

CHAPTER 1

INTRODUCTION

1.1 Introduction to the project

Remote Bluetooth this useful for controlling the Pc via Bluetooth connectivity within the limited distance. If we have this application in our mobile, no need Mouse. Therefore without Mouse we can Control Pc with the help of Bluetooth connectivity which is the main use of this application. The major defect of this device is, if both of the cells are out of Bluetooth range then it's difficult to get connection between both the smart phone and pc.

In this application mainly there are two Parts android and java. In this application Client side is android and server side is java. Server is able to find all the active Bluetooth devices but it can connect with only one device at a time. There is no way to connect simultaneously with all devices.

CHAPTER 2

LITERATURE SURVEY

2.1 EXISTING SYSTEM

Currently there are different applications which control the PC with Remote Bluetooth. In that one application in the existing systems remote Bluetooth is used for controlling the system but for controlling PC a separate device is used and in that it contains only four arrow keys: up, down, right, left.

In another application the Remote Bluetooth is used in the smart phone; it also does the operation only of the arrow keys but the Remote Bluetooth is for many other applications. Some applications are done in remote controlling but they are not in Bluetooth; they are in run on the Wi-Fi and hotspot. It is difficult to implement. In the existing systems not have the operation like cut and copy and paste and not have voice controller.

2.2 PROPOSED SYSTEM

Nowadays, the number of appliance services and their complexity has increased. There is a need to get familiar with the different operations of many remote control manufacturers, which is very confusing. To overcome this problem they have built these remote controllers, which are called Remote Bluetooth. This Remote Bluetooth is user friendly; it is made as the Android app. It supports for the Android higher version; it supports (7.1). It does the all most all operation. It does operation of four arrow keys such as up, down, right, left. And it does the operation of cut and copy and paste; we can cut the file, we can copy the file, we can paste the file. And the app contains the select button for select required file to open. Our project main feature we can control the PC by giving voice commands. The Remote Bluetooth can controlling with in the distance range is 0m to 50m.

Our complete project has two parts: one is Android and another is Java. A Client side Android and Server side is a Java. The Android is used in the smart phone and the Java is used in the PC.

2.2.1 FEATURES

- It is used in the meeting hall for giving PPT
- It can controlled by voice commands
- It is easily connect with the Pc
- In this we can do the cut the file, copy the file and paste the file
- It is support for the lower version of android device
- It is support for the higher version of android device
- It can controlled from 50m distance

2.3 IMPLEMENTATION TOOL

What we have used in our Project:-

J2ME

- It stands for Java 2 micro edition for applications which run on resource constrained devices (small scale devices) like cell phones, for example games.
- It is a stripped-down version of Java targeted at devices which have limited processing power and storage capabilities and intermittent or fairly low-bandwidth network connections. These include mobile phones, pagers, wireless devices and set-top boxes among others.
- J2ME is a reduced version of the Java API and Java Virtual Machine that is designed to operate within the limited resources available in the embedded computers and microcomputers.
- J2ME is a subset of J2SE with some API's added specifically for the wireless devices. Just like for J2SE we have JVM (Java Virtual machine) to run the java applications; the J2ME applications run on KVM (kilo Bytes Virtual machine), which is basically a subset of JVM having limited resources, for e.g., lacking support for floating point calculations.

J2ME Technology

- J2ME is build upon the configurations, profiles and other optional packages.
- Configuration is the set of basic APIs on top of which the additional packages are built. Configuration tells you how big the KVM is and helps in the interaction with the device (via APIs)
- There are two types of versions available as of now
 - CLDC 1.0
 - CLDC 1.1
- CLDC stands for Connected Limited Device Configuration.
- Profile gives you information about the mobile device and extends capabilities of configuration.
- It gives the control for the user interface, records storing etc. There are mainly two versions of this.
 - MIDP 1.0
 - MIDP 2.0
- MIDP stands for Mobile Information Device Profile.
- One major difference between the 1.0 and 2.0 is the advanced game APIs and the better key controls support 2.0 provides.

J2ME Architecture

- J2ME uses configurations and profiles to customize the Java Runtime Environment (JRE).
- As a complete JRE, J2ME is comprised of a configuration, which determines the JVM used, and a profile, which defines the application by adding domain-specific classes.
- The configuration defines the basic run-time environment as a set of core classes and a specific JVM that run on specific types of devices.
- The profile defines the application. Specifically, it adds domain-specific classes to the J2ME configuration to define certain uses for devices.
- The following graphic depicts the relationship between the different virtual machines, configurations, and profiles.
- It also draws a parallel with the J2SE API and its Java virtual machine.
- While the J2SE virtual machine is generally referred to as a JVM, the J2ME virtual machines, KVM and CVM, are subsets of JVM. Both KVM and CVM can be thought of as a kind of Java virtual machine.

Bluetooth

- One of the mobile phone facility is popular right now is a Bluetooth facility, which is almost all types of mobile phones have such facilities. Bluetooth is a well-known, short-range technology for Wireless Personal Area Networks (WPAN).
- Bluetooth is a wireless communication protocol, since it's a communication protocol, we can use Bluetooth to communicate to other Bluetooth enabled devices.
- Bluetooth is like any other communication protocol that you use every day, such as HTTP, FTP, SMTP or IMTP. Bluetooth has clients-server architecture; the one that initiates the connection is the client, and the one who receives the connection is the server.
- Bluetooth is a great protocol for wireless communication because it's capable of transmitting data at nearly 1MB/s, while consuming 1/100th of the power of Wi-Fi. Bluetooth is a short-range universal wireless connectivity standard for electronic appliances and mobile devices.
- A Bluetooth connection is the result of a complex device pairing process, and provides a channel on which many data services can be provided, such as voice, internet communication, file sharing, printer connection etc.

OBEX Protocol

- OBEX (abbreviation of Object Exchange, also termed IrOBEX) is a communications protocol that facilitates the exchange of binary objects between devices. It is maintained by the Infrared Data Association but has also been adopted by the Bluetooth Special Interest Group and the SyncML.
- The OBEX protocol provides support for object exchanges, and forms the basis for Bluetooth profile such as the synchronization profile and the File Transfer profile. The OBEX API is a separate optional package we can use either with the core Bluetooth package or independently.
- The OBEX protocol can be used over several different transmission media- Wired Infrared, Bluetooth radio and the others.

CHAPTER 3

HARDWARE AND SOFTWARE REQUIREMENTS

3.1 Hardware Required

The selection of hardware is very important in the existence and proper working of any software. In the selection of hardware, the size and the capacity requirements are also important.

Processor	:	Pentium-111 or higher
Primary memory [RAM]	:	4 GB DDR2 or DDR3
Secondary memory	:	40 GB HDD
Bluetooth	:	20 miter capacity
Mobile pone	:	android
Pc or laptops	:	with Bluetooth

3.2 Software Required

One of the most difficult tasks is that, the selection of the software, once system requirement is known is determining whether a particular software package fits the requirements. After initial selection further security is needed to determine the desirability of particular software compared whit other candidates. This section first summarizes the application requirement question and then suggests more detailed comparison.

Following are the software required:

Java development kit 8.0		
Net Beans 8.2		
Client side scripting	:	android based java scripting
Server side scripting	:	JSP
Operating system	:	windows xp/7/8/8.1/10

CHAPTER 4

SOFTWARE REQUIREMENT SPECIFICATION

Software requirement specification bridges the gap between the client or the user and the system developer. This documents the user needs accurately. SRS is the official statements of what is required of the system developer it includes both user requirements for the system and detailed specification of system requirements.

4.1 Purpose

The purpose of this report is to define the use of remote Bluetooth, this remote Bluetooth is user friendly it is made as the android app it support for the android higher version it support (7.1) .It do the all most all operation It do operation of four arrow keys such as up, down, right, left. And it do the operation of cut and copy and paste we can cut the file, we can copy the file, we can paste the file. and the app contend the select button for select required file to open .our project main the feature we can control the pc by giving voice commands. The Remote Bluetooth can controlling with in the distance range is 0m to 50m this remote Bluetooth is user friendly it is made as the android app it support for the android higher version it support (7.1) .It do the all most all operation It do operation of four arrow keys such as up, down, right, left. And it do the operation of cut and copy and paste we can cut the file, we can copy the file, we can paste the file. and the app contend the select button for select required file to open .our project main the feature we can control the pc by giving voice commands. The Remote Bluetooth can controlling with in the distance range is 0m to 50m.

4.2 Scope

This document describes the requirements of the system. It is meant for use by the developers and will be the basis for validating the final delivered system. This document details about the use of remote Bluetooth.

4.3 Functional Requirement:

The main function of this application to connect the pc with the smart phone . This application vitally based on Server and Client. Therefore mobile can search all the active Bluetooth devices. But the limitation of this application is user can connect with only one at a time.

CHAPTER 5

SYSTEM DEFINATION

5.1 Identification of the need:

Today in the generation of the mobile technology it has become need as well as fashion of using mobile in day-to-day life. Now a days it is necessary to save the energy of the system user

Bluetooth connect enables system user to control the computer system by mobile using Bluetooth. The user can control the computer system by mobile using Bluetooth. The user can control the system in the range of within 50 meters of the surrounding area some time engineer is giving presentation in a meeting he don't have more time to move to system and change the presentation slide so he can control the system and he can present his presentation in an interesting form.

5.2 Preliminary investigation:

The selection describes the need of our mobile and system application “Bluetooth control “in the controlling system. It can be used in many more platform like employee in the organization, or student in the school and/or collage and many more platform. It is easy to control the system.

The user can control the computer system by his mobile phone in the range of 50 meter. He can control it without touching the computer system

This feature of our project help in controlling the pc and fast accessing. Another feature of our project is we use the mobile application controlling system it can be done within a few second and save the time of the customer.

5.3 Feasibility study:

Feasibility study is done to answer very basic questions i.e. “is the project feasible”? A feasibility analysis is performed in order to evaluate possible solution and recommended the most faceable one. Feasibility study is performed on the system proposed to measure its workability and impact on the organization with its its ability to meet the user needs and efficient use of the resources

The feasibility study is a focused study, which aims to, answer a number of questions:

1. Does the system contribute to overall objectives of the organization?
2. Can the system be implemented using current technology and within the given cost and schedule constraints?
3. Can the system be integrated with other system which are already in place?The issue of whether or not the system contribute to business objective is critical. Carrying out feasible study involves assessment, information collection and report writing. The information assessment phase identifies the information, which is required to answer the above three sets of question.

5.3.1 Technical feasibility:

The technical feasibility is to answer the following question:

1. Can this work of the project be done with current equipment, existing software technology and available personnel?

If new technology is required, what is the likelihood that it can be developed? The technology used in for implementation of “blue connect” J2ME, Bluetooth, OBEX protocol, Wireless Toolkit.

5.3.2 Economical feasibility

The economical feasibility is intended to answer the following questions: -

1. Are the sufficient benefits in creating the system to make cost? Acceptable?
2. Are cost of not creating the system so great that the project must undertake?

Here by using our application we can control our system by mobiles in very fast in the old system access would be very slow whereas the urgency might be more.

CHAPTER 6

SYSTEM DESIGN

System design phase follows the system analysis. The objective of the system design is to improve the existing system or designs a new system as the case may be, and implementation the new system within the improved facilities. Generally a software system is designed into two phase's high-level design and low-level design. Design is more active analysis. Design is the first step in the development of any system or product. Design can be defined as “the process of Applying various techniques and principles for the purpose of defining a device, a process or system in sufficient detail to permit its physical realization”.

It involves four major steps they are:

1. Understanding how the system is working now;
2. Finding out what the system does now;
3. Understanding what the new system will do; and
4. Understanding how the new system will work

6.1 Data Flow Diagram (DFD)

The data flow diagram is used for classifying system requirements to major transformation that will become programs in system design. This is starting point of the design phase that functionally decomposes the required specifications down to the lower level of details. It consists of a series of bubbles joined together by lines.

Bubbles: Represent the data transformations.

Lines: Represents the logic flow of data.

Data can trigger events and can be processed to useful information. System analysis recognizes the central goal of data in organizations. This dataflow analysis tells a great deal about organization objectives are accomplished.

Dataflow analysis studies the use of data in each activity. It documents this finding in DFD's. Dataflow analysis give the activities of a system from the viewpoint of data where it originates how they are used or hanged or where they go, including the stops along the way from their destination. The components of dataflow strategy span both requirements determination and system's design. The first part is called dataflow analysis.

As the name suggests, we didn't use the dataflow analysis tools exclusively for the analysis stage but also in the designing phase with documentation

NOTATIONS USED IN DATA FLOW DIAGRAMS

The logic dataflow diagrams can be drawn using only four simple notations i.e., special symbols or icons and the annotation that associates them with a specific system. Since the choice of notation we follow, does not affect impede or catalyze the system process; we used three symbols from YOURDON notation and one from Gain and Sarson notation as specified below.

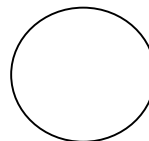
Element References

Symbols

Data Flow Process



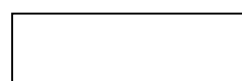
Process



Data Store



Source or Sink



Description:

- Process : Describes how input data is converted to output Data
- Data Store : Describes the repositories of data in a system
- Data Flow : Describes the data flowing between process, Data stores and external entities.
- Sources : An external entity causing the origin of data.
- Sink : An external entity, which consumes the data.

Context Diagram:

The top-level diagram is often called a “*context diagram*”. It contains a single process, but it plays a very important role in studying the current system. The context diagram defines the system that will be studied in the sense that it determines the boundaries. Anything that is not inside the process identified in the context diagram will not be part of the system study. It represents the entire software element as a single bubble with input and output data indicated by incoming and outgoing arrows respectively.

Types of data flow diagrams

DFDs are two types

1. Physical DFD

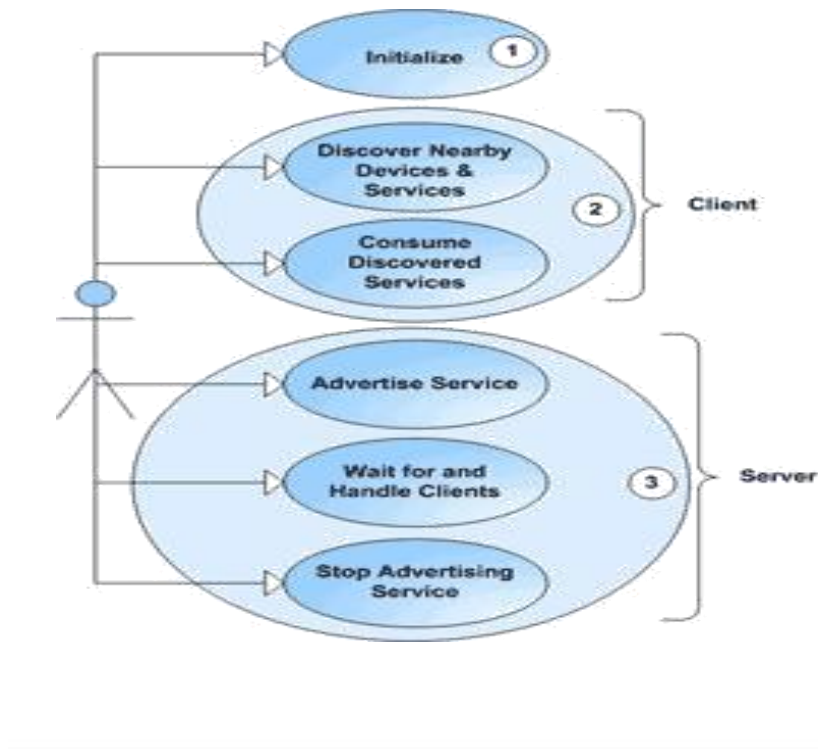
Structured analysis states that the current system should be first understand correctly. The physical DFD is the model of the current system and is used to ensure that the current system has been clearly understood. Physical DFDs shows actual devices, departments, people etc., involved in the current system

2. Logical DFD

Logical DFDs are the model of the proposed system. They clearly should show the requirements on which the new system should be built. Later during design activity this is taken as the basis for drawing the system’s structure charts.

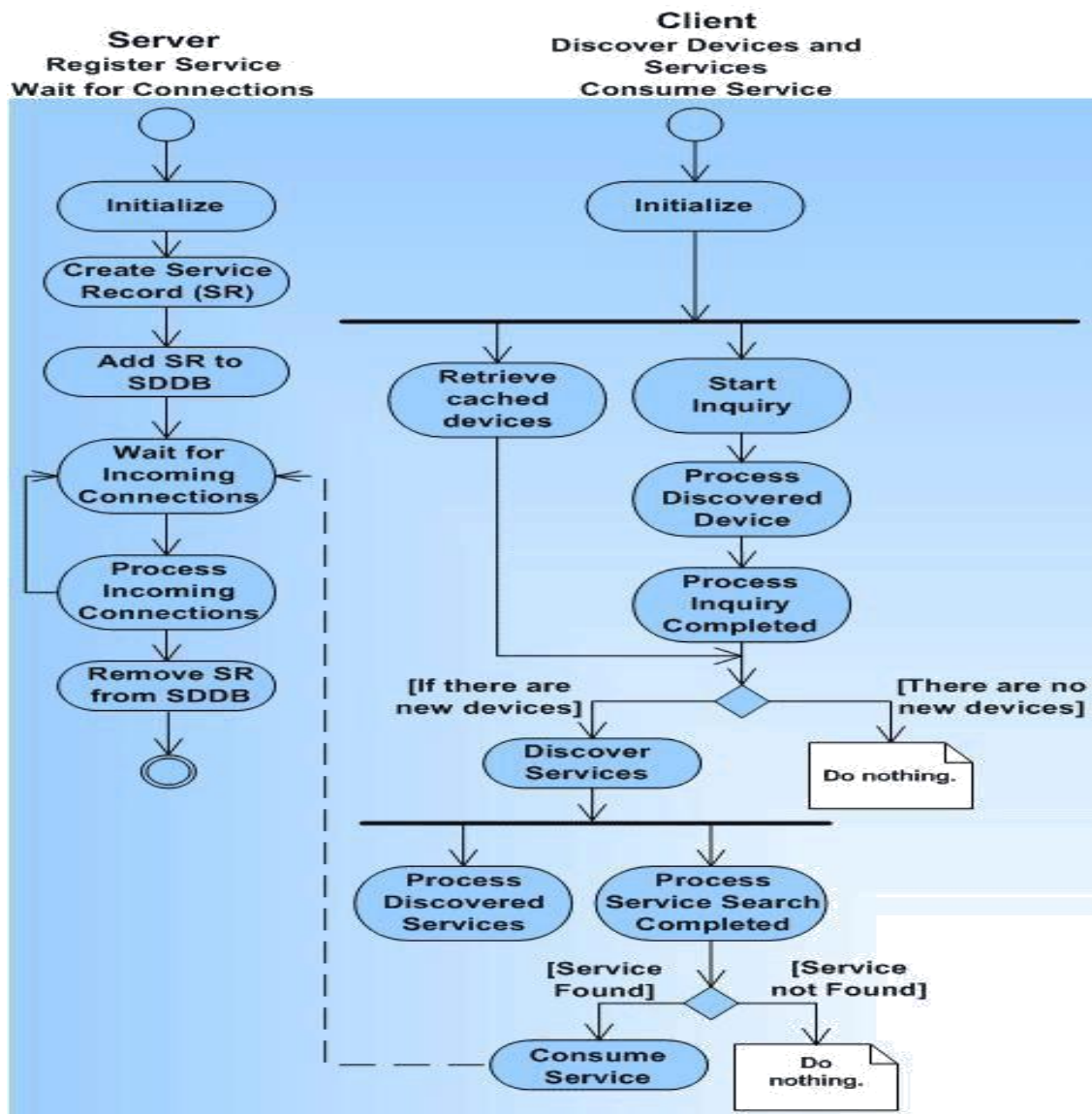
6.2 Use case Diagram

Diagrams



- **Initialization** - Any Bluetooth-enabled application, server or client, must first initialize the Bluetooth stack.
- **Client** - A client consumes remote services. It first discovers any nearby devices, then for each discovered device it searches for services of interest.
- **Server** - A server makes services available to clients. It registers them in the Service Discovery Database (SDDb), in effect advertising them. It then waits for incoming connections, accepts them as they come in, and serves the clients that make them. Finally, when the service is no longer needed the application removes it from the SDDb.

Activity Diagram:



6.3 Modularization Details

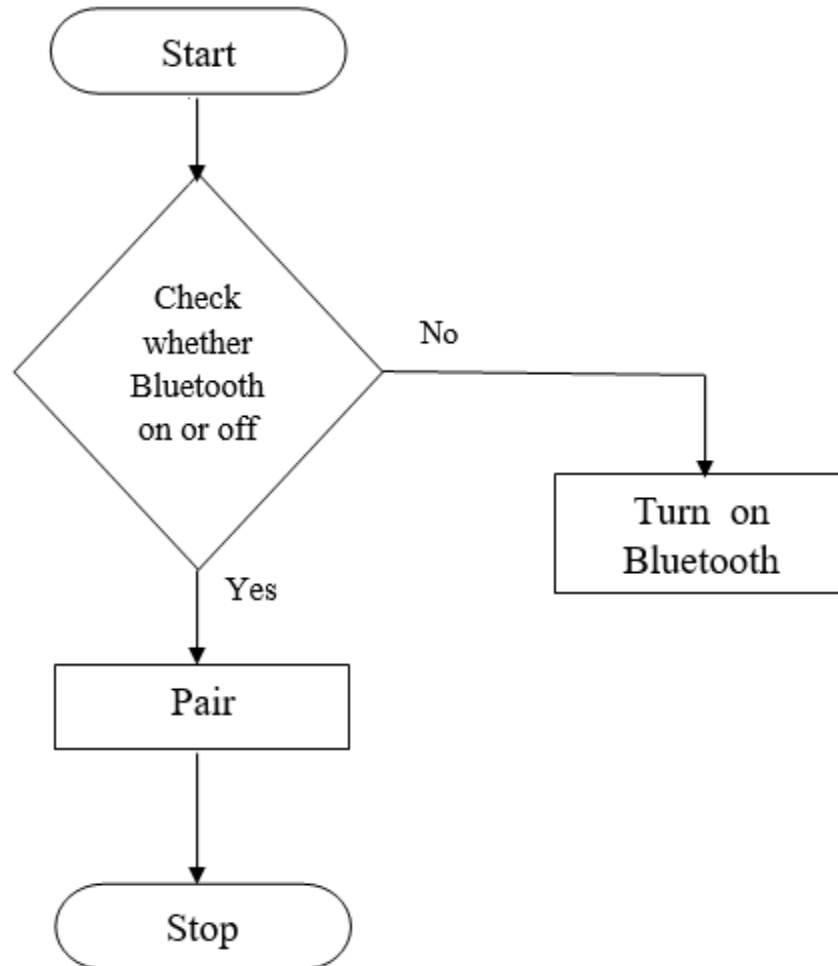
This “Remote Bluetooth” application has five modules

They are:-

1. Enable Bluetooth adopter
2. Paring with required device
3. Service Setup between PC and Mobile
4. Bluetooth adopter in Pc
5. Controlling Pc

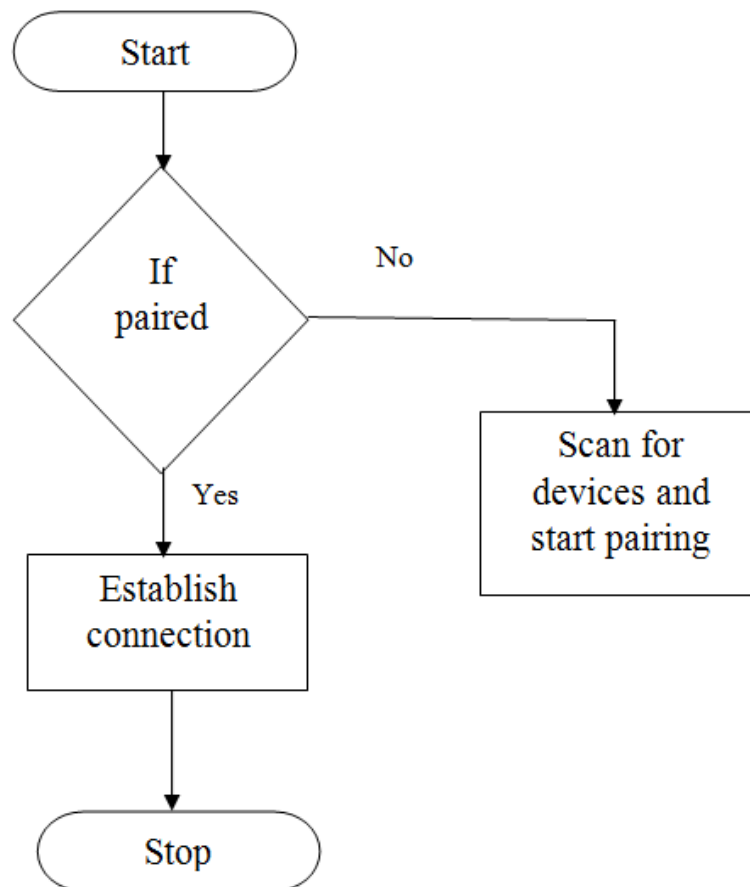
MODULES

1. Enable Bluetooth adopter



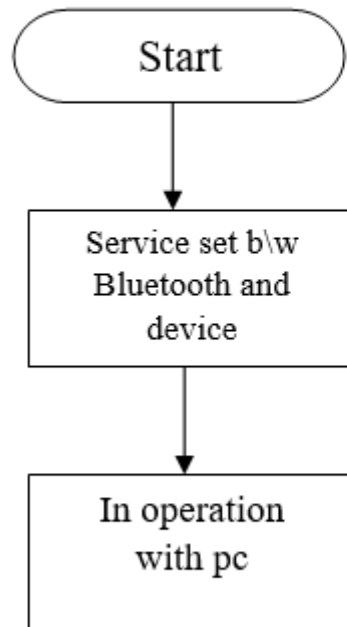
It is first stage and main part of the project. In this module it must check whether the Bluetooth is available or not if available then turn on. Here the Bluetooth plays important role after the turn on of the Bluetooth in the device next connect the device to the pc. for connecting pc scan it in the device

2.Paring with required device



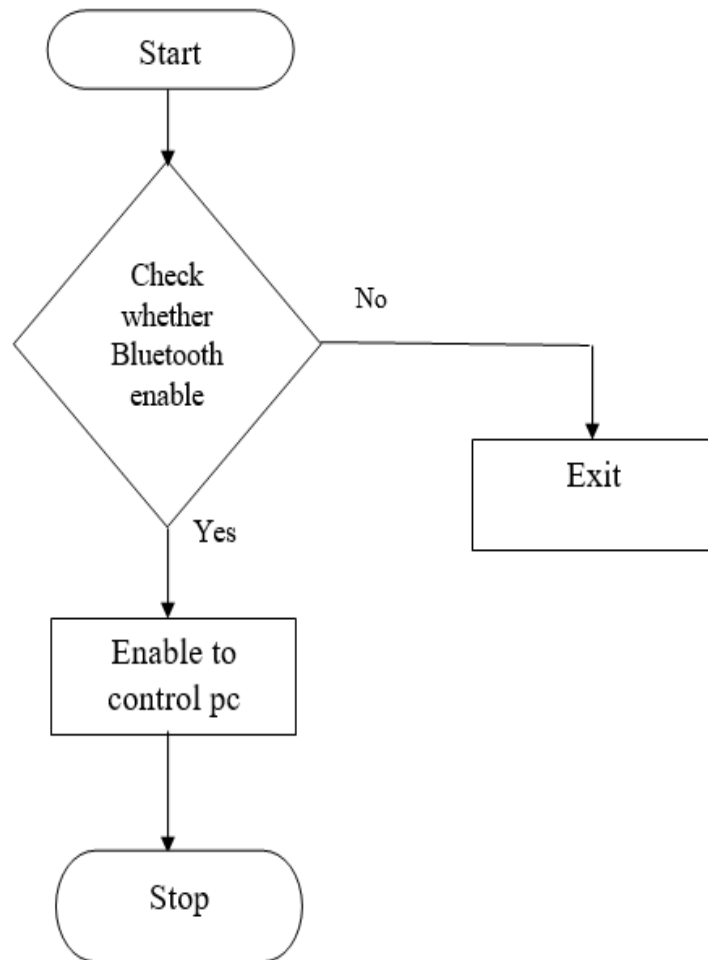
In the pc check any request of the Bluetooth then accept .And if not come check in already paired list. or click on the scan then if the required device is found then select it. And make pairing after the pairing done then the connection is made successfully.

3. Service Setup between PC and Mobile



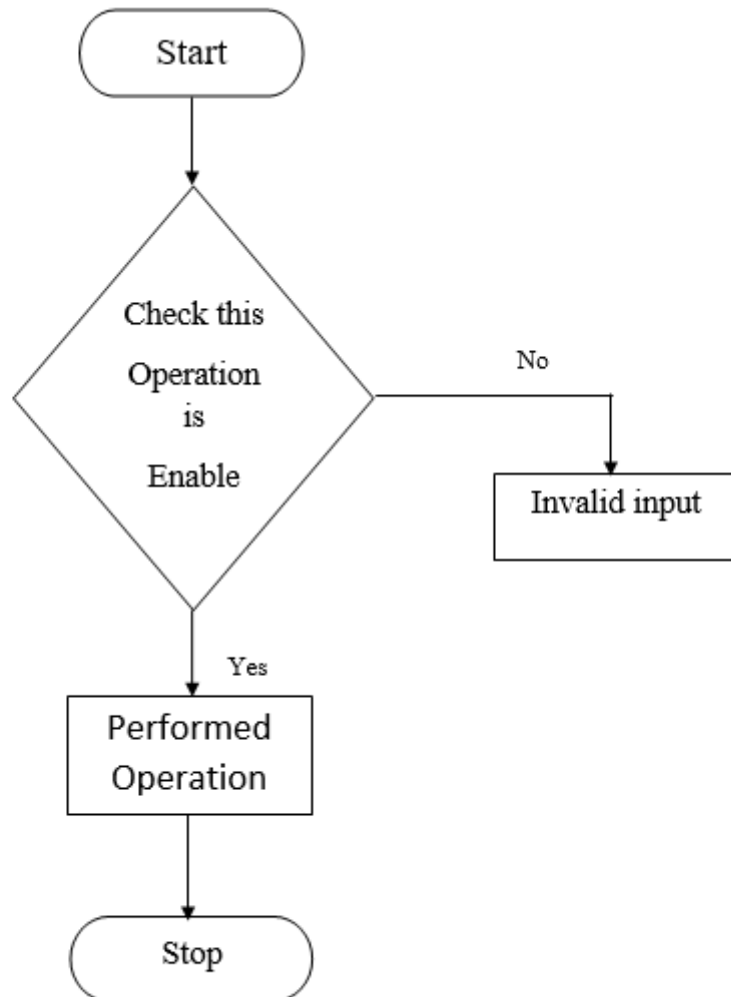
After successful connection it will build a setup called as service setup .this is between the device and pc. This setup for the data flow for receiving data sending data

4. Bluetooth adopter in Pc



It is first stage of the project it must check whether the Bluetooth is enable or not if available then turn on. Here the Bluetooth plays important role after the turn on of the Bluetooth in the device next connect the phone to pc. But the Bluetooth adapter is not available exit the project. And Pc give the notification is Bluetooth adapter is not available.

5. Controlling Pc



When the connection is establish give the input the input is in numeric format. The particular numeric value have a particular operations. When give inputs the Pc is search the input is enable or not if the input is enable performed particular operation. But input is wrong or not enable pc give the notification is the operation is not included in app.

CHAPTER 7

IMPLEMENTATION

7.1 Setting up an Android Studio Development Environment

Before any work can begin on the development of an Android application, the first step is to configure a computer system to act as the development platform. This involves a number of steps consisting of installing the Java Development Kit (JDK) and the Android Studio Integrated Development Environment (IDE) which also includes the Android Software Development Kit (SDK).

*** System Requirements**

Android application development may be performed on any of the following system types:

- Windows 7/8/10 (32-bit or 64-bit)
- Mac OS X 10.8.5 or later (Intel based systems only)
- Linux systems with version 2.11 or later of GNU C Library
- Minimum of 2GB of RAM (8GB is preferred)
- Approximately 4GB of available disk space
- 1280 x 800 minimum screen resolution

Installing the Java Development Kit (JDK)

The Android SDK was developed using the Java programming language. Similarly, Android applications are also developed using Java. As a result, the Java Development Kit (JDK) is the first component that must be installed.

Android Studio 2 development requires the installation of version 8 of the Standard Edition of the Java Platform Development Kit. Java is provided in both development (JDK) and runtime (JRE) packages. For the purposes of Android development, the JDK must be installed.

Windows JDK Installation

Assuming that a suitable JDK is not already installed on your system, download version 8 of the JDK package that matches the destination computer system. Once downloaded, launch the installation executable and follow the on screen instructions to complete the installation process.

Downloading the Android Studio Package

Most of the work involved in developing applications for Android will be performed using the Android Studio environment. The content and examples in this book were created based on Android Studio version 2.2.

At the time of writing, both Android Studio 2.2 and the Android 7 SDK are available in preview versions only. The location for downloading the Android Studio package will depend on whether or not the software is still in preview. Begin by checking the primary download page for Android Studio which can be found at the following URL:

<https://developer.android.com/studio/index.html>

If this page provides instructions for downloading Android Studio 2.2, perform the download from this page. If, on the other hand, the page provides access to Android Studio 2.1, you will need to download the latest preview edition of Android Studio 2.2 from the following web page:

<http://tools.android.com/download/studio/canary/latest>

From the appropriate page, select the package for your platform and operating system.

Installing Android Studio

Once downloaded, the exact steps to install Android Studio differ depending on the operating system on which the installation is being performed.

Installation on Windows

Locate the downloaded Android Studio installation executable file (named android-studio-bundle<version>.exe) in a Windows Explorer window and double click on it to start the installation process, clicking the Yes button in the User Account Control dialog if it appears.

Once the Android Studio setup wizard appears, work through the various screens to configure the installation to meet your requirements in terms of the file system location into which Android Studio should be installed and whether or not it should be made available to other users of the system. When prompted to select the components to install, make sure that the Android Studio, Android SDK and Android Virtual Device options are all selected.

Although there are no strict rules on where Android Studio should be installed on the system, the remainder of this book will assume that the installation was performed into C:\Program Files\Android\Android Studio and that the Android SDK packages have been installed into the user's AppData\Local\Android\sdk sub-folder. Once the options have been configured, click on the Install button to begin the installation process.

On versions of Windows with a Start menu, the newly installed Android Studio can be launched from the entry added to that menu during the installation. The executable may be pinned to the task bar for easy access by navigating to the Android Studio\bin directory, right-clicking on the executable and selecting the Pin to Taskbar menu option. Note that the executable is provided in 32-bit (studio) and 64-bit (studio64) executable versions. If you are running a 32-bit system be sure to use the studio executable.

Installing Additional Android SDK Packages

The steps performed so far have installed Java, the Android Studio IDE and the current set of default Android SDK packages. Before proceeding, it is worth taking some time to verify which packages are installed and to install any missing or updated packages.

This task can be performed using the Android SDK Settings screen, which may be launched from within the Android Studio tool by selecting the Configure -> SDK Manager option from within the Android Studio welcome dialog. Once invoked, the Android SDK screen of the default settings dialog will appear

Immediately after installing Android Studio for the first time it is likely that only the latest released version of the Android SDK has been installed. To install preview or older versions of the Android SDK simply select the checkboxes corresponding to the versions and click on the Apply button.

It is also possible that updates will be listed as being available for the latest SDK. To access detailed information about the packages that are available for update, enable the Show Package Details option located in the lower right hand corner of the screen. This will display information similar to that shown

Remote Bluetooth

To install the updates, enable the checkbox to the left of the item name and click on the Apply button.

In addition to the Android SDK packages, a number of tools are also installed for building Android applications. To view the currently installed packages and check for updates, remain within the SDK settings screen and select the SDK Tools tab as shown in

Within the Android SDK Tools screen, make sure that the following packages are listed as Installed in the Status column:

- Android SDK Build-tools
- Android SDK Tools
- Android SDK Platform-tools
- Android Support Repository
- Android Support Library
- Google Repository
- Google USB Driver (Windows only)
- Intel x86 Emulator Accelerator (HAXM installer)

In the event that any of the above packages are listed as Not Installed or requiring an update, simply select the checkboxes next to those packages and click on the Apply button to initiate the installation process.

Once the installation is complete, review the package list and make sure that the selected packages are now listed as Installed in the Status column. If any are listed as Not installed, make sure they are selected and click on the Install packages... button again.

An alternative to using the Android SDK settings panel is to access the Standalone SDK Manager which can be launched using the link in the lower left hand corner of the settings screen. The Standalone SDK Manager provides a similar list of packages together with options to perform update and installation tasks:

Making the Android SDK Tools Command-line Accessible

Most of the time, the underlying tools of the Android SDK will be accessed from within the Android Studio environment. That being said, however, there will also be instances where it will be useful to be able to invoke those tools from a command prompt or terminal window. In order for the operating system on which you are developing to be able to find these tools, it will be necessary to add them to the system's PATH environment variable.

Regardless of operating system, the PATH variable needs to be configured to include the following paths (where <path_to_android_sdk_installation> represents the file system location into which the Android SDK was installed):

<path_to_android_sdk_installation>/sdk/tools

<path_to_android_sdk_installation>/sdk/platform-tools

The location of the SDK on your system can be identified by launching the Standalone SDK Manager and referring to the Android SDK Location: field located at the top of the settings panel as highlighted in Once the location of the SDK has been identified, the steps to add this to the PATH variable are operating system dependent:

Windows 7

1. Right-click on Computer in the desktop start menu and select Properties from the resulting menu.
2. In the properties panel, select the Advanced System Settings link and, in the resulting dialog, click on the Environment Variables... button.
3. In the Environment Variables dialog, locate the Path variable in the System variables list, select it and click on Edit Locate the end of the current variable value string and append the path to the Android platform tools to the end, using a semicolon to separate the path from the preceding values. For example, assuming the Android SDK was installed into

C:\Users\demo\AppData\Local\Android\sdk, the following would be appended to the end of the current Path value:

;C:\Users\demo\AppData\Local\Android\sdk\platformtools;

C:\Users\demo\AppData\Local\Android\sdk\tools

4. Click on OK in each dialog box and close the system properties control panel.

Remote Bluetooth

Once the above steps are complete, verify that the path is correctly set by opening a Command Prompt window (Start -> All Programs -> Accessories -> Command Prompt) and at the prompt enter:

```
echo %Path%
```

The returned path variable value should include the paths to the Android SDK platform tools folders. Verify that the platform-tools value is correct by attempting to run the adb tool as follows:

```
adb
```

The tool should output a list of command line options when executed.

Similarly, check the tools path setting by attempting to launch the Android SDK Manager:

```
android
```

In the event that a message similar to the following message appears for one or both of the commands, it is most likely that an incorrect path was appended to the Path environment variable:

'adb' is not recognized as an internal or external command, operable program or batch file.

Windows 8.1

1. On the start screen, move the mouse to the bottom right hand corner of the screen and select Search from the resulting menu. In the search box, enter Control Panel. When the Control Panel icon appears in the results area, click on it to launch the tool on the desktop.
2. Within the Control Panel, use the Category menu to change the display to Large Icons. From the list of icons select the one labeled System.
3. Follow the steps outlined for Windows 7 starting from step 2 through to step
4. Open the command prompt window (move the mouse to the bottom right hand corner of the screen, select the Search option and enter cmd into the search box). Select Command Prompt from the search results.

Within the Command Prompt window, enter: echo %Path%

The returned path variable value should include the paths to the Android SDK platform tools folders. Verify that the platform-tools value is correct by attempting to run the adb tool as follows:

```
adb
```

Remote Bluetooth

The tool should output a list of command line options when executed.

Similarly, check the tools path setting by attempting to launch the Android SDK Manager:

android

In the event that a message similar to the following message appears for one or both of the commands, it is most likely that an incorrect path was appended to the Path environment variable:

'adb' is not recognized as an internal or external command, operable program or batch file.

Windows 10

Right-click on the Start menu, select System from the resulting menu and click on the Advanced system settings option in the System window. Follow the steps outlined for Windows 7 starting from step 2 through to step 4.

Updating the Android Studio and the SDK

From time to time new versions of Android Studio and the Android SDK are released. New versions of the SDK are installed using the Android SDK Manager. Android Studio will typically notify you when an update is ready to be installed.

To manually check for Android Studio updates, click on the Configure -> Check for Updates menu option within the Android Studio welcome screen, or use the Help -> Check for Update menu option accessible from within the Android Studio main window.

7.2 CODEING

7.2.1 ANDROID CODE

LAYOUTS

FIRST PAGE.XML

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/unnamed" >
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="BLUTOOTH CONNECT"
        android:id="@+id/connect_button"
        android:layout_centerVertical="true"
        android:layout_centerHorizontal="true"
        style="@android:style/Widget.Material.Light.Button.Inset" />
</RelativeLayout>
```

MAIN.XML

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:id="@+id/parentLayout"
    android:background="@drawable/ram" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceMedium"
        android:text="Left Text"
        android:id="@+id/title_left_text"
        android:visibility="gone"
        android:layout_centerInParent="true" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceMedium"
        android:text="Left Text"
        android:id="@+id/title_left_right"
        android:visibility="gone"
        android:layout_centerInParent="true" />
```



```
<android.support.v7.widget.AppCompatImageButton
    style="?android:attr/buttonStyleSmall"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="EXIT"
    android:id="@+id/exit"
    android:layout_alignParentBottom="true"
    android:layout_centerHorizontal="true" />

<android.support.v7.widget.AppCompatImageButton
    style="?android:attr/buttonStyleSmall"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="DELETE"
    android:src="@android:drawable/ic_input_delete"
    android:id="@+id/deleteButton"
    android:layout_alignParentBottom="true"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true" />

<android.support.v7.widget.AppCompatImageButton
    style="?android:attr/buttonStyleSmall"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="CUT"
    android:contentDescription="CUT"
    android:src="@drawable/abc_ic_menu_cut_mtrl_alpha"
    android:id="@+id/cutbutton"
    android:layout_alignParentTop="true"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true" />
```

```
<android.support.v7.widget.AppCompatImageButton
    style="?android:attr/buttonStyleSmall"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="COPY"
    android:src="@drawable/abc_ic_menu_copy_mtrl_am_alpha"
    android:id="@+id/copybutton"
    android:layout_below="@+id/cutbutton"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true" />

<android.support.v7.widget.AppCompatImageButton
    style="?android:attr/buttonStyleSmall"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="PASTE"
    android:src="@drawable/abc_ic_menu_paste_mtrl_am_alpha"
    android:id="@+id/pastebutton"
    android:layout_below="@+id/copybutton"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true" />

<android.support.v7.widget.AppCompatImageButton
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/closebutton"
    android:layout_alignParentTop="true"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true"
    android:src="@android:drawable/ic_menu_close_clear_cancel" />
```

<TextView

```
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceSmall"
    android:id="@+id/voice_command_text"
    android:layout_above="@+id/pastebutton"
    android:layout_alignRight="@+id/downbutton"
    android:layout_alignEnd="@+id/downbutton" />
```

<android.support.v7.widget.AppCompatImageButton

```
    style="?android:attr/buttonStyleSmall"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="DOWN"
    android:src="@drawable/ic_expand_more_black_24dp"
    android:id="@+id/downbutton"
    android:layout_marginTop="58dp"
    android:layout_below="@+id/send_command"
    android:layout_alignLeft="@+id/upbutton"
    android:layout_alignStart="@+id/upbutton" />
```

<android.support.v7.widget.AppCompatImageButton

```
    style="?android:attr/buttonStyleSmall"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="RIGHT"
    android:src="@drawable/ic_chevron_right_black_24dp"
    android:id="@+id/rightbutton"
    android:layout_above="@+id/downbutton"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true" />
```

```
<android.support.v7.widget.AppCompatImageButton
    style="?android:attr/buttonStyleSmall"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="LEFT"
    android:src="@drawable/ic_chevron_left_black_24dp"
    android:id="@+id/leftbutton"
    android:layout_above="@+id/downbutton"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true" />
```

```
<android.support.v7.widget.AppCompatImageButton
    style="?android:attr/buttonStyleSmall"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="UP"
    android:src="@drawable/ic_expand_less_black_24dp"
    android:id="@+id/upbutton"
    android:layout_below="@+id/voice_command_text"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="17dp" />
```

```
<android.support.v7.widget.AppCompatImageButton
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/send_command"
    android:text="ENTER"
    android:src="@android:drawable/ic_menu_add"
    android:layout_centerVertical="true"
    android:layout_alignLeft="@+id/voice_recognise"
    android:layout_alignStart="@+id/voice_recognise" />
```

```
<android.support.v7.widget.AppCompatImageButton
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/voice_recognise"
    android:src="@android:drawable/ic_btn_speak_now"
    android:elevation="0dp"
    android:layout_alignParentTop="true"
    android:layout_alignLeft="@+id/upbutton"
    android:layout_alignStart="@+id/upbutton" />
</RelativeLayout>
```

DEVICE NAME.XML

```
<?xml version="1.0" encoding="utf-8"?>
<!-- Copyright (C) 2009 The Android Open Source Project

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
-->
<TextView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:textSize="18sp"
    android:padding="5dp"/>
```

DEVICE_LIST.XML

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent" >
    <TextView android:id="@+id/title_paired_devices"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="@string/title_paired_devices"
        android:visibility="gone"
        android:background="#666"
        android:textColor="#fff"
        android:paddingLeft="5dp"/>
    <ListView android:id="@+id/paired_devices"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:stackFromBottom="true"
        android:layout_weight="1" />
    <TextView android:id="@+id/title_new_devices"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="@string/title_other_devices"
        android:visibility="gone"
        android:background="#666"
        android:textColor="#fff"
        android:paddingLeft="5dp" />
```

Remote Bluetooth

```
<ListView      android:id="@+id/new_devices"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:stackFromBottom="true"
    android:layout_weight="2"    />

<Button        android:id="@+id/button_scan"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="@string/button_scan"    />
</LinearLayout>
```


CONTENT MAIN.XML

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    app:layout_behavior="@string/appbar_scrolling_view_behavior"
    tools:context="sample.remotetooth.MainActivity"
    tools:showIn="@layout/activity_main">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!"/>
</RelativeLayout>
```

ACTIVITY MAIN.XML

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.design.widget.CoordinatorLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:fitsSystemWindows="true"
tools:context="sample.remotetooth.MainActivity">

    <android.support.design.widget.AppBarLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:theme="@style/AppTheme.AppBarOverlay">

        <android.support.v7.widget.Toolbar
            android:id="@+id/toolbar"
            android:layout_width="match_parent"
            android:layout_height="?attr/actionBarSize"
            android:background="?attr/colorPrimary"
            app:popupTheme="@style/AppTheme.PopupOverlay" />

    </android.support.design.widget.AppBarLayout>

    <include layout="@layout/content_main" />
```

```
<android.support.design.widget.FloatingActionButton  
    android:id="@+id/fab"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_gravity="bottom|end"  
    android:layout_margin="@dimen/fab_margin"  
    android:src="@android:drawable/ic_dialog_email"/>  
</android.support.design.widget.CoordinatorLayout>
```

MENU

OPETION MENU.XML

```
<menu xmlns:android="http://schemas.android.com/apk/res/android"
      xmlns:app="http://schemas.android.com/apk/res-auto"
      xmlns:tools="http://schemas.android.com/tools"
      tools:context="sample.remotetooth.MainActivity">
    <item android:id="@+id/scan"
          android:icon="@android:drawable/ic_menu_search"
          android:title="@string/connect"/>
    <item android:id="@+id/discoverable"
          android:icon="@android:drawable/ic_menu_mylocation"
          android:title="@string/discoverable"/>
</menu>
```

7.2.2 JAVA CODE

BLUECONNECT.JAVA PACKEGE

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package blueconnect;

/**
 *
 * @author Farhan
 */
public class BlueConnect {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
    }

}
```

PROCESSCONNECTIONTHREAD.JAVA PACKEGE

```
package blueconnect;

import java.awt.Robot;
import java.awt.event.KeyEvent;
import java.io.InputStream;

import javax.microedition.io.StreamConnection;

public class ProcessConnectionThread implements Runnable {

    private StreamConnection mConnection;

    // Constant that indicate command from devices
    private static final int EXIT_CMD = -1;
    private static final int KEY_RIGHT = 12;
    private static final int KEY_LEFT = 11;
    private static final int KEY_UP = 13;
    private static final int KEY_ENTER = 15;
    private static final int KEY_DOWN = 14;
    public static final int KEY_DELETE = 16;
    public static final int KEY_COPY = 17;
    public static final int KEY_CUT = 18;
    public static final int KEY_PASTE = 19;

    public ProcessConnectionThread(StreamConnection connection) {
        mConnection = connection;
    }

    @Override
    public void run() {
        try {
```

Remote Bluetooth

```
// prepare to receive data
InputStream inputStream = mConnection.openInputStream();

System.out.println("waiting for input");

while (true) {
    int command = inputStream.read();
    System.out.println("Data received: " + command);
    if (command == EXIT_CMD) {
        System.out.println("finish process");
        break;
    }

    processCommand(command);
}
} catch (Exception e) {
    e.printStackTrace();
}
}

/**
 * Process the command from client
 *
 * @param command the command code
 */
private void processCommand(int command) {
    try {
        Robot robot = new Robot();
        switch (command) {
            case KEY_RIGHT:
                robot.keyPress(KeyEvent.VK_RIGHT);
                System.out.println("Right");
        }
    }
}
```

```
// release the key after it is pressed. Otherwise the event just keeps getting triggered

robot.keyRelease(KeyEvent.VK_RIGHT);
break;
case KEY_LEFT:
    robot.keyPress(KeyEvent.VK_LEFT);
    System.out.println("Left");
    // release the key after it is pressed. Otherwise the event just keeps getting triggered

    robot.keyRelease(KeyEvent.VK_LEFT);
    break;
case KEY_UP:
    robot.keyPress(KeyEvent.VK_UP);
    System.out.println("UP");
    // release the key after it is pressed. Otherwise the event just keeps getting triggered

    robot.keyRelease(KeyEvent.VK_UP);
    break;
case KEY_ENTER:
    robot.keyPress(KeyEvent.VK_ENTER);
    System.out.println("Enter");
    // release the key after it is pressed. Otherwise the event just keeps getting triggered

    robot.keyRelease(KeyEvent.VK_ENTER);
    break;
case KEY_DOWN:
    robot.keyPress(KeyEvent.VK_DOWN);
    System.out.println("DOWN");
    // release the key after it is pressed. Otherwise the event just keeps getting triggered

    robot.keyRelease(KeyEvent.VK_DOWN);
    break;
case KEY_DELETE:
```



```
robot.keyPress(KeyEvent.VK_DELETE);
System.out.println("DELETE");
// release the key after it is pressed. Otherwise the event just keeps getting triggered

robot.keyRelease(KeyEvent.VK_DELETE);
break;
case KEY_COPY:
robot.keyPress(KeyEvent.VK_CONTROL);
robot.keyPress(KeyEvent.VK_C);
System.out.println("COPY");
// release the key after it is pressed. Otherwise the event just keeps getting triggered

robot.keyRelease(KeyEvent.VK_CONTROL);
robot.keyRelease(KeyEvent.VK_C);
break;
case KEY_CUT:
robot.keyPress(KeyEvent.VK_CONTROL);
robot.keyPress(KeyEvent.VK_X);
System.out.println("Cut");
// release the key after it is pressed. Otherwise the event just keeps getting triggered

robot.keyRelease(KeyEvent.VK_CONTROL);
robot.keyRelease(KeyEvent.VK_X);
break;
case KEY_PASTE:
robot.keyPress(KeyEvent.VK_CONTROL);
robot.keyPress(KeyEvent.VK_V);
System.out.println("Paste");
// release the key after it is pressed. Otherwise the event just keeps getting
triggered
robot.keyRelease(KeyEvent.VK_CONTROL);
robot.keyRelease(KeyEvent.VK_V);
break;
```

Remote Bluetooth

```
    }  
    } catch (Exception e) {  
        e.printStackTrace();  
    }  
}  
}
```

REMOTEBLUETOOTHSERVER.JAVA PACKEGE

```
package blueconnect;
```

```
public class RemoteBluetoothServer{
```

```
    public static void main(String[] args) {
```

```
        Thread waitThread = new Thread(new WaitThread());
```

```
        waitThread.start();
```

```
    }
```

```
}
```

WHITTHREAD.JAVA PACKEGE

```
package blueconnect;
```

```
import java.io.IOException;
```

```
import javax.bluetooth.BluetoothStateException;
```

```
import javax.bluetooth.DiscoveryAgent;
```

```
import javax.bluetooth.LocalDevice;
```

```
import javax.bluetooth.UUID;
```

```
import javax.microedition.io.Connector;
```

```
import javax.microedition.io.StreamConnection;
```

```
import javax.microedition.io.StreamConnectionNotifier;
```

```
public class WaitThread implements Runnable{
```

```
    /** Constructor */
```

```
    public WaitThread() {  
    }  
}
```

```
@Override
```

```
public void run() {  
    waitForConnection();  
}
```

```
/** Waiting for connection from devices */
```

```
private void waitForConnection() {  
    // retrieve the local Bluetooth device object  
    LocalDevice local = null;
```

```
    StreamConnectionNotifier notifier;  
    StreamConnection connection = null;
```

```
    // setup the server to listen for connection
```

```
    try {  
        local = LocalDevice.getLocalDevice();  
        local.setDiscoverable(DiscoveryAgent.GIAC);
```

```
        UUID uuid = new UUID("04c6093b00001000800000805f9b34fb", false);  
        System.out.println(uuid.toString());
```

Remote Bluetooth

```
String url = "btspp://localhost:" + uuid.toString() + ";name=RemoteBluetooth";
notifier = (StreamConnectionNotifier)Connector.open(url);
} catch (BluetoothStateException e) {
    System.out.println("Bluetooth is not turned on.");
    e.printStackTrace();
    return;
} catch (IOException e) {
    e.printStackTrace();
    return;
}

// waiting for connection
while(true) {
    try {
        System.out.println("waiting for connection...");
        connection = notifier.acceptAndOpen();

        Thread processThread = new Thread(new ProcessConnectionThread(connection));
        processThread.start();

        } catch (Exception e) {
            e.printStackTrace();
            return;
        }
    }
}
```

CHAPTER 8

SOFTWARE TESTING

8.1 IMPLEMENTATION PHASE

Introduction

The implementation phase is the period during which the system is used. The major activities in this phase are:

1. Complete conversion
2. Operate system
3. Evaluate system performance
4. Maintain system and manage changes

Usually, this phase is the longest of all the life cycle phases and is characterized by four distinct changes. Initially, the new system must be introduced into the business activity stream. This state is called ***changeover***. The changeover transaction may take weeks or even months. After it is completed, the system enters the operation and routine maintenance stage. Early in this stage, the evaluation should be made based on performance measurements that determine whether the specific benefits claimed for the system have been achieved. Finally, the new system, all operational systems, must be able to accommodate change. Change is perhaps the most important stage in the life of computer-based system. Whether or not change can be managed is the final measure of success or failure of the entire system effort.

The principle activities and documents that characterize the stages of the implementation phase are:

- System changeover
- Routine operation
- System performance evaluation
- System change
- System testing
- Reference manual preparation
- Personnel training

System Testing and Training:

System Testing is a process of executing a program with the explicit intention of finding errors, which cause program failure. There are two general strategies for testing software. They are:

Code Testing

Specification testing

Code Testing:

This strategy examines the logic of a program and has been carried out to identify three levels of correctness of programs. Possible correctness is first achieved by giving arbitrary inputs. Then the inputs are carefully selected to obtain predicted output. This gives the probable correctness. All potentially problematic areas are checked in this way for the software to achieve probable correctness. Absolute correctness can be demonstrated by a test involving every possible combination of inputs. However, this cannot be performed with the software but to the existence of the various possible combinations of the inputs and due to time restrictions.

Specification Testing:

The specifications are examined which states what the program should do and how it should perform under various conditions. Then test cases are developed for each condition or combinations of conditions and submitted for processing. By examining the results, it is determined whether the program performs according to its specified requirements.

Levels of Testing:

The two levels of Testing are

1. Unit Testing
2. System Testing

Unit testing is done for the programs making up the systems. It is focused to find out module errors and enables to detect errors in coding and logic that are contained in the module. Unit testing is performed from bottom-up, starting with the smallest and lowest levels modules and proceeding one.

At a time **System Testing** finds out the discrepancies between the system and its original objective, current specifications and systems documentation.

The training session consists of getting the users used to software by asking them to perform data entry in our presence and look into the problems if encountered.

Testing can be done in two ways.

1. Sample Tests
2. Real Tests

Sample Tests:

The software was tested with sample data that we randomly selected. I tested all functions with such random data and I was successful in getting accurate results. It was at this time I got to know certain intricacies of the system that I had overlooked.

Without much delay however, I got over the problems and managed to perfect the software at least to the extent possible.

Real Test:

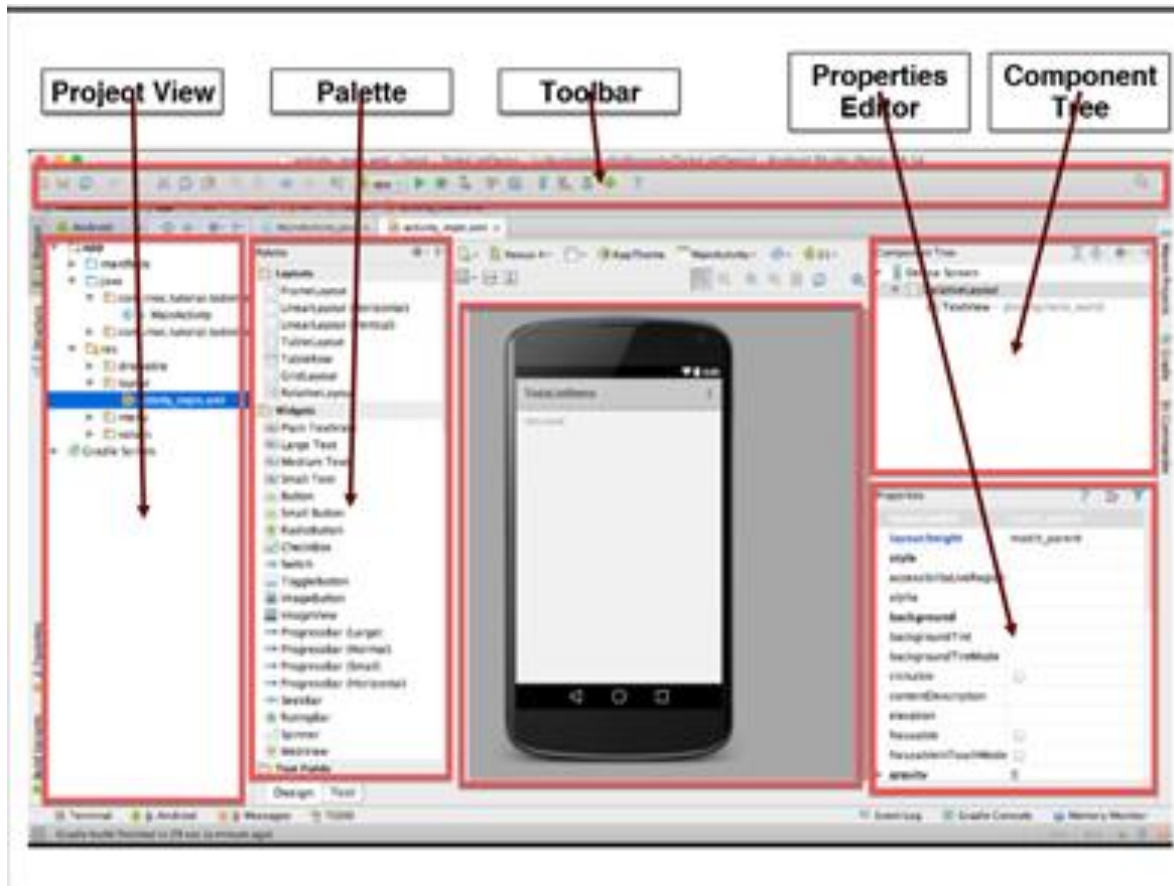
For the real test, I have planned to do in due course. I initialized the software and creation of entities through the updation module, transaction entries through the transaction module and generated reports with the estimations. The various information retrieval functions as per user need are also implemented.

CHAPTER 9

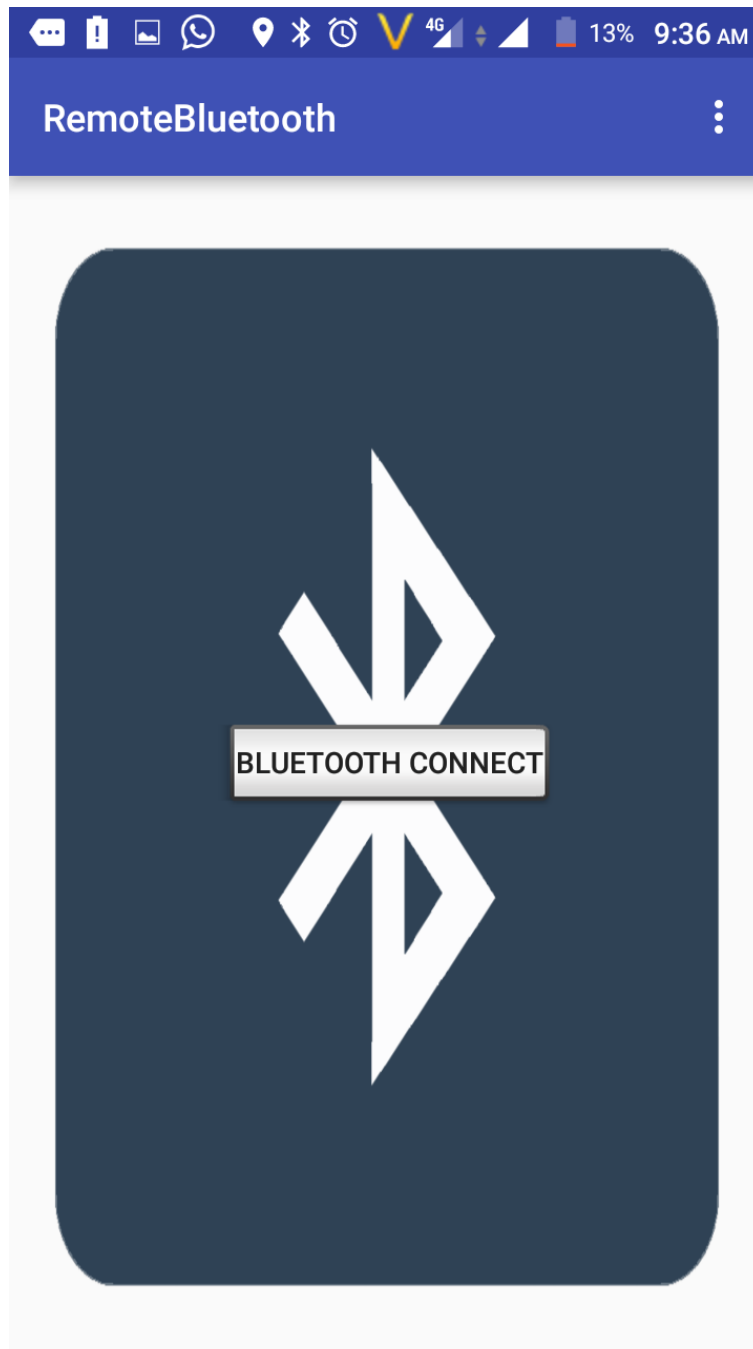
SCREEN SHOTS

9.1 Android Screen Shots

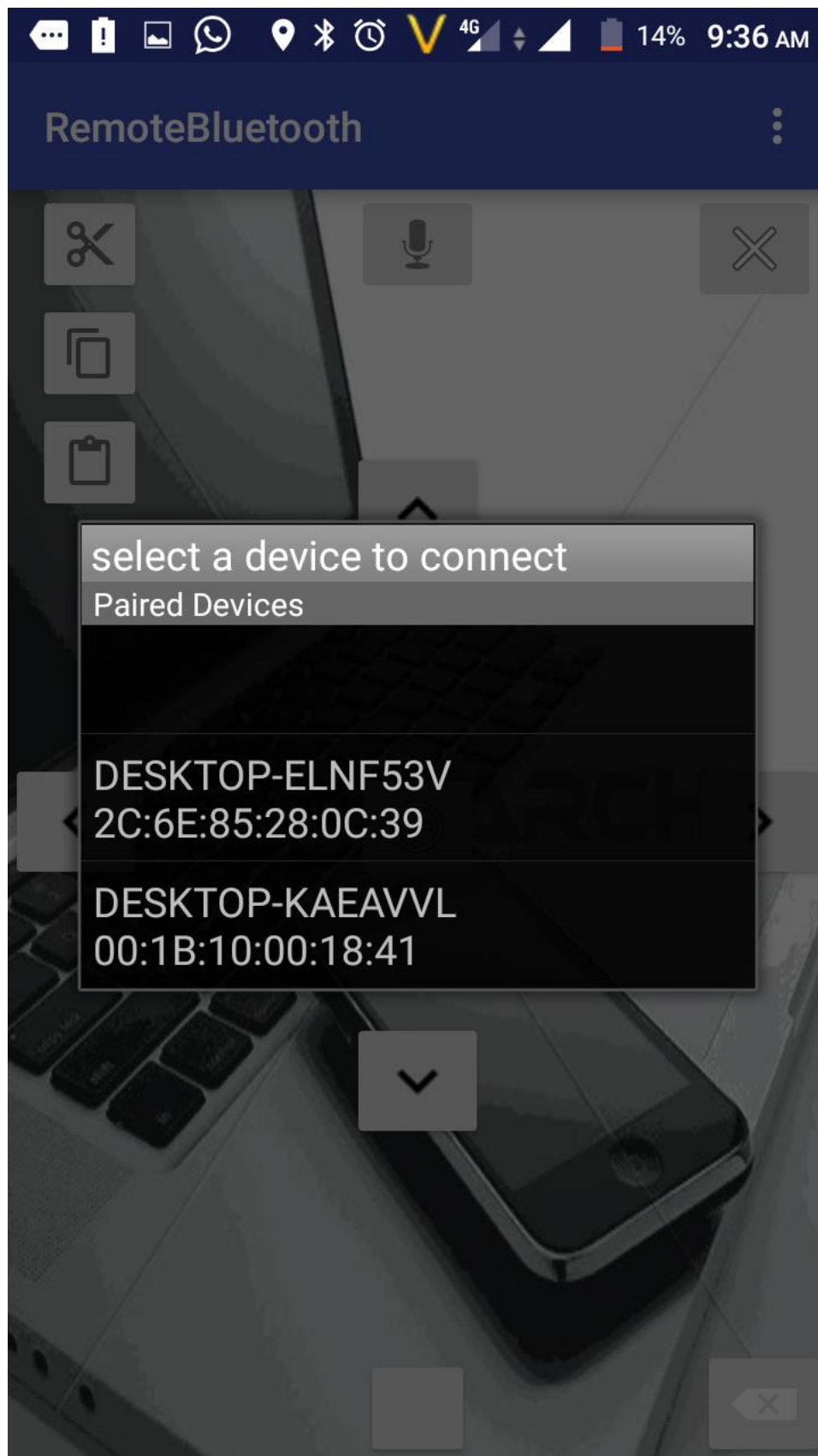
Android studio



HOME PAGE



CONNECTING



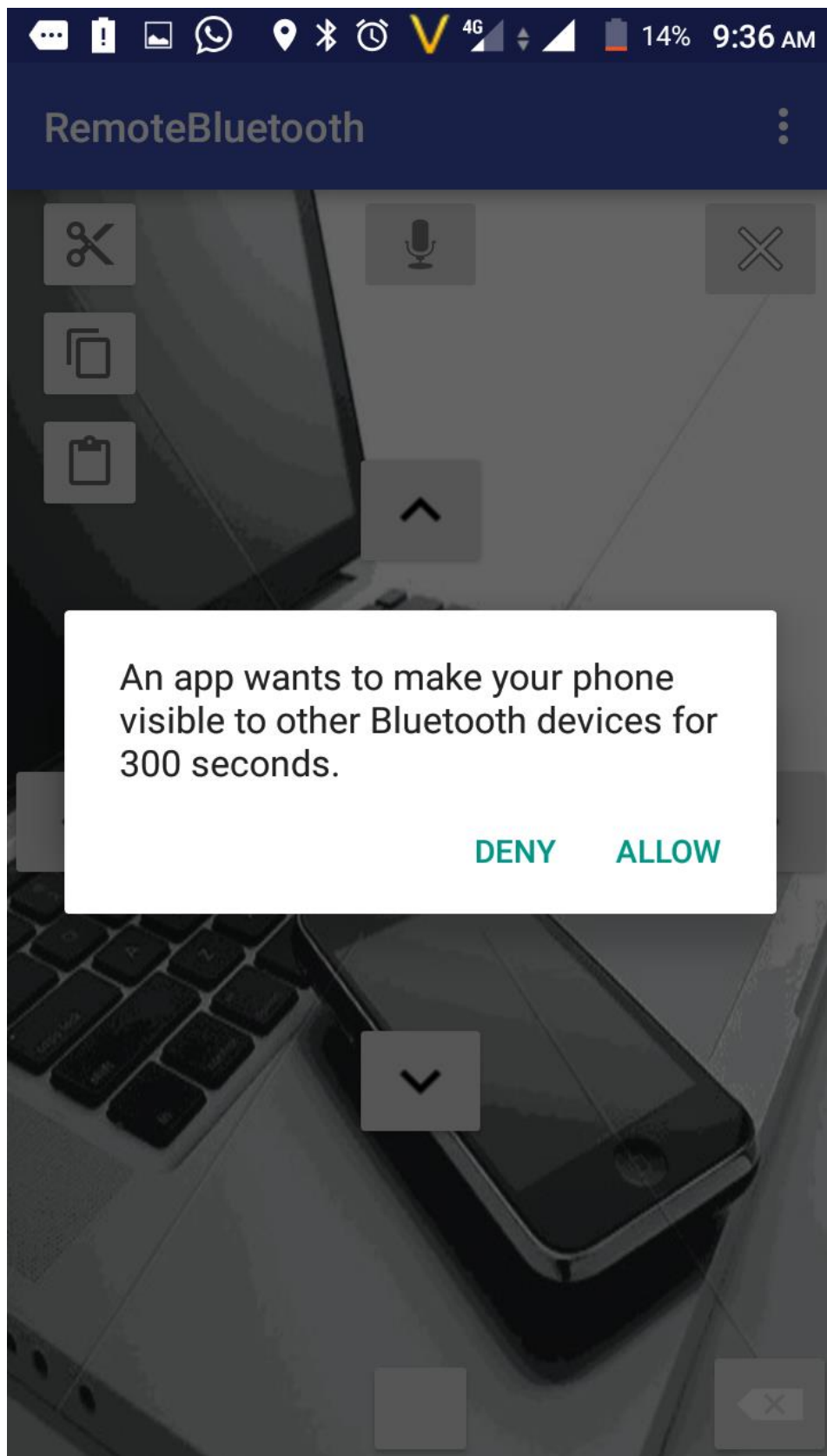
CONNECTED



OPENED MENU

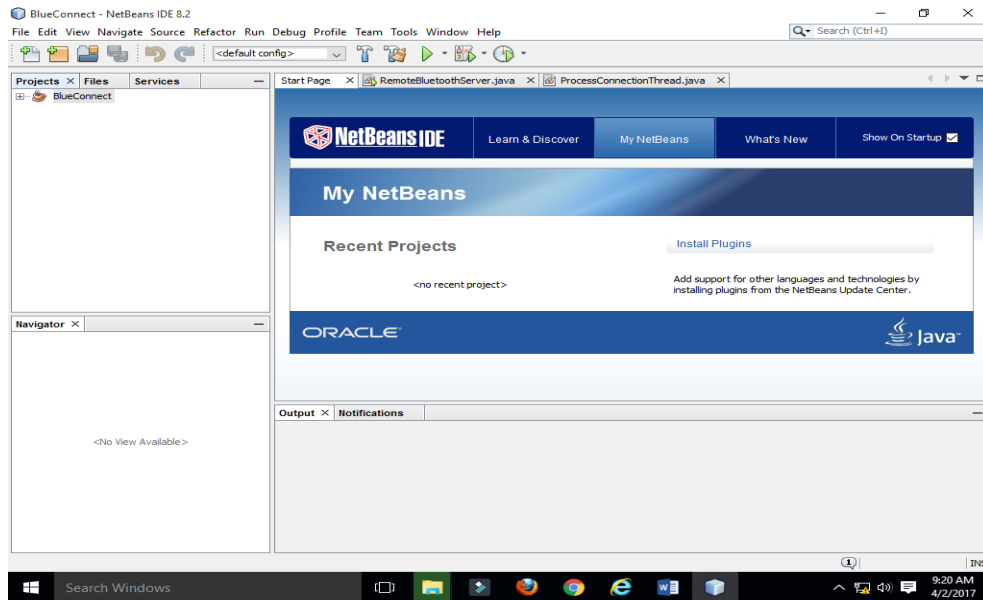


CONNECTION VISIBLE

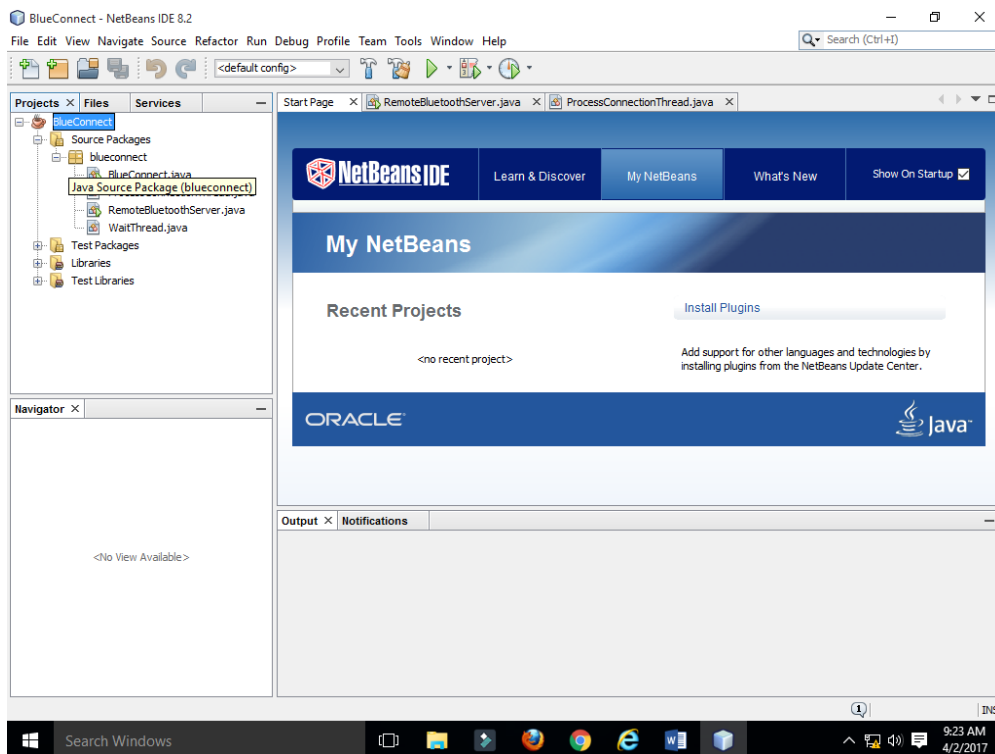


JAVA SCREENSHOTS

HOME PAGE

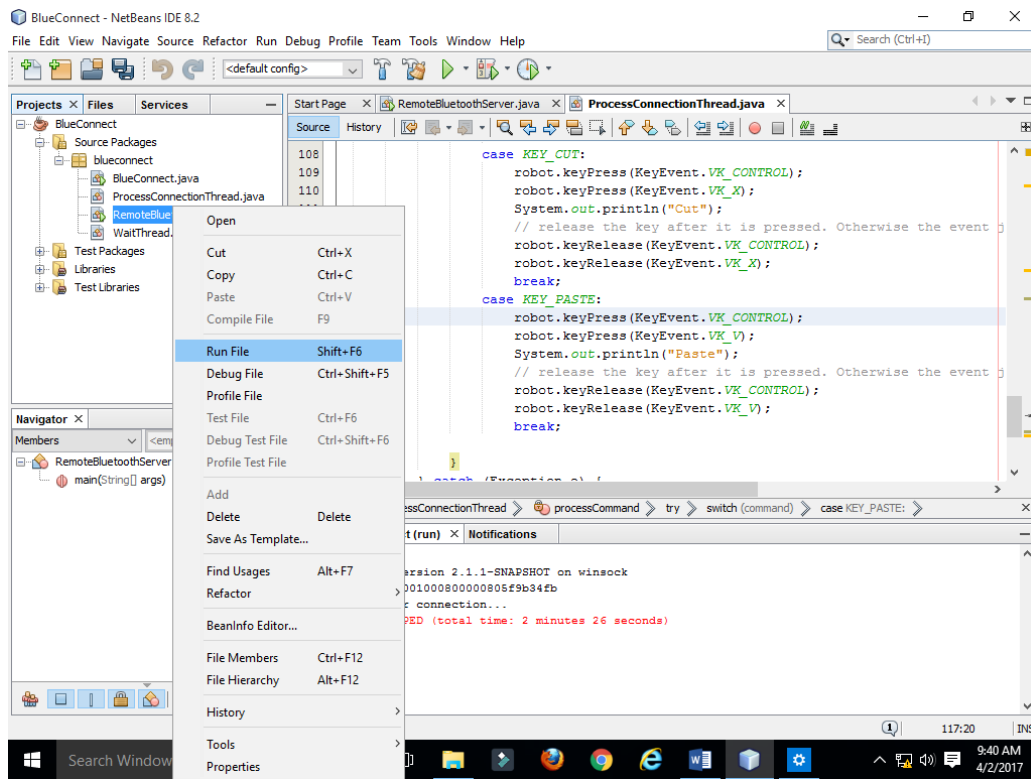


PACKAGE PAGE

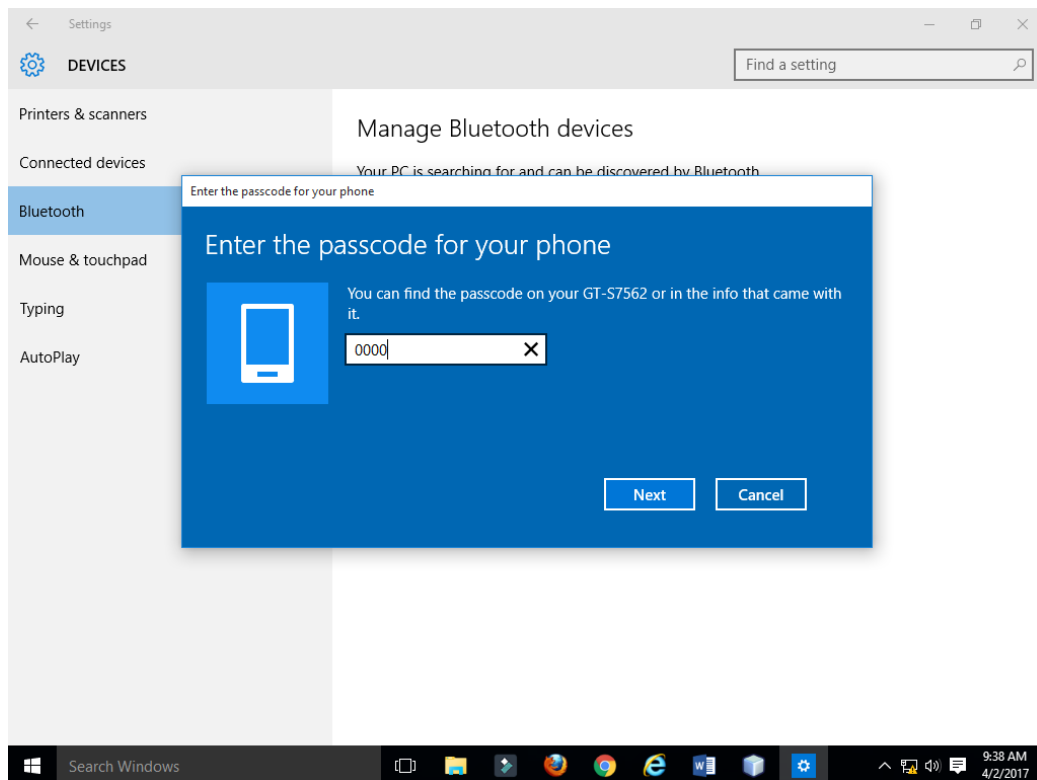


RUN PROJECT

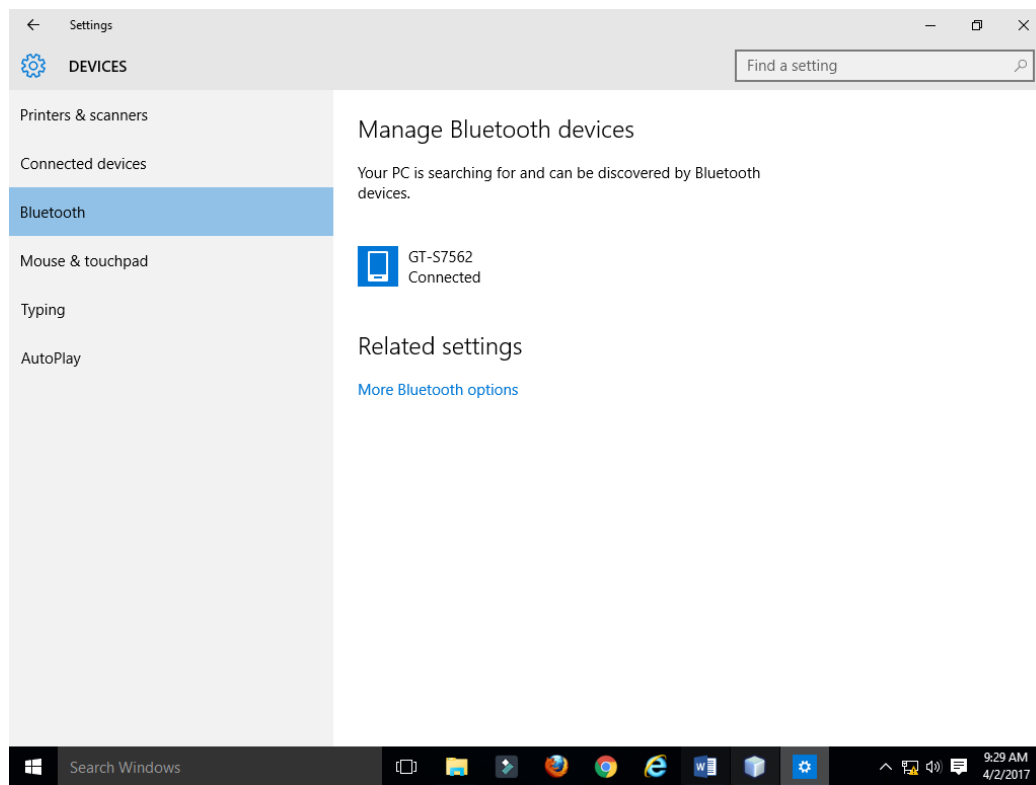
CONNECTION NOTIFICATION



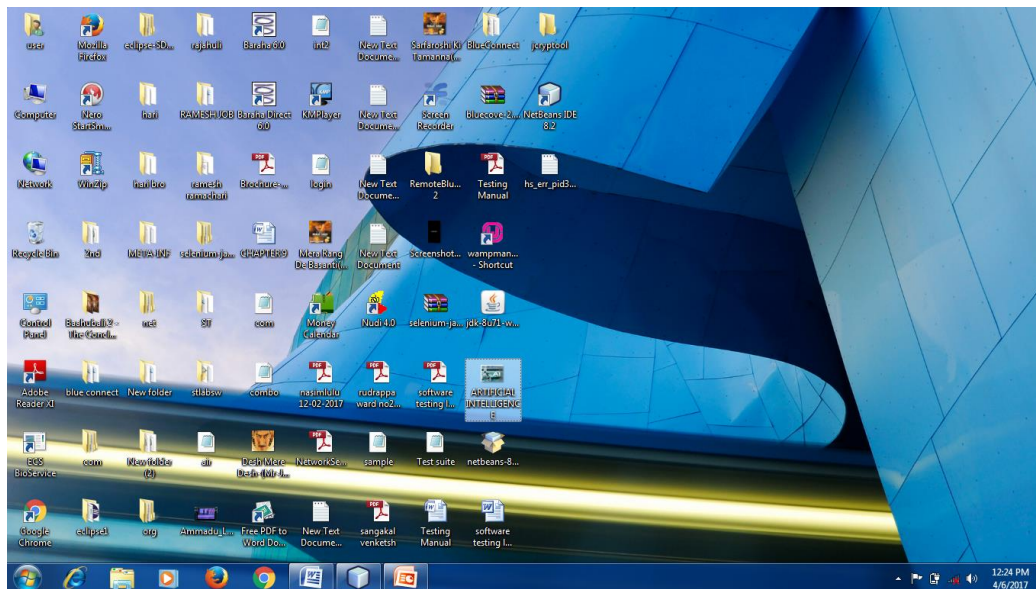
PARING DEVICE



DEVICE CONNECTED

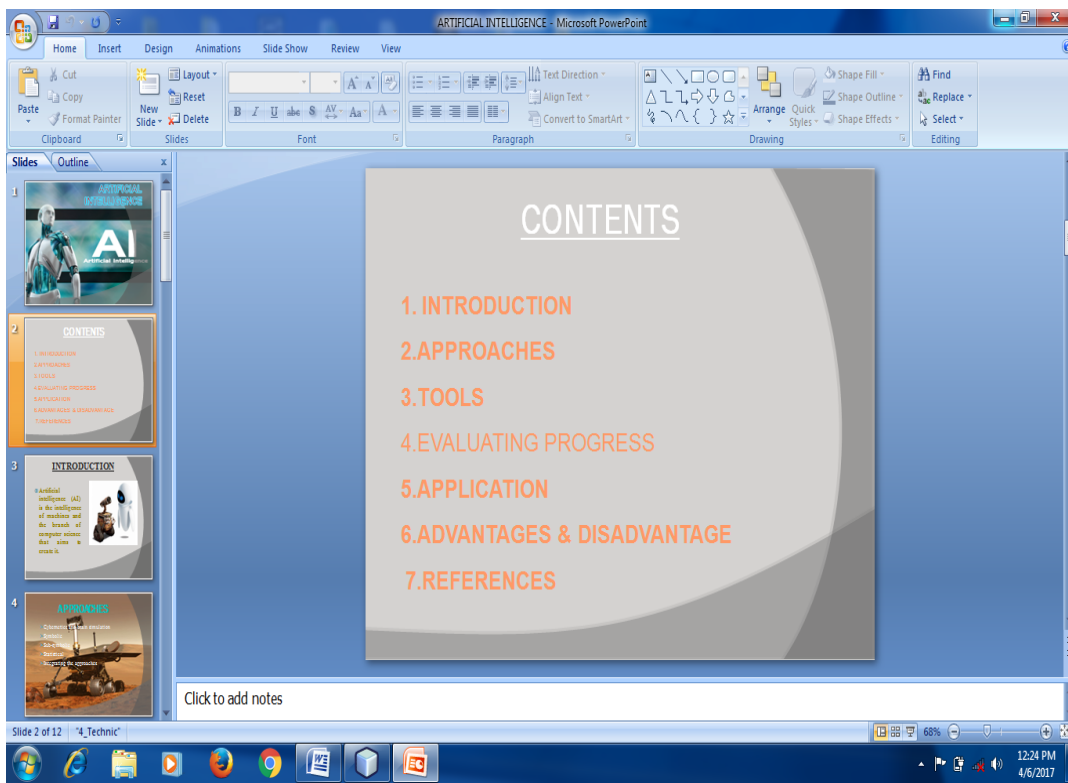
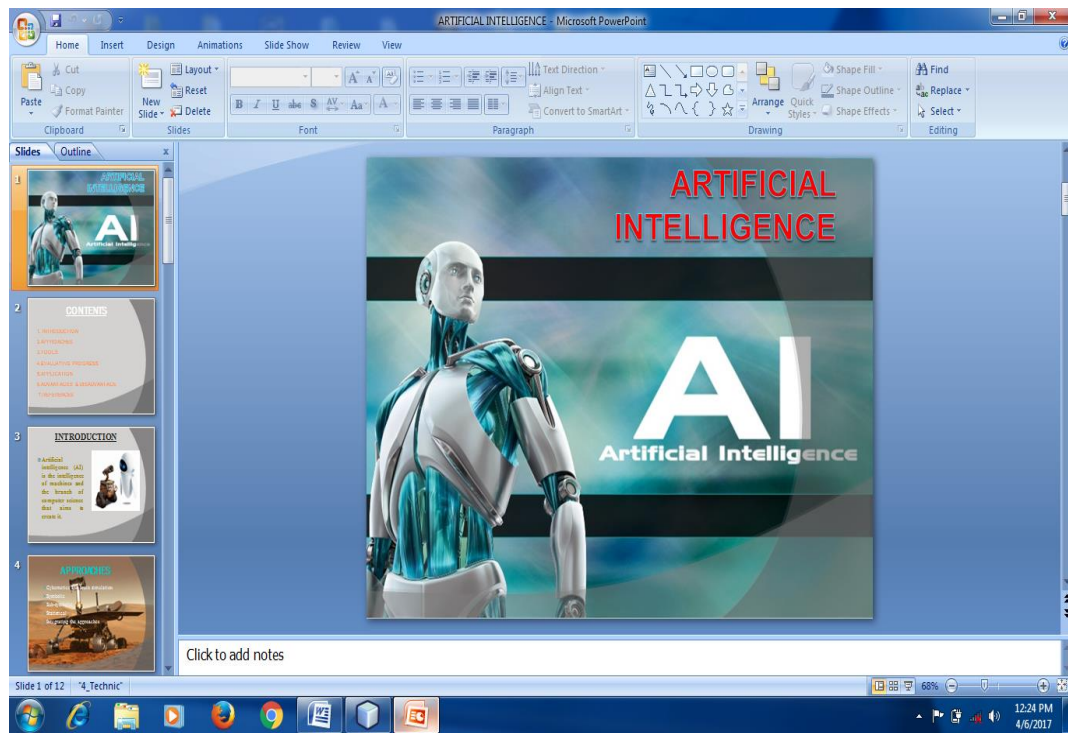


OPEN PROCESS

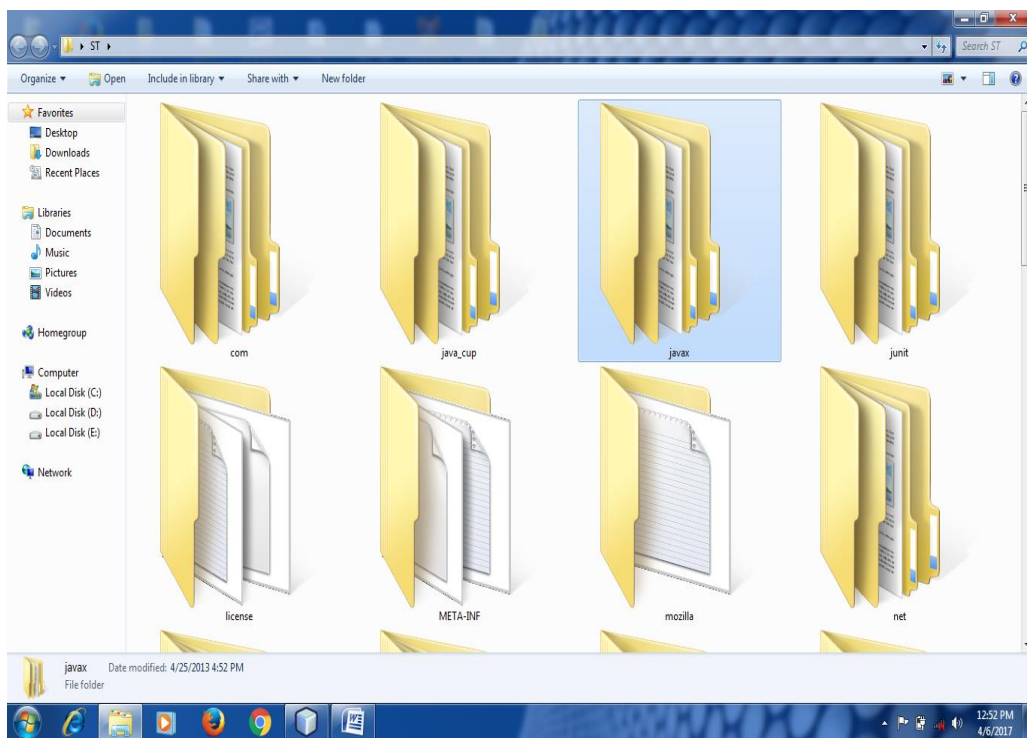
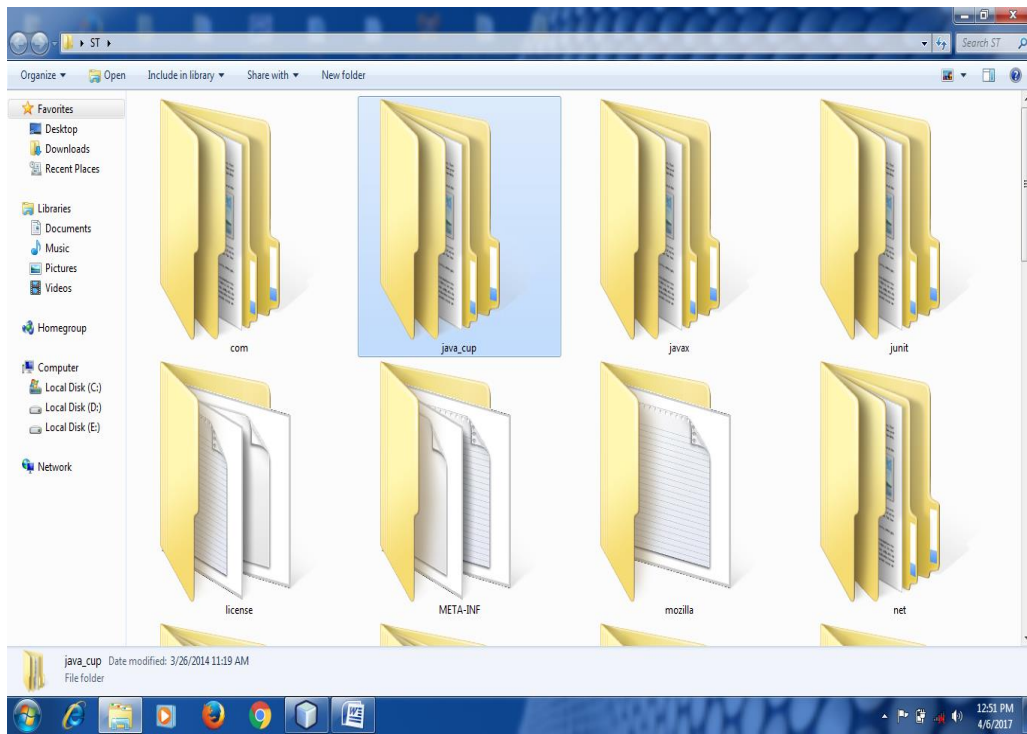


CONTROLLING PC VIA MOBILE PHONE

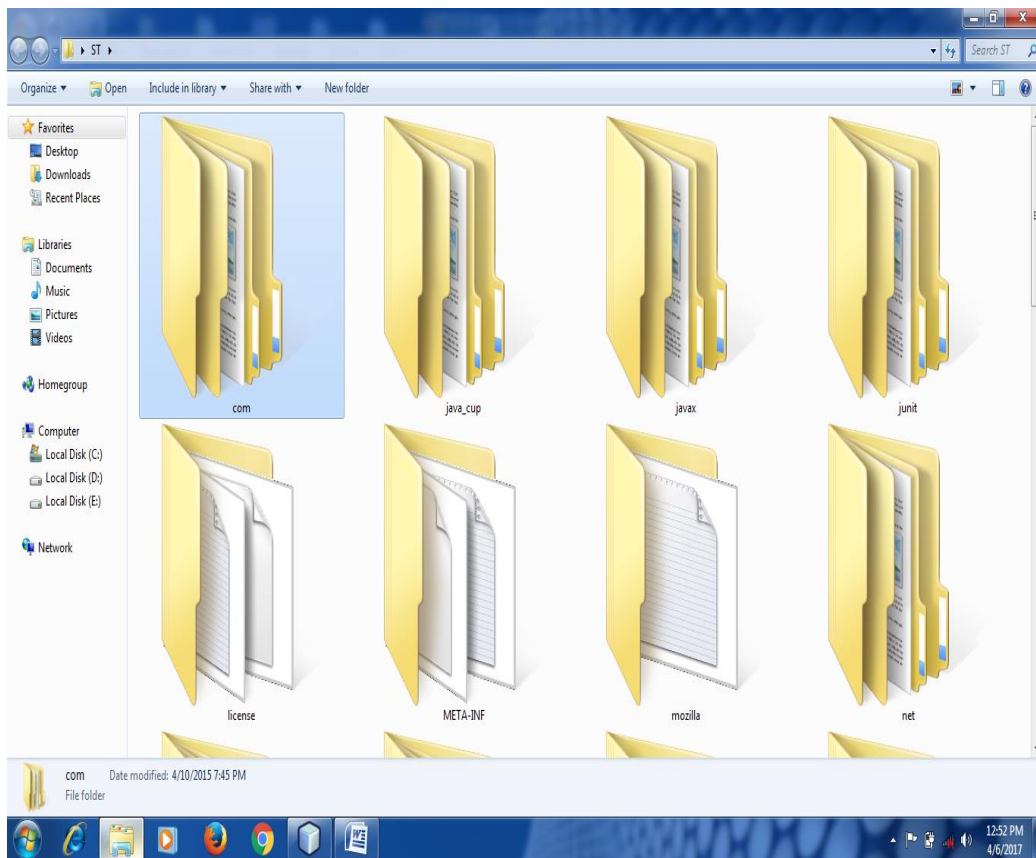
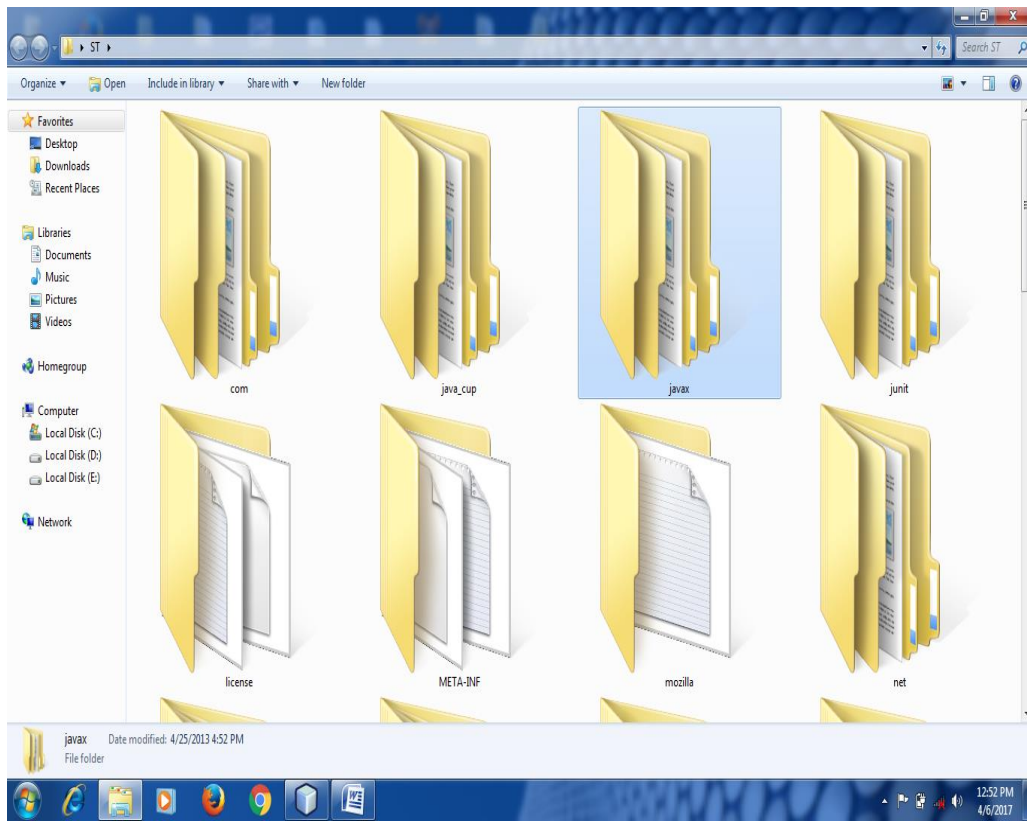
DOWN PROCESS



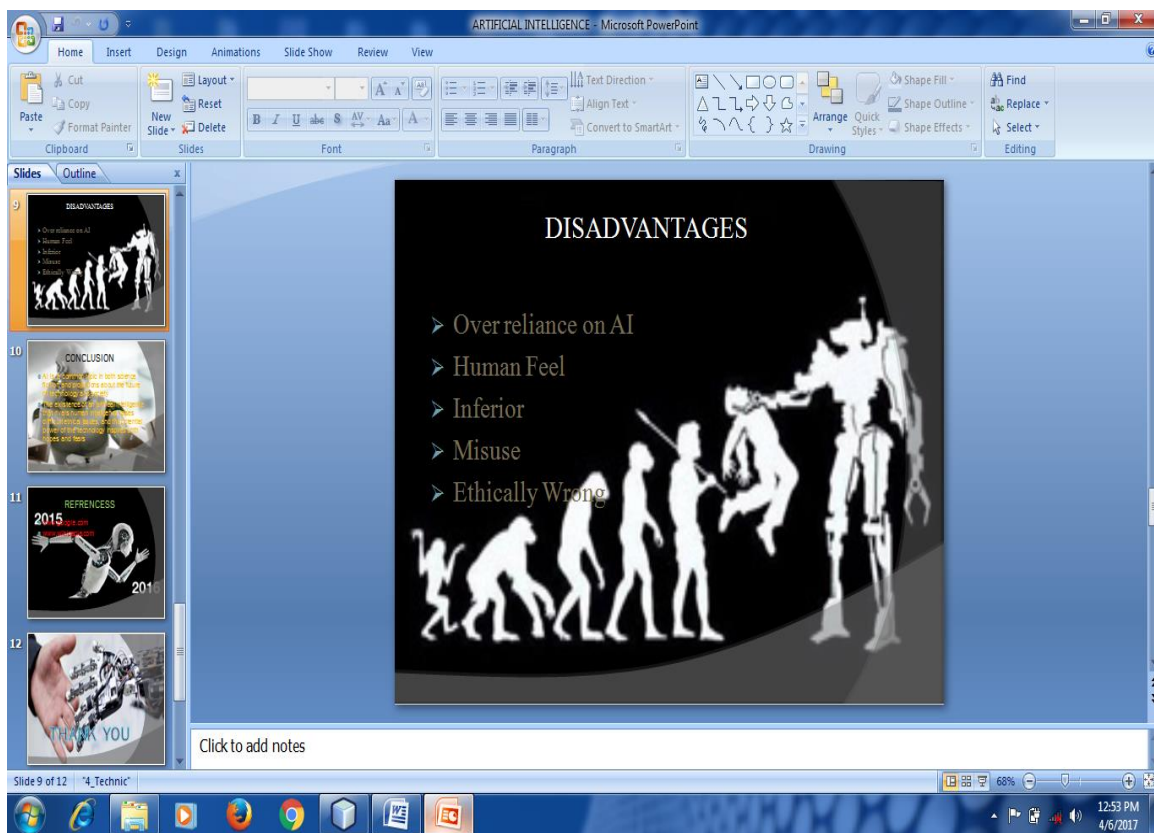
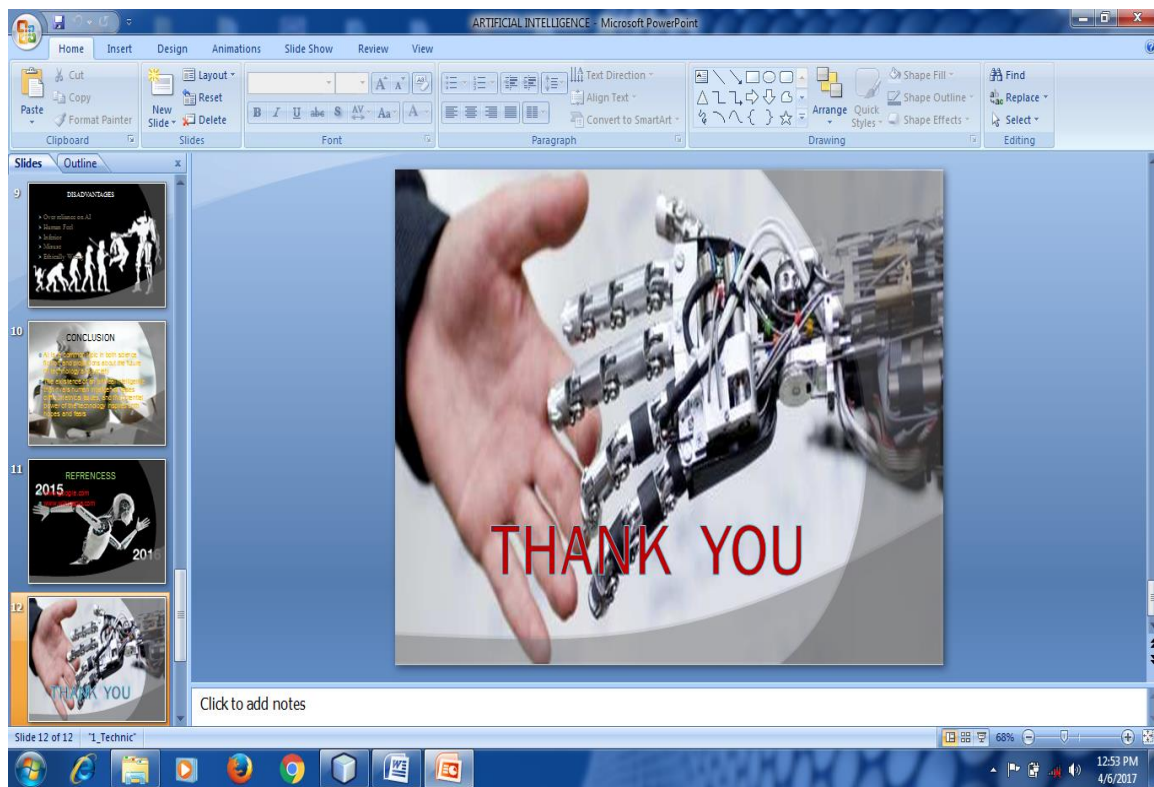
RIGHT PROCESS



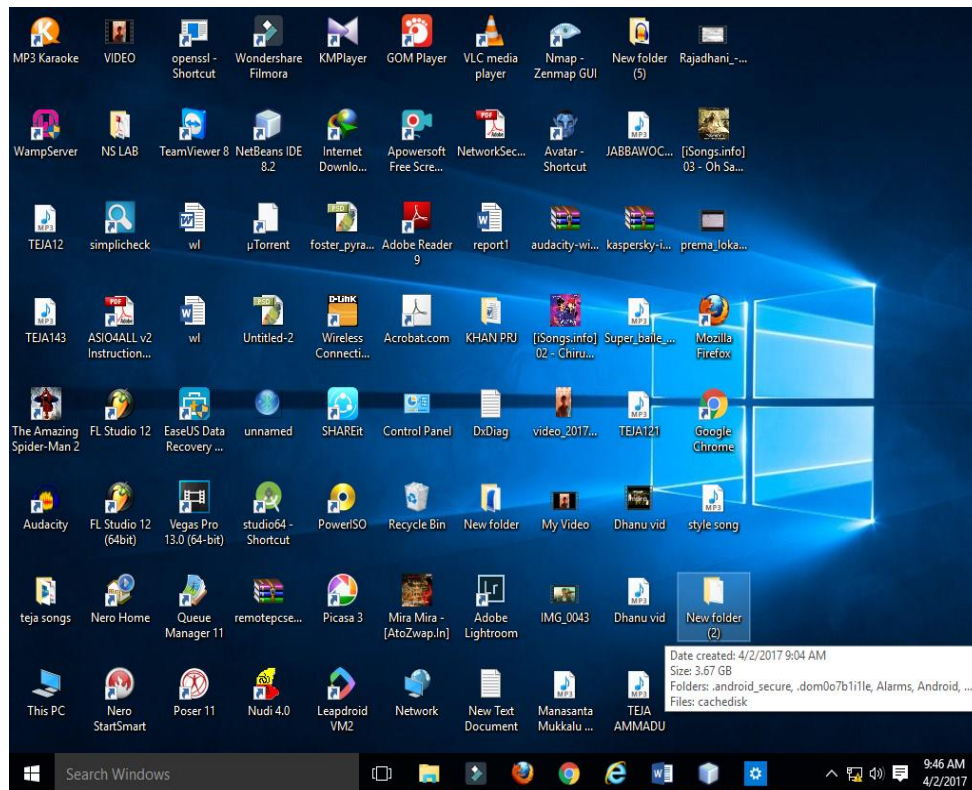
LEFT PROCESS



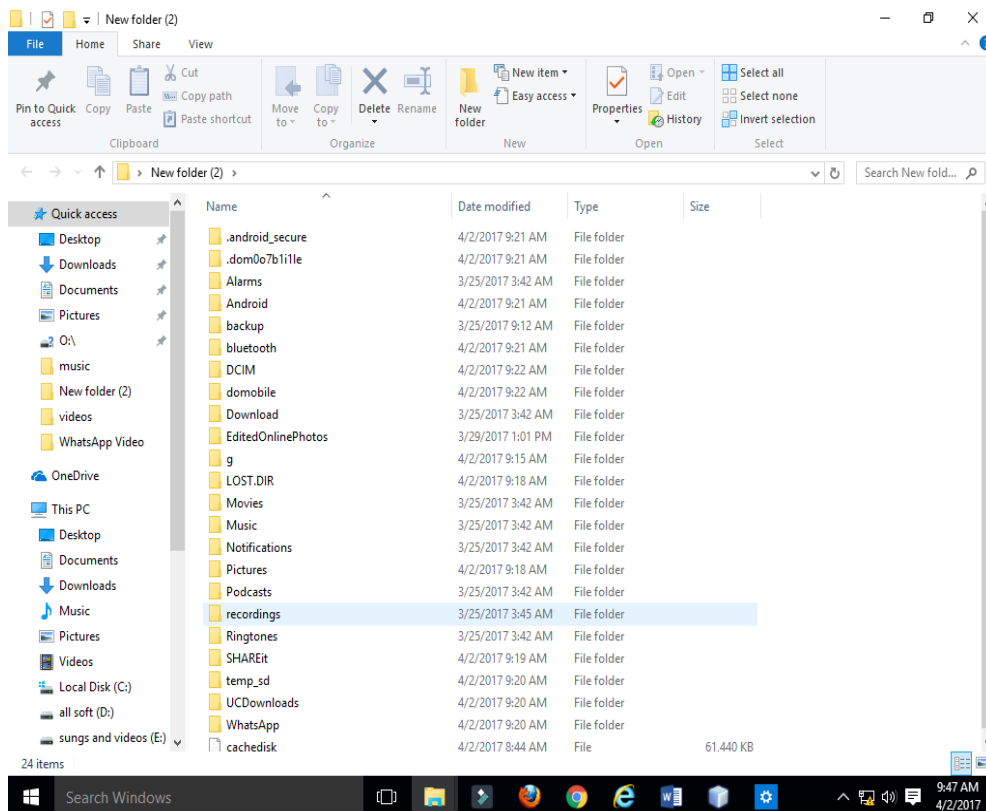
UP PROCESS



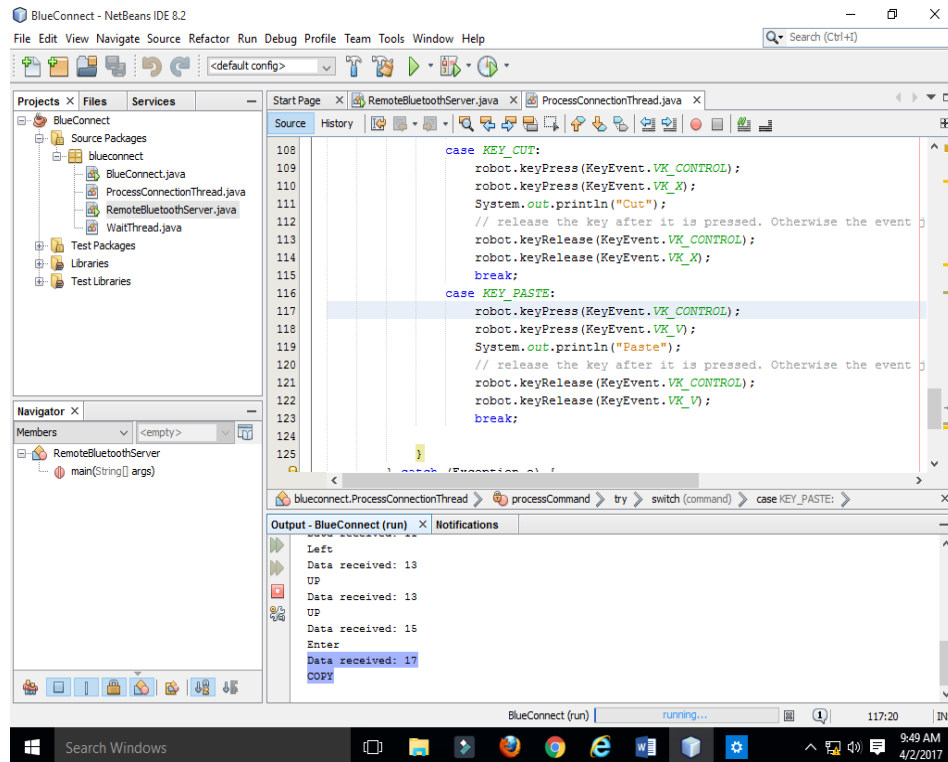
ENTER PROCESS



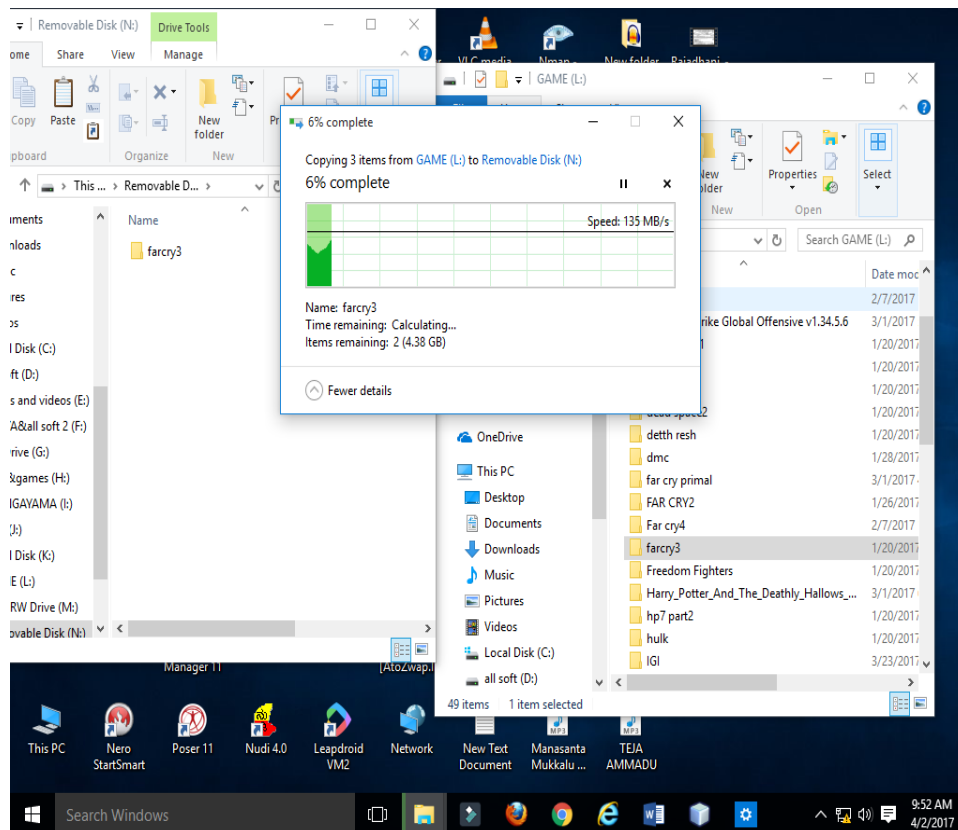
OPEN FILE PROCESS



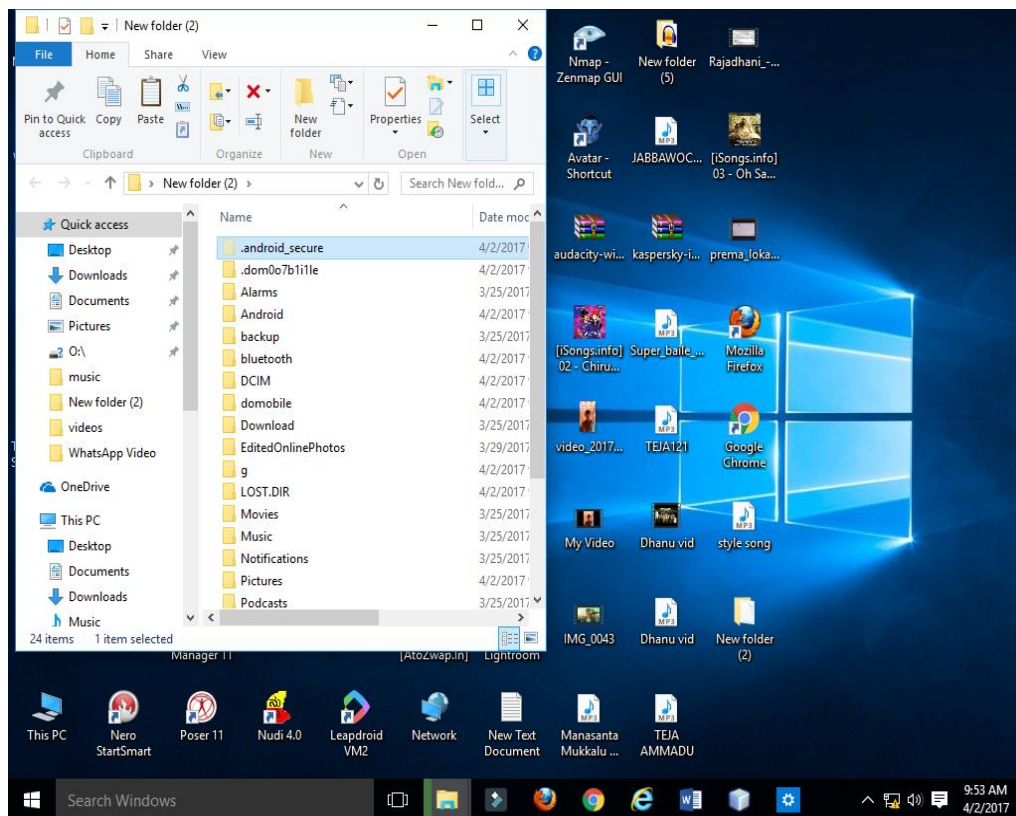
COPY PROCESS



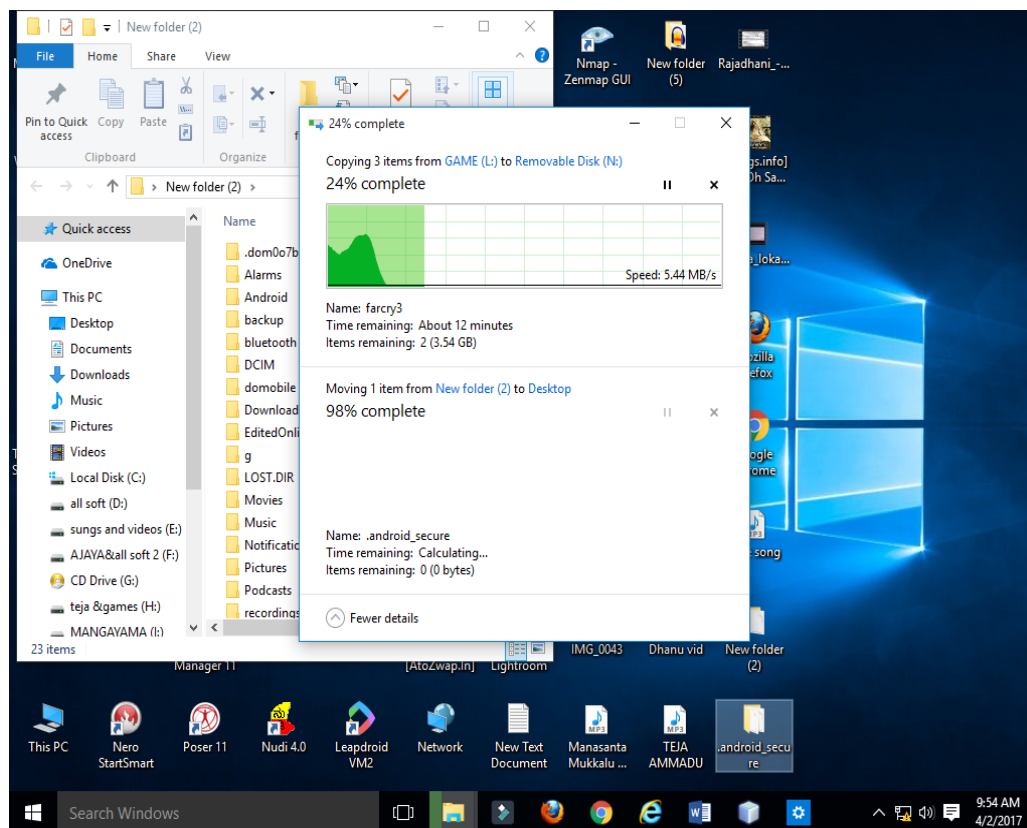
PASTE PROCESS



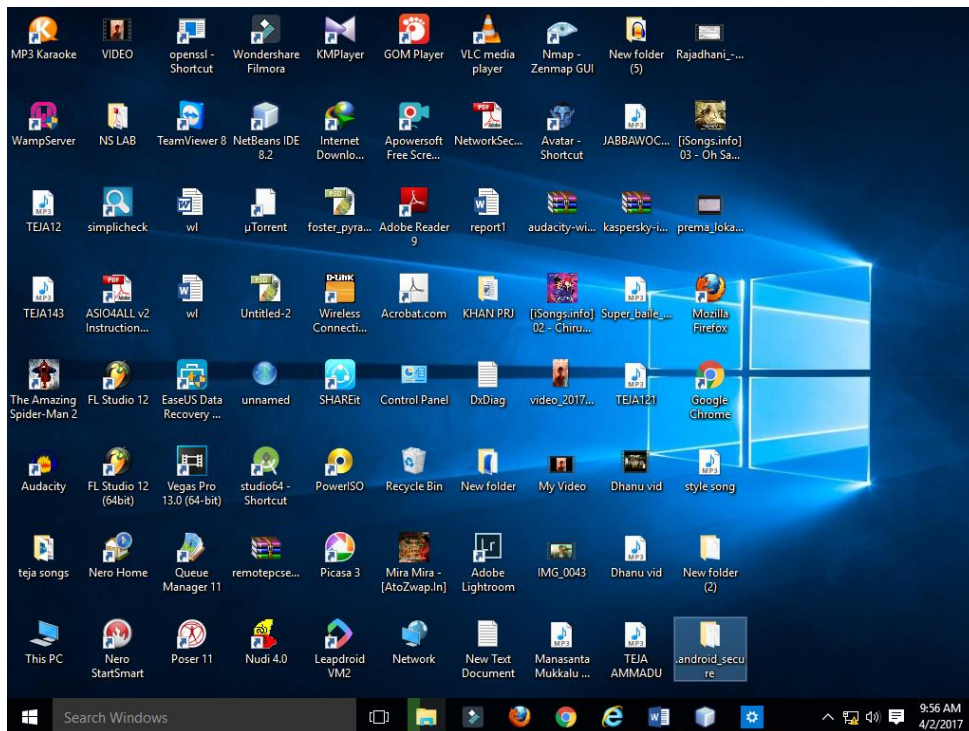
CUT PROCESS



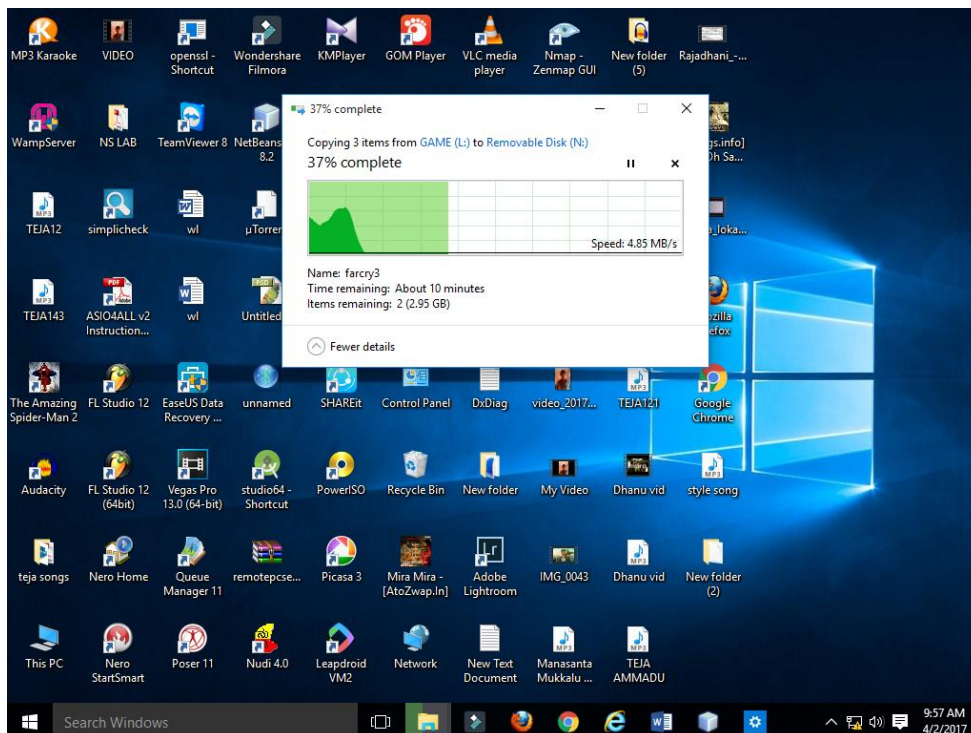
PASTE PROCESS



SELECT PROCESS



DELETE PROCESS



CHAPTER 10

CONCLUTION

Here by conclude that, our system is advanced system that has over come all draw back of existing system, by providing such as voice service we can control the pc by voice and we can perform cut, copy, paste, operation easily. Bluetooth connect is a real world system it is not only used to control the pc .

CHAPTER 11

FUTURE ENHANCEMENT

In Future by this project we can add any other operations like alt, ctrl buttons .and in the android device we can made the keyboard. We can connect phone to phone for communication by the Bluetooth with in building by without any internet connection.

BIBLIOGRAPHY

Bibliography :

- 1) Complete Reference
- 2) www.wikipedia.com
- 3) www.roseindia.com