to have a hard disk large enough to accommodate the operating system, server application, application software and other data files and databases required.

The wireless access point is required because it serves to distribute a WiFi (Wireless Fidelity) network which other computers can connect to in order to have access to the Radio. Ideally, the server is meant to be connected to the Intranet network of the University. However, for the purpose of simulation of the project as a prototype, the wireless router (access point) is used instead. The network distributed by the wireless

router purchased has to comply with IEEE 802.11b,g,n standards in order for it to be compatible with most of the computers available today.

The microphone is not part of the final project. It is only used to record advertisements and other jingles used on the radio station.

Other extra hardware components such as additional memory or Disk space for the server might have to be purchased during the implementation phase of this project.

Care should be taken to ensure that the hardware components are surge protected in order for them to be immune to power surges and other such disturbances.

CONFIGURATION

HARDWARE CONFIGURATION

After all the hardware has been purchased, the configuration phase is next. Here, the hardware components are connected together and configured properly to enable the maximum output from them. The Server operating system will be installed on the server computer, as well as the Server software required. Also, the wireless router has to be configured for effective transmission and maximum range as well as other settings to ensure that clients can connect to the server via the network created by the router. Finally, all those components have to be interconnected with each other using network cables, power cables, and other kind of connectors.

SOFTWARE CONFIGURATION

This Phase involves the installation of the Operating System, Server Software, and other application software to be used. It also involves installation of the firmware on the wireless router and configuring them appropriately.

The Operating System used in this Project is Fedora 14 Linux i386. This Operating system was chosen because it is a Linux based Server Operating System, and as such is more powerful than other alternatives. Also, it is more extensible and configurable than its alternatives. Thirdly, it is free and Open Source (FOSS) and help can be easily found online for it. Other bits of software used are listed below

i. Server Operating System – Fedora 14 Linux i386

ii. Streaming Server – Icecast2

iii. Web Server – Apache2

iv. Programming language – PHP, Python

1.5.2 SOFTWARE PHASE

This Phase of the project is the longest and most challenging. It involves development of applications which will run on the server and power the Intranet Radio Station.

The front end of the Radio is a web application. This is powered by a Web-server and displayed on a Browser. The Web Application/Website is developed using PHP (Hypertext Pre Processor) and MySQL (My Structured Query Language) for the server side, as well as HTML (Hypertext Markup Language) and CSS (Cascading Style Sheets) for the client side / presentation.

The backend of the Radio is powered by a Streaming Server known as Icecast2. Icecast2 streaming server requires configuration scripts to be written to properly calibrate it in order to perform the required action. Also, the stream which is being broadcast by the Icecast2 servers is generated using a script that is written with Python programming language.

All these components will be running on the server operating system. As mentioned above, the Fedora 14 Linux i386 OS has been chosen. This Operating System has to be efficiently configured with all the server components such as the DNS, DHCP, HTTP server, Database Server, FTP, etc. These services ensure that both the backend and frontend function properly; thereby making for a functioning Project.

1.5.3 TESTING

After the Software has been developed and installed appropriately on the hardware, the next step is to test the project confirm that it does what it is supposed to. In this case, the project is the Intranet Radio Transmitter. Its function is to transmit audio streams using Intranet Technology via WiFi. In this case, there will be three individual streams from different mount points called channels. All streams must be running simultaneously and serving different content. In order for the project to be called successful, all these have to be functional. Therefore, for the project, the testing phase involves confirming that the intranet radio is working properly under normal conditions. After this has been confirmed, the project can move on to the next and final stage.

1.5.4 DEPLOYMENT

This project can be deployed in a real world situation. For this to occur, it has to be implemented in a community which already has a wireless networking infrastructure. The server will be connected to that infrastructure, thereby automatically providing access to the Web Radio from anywhere within the network's range. Also, depending on the number of clients expected, the specification requirement for the server computer might have to be upgraded to accommodate more traffic.

1.6 THESIS ORGANIZATION

Chapter one is an introduction to the project work where the topic is explained, the aims and objectives are expanded upon, and the method of execution and testing are enumerated.

Chapter two is the literature review in which the theoretical background on which the project is based is explored in detail. Also, past work on the field is considered, chronicling their differences with the present thesis.

Chapter three is the Analysis and Design Chapter. Here, the project is analyzed in detail and the design decisions and why they were taken are explained

Chapter four deals with the deployment and testing of the project. At the end of Chapter four, there should be a working project which has been adequately tested.

Chapter five concludes the report. It features conclusion, achievements and recommendations.

CHAPTER TWO

LITERATURE REVIEW

2.1 INTRODUCTION

This chapter deals with the fundamental theoretical background and concepts necessary for the proper understanding of the scope of work carried out in this project. It also contains studies of previous works that have been done in this field and how they vary from this implementation.

It comes from studies of various academic texts, Internet articles, and samples of existing and related projects.

2.2 RELEVANT THEORETICAL CONCEPTS

INTRANET

The Intranet is a private computer network that uses Internet Protocol technology to securely share any part of an organization's information or network operating system within that organization. The term is used in contrast to Internet, a network between organizations, and instead refers to a network within an organization. Sometimes the term refers only to the organization's internal website, but may be a more extensive part of the organization's information technology infrastructure. It may host multiple private websites and constitute an important component and focal point of internal communication and collaboration [1].

Since Internet and intranet are similar, the same technologies are employed to in their design. Several protocols make up the intranet. They are enumerated as follows

TCP/IP (TRANSMISSION CONTROL PROTOCOL/ INTERNET PROTOCOL)

The Internet Protocol Suite is the set of communications protocols used for the Internet and other similar networks. It is commonly known as TCP/IP, named from two of the most important protocols in it: the Transmission Control Protocol (TCP) and the Internet Protocol (IP), which were the first two networking protocols defined in this standard. Modern IP networking represents a synthesis of several developments that began to evolve in the 1960s and 1970s, namely the Internet and local area networks, which emerged during the 1980s, together with the advent of the World Wide Web in the early 1990s.

The Internet Protocol (IP) Suite consists of four abstraction layers. From the lowest to the highest layer, these are the Link Layer, the Internet Layer, the Transport Layer, and the Application Layer. The layers define the operational scope or reach of the protocols in each layer, reflected loosely in the layer names. Each layer has functionality that solves a set of problems relevant in its scope [2].

URL (UNIFORM RESOURCE LOCATOR)

A Uniform Resource Locator (URL) is a Uniform Resource Identifier that specifies where an identified resource is available and the mechanism for retrieving it. The best-known example of the use of URLs is for the addresses of web pages on the World Wide Web [3].

DHCP (DYNAMIC HOST CONFIGURATION PROTOCOL)

The Dynamic Host Configuration Protocol (DHCP) is an auto configuration protocol used on IP networks. Computers that are connected to IP networks must be configured before they can communicate with other computers on the network. DHCP allows a computer to be configured automatically, eliminating the need for intervention by a network administrator. It also provides a central database for keeping track of computers that have been connected to the network. This prevents two computers from accidentally being configured with the same IP address.

In the absence of DHCP, hosts may be manually configured with an IP address. Alternatively IPv6 hosts may use stateless address auto configuration to generate an IP address. IPv4 hosts may use link-local addressing to achieve limited local connectivity.

In addition to IP addresses, DHCP also provides other configuration information, particularly the IP addresses of local caching DNS resolvers. Hosts that do not use DHCP for address configuration may still use it to obtain other configuration information.

There are two versions of DHCP, one for IPv4 and one for IPv6. While both versions bear the same name and perform much the same purpose, the details of the protocol for IPv4 and IPv6 are sufficiently different that they can be considered separate protocols [4].

WIFI (WIRELESS FIDELITY)

This is the name of a popular wireless networking technology that uses radio waves to provide wireless high-speed Internet and network connections. The WiFi Alliance, the organization that owns the Wi-Fi (registered trademark) term specifically defines Wi-Fi

as any "wireless local area network (WLAN) products that are based on the Institute of Electrical and Electronics Engineers' (IEEE) 802.11 standards."

Initially, Wi-Fi was used in place of only the 2.4GHz 802.11b standard, however the Wi-Fi Alliance has expanded the generic use of the Wi-Fi term to include any type of network or WLAN product based on any of the 802.11 standards, including 802.11b, 802.11a, dual-band, and so on, in an attempt to stop confusion about wireless LAN interoperability.

Wi-Fi works with no physical wired connection between sender and receiver by using radio frequency technology, a frequency within the electromagnetic spectrum associated with radio wave propagation. When an RF current is supplied to an antenna, an electromagnetic field is created that then is able to propagate through space. The cornerstone of any wireless network is an access point. The primary job of an access point is to broadcast a wireless signal that computers can detect and "tune" into. In order to connect to an access point and join a wireless network, computers and devices must be equipped with wireless network adapters [5].

The network to transmit the signals are received by computers and then interpreted. In the case of the World Wide Web, packets are sent and received. These packets are then interpreted into webpages that we can understand and interact with. In the case of the Radio, these packets are interpreted into sound which the media player can understand and interpret.

INTERNET RADIO

Internet radio is an audio service transmitted via the Internet. Music streaming on the Internet is usually referred to as webcasting since it is not transmitted broadly through wireless means [6].

There are two main areas associated with Internet Radio. The First is Webcast and the other is On-demand Radio.

WEBCAST

Webcasting involves streaming media, presenting listeners with a continuous stream of audio that cannot be paused or replayed, much like traditional broadcast media; in this respect, it is different from on-demand file serving. Internet radio is also different from podcasting, which involves downloading rather than streaming. Many Internet radio services are associated with a corresponding traditional radio station or radio network. Internet-only radio stations are independent of such associations.

Internet radio services are usually accessible from anywhere in the world—for example, one could listen to an Australian station from Europe or America [7]. This Project uses the Webcast methodology. This means that, the content streamed cannot be tampered with. This mimics the behavior of traditional radio stations and is already familiar with potential listeners.

ON-DEMAND RADIO

On-Demand Radio provides a stream of music for the listener when he wants it. In this type of Internet Radio, all the listeners do not get the same content at the same time. Each person listens to what he wants to at his/her own time. There are several music on-demand services in existence on the Internet now such as Pandora Radio, Last.fm, etc.

WORD WIDE WEB

The World Wide Web is the generic name for all the web pages that can be accessed via the Internet. The Web derives its name from the web-like structure constructed by the links between web pages.

At the heart lies the HyperText Markup Language (HTML) used to author the pages of a web site. Hypertext is electronic text that has cross-links within the document and between documents.

HYPERTEXT TRANSFER PROTOCOL (HTTP)

HTTP was developed for the delivery of HTML files. It is used for application level communication between the browser and the web server in distributed, collaborative, hypermedia information systems. The communications consist of requests and responses. The headers indicate the purpose of a request. The message uses the Uniform Resource Identifier (URI), in the form of the familiar URL or name (URN) to indicate the resource. In the response, MIME-like information describes the format of the data to be transferred. HTTP also is used as a generic protocol for communication between user agents and proxies or gateways to other Internet systems. This includes those supported by the RTSP, SMTP, and FTP protocols. In this way, HTTP allows basic access level to multimedia content sourced from a diverse range of applications.

This protocol is important for streaming, because it is used for the web pages that link to the content, and it is often the only choice for a communication protocol in simple network installations. Some company firewalls prevent access to regular streaming

media using real-time protocols, so HTTP is the only option for the delivery of media streams [8].

WEB SERVERS

At its most basic, the web server delivers a file in answer to a request from a browser.

i. The user asks for a page using a URL

ii. web server maps the URL to a filename

iii. server reads the file from disk

iv. server transmits the HTML file to the browser using HTTP [9]

In this project, the Wireless Router used would provide enough bandwidth to distribute the signals among a significant number of computers.

2.3 REVIEW OF PREVIOUS WORKS

Internet/Intranet Radio is by no means a new concept. There has been previous work done the area. Below, I give some insight into previous work that has been done in the field and how they differ from this project.

SMART RADIO – BUILDING MUSIC RADIO ON THE FLY

This project was involved in building a Web Radio. The Web Radio streams playlists created by the users. Therefore, it is an on-demand Radio. It also recommends tracks for the users based on their previous selections. As a result of the user system, people can recommend playlists for their friends to listen to. [10]

SIMILARITIES AND DIFFERENCES

This project above is similar to this project thesis in the fact that it is a web based radio system. However, its aim is not to mimic the behavior of the traditional radio but instead, provide a platform for creating playlists to listen to. In this regard, it is very different from this project.

SLURPCAST: A MEDIA STREAMING SERVER

This project implements a streaming server known as the Slurpcast streaming server. It studies the behavior of existing streaming servers such as Icecast (as seen in this project) and implements a similar functionality using the OCalm Programming language [11].

SIMILARITIES AND DIFFERENCES

The project above is similar to the subject of this thesis in the fact that it is a web radio project and it uses the streaming method. However, this project focuses more on the restructure of the streaming server itself. As such, it does not go into details about creating multiple channels, the stream source type, etc.

CHAPTER THREE

SYSTEM DESIGN AND ANALYSIS

3.1 INTRODUCTION

When carrying out a project, several factors are considered. In this chapter, I will outline the factors considered when designing the project and the decisions made. We will also explore the design used in carrying out the project in detail.

3.2 PROJECT ANALYSIS

Under analysis, we look at all the tools required for carrying out the project, both hardware and software tools.

3.2.1 HARDWARE

As stated earlier, the hardware for this project consists mainly of a Server computer and a wireless access point. However, several factors have to be considered to ensure that the hardware with the right set of features is used.

For the Server computer, the specifications have to be such that it can accommodate resource intensive applications and heavy network traffic. For all these to be attainable, the computer has to have a 10/100Mbps (10/100 Megabits per second) Ethernet NIC card, at least 1GiB (1 Gigabyte) of RAM (Random Access Memory), and at least Intel Pentium 3 Processor. Also, since there will be a lot of audio files stored on the server, the Hard disk must be at least 40GiB in size. For this project, having considered all the components needed, the configuration of the Server computer used is shown below

i. 2GB of RAM and above

ii. Intel Pentium4 Processor or newer

iii. 40GB Hard Disk or more

iv. 100Mbps Fast Ethernet NIC card

Other components that are required for normal operation of a computer are also present.

For the wireless access point, it is the networking device that distributes or broadcasts the signal. As such, it has to be powerful enough to broadcast steady streams of data and have a wide range so as to cover a wider area. The router used in this project is a Linksys WRT120N router. It supports IEEE 802.11n wireless standard. This means that, it can support speeds of up to 108Mbps which is the highest speed presently attainable with Wi-Fi technology.

3.2.2 SOFTWARE

The software components in use in this project are many and diverse. They can be split into several categories

i. Operating System

ii. Server Software (Backend)

iii. Client Software (Frontend) etc.

OPERATING SYSTEM (OS)

There are several major operating systems in existence today. Some of them are

i. Linux

ii. Apple Mac OSX

iii. Microsoft Windows

iv. BSD

v. UNIX [12]

Out of these, the most popular operating systems for servers are UNIX and Linux OSes [13]. There are several advantages to using the Linux Operating System and some of them are listed below

i. It is secure

ii. It is Free and Open Source (FOSS)

iii. There are several Distributions available mostly for free so finding one that fits your needs is easy

iv. It is stable, etc. [14]

In this project, a distribution of Linux known as Fedora from RedHat Computers is used. This is chosen because it is reputed to be the most cutting edge Linux version available and it can be easily configured to suit the needs of this project [15].

SERVER SOFTWARE

STREAMING SERVER

For this project, the streaming server, **icecast2**, is used. This is a free and powerful streaming server compatible with the Linux Server Operating System. It is also versatile enough to suit the needs of this project. [16]

MEDIA STREAM SOURCE CLIENT

This is an application which runs on the server. The main function of this application is to select the tracks to be streamed, and send it to the streaming server. It also has other functions such as, randomly selecting the tracks to stream, outputting the information on the currently streaming track and future tracks for use by the frontend software, etc. The Media Stream source client used in this project is developed from scratch for the project

WEB SERVER

Apache is the most popular webserver available today [17]. It is Opensource, and free to download and use. It is also compatible with the Linux server OS in use.[18]

FRONTEND SOFTWARE

The Frontend is powered by a Website or Web application. This web application runs on a Web browser. One of the Advantages to having the Frontend run on a web browser is that, most of the heavy processing to be done is performed on the server side, and so, the client computers do not need to be very powerful. Also, it affords for cross compatibility between different operating systems. Every computer Operating System that can access

the Internet and has a web browser can use the software. The Frontend software used in this project is developed from scratch for the project.

3.3 PROJECT DESIGN

The Aim of this project is to create an Intranet Radio Transmitter, whose function is to transmit different streams of audio in channels. The Audio is to be transmitted over the normal computer network protocols such as Ethernet and WiFi.

A user is to interface with the transmitter using a computer or similar device possessing a network interface card and a web browser.

The user with Administrator privileges will have the ability to add and/or remove audio tracks from any of the channels.

DESCRIPTION OF DESIGN

The use case diagram below shows a pictorial representation of a typical usage of this project

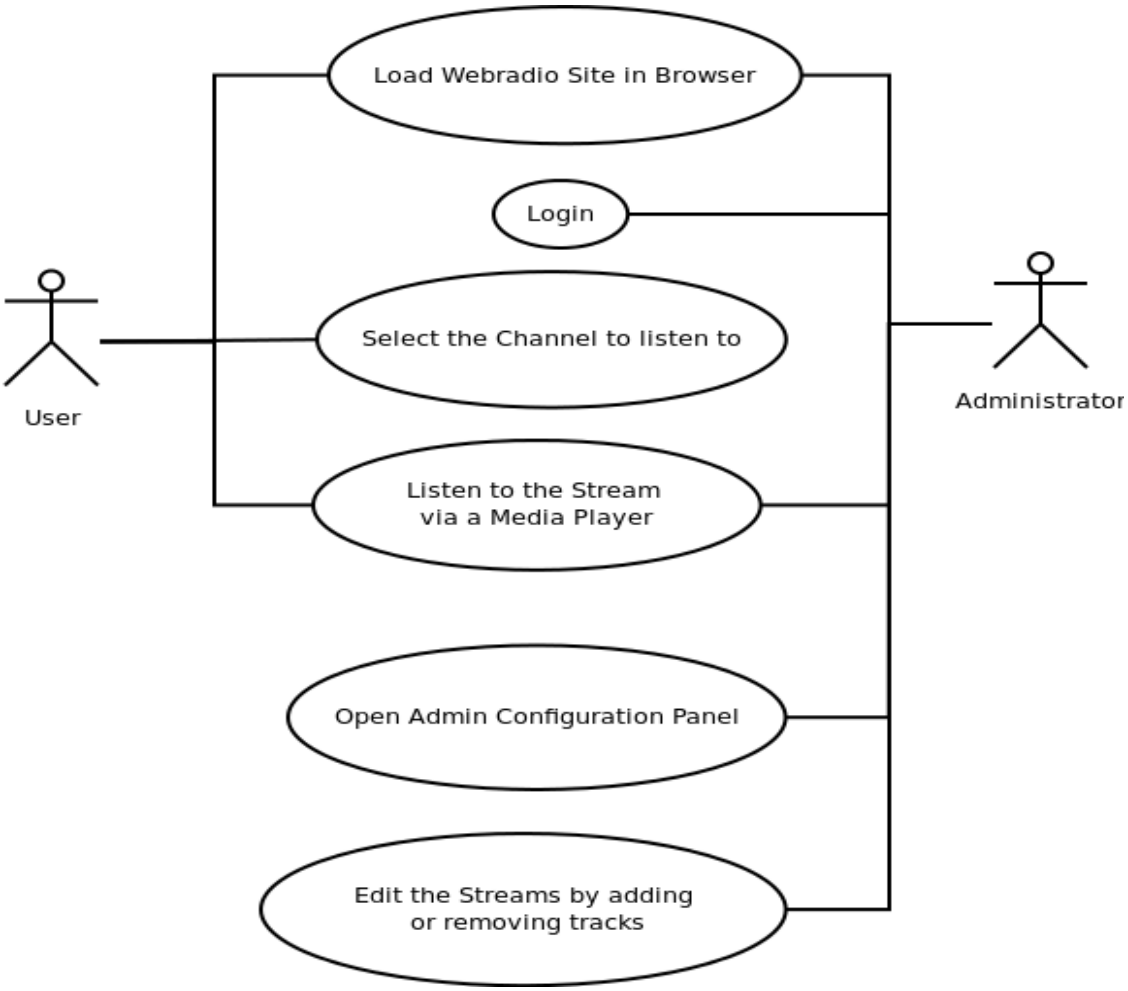


Fig 3.1: Use case Diagram for the project frontend

The human interaction with this project is done via the web interface. For this reason, a Website has to be developed to run on the web server with all the features necessary to use and/or administer the project. The diagrams below illustrate how the website frontend for the project will work and the flow of processes.

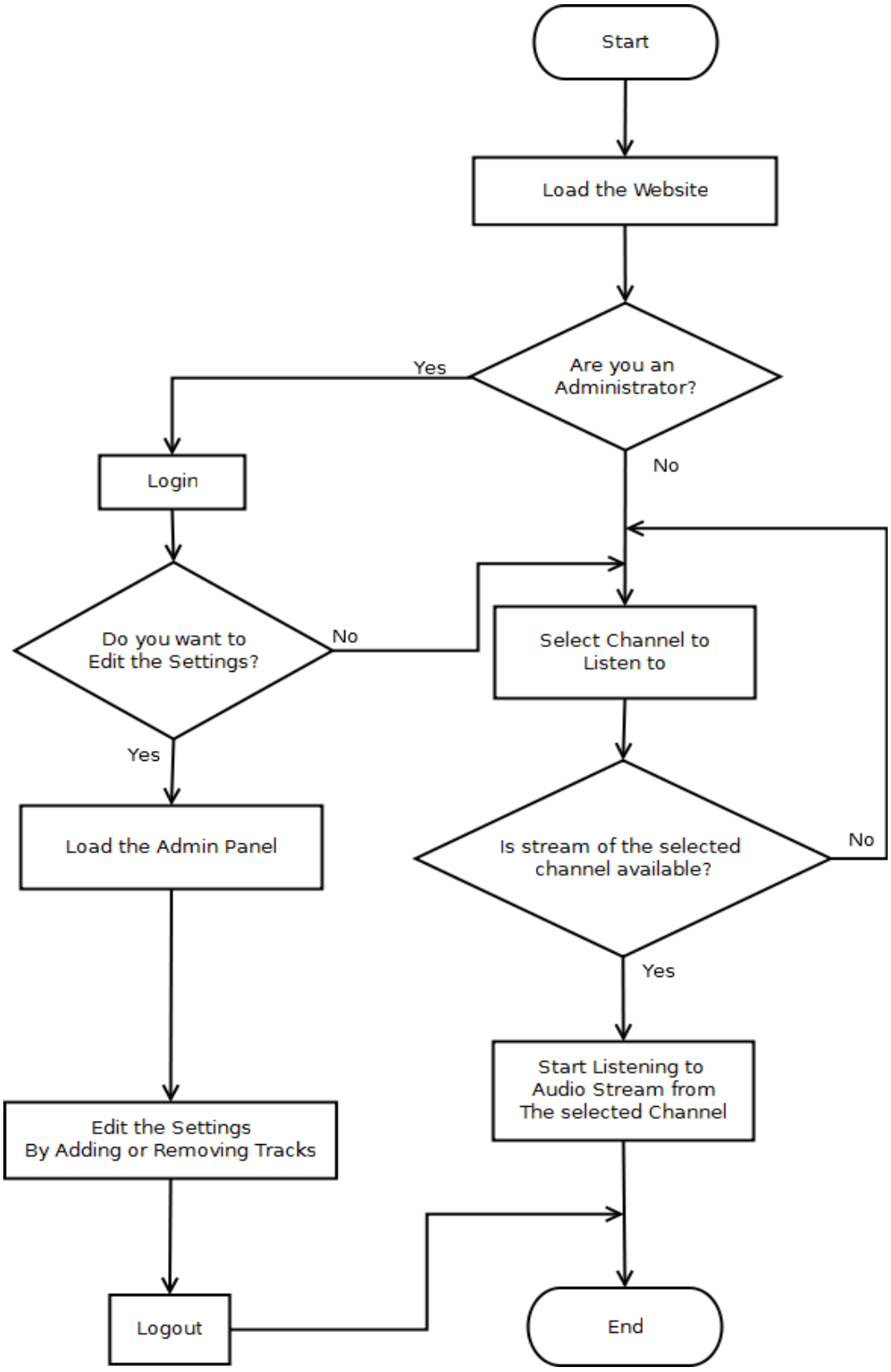


Fig 3.3: Frontend Flowchart

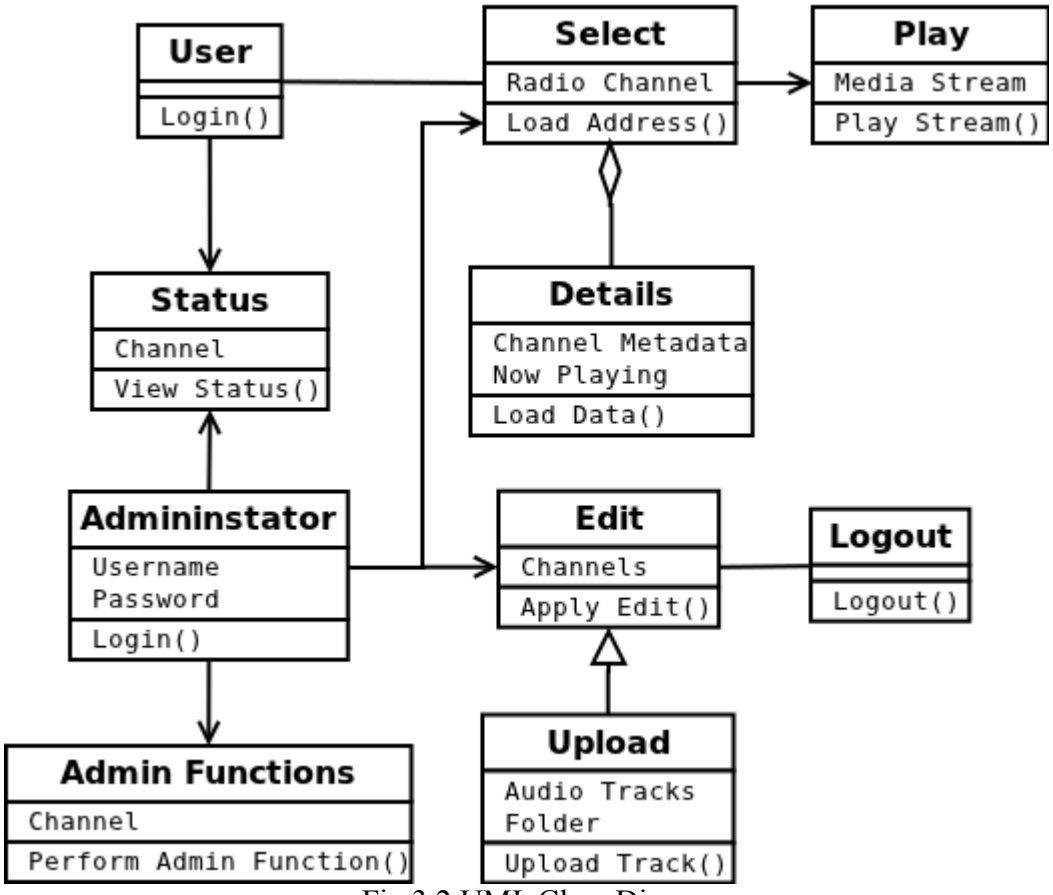


Fig 3.2 UML Class Diagram

MEDIA STREAM SOURCE CLIENT

A media Stream Source Client is a script which serves as the source of the data which is streamed by the server. It accesses the files from the directories specified in the script and also controls other functions such as specifying the number of streams, and outputting sound meta-data. The diagrams below show the processes involved in a media stream source client.

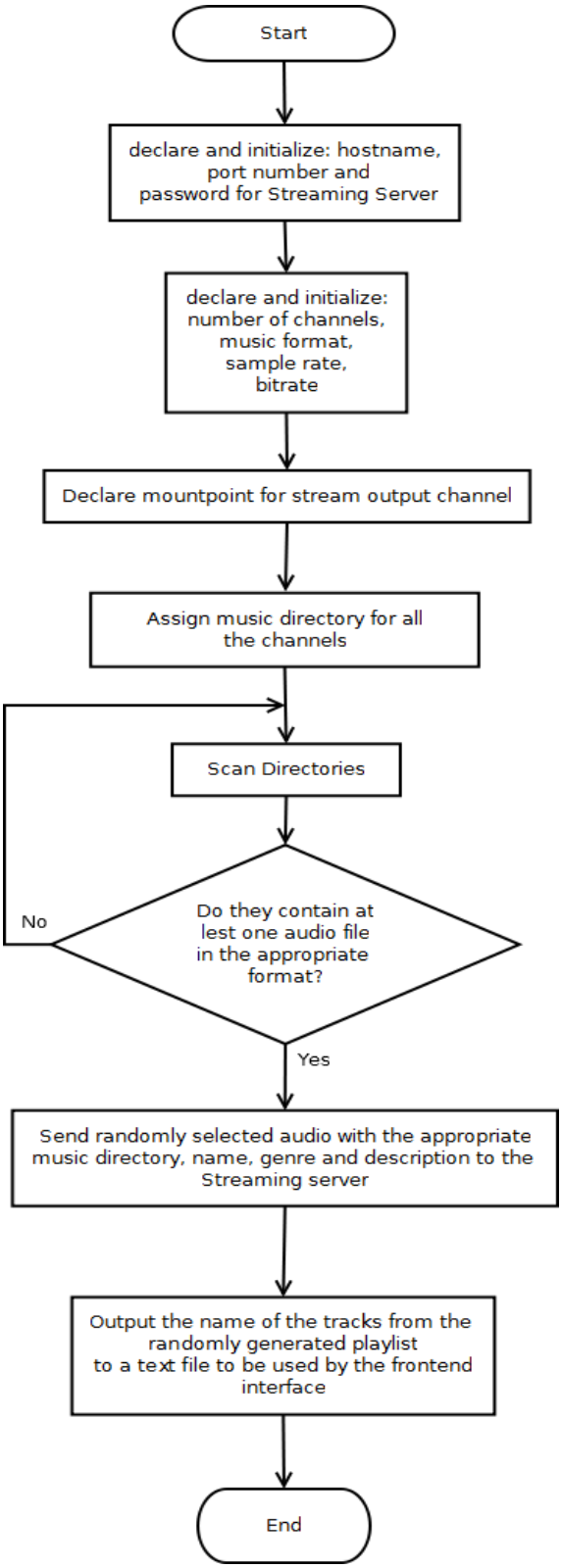


Fig 3.4: Stream Source Client Flowchart

SERVER

A single Server hardware is used. It serves as both the Web server and the streaming Server. Below is a Block diagrams which illustrates the server configuration.

THE STREAMING SERVER

The Function of the Streaming Server is to process the audio files into streams of data and output them continuously to be interpreted by a media player as audio. For this to mimic the behaviour of a conventional Radio, the data stream has to be continuous and cannot be influenced by the user or listener. The Streaming server receives its source from the media stream source client and processes the data it receives before broadcasting it. The streaming server broadcasts to all the mount points specified in the Stream source script and so, the different channels will be transmitting at once.

WEB SERVER

The Web Server is another important part of this project. The frontend Website which was discussed above is powered by the webserver. The Web Server uses the HTTP (Hypertext Markup Language) protocol by accepting HTTP requests from clients and then serving them usually a HTML (hypertext markup language) or XHTML document, although other file types are possible including images, documents, or other raw files. Most web servers also include logging, where sometimes detailed information about clients and client requests can be stored in log files, for later statistical analysis. A web server is either a computer program devoted to accepting HTTP requests, providing HTML/XHTML responses, and logging, or a computer that runs such a computer program. [18] Web servers are one of the primary components of the Internet and of

computer networking. Below is a flowchart which describes how the Web server works and the processes that occur.

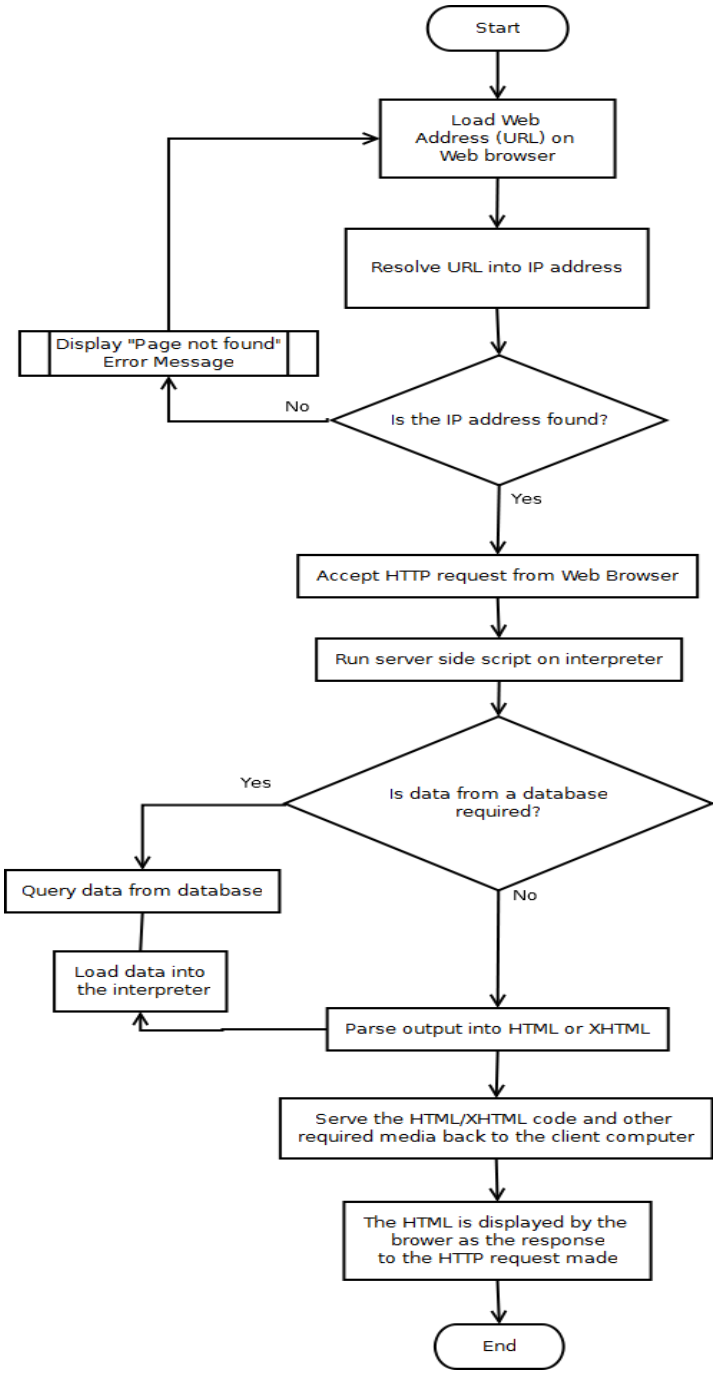


Fig 3.5: Webserver Flowchart

In order for this project to be completed successfully, all the components stated above must be implemented successfully. All of them must interact with each other to make a complete system.

OVERALL PROJECT DESIGN

The Intranet Radio, which is called CU Web Radio, is to be hosted on a website which will redirect to the URL (Uniform Resource Locator) http://192.168.1.100. There is a possibility that the URL can be configured to redirect to a web address similar to those found on the internet. However, that will require the existence of a DNS server. This URL will be configured from the DNS server to ensure that every user who connects to the appropriate network has access to the CU Web Radio by entering that address in their browser.

3.4 CU WEB RADIO CHANNELS

The Web Radio three major of several Channels. This is done in order to separate the content available and avail the users the opportunity of listening to different types of streams. The three major channels are described in detail below

3.4.1 GOSPEL MUSIC CHANNEL

This Channel features the best of the best in Gospel Music. Different genres exist ranging from Classical Music, Contemporary Gospel and even Gospel Rap and Hip-Hop. It is the perfect channel to tune into when you want to cool off after a long day's work while listening to edifying tunes. It is also perfect to put one into the mood to hear the word for the day from God. Some of the artistes featuring on the channel include

Don Moen, Kirk Franklin, Deitrick Haddon, Cece Winans, Sinachi, Frank Edwards Eben, etc.

This channel is further subdivided into individual channels for the different streams as follows

i. Gospel contemporary music channel

ii. Gospel classical music channel

iii. Gospel indigenous music channel

iv. Gospel rock music channel

v. Gospel rap/hip-hop music channel

3.4.2 ANNOUNCEMENTS CHANNEL

The Announcements Channel runs advertisements, relevant announcements from management, staff, and even students.

3.4.3 MESSAGES CHANNEL

The Messages Channel features different men of God expand on his word and give the message for the day, and season.
The men of God ministering on this channel include Bishop David Oyedepo, Pastor Adeboye, Bishop T.D Jakes, Pastor Chris Oyakilome, Bishop Abioye, etc.

All these channels will have jingles which intermittently play to remind the listener that they are listening to CU Web Radio.

The Website to be designed for CU Web Radio features pages for all the channels, as well as a status page to view the status of the channel status, an Admin page to view number of people connected, and even remove listeners.

There will also be the option of adding/ removing Tracks from the different channels via FTP (File Transfer Protocol).

The channels will continue running for as long as the server is on. The source client script will be configured such that the audio tracks are selected at random, and the jingles are played after every 3 tracks.

3.5 CONCLUSION

This chapter outlined the design employed in the implementation of this project. This design involves a lot of individual components which have to be interfaced properly in order for the outcome to be as required.

The frontend design is done to complement the backend and expose features of the project to the user as easily as possible.

For the backend, the best tool for the job is used in every instance in order to obtain a successful and sustainable outcome when implemented.

CHAPTER FOUR

SYSTEM IMPLEMENTATION AND TESTING

4.1 INTRODUCTION

This Chapter deals with the actual steps taken to actualize this project. Intricate details of the Design process will be explored in the Chapter. Also, after explaining the steps taken to implement this project, the steps taken to test it also have to be explored in order to satisfy the curiosity of the readers wanting to know if the project works; or wanting to duplicate it themselves.

4.2 IMPLEMENTATION

In order to implement a project such as this, everything has to be done in modules. Therefore, once again, I will split the process into hardware and Software parts. Eventually, both parts will be merged in order to have the completed project.

4.2.1 HARDWARE IMPLEMENTATION

True to the specifications for the Server computer given in Chapter 2, a Server computer was purchased for the Intranet Radio project with the following specifications.

i. Intel Pentium 4 Celeron Processor

ii. 2GB or RAM

iii. 40GB of Harddisk

iv. A 100Mbps Fast Ethernet NIC (Network Interface Card)

v. Dell Optiplex GX280 Series Model

Also, a Linksys-Cisco 120N Wireless Router was purchased. It features 802.11n technology with speeds of up to 108Mbps.

Table 4.1 Bill of Quantities

S/N	DESCRIPTION	QUANTITY	UNIT PRICE(₦)	TOTAL(₦)
1.	Dell OptiPlex 280 Series Desktop Computer	1	23,000	23,000
2.	Cisco Linksys WRT120N Wireless Access Point	1	10,000	10,000
	Total			33,000

SETTING UP THE HARDWARE

Setting up the server computer involves connecting all the individual components together and installation of the Operating System. This is a very straight forward process and there is no need to explain much on this.

Setting up the Router however, is a little bit more tasking. Certain important settings have to be made on the router to ensure that it is secure, as well as configured properly for the project at hand.

Since the router is meant to distribute a wireless network for the clients to connect to in order to access the web radio, it has to have an SSID (System Set Identity) that is unique and descriptive in order for the clients to know it is the right network to connect to. For this project, the router's SSID is configured to “CUWebRadio”.

Some other settings carried out on the server are:

i. **DNS IP address**: In this case "192.168.1.100" because the Server Computer is also used as a DNS server

ii. **DHCP settings**: In this case, it is enabled so as to give the router permission to assign IP addresses to all the hosts that connect within a certain predefined range.

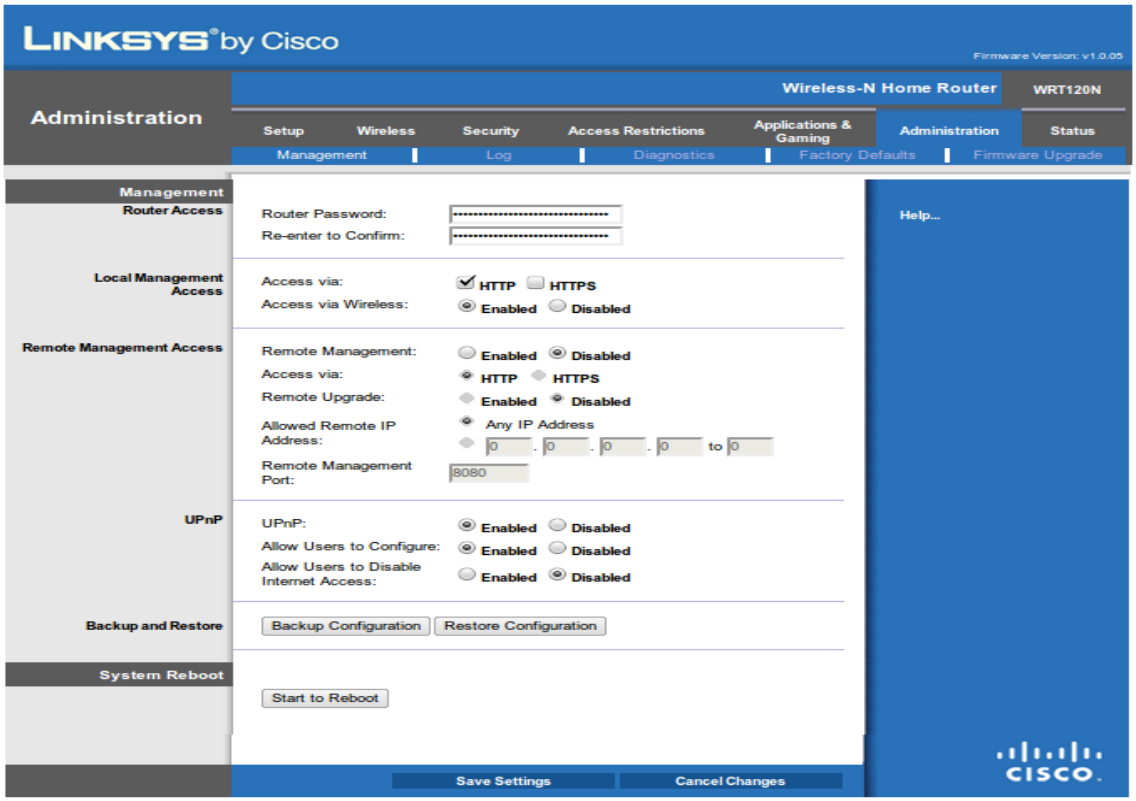
iii. **SSID encrypted key**: An encrypted key using the WEP encryption is used. The WEP is not the most secure encryption method available for Wi-Fi networks; however, it is the most compatible with most Wi-Fi enabled devices. For this project, the network is left open. This ensures that users do not need to know any pass key to connect to the network.

iv. **Router Web based settings security**: All routers have a web based interface where the settings can be made. However, by default, they are all configured to have very simple username and password to login into this administration panel. It is therefore a security risk to leave these settings the way they are when deploying a network using the router because this leaves your administration panel venerable to anyone with technical knowhow. It is therefore imperative to change the username and password to something less easily guessed.

v. **DHCP IP Address Range**: Since all the users need to be connected to the network to be able to access the network, they also need to have the same range of IP addresses. To ensure that this is done, a range of IP addresses to be assigned to clients can be configured on the Router. For this project, the range is between 192.168.1.100 and 192.168.1.255.

vi. **Assigning a Static IP address to a MAC Address**: MAC Address is a way of identifying computers. All computers have a fixed MAC address. Some computers, such as the servers, need to have a static IP address irrespective of the DHCP server settings. Therefore, this IP address can be assigned permanently to that MAC address on the Router [19].

Fig



4.2: The Linksys WRT120N Router Basic Setup Page

4.2.2 SOFTWARE IMPLEMENTATION

The software components of this project are many and different. Therefore, to effectively describe the implementation of all the software aspects of this project, they will have to be split into categories.

INSTALLATIONS

The Server computer is useless without the Operating System running on it. The Operating System has to be installed and configured for the purpose of deploying the CU Web Radio.

Firstly, the Fedora 14 Linux i386 Operating System has to be installed. Fedora is a fast, stable, and powerful operating system for everyday use built by a worldwide community of friends. It's completely free to use, study, and share [20].

INSTALLING THE SERVER OS

To install the server OS, the install media (DVD) has to be booted and then, the instruction following will have to be adhered to. Although the installation process is straightforward as previously acknowledged, it is important to highlight some of the decisions made during the installation.

Firstly, since the streaming server also requires a web server, I ensured that during installation, all the components pertaining to Web Server, such as Bind9 DNS server, MySQL database server, Apache HTTP server, etc. were properly installed. Also, the GNOME Desktop Graphical User Interface (GUI) was installed in order for there to be easy interaction with the server without always needing to resort to commands.

Finally, after the installation has been completed, the streaming server, icecast2, has to be installed. This is done by downloading the appropriate files from the Internet and installing it appropriately.

INSTALLING THE STREAMING SERVER

The Icecast Streaming Server is an integral part of this project. It is not pre-installed along with the Server Operating System. Therefore, it has to be installed manually. On Linux Operating Systems, applications are stored on repository servers on the Internet. The fastest way to download and install them is by running a command on the command prompt. Therefore, in order to install the Icecast streaming server, the following command was used.

sudo yum install icecast

The Source Client for the Icecast Server is a script written in Python Programming language. The Linux Server Operating System already has a Python Interpreter per-installed. However, the Icecast Streaming Server needs a library to interface with the Python script. This also has to be installed and using a similar method to the one above, we use the following command.

sudo yum install python-shout

Once all these components have been Installed, the Server is then configured.

4.3 CONFIGURATIONS

Several configurations have to be made. These range from Apache server configuration to Icecast Server configuration, to DNS configuration.

On the Linux Server Operating System, all the configuration is done using plain text and sometimes XML(Extensible Markup Language) files. These files are situated in various locations on the file system depending on the configuration being made.

The first configuration made on the server is the Web Server configuration. The Configuration file for the Apache server is found in "*/etc/httpd/conf/httpd.conf"* on the server harddisk. It contains configuration options for the Web Server such as the Hostname, availability of virtual hosts, location of the root folder where the website files will be, etc.

Another equally important configuration to be made is for the Icecast Server. The configuration file for the Icecast server can be found in *"/etc/icecast.xml"*. This file contains settings such as the hostname of the server, the DNS address, the number of streams to be broadcast at once, the maximum number of clients that can be connected to server at any given point in time, the listening port address, etc.

After all the configurations have been completed, the services installed (httpd for Apache, icecast, etc.) have to be configured to start automatically along with the startup of the computer. This is achieved by using the following command in the Linux command prompt

```
chroot --add icecast
```

Once all this had been done, the next step was to develop and host the website for the project.

4.4 WEBSITE DESIGN

The website uses HTML, CSS and Jquery (Javascript) technology. It was built to be visually appealing and functional at the same time. The web design tools used to design the website are Photoshop and Dreamweaver. The website was designed to highlight all

the features of the Web radio, as well as enable some other features for the end users. The Figure below is the tree diagram of the website.

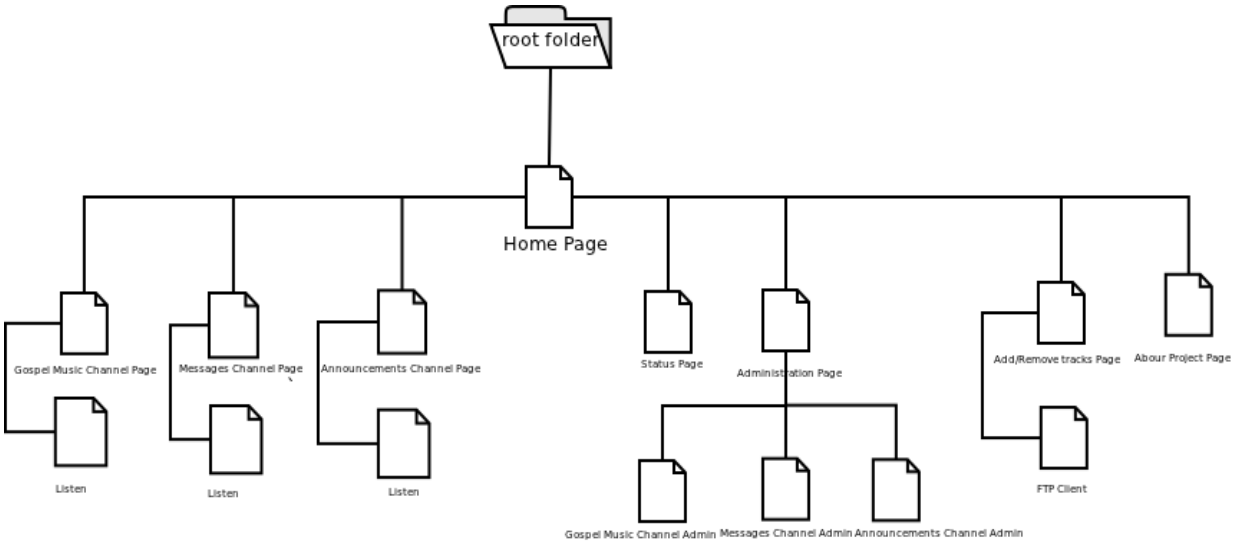


Fig 4.7: CU Web Radio Website Hierarchy

The Website consists of several pages which can either be for accessing the web Radio or administering it. The design was made to be simple not to confuse the users and to present the services the site provides to the user without him having to navigate to several pages.

Below are some screenshots from the Website after development.

HOME PAGE

The home page is the first page that the user encounters when loading the project. It features links to all the other pages.

Fig 4.8: CU Web Radio Homepage

STATUS PAGE

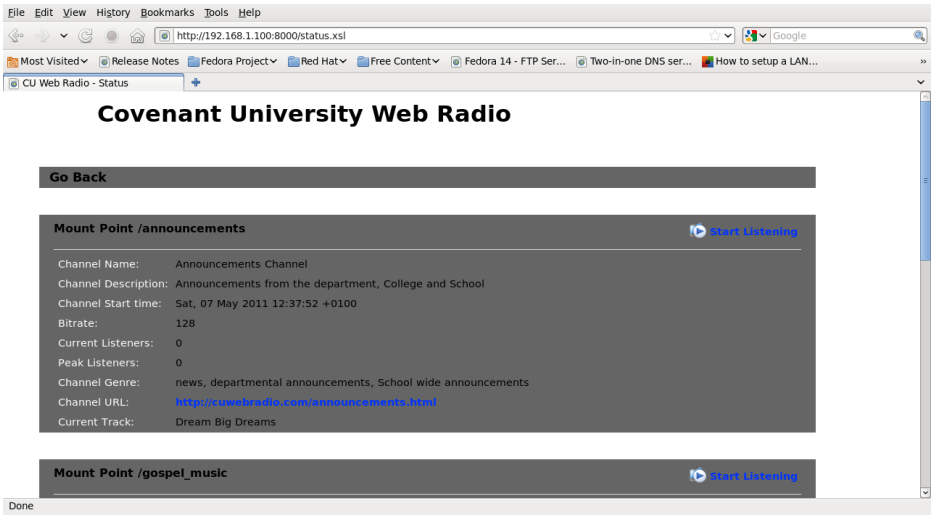


Fig 4.9: CU Web Radio "Status" Page

The status page shows the state of all the channels with features such as the name of the channel, the number of clients connected, the URL of the channel, etc.

Other pages on the Website include:

i. Administration Panel Page

ii. Add Tracks page

iii. About Project Page

iv. Channels Page

4.5 DEVELOPMENT OF THE ICECAST SOURCE CLIENT SCRIPT

The Website is just a front-end; an interface which the user accesses in order to listen to the Web Radio. The major work is done on the back-end with the Streaming and Web Servers. The Streaming server, just like the web server, cannot function except it has been properly configured. Having already properly configured the Streaming server, the next step to take in order to get it working is to develop a “Source Client”. A Source Client is an application which creates the number of streams which the server will have, serve the files to the server to stream, as well as other features which can be added as the programmer so sees fit.

The Source Client used for this project was developed with the Python[21] Programming language and is named “icecastscript.py”. It uses the “Python-shout” library which integrates the python programming language with the Icecast server.

The features it possesses are:

i. Starting multiple streams simultaneously

ii. creating mount points for the streams

iii. Scanning specified folders for audio tracks for the three channels and sending them to the server in a random order

iv. Interrupting the normal stream after a specified number of tracks by adding a jingle in between. The number of tracks is set at 3 but can be modified as seen fit.

v. Output a specified number of future tracks into a text file.

vi. After the development of the source client, it has to be configured to reflect the settings of the icecast.xml file so that the script can successfully connect to the server.

The source client, being a python script, has to have the python interpreter installed on the server for it to work.

All Linux Operating Systems already have the python interpreter installed on them though, so there is no need to worry about this. In order to run the script, the following command has to be executed on the command prompt.

```
python icecastscript.py
```

STARTING THE PROJECT FOR TESTING

After all the installations and configurations have been completed, the next step is to start the server and the applications, so that the project can be effectively tested to ensure that it is working according to specifications.

i. To start the project, the following steps should be taken

ii. Ensure that the Server Computer and

iii. Boot the server computer

iv. Start the services (httpd, icecast, dns,etc,). All these services should already be running already

v. Start the source client (this can be configured to start automatically also.

4.6 TESTING

The testing phase is a very important phase in the project work. Here, we confirm that the project is working as we expect it to and as it is supposed to.

The testing of this project was done in two phases. This is done by first testing the project locally, and then remotely.

TESTING LOCALLY

Testing this project locally involves testing it out on the server where all the components are installed. Since this is a web based project, to access the project means that it has to be loaded on a web browser. Therefore, the following steps were taken to test the project Locally,

i. Open the browser

ii. Load the website by going to the localhost address (http://localhost)

iii. After the site loads, click on the links to confirm that they are active

iv. Select a particular channel and open its page

v. Click on the link to start listening to the channel and wait for the stream to load

vi. Confirm that the stream has loaded when you start hearing the sound from that particular channel.

REMOTE TESTING

The remote testing involves testing out the project over a network. This means that, although the project might have been confirmed to be working on the server which it was configured on, it is only successful if it also works over a network such that, client computers and other mobile devices which have Wi-Fi capabilities and can play mp3 streams can also access the web Radio. To confirm this, a laptop computer and an Apple iPod touch device was used.

The steps taken for each are as follows:

i. ensure that the server is running

ii. connect the device to the network provided by the wireless router

iii. open the device's web browser

iv. load the appropriate web or IP address (in this case, the IP address to load on the browser is http://192.168.1.100. Alternatively, a correctly configured DNS server will resolve that IP address to http://webradio.cu.)

v. confirm that the website has loaded and then test out the links to confirm that they are all working

vi. Load the page of one of the channels

vii. Click on the link to start streaming from one of the channels and wait to ensure that the stream starts.

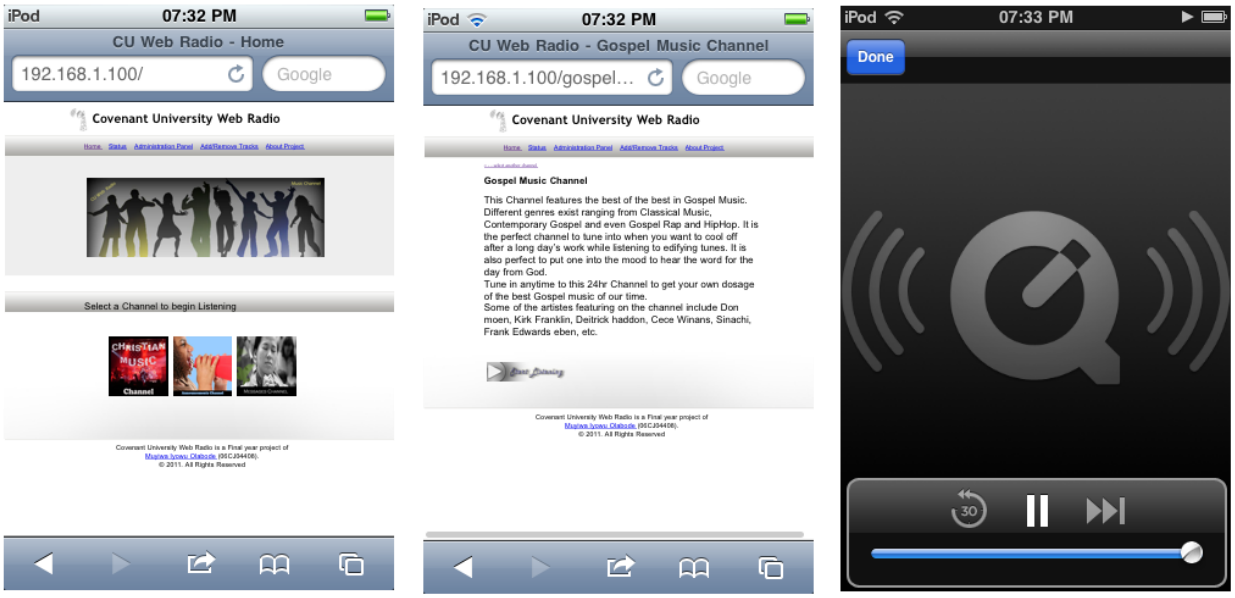


Fig 4.18: Screenshot of an iPod Touch with the project running over the network using the web browser and streaming with the Quicktime browser Plugin.

After carrying out these tests successfully, it was concluded that the project is running successfully and smoothly as expected.

CHAPTER FIVE

CONCLUSION

5.1 INTRODUCTION

This Project aimed at creating a Web Radio Transmitter making use of the Intranet network available at Covenant University. Regardless of the challenges encountered, it is safe to say that this goal was actualized. Making effective use of existing technologies and augmenting them with new ideas, this project has been successfully executed, thereby presenting Covenant University with a Web Radio that can be integrated into the existing infrastructure as is or extended to enable additional functionality, thereby making it more robust and useful.

5.2 CHALLENGES ENCOUNTERED

i. Several challenges were encountered during the execution of this project and some of them are highlighted below

ii. The server applications used are difficult to configure and information on them are not always very explicit.

iii. The Source client script had to be written in Python programming language and so, I had to learn the language to a reasonable extent

iv. The time available for carrying out the project compared to the amount of knowledge to be acquired was not commensurate. This meant that not all the features that might have been desirable were actualized.

5.3 ACHIEVEMENTS

Despite the challenges encountered, the project was still carried out with a high degree of success. The Web Radio (called CU Web Radio) was designed and implemented, and it was functional. It featured Channels streaming Music, Announcements and Messages. It also featured an administration panel with capabilities such as deleting a user and seeing the total number of users connected. The ability for administrators to upload tracks to the server was also implemented.

As a result, this approach greatly reduces the cost of entry into the Radio service for a society such as Covenant University.

5.4 RECOMMENDATIONS

This project, I believe, will be helpful to a society such as ours. It will therefore be a waste to let it go as just another final year project. Steps can be taken to integrate it into Covenant University Intranet Network, thereby enabling this feature for as many as would want it.

Enhancements can be made to this project. Other features such as live recording and track scheduling can be added, to make it more like a complete Radio station.

All these and more are recommended on the future of this project. The whole process of this report has exposed to me several previously unknown things concerning the workings of the Computer Operating System, Audio Streaming, audio encoding and decoding, data transfer, network protocols, amongst others. It has been a worthwhile learning process and the satisfaction of a working product makes it all the more worth it.