INTERNATIONAL JOURNAL OF RESEARCH – GRANTHAALAYAH

A knowledge Repository

Science

SHARE AND USE

Govinda Naik R[1], Manikanta N[2], Naram Pooja Reddy[3], S Krishna[4], Manjunath C R[5]

[*1, 2, 3, 4]UG Students, Department of CSE, SET-JU, Bangalore, India

[5] Associate Professor, Department of CSE, SET-JU, Bangalore, India

Abstract

Android is one of the most user friendly mobile platforms that have grasped the attention of smart phone users around the globe. According to stats available there are 1.4 billion people who have used Android mobile platform. And this has led to the development of android application exponentially; there are around 2.8 million android applications available in play store. Many of the available application do not fit into low specification smart phones due to some reasons.
Share and Use is an android application that lets you use another smart phones application by your phone without actually installing it on your phone. Share and Use lets user to use application installed on other phones and access applications on their smart phones. Share and Use helps phones with lower specification android phones to use applications that are not be able to install in them.

Keywords: Android; Friendly Mobile Platform; Share and Use.

1. Introduction

In the past, mobile phones were mostly about making phone calls. They had a number pad, a digital phone book and a pick-up/hang-up button and not much more. Now smart phones offer so much more – they're really fully-fledged computers that you can fit in your pocket. They can run programs, games, access the internet, send email and much more. Nearly all smart phones now use touch screen controls. Instead of having hardware buttons like before, one side of the phone is taken up mostly by a touch screen that you control using taps and gestures. There aren't even any number buttons; when you want to make a call, a number pad will pop up on the touch screen. Becoming familiar with a smart phone can take a little bit of practice. But when you do become familiar with it, you'll find that a smart phone can do more than you ever thought possible on a mobile phone.

Constraint of phones:
1) Limited Memory.
2) Limited internal Storage.
3) Screen resolution.

So to overcome from above mentioned constraints we need to develop an application which can overcome all of the above constraints. Lots of applications are available which cannot be installed in smart phones. Sharing an Application from device, installing and using is a huge task for users. In this new era of Smartphone's most of the works are carried using smart phones with apps in it. Most of the phone does not have the capability to install all the apps and use them.

1.1. Problem Statement

Too many readily available applications are available on store but all of those apps do not fit into one mobile or several. Even the modern specification of the smart phones are upgraded it is impossible to have many utility apps in one. And phones running on lower specification cannot enjoy applications developed for higher versions of android.

1.2. Scope

There are readily available applications in android store for file and application transfer. Lower specification android phones suffer memory challenge and low hardware components. The scope of the proposed work is to develop an android application that helps android devices of lower specification to have an access to the applications that are made for high end android devices. The user does not use the application or document files frequently, uses once a while this system provides the user to use the particular selected file for once

2. Existing Systems

2.1. SHAREit

SHAREit is an application to transfer files from a Wi-Fi compatible device to another similar one. Users can use it to transfer files including photos, videos, music, contacts, apps and any other files.

The scenarios were analyzed for common properties; from these we derived user and system/content provider requirements, and then the technical requirements. These requirements were then prioritized according to whether we intended to support them in the demonstrator system or study the issues.

The system requirements fell into the following categories: The Share it Set-Top box, types of Content, content distribution, security, content Creation, rights Management, discovery of content and User Groups, personalization.

Rather than give a detailed list of the system requirements in this paper, a set of example applications is given in the following section which illustrates the kinds of applications and scenarios that the Share it platform supports.

2.2. InkWire

Inkwire is an Android-to-Android screen sharing and remote assistance application. With it, two individuals can pair devices without the need for ADB or a computer. Once two people are paired via an access code, they can get to work on whatever they may need to do.
STEPS

- Just Install InkWire app both Host and Client phones
- Now open InkWire app on Host device (device that you want to access) and tap on Share.
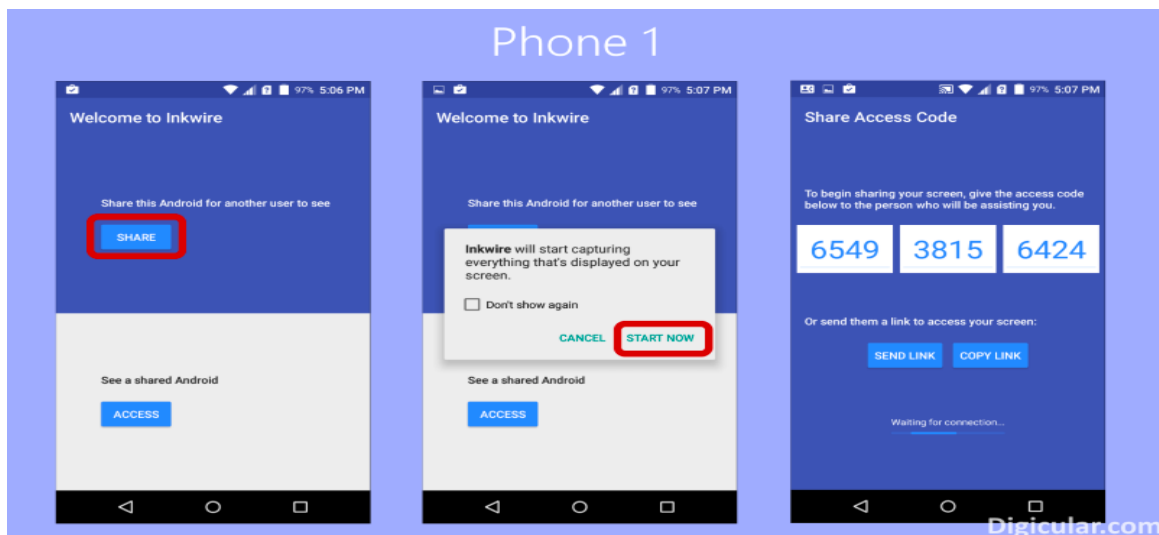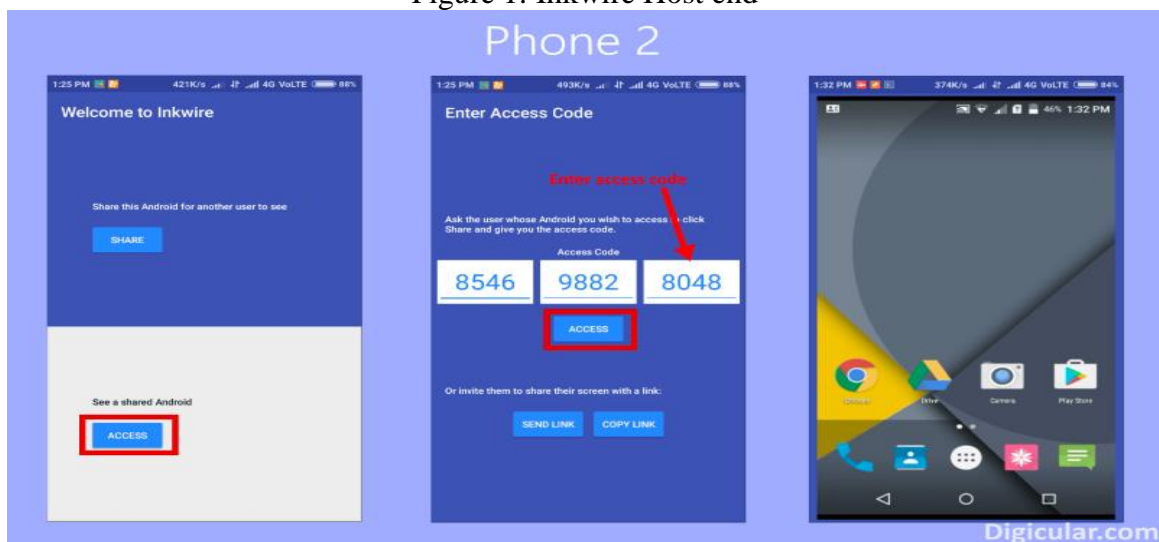- Now you will see a 16-digit code.

Figure 1: Inkwire Host end

Figure 2: InkWire Client end.

3. Proposed Idea

As android has gained its popularity, there is an exponential growth in android application development. As a result millions of applications are in existence. Storing more apps is headache for the user. Share and use helps in sharing an application between the devices not necessarily installed in both the devices. If an application is present in one device then the same can be used at the other end without installing it in that device.

4. Design and Implementation

4.1. Architecture of the Proposed System

The proposed system architecture is simple that contains the interaction calls that are used while the system is functioning. The devices are connected over Wi-Fi connection and request will be sent by the connecting device to the host device. Then the file to be viewed will be transferred to the host. Concurrently the casting happens by reconnecting of the Wi-Fi and then the required file will be viewed by the requesting device only. Interaction ends between the devices once the host device exits the application.
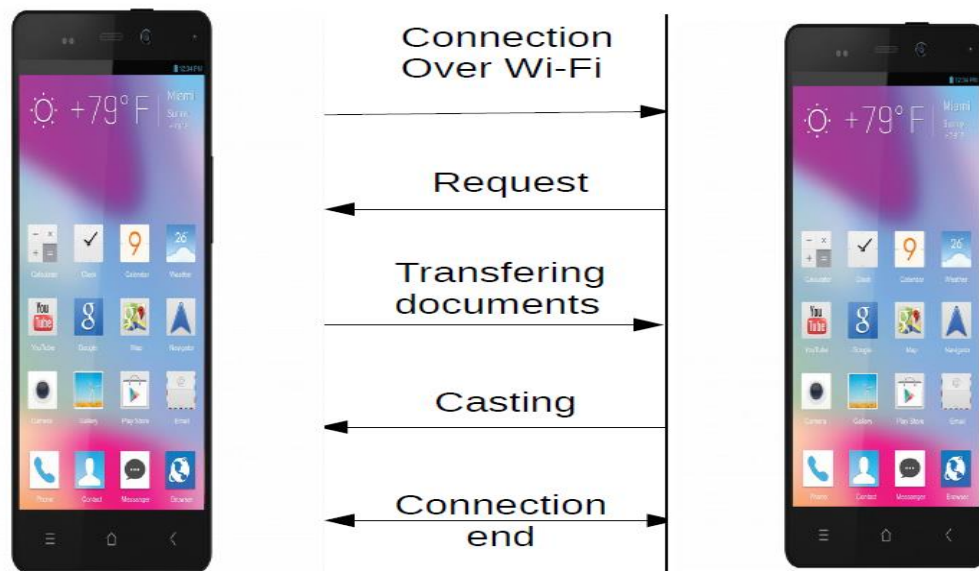

Figure 3: System Architecture

Control Flow Diagram

At the start of the application it checks whether the Wi-Fi is enabled or not. If not it enables the Wi-Fi of the device. Next stage it searches for the nearby Wi-Fi direct device available. The connection will be established to the selected device. Then the host device gets the file from another device seeking service from it. Accessing and viewing of the file permission will been given by the host device
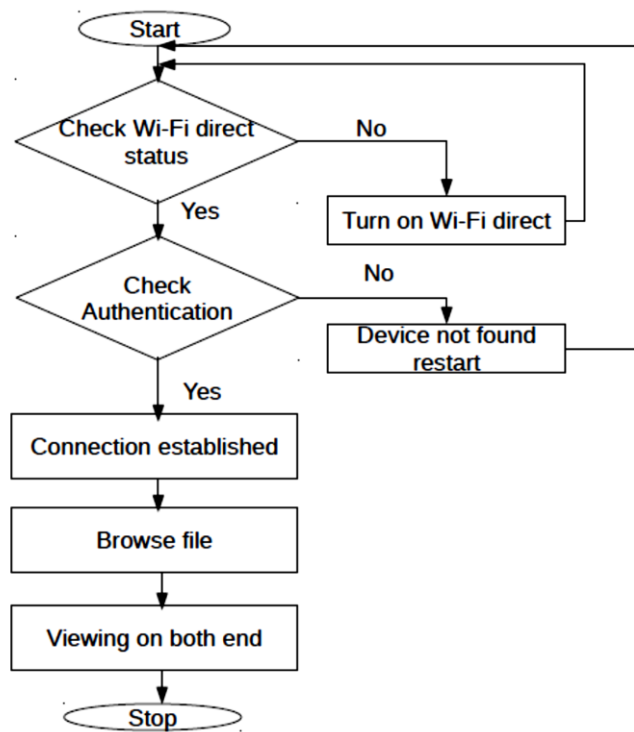
Figure 4: Process flow diagram

As the application starts on both the devices, it checks for the Wi-Fi status of the device. The Wi-Fi is enabled even if it is in disabled and then the connection is established between the devices which want to interact with each other. If the devices fail to connect to each other then the controls go backs to initial state. The application won't further service till it connects to one of the Wi-Fi enabled device nearby. Once the connection is established, the file to viewed (e.g., Image,Video, pdf) is selected and the same will be accessed by the other end device. Then the file is opened by the appropriate application available by the host device and viewed at both the end devices.

Sequence Diagram

Let us take two devices Host A and Host B where Host A is the device which needs service from the other host (i.e., Host B). Host B will send a request for the connection to the Host A, intern Host A accepts the request and sends the response. And the connection is established between the hosts. Host A shares the file for which it needs service from another host. Host B receives and stores in a temporary storage space and opens it internally by the suitable application available in it. Then the screen is shared with the service seeker (Host B).
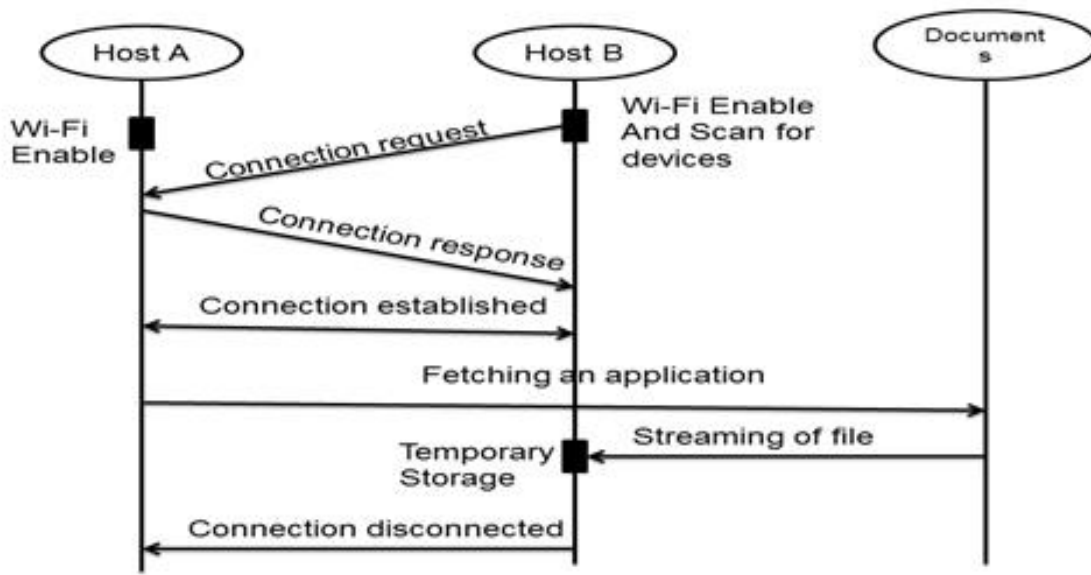
Figure 5: Sequence Diagram

By taking an example of a scenario we could better explain the sequence process happening in the application. Host A is the device which has a PDF file in it but not the supporting application to access it. Host B is the device that has PDF supporting application. Host A will access the Host B's application through Share and Use. Host A and Host B will be connected to each other through Wi-Fi. Then Host B finds the file to be opened and opens it and the same is viewed at the other end.

5. Conclusions

New era smart phones offer so much more – they're really fully-fledged computers that you can fit in your pocket. They can run programs and games, access the internet, send email and sending files and videos. In present era everyone is having mobile and in the same way thousands of applications are available in play store. Users are facing the problem to install apps or downloading some files because of storage consumption and lack of hardware specification. In the proposed system storage are not needed you can just view the file after viewing file will be closed no storage will be required for this. This will be fast when compared to other because internet connection is not needed for this we are connected this by using Wi-Fi and hotspot. This project can be future extended by giving interaction to the client device while the server device is casting the screen and the server device could be used for other activity. The work may be extended to all the type of files (Ex Image, Video etc) for the application.

References

[1]   Frans de Jong, "Deliverable 3: Description of example applications", Public deliverable from the Share It! project. IST-2000-28703.
[2]   J. McParland, M. Morris, S. Leban, A. Parnall, A. Hickman, M. Ashley, Haataja, F. de Jong, "myTV: a practical implementation of TV-Anytime on DVB and the Internet", IBC, 2001.

[3]   J. Walker; O.J. Morris; B. Marusic "Share it! - the architecture of a rights-managed network of peer-to-peer set-top-boxes"

[4]   D. Brookshier, "JXTA: Java P2P Programming", SAMS, 2002.

[5]   S. Kosta, et al., "Clone2Clone (C2C): Peer-to-Peer Networking of smart phones on the android ", in 2013.

[6]   K. Lee, J. Lee, Y. Yi, I. Rhee, S. Chong, "Mobile data offloading: how much can wifi deliver?", SIGCOMM., 2010.

[7]   H. Luo, R. Ramjee, P. Sinha, L. E. Li, S. Lu, "Ucan: a unified cellular and ad-hoc network architecture", MobiCom, 2003.

*Corresponding author.
E-mail address: govindnaik1995@gmail.com