Chat-app.apk is in the root directory of zip file.

Chat app

- ChatActivity
    - Start “messagesAdapter” and setAdapter to listview
    - bindService to ChatService by intent. ChatService is declared in AndroidManifest.xml and get Service from “onServiceConnected()”
    - Initiate ResultReceiverWrapper in onCreate and setReceiver to ChatActivity in onResume() and setReceiver to null in onPause()
    - In onClick(), call chatService.send() to send variable to service
    - onReceiveResult() will receive call back when sent message.
    - onDestroy() will unbindService()
    -
- ChatService
    - onCreate(), create HandlerThread, start thread, and send looper of the thread to SendHandler. Create thread with ReceiveThread to start receive thread.
    - onDestroy(), interupt all thread, quit looper and close socket.
    - onSharedPreferenceChanged(), when the port change close socket and create new one. Then start new Thread for receive thread.
    - send(), obtainMessage from sendHandler and sendMessage with bundle arguments.
    - handleMessage in SendHandler will be called after sendHandler.sendMessage() has been called. It will manage all data and sent to socket.
    - ReceiverThread will be run in while loop to wait for new coming data.
        - This thread will be block when reach to chatSocket.receive(receivePacket) until new message receive.

Problem :

Proplem with Gradle version, the version and dependency of grandle is not the same so I changed it.

- Project folder > gradle>wrapper > gradle-wrapper.properties
    - **distributionUrl=https\://services.gradle.org/distributions/gradle-3.3-all.zip**
- Project folder > build.gradle ::: which is a gradle plugin that compatible with gradle
    - dependencies {
    -   classpath **'com.android.tools.build:gradle:2.3.0'**

AsyncContentResolver

- In boiler plate,
    - contentResolver = new AsyncContentResolver(context.getContentResolver());
    - So it will not be call the same content resolver when persistAsync called. I change to getAsyncResolver();

Port in preference change

- In method onSharedPreferenceChanged, chatSocket is closed and reinitiate so receiveThread has to be renew too. But because ReceiverThread is running asynchronously so sometimes chatSocket is closed after new receiveThread and socketOK will be false (from catch exeption). It makes the socketOK=false. I fixed this problem by move socketOK to local variable of ReceiverThread. So socketOK of each thread will be individual.