**A Project On**

**Streaming Control System**



PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE AWARD OF THE DEGREE OF

**BACHELOR OF SCIENCE IN**

**(SOFTWARE ENGINEERING)**

**SUBMITTED BY**

|  |  |
| --- | --- |
| **NAME** | **ID** |
| GETACHEW SHAREW | 06353/10 |
| ZINABU MOLLA |  |
| SEGEN ATIKLT |  |

Under the Guidance of

Instructor Belsti Yenegeta

**Department of Software Engineering**

**College of Computing**

Debre Berhan University

Debre Berhan Ethiopia

**6/10/2014 EC**

**Declaration**

The Project is our own and has not been presented for a degree in any other university and all the sources of material used for the project/thesis have been regularly acknowledged. (Name and Signature up to the number of the project group members)

The material borrowed from other source and incorporated in the project has been duly acknowledged and /or referenced.

We understand that we ourselves could be held responsible and accountable for plagiarism, if any, detected later on.

**Name Signature**

1. Getachew Sharew …………………….
2. Zinabu Molla …………………….
3. Segen Atiklt …………………….

**College:** Computing College

**Program**: Software Engineering

------------------------------------- --------------------------------------------- --------------------

**Name of Adviser Signature Date**

It is approved that this project has been written in compliance with the formatting rules laid down by the school of the university.

**Acknowledgement**

Special thanks for our adviser, Instructor Belsti for continuous support in every phase of the project and who continuously provides us his valuable advice to prepare this project document. Lastly, we would like to acknowledge all our friends for their morals, idea, materials during the courses of the years.

**Tables of content**

**page No**

Table of Contents

[**List of Tables** vi](#_Toc105940112)

[**List of Figures** vi](#_Toc105940113)

[**Chapter One** 1](#_Toc105940114)

[**1.** **Introduction** 1](#_Toc105940115)

[**1.1 Background** 1](#_Toc105940116)

[**1.2 Statement of the problem** 2](#_Toc105940117)

[**1.3. Objective of the Project** 2](#_Toc105940118)

[**1.3.1 General Objective** 2](#_Toc105940119)

[**1.3.2 Specific Objective** 2](#_Toc105940120)

[**1.4. Scope of the Project** 3](#_Toc105940121)

[**1.5. Limitation of the Project** 4](#_Toc105940122)

[**1.6. Significance of the Project** 4](#_Toc105940123)

[**1.7. Methodology of the Project** 5](#_Toc105940124)

[**1.7.1 Data analysis methodology** 5](#_Toc105940125)

[**1.7.2 Implementation Methodology** 5](#_Toc105940126)

[**1.7.3 tool selection** 5](#_Toc105940127)

[**1.7.4 Design Methodology** 6](#_Toc105940128)

[**Chapter two** 7](#_Toc105940129)

[**2.Description of The Existing System** 7](#_Toc105940130)

[**2.1. Introduction** 7](#_Toc105940131)

[**2.2 Players in the existing** 7](#_Toc105940132)

[**2.3 Business rules** 7](#_Toc105940133)

[**2.4 Problems of the current system** 7](#_Toc105940134)

[**2.5 Practice to be preserved from existing system** 8](#_Toc105940135)

[**2.5.1. Proposed System** 8](#_Toc105940136)

[**2.6 Functional Requirements** 8](#_Toc105940137)

[**2.7 Non-functional requirements** 9](#_Toc105940138)

[**Chapter Three** 10](#_Toc105940139)

[**3. System Analysis** 10](#_Toc105940140)

[**3.1 Introduction** 10](#_Toc105940141)

[**3.2. user constraints** 10](#_Toc105940142)

[**3.3. System Use case** 10](#_Toc105940143)

[**3.3.1. Use case modeling** 11](#_Toc105940144)

[**3.3.2. Use Case Description** 12](#_Toc105940145)

[**3.3.3. Activity Diagram** 26](#_Toc105940146)

[**3.3.4. Class Diagram** 41](#_Toc105940147)

[**3.3.5 Sequence Diagram** 44](#_Toc105940148)

[**3.3.6 User Interface Design** 59](#_Toc105940149)

[**3.4. Business rules** 63](#_Toc105940150)

[**3.5 state chart diagram** 63](#_Toc105940151)

[**Chapter 4** 71](#_Toc105940152)

[**4.1 System Design** 71](#_Toc105940153)

[**4.2 Architectural Design** 71](#_Toc105940154)

[**4.3 Deployment Diagram** 72](#_Toc105940155)

[**4.4 Sub System Decomposition** 73](#_Toc105940156)

[**4.5 Component Diagram** 74](#_Toc105940157)

# **List of Tables**

[Table 1: Documentation - Lock the computer 16](#_Toc105671252)

[Table 2 Documentation - Unlock the computer 17](#_Toc105671253)

[Table 3: Documentation - Shutdown the computer 18](#_Toc105671254)

[Table 4: Documentation - Restart the computer 19](#_Toc105671255)

[Table 5: Documentation - Scroll Up Computer content 20](#_Toc105671256)

[Table 6:Documentation - Scroll Down Computer content 21](#_Toc105671257)

[Table 7: Documentation - Volume Adjustment: VLC 22](#_Toc105671258)

[Table 8: Documentation - Fetch Computer Hard Drive 23](#_Toc105671259)

[Table 9:Documentation- Pause Vlc 24](#_Toc105671260)

[Table 10:Documentation- Resume Vlc 25](#_Toc105671261)

[Table 11:Documentation- Seek Forward Vlc 26](#_Toc105671262)

[Table 12:Documentation- Seek Backward Vlc 27](#_Toc105671263)

[Table 13:Documentation- Play Next 28](#_Toc105671264)

[Table 14:Documentation- Play Previous 29](#_Toc105671265)

# **List of Figures**

[Figure 1:Use case Diagram model 11](file:///C:\Users\Stu\3D%20Objects\version222Project.docx#_Toc105939985)

[Figure 2:Activity Diagram- Lock the computer 27](#_Toc105939986)

[Figure 3:Activity Diagram- Unlock the computer 28](#_Toc105939987)

[Figure 4:Activity Diagram- Shut down the computer 29](#_Toc105939988)

[Figure 5:Activity Diagram- Restart the computer 30](#_Toc105939989)

[Figure 6:Activity Diagram- Fetch the computer Hard drive 31](#_Toc105939990)

[Figure 7:Activity Diagram- Scroll Up the Computer Content 32](#_Toc105939991)

[Figure 8:Activity Diagram- Scroll Down the Computer Content 33](#_Toc105939992)

[Figure 9:Activity Diagram- Adjust the Volume VLC media Player 34](#_Toc105939993)

[Figure 10:Activity Diagram- Pause VLC 35](#_Toc105939994)

[Figure 11:Activity Diagram- Resume VLC 36](#_Toc105939995)

[Figure 12:Activity Diagram- Seek Forward VLC 37](#_Toc105939996)

[Figure 13:Activity Diagram- Seek Backward VLC 38](#_Toc105939997)

[Figure 14:Activity Diagram- Play Next 39](#_Toc105939998)

[Figure 15:Activity Diagram- Play Previous 40](#_Toc105939999)

[**Figure 16: Class diagram** – Control VLC 42](#_Toc105940000)

[**Figure 17:** Class Diagram – Control the Computer 43](#_Toc105940001)

[Figure 18:Sequence Diagram – Lock the Computer 45](file:///C:\Users\Stu\3D%20Objects\version222Project.docx#_Toc105940002)

[Figure 19:Sequence Diagram – Unlock the Computer 46](file:///C:\Users\Stu\3D%20Objects\version222Project.docx#_Toc105940003)

[Figure 20:Sequence Diagram – Shut down the computer 47](file:///C:\Users\Stu\3D%20Objects\version222Project.docx#_Toc105940004)

[Figure 21:Sequence Diagram – Restart the Computer 48](#_Toc105940005)

[Figure 22:Sequence Diagram – Scroll Down the Pdf, Ppt, Word 49](#_Toc105940006)

[Figure 23: Sequence Diagram – Scroll Up Ppt, Pdf, Word 50](#_Toc105940007)

[Figure 24:Sequence Diagram – Seek Backward VLC 51](file:///C:\Users\Stu\3D%20Objects\version222Project.docx#_Toc105940008)

[Figure 25: Sequence Diagram – Seek Forward VLC 52](file:///C:\Users\Stu\3D%20Objects\version222Project.docx#_Toc105940009)

[Figure 26: Sequence Diagram – Volume adjustment of VLC 53](#_Toc105940010)

[Figure 27: Sequence Diagram – Fetch the computer Data 54](#_Toc105940011)

[Figure 28: Sequence Diagram – Play next 55](#_Toc105940012)

[Figure 29: Sequence Diagram – Vlc Pause 56](#_Toc105940013)

[Figure 30: Sequence Diagram – Play previous 57](#_Toc105940014)

[Figure 31: Sequence Diagram – Resume VLC 58](#_Toc105940015)

[Figure 32: User interface - playlist 60](#_Toc105940016)

[Figure 33: User interface - files 61](file:///C:\Users\Stu\3D%20Objects\version222Project.docx#_Toc105940017)

[Figure 34: User interface – Drawer Page 62](file:///C:\Users\Stu\3D%20Objects\version222Project.docx#_Toc105940018)

[Figure 35: State Diagram – Lock the computer 64](#_Toc105940019)

[Figure 36: State Diagram – Un Lock the computer 64](file:///C:\Users\Stu\3D%20Objects\version222Project.docx#_Toc105940020)

[Figure 37: State Diagram – shutdown the computer 65](file:///C:\Users\Stu\3D%20Objects\version222Project.docx#_Toc105940021)

[Figure 38: State Diagram – Restart the computer 65](file:///C:\Users\Stu\3D%20Objects\version222Project.docx#_Toc105940022)

[Figure 39: State Diagram – Scroll down the ppt, pdf, word on computer 66](file:///C:\Users\Stu\3D%20Objects\version222Project.docx#_Toc105940023)

[Figure 40: State Diagram – Scroll up the Ppt, Pdf, Word on computer 66](file:///C:\Users\Stu\3D%20Objects\version222Project.docx#_Toc105940024)

[Figure 41: State Diagram – fetch the computer files 67](#_Toc105940025)

[Figure 42: State Diagram – Vlc seek backward 67](file:///C:\Users\Stu\3D%20Objects\version222Project.docx#_Toc105940026)

[Figure 43: State Diagram – Vlc seek forward 68](file:///C:\Users\Stu\3D%20Objects\version222Project.docx#_Toc105940027)

[Figure 44: State Diagram – Vlc volume adjustment 68](file:///C:\Users\Stu\3D%20Objects\version222Project.docx#_Toc105940028)

[Figure 45: State Diagram – VLC play next 69](#_Toc105940029)

[Figure 46: State Diagram – VLC play previous 69](#_Toc105940030)

[Figure 47: State Diagram – VLC pause 70](#_Toc105940031)

[Figure 48: State Diagram – VLC resumed 70](#_Toc105940032)

[Figure 49: Architectural Design 71](#_Toc105940033)

[Figure 50: Deployment Diagram 72](file:///C:\Users\Stu\3D%20Objects\version222Project.docx#_Toc105940034)

[Figure 51: Sub-System-Decomposition 73](#_Toc105940035)

[Figure 52: Component Diagram 74](#_Toc105940036)

# **Chapter One**

# **Introduction**

Remote control is a device controlling another device such as computer, television, radio, video, recording device. Remote control can be operating via infrared signal, Wi-Fi, Bluetooth. Remote control may control a variety of things such as, volume, channel, software such as media player, word process, presentation, pdf etc. in general we can explain remote as a clicker, flipper, changer. Streaming control system is an automated control system design to control the basic functionality of our computer lock, unlock, shutdown, restart and that enables the users who uses VLC media player to control the necessity features of the media player. This automated system also supports the users to manage the whole disks of their computer via their mobile.

## **1.1 Background**

A remote control is a component device used to operate computer, TV sets, multimedia players, and machine through wireless connection. Before remote control TV watcher had to plod to their television to change the channel and volume using rotating dials or buttons in 1950. Electronic manufacturer *Zeniz* introduce the lazy bones remote, unfortunately it uses a long snaking cable that turned as many ankles as it did, channels. In the mid-1950, *Zeniz* engineer *Ugene* *Polly* devised his flashmatic TV remote which used directional flashes or light to control the system, in 1956 Polly college Robert Adler, created the space command control which employed high frequency, ultrasonic sound instead of light.

## **1.2 Statement of the problem**

So far, most of the user uses the system themselves to control the computer, to use Microsoft products and for streaming files to the computer such as handling the media player. Therefor it needs a remote control to handle all the operations above. Generally, the existing system have the following problems

* does not support controlling of the computer workstation.
* Does not support content scrolling.

## **1.3. Objective of the Project**

### **1.3.1 General Objective**

The general objective of the project is to develop streaming control system.

### **1.3.2 Specific Objective**

To achieve the above-mentioned general objective, the project includes the following specific objective:

* To Study the existing system and identifying the problems under the existing system.
* To make detailed the system elicitation
* To make a detailed system analysis
* To make a detailed system design
* Design a new system that can overcome the problem of the current system.
* To implement the new system.
* To Test the system and deploy it.

## **1.4. Scope of the Project**

Project scope is the part of project planning that involves determining and documenting a list of specific project goals, deliverable, features, functions, tasks. It is specifically concerned with what problem the new proposed system address. The scope of the project is listed below

* Lock the computer
* Unlock the computer
* Restart the computer
* Shutdown the computer
* Scrolling up and down editing and none editing software’s such as PowerPoint, Microsoft word and pdfs respectively.
* Controlling all the basic functionality of the most popular media player which is VLC media player such as play, pause, resume, stop, skip next, getting playlist etc.
* Fetching list of the computer hard drive in to mobile device.

## **1.5. Limitation of the Project**

The limitations of this project are:

* The unlock operation is not supported on windows platform due to Microsoft windows privacy policy.
* The system will scroll any foreground application.

## **1.6. Significance of the Project**

The project gives a way for automating controlling mechanism of the device. And

Some of the benefits of the projects are listed below:

* To build the basic functionality of the VLC player.
* To control workstation of the computer and scrolling contents.
* Give more freedom of movement for the user. It should work as long as you are connected to the server
* The system keeps operational safety for the user.
* To make ease of use.

## **1.7. Methodology of the Project**

### **1.7.1 Data analysis methodology**

For the system analysis and design, we would use object-oriented system analysis design. Object oriented programming language is based on class and objects. it is used to structure a program into a simple, reusable piece of code. The entire software program runs by having multiple objects interact with objects to create large programs. The goal of object-oriented approach is to make system elements more reusable, thus improving system quality and the productivity of systems analysis and design.

### **1.7.2 Implementation Methodology**

* For the server side we use python, java
* For the client side we use java as java can be both server and client side.

### **1.7.3 tool selection**

Tools and techniques that we are going to use on this system are:

* Android studio
* JDK
* Python environment
* Microsoft Visual Studio Code
* Adobe Photoshop
* Microsoft Word and Linux WPS writer.

### **1.7.4 Design Methodology**

A software design methodology provides a logical and systematic means of proceeding with the design process as well as a set of guidelines for decision-making. The design methodology provides a sequence of activities, and often uses a set of notations or diagrams. Among different design methodology we will develop our system application using OOP.

**Object oriented design Paradigm** (OOP) is approach to problem solving where all computations are carried out using objects. An object is a component of a program that knows how to perform certain actions and how to interact with other elements of the program. Objects are the basic unites of object-oriented programming.

**1.7.4 Feasibility Analysis**

Feasibility means answering questions relating to the utility and viability of the system that is going to be developed & it is the measure of how beneficial or practical is streaming control system to the user. To get user acceptance and making the system easily understandable and accessible the new system considers the following feasibilities: -

**1.7.5 Economic feasibility**

Economic feasibility determines the costs and benefits related to the developing project. It also identifies the resources that are needed for the project. So, economically our system needs

* Personal Computer
* Mobile Phone
* Paper used to print the documentation etc.

**1.7.6 Technical feasibility**

The technical feasibility will focus on gaining and understanding the present technical resources of the organization and their applicability to the expected need of the proposed system. It is also the measure of the practicality of a specific technical solution and the availability of technical resources. In technical feasibility we should notify that our new system can implement with current technology and also the user can simply use that technology. The purpose of technical feasibility assessment is to have an understanding about the capability of users to construct a system. The question here is: "Do we have the technical and application familiarity for the construction and running of the system?”. With respect to this project, the following factors should be taken in to consideration.

**Familiarity with the application and technology**

From technical feasibility aspect to develop proposed system, we are capability to develop such project. We have experience in all aspects of technology tools to develop such project. To develop proposed system, we are familiar with programming language, python, java and some other APIs as well as xml, JSON data.

**Compatibility:** the system will expect to be independent of any hardware or software version of any computer system, which indicates the system were supported by any operating system.

**1.7.7 Operational feasibility**

Refers to weather the proposed system can be implemented and operated with the staff the organization have and meet its intended purpose. From the user’s perspective our system fully operational feasible as it just requires some knowledge of computer. There are enough validations available so operator does not require any special technical knowledge. Some factors that our system to be operational feasible:

* It is user friendly interface
* It uses APIs
* It provides an easier access of the computer file via mobile.
* It has a better speed to send the command as long as it has high WIFI bandwidth.

**1.8 Target Beneficiary of the system**

1. Disk jockey (DJ): - The primary job of DJ is recording, playing, and managing music and videos. our system is developed to handle all the above functionalities So, the system is required by the DJ.
2. A Person who teaches a Video Tutorial: this person can also be the target beneficiary of our system. He/she can easily use that app.
3. A lecturer at college: any lecturer can use that app by scrolling the computer content remotely during his teaching time at class.
4. Owner of VLC: this is the least target beneficiary of the system because we only use his product.

**Requirement Specification**

A requirement specification, which is a set of documented requirements to be satisfied by a material, design, product, or service. As a requirement specification we use a Testing mythology.

After we completely develop our system starting from requirement gathering to the implementation, we would use Testing process. Some of the testing processes that we would use are:

1. Unit Testing: it tests each unit of components and sub-components of the system. So, our system should be testing each unites of tests to make sure that the software is bug free. But after testing cannot say that it would be completely free from bugs.
2. Integration Testing: it comes after unit testing is finished and tests modules of the system and its dependent modules. So, we would apply integration testing to test each of our module functionalities to make sure that they operate correctly as intended.
3. System Testing: it comes after integration testing and it is used to test the whole system specially its functionality. So, we would then apply system testing to check its full functionality of the system.

**Assumption And Constraint.**

From the user’s perspective There are some assumptions to be considered: -

* The user should know the basic computer skills.
* The user should be able to know about networking concept a little.
* The user should be able to use mobile app effectively.

In terms of Constraints, we can describe the project through: -

* Time: the project should be finished on time.
* Cost: our project should be finished within defined budget.
* Scope: our project should cover all the specified requirements functionality.

# **Chapter two**

## **2.Description of The Existing System**

### **2.1. Introduction**

The main reason for writing this Current requirement specification (SRS) document is to find out the problem of the existing system and then to propose the new system. The requirements that specified below will describe the functions that the software is responsible and packages that it should include. These requirements are used by the developers and should use for validating the final delivered system.

### **2.2 Players in the existing**

Currently the system has only one player who is a user (a person that uses the system).

### **2.3 Business rules**

The business rule of the existing system is only user should to have

* VLC media player
* Mobile Application

### **2.4 Problems of the current system**

The problems of the current system is a user can’t control the computer instead it only controls the VLC media player. Some of the problems of the current system is that:

* Does not support content scrolling.
* Does not support lock and unlock the computer.
* Does not support restart and shutdown.
* Does not support opening any other file without music and video.

### **2.5 Practice to be preserved from existing system**

We can’t judge that the current system is completely problematic. There are a lot of best practices that needs to be preserved. To mention them:

* Fetching the computer hard-drive
* Pause and resume the current playing audio
* Adjusting the volume of the VLC
* Seeking the position of the audio and so on.

## **2.5.1. Proposed System**

The proposed system (Streaming control system) is well planned and designed than the previous system as it contains a bunch of operations are performed over the current system and planned to simplify some of the most interested operations of users. It is designed based on object-oriented programming as OOP is easy and powerful tool to implement and use it. Some of the extended features are listed below:

* Scrolling computer content.
* Lock and unlock the computer workstation.
* Trigger a screen-shot to the foreground activity.
* Not only streaming audio and video VLC but also opening any URI based on XDG-OPEN API.

# **2.6 Functional Requirements**

Functional requirements for mobile applications describe what specifically needs to be implemented in a particular system or product and what actions users have to take to interact with the software. They determine what the system should do.

Functional requirements describe the relations between the mobile application and the user or the environment. Here the ‘relations’ means the direct or indirect interactions between the user and the mobile app. Hence the system application has the following functional requirements.

* Lock the computer
* Unlock the computer
* Restart the computer
* Shut down the computer
* Scrolling up and down
* Fetch all remote computer hard drive
* Controlling all the functionality of the VLC media player

# **2.7 Non-functional requirements**

**Non-functional requirements:** These are basically the quality constraints that the system must satisfy according to the requirement specified. The priority or extent to which these factors are implemented varies from one project to other. They are also called non-behavioral requirements.

Some of the basic non-functional requirements of the proposed system are:

* **Portability**: our system should be portable on different environment such as Linux and windows.
* **Maintainability**: our system should be maintainable in case of any bugs or failure are available.
* **Scalability**: our system should be scalable if any additional features are needed to extend the system.
* **Performance**: our system should be fast under reliable local connection.
* **Quality**: based on the above-mentioned assumptions such as portability, performance our system will be qualitied.

# **Chapter Three**

## **3. System Analysis**

### **3.1 Introduction**

**A system analysis** is a process of collecting and interpreting facts, identifying the problems, and decomposition of a system into its components. System analysis is conducted for the purpose of studying a system or its parts in order to identify its objective. It is a problem-solving technique that improves the system and ensures that all the components of the system work efficiently to accomplish their purpose.

### **3.2. user constraints**

* The graphical user interface (GUI) supports only English language.
* In order to use the system, the user must have both the desktop application (Server) and the client-side application which is (mobile app).
* The user must first connect both the computer and the app first.

### **3.3. System Use case**

**Use-Case** describes the complete follow of activities that provide value to the user. Use case represents what the users want our system to do for them. A use case is the specific way of using the system by using some part of the functionality, a use case is thus a special sequence of related transactions performed by a user and a system. each use case is a complete course of events in the system from users respective.

# **3.3.1. Use case modeling**

**Use-case model** is defined as a model which is used to show how users interact with the system in order to solve a problem. As such, the use case model defines the user's objective, the interactions between the system and the user, and the system's behavior required to meet these objectives. Various model elements are contained in use-case model, such as actors, use cases, and the association between them. So, we use a use-case diagram to graphically portray a subset of the model in order to make the communication simpler

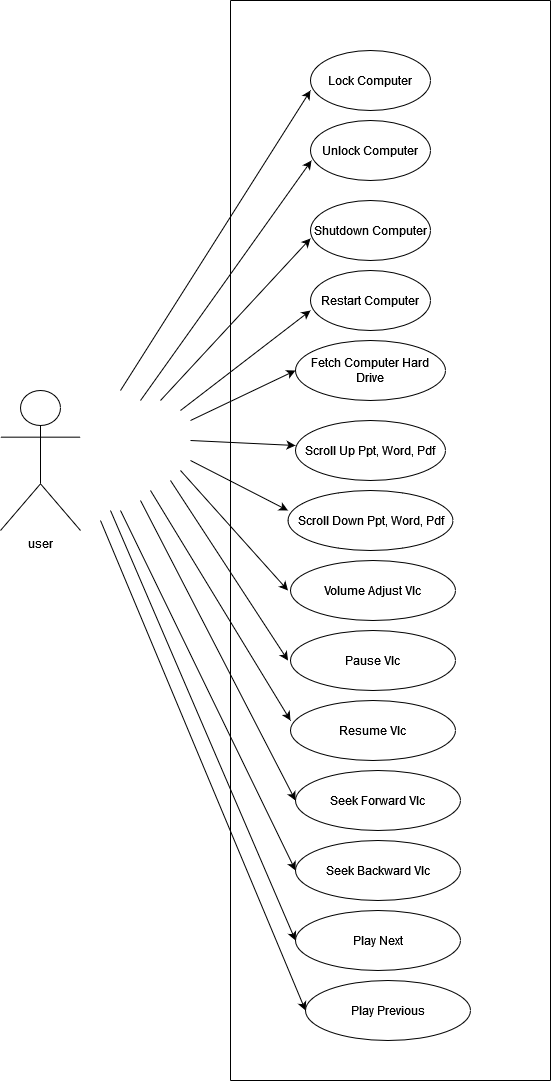


Figure 1:Use case Diagram model

|  |  |  |
| --- | --- | --- |
| **Use case name** | Lock the Computer | |
| **User** | Any user who wants use the application | |
| **Description** | the user can lock the computer remotely. | |
| **Precondition** | The connection between the mobile and computer must establish first. | |
| **Basic course of action** | **User action** | **System response** |
|  | 1.The user opens the mobile App. 3. The user clicks on ‘Control Tab’  5. the user clicks the button “Lock”.  7. End use case | 2. The system will display the main activities.  4.The system will display the control tab layout content.  6.The system will lock the computer. |
| **Alternative course of action** | Check if the connection between client and server is broken. | |

# **3.3.2. Use Case Description**

Table 1: Documentation - Lock the computer

|  |  |  |
| --- | --- | --- |
| **Use case name** | Unlock the Computer | |
| **User** | Any user who wants use the application | |
| **Description** | the user can unlock the computer remotely. | |
| **Precondition** | The connection between the mobile and computer must establish first. | |
| **Basic course of action** | **User action** | **System response** |
|  | 1.The user opens the mobile App. 3. The user clicks on ‘Control Tab’  5. the user clicks the button “Unlock”.  7. End use case | 2. The system will display the main activities.  4.The system will display the control tab layout content.  6.The system will Unlock the computer. |
| **Alternative course of action** | Check if the connection between client and server is broken. | |

Table 2 Documentation - Unlock the computer

|  |  |  |
| --- | --- | --- |
| **Use case name** | Shutdown the Computer | |
| **User** | Any user who wants use the application | |
| **Description** | the user can Shut down the computer remotely. | |
| **Precondition** | The connection between the mobile and computer must establish first. | |
| **Basic course of action** | **User action** | **System response** |
|  | 1.The user opens the mobile App. 3. The user clicks on ‘Control Tab’  5. the user clicks the button “Shutdown”.  7. End use case | 2. The system will display the main activities.  4.The system will display the control tab layout content.  6.The system will Shut down the computer. |
| **Alternative course of action** | If the system does not operate it’s intended functionality, check if the connection between client and server is broken or not. | |

Table 3: Documentation - Shutdown the computer

|  |  |  |
| --- | --- | --- |
| **Use case name** | Restart computer | |
| **User** | Any user who wants use the application | |
| **Description** | the user can restart the computer remotely. | |
| **Precondition** | The connection between the mobile and computer must establish first. | |
| **Basic course of action** | **User action** | **System response** |
|  | 1.The user opens the mobile App. 3. The user clicks on ‘Control Tab’  5. the user clicks the button “Restart”.  7. End use case | 2. The system will display the main activities.  4.The system will display the control tab layout content.  6.The system will Restart the computer. |
| **Alternative course of action** | If the system does not operate it’s intended functionality, check if the connection between client and server is broken or not. | |

Table 4: Documentation - Restart the computer

|  |  |  |
| --- | --- | --- |
| **Use case name** | Scroll up Pdf, Ppt, Word | |
| **User** | Any user who wants use the application | |
| **Description** | the user can scroll Pdf, Ppt, Word on computer remotely. | |
| **Precondition** | The connection between the mobile and computer must establish first. | |
| **Basic course of action** | **User action** | **System response** |
|  | 1.The user opens the mobile App. 3. The user clicks on ‘Control Tab’  5. the user clicks the button “Scroll up”.  7. End use case. | 2. The system will display the main activities.  4.The system will display the control tab layout content.  6.The system will scroll the computer content |
| **Alternative course of action** | If the system does not operate it’s intended functionality, check if the connection between client and server is broken or not. | |

Table 5: Documentation - Scroll Up Computer content

|  |  |  |
| --- | --- | --- |
| **Use case name** | Scroll Down Pdf, Ppt, Word | |
| **User** | Any user who wants use the application | |
| **Description** | the user can Scroll Down Pdf, Ppt, Word via his/her own mobile phone remotely. | |
| **Precondition** | The connection between the mobile and computer must establish first. | |
| **Basic course of action** | **User action** | **System response** |
|  | 1.The user opens the mobile App. 3. The user clicks on ‘Control Tab’  5. the user clicks the button “Scroll down”.  7. End use case. | 2. The system will display the main activities.  4.The system will display the control tab layout content.  6.The system will scroll down the computer content. |
| **Alternative course of action** | If the system does not operate it’s intended functionality, check if the connection between client and server is broken or not. | |

Table 6:Documentation - Scroll Down Computer content

|  |  |  |
| --- | --- | --- |
| **Use case name** | Volume Adjustment VLC | |
| **User** | Any user who wants use the application | |
| **Description** | the user can adjust the volume of remotely playing VLC media player. | |
| **Precondition** | The connection between the mobile and computer must establish first. | |
| **Basic course of action** | **User action** | **System response** |
|  | 1.The user opens the mobile App. 3. The user clicks on ‘Playlist tab’  5. the user can increase the volume using the progress bar.  7. The user decreases the volume using progress bar.  9. End use case. | 2. The system will display the main activities.  4.The system will display the Playlist tab layout content.  6.The VLC media player will increase its volume  8. The VLC media player will decrease its volume |
| **Alternative course of action** | If the system does not operate it’s intended functionality, check if the connection between client and server is broken or not. | |

Table 7: Documentation - Volume Adjustment: VLC

|  |  |  |
| --- | --- | --- |
| **Use case name** | Fetches Computer Hard Drive | |
| **User** | Any user who wants use the application | |
| **Description** | the user can fetch all computer hard drive files on his/her mobile phone remotely. | |
| **Precondition** | The connection between the mobile and computer must establish first. | |
| **Basic course of action** | **User action** | **System response** |
|  | 1.The user opens the mobile App. 3. The user clicks on ‘Files tab’  5. End use case. | 2. The system will display the main activities.  4.The system will display the computer’s hard drive list. |
| **Alternative course of action** | If the system does not operate it’s intended functionality, check if the connection between client and server is broken or not. | |

Table 8: Documentation - Fetch Computer Hard Drive

|  |  |  |
| --- | --- | --- |
| **Use case name** | Pause Vlc | |
| **User** | Any user who wants use the application | |
| **Description** | the user can Pause Vlc via his/her mobile phone remotely. | |
| **Precondition** | The connection between the mobile and computer must establish first. | |
| **Basic course of action** | **User action** | **System response** |
|  | 1.The user opens the mobile App. 3. The user clicks on ‘Play list tab’  5. the user clicks on pause icon.  7. End use case. | 2. The system will display the main activities.  4.The system will display the playlist tab layout content.  6.The VLC media player will pause. |
| **Alternative course of action** | If the system does not operate it’s intended functionality, check if the connection between client and server is broken or not. | |

Table 9:Documentation- Pause Vlc

# 

|  |  |  |
| --- | --- | --- |
| **Use case name** | Resume Vlc | |
| **User** | Any user who wants use the application | |
| **Description** | the user can Resume computer Vlc via his/her mobile phone remotely. | |
| **Precondition** | The connection between the mobile and computer must establish first. | |
| **Basic course of action** | **User action** | **System response** |
|  | 1.The user opens the mobile App. 3. The user clicks on ‘Play list tab’  5. the user clicks on Resume icon.  7. End use case. | 2. The system will display the main activities.  4.The system will display the playlist tab layout content.  6.The VLC media player will resume. |
| **Alternative course of action** | If the system does not operate it’s intended functionality, check if the connection between client and server is broken or not. | |

Table 10:Documentation- Resume Vlc

|  |  |  |
| --- | --- | --- |
| **Use case name** | Seek Forward position of Vlc playing music | |
| **User** | Any user who wants use the application | |
| **Description** | the user can Seek Forward the computer Vlc via his/her mobile phone remotely. | |
| **Precondition** | The connection between the mobile and computer must establish first. | |
| **Basic course of action** | **User action** | **System response** |
|  | 1.The user opens the mobile App. 3. The user clicks on ‘Play list tab’  5. the user clicks on Seek Forward icon.  7. End use case. | 2. The system will display the main activities.  4.The system will display the playlist tab layout content.  6.The VLC media player will seek forward. |
| **Alternative course of action** | If the system does not operate it’s intended functionality, check if the connection between client and server is broken or not. | |

Table 11:Documentation- Seek Forward Vlc

|  |  |  |
| --- | --- | --- |
| **Use case name** | Seek Backward positions of Vlc playing music | |
| **User** | Any user who wants use the application | |
| **Description** | the user can Seek Backward computer Vlc via his/her mobile phone remotely. | |
| **Precondition** | The connection between the mobile and computer must establish first. | |
| **Basic course of action** | **User action** | **System response** |
|  | 1.The user opens the mobile App. 3. The user clicks on ‘Play list tab’  5. the user clicks on Seek Backward icon.  7. End use case. | 2. The system will display the main activities.  4.The system will display the playlist tab layout content.  6. The VLC media player will seek backward. |
| **Alternative course of action** | If the system does not operate it’s intended functionality, check if the connection between client and server is broken or not. | |

Table 12:Documentation- Seek Backward Vlc

|  |  |  |
| --- | --- | --- |
| **Use case name** | Play Next | |
| **User** | Any user who wants use the application | |
| **Description** | the user can play next audio or video that the computer Vlc via his/her mobile phone. | |
| **Precondition** | The connection between the mobile and computer must establish first. | |
| **Basic course of action** | **User action** | **System response** |
|  | 1.The user opens the mobile App. 3. The user clicks on ‘Play list tab’  5. the user clicks on Play next icon.  7. End use case. | 2. The system will display the main activities.  4.The system will display the playlist tab layout.  6.The VLC media player will play next |
| **Alternative course of action** | If the system does not operate it’s intended functionality, check if the connection between client and server is broken or not. | |

Table 13:Documentation- Play Next

|  |  |  |
| --- | --- | --- |
| **Use case name** | Play Previous | |
| **User** | Any user who wants use the application | |
| **Description** | the user can play previous audio or video that the computer Vlc via his/her mobile phone. | |
| **Precondition** | The connection between the mobile and computer must establish. | |
| **Basic course of action** | **User action** | **System response** |
|  | 1.The user opens the mobile App. 3. The user clicks on ‘Play list tab’  5. the user clicks on Play Previous icon.  7. End use case. | 2. The system will display the main activities.  4.The system will display the playlist tab layout.  6. The VLC media player will play previous |
| **Alternative course of action** | If the system does not operate it’s intended functionality, check if the connection between client and server is broken or not. | |

Table 14:Documentation- Play Previous

# **3.3.3. Activity Diagram**

* Activity diagram is another important diagram in UML to describe the dynamic aspects of the system.
* Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.
* The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc.

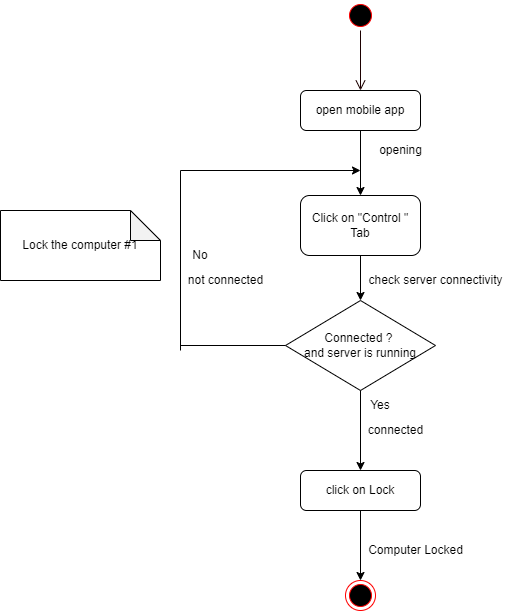


Figure 2:Activity Diagram- Lock the computer

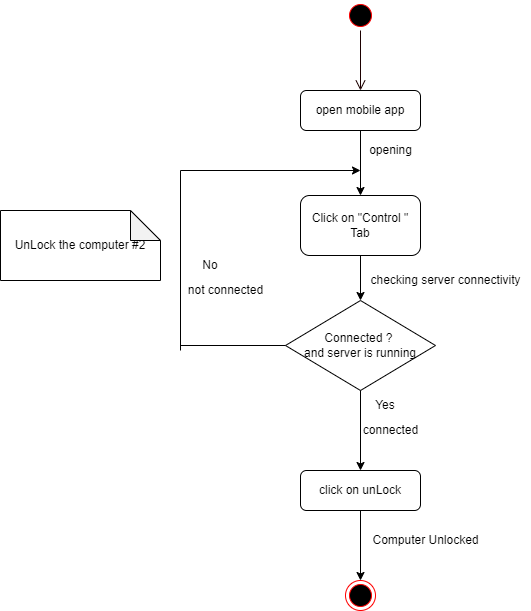


Figure 3:Activity Diagram- Unlock the computer

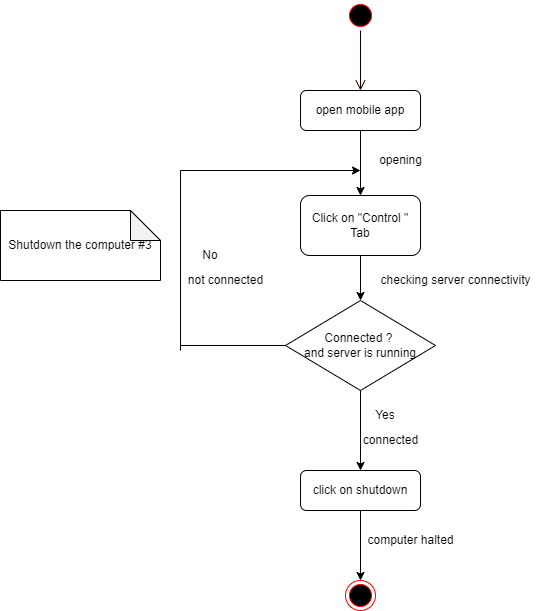


Figure 4:Activity Diagram- Shut down the computer

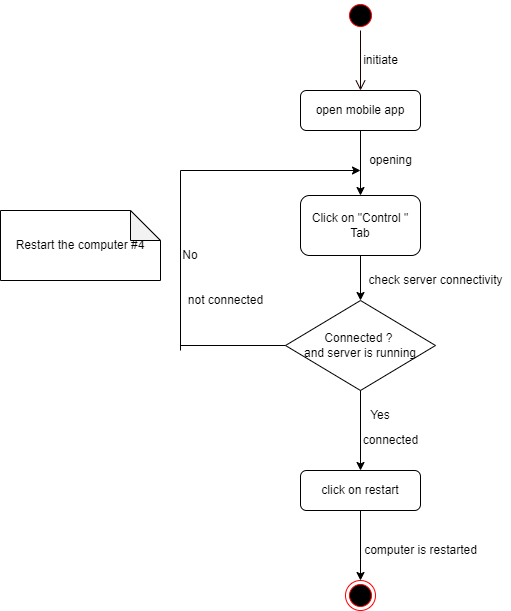


Figure 5:Activity Diagram- Restart the computer

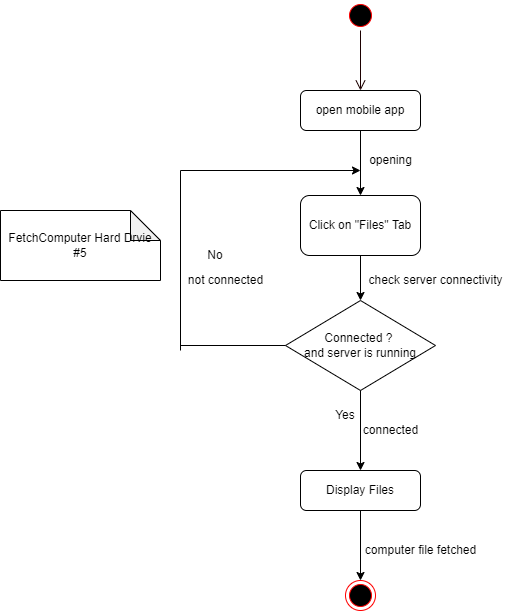


Figure 6:Activity Diagram- Fetch the computer Hard drive

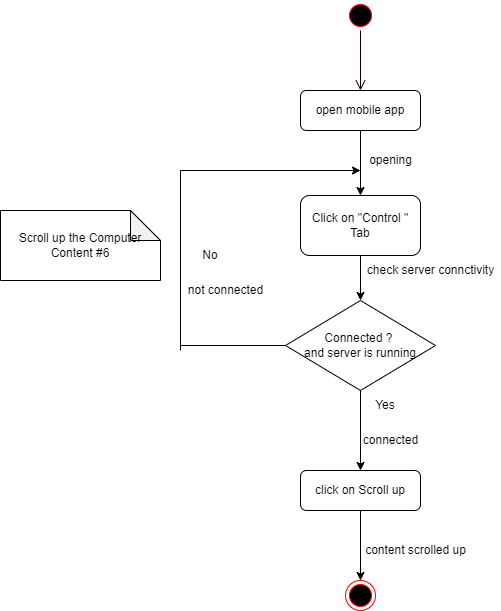


Figure 7:Activity Diagram- Scroll Up the Computer Content

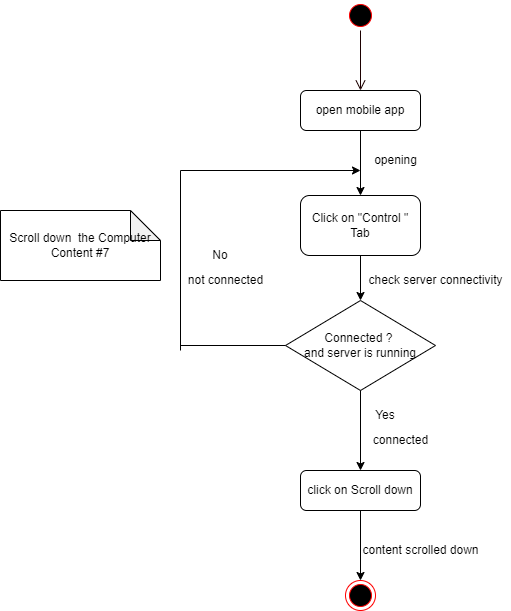


Figure 8:Activity Diagram- Scroll Down the Computer Content

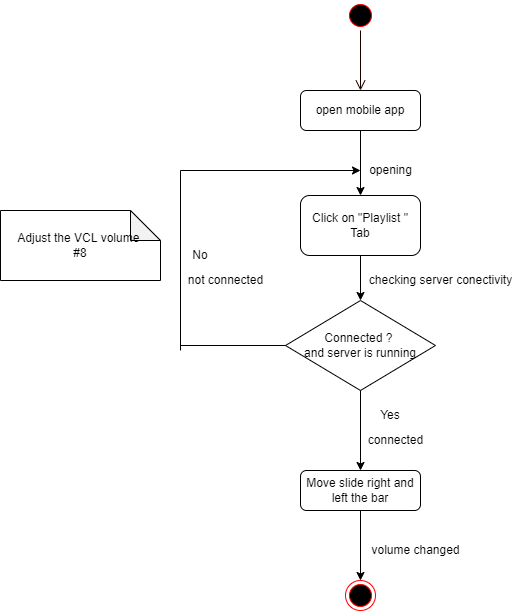


Figure 9:Activity Diagram- Adjust the Volume VLC media Player

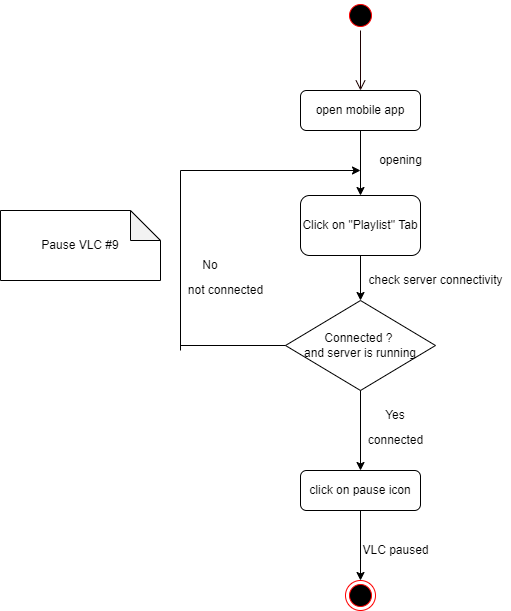


Figure 10:Activity Diagram- Pause VLC

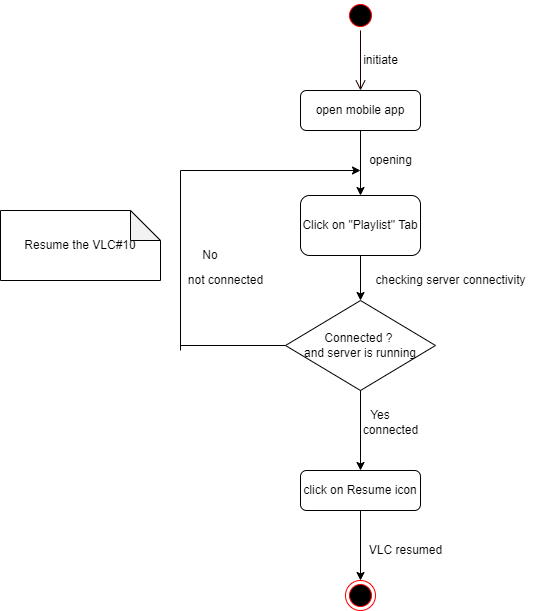


Figure 11:Activity Diagram- Resume VLC

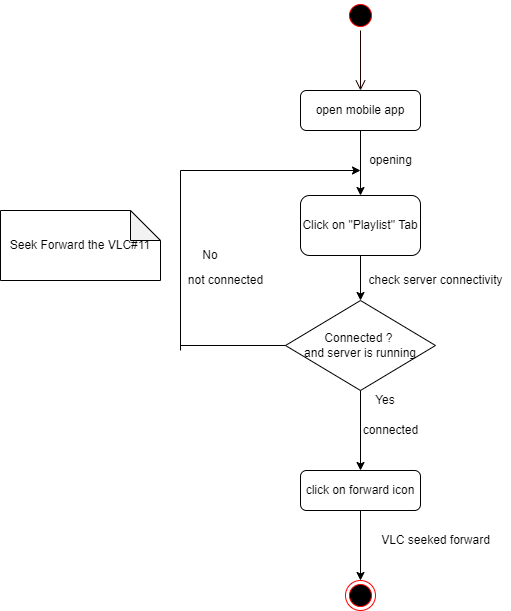


Figure 12:Activity Diagram- Seek Forward VLC

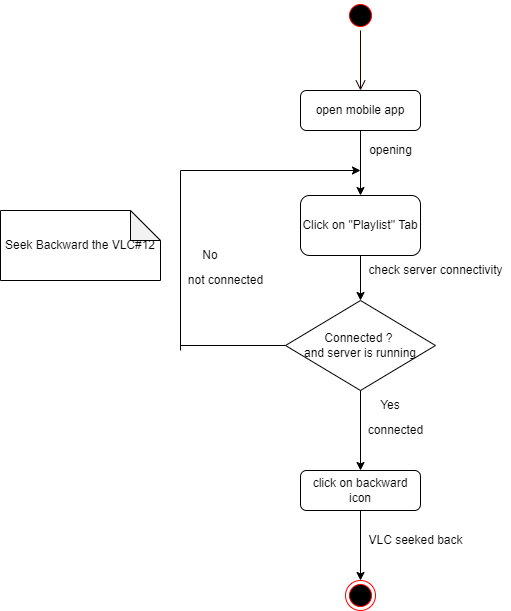


Figure 13:Activity Diagram- Seek Backward VLC

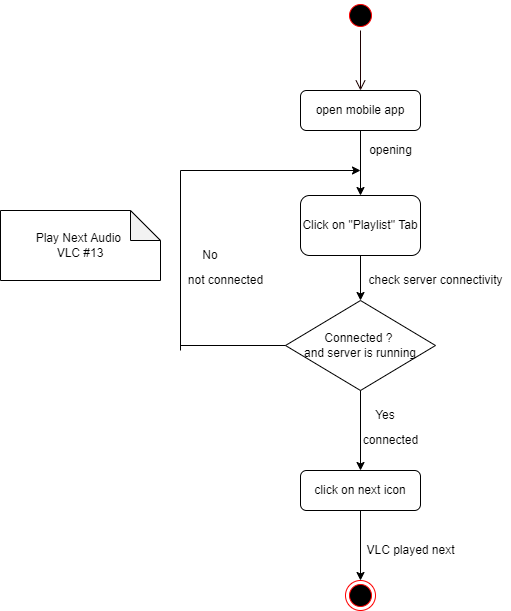


Figure 14:Activity Diagram- Play Next

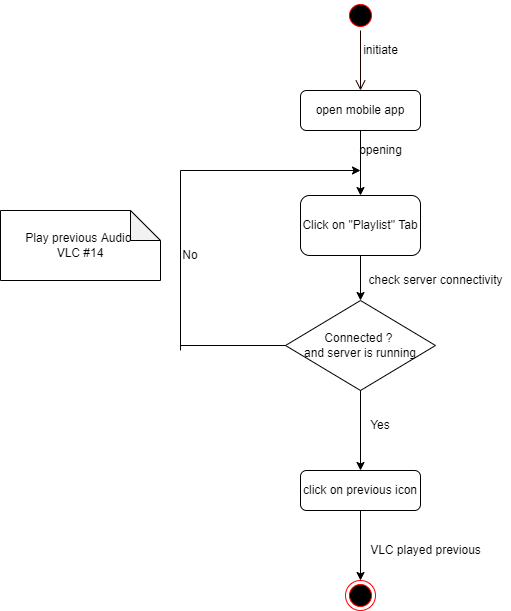


Figure 15:Activity Diagram- Play Previous

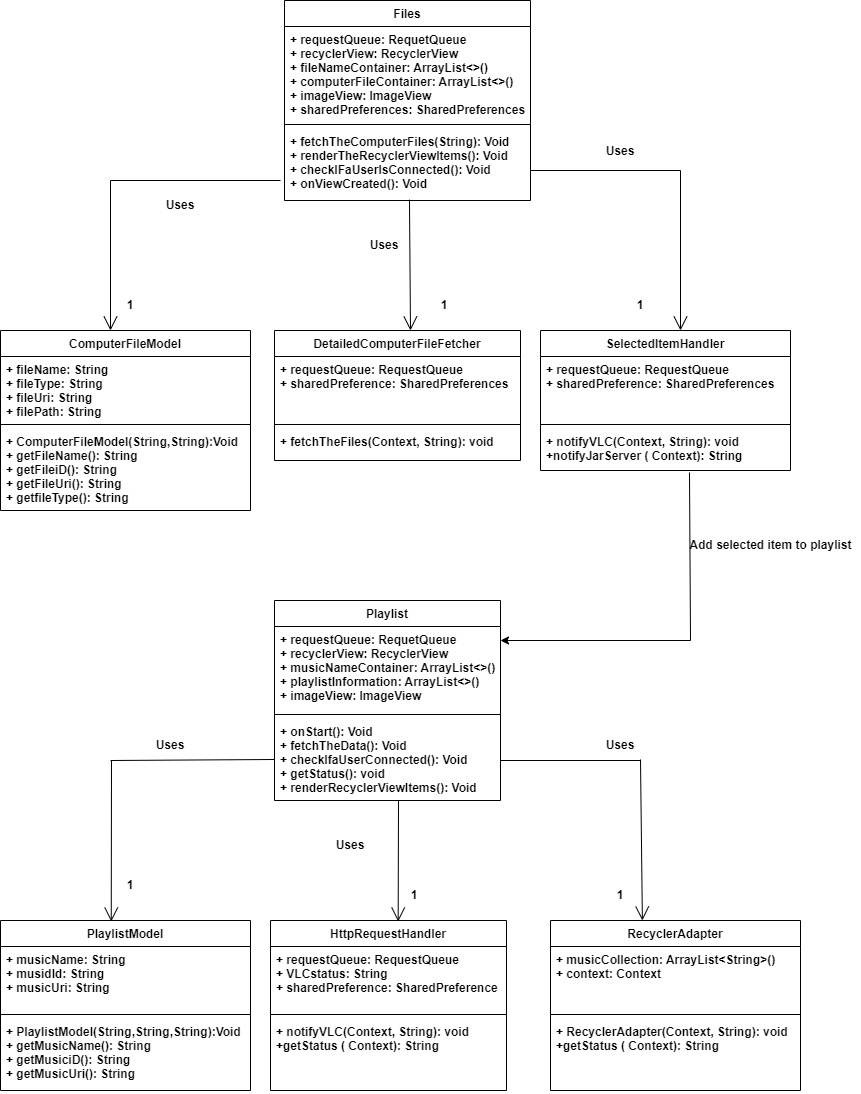
# **3.3.4. Class Diagram**

**Class** is a description of a set of objects that share the same attributes, operations, relationships, and semantics. Graphically, a class diagram is drawn as a rectangle with three compartments holding the class name, attributes, and operation. Class Names: Every class must have a name that distinguishes it from other classes. A name is a textual string. That name alone is known as a simple name; a qualified name is the class name prefixed by the name of the package in which that class lives.

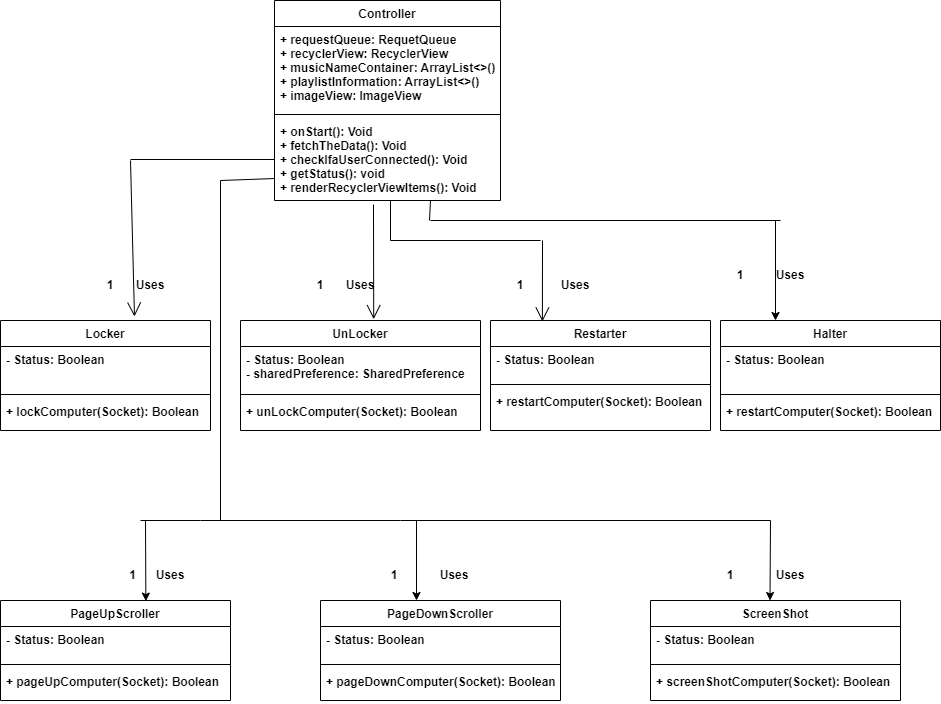
**Attributes**: An attribute is a named property of a class that describes a range of values that instances of the property may hold. A class may have any number of attributes or no attributes at all. An attribute represents some property of the thing you are modeling that is shared by all objects of that class. Operations:

**An operation**: is the implementation of a service that can be requested from any object of the class to affect behavior. In other words, an operation is an abstraction of something you can do to an object that is shared by all objects of that class. A class may have any number of operations or no operations at all

# 



**Figure 16: Class diagram** – Control VLC



**Figure 17:** Class Diagram – Control the Computer

# **3.3.5 Sequence Diagram**

The sequence diagram in this project is used primarily to show the interactions between objects in the sequential order that those interactions occur. During the requirements phase of a project, analysts can take use cases to the next level by providing a more formal level of refinement. When that occurs, use cases are often refined into one or more sequence diagrams.

The main purpose of a sequence diagram is to define event sequences that result in some desired outcome. The focus is less on messages themselves and more on the order in which messages occur; nevertheless, most sequence diagrams will communicate what messages are sent between a system's objects as well as the order in which they occur.

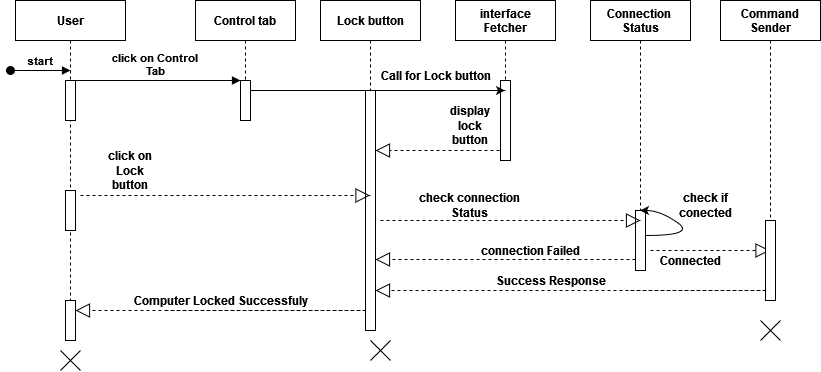


Figure 18:Sequence Diagram – Lock the Computer

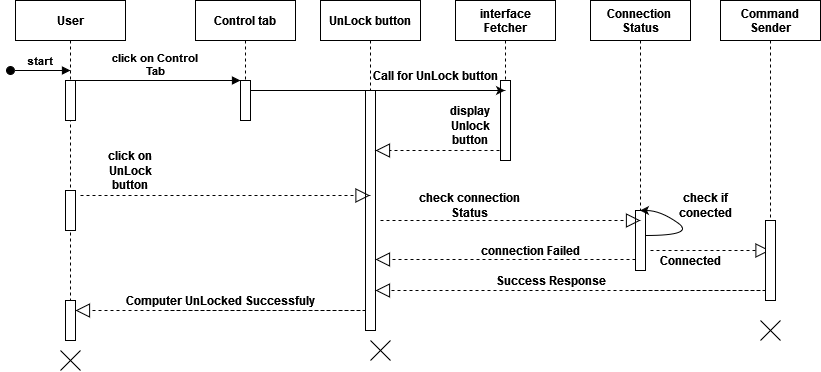


Figure 19:Sequence Diagram – Unlock the Computer

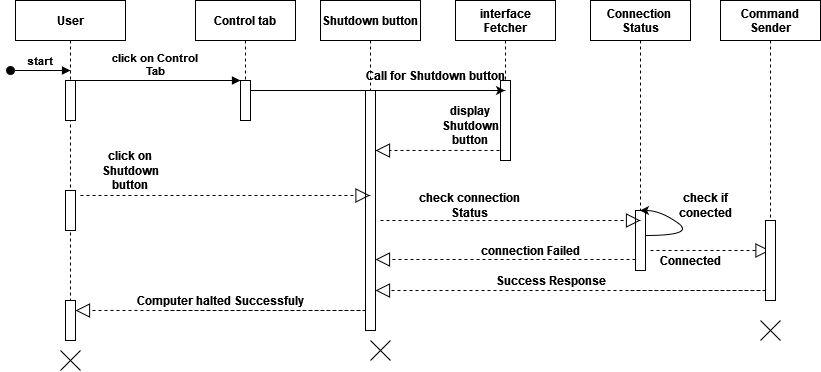


Figure 20:Sequence Diagram – Shut down the computer

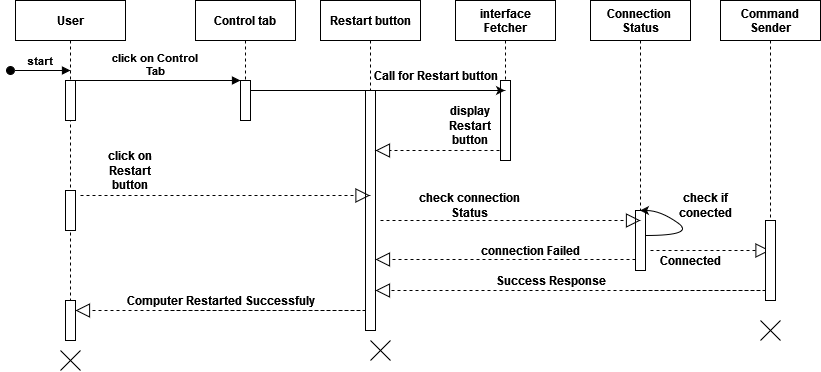


Figure 21:Sequence Diagram – Restart the Computer

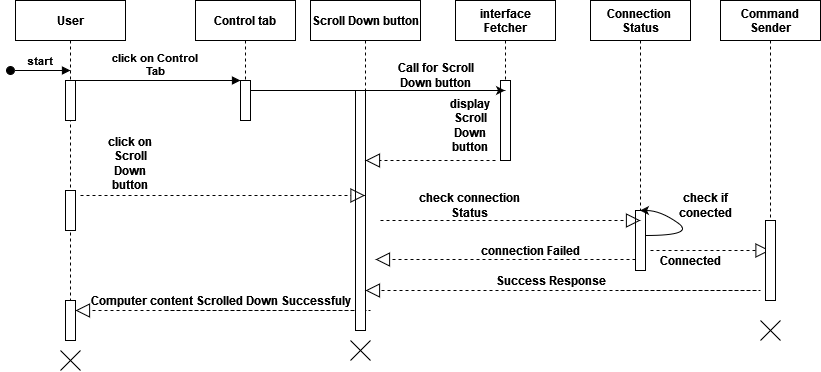


Figure 22:Sequence Diagram – Scroll Down the Pdf, Ppt, Word

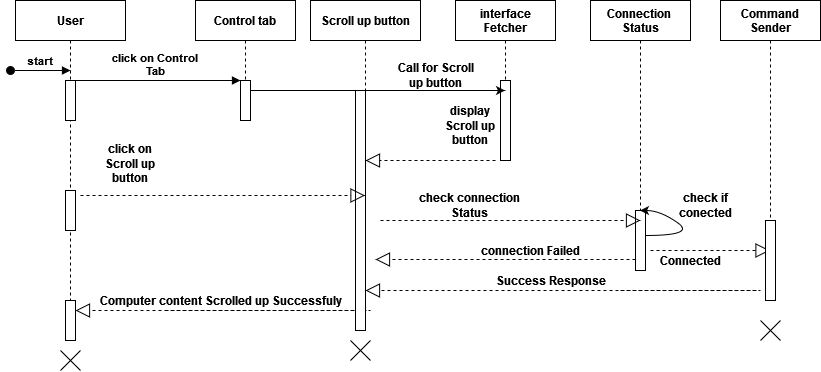


Figure 23: Sequence Diagram – Scroll Up Ppt, Pdf, Word

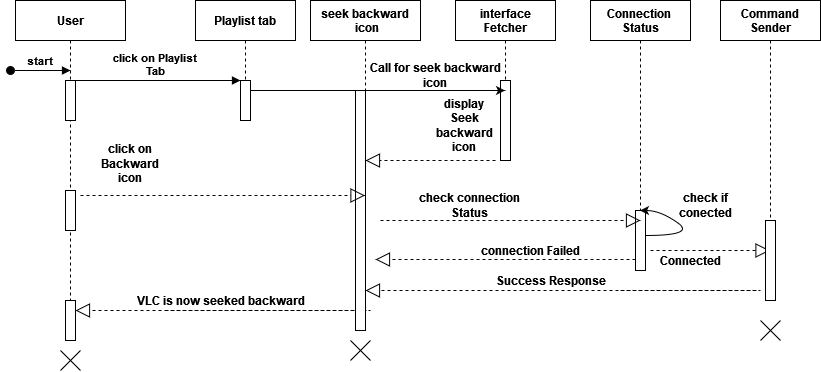


Figure 24:Sequence Diagram – Seek Backward VLC

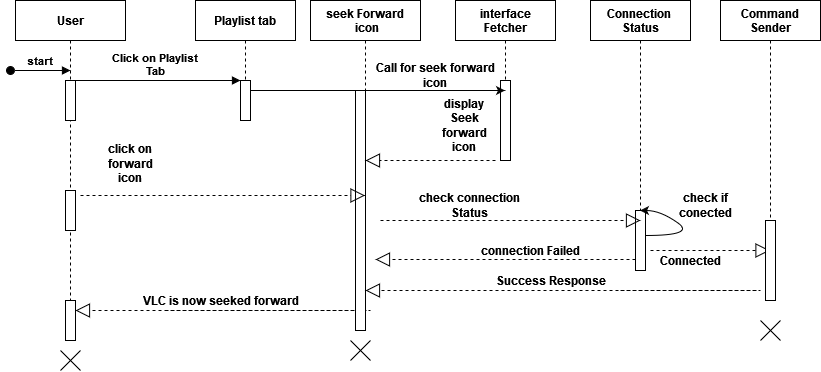


Figure 25: Sequence Diagram – Seek Forward VLC

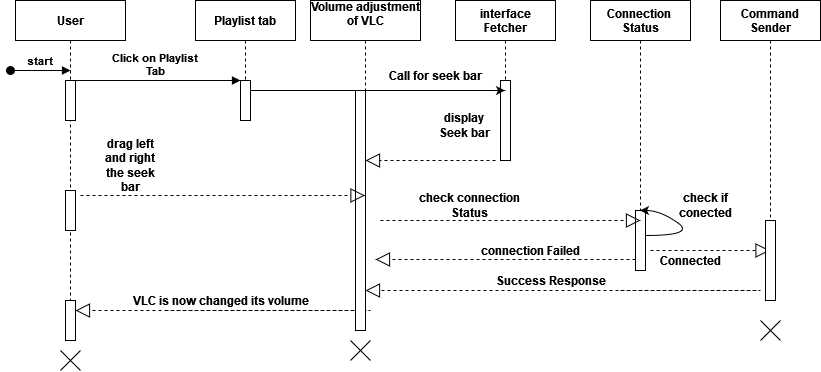


Figure 26: Sequence Diagram – Volume adjustment of VLC

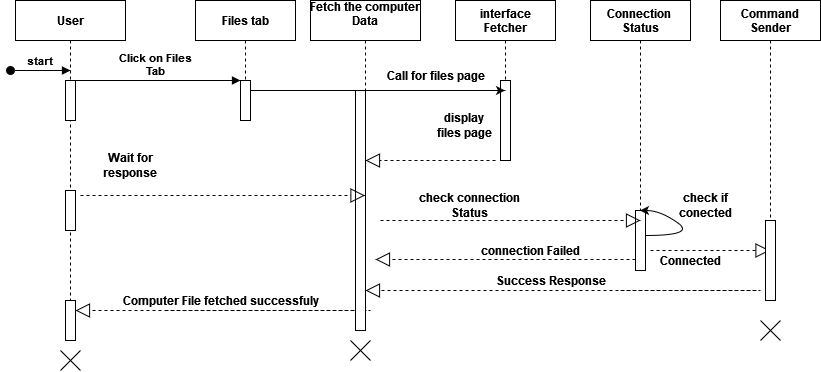


Figure 27: Sequence Diagram – Fetch the computer Data

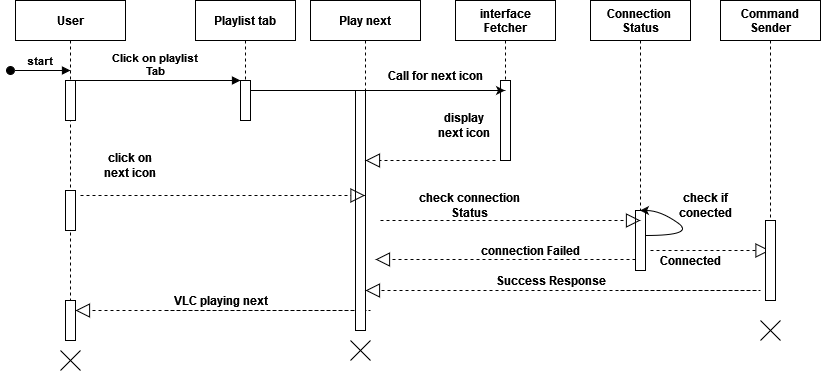


Figure 28: Sequence Diagram – Play next

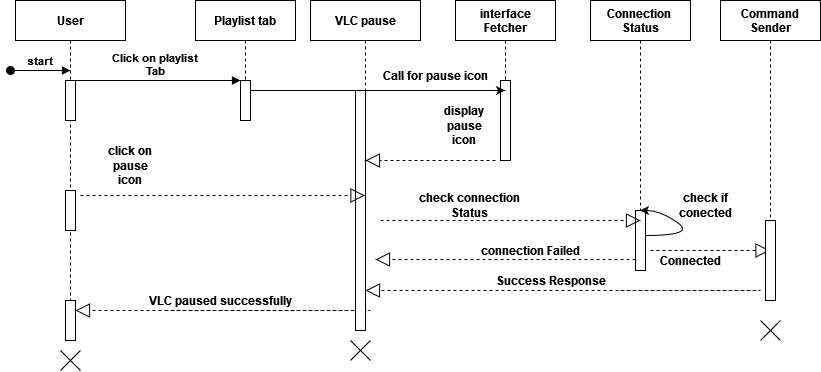


Figure 29: Sequence Diagram – Vlc Pause

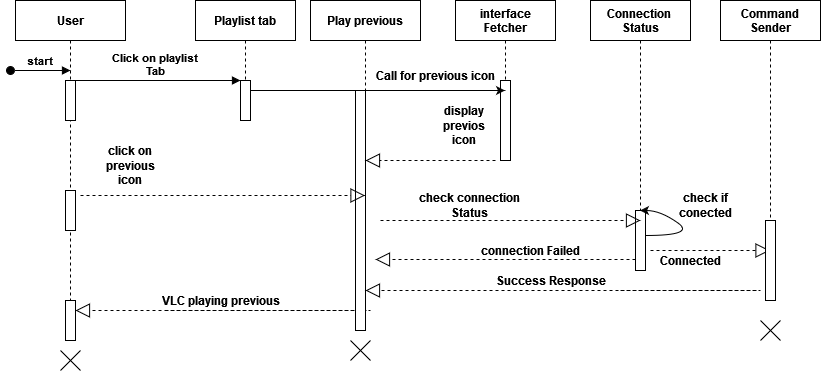


Figure 30: Sequence Diagram – Play previous

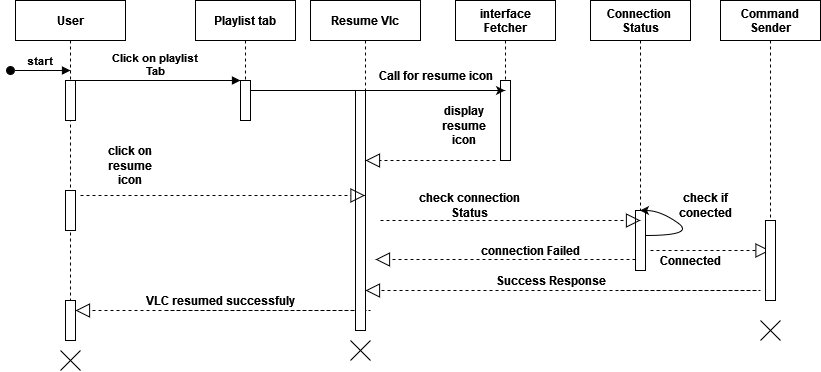


Figure 31: Sequence Diagram – Resume VLC

# **3.3.6 User Interface Design**

**User interface design** or user interface engineering is the design of computers, appliances, machines, mobile applications, software applications, and websites with the focus on the user’s experience and interaction. The goal of use interface design is to make the user’s interaction as simple as and efficient as possible, in terms of accomplishing use goals what is often called user=centered design. User interface focuses on anticipating what a user might need to do and ensuring that interface has elements that are easy to access, understand and use to facilitates those actions. UI brings together concepts from interaction design, visual design and information architecture.

**Playlist**: This tab is the page that contains all playlist item found on VLC. It will be fetched when you enter for the first time to your main Activity or simply when you open your app.



Figure 32: User interface - playlist

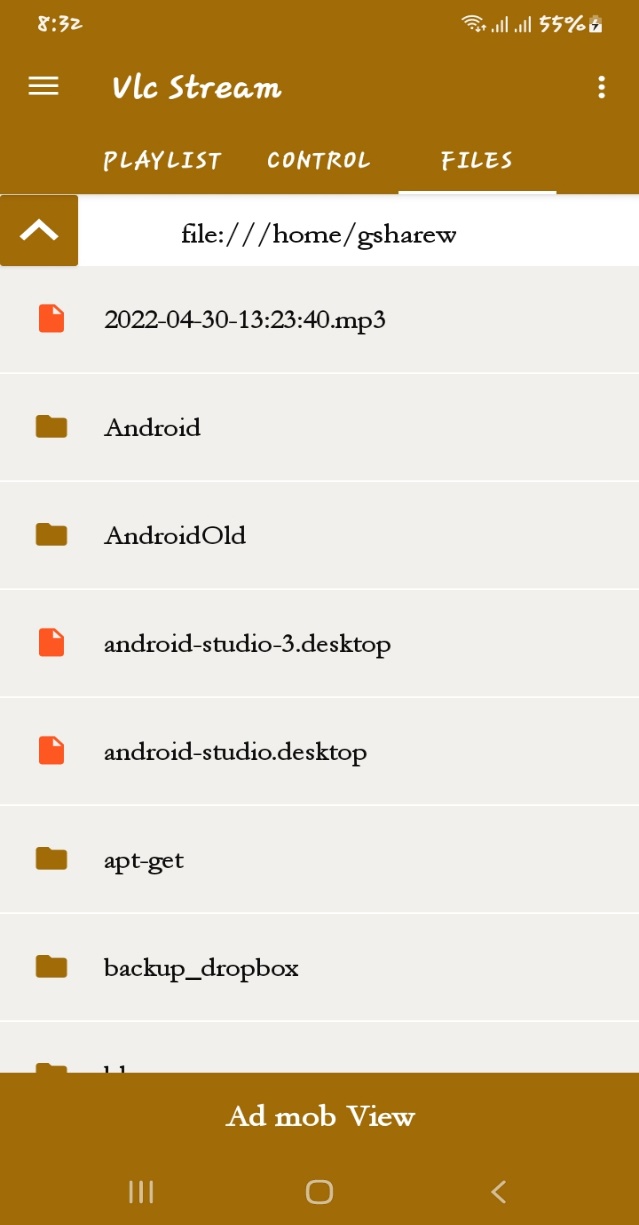
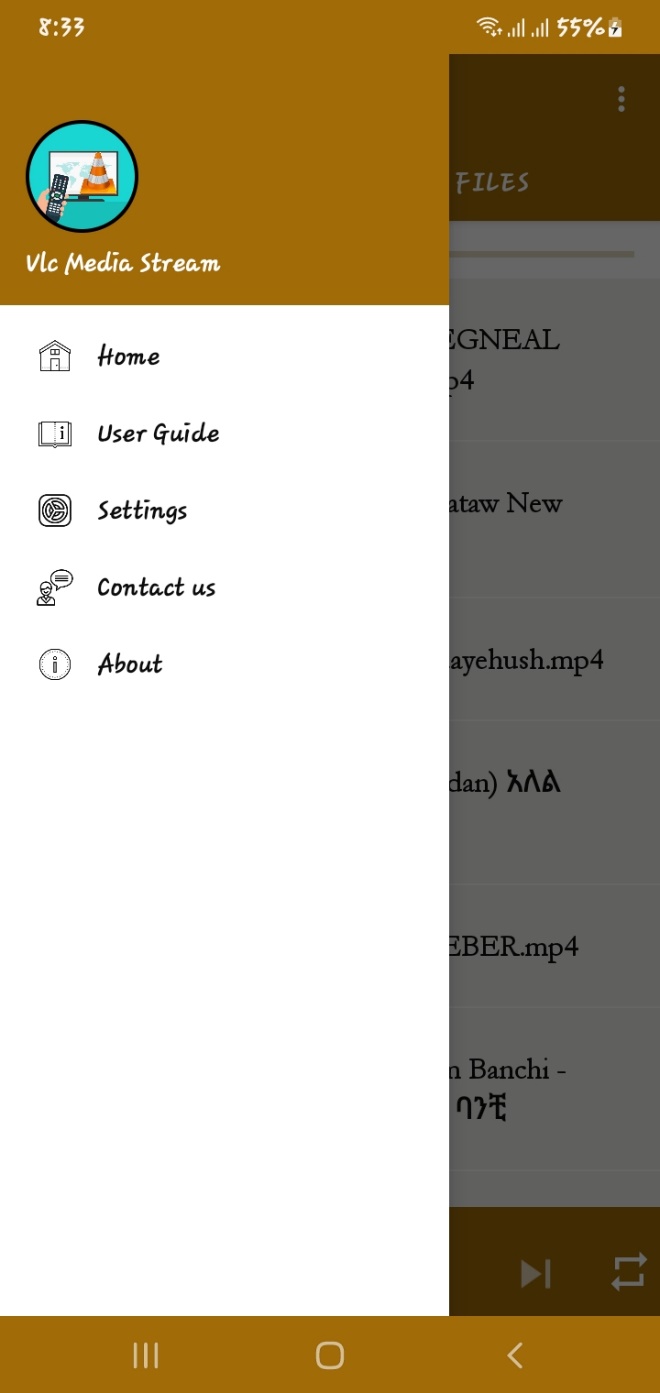
**Files:** this tab is the page that contains all the computer hard drive.

Figure 33: User interface - files

**Drawer Page**: this is the drawer page that contains all the user interface all in one in a vertical alignment. Helps you to navigate through pages.

Figure 34: User interface – Drawer Page

****

# **3.4. Business rules**

The business rule to this system includes some parameters such that

* Computer Server
* VLC Media Player
* Pdf Reader
* Word Reader
* PowerPoint Reader and so on.

# **3.5 state chart diagram**

State chart diagram is used to describe the states of different objects in its life cycle. Emphasis is placed on the state changes upon some internal or external events. These states of objects are important to analyze and implement them accurately. State chart diagrams are very important for describing the states.

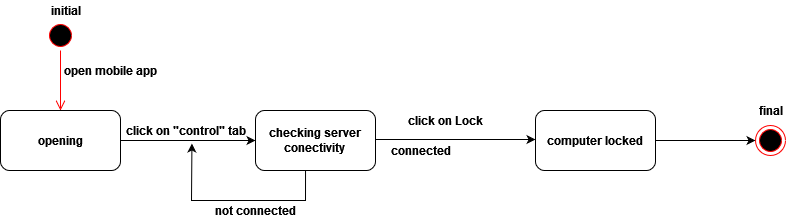


Figure 35: State Diagram – Lock the computer

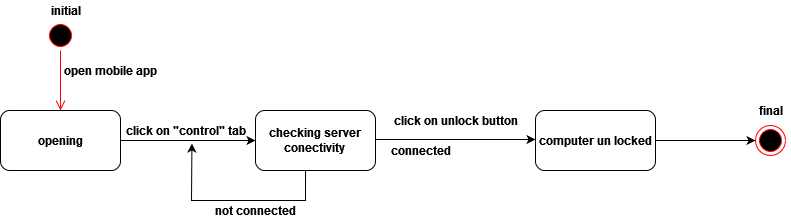


Figure 36: State Diagram – Un Lock the computer

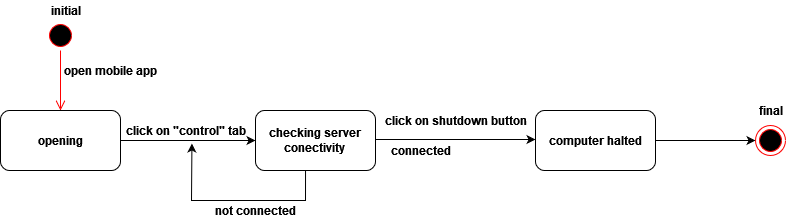


Figure 37: State Diagram – shutdown the computer

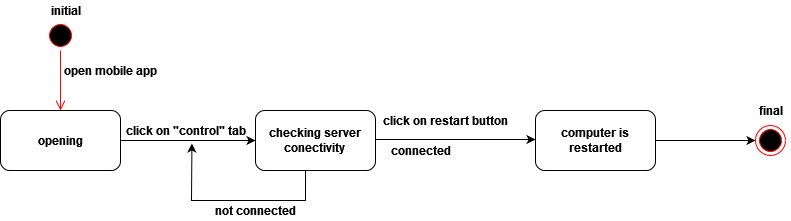


Figure 38: State Diagram – Restart the computer

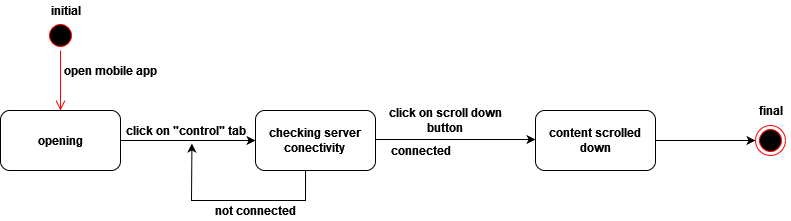


Figure 39: State Diagram – Scroll down the ppt, pdf, word on computer

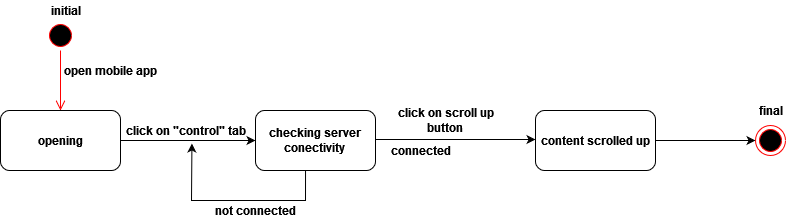


Figure 40: State Diagram – Scroll up the Ppt, Pdf, Word on computer

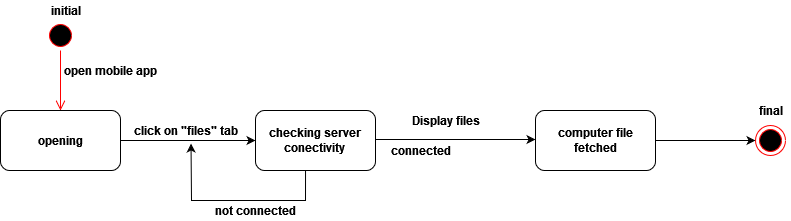


Figure 41: State Diagram – fetch the computer files

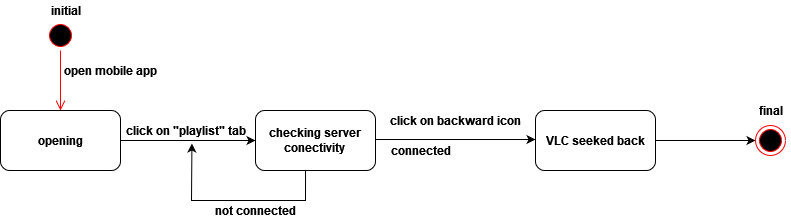


Figure 42: State Diagram – Vlc seek backward

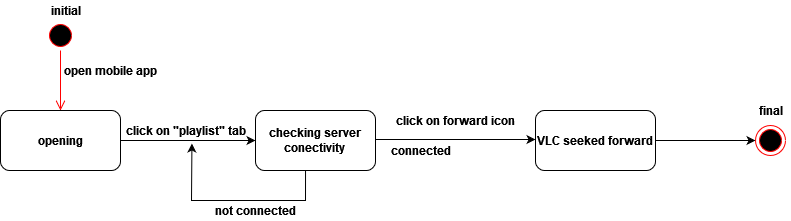


Figure 43: State Diagram – Vlc seek forward

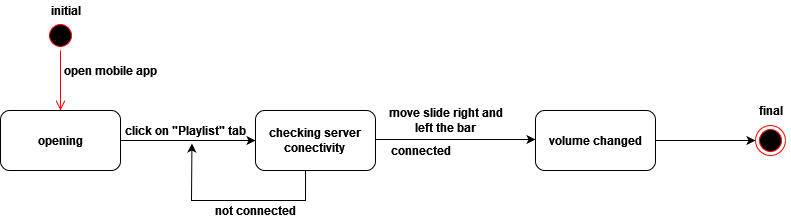


Figure 44: State Diagram – Vlc volume adjustment

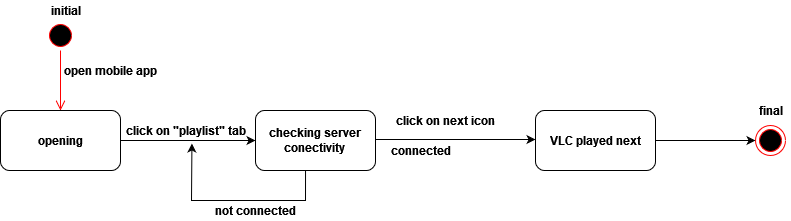


Figure 45: State Diagram – VLC play next

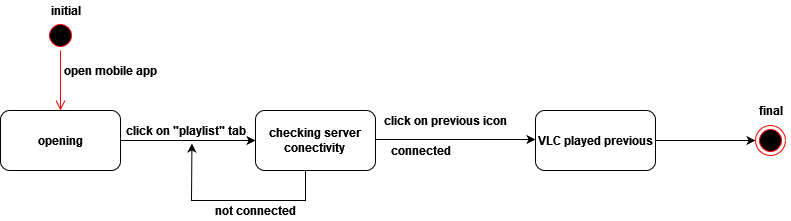


Figure 46: State Diagram – VLC play previous

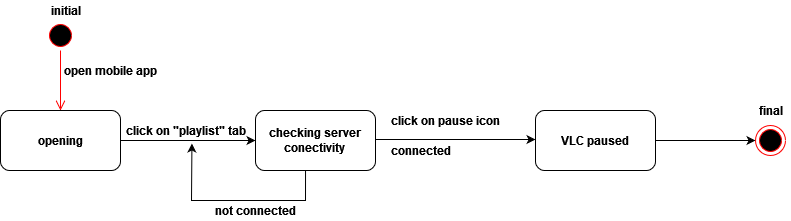


Figure 47: State Diagram – VLC pause

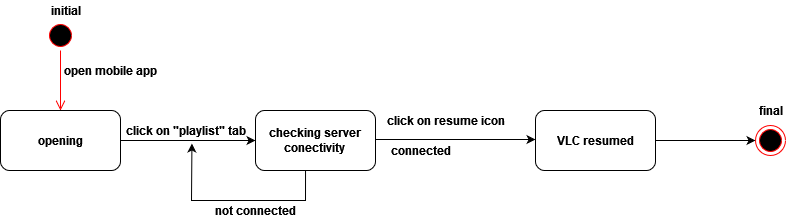


Figure 48: State Diagram – VLC resumed

# **Chapter 4**

## **4.1 System Design**

**System Design** is the process of designing the architecture, components, and interfaces for a system so that it meets the end-user requirements. So, in this project we are going to include some of the system designs such as, Architectural design, component diagram, subsystem decomposition and deployment diagram.

## **4.2 Architectural Design**

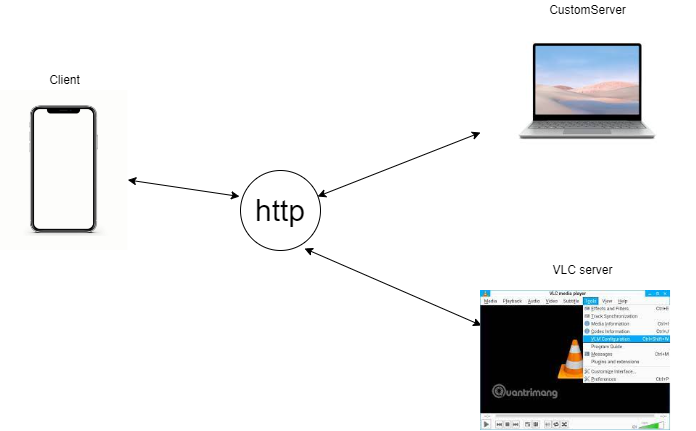
****It is the architecture that determines the type of interactions that the components are going to have. The architecture that this work uses is client/server-based architecture. In this type of architecture, the server is responsible to receive a request from the client and respond to the request, whereas the client is responsible to interact with that of the users of the system. The server parts of this work are of two types. The first type is a web server, which is responsible to receive client’s request through http protocol and responds accordingly. The second type of server is that a VLC server that communicates over http protocol.

Figure 49: Architectural Design

## **4.3 Deployment Diagram**

**A Deployment Diagram** depicts how the deploy-able units of the system applications, components, and data stores are assigned to various nodes, as well as how the nodes communicate with each other and with devices. It’s useful both as a map of your system and as a means for studying the load across your System.

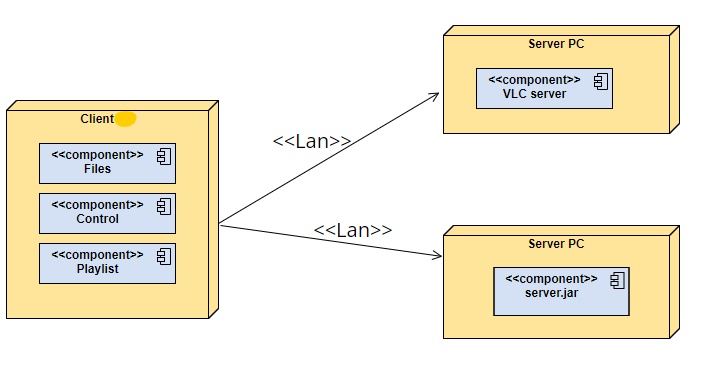
it shows the execution of architecture of a system, including nodes such as hardware or software execution environment, and the middle-ware connecting them. Deployment diagram are typically used to visualize the physical hardware and software’s of the system.

Figure 50: Deployment Diagram

## **4.4 Sub System Decomposition**

In order to reduce the complexity of the system and gaining insight into the identity of the constituent components, our system is decomposed in to sub-systems which are a collection of classes, associations, operations, events and constraints that are closely interrelated with each other.

Streaming control system has the following sub-systems.

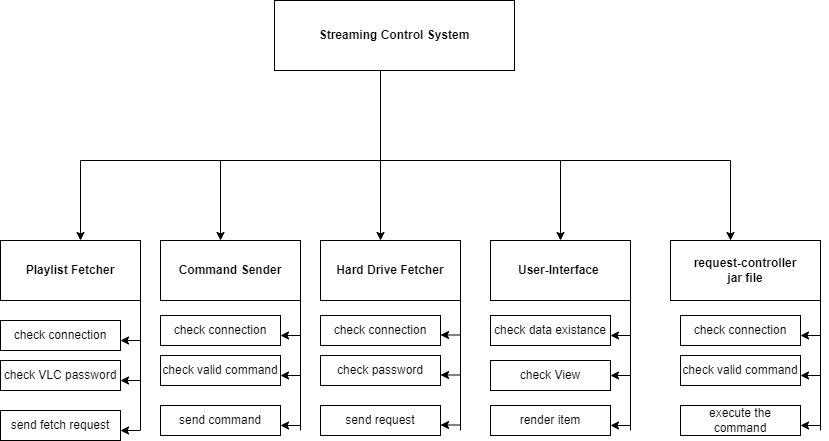


Figure 51: Sub-System-Decomposition

## **4.5 Component Diagram**

A component diagram breaks down the actual system under development into various high levels of functionality. Each component is responsible for one clear aim within the entire system and only interacts with other essential elements on a need-to-know basis.

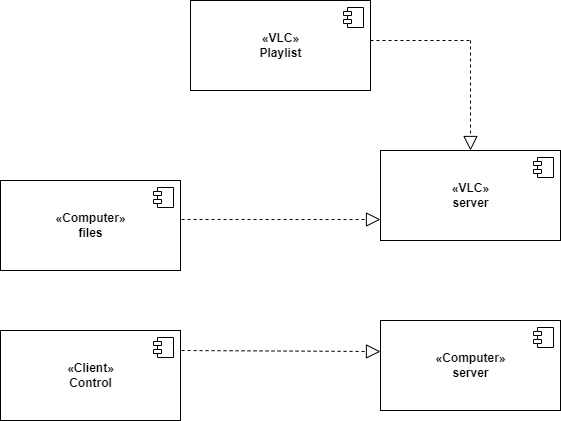


Figure 52: Component Diagram

**4.6 Conclusion**