GHANA COMMUNICATION TECHNOLOGY UNIVERSITY (GCTU)

FACULTY OF ENGINEERING

DEPARTMENT OF TELECOMMUNICATIONS ENGINEERING

Topic:

DESIGN AND IMPLEMENTATION OF A DESKTOP BASED SYSTEM TO ENHANCE TEACHING USING SCREEN SHARING TECHNOLOGY THROUGH WLAN USING GCTU AS A CASE STUDY

A Project Work Submitted in Partial Fulfillment of the Requirements For

BSc. in Telecommunication Engineering

BY:

ADABOGO EMMANUEL - 040918291

AMEDOADZI GERTRUDE FAFALI - 040119090

SUPERVISOR:

DR SOLOMON NSOR ANABIAH

APRIL ,2022

DECLARATION

This project is presented as part of the requirements for BSc. in Telecommunication Engineering awarded by Ghana Technology University College. I hereby declare that this project is entirely the result of hard work, research, and inquiries. I am confident that this project work is not copied from any other person. All sources of information have however been acknowledged with due respect.

AUTHORS

ADABOGO EMMANUEL SIGNATURE: ....

STUDENT ID: 040918291 DATE: ……………………………

AMEDOADZI GERTRUDE FAFALI SIGNATURE............................

STUDENT ID: 040119090 DATE: …………………………

SUPERVISOR

DR SOLOMON NSOR ANABIAH SIGNATURE...................................

DATE: …………………………

HEAD OF DEPARTMENT

MR ISAAC HANSON SIGNATURE...................................

DATE: ……………………………

ABSTRACT

Virtual screensharing Application such as Zoom, TeamViewer etc. comes with a monthly subscription plan aside the internet data cost which is on the rise since last year. Lecturers and students also often waste valuable time trying to fix projector technical issues, aside issues involving projector damage, which usually affect display screens with dead pixel and rainbow effects which affect the viewing angles in classrooms. All these interferes with smooth projection of screens. Moreover, Students in a separate room need new projectors installed. In this project work, we used the webRTC technology to design a desktop-based system to enable both lecturers share screen to student within and around the university campus. JavaScript is the main languages used in building the backend part of the system (Server). HTML and CSS were used in building the user interface for both Server and Client. The average video/ audio streaming delay was recorded less than 300ms. The system could connect up to a maximum of 26 participants within a single 4G WIFI interface. Compared to the maximum number of 20 participants excluding audio that the systems in the review could connect, our system is scalable enough. In terms of functionality, the system in this project has extra functionality such as messaging and filesharing systems as well audio streaming capabilities. 3 people were able to connect with optimum efficiency and graphic quality through the 4x4 Multiple Input Multiple Output (MIMO) routers across campus. The project was tested by multiple participants and it yielded a satisfactory result.

TABLE OF CONTENT

LIST OF FIGURES

ACKNOWLEDGEMENT

.

CHAPTER ONE

INTRODUCTION

1.1 Background of Study

Real-time screen-sharing technology has gone through a series of development and improvement throughout the past years. This provides many universities in Ghana and across the globe with access to remote teaching and learning (online learning) through client communication. Ben Mulhollad (2021) concluded in his research article that the best and popular screen sharing application and presentation software application now are Zoom, Google Meet, TeamViewer and etc . These applications come with a cost even for users within the university campus. It is also observed that, lecturers and students waste valuable time on technical issues having to deal with projector connectors. According to Hittachi Digital Media Group (2016), taking notes more effectively in various teaching ways such creating learning games, lecturing using varied media, and effectively utilizing class hours are just a some of the advantages of utilizing a wireless projector.

Wireless Local Area Network (WLAN) is a computer network that connects more than one computer devices together using electromagnetic waves as a medium of transmission within a small geographical area such as a university campus. WLAN is now a common and important resource for end-users. Examples are people with laptops who use WLAN to minimize the use of the more expensive 3G and 4G connection. In this regard, it is very important that WLAN is utilized within our university to provide lecturers and students with free and accessible screen-sharing system to aid in the enhancement of teaching within classrooms and across campuses. This would also give lecturers and students an alternative to share screen using computers when projectors or connectors malfunction. In D. Skvorc et al. (2014) research work, their analysis clarified that latency and network traffic are the main generated criteria that negatively affect network system design. But with the current standard, IEEE 802.11n Wireless Fidelity (WiFi) with multiple antennas across campuses, the established WLAN can provide a data speed of up to 600 Mb/s. Traffic scheduling is a crucial part of any Access Points (APs) using IEEE 802.11n to ensure that data is provided in accordance with user needs.

To further ensure scalability by maximizing the number of students connected to a lecture session, the researchers are employing a frame comparison and deletion algorithm to ensure no multiple frames are sent over the network. Patil, Shweta et al, (2015) concluded that an average of 24 frames per second is good for a normal video streaming but employing the frame comparison and deletion algorithm, we could reduce this frame buffering to up zero or one frame per second with a smooth presentation delivery. This is possible because displayed slides are only sent once across the network. This would further minimize network traffic. This project would be implementing a system that would help enhance teaching and learning on campus.

1.2 Problem Statement

There are so many virtual classroom applications available. Some of which are Google meet, Zoom, TeamViewer, etc. These applications come with a monthly subscription plan. According to a Publication by FutureLearn, (2021), an average of £46 is spent on monthly subscriptions per a lecturer and this is predicted to increase in coming years. Students also face a similar challenge as the 5 gigabyte data offered to students in the previous months could only take them less than a week to carry on research activity and joining online classes.

Also, it is observed that lecturers and students often waste a lot of time trying to fix projector technical issues. These technical challenges include, Virtual Graphics Adaptor (VGA) connector failures, VGA to HDMI connector failures and also projector malfunctioning in certain classrooms. All these interfere with smooth projection of screens.

Another issue is projector damage, which usually affects display screens with dead pixels and rainbow effects. In addition to issues like blurry or shaded images from the projector, or because the bulb life is longer than the mean time between failure (MTBF). These types of projector issues are frequently seen in locations with unstable power conditions or frequently occurring, unexpected blackouts. There are restrictions on the hardware projector's ability to see the projection screen. Therefore, the viewing angle has a significant impact as well. You need a projector with a better resolution if you want the audience to be able to see the screen clearly. A narrow viewing angle from the screen projector prevents the audience on the sides from fully enjoying the presentation since projectors can only be used in certain locations or positions in the rooms. If the audience is in a separate room, new projectors need to be installed in each of those rooms.

1.3 Research Aim and Objectives

1.3.1 Research aim

The main objective is to design and implement a screen sharing software application using the school WLAN as a transmission medium.

1.3.2 Specific objectives:

1. Implement a real-time half-duplex video sharing system from lecturer to students

2. Implement a real-time full-duplex audio sharing mechanism to ensure discussion between lecturers and students

1.4 Significance of Study

According to research conducted by Jakhar et al, (2020), the result of the 2020 pandemic has undoubtedly shifted educational systems. Schools around the world have hurried to digitally transform themselves and be able to respond to the demands of remote education. However, now that schools are becoming stable and back to their normal routine, successfully completing this project would improve the learning procedures on campus without having to deal with monthly virtual class application subscriptions. According to CISCO, (2017), video IP traffic is expected to grow around 82% of the world internet traffic before the end of 2022. Also, Morris, C. et al, (2014) concluded in their findings that online screen sharing is now an essential means for teaching and learning in tertiary education. The successful completion of this wireless screen sharing system would give our school another alternative to sharing screens during lectures and presentation tutorials. Teachers are no longer needed to spend valuable class time juggling cables and mismatched adapters. The Information Technology (IT) department will also greatly benefit from this. Since maintenance and support would become easier. In a sense, they would have lesser schedules than usual on campus. Micheal S. et al, (2022) also conducted a research study to investigate the influence of screen-sharing technologies and finally concluded that, lecturers responded that a high mobility and an improvement in ease of content sharing with deeper leaner cognition as a factor to improve learning and teaching with screen sharing technologies.

It doesn't matter if you are delivering a PowerPoint presentation, a word document, or guiding a step-by-step tutorial for computer-based tasks within the school campus. successfully completing this project would be the perfect tool. This project is budget friendly and an effective way to enhance a student learning environment.

1.5 Scope and Limitation

1.5.1 The Scope of this project work

The system will be used on Ghana Technology University College campus for distance presentation

1.5.2 Limitation of this project work

1. The system does not scale well on actual real time video streaming. Which is to say it is limited to only presentation based learning. This is because the system operates based on frame comparison algorithms and limits scalability.

1.6 Brief Methodology

1.6.1 Design concept

The project employs the waterfall software development module. Following these steps is a linear sequential Software Development Life Cycle (SDLC) paradigm. Analysis, Design, Implementation, Testing, and Maintenance are requirements. Manzor Ahmad et al, (2015) concluded that this module is recommended for strict deadlines and projects with a good design process.

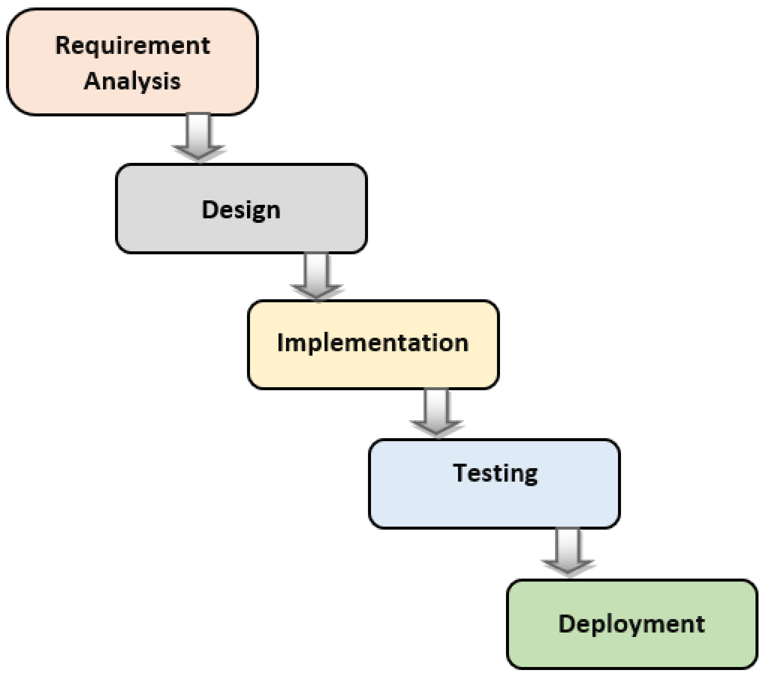

Figure 1.1 Waterfall Development Module

1.6.2 Software required:

- Microsoft visual studio Professional 2019 as server site Integrated Development Environment (IDE)
- Visual Studio Code as client site IDE

1.6.3 Frameworks and Libraries required:

- Web Real Time Communication (WebRTC) technology
- Socket.io
- Express.js

1.6.4 Hardware requirement:

- Intel Core i5-4234U 2.4GHz Processor computer
- Random Access Memory (RAM) of 8 Gigabyte
- Hard Disk Drive (HDD) of 500 Gigabyte Capacity

1.7 Organization of Project

The layout of the project is as follows; The introduction, study background, issue description, major and secondary objectives, importance of the study, scope and limits, methodology, and project organization are all covered in Chapter 1. The literature review is covered in Chapter 2 to provide analysis of related and recent research works. In Chapter 3, the project's methodology is covered in great depth. The short technique from chapter 1 is expanded upon here. Results and analysis, or the evaluation and analysis of results are topics covered in Chapter 4. Chapter 5 serves as the epilogue. In other words, it concludes the entire work by briefly describing the project's restrictions, suggestions, and references.

CHAPTER TWO

LITERATURE REVIEW

2.1 Overview

This chapter examines pertinent related literature in light of already published works in the study's chosen field. It outlined the project and provided a thorough analysis of all its components. The chapter will go into further detail on the theories and methods used in other research works. It entails an evaluation of pertinent project efforts on WLAN, audio processing, video streaming, and data routing needs and applications. Reviews in this chapter are drawn from online resources and journal publications.

2.2 Current Issues of Concern

Projectors in our classrooms are fully connected to computers with cables to enable screen sharing during teaching or class presentation. These wired connections use old connectors like the Video Graphics Array (VGA). A typical computer video output connector is the Visual Graphics Array (VGA) connection. Othman, Fazzami et al. (2019) mentioned in their work that the 15-pin connection was initially introduced in 1987 with the International Business Machines Personal Computers (IBM PS/2) and its VGA graphics systems. Since then, it has become a standard for PCs, projectors, high-definition television sets and monitors. However, new generation computers do not include this port for screen sharing which makes it difficult for some lecturers to share their screens with students. Though using VGA to High-Definition Multimedia Interface (HDMI) converters are a solution to this problem, these converters sometimes get faulty making both lecturers and students waste valuable time trying to get fully functioning converters. With this problem Wireless screen sharing systems are an important adaptation to provide an alternative to efficiently sharing screens from lecturers to students without having to worry about projectors or adaptors malfunctioning. This system would also ensure a perfect compatibility with our new generation computers.

2.3 Definitions of Related Areas

This section reviews definitions that are closely related to the research topic, they would help bring clarity and further understanding to the project.

2.3.1 Screen Sharing

A publication by Teamviewer, (2022) defines screen sharing as a process of sharing the display of one's computer screen with another or several computer devices, also referred to as desktop sharing. This can encompass all of the components on a screen or just one window, giving you total control over your desktop's exposure and ensuring your privacy. By using TeamViewer to share your screen, you may show any material on your device to friends, colleagues, or clients without the need to upload files such as presentations, documents, photographs, and even films. Furthermore, this screen sharing system allows connected receivers to not only view the content on the shared device, but also to observe as the user navigates the interface and makes changes in real time.

2.3.2 WLAN (Wireless Local Area Network)

A WLAN (wireless local area network) is a type of network that allows for wireless device connectivity and communication. WLAN devices communicate using Wi-Fi as opposed to a traditional wired LAN (Local Area Network), which connects devices using Ethernet connections. While a WLAN looks different from a traditional LAN, it functions identically. Dynamic Host Configuration Protocol (DHCP) is used to normally connect and set up new devices. Keerio Imran et al, (2018) added that both wired and wireless connections are available. Though in LAN medium fiber optic cable is the primary stakeholder for meeting bandwidth demands, it has mobility limitations. As a result, WLAN is used to meet bandwidth demands while simultaneously providing mobility to users. They can connect with other network devices in the same manner as wired devices. The fundamental distinction is in the manner in which data is conveyed. According to G. R. Hiertz et al, (2010), a LAN sends data in a sequence of Ethernet packets through physical connections. Packets are sent through electromagnetic waves over a WLAN.

2.3.3 RTC (Real Time Communication)

Alissa Irei (2020) defined Real-time communication (RTC) as a text, picture, audio, and video style of communication that allows all participants to share information instantaneously or with little delay. Because the number of internet users has continued to rise in recent years, RTC services are essential. D.Sivasankar et al, (2017) also stated in a research paper that, the amount of data transferred over the network is increasing, resulting in an increase in data traffic at the same time. Hence, more flexibility and the capacity to respond and adjust performance at runtime are required by the RTC system.

2.3.4 RDP (Remote Desktop Protocol)

Microsoft (2020) defines RDP as Windows-based programs operating on a server, the Microsoft Remote Desktop Protocol (RDP) offers remote display and input capabilities across network connections. Multiple LAN protocols and various network topologies are supported by RDP. According to Wesley and Brien (2022), Remote Desktop Protocol (RDP) is a secured network communications protocol developed by Microsoft. It gives users remote access to their physical workstation PCs and enables network managers to remotely troubleshoot specific user issues.

2.3.5 WebSocket

K Singh and V. Krishnas, (2013) defined WebSock as an implemented system which is used in web-based and platform development applications using already-built communication widgets such as socket.io.

2.4 REVIEW OF RELATED WORKS

This section discusses a review of some studies of other authors that are closely related to this work. The section briefly describes the working of the systems in these related works, the methodology the authors use as well as the strength and weaknesses of these related works.

2.4.1 Reviewed Work One

Design and implementation of real-time wireless projection system based on ARM embedded system. By; (Zhaohua Long, Hao Tang, Junhua Huang. 2018)

In this paper, the authors discussed that long and many classic audio-visual products are changing over time but traditional projectors are still the most used screen sharing equipment in conferences and other educational gatherings. They added that these projectors failed to adopt and implement wireless network technologies although they are widely utilized in education, and other conference meetings. Project usage is only limited to cable connections. As a result, a wired connection working in wireless mode has become a pressing issue. However, using wireless display technology is easy and convenient for a dependable high-quality product design. In regards to the issues mentioned above, the authors developed a wireless projection system and presented a weight-based frame deletion approach to improve video transmission quality to some extent.

Methodology Used

The authors divided the system into two parts. A server and a client as designed in a project by (P.Wu and Z.LONG, 2013). The server is designed to communicate with the client through a control message. The system client functions by taking screenshot data from the desktop and encoding these image data to reduce their sizes. The client system further transmits the image data through a wireless network to ensure smooth data transmission and. The server also works by receiving the encoded image data, decoding these encoded data and rendering them on a display device. The authors used an advanced encoding technology H.264 to design the system.

Advanced encoding technology H.264

Dong, Jie & Xin (2015), defined H.264 also known as MPEG-4 Part 10 as an advanced video coding algorithm that uses a block-oriented, motion-compensated video compression standard for video

Frame Deletion Strategy

The authors further deployed a frame deletion strategy to minimize the number of image frames sent over the network per a given time. This strategy uses a mean square error (MSE) algorithm to compare previous frames with current frames. If there is no error, the frame is discarded to avoid sending similar frames over the network. But if there is an error, the frame is sent over the network.

Shu-dong, L. et al, (2008) also mentions in his research that the weight-based algorithm enhances, reduces delay, service quality and optimizes real-time customer service systems.

Strength of the study

This research provided gatherings, conferences, and classrooms with the following benefits

1. The project provided an easy and reliable means of wireless screen sharing using a wireless network.
2. The provided weight-based frame deletion strategy in the system improved delay and Quality of Service (QoS) with higher resolution screen sharing.

Weakness of the study

1. The limitations of this project are that, although it provides a higher resolution screen sharing with minimized delay, The project does not scale well since it can only provide connection between two devices. Example, the system can only connect a computer to only a single screen or monitor.
2. The system did not include any audio transmission functionality

2.4.2 Review Work Two

Real-time screen sharing using web socket of presenting without projector. By; (Irfan Darmawan, Alam Rahmatulloh Rohmat Gunawan, 2019)

In this project, the authors discuss that Presentation is usually done with projectors connected to laptop devices or a Personal Computer (PC). On a certain screen or a wall painted white, participants view a presentation slide show. However, when the projectors' technology is unavailable, the presentation process may not go as smoothly as it could. Other issues, such as projector damage, dead pixels, and hue effects from the projector, frequently cause the show to be permanently disrupted. The authors further stated that one solution to difficulties presented in prior studies is to give a presentation without utilizing a projector. Presenters and personal computers, on the other hand, can still utilize a specific server to access those used by clients.

Methodology used

Hardware and software have been developed for screen sharing. Using a PC or laptop acting as a server, the approach enables presenters to share the presentation display via a wireless network. Using a smartphone or Tablet PC linked to the same wireless network as the server, participants can watch the presentation display as clients. Below is a common illustration of the system architecture. Three main pieces of hardware make up the system architecture: an access point for wireless network infrastructure support, a laptop or PC server for presenters, and smartphones or tablet computers for audience participants.

Strength of the Study

1. The WebSocket technology employed in this work has been shown to provide full-duplex communication, allowing for real-time presentation activities when using the screen sharing idea.
2. The system does not require internet connectivity, all that is required is to install and execute software and join to the same network.

Weakness Of the Study

1. Their client system is developed using the Kotlin programming language which is limited to only android support systems. In other words, their system does not support desktop-based computers such as laptops and windows cell phones.
2. No tests have been undertaken in this study to compare the most efficient image types. In order to obtain the screen sharing quality that is more ideal, efficient and swift, a more in-depth search on the sorts of frames that may be shared between the server and client using a Web Socket was not conducted in the system by the researchers.
3. The test result indicated that the system is limited to a maximum of 20 smart phones.
4. In this project the authors did not include audio transmission from the client to the server. This limits the class session to only a classroom

2.4.3 Related Work Three

On the performance comparisons of native and clientless real-time screen-sharing technologies. By; (Huang C.Y, Cheng Y.C   et al, 2021)

The authors in this project, investigated the performance of several screen-sharing solutions, which may be divided into two categories: native and clientless. Clientless ones operate directly in web browsers, whereas native ones require users to install special-purpose software. Extensive tests are carried out in three stages. Firstly, the authors choose a set of native and clientless screen-sharing technologies that are most representative. Secondly, they utilize several performance indicators. The authors further present a systematic measuring approach for evaluating screen-sharing solutions under various and dynamic network settings. Finally, the authors run a rigorous test and conduct in-depth research that determines the performance difference between clientless and native screen-sharing systems. The authors discovered that a WebRTC-based solution performs the best overall. It uses a maximum of 3 Mbps bandwidth while maintaining a high decoding ratio and providing good video quality.

Strength Of the Study

1. Based on the study the authors are able to determine that WebRTC performs best with 3 Mbps bandwidth among the clientless technologies.
2. Furthermore, under fluctuating network conditions, webRTC results in a consistent high decoding ratio and video quality.

Weakness Of the Study

1. The implemented system is only limited to video transmission without audio

2.4.4 Reviewed Work Four

Performance Analysis of Data Transmission on WebSocket for Real-time Communication by; (Alam Rahmatulloh, Irfan Darmawan, Rohmat Gunawan, 2019)

The authors in this review work, did a measurement-based study based on the magnitude of the data used (in bytes per second) between the server and client during the screen-sharing operation. This project's aim was to determine the efficient and lightweight image typed best suited for wireless transmission over the LAN or WAN using WebSocket in native application (without web browsers). Following experimental findings on picture file formats including RAW, JPEG, BMP, WEBP, and PNG. After running the Screen Sharing Server and Screen Sharing Client programs for 20 frame per seconds (fps), the authors came to the conclusion that Joint Photographic Expert Group (.JPG) is the image file format that has the greatest average frame value sent at ten frames per second but instead consumes an average of 590,675.9 (Kb) data with a support for Real Time Communication (RTC). On the other hand, the image file type (.WEBP) required the least amount of data when data transfer between the server and client began, 44,970.6 bytes (Kb).

Methodology used

The Screen Sharing Client Application, created in Kotlin, and the Screen Sharing Server Application, created in C++, are the two main programs on this system. The created technology enables real-time screen sharing over WebSocket. Only a standard laptop or PC linked to a wireless network will do for the presenter. Using smartphones wirelessly linked to the same network as the server, attendees may watch the presentation display as clients. This study's objective is to evaluate RTC data transmission performance utilizing WebSocket-based apps without access to the internet. Although WebSocket was utilized to build the screen-sharing system in this study, neither a web server nor a web client are required. The three operations that comprise capturing the PC server screen display, saving the results of the capture in an image file format, and sending the image file from the server to the client. The Screen Sharing Client program receives each frame picture delivered by the server when it reaches the client side and displays it on the client screen. The computer server display is captured, the results are saved in an image file format, and the image file is sent from the server to the client. When it reaches the client side through the network, the Screen Sharing Client application gets each frame picture sent by the server and displays it on the client screen using a browser.

Strength of study

1. The study shows that WEBP file format is more recommended on efficient systems with slower network speed to maximize data throughput.

2. The study also shows that JPEG has lesser encoding and decoding time but has bigger file sizes which may affect data throughput.

3. The implemented systems support both half-duplex or full-duplex mode of transmission.

Weakness of study

1. The study was only limited to Image data comparison. It did not include audio data format.

2.4.5 Related Work Five

A hardware-accelerated system for high resolution real-time screen sharing. By (Yang Siyu, Bin Li, You Song, Jizheng Xu & Yan Lu, 2018)

The authors describe a hardware-accelerated approach for high quality screen sharing in this work. To make processing and compressing frames less computationally demanding, inter-frame correlation is looked at in the context of real-time screen content sharing. They went further to provide a multiple codec strategy that sets up and chooses the right encoders for newly updated screen data. In order to reduce encoding complexity and improve latency performance, an alternative frame split option for metadata processing that divides tiny yet distant screen changes into distinct frames. In terms of encoding time for typical screen sharing scenarios, their evaluation shows that the multiple codec method performs better than the usual single codec. Additionally, the frame split mode in metadata processing significantly reduces the computational complexity in interactive scenarios while just slightly increasing network traffic usage.

Strength of the study

1. The developed screen-sharing system has a low end-to-end latency (17-65ms) according to the latency measurement, which may be utilized to fulfill various real-world applications.

2. The border quality artifact brought about by the multiple codec technique and the rate control procedure is then analyzed, and it is shown that such an impact is tolerable on fast network connections.

Weakness of the study

1. We could not find any weakness to this review work.

2.5 Summary of All the Five Related Reviews

The related work clearly illustrates the different methods used in wireless screensharing with each method's strength and limitations.

The first related work shows how the frame deletion strategy can improve the quality of graphics while ensuring efficiency of the network by offloading traffic as only selected frames are sent across the network with the system client. This means that snapshot frames are compared to previous snapshot frames to ensure that similar frames are not sent more than once through the network. Also, the network implementation in the first reviewed work employed the WLAN technology and screen sharing is limited to only one device from the client. This means that you can only share your screen to a monitor. The second reviewed work used the Websocket technology to share screens using http and https request. On the other hand a client system is not required in this method since laptops and other mobile devices already have web browsers embedded or installed in them. The system uses the web browser to request frames from the screen sharing computer through the network. This makes the system in reviewed work two easily accessible but also limits its scalability with network traffic. This system also fails to employ audio sharing. The Third related work compares various clientless systems by rigorously testing their efficiency and bandwidth. WebRTC performs best overall but within the native applications, raw socket performs best with little overhead. Reviewed work four compares different frames types used in screen sharing systems with WEBP frame format being the most lightweight among all with 44,970 Kb and JPEG giving the best average framerate of 10fps (frames per second) in low performing systems. Finally, review work five employs a multiple encoding technology that greatly improves low end-to-end latency (17-65ms) according to the latency measurement, which may be utilized to fulfill various real-world applications.

2.6 Identified Gaps from The Five Related Work

From the weaknesses of all the five related reviewed work, the following gaps were identified.

1. **The Implemented Systems Lack Audio Transmission:** Although from all the five related reviewed works the system performed so well with the various screen sharing technologies, the authors did not include any audio transmission mechanism. This reduced the scope of their work within a classroom.
2. **They had Limited Scalability:** All the systems the authors designed could connect a maximum of 20 devices which is not scalable enough.

2.7 Setup of Current Screen Sharing Systems from The Reviewed Works

With the current design, the client system collects frames from the device, encodes these frames and sends them through the WLAN to the server. The server on the other hand receives these frames, decodes them and render them within the server application as shown below.

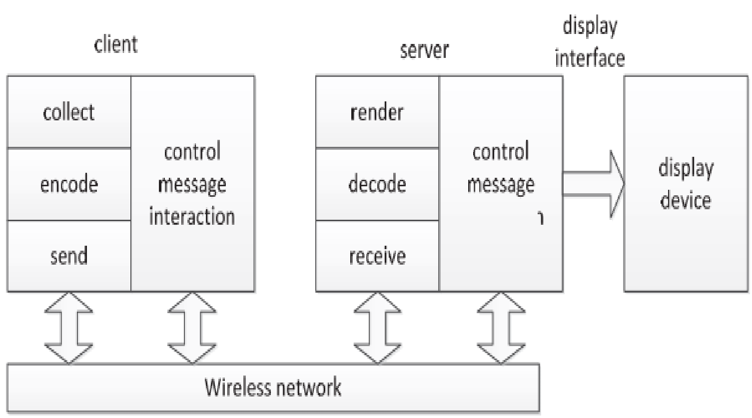

Figure 2.1 Design of the current systems (zhaohua et al, 2018)

2.8 Proposed Design of The Screen Sharing System

With this proposed design, the first two stages include both audio data and picture frames eliminating the lack of audio gaps. In these stages, both audio and frames are collected from the system and encoded separately. At the compare stage, the encoded audio bits are immediately sent through the network to the client systems but each picture frame taken is compared to the last frame sent. If both are similar, the frame is dropped. Otherwise, the current frame is sent through the network to the client. This process ensures that network traffic is reduced to the minimum and improves scalability as well. Below is the design of the proposed system.

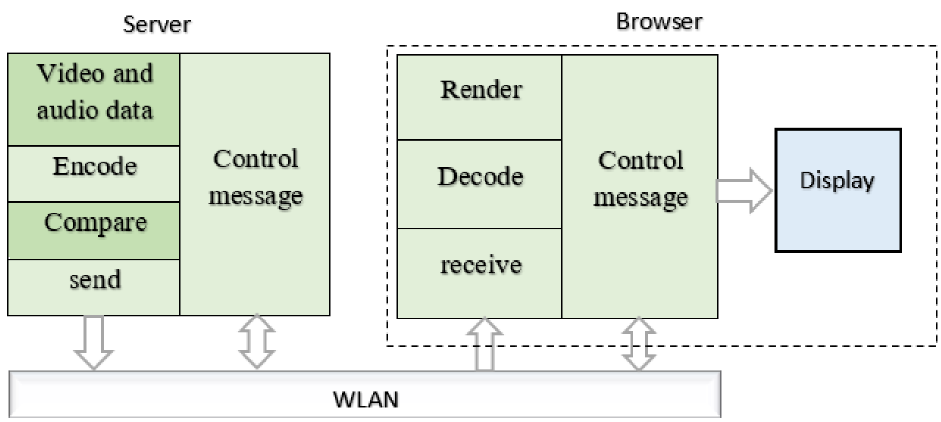

Figure 3.2 Design of the proposed system (Adabogo and ….,2022)

When choosing technologies for the design and implementation of this system, it is important to pay attention to factors such as effectiveness, fault tolerance. Flexibility and reusability, time and cost efficiency, user friendliness, attractiveness interfaces, future maintenance and overall performance. Based on the factors mentioned, the proposed system design is employing the WebRTC technology, multiple encoding algorithm and frame selection technology in designing the system to ensure a more robust and a scalable system that can accommodate up to 30 students in virtual classrooms.

2.9 Conclusion

The majority of software or hardware developed projects may be implemented using a number of different technologies. Certain technological combinations are better suited to the implementation of some tasks. Therefore, technology choice is crucial and important to software development studies. A system with unneeded faults and mistakes will result from the use of an improper language or framework, and a software implementation that fails will result from the use of an inappropriate set of technologies. Poorly picked technologies may be out-of-date, more complicated, inappropriate for the sector, unable to sustain the present system, posing integration issues, and most likely failing to meet the objectives set out. These related works have given insight on how to achieve our objective by giving us empirical data on the best frame file type to use, best technology to choose and algorithms to employ for an efficient system design as well as performance. The system in this research work would be employing webRTC technology to ensure device compatibility and low latency. Meaning, devices with different platforms can connect to the system with low latency which averages 1 second.

CHAPTER THREE

METHODOLOGY

3.0 Overview

This project is intended to use the Waterfall software development module. Following these steps is the Linear Sequential Software Development Lifecycle (SDLC) paradigm. The waterfall approach was the first SDLC model widely used in software engineering to ensure project success. The "waterfall" approach divides the entire software development process into discrete phases. Typically, in this waterfall model, the output of one phase serves sequentially as the input to the next phase. Analysis, design, implementation, testing and maintenance are requirements. Manzor Ahmad et al. (2015) concluded that this module is recommended for projects with tight deadlines and good design processes.

The project would be employing the WebRTC technology along with the Chrome browser to enable screen sharing using WLAN.

3.1 Design Concept

The classic waterfall model divides the life cycle into a series of phases. This model assumes that a phase can start after the previous phase has completed. That is, the output of one phase becomes the input of the next phase. In this way, the development process can be seen as a series of flows in the waterfall. Here the phases do not overlap. Each software developed is different and should follow the proper SDLC approach based on internal and external factors. Situations where the waterfall model is most appropriate are:

1. The requirements are well documented and well defined.

2. Product definition is stable.

3. Technology is understood and is not dynamic.

4. There are no ambiguous requirements.

5. Ample resources with required expertise are available to support the product.

6. The project is short.

3.2 The Waterfall Model

The waterfall model is a classic model used in the system development life cycle, which takes a linear and sequential approach to creating systems. It is called a waterfall because the model systematically evolves downward from one phase to another as shown in the figure below

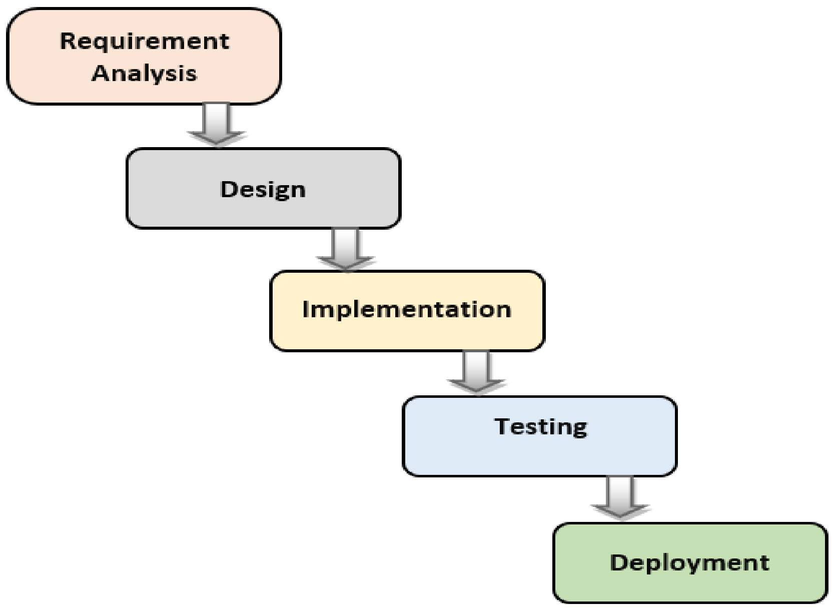

Figure 4.3 Waterfall Development Module

1. **Requirement Gathering and analysis**

   All possible requirements for the system to be developed are recorded during this phase and documented in the specification. Example Based on an analysis of the requirements provided by the customer, you can ask questions such as: Can both teachers and students share a screen? What is the bandwidth of the school WiFi? How many users are expected to be active per minute in the application?

2. **System Design**

   In this phase, we review the requirements specification for the first phase and prepare the system design. This system design specifies the hardware and system requirements and helps define the overall system architecture. For example, high-level design of software, low-level design, landing page design, video conferencing room design and system architecture were collaborated and discussed. This includes a description of redundant backup and failover capabilities to ensure that your system is always accessible.

3. **Implementation**

With input from system design, the system is first developed in small programs, so-called units, which are then integrated in the next stage. Each unit is developed and tested for functionality called unit testing. For example, based on the complexity of the application, the developer integrates features such as security checks, audit logs, etc. to ensure the application works smoothly.

4. Integration and Testing

All units developed in the implementation phase are integrated into the system after testing each unit. After integration, the entire system is tested for errors and failures. In this work, testers with reserved domain knowledge were also used in the project to test the application from a domain perspective. Application security, performance and bandwidth analysis is tested.

5. **Deployment of system**

Once the functional and non-functional tests are completed. The teaching system would be deployed on Ghana Communication Technology University (GCTU) campus. When performing this step, we ensured that the school campus WLAN environment is working and that the test exit criteria are met. It was important and we performed environment sanity checks after deploying the application to ensure that the application is working properly.

6. **Maintenance**

This process takes place immediately after installation and includes making necessary changes to the user interface, adding functionalities and modifying or correcting aspects related to system performance problems in the screen sharing system.

There are some issues that arise in the client environment. A patch has been released to fix these issues. Also, some better versions will be released to improve the product. Maintenance will be performed to roll out these changes to customer environments. Example: The screen sharing system consists of different sections responsible for performing separate functions such as the audio, video, visual message, etc. Also, it is important to keep these sections up to date. Bugs in live applications can cost a lot of money.

These issues that occur during the maintenance phase are most often due to changes made by customers or users after the installation and testing phases are complete. Maintenance includes "corrective maintenance," which fixes problems that were not detected during development, "adaptive maintenance," which is performed when moving environments to improve system performance, and "adaptive maintenance," which is performed at the request of users. Full maintenance (maintenance aimed at improving the functioning of the software).

3.2.1 Benefits of Waterfall Model Approach

The advantage of waterfall development is that it allows for departmentalization and control. You can set timelines with deadlines for each development phase, and your product can move through the phases of your development process model. Development extends from concept through design, implementation, testing, installation, troubleshooting, operation and maintenance. Each stage of development proceeds in a strict order.

3.2.1.1 Some of the major benefits of the Waterfall Model are as follows;

1. Simple and easy to understand and use

2. Easy to manage due to the rigidity of the model. Each phase has specific deliverables and a review process.

3. Phases are processed and completed one at a time.

4. Works well for smaller projects where requirements are very well understood.

5. Clearly defined stages.

6. Well understood milestones.

7. Easy to arrange tasks.

8. Process and results are well documented.

3.3 Requirement

The requirement of this project work is categorized in two parts. The Hardware requirement and the Software requirement.

3.3.1 Hardware requirement:

1. Intel Core i5-4234U 2.4GHz Processor computer
2. 8 GB DDR 4 Random Access Memory (RAM)
3. 500 GB Hard Drive Capacity
4. Routers

3.3.2. Software Requirement

This section explains the various tools used in the design and implementation of the said system. These software tools include.

3.3.2. *1 Web Browser (Firefox Web Browser).*

Web browsers provide built-in Application Programming Interfaces (APIs) for a variety of tasks, including dynamically generating HTML and CSS styles, gathering and modifying camera video streams, and producing 3D images and audio samples. The major requirement of the webRTC technology is the web browser since the technology connects different web browsers from different computer devices.

3.3.2.*2 Visual Studio Code as client site IDE (VS code)*

Microsoft Visual Studio is Microsoft's integrated development environment (IDE). It is used to develop not only computer programs but also websites, web applications, web services and mobile applications. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store, and Microsoft Silverlight. It can generate native and managed code. VS code is the IDE used in the development of both the client and server system.

3.4 Analysis phase

This phase explains the various software technologies employed in this project work. It also explains further the workings of these software systems in this research work. The following are the technologies used in developing both the client and server software systems.

3.4.1 System Software Analysis

In this section, we analyzed the various technologies available and selected a few that worked best for the development of the system in this research work. The section also explains the role and function of these technologies and why we are deploying them in this work. Below are various technologies used in the design and implementation of the screen sharing system.

3.4.1.1 WebRTC

WebRTC is a real communication system that adds real-time communication capabilities to an application. WebRTC is implemented based on an open standard created in all web browsers. By permitting the interchange of video, speech, and general data between peers, WebRTC enables us to develop and design efficient voice- and video-communication systems. Modern browsers and native clients for all key platforms both support the technology. WebRTC technologies are also supported by all of the major browsers as common JavaScript APIs that are implemented as an open web standard for native clients like iOS and Android apps, the same functionality is available as a library. Open-source WebRTC is supported by businesses like Apple, Google, Microsoft, and Mozilla. Typically, a WebRTC application will follow a standard application flow.

creating peer connections, finding peers, and starting streaming after gaining access to the media devices through a signaling system.

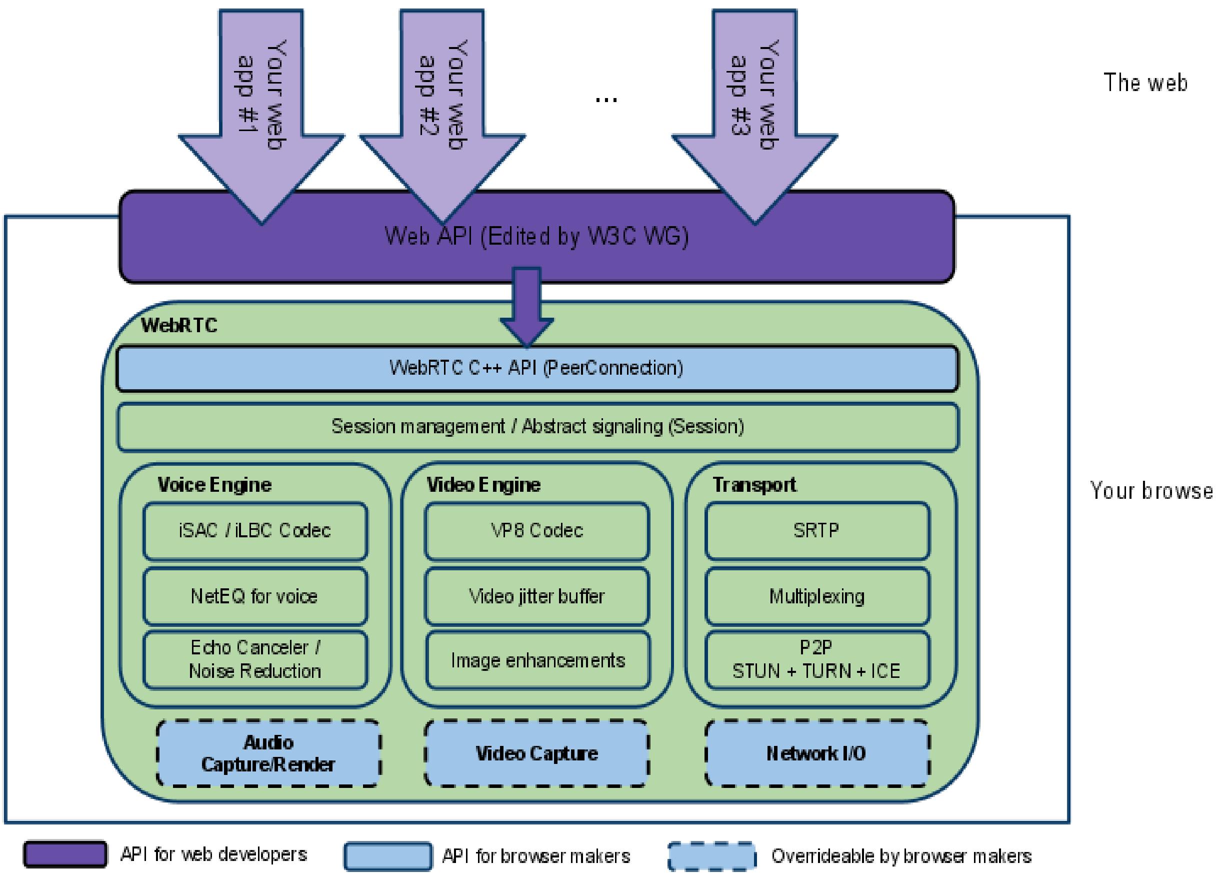

Figure 5 Architecture of the WebRTC technology (W3C WG, 2022)

3.4.1.2 WEBAPI (Web Application Programming Interface)

Web Application Programming Interface (API) is a web-based API that is used by web developers to create applications that resemble video chat on the internet. Browser developers have participated in the implementation of the WebRTC C++ API and the capture/render hooks which are implemented in several browsers. Web API such as WebRTC is the main protocol which is used in the implementation of this web-based application such as the client and server system in this project work

3.4.1.3 Hypertext Markup Language (HTML)

According to Modzilla group(2022), The coding that organizes a web page's content is called HTML (Hyper Text Markup Language). Content may be organized using paragraphs, a list of bulleted points, graphics, and data tables, among other options. HTML is a markup language that specifies how your material is organized. HTML is made up of a number of components that you may employ to enclose or wrap certain portions of the content to alter how it appears or behaves. The surrounding tags can italicize words, make the font larger or smaller, make a word or image hyperlink to another location, and more. Check out the following line of material as an illustration:

```html
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

<h1>This is a Heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

Figure 6 Sample code of HTML

3.4.1.4 JavaScript

JavaScript is a computer language that makes your website more interactive. This occurs in video games, in the way that buttons react or how data is entered on forms, in dynamic style, in animation, etc. JavaScript is a potent programming language that may increase a website's responsiveness. Brendan Eich is the one who created it. JavaScript is adaptable and user-friendly for newcomers. It enables users to be able to develop games, animated 2D and 3D visuals, thorough database-driven programs, and much more as you gain expertise. Although very small in size, JavaScript is quite versatile. On top of the fundamental JavaScript language, programmers have created a range of tools, enabling a large amount of capability with little effort. In that regard, we are using the javaScript as the main programming language for developing this system. The following are technologies or tools developed using the javaScript language.

3.4.1.5 Cascading Style Sheet (CSS)

The process of making web pages attractive is made easier with the use of CSS, often known as cascading style sheets. Students and working professionals who want to excel as software engineers, particularly those in the web development field, use CSS application UX design. We are use CSS in this project work to,

1. Create a Stunning Landing Page and Video conference room - CSS controls how a web page looks and feels. The color of the text, the font style, the spacing between paragraphs, the size and arrangement of columns, the background pictures or colors used, layout designs, differences in display for various devices and screen sizes, and a number of other effects may all be controlled using CSS. For an efficient video conferencing system CSS is the main web page design tool in this research work.

2. Control the web - CSS is simple to learn and comprehend yet offers effective control over how an HTML document is presented. CSS is most frequently used in conjunction with HTML or XHTML as markup languages.

3. Lightweight – Bandwidth is a concerning issue in the school WLAN. Once we used HTML, CSS is the most lightweight tool and simpler to comprehend with other related technologies, such as javaScript, php, or angular.

3.5 DESIGN

In the third stage of the waterfall software development module (SDL), we have created an efficient and thoroughly planned design for our presentation-based learning system using a low level and high-level design plan. We used Microsoft Word in designing both low-level and high-level system architecture. Later with the implemented design, each section was programmed using Visual studio code with frequent testing using the Firefox web browser.

3.5.1 System design diagram

This section shows the block diagram of the server-client relationship in the presentation-based teaching system. The section also explains the workings of each section in the design diagram. Diagram below show the block diagram of the design system

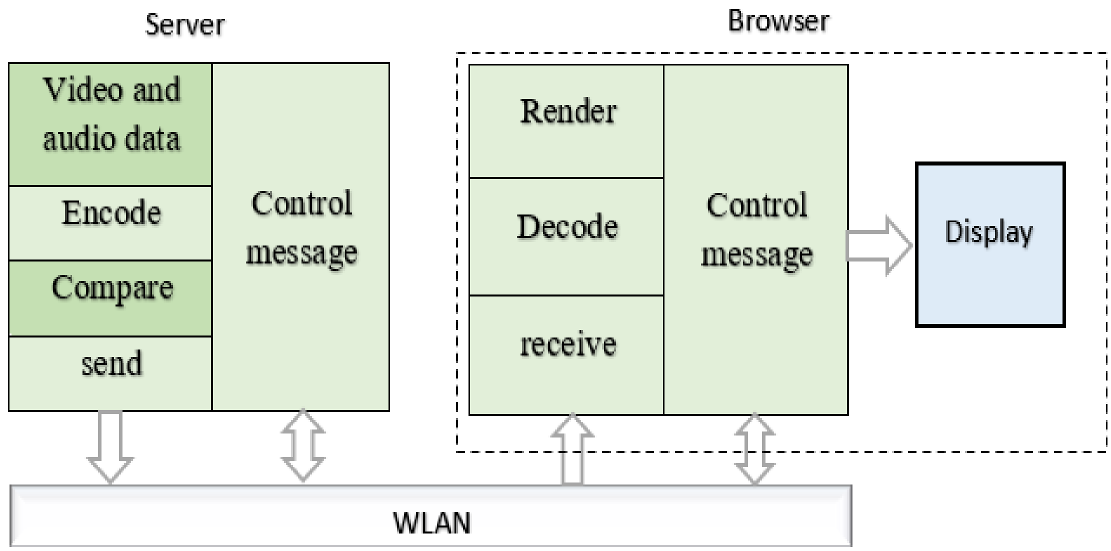

Figure 7 block diagram of the design system (Adabogo and Amedoadzi, 2022)

3.5.2. Server system operation

The server is the backend section of the system. It uses a web browser as its interface to create the meeting room required for the client systems to join and receive the shared screen data packets. The server also connects the clients together to enable them to directly send text messages from one connected client to another. Socket.IO is the technology we deployed to connect the client systems together. This is so because Socket.IO has a perfect event listing mechanism that has a fast response time.

To ensure bitstreams are properly processed with minimal loss, the following section as indicated in figure 3.5 is explained in details below.

3.5.2.1 Video and audio data

With the WebRTC technology embedded in web browsers, the server system uses the WebRTC technology to take a snapshot of the selected screen sharing windows. With screen sharing, the user has an option either to share the view of a selected window or share the entire computer screen. The server system also uses the WebRTC technology to access the computer microphone for voice recording.

3.5.2.2 Encoding:

WebRTC uses the Advanced Encoding Technology (H.264) as its encoding algorithm. H.264 is an industry standard for video compression, the process of converting digital video to a format that takes up less space for storage or transmission. Video compression (or video encoding) is an essential technology for applications such as digital television, DVD-Video, mobile television, video conferencing, and video transmission over the Internet. Standardizing video compression with H.264 allows video streaming from different browser developers (e.g. encoders, decoders, and storage media) to be compatible with each other. The H.264 encoder converts the video and screenshot picture frames to a compressed format and its corresponding decoder converts the compressed video back to an uncompressed format. The H.264 video encoder performs prediction, conversion, and encoding processes to produce a compressed H.264 bitstream. The H.264 video codec performs additional decoding, inverse transformation, and reconstruction procedures to produce the decoded video sequence.

3.5.2.3 Frame Comparison

Frame Deletion Strategy The authors further implement a frame-rejection strategy to minimize the number of image frames sent over the network at any given time. This strategy uses the mean square error (MSE) algorithm to compare previous frames with current frames. If there are no errors, the frame is discarded to avoid sending similar frames across the network. But if there is an error, the frame is sent over the network. Shu-dong, L. et al, (2008) also mentioned in their study that the weight-based algorithm improves latency reduction, service quality and real-time customer service system optimization. WebRTC 2022 uses this algorithm after encoding to drop duplicate frames while improving throughput.

3.5.2.4 Send

Bit stream encoding the video coding process produces a number of values that must be **encoded** to form the compressed bit stream. Socket.IO is the javaScript framework used in research work for packet routing as well sending picture frames from the server system to client system. Socket.IO also undertake the following processes:

- Quantized transform coefficients
- Information to enable the decoder to re-create the prediction
- Information about the structure of the compressed data and the compression tools used during encoding
- Information about the complete video sequence.
- These values and parameters (**syntax elements**) are converted to binary code using variable-length encoding and arithmetic code. Each of these encryption methods produces an efficient and compact binary representation of the information. The encoded bit stream can then be stored and transmitted. Decoder processes

3.5.3 Client System Operations (Firefox web browser)

3.5.3.1 Received

Receiving bit streams from WLAN is the first step the client undergoes. In this step, bit-errors are checked to select successfully received frames. An acknowledgement is sent back to the sender in the client system to a success or failure in receiving the frames. If the acknowledgement is a failure, the frames are resent back here to the receiver. Socket.IO handles the connection bit-error checking mechanism.

3.5.3.2Decode

The second step in the client system is Bit stream decoding. WebRTC uses a H.264 decoder to receive the compressed H.264 bitstream, decompress the bit stream and decode it back to its original form for rendering. H.264 decodes each of the syntax elements and extracts the information described above (quantized transform coefficients, prediction information, etc.). This information is then used to reverse the coding process and recreate a sequence of video images.

3.5.3.3 Render

The web browser now displays the sequence of images as a video to the user.

3.5.4 Session Traversal Utilities for NAT (STUN)

Real-time phone, video, and messaging over IP networks are supported by an IETF protocol. STUN offers a way to connect with users who are protected by a Network Address Translation (NAT) firewall, which masks their IP addresses inside the local network (LAN). The initiating party makes a request to the STUN server, which keeps track of the computer or phone's IP address (for video). Using protocols like WebRTC or ICE, a peer-to-peer connection is then created after that. STUN does not require router configuration, in contrast to application-layer gateways (ALGs), which also facilitate two-way communication via NATs.

3.5.5 System Network infrastructure with WebRTC

Peer-to-peer communication is made possible via WebRTC, but it still requires servers so that clients may share metadata, deal with NATs and firewalls, and coordinate communication through a process called signaling. The building of a signaling service and how to handle the peculiarities of real-world connection with STUN servers are demonstrated in this article. Additionally, it describes how WebRTC applications may manage multiparty conversations and communicate with services like VoIP and PSTN (also known as telephones).

WebRTC programs employ an intermediary server for metadata signaling, but once a session has been formed, RTCPeerConnection tries to connect clients directly or peer-to-peer for actual video and data streaming. Every WebRTC endpoint would have a distinct address for it to exchange with other peers to communicate directly in a simpler world.

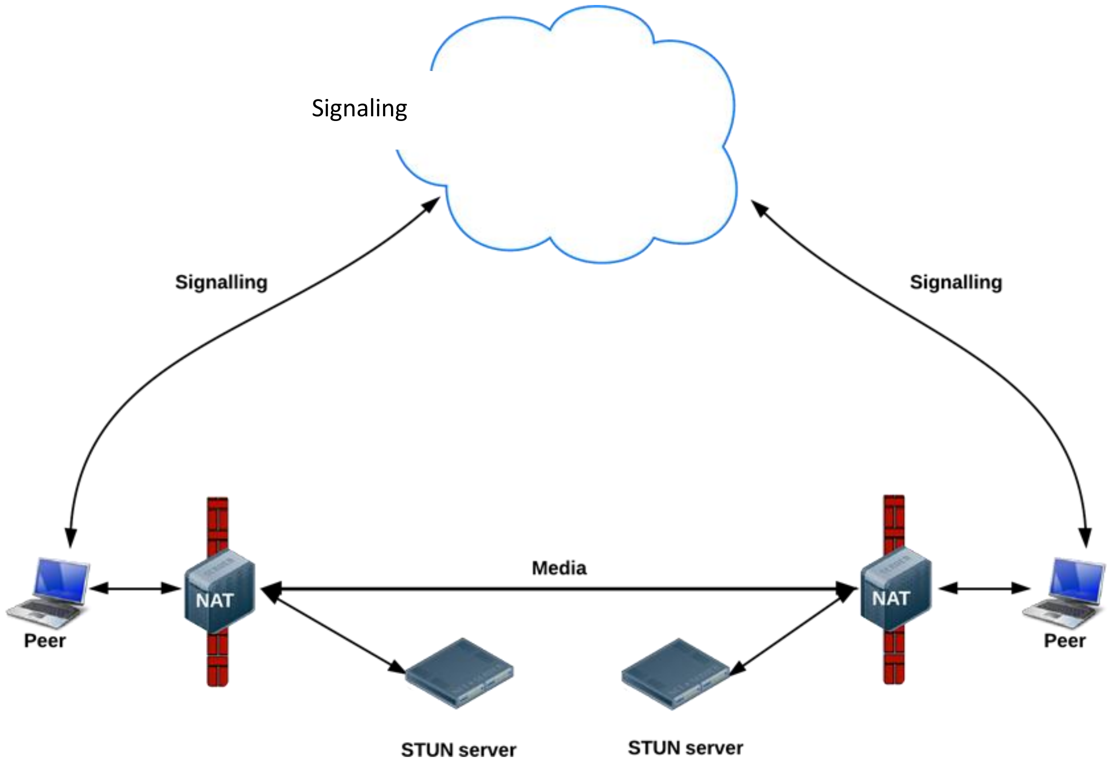

Figure 8 signaling and peer to peer connection

3.5.6 High-level design of software,

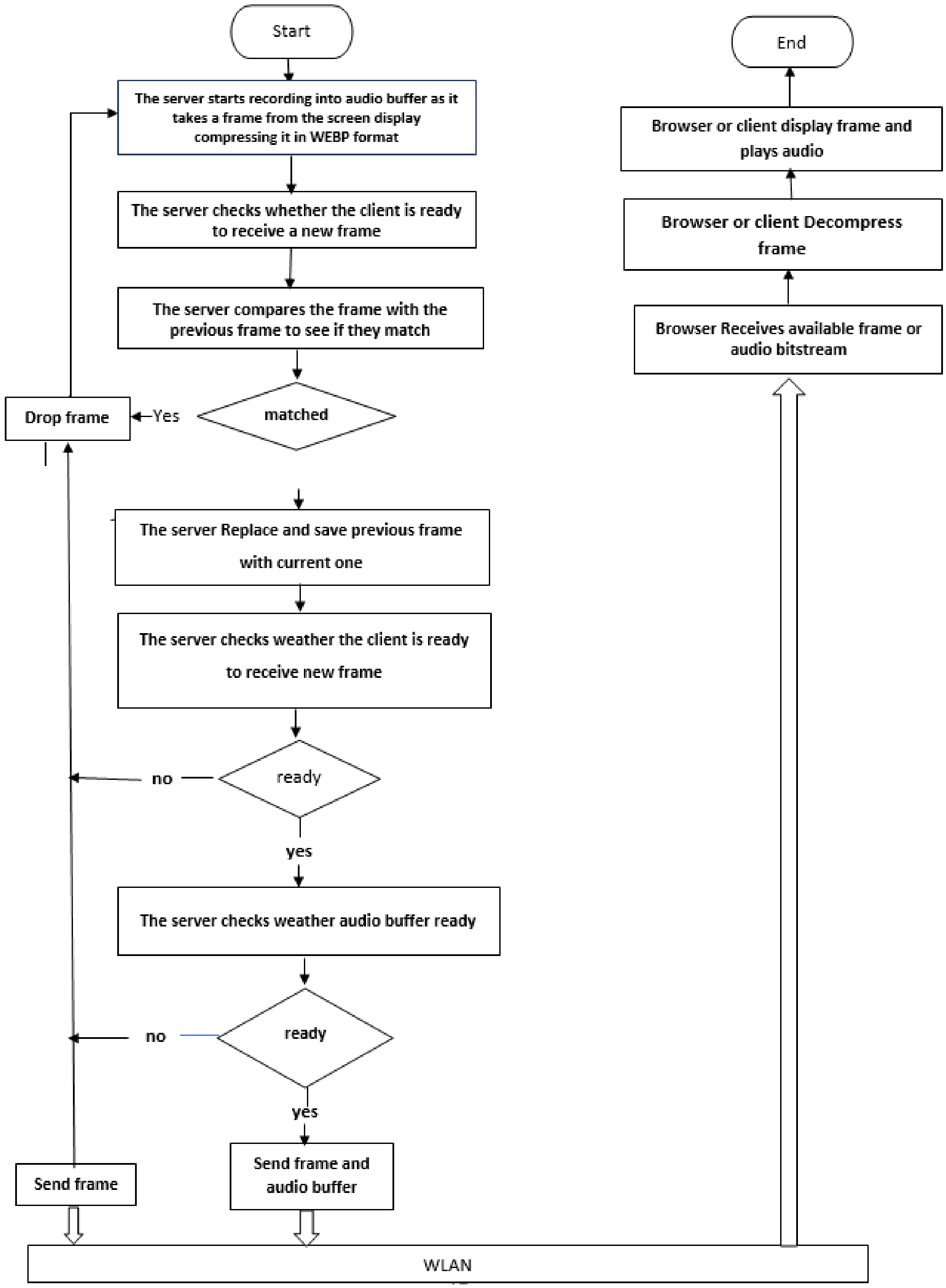

Figure 9 High level system flow chart

3.6 Functionality of The System

The system is designed with a very simple interface. Both the Presenter and Participants use the web browser as an interactive interface. Since the Presenter is hosting the Server, the Server identifies Present the localhost address (127.0.0.1) IP address.

3.6.1 Functionality of The Presenter.

1. The presenter generates links for the Client authentication and connection.
2. Create a meeting with a generated meeting Identification number (ID) for the Client to use
3. The presenter controls the screen sharing process and messaging as the server manages interaction between the client to client and client to server signaling.
4. Provide functionality to View all participants in the meeting.

3.6.2 Functionality of the Participants

1. Provide functionality for the Client to connect to the server with a URL.
2. Provide functionality for the Client to connect to the server with a Meeting Url.
3. Provide functionality for messaging within the meeting room
4. It also has functionality for the Client to mute or unmute for class interaction within the meeting room.
5. The Client also provides functionality to view all participants within the Meeting.

3.6.3 Peer-to-Peer Connection Between Participants and Presenter.

In this section, we describe how connection is established between devices through the WebRTC technology. The steps below outline the processing involved in establishing a successful connection.

1. When the signaling server finishes establishing signaling connection with the running client device, the running device creates a local description which is a description that

contains the running device information such the IP address, port number and other headers necessary for peer connection.

2. A processing event checks the signaling connection to see if negotiation is necessary for the peer connection. If negotiation is necessary, the device uses the local description to create an offer which is sent to the other peer through the signaling server.


```
▸ Object { type: "offer", sdp: "v=0\r\no=- 8653969059197948678 2 IN IP4 127.0.0.1\r\ns=-\r\nt=0 0\r\na=group:BUNDLE 0\r\na=extmap-allow-
mixed\r\na=msid-semantic: WMS S225gBj81Y1h0SUUy7KRzCOOuS3mgCaPtPh9\r\nm=video 9 UDP/TLS/RTP/SAVPF 96 97 102 122 127 121 125 107 108 109 124
120 39 40 45 46 98 99 100 101 123 119 114 115 116\r\nc=IN IP4 0.0.0.0\r\na=rtcp:9 IN IP4 0.0.0.0\r\na=ice-ufrag:SFiU\r\na=ice-
pwd:MJwzCqegoP5rEL4JXpJer9Tm\r\na=ice-options:trickle\r\na=fingerprint:sha-256
FA:4B:77:59:24:B9:79:F1:19:7E:20:10:9D:02:33:B1:27:72:39:4D:E9:C6:89:1F:66:57:2C:F1:2D:10:61:2D\r\na=setup:actpass\r\na=mid:0\r\na=extmap:1
urn:ietf:params:rtp-hdrext:toffset\r\na=extmap:2 http://www.webrtc.org/experiments/rtp-hdrext/abs-send-time\r\na=extmap:3 urn:3gpp:video-
orientation\r\na=extmap:4 http://www.ietf.org/id/draft-holmer-rmcat-transport-wide-cc-extensions-01\r\na=extmap:5 http://www.webrtc.org
/experiments/rtp-hdrext/playout-delay\r\na=extmap:6 http://www.webrtc.org/experiments/rtp-hdrext/video-content-type\r\na=extmap:7
http://www.webrtc.org/experiments/rtp-hdrext/video-timing\r\na=extmap:8 http://www.webrtc.org/experiments/rtp-hdrext/color-space\r\na=extmap:9
urn:ietf:params:rtp-hdrext:sdes:mid\r\na=extmap:10 urn:ietf:params:rtp-hdrext:sdes:rtp-stream-id\r\na=extmap:11 urn:ietf:params:rtp-
hdrext:sdes:repaired-rtp-stream-id\r\na=sendrecv\r\na=msid:S225gBj81Y1h0SUUy7KRzCOOuS3mgCaPtPh9 cce13760-55bf-42e5-b32b-feb9d621c789\r
\na=rtcp-mux\r\na=rtcp-rsize\r\na=rtpmap:96 VP8/90000\r\na=rtcp-fb:96 goog-remb\r\na=rtcp-fb:96 transport-cc\r\na=rtcp-fb:96 ccm
fir\r\na=rtcp-fb:96 nack\r\na=rtcp-fb:96 nack pli\r\na=rtpmap:97 rtx/90000\r\na=fmtp:97 apt=96\r\na=rtpmap:102 H264/90000\r\na=rtcp-fb:102
goog-remb\r\na=rtcp-fb:102 transport-cc\r\na=rtcp-fb:102 ccm fir\r\na=rtcp-fb:102 nack\r\na=rtcp-fb:102 nack pli\r\na=fmtp:102 level-
asymmetry-allowed=1;packetization-mode=1;profile-level-id=42001f\r\na=rtpmap:122 rtx/90000\r\na=fmtp:122 apt=102\r\na=rtpmap:127 H264/90000\r
\na=rtcp-fb:127 goog-remb\r\na=rtcp-fb:127 transport-cc\r\na=rtcp-fb:127 ccm fir\r\na=rtcp-fb:127 nack\r\na=rtcp-fb:127 nack pli\r\na=fmtp:127
level-asymmetry-allowed=1;packetization-mode=0;profile-level-id=42001f\r\na=rtpmap:121 rtx/90000\r\na=fmtp:121 apt=127\r\na=rtpmap:125
H264/90000\r\na=rtcp-fb:125 goog-remb\r\na=rtcp-fb:125 transport-cc\r\na=rtcp-fb:125 ccm fir\r\na=rtcp-fb:125 nack\r\na=rtcp-fb:125 nack
pli\r\na=fmtp:125 level-asymmetry-allowed=1;packetization-mode=1;profile-level-id=42e01f\r\na=rtpmap:107 rtx/90000\r\na=fmtp:107 apt=125\r
\na=rtpmap:108 H264/90000\r\na=rtcp-fb:108 goog-remb\r\na=rtcp-fb:108 transport-cc\r\na=rtcp-fb:108 ccm fir\r\na=rtcp-fb:108 nack\r\na=rtcp-
fb:108 nack pli\r\na=fmtp:108 level-asymmetry-allowed=1;packetization-mode=0;profile-level-id=42e01f\r\na=rtpmap:109 rtx/90000\r\na=fmtp:109
apt=108\r\na=rtpmap:124 H264/90000\r\na=rtcp-fb:124 goog-remb\r\na=rtcp-fb:124 transport-cc\r\na=rtcp-fb:124 ccm fir\r\na=rtcp-fb:124 nack\r
\na=rtcp-fb:124 nack pli\r\na=fmtp:124 level-asymmetry-allowed=1;packetization-mode=1;profile-level-id=4d001f\r\na=rtpmap:120 rtx/90000\r
\na=fmtp:120 apt=124\r\na=rtpmap:39 H264/90000\r\na=rtcp-fb:39 goog-remb\r\na=rtcp-fb:39 transport-cc\r\na=rtcp-fb:39 ccm fir\r\na=rtcp-fb:39
nack\r\na=rtcp-fb:39 nack pli\r\na=fmtp:39 level-asymmetry-allowed=1;packetization-mode=0;profile-level-id=4d001f\r\na=rtpmap:40 rtx/90000\r
\na=fmtp:40 apt=39\r\na=rtpmap:45 AV1/90000\r\na=rtcp-fb:45 goog-remb\r\na=rtcp-fb:45 transport-cc\r\na=rtcp-fb:45 ccm fir\r\na=rtcp-fb:45
nack\r\na=rtcp-fb:45 nack pli\r\na=rtpmap:46 rtx/90000\r\na=fmtp:46 apt=45\r\na=rtpmap:98 VP9/90000\r\na=rtcp-fb:98 goog-remb\r\na=rtcp-fb:98
transport-cc\r\na=rtcp-fb:98 ccm fir\r\na=rtcp-fb:98 nack\r\na=rtcp-fb:98 nack pli\r\na=fmtp:98 profile-id=0\r\na=rtpmap:99 rtx/90000\r
\na=fmtp:99 apt=98\r\na=rtpmap:100 VP9/90000\r\na=rtcp-fb:100 goog-remb\r\na=rtcp-fb:100 transport-cc\r\na=rtcp-fb:100 ccm fir\r\na=rtcp-
fb:100 nack\r\na=rtcp-fb:100 nack pli\r\na=fmtp:100 profile-id=2\r\na=rtpmap:101 rtx/90000\r\na=fmtp:101 apt=100\r\na=rtpmap:123 H264/90000\r
\na=rtcp-fb:123 goog-remb\r\na=rtcp-fb:123 transport-cc\r\na=rtcp-fb:123 ccm fir\r\na=rtcp-fb:123 nack\r\na=rtcp-fb:123 nack pli\r\na=fmtp:123
level-asymmetry-allowed=1;packetization-mode=1;profile-level-id=64001f\r\na=rtpmap:119 rtx/90000\r\na=fmtp:119 apt=123\r\na=rtpmap:114
red/90000\r\na=rtpmap:115 rtx/90000\r\na=fmtp:115 apt=114\r\na=rtpmap:116 ulpfec/90000\r\na=ssrc-group:FID 2365982203 1055998543\r
\na=ssrc:2365982203 cname:GKybVq8D4Pww3+AB\r\na=ssrc:2365982203 msid:S225gBj81Y1h0SUUy7KRzCOOuS3mgCaPtPh9 cce13760-55bf-42e5-b32b-
feb9d621c789\r\na=ssrc:1055998543 cname:GKybVq8D4Pww3+AB\r\na=ssrc:1055998543 msid:S225gBj81Y1h0SUUy7KRzCOOuS3mgCaPtPh9 cce13760-55bf-42e5-
b32b-feb9d621c789\r\n" }
```


Figure 10 Sample of An offer message

3. Through the signaling server, other devices receive this offer. Upon receiving the Offer, these devices create a remote description which contains information such as these devices IP addresses, ports and other necessary heads. Upon creating the remote description, these devices create an Answer package which is sent back to the received destination (running device) through the signaling server.

```
Answer                                                                                                    app.js:380:25
▸ RTCSessionDescription { type: "answer", sdp: "v=0\r\no=mozilla...THIS_IS_SDPARTA-99.0 6916789179638004126 0 IN IP4 0.0.0.0\r\ns=-\r\nt=0
0\r\na=sendrecv\r\na=fingerprint:sha-256 7B:E6:70:90:15:E1:71:4A:03:6F:E6:8E:93:BC:65:22:A6:15:50:B3:88:79:C2:9E:75:05:D2:6E:B8:C2:CC:F8\r
\na=group:BUNDLE 0\r\na=ice-options:trickle\r\na=msid-semantic:WMS *\r\nm=video 9 UDP/TLS/RTP/SAVPF 96 97 125 107 108 109 98 99\r\nc=IN IP4
0.0.0.0\r\na=recvonly\r\na=extmap:1 urn:ietf:params:rtp-hdrext:toffset\r\na=extmap:2 http://www.webrtc.org/experiments/rtp-hdrext/abs-send-
time\r\na=extmap:4 http://www.ietf.org/id/draft-holmer-rmcat-transport-wide-cc-extensions-01\r\na=extmap:5/recvonly http://www.webrtc.org
/experiments/rtp-hdrext/playout-delay\r\na=extmap:9 urn:ietf:params:rtp-hdrext:sdes:mid\r\na=fmtp:125 profile-level-id=42e01f;level-asymmetry-
allowed=1;packetization-mode=1\r\na=fmtp:108 profile-level-id=42e01f;level-asymmetry-allowed=1\r\na=fmtp:96 max-fs=12288;max-fr=60\r
\na=fmtp:97 apt=96\r\na=fmtp:107 apt=125\r\na=fmtp:109 apt=108\r\na=fmtp:98 max-fs=12288;max-fr=60\r\na=fmtp:99 apt=98\r\na=ice-
pwd:0352ddcd9256c6d94243a30c6030722e\r\na=ice-ufrag:f2731d9b\r\na=mid:0\r\na=rtcp-fb:96 nack\r\na=rtcp-fb:96 nack pli\r\na=rtcp-fb:96 ccm
fir\r\na=rtcp-fb:96 goog-remb\r\na=rtcp-fb:96 transport-cc\r\na=rtcp-fb:125 nack\r\na=rtcp-fb:125 nack pli\r\na=rtcp-fb:125 ccm fir\r\na=rtcp-
fb:125 goog-remb\r\na=rtcp-fb:125 transport-cc\r\na=rtcp-fb:108 nack\r\na=rtcp-fb:108 nack pli\r\na=rtcp-fb:108 ccm fir\r\na=rtcp-fb:108 goog-
remb\r\na=rtcp-fb:108 transport-cc\r\na=rtcp-fb:98 nack\r\na=rtcp-fb:98 nack pli\r\na=rtcp-fb:98 ccm fir\r\na=rtcp-fb:98 goog-remb\r\na=rtcp-
fb:98 transport-cc\r\na=rtcp-mux\r\na=rtcp-rsize\r\na=rtpmap:96 VP8/90000\r\na=rtpmap:97 rtx/90000\r\na=rtpmap:125 H264/90000\r\na=rtpmap:107
rtx/90000\r\na=rtpmap:108 H264/90000\r\na=rtpmap:109 rtx/90000\r\na=rtpmap:98 VP9/90000\r\na=rtpmap:99 rtx/90000\r\na=setup:active
\r\na=ssrc:1545379193 cname:{639d1a7f-4b0c-46ba-88b7-56b00ea6e698}\r\n" }
```


Figure 11Sample An Answer message

4. Getting an answer package as an acknowledgement, the running device creates an ICE candidate with the Answer package.

5. At this stage if the remote devices are within a WAN a STUN server is used to locate the best possible route for a Peer-to-Peer connection to be established.

6. But as we are using WLAN in our system, a Peer-to-Peer connection is established between all the devices. At this point, Video and audio streaming can be initialized for streaming within all the connected devices.

7. Peer-to-Peer connection established, the devices interact directly between each other rather than using the signaling server.

8. On the other hand, text messaging uses the signaling server for communication between devices.

3.7 Summary of Chapter

We used the software development life cycle (SDLC) in this chapter. We used the waterfall SDLC module in the design. We added various software and related technologies needed for the implementation of this work. JavaScript is the main language used in building the backend part of the system (Server). HTML and CSS were used in building the user interface for both Server and Client. A personal computer (PC) is used for the deployment of the system since only these PC’s can host the Server system. The school WLAN is used for the transmission and receiving of data packets between devices.

CHAPTER FOUR

ANALYSIS AND RESULT

4.0 Overview

In this section, we will test the functionalities and workings of the system. We would also be gathering results for an in depth analysis of the system and the core system functionalities such as key button labels actions. Therefore, we carefully perform a series of indebted tests on the system to monitor the desktop base screen sharing system performance.

4.1. Presenter's User Interface

The Presenter hosts the server system with it being a personal computer. The presenter's user interface is designed with simplicity in mind. It has two key buttons which are listed below.

- ***Generate Meeting URL button***; The button allows the presenter to generate or create a shared link for the participants. This link enables the participants to connect to the virtual class. After generating a meeting link, there is a copy icon which enables the presenter to copy the link directly.
- ***Create Meeting button;*** This button creates a virtual class for presentation to begin

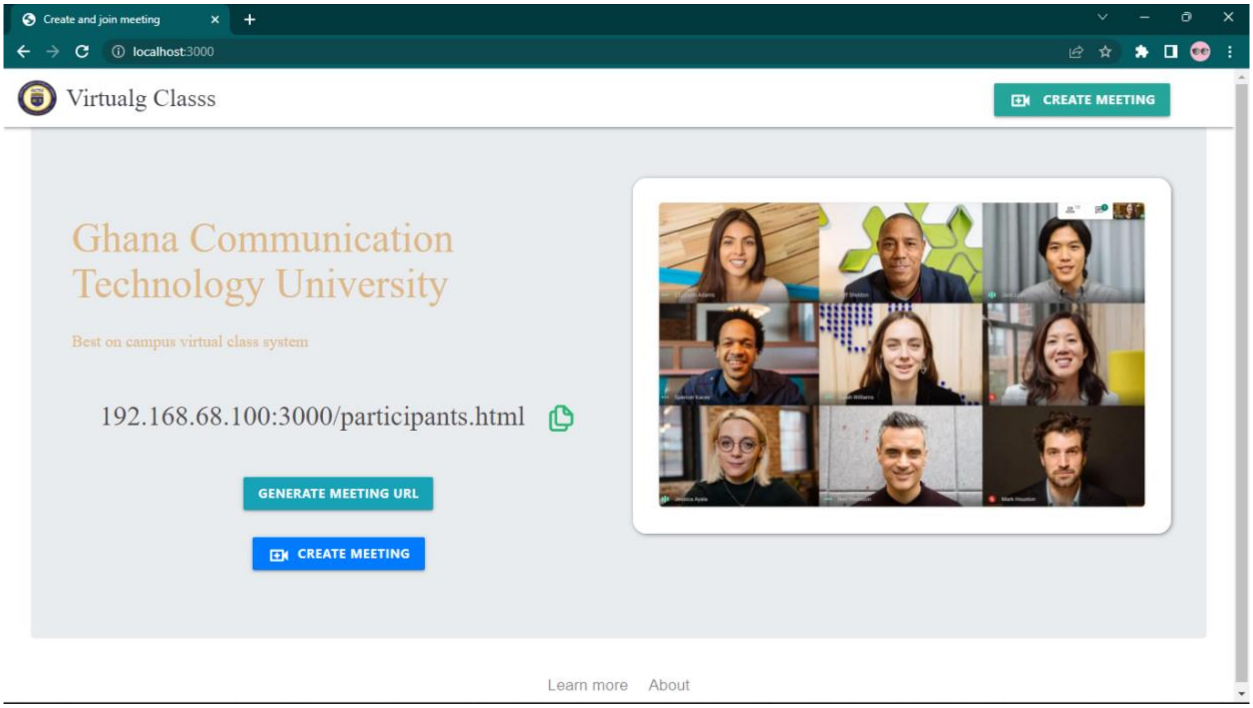

Figure 12 Presenter Interface (Adabogo, Amaduazi. 2022)

4.1.2 Presenter's Meeting room.

The create meeting button in the user interface above creates and opens this presentation meeting room. This meeting room is specifically designed for the presenter as it contains functionalities only for the presenter. The meeting room has five sections which is shown in the figure below.

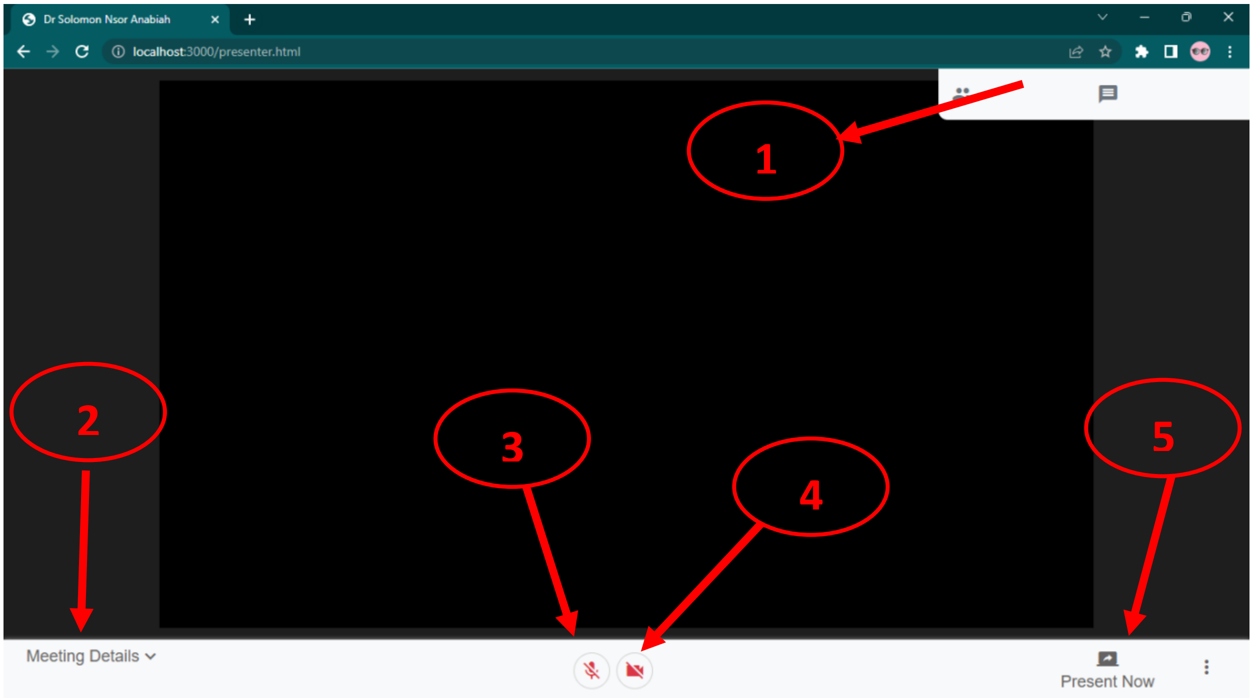

Figure 13 Presenter Meeting room (Adabogo, Amaduazi. 2022)

Below are the names and functions of each bottom on the screen view.

1. The notification pane: This pane shows notifications such as number of users connected to the virtual class and number of incoming messages received by the presenter. It is pane at the upper right corner of the meeting room.

2. The Meeting Details button: This button opens up the meeting details pane which contains the participants view in which the names of participants in the class is displayed. It also opens up the Chat section where the presenter can send text messages and view received messages from participants.

3. The Mic button: The mic button allows the presenter to mute and unmute his/her mic during the presentation.

4. The video button: This button allows the user to use the camera during presentation.

5. **The Present Now button:** This button allows the presenter to share his/her screen. It has options such as sharing an entire screen, sharing a window or sharing a specific browser tab. This is shown in the figure below

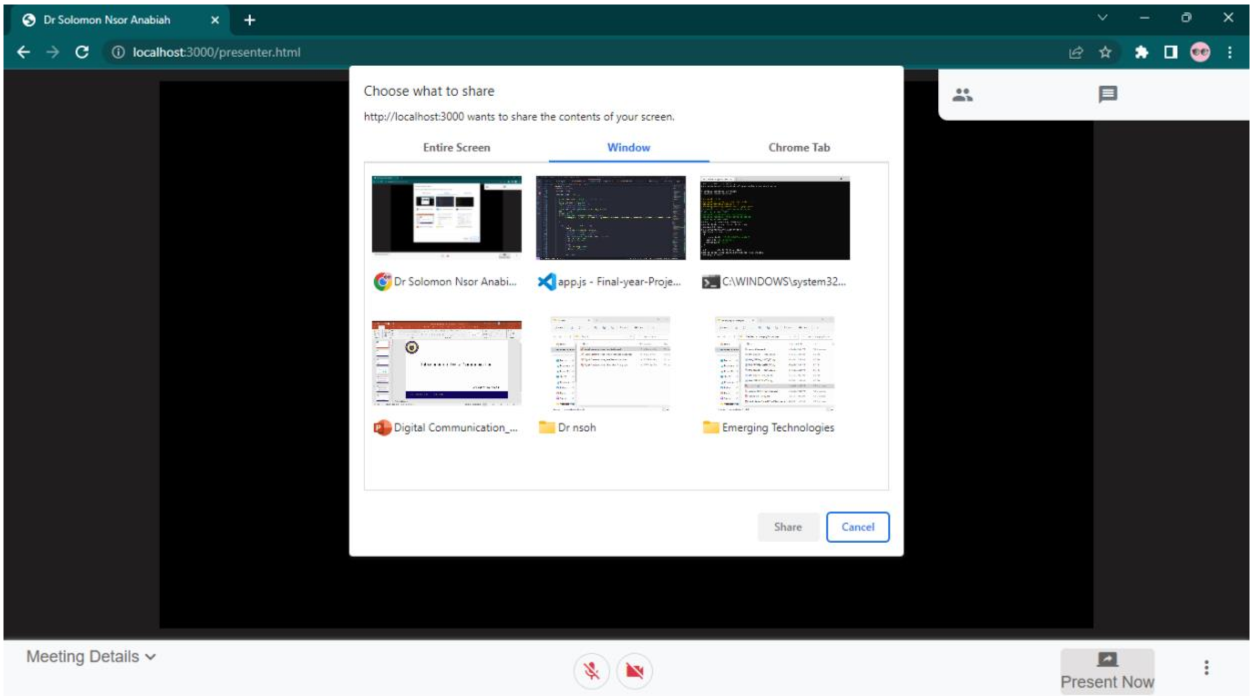

Figure 14 Presenter Meeting room (Adabogo, Amaduazi. 2022)

4.2 Client user interface

The client is simply a browser on any connected device to the server. The client system represents the participants in the virtual classroom. Through the link shared by the Server system in the presenter's device, participants using the client server are able to connect to the virtual classroom. The participants have fewer functionality compared to the presenters. Example the participants do not have functionalities such as video camera sharing and screen sharing unlike the presenters. The client system has two user interfaces. One for personal computers and the other for mobile devices. This is further explained below

4.2.1 Client's Personal Computer user interface

The figure below shows the user interface of a participant using a personal computer in an ongoing classroom.

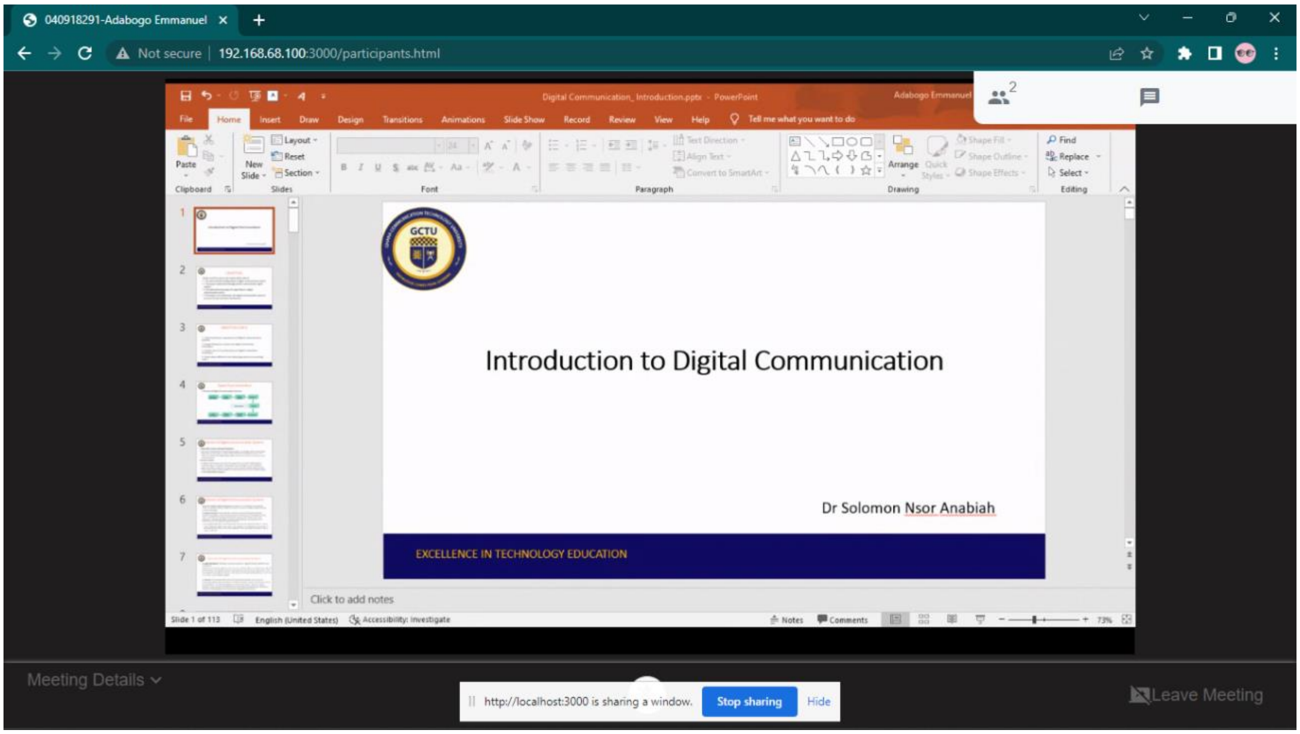

Figure 15 Presenter Meeting room (Adabogo, Amaduazi. 2022)

4.2.2 Clients Mobile Interface View.

The figure below shows the mobile view of a participant. The more button beside the microphone icon is the same as the meeting details button in the personal computer view.

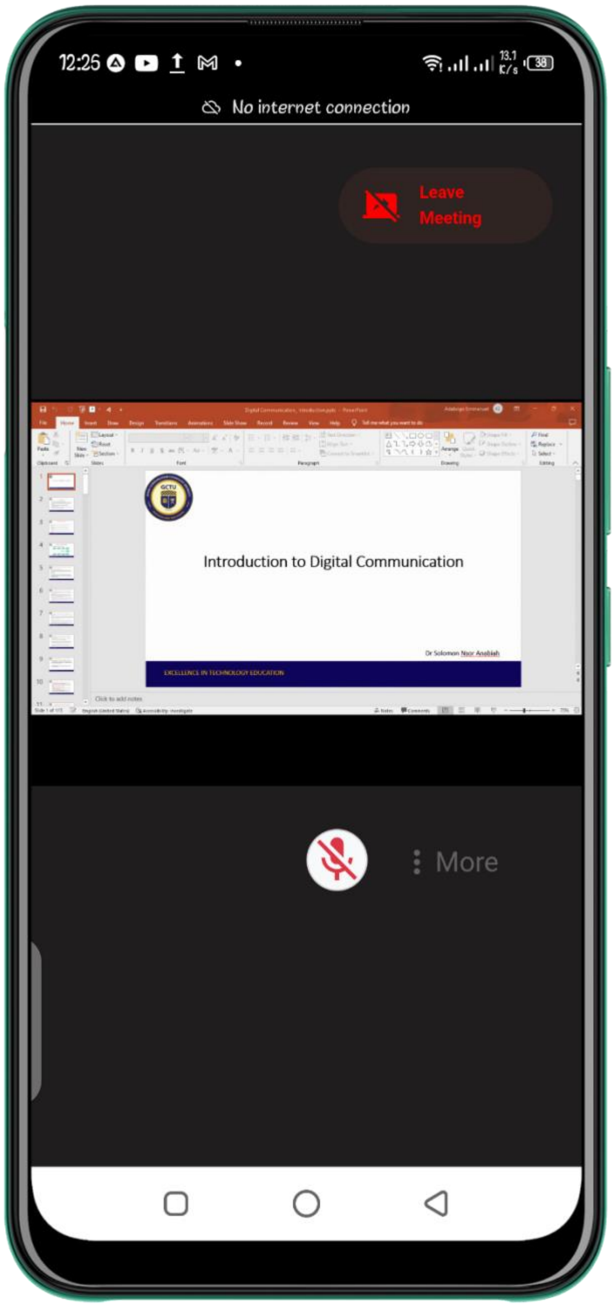

Figure 16 Mobile screen (Adabogo, Amaduazi. 2022)

4.3.0 Common functionalities

This section explains the functionalities that both presenter and participant share in the meeting details pane. These shared functionalities are listed below.

4.3.1. Participants panel

The participants view shows the names and number of participants in the virtual classroom. There is the mobile view and the personal computer (PC) view.

4.3.1.0 PC view of the Participants Panel

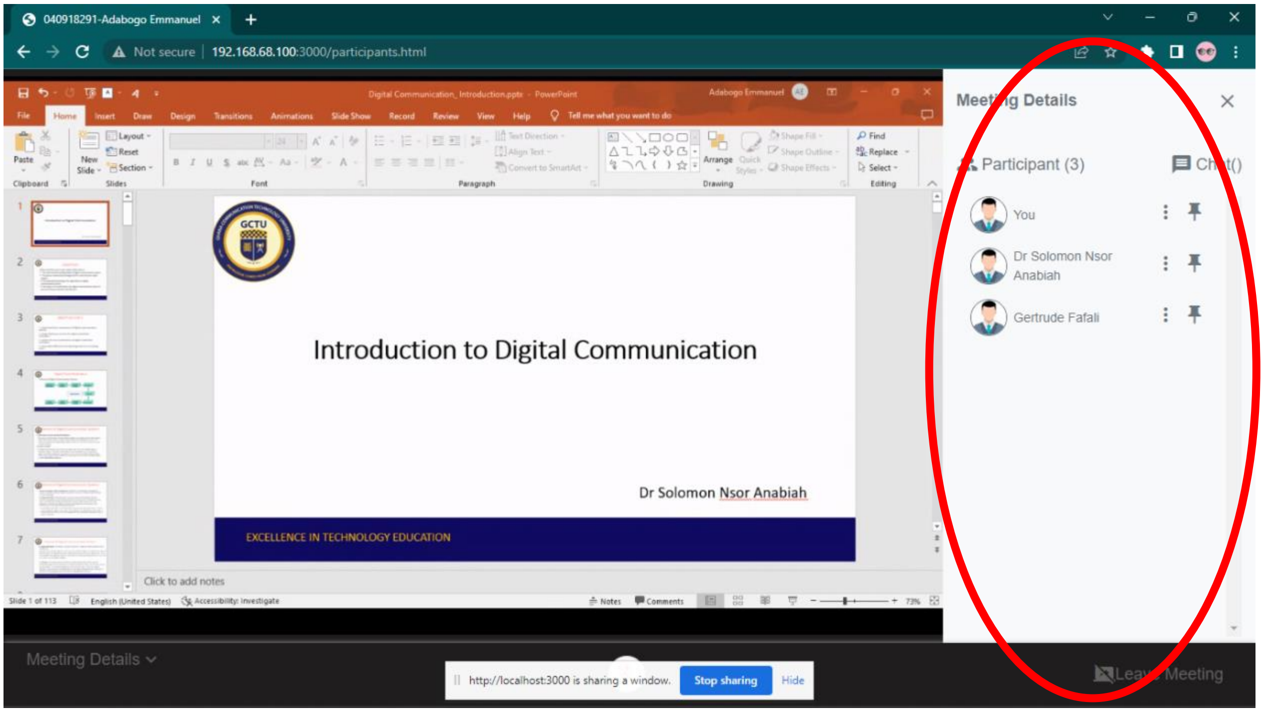

Figure 17 Presenter Meeting room (Adabogo, Amaduazi. 2022)

4.3.2 Participants Mobile View

The Mobile view of the participant panel is shown below,

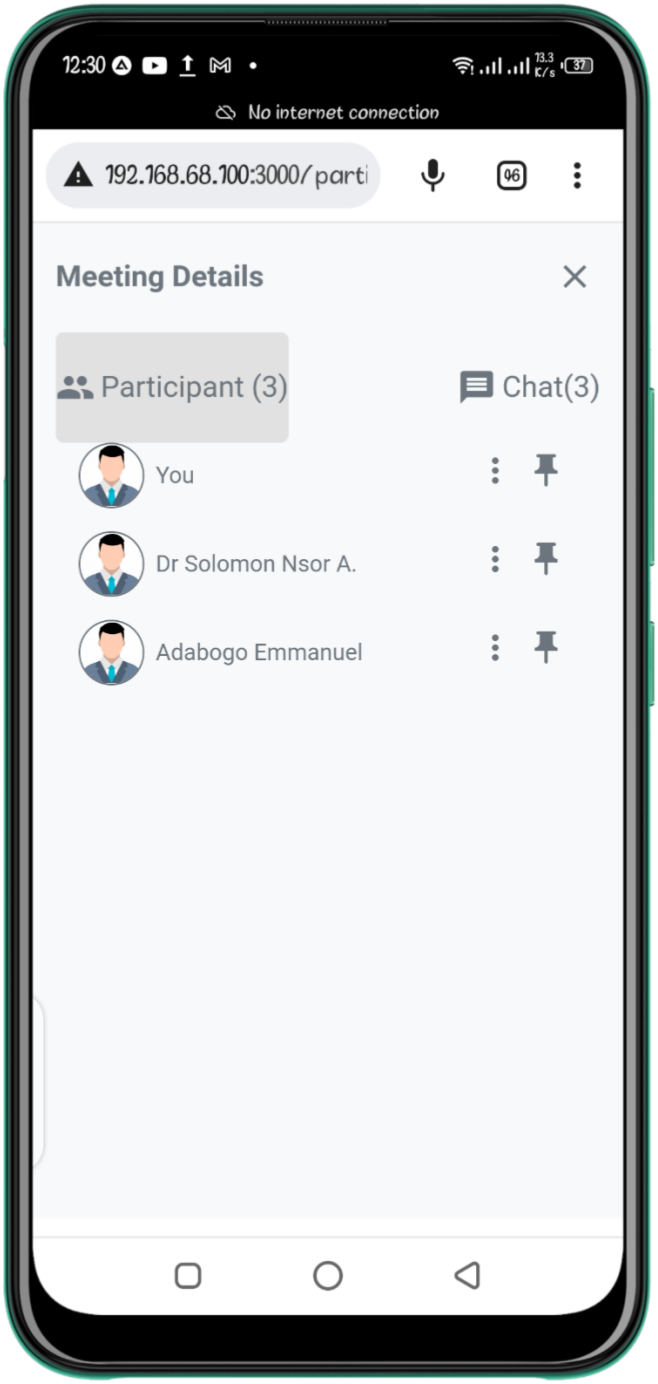

Figure 18 Mobile Participant Pane (Adabogo, Amaduazi. 2022)

4.3.3 Messaging Panel

The message section in the Meeting detail pane allows both participants and presenters to share text messages. The views of the messaging panels are listed below.

4.3.2.1 Personal Computer View of the Messaging Panel

The figure below shows the interface of the messaging panel on a personal computer.

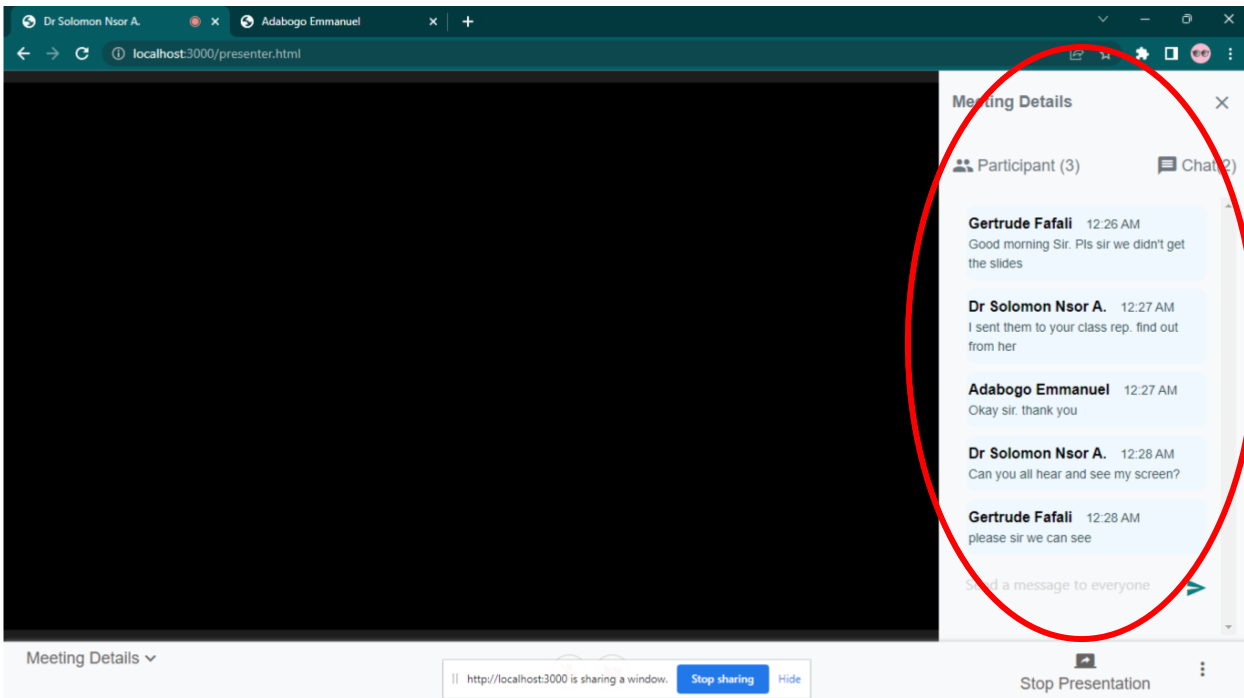

Figure 19 Messaging Panel PC View (Adabogo, Amaduazi. 2022)

4.3.2.2 Mobile View of the Messaging Panel

The figure below shows the interface of the messaging panel on a mobile device

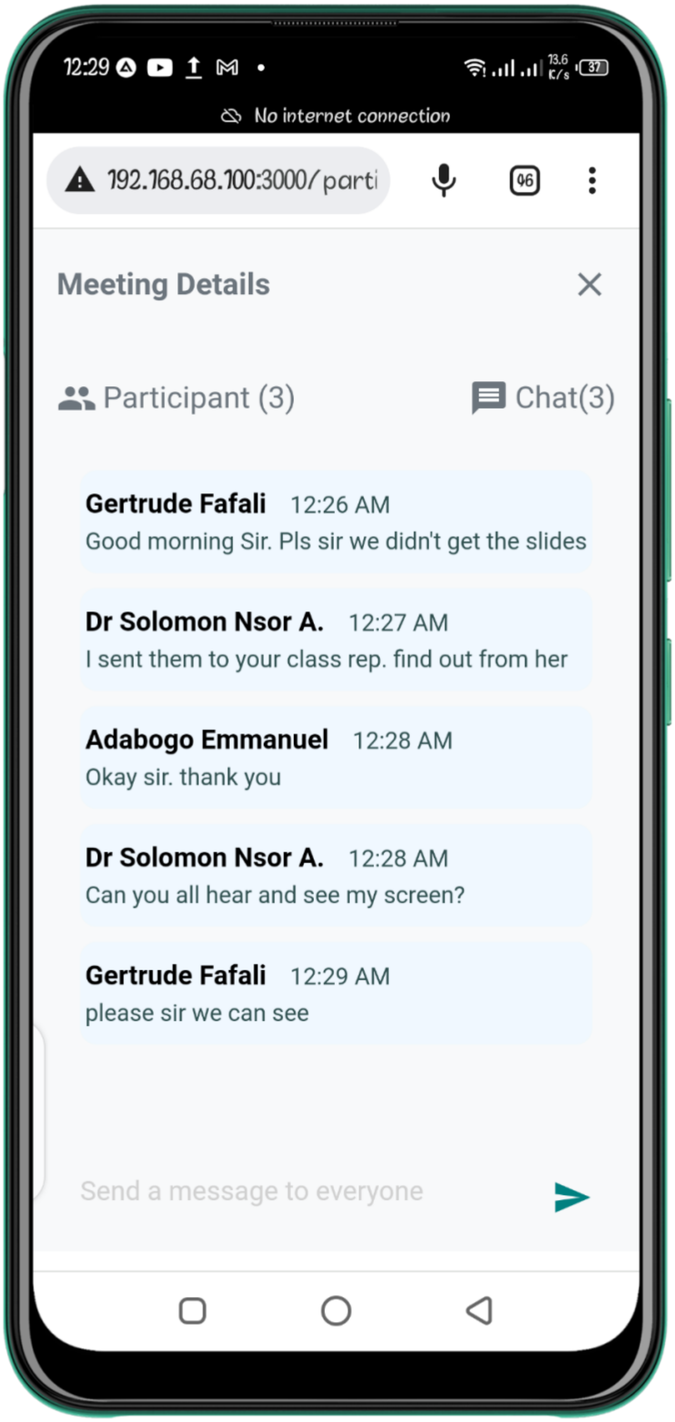

Figure 20 Messaging Panel Mobile View (Adabogo, Amaduazi. 2022)

4.4 Evaluation and testing

From the literature review, we were able to approach the project from the various works done by several authors. From those works we were able to modify and enhance this system's functionalities and performance. The system worked so well and video and audio streaming processing too worked so well. The file sharing and messaging system too worked so well. We were able to test the system by inviting students to join the virtual class with their phones and personal computers using a link. The computer running the server created a meeting with a link that was used for the other devices to join as participants. The response from them was satisfactory.

4.5 Results

After testing the overall performance of the signaling server, the presenter and participant interface responsibility was satisfactory. Both interface designs were reported to be very simple for navigation as well as easy to use. The messaging system was also reported to be useful as students can choose to ask questions through it rather than using audio. The average video/ audio streaming delay was recorded less than 300ms. The system could connect up to a maximum of 26 participants within a single 4G WIFI interface. Compared to the maximum number of 20 participants excluding audio that the systems in the review could connect, our system is scalable enough. In terms of functionality, the system in this project has extra functionality such as messaging system, file sharing system audio streaming. 34 people were able to connect with optimum efficiency and graphic quality through the 4x4 Multiple Input Multiple Output (MIMO) routers across campus. The project was tested by multiple participants and it yielded a satisfactory result.

4.5.1 Benefits of the system

With the result above and the objectives mentioned in chapter one of this project, the system in this research work is satisfactory and has achieved its objectives respectively. The project would benefit both students and lectures in the following ways.

4.6 Summary of Chapter

In this chapter, the design of the project is successfully completed. The project's video sharing and audio sharing capabilities were tested among individuals with devices such as laptops and mobile phones and the result was satisfactory. The text messaging functionality was also tested and the result was satisfactory too. The individuals or participants involved in the testing procedure also reported the user interface was simple and user friendly.

CHAPTER FIVE

CONCLUSION, LIMITATION AND RECOMMENDATION

5.0 Overview

This is the final chapter in this project work. In this chapter, we conclude and finalize the purpose and workings of this project work. We also make recommendations for researchers who are interested in making this project better in the future.

5.1 Conclusion

In this project work, we designed and implemented a desktop base screen sharing system to improve learning using the school Wireless Local Area Network (WLAN). The project was meant to improve learning by lowering data and other video conference subscription costs. Aside from the on campus real time virtual classroom learning and teaching functionality, this project also ensures virtual teaching between lectures and students in a classroom. The project has been successfully designed and implemented using the WebRTC technology. Up to 26 people could connect through a 4G Wifi with efficient system performance and good picture quality. 34 people were able to connect with the same efficiency and graphic quality through 4x4 Multiple Input Multiple Output (MIMO) routers across campus. The project was tested by multiple participants and it yielded a satisfactory result.

5.2 Limitation

The application in this project has a few limitations which are mentioned below,

1. The workings of the system in this project work are limited and can only be accessible through a WLAN network. Students on other campuses cannot join a class session due to this limitation.
2. The server operation is limited to desktop computers running the windows operating system.
3. A faster router is recommended to ensure more user connectivity.

5.3 Recommendation

Using WebRTC technology, this project is robust and the server uses less power during the screen sharing process. This is as a result of the frames being shared using peer-to-peer connection. Although the system has a satisfactory result, we have a few recommendations to suggest for researchers who would be interested in making this project work better.

1. The server system could be hosted on the internet to ensure users or students across the world could join a class instead of the limitation on a local network which is only accessible on the school WLAN.
2. The server system could be redesigned to be compatible on other operating systems such macOS and Android devices.
3. Functionalities such as video recording could also be included to ensure lectures can record sessions during real time virtual classes.

REFERENCES

1. Alissa Irei. (2020). What is real-time communications." [Daring]. Tersedia pada: What is Real_Time_Communications(RTC).https://www.techtarget.com/searchunifiedcommunications/definition/real-time-communications

2. A. Rahmatulloh, I. Darmawan and R. Gunawan, "Performance Analysis of Data Transmission on WebSocket for Real-time Communication," 2019 16th International Conference on Quality in Research (QIR): International Symposium on Electrical and Computer Engineering, 2019, pp. 1-5, doi: 10.1109/QIR.2019.8898135.

3. Ben Mulholland(2021). The 11 best screen sharing Apps (And how to use them), page. The 11 Best Screen Sharing Apps (And How to Use Them) | Process Street | Checklist, Workflow and SOP Software

4. CISCO, "Cisco Visual Networking Index : Forecast and Trends, 2017-2022," Tech. Rep., 2019.

5. D.Sivasankar, "Enhancing Quality of Service in Real Time Communication services for Network on Chip," vol. 9, no. 2, hal. 233–240, 2017.

6. D. Skvorc, M. Horvat, dan S. Srbljic,(2014) "Performance evaluation of Websocket protocol for implementation of full-duplex web streams," DOI:10.1109/MIPRO.2014.6859715

7. Dong, Jie & Xin, Jitao & Zhuang, Peng. (2015). A Study on H.264 Live Video Technology with Android System. 525. 123-132. 10.1007/978-3-662-47791-5_15.

8. FutureLearn, (2021). Learning subscription: the Education Trend of 2020. https://www.futurelearn.com/info/blog/learning-subscriptions-2020

9. G. R. Hiertz et al., "IEEE 802.11s: The WLAN Mesh Standard," in IEEE Wireless Communications, vol. 17, no. 1, pp. 104-111, February 2010, doi: 10.1109/MWC.2010.5416357.

10. Gajanayaka, Sidath. (2018). Projectron: Portable Embedded Device For Smart Screen Sharing Using Wireless Video Transmission. 10.13140/Rg.2.2.13971.53287.

11. Hitachi Digital Media Group, "5 benefits of teaching with an Interactive Projector,"Hitachi Digital Media Group, 2016. [Online]. Available: http://www.hitachidigitalmedia.com/en-gb/news/blog/5-benefits-of-teaching-with-an-interactive-projector

12. Huang, CY, Cheng, YC, Huang, GZ, Fan, CL & Hsu, CH (2021), 'On the Performance Comparisons of Native and Clientless Real-Time Screen-Sharing Technologies', *ACM Transactions on Multimedia Computing, Communications and Applications*, vol. 17, no. 2, 54. https://doi.org/10.1145/3437881

13. Irfan D. Darmawan, A. Rahmatulloh and R. Gunawan, "Real-time Screen Sharing Using Web Socket for Presenting Without Projector," 2019 7th International Conference on Information and Communication Technology (ICoICT), 2019, pp. 1-6, doi: 10.1109/ICoICT.2019.8835356.

14. Jakhar, D., Kaur, I., & Kaul, S. (2020). Screen mirroring, screen casting and screen sharing during COVID-19: what dermatologists should know. Clinical and experimental dermatology, 45(6), 750–751. https://doi.org/10.1111/ced.14247

15. K. Singh dan V. Krishnaswamy, “Building communicating web applications leveraging endpoints and cloud resource service,” , hal. 486–493, 2013.

16. Keerio, Imran & Lashari, Haque Nawaz & Rind, M. & Kumar, Kamlesh & Chhajro, M. & Maitlo, Abdullah. (2018). Comparative Study of Existing and Forthcoming WLAN

Technologies.18.https://www.researchgate.net/publication/325817830_Comparative_Study_of_Existing_and_Forthcoming_WLAN_Technologies

17. Manzor Ahmad. R., Vivek Bhatnagar. (2015) A comparative study of software development life cycle model. www.ijaiem.org ISSN 2391-4847

18. Micheal, S., Jennifer, W. M. L., & Matt Bower. (2022). Investigating the pedagogies o screen-sharing contemporary learning environment-A mixed methods analysis. DOI:10.1111/jcal.12647

19. Microsoft. (2020). Remote Desktop Protocol (Windows), *May 2017, [online] Available: https://msdn.microsoft.com/en-us/library/aa383015(v=vs.85).aspx*.

20. Morris, C., & Chikwa, G. (2014). Screencasts: How effective are they and how do students engage with them? Active Learning in Higher Education, 15(1), 25–37. https://doi.org/10.1177/1469787413514654

21. Othman, Fazzami & Yusoff, Zaharah & Abdul Rasam, Abdul Rauf. (2019). Isovist and Visibility Graph Analysis (VGA): Strategies to evaluate visibility along movement pattern for safe space. IOP Conference Series: Earth and Environmental Science. 385. 012024. 10.1088/1755-1315/385/1/012024.

22. Patil, Shweta & Nesaragi, Naimahmed & Kumar B, Pradeep & Ananda, C.M.. (2015). A novel approach for fiber channel based video protocol. 307-312. 10.1109/ICSTM.2015.7225432.

23. P. Wu, Z. LONG, L. ZHANG. (2013) Research and design of wireless projector gateway on Wi-Fi Direct [J]. Computer Engineering and Design, 06: 2223-2227.

24. Shu-dong, L. (2008). Development of Bootloader Based on ARM9 and Embedded Linux System. Microcomputer Information.

25. TeamViewer (2022) What Is Screen Sharing and How Does It Work?. Available https://www.teamviewer.com/en/info/screen-sharing/

26. Wesley Chai and Brien Posey (2022), Remote desktop protocol (RDP). Available: https://www.techtarget.com/searchenterprisedesktop/definition/Remote-Desktop-Protocol-RDP

27. Yang, Siyu & Li, Bin & Xu, Jizheng & Lu, Yan. (2018). A Hardware-Accelerated System for High Resolution Real-Time Screen Sharing. IEEE Transactions on Circuits and Systems for Video Technology. PP. 1-1. 10.1109/TCSVT.2018.2809690.

28. Mozilla Group(2022). WebRTC API. https://developer.mozilla.org/en-US/docs/Web/API/WebRTC_API https://study.com/academy/lesson/what-is-a-web-browser-definition-examples-quiz.html

29. Socket.IO term(2022). What Socket.IO is and it features. https://socket.io/docs/v4/#what-socketio-is

30. OpenJS Foundation (2022). What is Jquery .https://jquery.com/