Recent Advances in Technology and Engineering (RATE-2017)
6th National Conference by TJIT, Bangalore

International Journal of Science, Engineering and Technology
An Open Access Journal

Real Time Audio Broadcast Application

[1]Abdul Mateen, [2] Abhilash C, [3] Abhishek K, [4]Arun N, [5]Srinivasa H P

Abstract

The proposed application can be used for the live broadcast of audio from the host or recording device to the client or streaming device that are connected to the same network. It also allows the host device to start and stop playback of the audio or video file on all the devices connected in the same group simultaneously. As an additional feature, the application provides a screen sharing feature which allows the client devices to stream the display of the host device in real time. This application provides an effective way of sharing content at a time when mobile multimedia information is becoming the dominant form of information.

Keywords: live, broadcast, audio, video, synchronization and screen share etc.

1. Introduction

Mobile multimedia information is becoming the dominant form of information. Majority of smartphones are equipped with the hardware that supports real time sharing of multimedia. The application will be using a local Wi-Fi hotspot over which the audio and video stream will be broadcasted. The main purpose of the application is to allow multiple devices to be able to stream the content playing on the host device with a negligible delay. The application will also allow users to utilize the hardware resources of different devices to play the same audio/video content.

Users consume audiovisual content through a multit ude of electronic devices which range from specialized audio devices, such as surround sound speaker systems, to display monitors, such as high definition televisions. In some instances, the user may use multiple devices in parallel to play the file. There are many Android applications like Chorus and Sound Seeder, which provide these features but they do not support video synchronization along with the audio playback. They also don't provide real time broadcasting. Most applications that provide broadcasting require both the host device and the other connected devices to have an active internet connection.

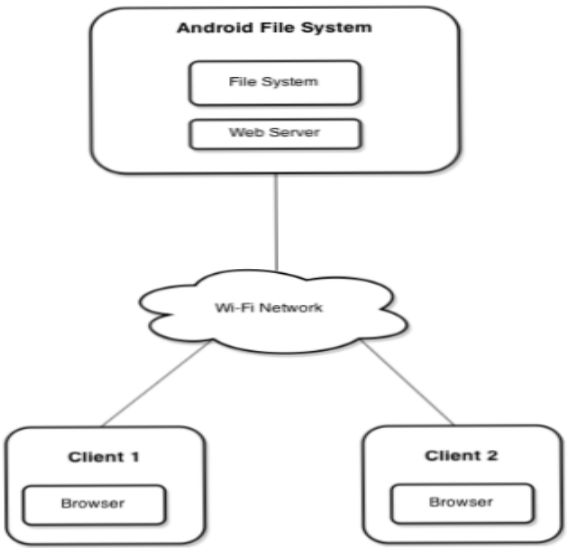

Figure 1: System Architecture

2. Related Works

A synchronization algorithm [1] compactly expressed in a single decision rule that is suitable for consumer cellular phones with limited computational resources. The proposed method reduces the computational amount required for synchronization by more than one-tenth of the conventional algorithm but it is mainly useful for only embedded processors.

DVB-H includes a forward error correction (FEC) scheme known as multiprotocol encapsulation (MPE) FEC, which provides equal error protection

(EEP) to the transmitted media streams. This paper analyzes the different error protection methods being proposed. The proposed method is more useful when the error rate is very high. [2]

The application designed to receive audio streams from multiple devices [3] simultaneously and to send them. The main design considerations of the application, such as audio recording and playing, audio coding or data transmission, are explained. The application provides efficient voice transmission over Bluetooth and Wi-Fi with the tools that Android public APIs offer. The mobile computing allows transmission of data, voice and video via a computer or any other wireless enabled device without having to be connected to a fixed physical link [4]. It Provides facility of streaming audio/video from one device to another without internet connection. But there might be a change in quality of the stream in case of large files. An Audio and video synchronization system is proposed in [5] that can be used to synchronize audio and video of a content playing on separate devices. There is no delay after the Audio/Video stream begins over multiple devices but an initial delay for stream to begin exists. Android Mobile Screen sharing and Recording application is helpful in many cases. If there is a need to illustrate about something then one can first record it and then share the screen someone. Whether you want to train other people on a particularly difficult program or show everyone how to fix a problem, recording a screen is an easy way to do it. In this we can share our device screen on to another device so that other can see what is happening on first device [6]. Majority of Smartphone equipped with hardware and cameras on their smart phones. [7] that support real time video processing and ad-hoc wireless communication. This Smartphone provide a real time video streaming over multiple hops between peer devices. Majority of Smartphone allows a real time video streaming over multiple wireless hops between peer devices. Phones create a client mesh network (ad-hoc net work of devices) by establishing a wireless link between the phones which are in a communication range. In this client mesh network each phones is able to produce and consume video and also relay to forward video to its next hop neighbour users can share what they see by using peer-to-peer video streaming. File Transferring is rapidly growing in the field of technology as the capability of devices to store files is increasing. The most direct way for transferring files between an android and laptop requires a data cable. File transfer between two android devices requires Bluetooth, which is slow, or file transfer apps that is required to be present on both the device. Also file transfer between an iOS and an Android is simply not possible due to the restriction of the iOS. This restriction between devices with different Operating Systems creates a problem for file sharing [8]. The emergence of multiple devices brings about the multi-screen environment in which we can utilize the screens (display areas) of multiple devices simultaneously for the convergence services and interact with multiple screens for better user experience. This paper discusses about the media synchronization issues in multi-screen environments that introduce new pattern in media consumption [9]. Peer-to-peer (P2P) is a very popular type of network communication for file sharing, where the data transfer is faster as it does not need to pass through the services of a server to share files and data. At present, peer-to-peer (P2P) communication is considered as a problem by both the wired and wireless Internet providers as it generates a large amount of traffic. This problem is substantial for mobile networks, as they not only have to serve users of mobile devices, but also the mobile subscribers that prefer mobility over traffic. [10]

3. Proposed System

The application allows for the broadcasting of audio over a network with a brief initial delay after which media is streamed in a continuous manner. It also deals with the synchronization of audio and video between devices connected to the same Wi-Fi network. The system synchronizes a first clock signal of the first device with the clock signal of the second device. It then detects the message received by the second system and the message is stamped with the playback start time. It compares this with the current clock time to calculate the time difference. The system then adjusts the playback at the second device based on the calculated time difference.

3.1 Device Initialization Module

The main purpose of this module is to establish a connection between the host device and the client devices. If all the devices are not connected to the wifi notify them and tell them to connect.

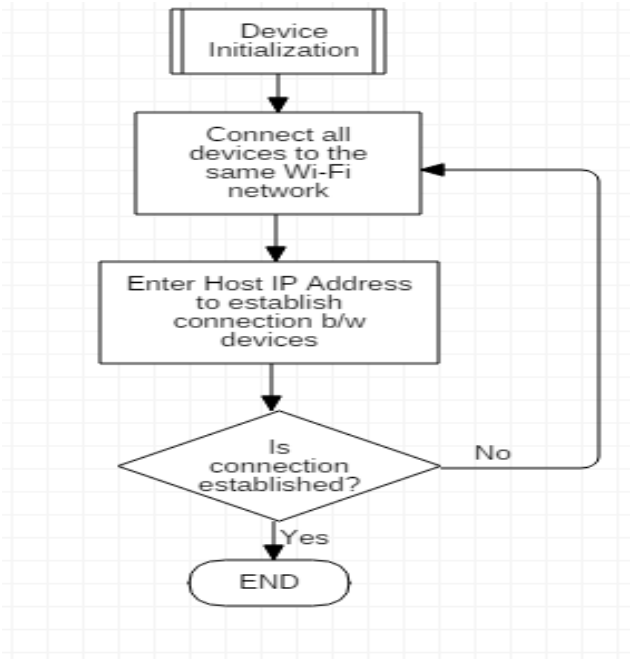

Figure 2: Diagram for device initialization module.

3.2 Audio and Video synchronization Module

This module is used to allow the client devices to stream the content playing on the host device in a synchronized manner. User can select from an option audio or video. If user selects audio then audio synchronisation will open else video synchronisation opens.

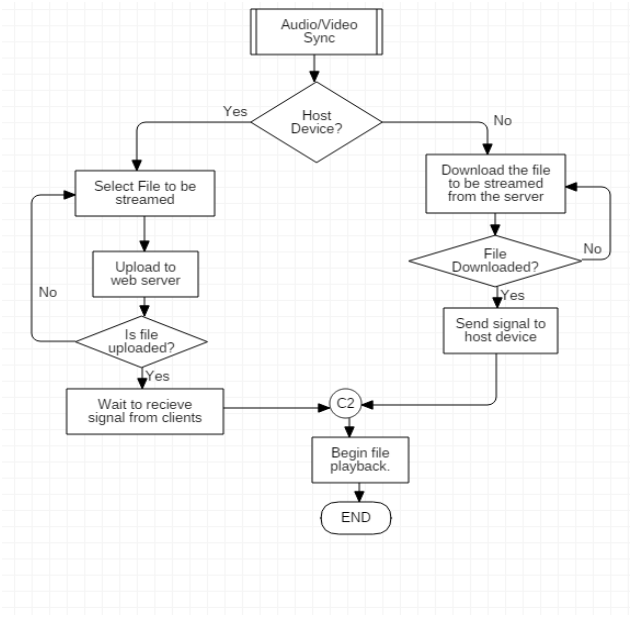

Figure 3: Diagram for Audio and Video synchronization module.

3.3 Live Audio broadcasting Module

This module allows the client devices to stream the audio being captured on the host device in real time. There will be an initial delay of 10 seconds before playback starts for the first time. Once the playback starts there will be continuous streaming.

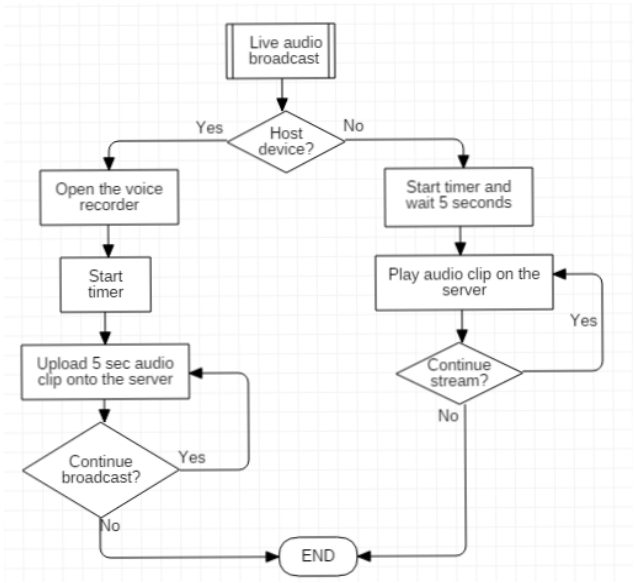

Figure 4: Diagram for Audio broadcasting module.

3.4 Screen sharing Module

This module allows the client device to stream the display of the host device by taking snapshots at regular intervals. It will take screenshots every 2 seconds and uploading.

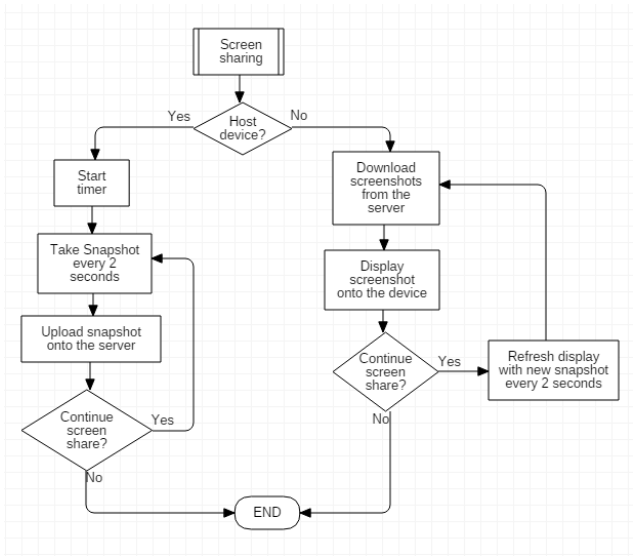

Figure 5: Diagram for screen sharing module

4. Application Process

The application begins with an activity asking the user if they are hosting the connection or joining an already existing connection. To host, the user must enter the group name and a password. To connect, the user will select a group from the list and enter its password.

After this, the application will display a screen with all the different features provide by the application. In audio broadcast, the host opens a recorder and uploads streams of 10 second audio clips of roughly about 15 to 20 KB to the server. The other devices in the group download these files and play them in a continuous manner until the stream is stopped.

The audio and video synchronization module is also done using the database. Once command is given to start playback on the host device, the database is updated with a value but the audio/video is delayed. The client devices check the database for any change at constant intervals. Once a change is detected, the clients update the server with confirmation. Once host receives the confirmation, the audio/video file starts playing on all the devices simultaneously.

5. Application Development

The application is mainly built for Android smart phones. It is developed in android platform using android studio. The languages being used are mainly Java and XML. The database used here is Firebase.

6. Results and Discussion

The proposed application has been tested using two android smart phones. Figure 6 shows the screen which is seen soon after we open the application. This page allows users to either create a group or to join a pre existing group. Based on the role selected it will take us to the respective page. For host device, a page is shown where group is to be created as shown in figure 7. After registering, user can now login to exiting group using the password. Figure 8 shows the available groups.

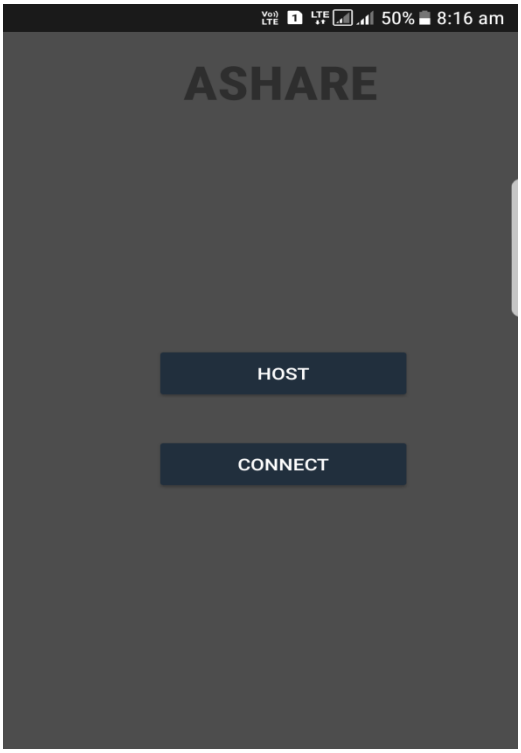

Figure 6: Home Screen

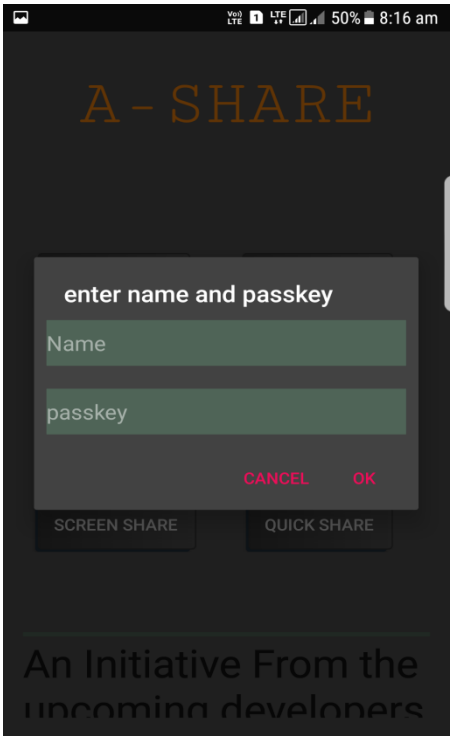

Figure 7: Creating Group.

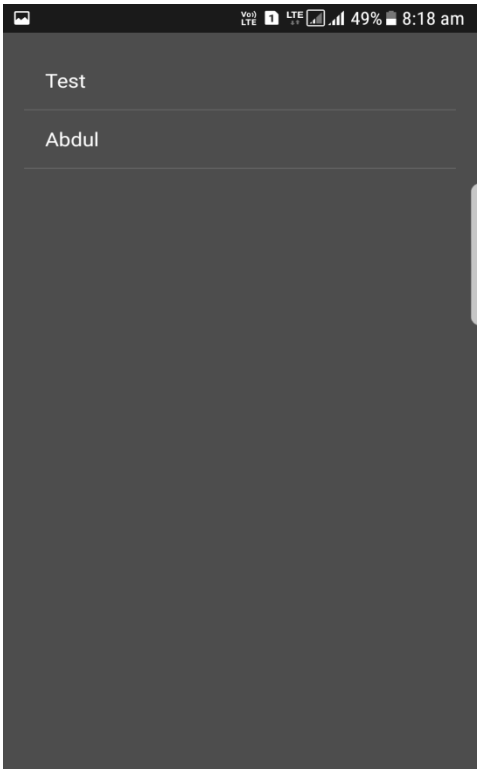

Figure 8: Joining Group (client side)

Once a group is joined, the next page shows all the different features and modules of the application. Figure 9 shows the features.

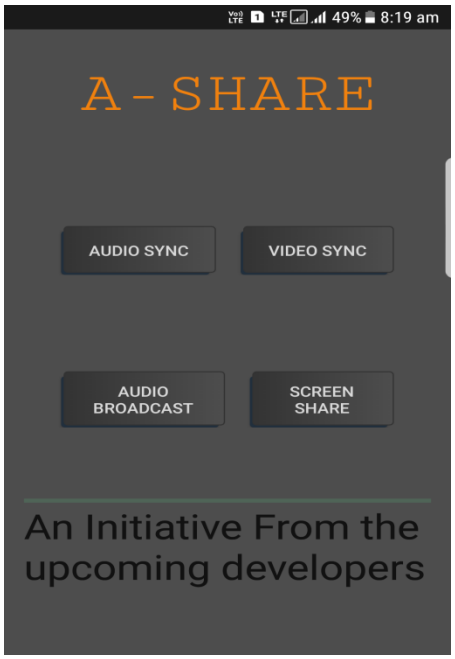

Figure 9: App Modules

On clicking the audio sync , the host gets the list of all the songs present in the internal storage from his any song can be played.

Figure 10: Audio sync (host side)

The name of the audio being played by the user is shown on the screen and is being played . Fig 11 shows the screen on the client side.

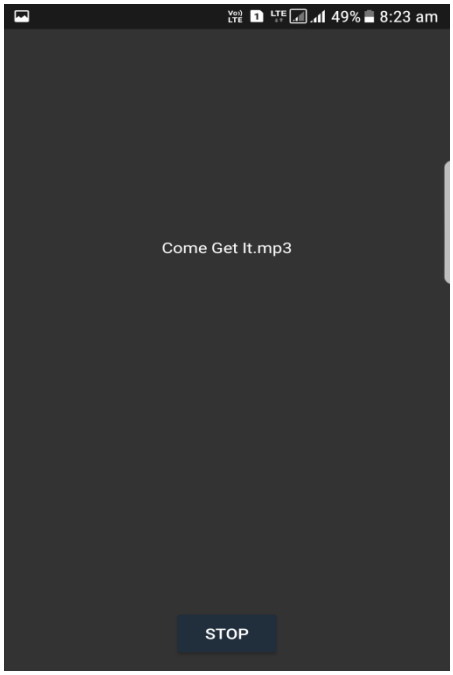

Figure 11: Audio sync (client side)

The list of videos present on the host side are being displayed so that the selected video can be played on both the devices.

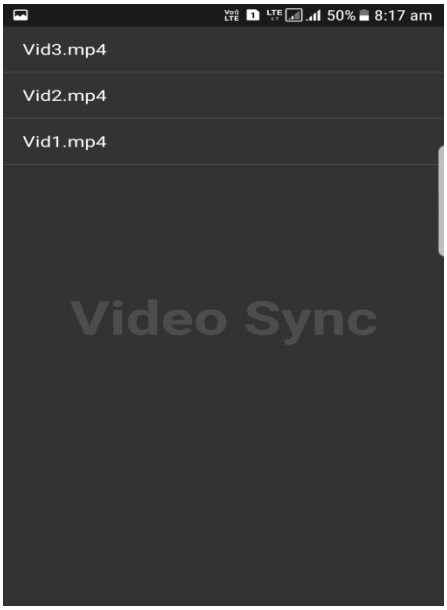

Figure 12: Video sync (host side)

The video starts playing on the client device when started by the host.

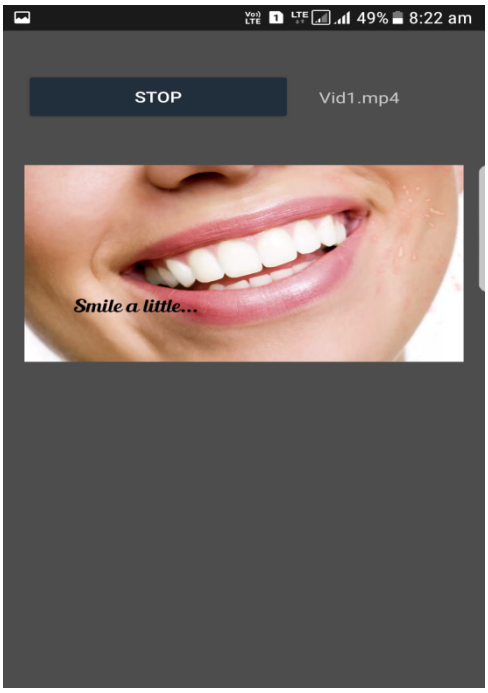

Figure 13: Video sync(client side)

The host can take screenshots and upload them to the server so that the client can view it.

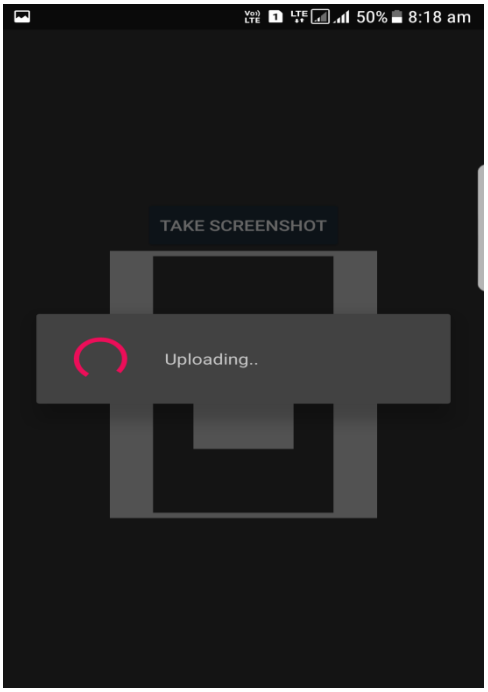

Figure 14: Screenshot (host side)

The screenshot taken in the host side can be viewed by the client by clicking the view image button.

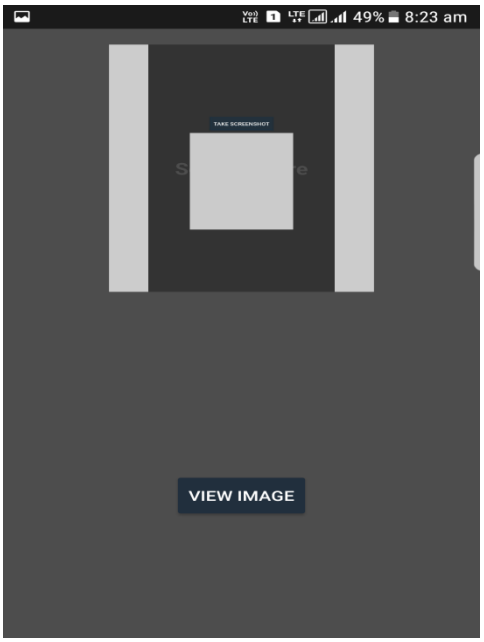

Figure 15: View screenshot (client side)

Any audio file can be shared among devices by using this feature

Figure 16: The list from which any file can be shared.

Conclusion

We present an Android Application that provides real time audio broadcasting from host device to client devices. There is an initial delay of five seconds after which the audio stream is uninterrupted. The application can also synchronize the audio and video playing on client devices with that on the host device. This is achieved by making the host device to send a signal to the client devices to notify them the time at which they should start playing the media. It also provides a screen sharing feature which is implemented by taking and sending snapshots of the application to the client. Screen sharing does not work when the application is running in background since Android does not allow this application to take snapshots of other applications.

References

[1] Chanwoo Kim, Kwang-deok Seo. "Efficient Audio/Video Synchronization method for video telephony system in consumer cellular phones", IEEE, 2006

[2] Miska M. Hannuksela, Vinod Kumar Malamal Vadakital and Satu Jumisko-Pyykko. "Comparison of Error protection methods for Audio Video Broadcast over DVB-H",EURASIP Journal, 2007

[3] Roman Belda, Pau Arce, Ismael De Fez. "Android real time audio communications over local wireless", Waves, 2012

[4] Madhav D.Ingale, Pranit Bhor, Ganesh Hargude, Samar Deshpande, "Peer to Peer Video, Audio streaming over Wi-Fi network on mobile device", IEEE, 2014

[5] Julien Marchand, Alexander Taboriskiy. "Synchronizing audio and video playback on multiple devices", Defensive Publication Series, 2015.

[6] Anureet Kaur "Development of Android mobile screen sharing and recording app", IRACST, Vol.6, No.2, 2016

[7] Preeti Yadvendra "The video streaming over Wi-Fi network application client on the android platform", IRACST, Vol.3, No.4, 2013

[8] Natasha Raul, Saumidh Mhatre, Sanket Pawar, Upendra Nagmallewar. "Wireless file transfer using web server on an android device", International Journal of Innovative and Emerging research in Engineering, Vol.2, Issue.2, 2015

[9] Geun-Hyung Kim, Sunghwan Kim. "Inter device media synchronization in multi screen environment", 2014

[10] Akmal Nurhananie Abd Rahman, Mohamed Hadi Habebi, Mohamed Ismail. "P2P File sharing for android devices supporting IEEE 802.15.4 Zigbee radios", ARPN Journal of Engineering and Applied Sciences, Vol.10, No.2, 2015

Author's Details:

[1234] Department of Computer Science & Engineering, T. John Institute of Technology, Bengaluru,
mateenhuda@gmail.com
abhigowdac@gmail.com
abhishek9520@gmail.com
aruntejasvi5@gmail.com
[5] Associate Professor, Dept of CSE, T. John Institute of Technology, Bengaluru, srinihp@yhoo.co.in