

ABSTRACT

Project Title: Bluetooth Chat Application

Category: Application Development

SOFTWARE ENGINEERING PROJECT

GROUP NAME: VAI BROTHERS

GROUP MEMBER: MD TOUSHIF PRAMANIK (F17040118)

MD SAJEDUL ISLAM (F17040117)

HOSNE ARA (F17040101)

MAHBUB AL HOSSAIN (F17040105)

SHAH MOHAMMAD SIMRAN BILLAH (F17040115)

Description:

Bluetooth Chatting, messenger to send short messages over short distances, communication doesn't result in any charges, messages are sending directly between devices, range depends on the device and obstacles, sending can sometimes take little longer if program have difficulties, observe status icon next to recipient name. Communication takes place directly between two devices equipped with this software, devices must be within communication range, devices must be paired and Bluetooth adapter must be enabled.

Documentation for Users

How to obtain and install:

This app is available on github.com (an open source platform for developers). The link is added in this file. Anyone can go to this link, download and install this app. An Android user can also obtain this file by any device to device sharing method.

Link -

- The installation process for this app is similar to any other app available on official websites of secured apps (such as Google play store, Mi App store) and doesn't require any prerequisites. For instance some third party app requires root permission. But this app has no such requirements. A user can simply obtain and install it without changing anything. However there is only one challenge, user might need to give permission to install app from unknown sources. Once this permission is given, the app can be successfully installed.
- To run and use this app, there is no need to include any libraries, emulator or third party app. This is a compact app with all predetermined settings set.
- This app requires permission to use Bluetooth and device IP to run. It doesn't need access to internet, SIM card, contacts etc. Once the app is open, user need to pair his mobile phone with another android device that has the same version of this app installed. Once the phones are connected via Bluetooth, two users are automatically connected in the chat app and can start communicating. The apps and phones will be connected as long as the users want. Functional requirements for this app is listed in short below,
 - Check whether device is in range.
 - PING function.
 - Send Message.
 - Receive Message.

Documentation for Developers:

To develop a better version of this app or to make a similar one, we encourage other developers to follow our source codes. Original source code for the latest stable version can be found in this link -----

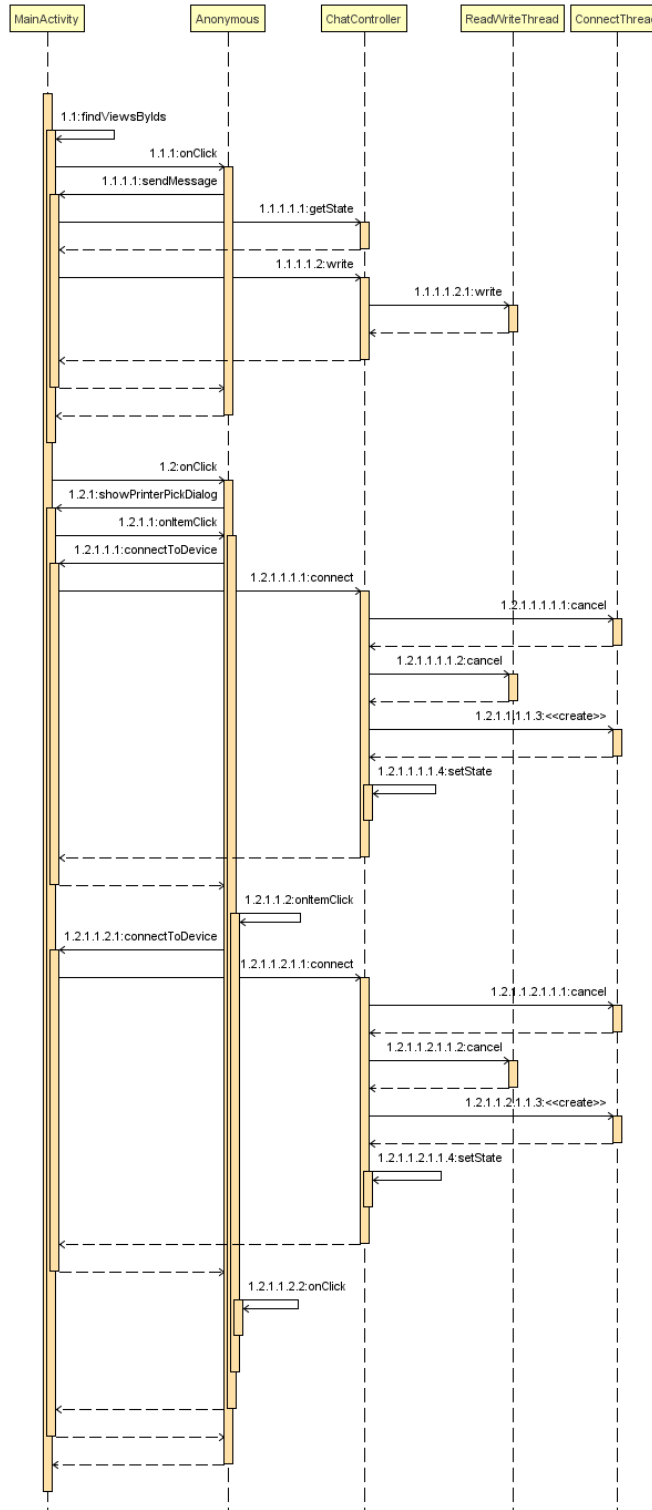
Hardware Requirements:

1. Android mobile with a minimum version 2.2.
2. Processor is at least 500MHZ.
3. RAM is at least 170MB.

Software Requirements:

1. Mobile Operating System: Android 2.2 or Later
2. Server Operating System: Windows XP or Later
3. Tools (IDE): Android Studio
4. User Interface: XML
5. Code Behind: JAVA and XML
6. Telephony Service: Yes
7. Internet: Not needed

Sequence Diagram:

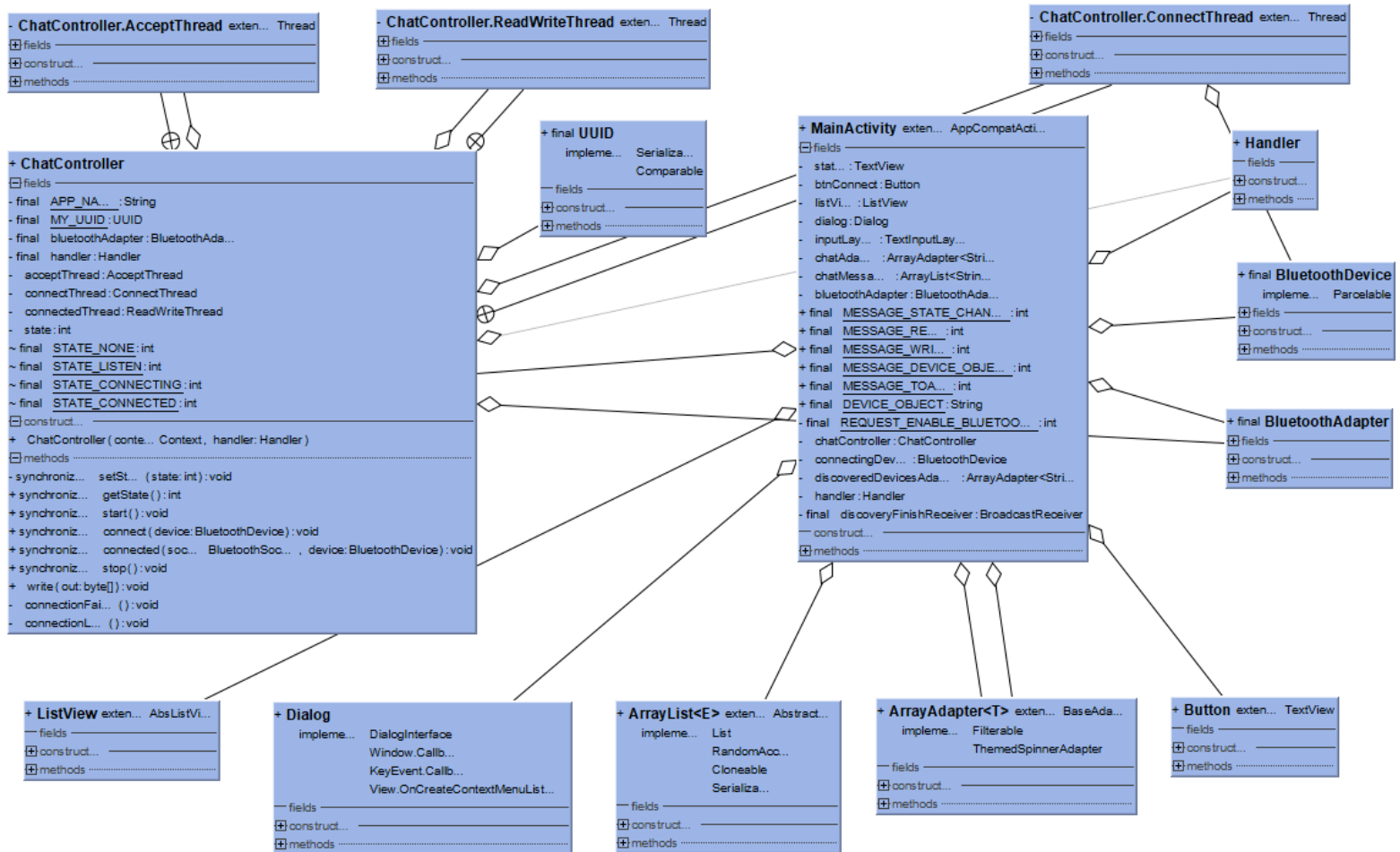


Simple UML design :

This design is directly from our code ,

It is made by the help of android studio plugins .

And the plugin's name is “SimpleUML Diagram”



For now, there is no options for users or developers to report a bug. But we are planning on adding this feature on newest release.

To accomplish this project finally, we divided this task into 5 different parts and each member finished one part of the whole project. But it doesn't mean one is ignorant about other parts of the project. The tasks was divided into these parts –

1. Environment Construct
2. Project UI design
3. Chat Controller main activity
4. Presentation

Environment Construct: In order to start making this app, first we need to create and develop app environment in windows. For this we used android studio version 3.5 on windows 10. The basic version of this app can also be made on lower versions on android studio on windows 10, windows 8, windows vista and XP. This part of project was done by *MD MAHBUB AL HOSSAIN*, Student id – *F17040105*.

I build and run the project with android studio by some steps,

Run process:

- In the toolbar, select your app from the run configurations drop-down menu.
- From the target device drop-down menu, select the device that you want to run your app on.
- Click **Run**

Build process:

Displays the tasks Gradle executes as a tree, where each node represents either a build phase or a group of task dependencies.

Sync tab: Displays tasks that Gradle executes to sync with your project files. Similar to the **Build** tab, if you encounter a sync error, select elements in the tree to find more information about the error.

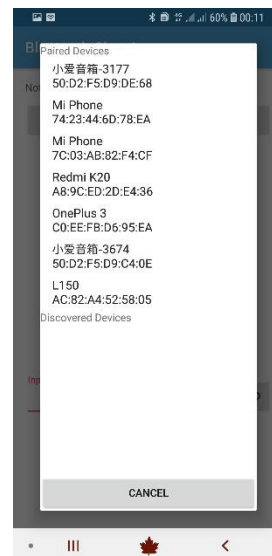
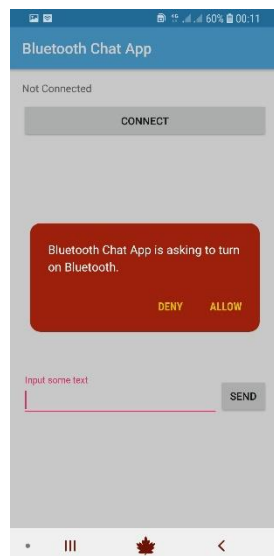
Restart: Performs the same action as selecting **Build > Make Project** by generating intermediate build files for all modules in your project.

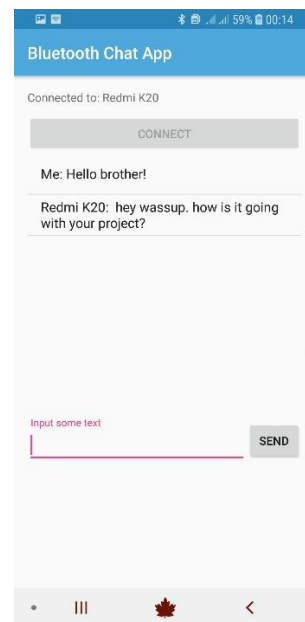
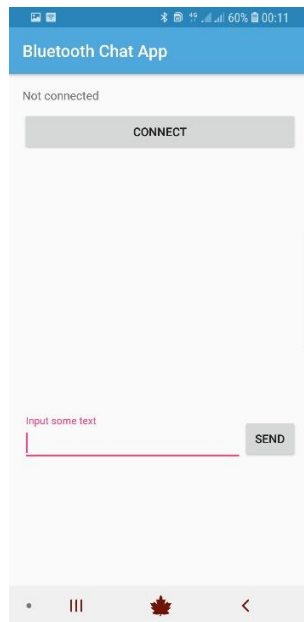
Toggle view: Toggles between displaying task execution as a graphical tree and displaying more detailed text output from Gradle—this is the same output you see in the **Gradle Console**.

Project UI design: The project UI i.e. the layout of this app is a single view page. The connecting window, chat information, connectivity and other information are shown in the same page view. The app does not store any data, therefore there is no see data page on this app. Also all settings are predetermined by developers, so no settings layout is added. This part of the project was finished by *MD Toushif Pramanik*, Student id – *F17040118*.

In the Layout Editor, I build layouts by dragging UI elements into a visual design editor instead of writing the layout XML by hand. The design editor can preview layout on different Android devices and versions, and I can dynamically resize the layout to be sure it works well on different screen sizes. I worked several field in project UI design that was,

- **Design and blueprint**
- **Screen orientation and layout variants**
- **Device type and size**
- **API version**
- **App theme**





Chat Controller / main activity: This can be considered as the main part of this project. This part is mainly responsible for Bluetooth control, connectivity, chatting and etc. This required a lot of coding and constructing. This was a hard part of this project and successfully done by *MD Sajedul Islam*, Student ID – *F17040117*.

Coding part was so hard because it is the first time for us to make an android app.

We divided our coding into two parts. One is chat controller and another part is Mainactivity. As you want, we have added the comment line each of the following steps in the coding part. We cannot discuss all the things what we have done in the code because then it will be more than one page.

But I want to mention one thing that is very important for our project. As we made a Bluetooth app, we must need to import “android.bluetooth.BluetoothAdapter”.

Without importing this, our apps will not run. Also we need to import “android.bluetooth.BluetoothDevice” in the code.

We also took help from many website to build this app and for the coding part. They are:

YouTube , Google ,projects geek , github, getstream.oi etc .

Presentation: All the works done by our team and development process we finished will be demonstrated and presented by *Shah Mohammad Simran Billah* – Student ID - *F17040115*, who also contributed in Environment Construct and Project UI design.

Leading from the front:

I am Hosne Ara (ID- F17040101) worked as the team leader of the team. I was responsible for choosing the idea of our project and helping my teammates to accomplish the tasks consecutively. I had to look after all the teammates of my team if they are organizing everything carefully. After a lot of efforts of my team members, I rechecked the project by myself and found that something was wrong with executing it. I found that gradle wasn't synchronized properly.