

# Steer-Mee



**HAIDER ALI TARIQ**  
**01-134132-050**  
**NABEEL SHAHZAD**  
**01-134132-123**

## **Bachelor of Science in Computer Science**

**Supervisor: Mr. Sayed Khushal  
Shah**

Department of Computer Science  
Bahria University, Islamabad

June 2017

# Certificate

We accept the work contained in the report titled “Steer-Mee”, written by Mr. Haider Ali Tariq and Mr. Nabeel Shahzad as a confirmation to the required standard for the partial fulfillment of the degree of Bachelor of Science in Computer Science.

Approved by . . . :

Supervisor: Mr. Sayed khushal Shah (Assistant Professor)

---

Internal Examiner: Dr. Muzammil (Associate Professor)

---

External Examiner: Miss Hira Jaffar (Assistant Professor)

---

Project Coordinator: Dr. Arif Ur Rahman (Assistant Professor)

---

Head of the Department: Dr. Faisal Bashir (Associate Professor)

---

# Abstract

Steer-Mee is basically a new revolution in the field on android phones which allows the 2 users to share their screens with each other such that one user can see the screen of the other user and vice versa. The idea of Steer-Mee has come from the desktop application which is named as Team Viewer which allows the people to share their screens with one another to take some help or to do the random meetings. Basically Steer-Mee will work on the principal of the screenshot capturing and then to sending them to the other person in the real time continuously such that the other person cans see them on the phone screen and remain up to date with the continuous operations being performed on the other person's screens.

# Acknowledgments

There are many efforts involved in the project by some individuals. First of all, I would like to acknowledge Mr. Sayed Khushal Shah who is my supervisor and inspiration behind this project. More, I will like to acknowledge our head of department Dr. Faisal Basher who was really a great motivation.

HAIDER ALI TARIQ

NABEEL SHAHZAD  
Islamabad, Pakistan

June 2016

# Table of Content

<b>CHAPTER 1</b> .....	<b>6</b>
<b>INTRODUCTION</b> .....	<b>6</b>
<b>1.1. Objective</b> .....	<b>6</b>
<b>1.2. Problem Description</b> .....	<b>6</b>
<b>1.3. Methodology</b> .....	<b>7</b>
<b>1.4. Project Scope</b> .....	<b>7</b>
<b>1.5. Feasibility Study</b> .....	<b>8</b>
<b>1.6. Solution Application Areas</b> .....	<b>8</b>
<b>CHAPTER 2</b> .....	<b>9</b>
<b>LITERATURE REVIEW</b> .....	<b>9</b>
<b>2.1. Previous Work</b> .....	<b>9</b>
2.1.1. Team viewer .....	<b>9</b>
2.1.2. Cross Loop .....	<b>10</b>
<b>2.2. Proposed Solution</b> .....	<b>10</b>
<b>CHAPTER 3</b> .....	<b>11</b>
<b>REQUIREMENT SPECIFICATIONS</b> .....	<b>11</b>
<b>3.1. Functional Requirements</b> .....	<b>11</b>
<b>3.2. Quality Requirements</b> .....	<b>11</b>
3.2.1. Availability.....	<b>11</b>
3.2.2. Efficiency .....	<b>11</b>
3.2.3. Robustness .....	<b>11</b>
3.2.4. Reliability.....	<b>12</b>
3.2.5. Modifiability (Future updates) .....	<b>12</b>

3.2.6.    Legal and licensing .....	12
<b>3.3. Non-Functional Requirements.....</b>	<b>12</b>
<b>3.4. Use Case.....</b>	<b>12</b>
<b>CHAPTER 4.....</b>	<b>17</b>
<b>DESIGN.....</b>	<b>17</b>
<b>4.1. System Overview.....</b>	<b>17</b>
<b>4.2. System Architecture .....</b>	<b>18</b>
<b>4.3. Design and Architecture.....</b>	<b>19</b>
4.3.1.    Design Models [along with descriptions].....	19
4.3.2.    Screen Sharing.....	19
<b>4.1. High level Design: .....</b>	<b>20</b>
4.1.1.    Chat and File sharing:.....	20
4.1.2.    Class Diagram:.....	21
<b>4.2. Low level Design.....</b>	<b>22</b>
4.2.1.    Sequence Diagram .....	22
4.2.2.    Collaborative Diagram.....	23
4.2.3.    Data Flow Diagram:.....	24
<b>4.3. Graphical User Interface: .....</b>	<b>24</b>
<b>CHAPTER 5.....</b>	<b>27</b>
<b>SYSTEM IMPLEMENTATION.....</b>	<b>27</b>
<b>5.1. Tools and Technique .....</b>	<b>27</b>
5.1.1.    Android Studio .....	27
5.1.2.    Methodology.....	27
<b>5.2. Language.....</b>	<b>28</b>
<b>CHAPTER 6.....</b>	<b>29</b>
<b>SYSTEM TESTING AND EVALUATION.....</b>	<b>29</b>
<b>6.1. Functional Testing.....</b>	<b>29</b>
6.1.1.    Unit Testing .....	29
6.1.2.    Integration Testing.....	29
6.1.3.    Black Box Testing .....	30

6.1.4. White Box Testing.....	30
<b>6.2. Performance Testing.....</b>	<b>30</b>
<b>6.3. Acceptance Testing.....</b>	<b>30</b>
<b>6.4. Test Cases .....</b>	<b>31</b>
6.4.1. Test Case 1: Screen Sharing.....	31
6.4.2. Test Case 2: Chat.....	31
6.4.3. Test Case 3: File sharing.....	31
<b>6.5. Tool Use .....</b>	<b>32</b>
<b>CHAPTER 7.....</b>	<b>33</b>
<b>CONCLUSIONS AND FUTURE WORK.....</b>	<b>33</b>
7.1. Conclusion .....	33
7.2. Future Work.....	33
<b>REFERENCES.....</b>	<b>34</b>

# List of Figures

<b>Figure 1.</b> Software Development life cycle.....	7
<b>Figure 2.</b> High end design of the application .....	13
<b>Figure 3.</b> Chat.....	13
<b>Figure 4.</b> File share.....	14
<b>Figure 5.</b> Screen Share .....	15
<b>Figure 6.</b> System Failure.....	16
<b>Figure 7.</b> MVC .....	18
<b>Figure 8.</b> Share screen activity.....	20
<b>Figure 9.</b> Chat Activity Diagram .....	20
<b>Figure 10.</b> Figure 4.4: file share activity.....	21
<b>Figure 11.</b> Class diagram .....	21
<b>Figure 12.</b> Chat sequence .....	22
<b>Figure 13.</b> File share sequence .....	22
<b>Figure 14.</b> Screen share sequence .....	23
<b>Figure 15.</b> Collaboration diagram .....	23
<b>Figure 16.</b> data flow diagram.....	24
<b>Figure 17.</b> Main Menu .....	25
<b>Figure 18.</b> Screen Sharing .....	25
<b>Figure 19.</b> Chat.....	26
<b>Figure 20.</b> File Share .....	26



# List of Tables

<b>3.3 USE CASE TABLE1 .....</b>	<b>15</b>
<b>3.5 USE CASE TABLE2 .....</b>	<b>16</b>
<b>3.7 USE CASE TABLE3 .....</b>	<b>17</b>
<b>6.1. TEST CASE 1 .....</b>	<b>31</b>
<b>6.2. TEST CASE 2 .....</b>	<b>31</b>
<b>6.3. TEST CASE 3 .....</b>	<b>31</b>
<b>7. TOOL USE .....</b>	<b>32</b>

# Chapter 1

## Introduction

The purpose of this software design document is to provide a low-level description of the Steer-Mee, providing insight into the structure and design of each component. In short, this document is meant to equip the reader with a solid understanding of the inner workings of the Steer-Mee system.

### 1.1. Objective

The main purpose of making this project is that the person from whom one is asking for the solution and sort out the best possible solution by controlling the device of the problem faced person by using this application on both of the devices. This would bring a revolution in the smart phone world and no one has to search on the internet for the problem being faced by him/her and wasting the time reading blogs on the internet. One can give access to any of the friend who is trusted person and can also watch the each and every step being performed by the controller, this would help the problem face next time solve his/her own problem.

### 1.2. Problem Description

Nowadays, smart phones are carried by almost each and every person. Sharing files, Chatting is common among all. Some of the users don't know each and every practice of the mobile phones and problems arise in front of them. He/she can chat and ask the other user to provide screen access to the second user, where the problem solver will solve the user's problem. Secondly files sharing is common against everyone but thing is some times larger files take a lot of time to transfer. This file transfer would be fast on the same network as compared to others. In addition to this one can share the screens from one mobile phone to the other phone and the next person will be able to see the screen of the next person if there is any of the difficulty to the person who wants to share the problem to the other friend there is very difficult to tell verbally.

### 1.3. Methodology

The SLDC (Software Development Life) Cycle used for the composing and programming of Steer-Me is Iterative method. Iterative method is more reliable and easy to use as we can change the code and easily remove the bugs after testing the project again and again so it's more reliable to use the iterative method to easily tackle with the bugs and the malfunctioning of the modules.

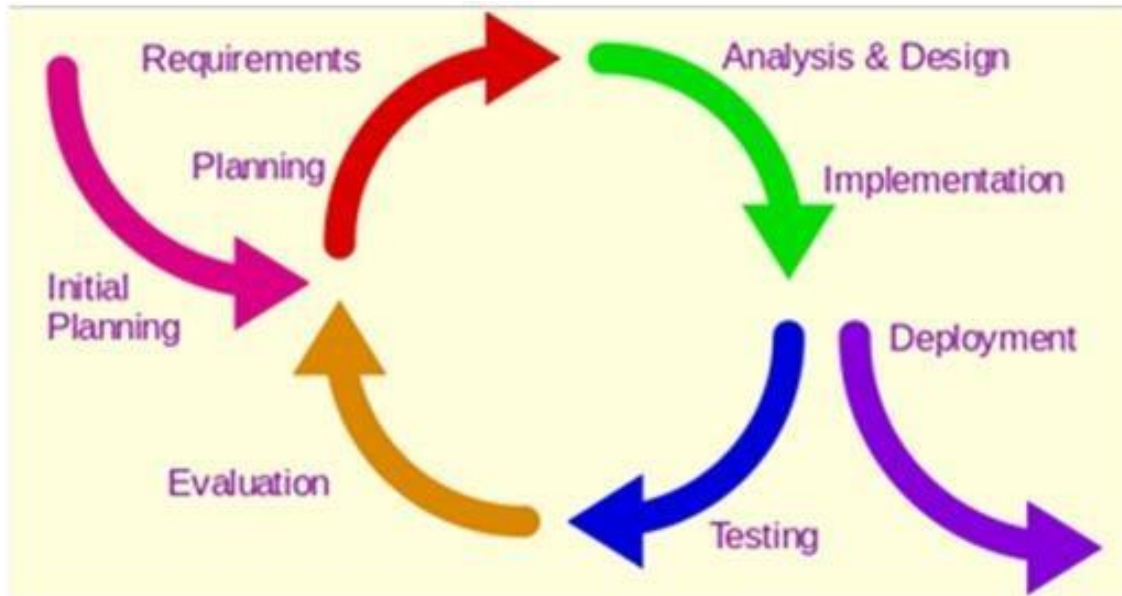


Figure 1. Software Development life cycle

### 1.4. Project Scope

“Steer-Mee” a smart phone application which would be installed in 2 different mobile phones. One person can share the screen of the smart phone with the other person over a same network. If any of them is facing any sort of problem regarding the mobile phone software, he/she can provide the access of the phone to the other person via same network where the person whom access is given can access some of the features of the mobile phone of the person who has provided the access. The first person will generate a password which would be provided by the other person who is going to control the screen of the other. When the second person will provide the similar password, the screen of the first person is ready to be controlled by the second. The second person now can access and control some of the mobile phone features of first person (Similar like in team viewer, one person controls the computer of the other person and solves the problems regarding the software). The person whose mobile phone is being controlled can stop giving the access to the other person at any time he/she wants. The other feature that this application would provide is “CHAT”

that enables the users to trade the passwords with each other.

The person who wants to access the other person's mobile features would first input the password that has been set by the first person and then will be able to gain the access to the other person's mobile phone features. Chat also includes the feature of file sharing which would allow the users to share pictures.

## **1.5. Feasibility Study**

Risks Involved: The risks that involved in Steer-Mee are following.

- No proper online server
- Slow internet
- Same network connection
- Connection lost
- Very big file transfer

Resource Requirement: a laptop/desktop which can run the android studio or having more than 2 GB of Random -access Memory, windows 7 or greater, processor > 2.40GHz. Two android mobile phones will be needed for the testing.

## **1.6. Solution Application Areas**

This would bring a revolution in the smart phone world and no one has to search on the internet for the problem being faced by him/her and wasting the time reading blogs on the internet. One can give access to any of the friend who is trusted person and can also watch the each and every step being performed by the controller, this would help the problem facer next time solve his/her own problem. In fact, this application can be used each and everywhere as it is a problem-solving application.

# Chapter 2

## Literature Review

This part will be explaining how currently this application will be used on the individual and the company levels.

### 2.1. Previous Work

This application is a complete new and innovative idea and this or any similar application isn't been made in the past which makes it more different and unique than others. The main reason that this type of application is not made before is that android phone doesn't give the screen access so easily and it has to be rooted first to delete the Google permissions and then one can do anything with the application. We will try not to root the mobile phones as after rooting the Google linkage with the device gets finished and the phone is free to use for the testing and debugging the applications. Some of the example of previous work is following:

#### 2.1.1. Team viewer

Team Viewer is a proprietary computer software package for remote control, desktop sharing, online meetings, web conferencing and file transfer between computers. Team Viewer may be installed with an installation procedure, although the 'Quick Support' version will run without installation. To connect to another computer to install team viewer administrator-tor access is required, but once installed it can be run by any user. When Team Viewer is started on a computer, it generates a partner ID and password (user-defined passwords are also supported). To establish a connection between a local client and a remote client, Team Viewer generated ID and password of either client are required. The local client requires the remote client's ID and password to gain control over the remote client, whereas the remote client requires the local client's ID and password to gain control over the local client.

### **2.1.2. Cross Loop**

Cross Loop was a remote desktop service from Cross Loop Inc., which allowed users to share their computer screens and collaborate with others over the Internet. The Cross Loop Marketplace was a place where people could get remote IT support 24 hours a day from a global network of experts through secure screen sharing technology. Individuals and businesses could get assistance with computers, mobile devices, peripherals, software and training.

- Cross Loop Pro – subscription-based collection of tools for remote access and customer management.
- Cross Loop Remote Access – subscription-based remote access software.
- Cross Loop Free – A free screen-sharing and collaboration tool.

## **2.2. Proposed Solution**

Anything that is being used in the application is completely new and unique that makes it more good and unique. The main module i.e. the screen sharing is never done before in two of the mobile phones. There are Applications like team viewer and Cross Loop etc. but these are for pc to pc not for phone to phone screen sharing, there are also applications of screen sharing on play store these are fake applications and are not in working state. The other features such as chat and the file sharing is done which is a quite common feature in the application and is most commonly used in many of the applications.

# Chapter 3

## Requirement Specifications

### 3.1. Functional Requirements

Functional requirements are those requirements that are doing the main work in the application and due to which an application is known as. Here are some of the functional requirements of the application which are the main modules and that are doing the work which is the main feature of Steer-Mee.

### 3.2. Quality Requirements

#### 3.2.1. Availability

This is a standalone application and will be available when the user will trigger. In addition to this, it will be available when the server or the local host will be turned on and the two phones are connected to the internet.

#### 3.2.2. Efficiency

System will ensure efficiency by providing fast solution to all modules.

#### 3.2.3. Robustness

System maintains every log on run time so there is no fear of loss of data. System will ensure the robustness until the operating system (Android) crashes. In addition to this if there is slow internet then there may be a chance of losing some of the screen shots that are being sent by the sender to the receiver i.e. the frames.

### **3.2.4. Reliability**

System will ensure the high-level reliability as it is made on the advanced security measures as it is essential to keep it secure because it is a very critical matter of sharing the private data via a screen shot to the other person.

### **3.2.5. Modifiability (Future updates)**

This is the beta version of the software which will be first released and we will take the user reviews what they say and then we can surely upgrade it to the next version. There are many certain things where we can add other features. So with the passage of time we will do that as well.

### **3.2.6. Legal and licensing**

System will be built on Android Studio 2.2.3 platform and will require a regular developer license of the android developer technologies. Make sure that all the sdk's of the android studio are properly installed and the packages are up to date.

## **3.3. Non-Functional Requirements**

The speed of the software is relatively faster than the other similar applications. It sends 30-60 frames depending on the RAM of the mobile phone. The mobile phone sends the frames and then the other phone keeps on checking the updated data from the 1st one, as soon as the 1st phone sends the more frames, the 2nd phone displays that updated frames and now the screen controlling is started as the screen frames are streaming in the video form. This all is happening over the same network and the communication is being done by the router. The phone which is receiving the frames will continuously update the latest possible frame and will show it on the screen.

## **3.4. Use Case**

Use cases are a widely used and highly regarded format for capturing requirements. Before writing functional requirement use cases can help you to understand the requirements in the way user expect. Following table presents you the template to write use case(s) to write each section with example.



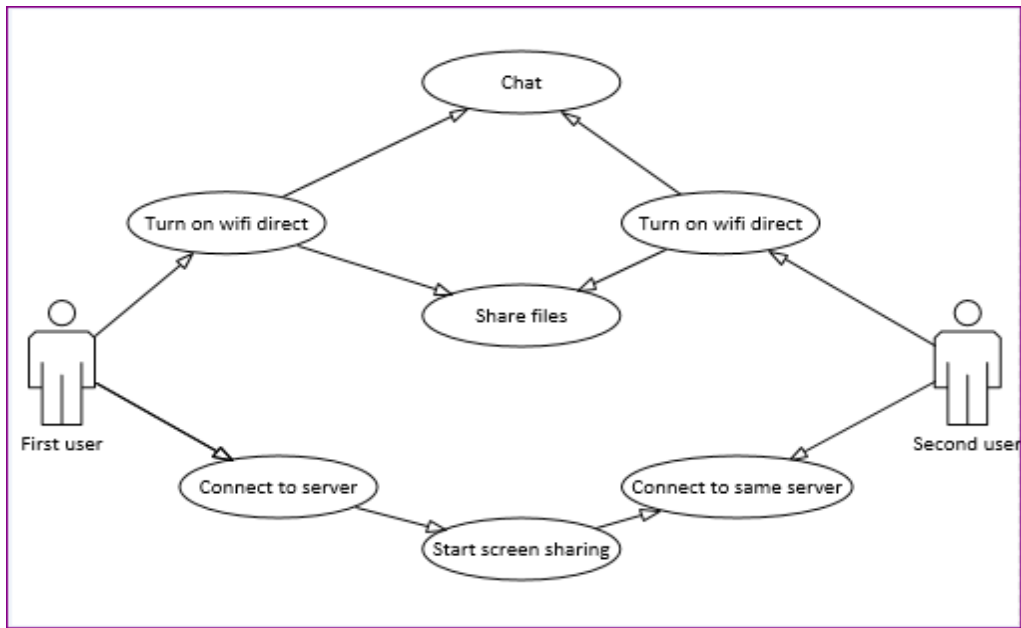


Figure 2. High end design of the application

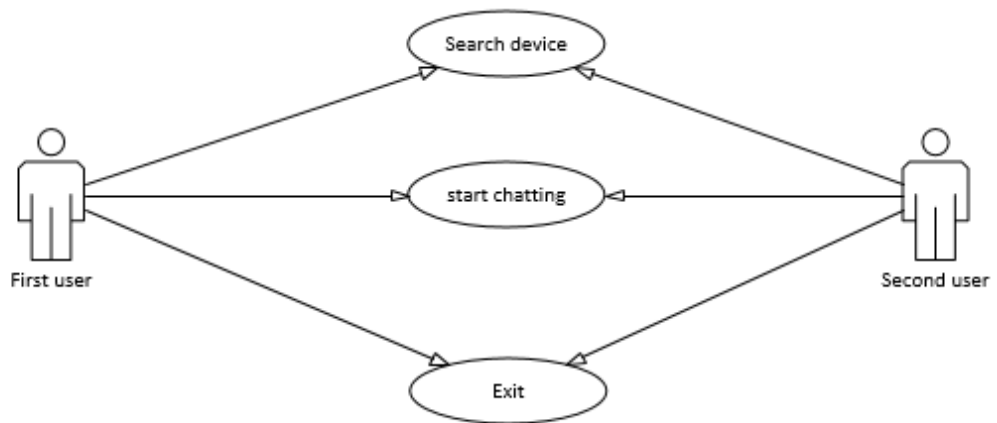


Figure 3. Chat

Figure 3.4.1:Use Case table1

<b>Use Case ID:</b>	UC-01
<b>Use Case Name:</b>	Chat
<b>Actors:</b>	User
<b>Description:</b>	User shall be able to chat with the other person.
<b>Trigger:</b>	User clicks on chat button.
<b>Pre-Condition:</b>	Both phones should be connected on same network.
<b>Post-Condition:</b>	Successful connection will allow to communicate.
<b>Normal Flow:</b>	<ol style="list-style-type: none"> <li>1. Click on the chat button on homepage.</li> <li>2. Turn on the Wi-Fi Direct option on the mobile phone.</li> <li>3. Connect the both mobile phones to one another via Wi-Fi Direct technology so that they establish peer to peer connection.</li> <li>4. Click on the Search button.</li> <li>5. Select user to chat.</li> <li>6. Start the chat.</li> </ol>
<b>Exception:</b>	In Step 5 of the normal flow , if the user enters the peers who is not on the peer. System shall redirect to homepage.
<b>Includes:</b>	UC-01

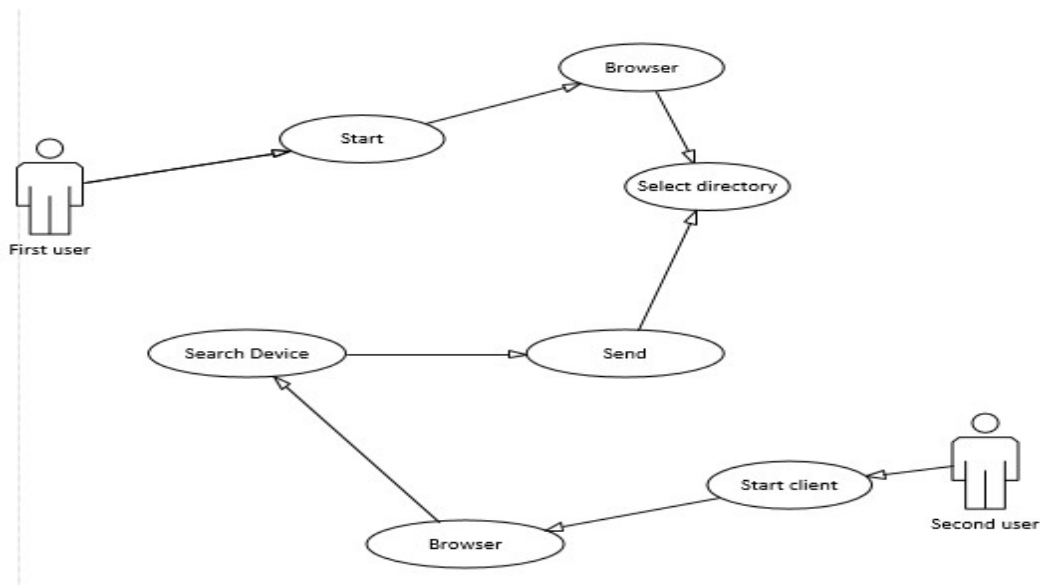
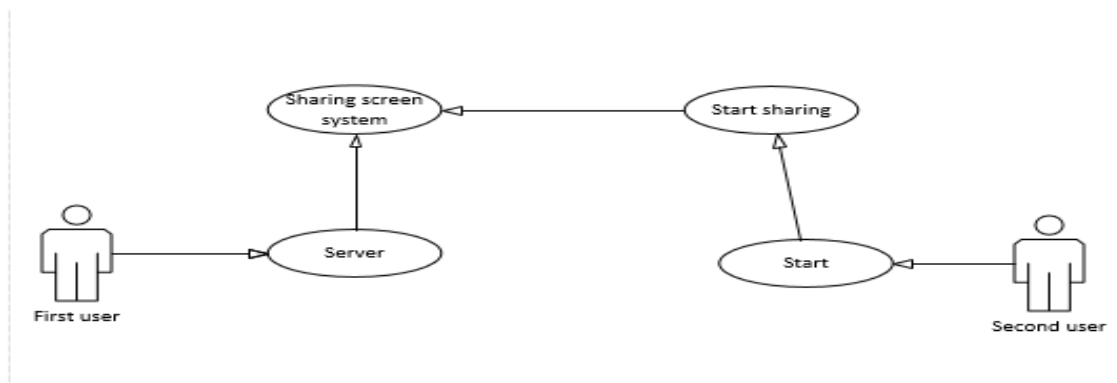


Figure 4. File share

Figure 3.5: Use case table2

<b>Use Case ID:</b>	UC-02
<b>Use Case Name:</b>	File sharing
<b>Actors:</b>	User
<b>Description:</b>	User shall be able to share file with the other person.
<b>Trigger:</b>	User clicks on file sharing button.
<b>Pre-Condition:</b>	Both phones should be connected on same network.
<b>Post-Condition:</b>	Successful connection will allow to communicate.
<b>Normal Flow:</b>	<ol style="list-style-type: none"> <li>1. Click on the file sharing button on homepage.</li> <li>2. Click on the search option.</li> <li>3. Select user to share the files.</li> <li>4. Send the files.</li> </ol>
<b>Exception:</b>	In step 4 of the normal flow, if the user enters the peers who is not on the peer.
<b>Includes:</b>	Uc-02

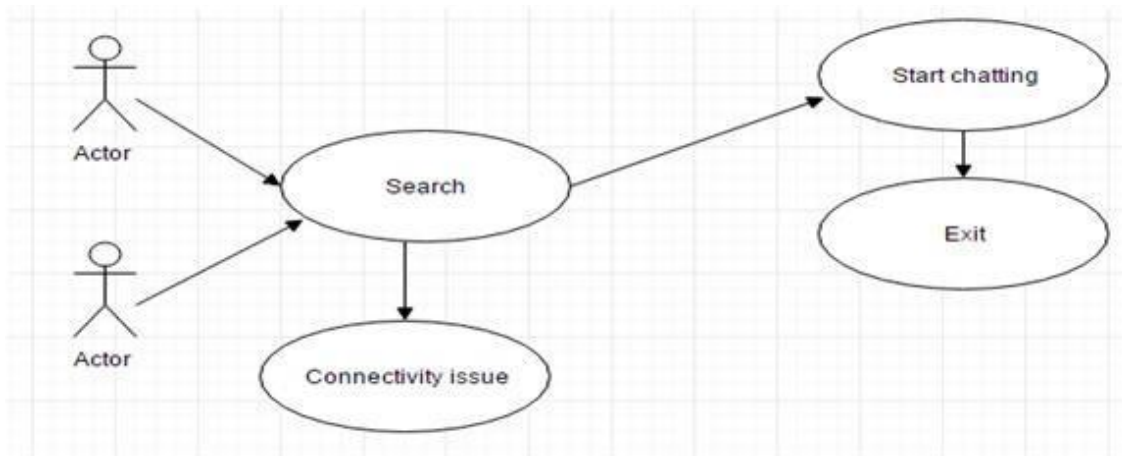
Figure 5. Screen Share



**Figure 3.7: Use case table3**

<b>Use Case ID:</b>	UC-03
<b>Use Case Name:</b>	Share the screen
<b>Actors:</b>	User
<b>Description:</b>	User shall be able to share his/her screen with the other program.
<b>Trigger:</b>	User clicks on Screen sharing.
<b>Pre-Condition:</b>	Both phones should be connected on same network.
<b>Post-Condition:</b>	Successful connection will allow to share the screens of both.
<b>Normal Flow:</b>	<ol style="list-style-type: none"> <li>1. Click on the screen share button on homepage.</li> <li>2. Start Apache server from the machine.</li> <li>3. See the system ip address.</li> </ol>
<b>Exception:</b>	In step 4 of the normal flow, if the user enters the peers who is not on.
<b>Includes:</b>	Uc-03

**Figure 6. System Failure**



# Chapter 4

## Design

### 4.1. System Overview

“Steer-Mee” a smart phone application which would be installed in 2 different mobile phones. One person can share the screen of the smart phone with the other person over a same network. If any of them is facing any sort of problem regarding the mobile phone software, he/she can provide the access of the phone to the other person via same network where the person whom access is given can access some of the features of the mobile phone of the person who has provided the access. The first person will generate a password which would be provided by the other person who is going to control the screen of the other. When the second person will provide the similar password, the screen of the first person is ready to be controlled by the second. The second person now can access and control some of the mobile phone features of first person (Similar like in team viewer, one person controls the computer of the other person and solves the problems regarding the software). The person whose mobile phone is being controlled can stop giving the access to the other person at any time he/she wants. The other feature that this application would provide is “CHAT” that enables the users to trade the passwords with each other. The person who wants to access the other person’s mobile features would first input the password that has been set by the first person and then will be able to gain the access to the other person’s mobile phone features. Chat also includes the feature of file sharing which would allow the users to share pictures.

## 4.2. System Architecture

System Architecture used for the development of our project is MVC (Model View Controller). This architecture style is used because it provides clarity of design, efficient modularity. It is distributable so we can reuse code. It is a flexible approach in which each time an increment can be made where required during the development of software product.

The purpose of using MVC architecture is to divide application logic from the presentation and business logic. The advantage of using MVC:

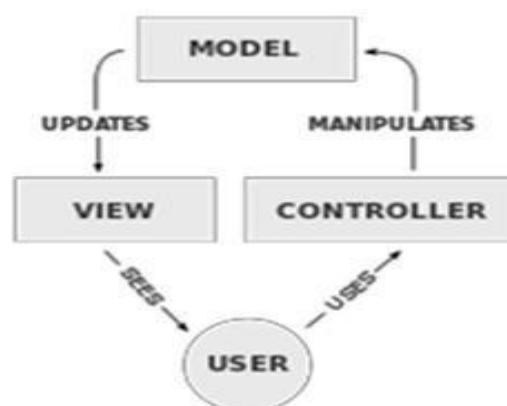
- Easier maintenance, testing, update the application
- Flexibility in planning and implementing object Model.
- Separate layer of project
- Parallel development of objects
- The application is extensible and scalable

**Model** – Business logic. It represents data structures. Provide functions for retrieve, insert and change information in our database.

**View** – Presentation logic. It is the information that is being presented to a user.

**Controller** – Application logic. It joins everything together specially data and generates the view for the user.

Figure 7. MVC



## **4.3. Design and Architecture**

This chapter will discuss the design and architecture of your system.

### **4.3.1. Design Models [along with descriptions]**

The applicable models may include:

- Class Diagram
- Sequence Diagram
- State Transition Diagram
- Data Flow Diagram
- Schematic Diagram(only for hardware projects)

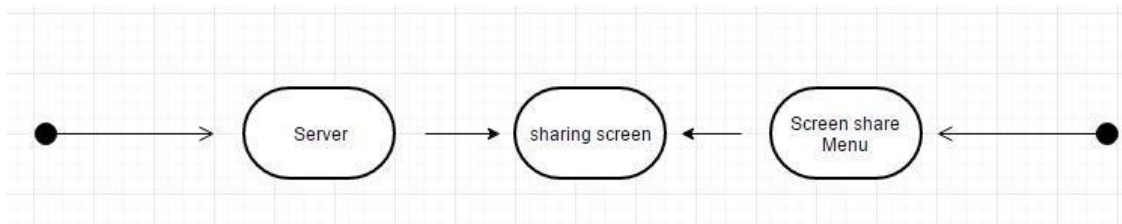
Insert applicable system models here. You should be clear about all the concepts used in your diagrams for example for class diagram you should know about aggregation, composition, inheritance /generalization. Also ensure visibility of all diagrams. Class diagram and associated models shall only be necessary for object oriented approach. In case of procedural, create a DFD. Data flow diagram should be extended to 2-3 levels. It should clearly list all processes, their sources/sinks and data stores.

### **4.3.2. Screen Sharing**

To share the screen to the other person, one will have to be connected on a same network on which the other person whom is to be receiving the screen shots is connected. And the first person will be connected now on the network where the machine is already running a server. The other person will then input the same IP address and the screen of one person will be completely shown and accessed by the other person who provided the IP address of the machine server.

The activity diagram of the screen sharing is given following:

Figure 8. Share screen activity



## 4.1. High level Design:

### 4.1.1. Chat and File sharing:

If a person has to share the file with the other person, he / she will have to sign up by a simple process and select the phone number of the friend and send that particular file or the picture etc.; to other. Similarly one person can chat with the other person and send and receive the text messages. This chat feature can be used in a more better way in order to share the password that one person has generated and wants to share it with the other person whom he / he wants to give access to.

Figure 9. Chat Activity Diagram

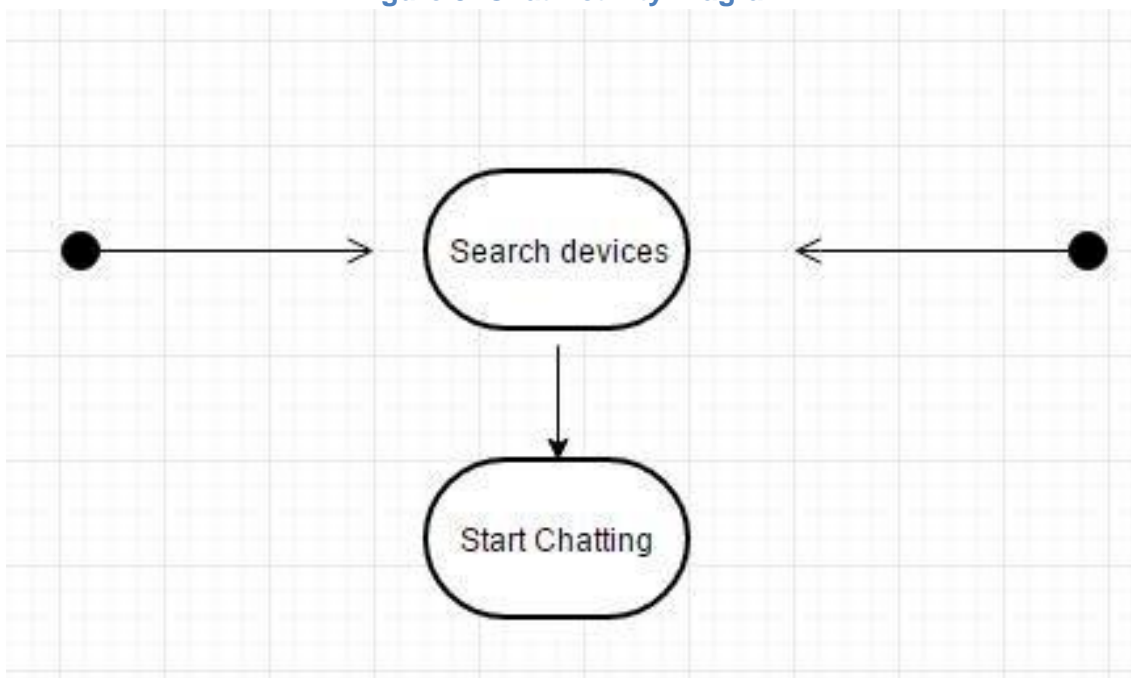
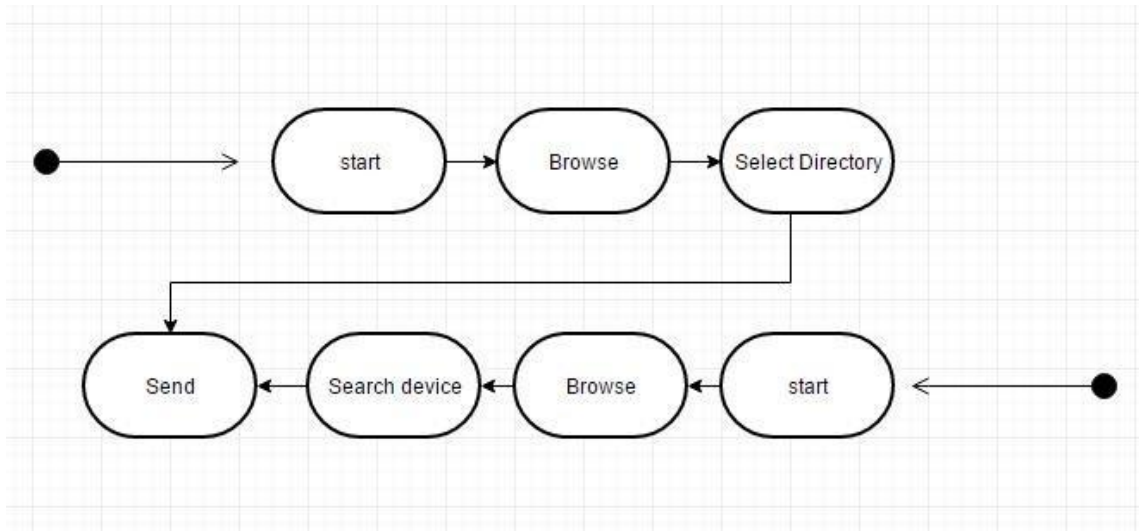


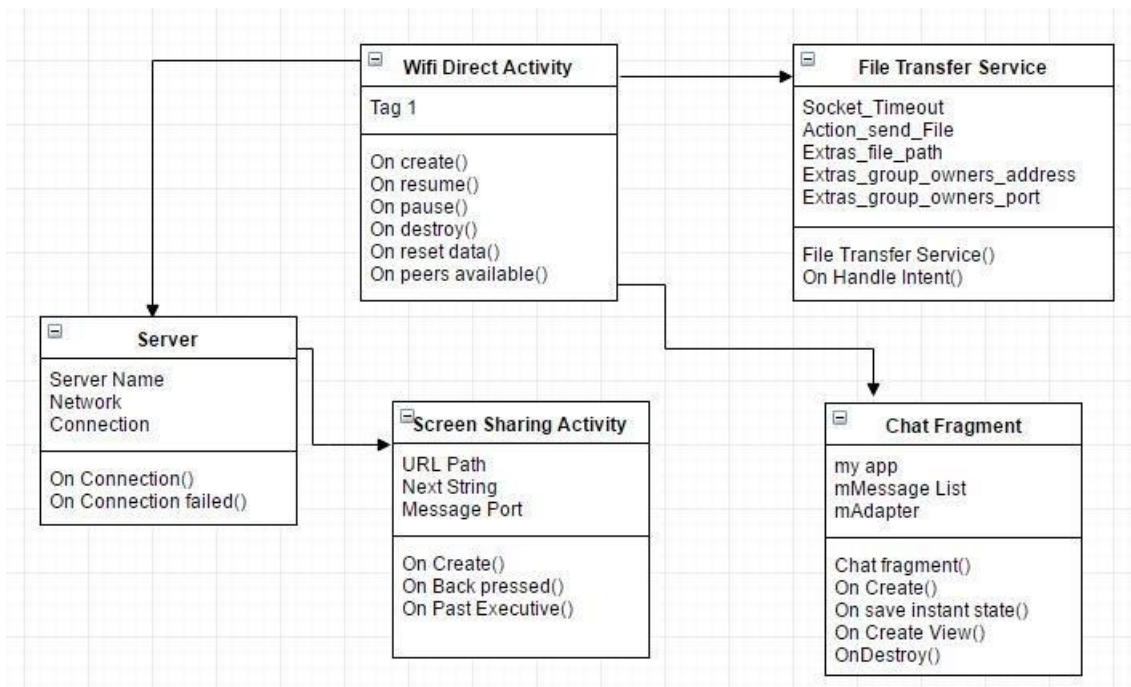


Figure 10. Figure 4.4: file share activity



### 4.1.2. Class Diagram:

Figure 11. Class diagram

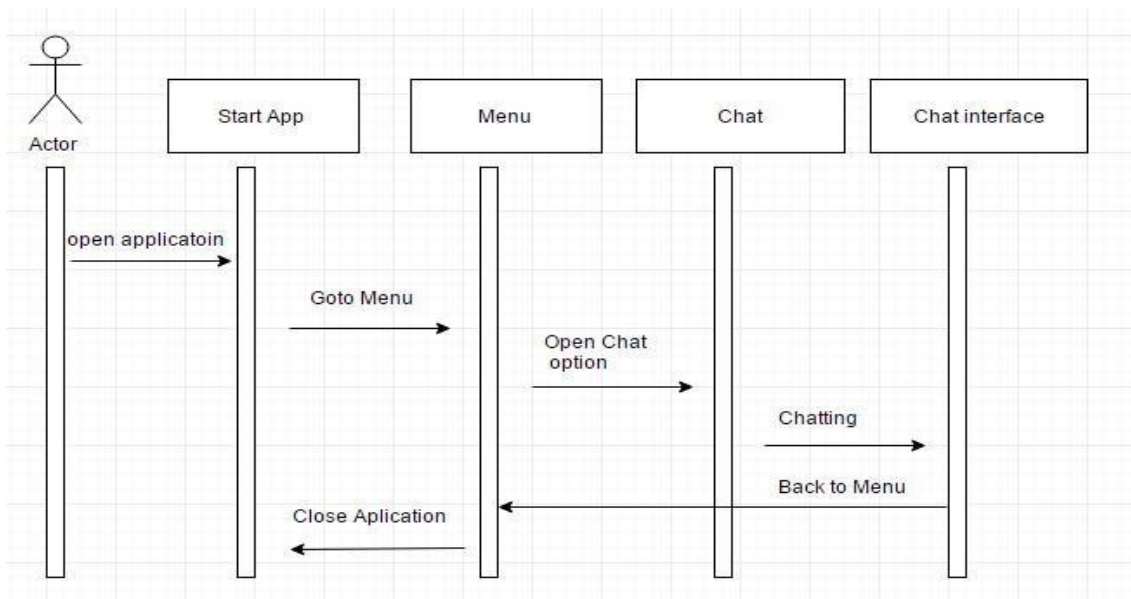


## 4.2. Low level Design

### 4.2.1. Sequence Diagram

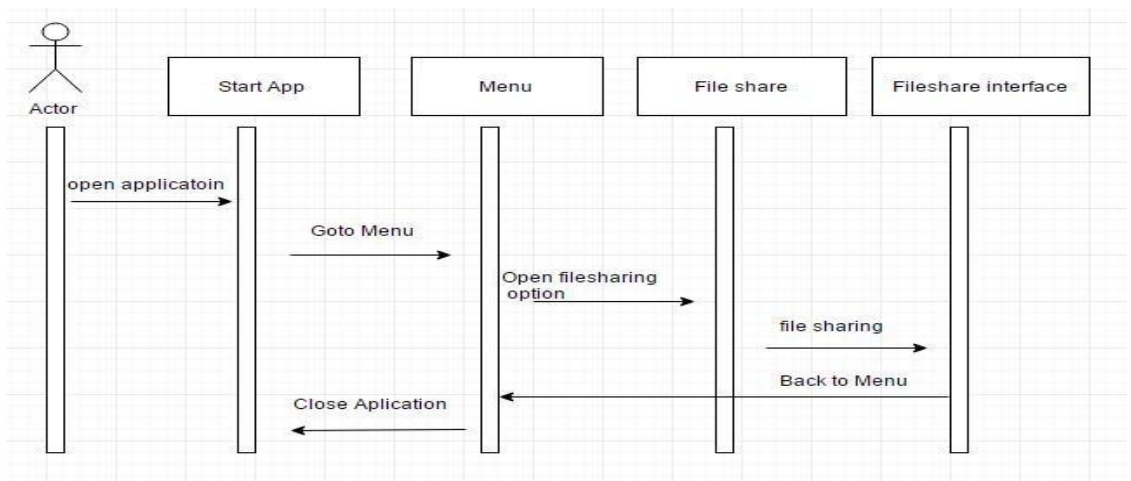
Chat:

Figure 12. Chat sequence



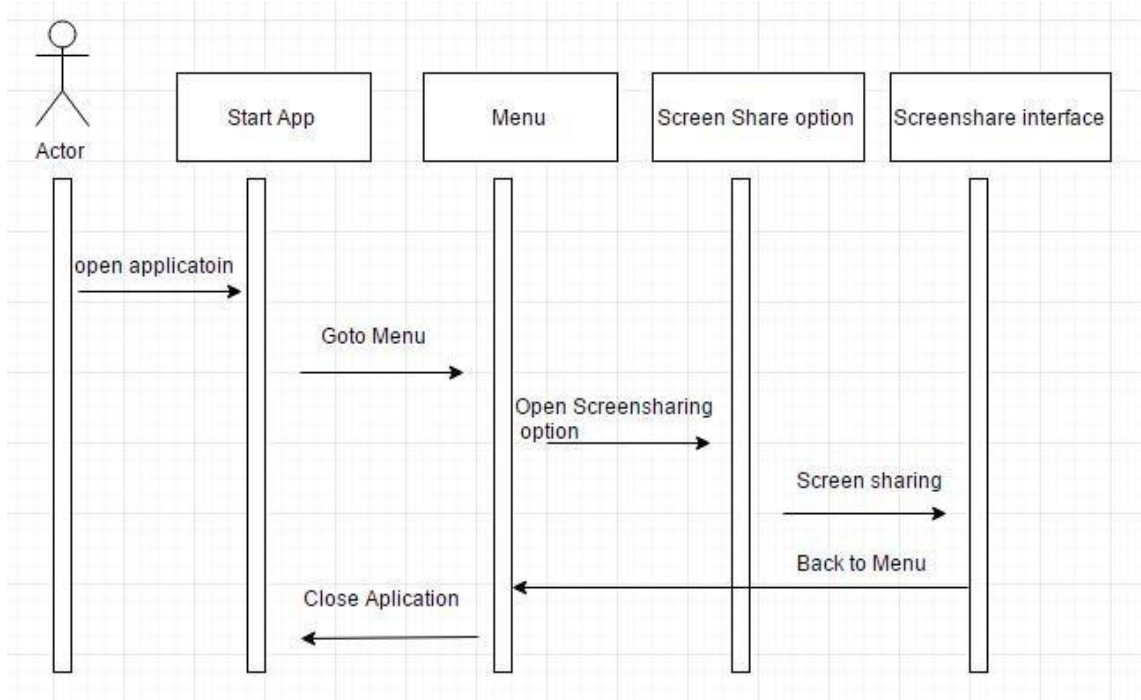
File Sharing:

Figure 13. File share sequence



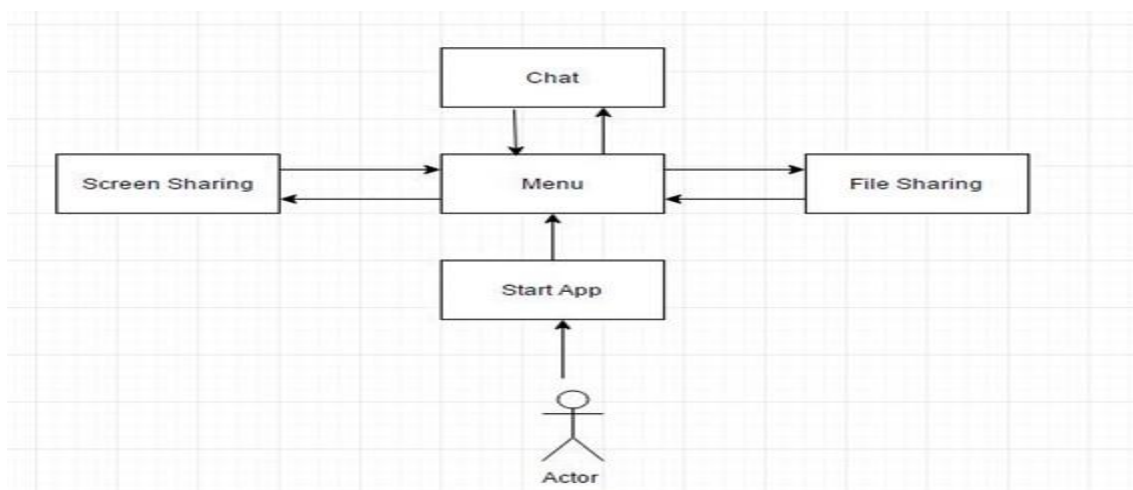
## Screen Sharing:

Figure 14. Screen share sequence



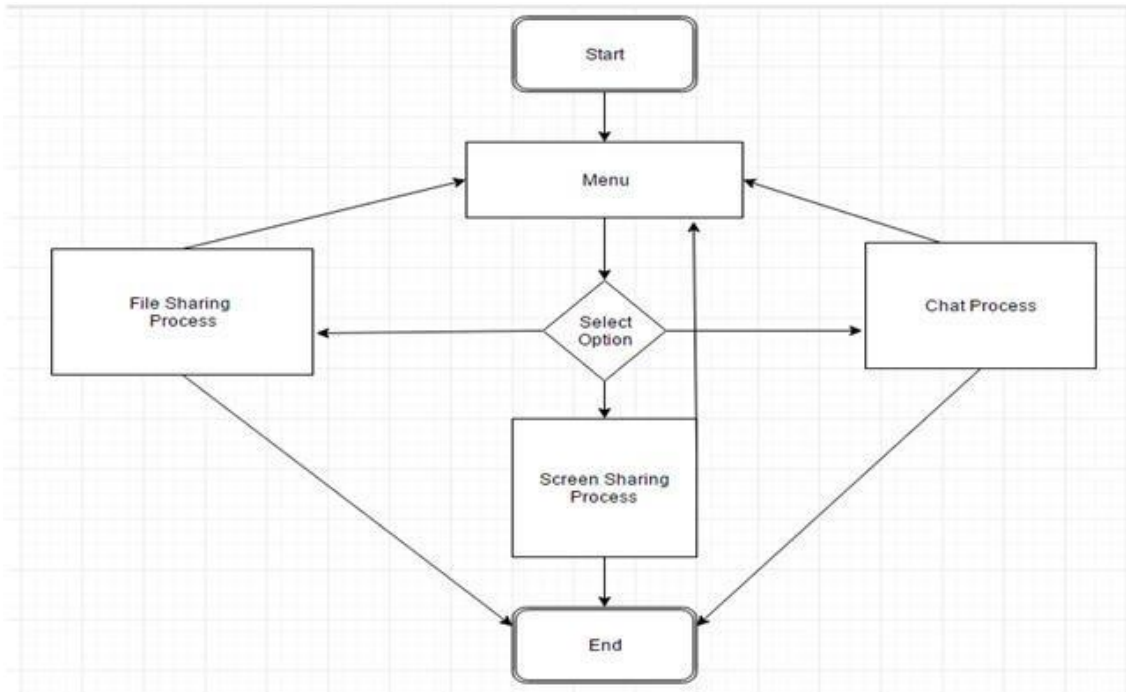
## 4.2.2. Collaborative Diagram

Figure 15. Collaboration diagram



### 4.2.3. Data Flow Diagram:

Figure 16. data flow diagram



### 4.3. Graphical User Interface:

The graphical user interface (GUI) is a type of user interface that allows users to interact with electronic devices through graphical icons and visual indicators such as secondary notation, instead of text-based user interfaces, typed command labels or text navigation. GUIs were introduced in reaction to the perceived steep learning curve of command-line interfaces (CLIs) which require commands to be typed on a computer keyboard.

Figure 17. Main Menu

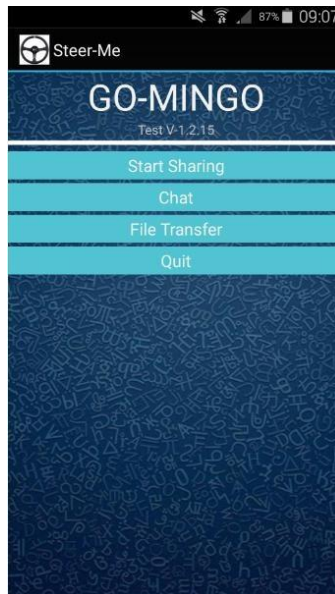


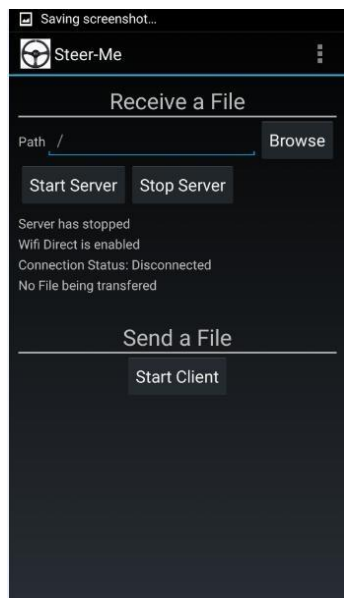
Figure 18. Screen Sharing



Figure 19. Chat



Figure 20. File Share



# Chapter 5

## System Implementation

### 5.1. Tools and Technique

#### 5.1.1. Android Studio

Android Studio is the official integrated development environment (IDE) for Android platform development. Android Studio offers even more features that enhance your productivity when we building Android applications, such as:

- A Gradle -based build system.
- Build variants and multiple APK file generation.
- Code templates to help you build common app features.
- A rich layout editor with support for drags and drop the me editing.
- Link tools to catch performance, usability, version compatibility,
- Code shrinking with Pro Guard and resource shrinking with Gradle.
- Built in support for google cloud platform, making it easy to integrate.

#### 5.1.2. Methodology

The SLDC (Software Development Life) Cycle used for the composing and programming of Steer-Me is Iterative method. Iterative method is more reliable and easy to use as we can change the code and easily remove the bugs after testing the project again and again so it's more reliable to use the iterative method to easily tackle with the bugs and the malfunctioning of the modules. A smart phone application which would be installed in 2 different mobile phones. One person can share the screen of the smart phone with the other person over a same network. If any of them is facing any sort of problem regarding the mobile phone software, he/she can provide the access of the phone to the other person via same network where the person whom access is given can access some of the features of the mobile phone of the person who has provided the

access. The first person will generate a password which would be provided by the other person who is going to control the screen of the other. When the second person will provide the similar password, the screen of the first person is ready to be controlled by the second. The second person now can access and control some of the mobile phone features of first person (Similar like in team viewer, one person controls the computer of the other person and solves the problems regarding the software). The person whose mobile phone is being controlled can stop giving the access to the other person at any time he/she wants. The other feature that this application would provide is "CHAT" that enables the users to trade the passwords with each other. The person who wants to access the other person's mobile features would first input the password that has been set by the first person and then will be able to gain the access to the other person's mobile phone features. Chat also includes the feature of file sharing which would allow the users to share pictures.

## **5.2. Language**

Java language has been used for development of this application. Java is a widely used object oriented language, which supports a wide range of libraries. It is also the mother language to android operating system.

PHP language is use in server in order to fetch/crawl data from the specific Google search result page.



# Chapter 6

## System Testing and Evaluation

Software testing is a process to test the application and find the errors and remove the errors. It is a process to check the requirement of application. The techniques of software testing are:

- Function testing
- Performance testing
- Acceptance testing

### 6.1. Functional Testing

Functional testing of application has to deal with the functionality of application. Functional testing is performed by giving the inputs to the function and getting the output. Then check whether the output is correct according to the input. The types of functional requirements are:

- Unit Testing
- Integration Testing
- Black Box Testing
- White Box Testing

#### 6.1.1. Unit Testing

In unit testing the system is divided into different modules and then test each module to check its functionality.

#### 6.1.2. Integration Testing

Test the complete project to check if the components of the project have been integrated properly. Components of our project include android studio activities files, PHP files.

### **6.1.3. Black Box Testing**

The technique of testing without having any knowledge of the interior workings of the application. The tester is oblivious to the system architecture and does not have access to the source code. Tester providing inputs and check outputs without knowing how and where the inputs are worked upon.

### **6.1.4. White Box Testing**

White box testing is the detailed investigation of internal logic and structure of the code. The tester needs to have a look inside the source code and find out which unit/chunk of the code is behaving inappropriately.

## **6.2. Performance Testing**

It is a process to determine the efficiency of the application. This process involves measuring the response time of the application. This process checks the reliability, effectiveness and accuracy of the application. Performance testing has following types.

- Stress Testing
- Configuration Testing
- Regression Testing
- Security Testing

## **6.3. Acceptance Testing**

Acceptance testing is process to test the acceptability of the system. This process evaluates the system to check whether the system is acceptable for delivery. In our project the installation of application in involve in acceptance testing. Installation of application involves:

- To verify the installation of the complete system on device
- To verify the functional and nonfunctional requirements of the system after installation on device.

## 6.4. Test Cases

### 6.4.1. Test Case 1: Screen Sharing

Table 6.1: Test case Table1

Test ID	TC-01
Test Name	Screen Sharing
Pre-Condition	User must have Wi-Fi direct.
Actions	1. Tap on the screen sharing button.
Exception Result	Screen Sharing

### 6.4.2. Test Case 2: Chat

Table 6.2: Test case Table2

Test ID	TC-02
Test Name	Chat
Pre-Condition	Friend must be added to your list.
Actions	1. Tap on the number whom to chat. 2. Click on chat option. 3. Write a message.
Exception Result	Message/query sent.

### 6.4.3. Test Case 3: File sharing

Table 6.3: Test case Table3

Test ID	TC-03
Test Name	File Sharing
Pre condition	You must add the friend whom to send an image.
Actions	1. Tap on the friend whom to send. 2. Tap on chat option. 3. Tap on image sending button. 4. Select image to send. 5. Tap on send.
Exception Result	1. Image sent 2. Invalid recipient

## 6.5. Tool Use

Table 7: tool use

<b>Tool Name</b>	<b>Tool Description</b>	<b>Applied on [list of related test cases/FR/NFR]</b>	<b>Results</b>
Android Studio	Studio is used to program the android smart phone application its coding is done in java language.	Applied on Chat, File Sharing, socket programming, and screen sharing.	All the modules are working fine.
Photoshop	It is used to design the layout and the design of the application i.e. home page and the various screen that	Applied on the front end of the application and on the different screens and layout files.	All of the products are working fine.

# Chapter 7

## Conclusions and Future Work

### 7.1. Conclusion

The product “Steer-Mee” is a dynamic software application which is providing many useful features. This is a big social contribution. One of main module is helping user Solve the Software regarding problem. The module and its feature have been given secure usage of device. The application is bringing a new idea. Although idea is quite simple but unique. The main purpose of making this project is that the person from whom one is asking for the solution and sort out the best possible solution by controlling the device of the problem faced person by using this application on both of the devices. This would bring a revolution in the smart phone world and no one has to search on the internet for the problem being faced by him/her and wasting the time reading blogs on the internet. One can give access to any of the friend who is trusted person and can also watch the each and every step being performed by the controller, this would help the problem facer next time solve his/her own problem. The product even can be featured more if some new idea is merged with previous idea.

### 7.2. Future Work

Steer-Mee application can be improved and extended by adding the functionality end-less. It will bring revolution in the field of screen controlling remotely. As this application is made for the first ever time and there is no other similar application to this. Now the users will be able to share the screens with one another easily using the Wi-Fi Direct technology. In addition to this the chat module will be able to share the passwords and the IP addresses with one another's.

# References

- [1].  
Esquiro, M. (2017).Team Viewer-Desktop sharing.*thesis-1*, 3.
- [2].  
archana jadhav<sup>1</sup>, vipul oswal<sup>2</sup>, sagar Madane<sup>3</sup>, Harshal Zope<sup>4</sup>. (2012). VNC ARCHITECTURE BASED REMOTE DESKTOP ACCESS THROUGH ANDROID MOBILE PHONES.
- [3].  
Global journals Inc. (USA).(2012). Global Journal of Computer Science and Technology Network, Web & Security.
- [4].  
Lowe,David G.(2004)"Client Server Sharing "
- [5].  
joby, A. P. (2016). Chat for Android Smartphone.