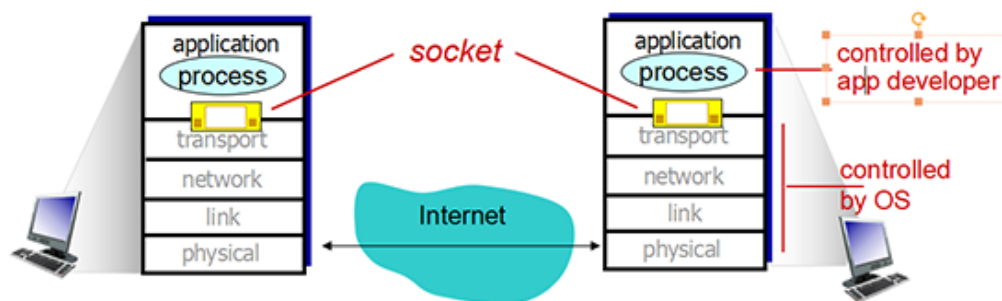




## Socket Connection trong Android



- Socket là 1 công cụ để thực hiện việc kết nối ngay lập tức giữa Client và Server cho các ứng dụng Realtime.
- **Socket là điểm cuối end-point trong liên kết truyền thông hai chiều (two-way communication) biểu diễn kết nối giữa Client – Server.** Các lớp Socket được ràng buộc với một cổng port để các tầng TCP (TCP Layer) có thể định danh ứng dụng mà dữ liệu sẽ được gửi tới



## □ Câu lệnh Java Client/Server

- Lớp ClientSocket cung cấp giao diện socket phía client. A client có thể thiết lập một kết nối tích cực đến một máy chủ từ xa bằng cách tạo một thể hiện của Socket như sau: **Socket <connection>= new Socket (<host name>, <port number>);** với tham số tên của máy chủ để kết nối; và <số cổng> phải khớp với số cổng ở cuối máy chủ.
- Lớp ServerSocket cung cấp giao diện socket phía server. Để thiết lập kết nối máy chủ và liên kết nó với một số cổng cụ thể, chúng ta nên tạo một phiên bản ServerSocket như sau: **ServerSocket <connection> = new ServerSocket(<port number>);**
- Sau khi kết nối được thiết lập, máy khách và máy chủ có thể đọc và ghi vào socket bằng các luồng đầu vào và đầu ra. Các luồng trong Java được sử dụng cho Đầu vào/Đầu ra. Chúng là các chuỗi dữ liệu được sắp xếp theo thứ tự có nguồn (luồng đầu vào) hoặc đích (luồng đầu ra).

```
dependencies {
    implementation ('io.socket:socket.io-client:2.0.0') {
        exclude group: 'org.json', module: 'json'
    }
}
```

```
<!-- app/AndroidManifest.xml -->
<manifest xmlns:android="http://schemas.android.com/apk/res/android">
    <uses-permission android:name="android.permission.INTERNET" />
    ...
</manifest>
```

```
import io.socket.client.IO;
import io.socket.client.Socket;

private Socket mSocket;
{
    try {
        mSocket = IO.socket("http://chat.socket.io");
    } catch (URISyntaxException e) {}
}

@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    mSocket.connect();
}
```

```
private EditText mInputMessageView;

private void attemptSend() {
    String message = mInputMessageView.getText().toString().trim();
    if (TextUtils.isEmpty(message)) {
        return;
    }

    mInputMessageView.setText("");
    mSocket.emit("new message", message);
}
```

```
@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);

    mSocket.on("new message", onNewMessage);
    mSocket.connect();
}
```

```
@Override
public void onDestroy() {
    super.onDestroy();

    mSocket.disconnect();
    mSocket.off("new message", onNewMessage);
}
```

```
import io.socket.emitter.Emitter;

private Emitter.Listener onNewMessage = new Emitter.Listener() {
    @Override
    public void call(final Object... args) {
        getActivity().runOnUiThread(new Runnable() {
            @Override
            public void run() {
                JSONObject data = (JSONObject) args[0];
                String username;
                String message;
                try {
                    username = data.getString("username");
                    message = data.getString("message");
                } catch (JSONException e) {
                    return;
                }

                // add the message to view
                addMessage(username, message);
            }
        });
    }
};
```



## HƯỚNG DẪN KẾT NỐI APP VỚI BLUETOOTH

Khoa Công nghệ Thông tin  
Đại học Sư phạm Kỹ thuật TP.HCM

## □ Bước 1: Cấp quyền AndroidManifest.xml

```
<!--Cấp quyền Bluetooth-->
```

```
<uses-permission android:name="android.permission.BLUETOOTH" />
```

```
<uses-permission android:name="android.permission.BLUETOOTH_ADMIN" />
```

```
<uses-permission android:name="android.permission.BLUETOOTH_CONNECT" />
```

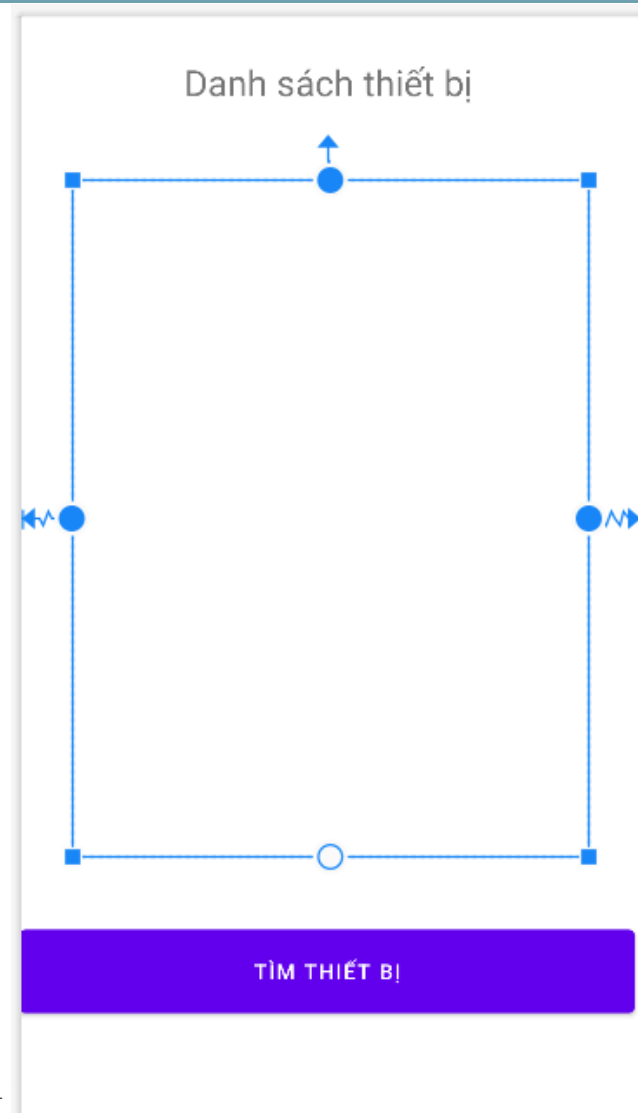
```
<uses-permission android:name="android.permission.BLUETOOTH_SCAN" />
```

Thư viện hình lấy trên khóa học



## □ Bước 1: Xây dựng Giao Diện activity\_main.xml

```
<androidx.constraintlayout.widget.ConstraintLayout
    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"
    tools:context=".MainActivity">
    <TextView
        android:id="@+id/textView2"
        android:layout_width="404dp"
        android:layout_height="51dp"
        android:text="Danh sách thiết bị"
        android:textAlignment="center"
        android:textSize="24sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.285"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.039" />
```

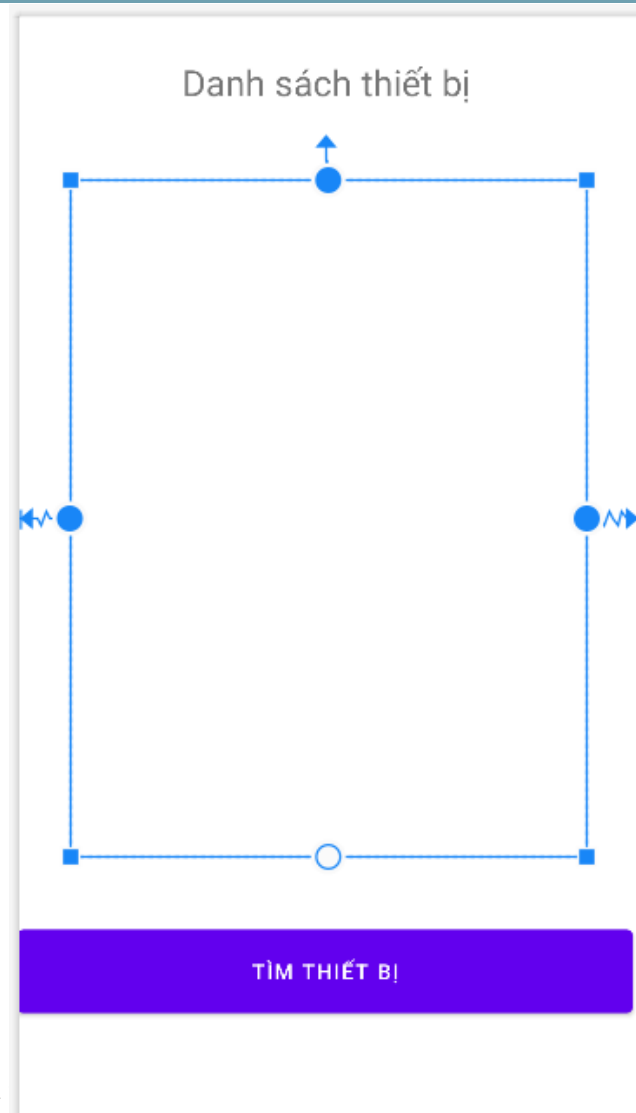


## □ Bước 1: Xây dựng Giao Diện activity\_main.xml

```
<ListView
    android:id="@+id/listTb"
    android:layout_width="343dp"
    android:layout_height="450dp"
    android:layout_margin="30dp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textView2" />

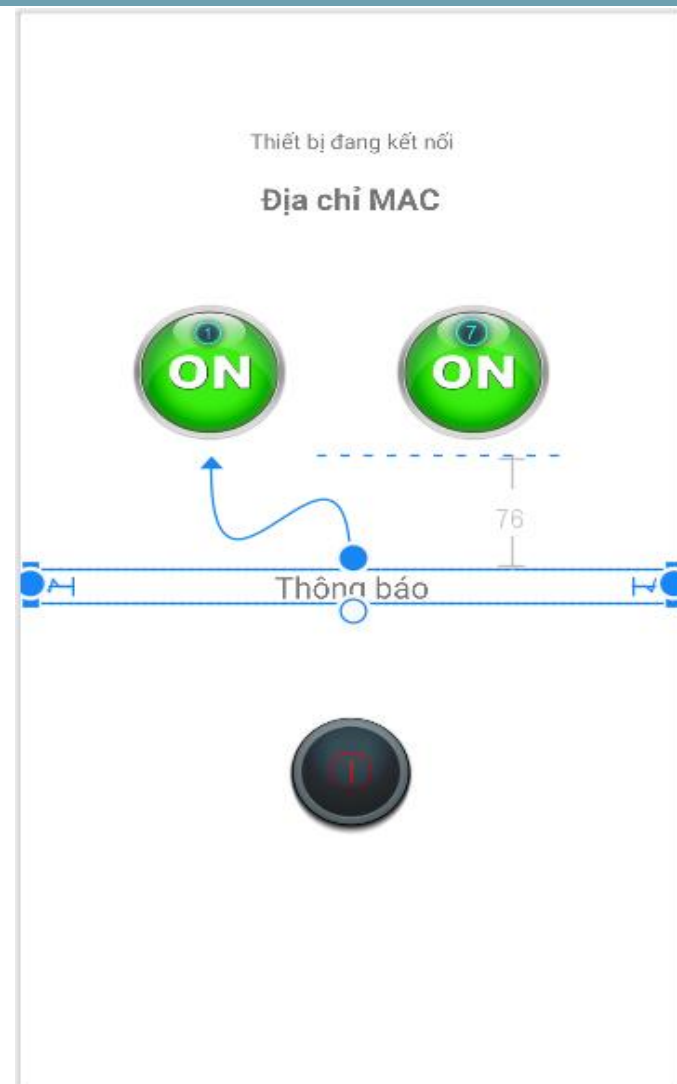
<Button
    android:id="@+id/btnTimthietbi"
    android:layout_width="409dp"
    android:layout_height="67dp"
    android:layout_marginTop="44dp"
    android:layout_marginEnd="4dp"
    android:text="Tìm Thiết Bị"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/listTb" />

</androidx.constraintlayout.widget.ConstraintLayout>
```



## □ Bước 1: Xây dựng Giao Diện activity\_control.xml

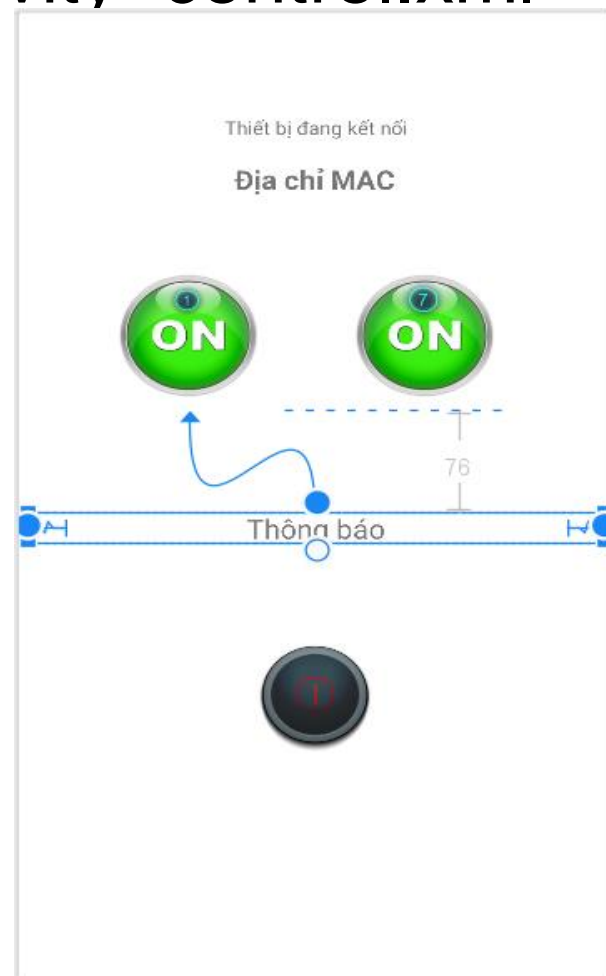
```
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".BlueControl">
    <ImageButton
        android:id="@+id/btnTb1"
        android:layout_width="116dp"
        android:layout_height="112dp"
        android:layout_marginStart="60dp"
        android:layout_marginTop="92dp"
        android:background="@drawable/tb1on"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView" />
    <ImageButton
        android:id="@+id/btnTb2"
        android:layout_width="116dp"
        android:layout_height="112dp"
        android:layout_marginTop="92dp"
        android:layout_marginEnd="60dp"
        android:background="@drawable/tb7on"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.813"
        app:layout_constraintStart_toEndOf="@+id/btnTb1"
        app:layout_constraintTop_toBottomOf="@+id/textView" />
```



## □ Bước 1: Xây dựng Giao Diện activity control.xml

```
<TextView
    android:id="@+id/textV1"
    android:layout_width="400dp"
    android:layout_height="23dp"
    android:layout_marginStart="32dp"
    android:layout_marginTop="76dp"
    android:layout_marginEnd="32dp"
    android:gravity="center_horizontal|center_vertical"
    android:text="Thông báo"
    android:textSize="20sp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/btnTb1" />

<TextView
    android:id="@+id/textView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="76dp"
    android:text="Thiết bị đang kết nối"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
```

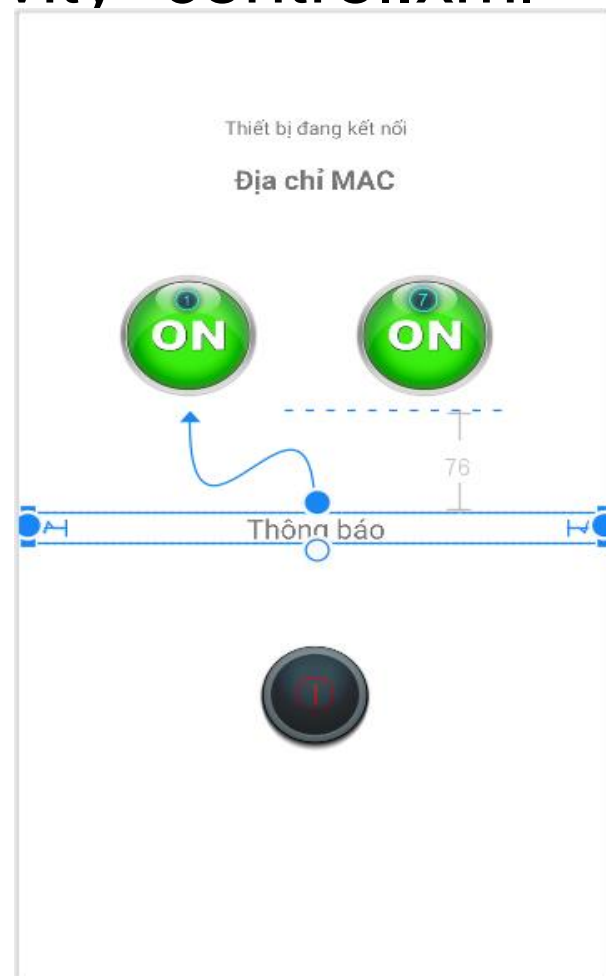


## □ Bước 1: Xây dựng Giao Diện activity control.xml

```

<TextView
    android:id="@+id/textViewMAC"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="16dp"
    android:text="Địa chỉ MAC"
    android:textSize="20sp"
    android:textStyle="bold"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.498"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textView" />

<ImageButton
    android:id="@+id/btnDisc"
    android:layout_width="90dp"
    android:layout_height="90dp"
    android:layout_alignParentStart="true"
    android:layout_alignParentLeft="true"
    android:layout_marginTop="76dp"
    android:background="@drawable/btnnotconnect"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.498"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textV1" />
</androidx.constraintlayout.widget.ConstraintLayout>
    
```



## □ Bước 2: Viết Code điều khiển MainActivity.java

```
public class MainActivity extends AppCompatActivity {  
    Button btnPaired;  
    ListView listDanhSach;  
    public static int REQUEST_BLUETOOTH = 1;  
    //Bluetooth  
    private BluetoothAdapter myBluetooth = null;  
    private Set<BluetoothDevice> pairedDevices;  
    public static String EXTRA_ADDRESS = "device_address";  
}
```

## □ Bước 2: Viết Code điều khiển MainActivity.java

```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    //ánh xạ
    btnPaired = (Button) findViewById(R.id.btnTimthietbi);
    listDanhSach=(ListView) findViewById(R.id.listTb);
    //Kiểm tra thiết bị có Bluetooth
    myBluetooth = BluetoothAdapter.getDefaultAdapter();
    if(myBluetooth == null){
        //Show a message. that the device has no bluetooth adapter
        Toast.makeText(getApplicationContext(), text: "Thiết bị Bluetooth chưa bật", Toast.LENGTH_LONG).show();
        //finish apk
        finish();
    }else if(!myBluetooth.isEnabled()){
        //Ask to the user turn the bluetooth on
        Intent turnBTon = new Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);
        if (ActivityCompat.checkSelfPermission( context: this
            |, Manifest.permission.BLUETOOTH_CONNECT) != PackageManager.PERMISSION_GRANTED) {
            Toast.makeText(getApplicationContext(), text: "Thiết bị Bluetooth chưa bật", Toast.LENGTH_LONG).show();
        }
        Toast.makeText(getApplicationContext(), text: "Thiết bị Bluetooth đã bật", Toast.LENGTH_LONG).show();
        // startForResult.launch(turnBTon);
        startActivityForResult(turnBTon, REQUEST_BLUETOOTH);
    }
    //Kết thúc kiểm tra thiết bị có bluetooth
    //thực hiện tìm thiết bị
    btnPaired.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) { pairedDevicesList(); //gọi hàm tìm thiết bị }
    });
}
```

## □ Bước 2: Viết Code điều khiển MainActivity.java

```
//viết hàm
private void pairedDevicesList() {
    if (ActivityCompat.checkSelfPermission( context: this
        , Manifest.permission.BLUETOOTH_CONNECT) != PackageManager.PERMISSION_GRANTED) {
        pairedDevices = myBluetooth.getBondedDevices();
        ArrayList list = new ArrayList();

        if (pairedDevices.size() > 0) {
            for (BluetoothDevice bt : pairedDevices) {
                if (ActivityCompat.checkSelfPermission( context: this
                    , Manifest.permission.BLUETOOTH_CONNECT) != PackageManager.PERMISSION_GRANTED) {
                    Toast.makeText(getApplicationContext(), text: "Danh sách thiết bị Bluetooth đã bật", Toast.LENGTH_LONG).show();
                    list.add(bt.getName() + "\n" + bt.getAddress()); //Get the device's name and the address
                }
            }
        }
        else
        {
            Toast.makeText(getApplicationContext(), text: "Không tìm thấy thiết bị kết nối.", Toast.LENGTH_LONG).show();
        }

        final ArrayAdapter adapter = new ArrayAdapter( context: this, android.R.layout.simple_list_item_1, list);
        listDanhSach.setAdapter(adapter);
        listDanhSach.setOnItemClickListener(myItemClickListener); //Method called when the device from the list is clicked
        return;
    }
}
```



## □ Bước 3: Viết Code điều khiển BlueControl.java

```
public class BlueControl extends AppCompatActivity {
    //public static final int REQUEST_BLUETOOTH = 1;
    ImageButton btnTb1, btnTb2, btnDis;
    TextView txt1, txtMAC;
    BluetoothAdapter myBluetooth = null;
    BluetoothSocket btSocket = null;
    private boolean isBtConnected = false;
    Set<BluetoothDevice> pairedDevices1;
    String address = null;
    private ProgressDialog progress;
    int flaglamp1;
    int flaglamp2;
    //SPP UUID. Look for it
    static final UUID myUUID = UUID.fromString("00001101-0000-1000-8000-00805F9B34FB");
```

## □ Bước 3: Viết Code điều khiển BlueControl.java

```
@Override
protected void onCreate(@Nullable Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    Intent newint = getIntent();
    address = newint.getStringExtra(MainActivity.EXTRA_ADDRESS); //receive the address of the bluetooth device
    setContentView(R.layout.activity_control);
    //ánh xạ
    btnTb1 = (ImageButton) findViewById(R.id.btnTb1);
    btnTb2 = (ImageButton) findViewById(R.id.btnTb2);
    txt1 = (TextView) findViewById(R.id.textV1);
    txtMAC = (TextView) findViewById(R.id.textViewMAC);
    btnDis = (ImageButton) findViewById(R.id.btnDisc);
    new ConnectBT().execute();//Call the class to connect
    btnTb1.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) { thietTbi1(); //gọi hàm }
    });
    btnTb2.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) { thietTbi7(); //gọi hàm }
    });
    btnDis.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) { Disconnect(); }
    });
}
```

## □ Bước 3: Viết Code điều khiển BlueControl.java

```
//viết hàm
private void thietTbi1() {
    if (btSocket != null) {
        try {

            if (this.flaglamp1 == 0) {
                this.flaglamp1 = 1;
                this.btnTb1.setBackgroundResource(R.drawable.tb1on);
                btSocket.getOutputStream().write("1".toString().getBytes());
                txt1.setText("Thiết bị số 1 đang bật");
                return;
            } else {
                if (this.flaglamp1 != 1) return;
                {
                    this.flaglamp1 = 0;
                    this.btnTb1.setBackgroundResource(R.drawable.tb1off);
                    btSocket.getOutputStream().write("A".toString().getBytes());
                    txt1.setText("Thiết bị số 1 đang tắt");
                    return;
                }
            }
        }

    } catch (IOException e) {
        msg(s: "Lỗi");
    }
}
```

```
private void Disconnect() {
    if (btSocket != null) //If the btSocket is busy
    {
        try {
            btSocket.close(); //close connection
        } catch (IOException e) {
            msg(s: "Lỗi");
        }
    }
    finish(); //return to the first layout
}
```

## □ Bước 3: Viết Code điều khiển BlueControl.java

```
private void thiettb17() {
    if (btSocket != null) {
        try {
            if (this.flaglamp2 == 0) {
                this.flaglamp2 = 1;
                this.btnTb2.setBackgroundResource(R.drawable.tb2on);
                btSocket.getOutputStream().write("7".getBytes());
                txt1.setText("Thiết bị số 7 đang bật");
                return;
            } else {
                if (this.flaglamp2 != 1) return;
                {
                    this.flaglamp2 = 0;
                    this.btnTb2.setBackgroundResource(R.drawable.tb2off);
                    btSocket.getOutputStream().write("6".getBytes());
                    txt1.setText("Thiết bị số 7 đang tắt");
                    return;
                }
            }
        }
    }
} catch (IOException e) {
    msg(s: "Lỗi");
}
}
```

## □ Bước 3: Viết Code điều khiển BlueControl.java

```
private class ConnectBT extends AsyncTask<Void, Void, Void> // UI thread
{
    private boolean ConnectSuccess = true; //if it's here, it's almost connected
    @Override
    protected void onPreExecute() {
        //show a progress dialog
        progress = ProgressDialog.show(context: BlueControl.this, title: "Đang kết nối...", message: "Xin vui lòng đợi!!!");
    }
    @Override
    protected Void doInBackground(Void... devices) //while the progress dialog is shown, the connection is done in background
    {
        try {
            if (btSocket == null || !isBtConnected) {
                myBluetooth = BluetoothAdapter.getDefaultAdapter(); //get the mobile bluetooth device
                //connects to the device's address and checks if it's available
                BluetoothDevice dispositivo = myBluetooth.getRemoteDevice(address);
                if (ActivityCompat.checkSelfPermission(context: BlueControl.this
                    , Manifest.permission.BLUETOOTH_CONNECT) != PackageManager.PERMISSION_GRANTED) {
                    //create a RFCOMM (SPP) connection
                    btSocket = dispositivo.createInsecureRfcommSocketToServiceRecord(myUUID);
                    BluetoothAdapter.getDefaultAdapter().cancelDiscovery();
                    btSocket.connect(); //start connection
                }
            }
        }
        catch (IOException e)
        {
            ConnectSuccess = false; //if the try failed, you can check the exception here
        }
        return null;
    }
}
```

## □ Bước 3: Viết Code điều khiển BlueControl.java

```
@Override
protected void onPostExecute(Void result) //after the doInBackground, it checks if everything went fine
{
    super.onPostExecute(result);

    if (!ConnectSuccess)
    {
        msg(s: "Kết nối thất bại ! Kiểm tra thiết bị.");
        finish();
    }
    else
    {
        msg(s: "Kết nối thành công.");
        isBtConnected = true;
        pairedDevicesList1();
    }
    progress.dismiss();
}
```

```
// fast way to call Toast
```

```
private void msg(String s) {
    Toast.makeText(getApplicationContext(), s, Toast.LENGTH_LONG).show();
}
```

## □ Bước 3: Viết Code điều khiển BlueControl.java

```
private void pairedDevicesList1() {

    if (ActivityCompat.checkSelfPermission(context: this
        , Manifest.permission.BLUETOOTH_CONNECT) != PackageManager.PERMISSION_GRANTED) {
        pairedDevices1 = myBluetooth.getBondedDevices();

        if (pairedDevices1.size() > 0) {
            for (BluetoothDevice bt : pairedDevices1) {
                txtMAC.setText(bt.getName() + " - " + bt.getAddress()); //Get the device's name and the address
            }
        } else {
            Toast.makeText(getApplicationContext(), text: "Không tìm thấy thiết bị kết nối.", Toast.LENGTH_LONG).show();
        }
    }
}
```

## □ Bước 4: Nạp Code vào thiết bị

```
#include <SoftwareSerial.h>
SoftwareSerial mySerial(10, 11);
#define tb1 2
#define tb2 3
char val;
String statustb1,statustb2;
void setup() {
    pinMode(tb1,OUTPUT);
    digitalWrite(tb1,LOW);
    pinMode(tb2,OUTPUT);
    digitalWrite(tb2,LOW);
    mySerial.begin(9600);
    Serial.begin(9600);
}
void loop() {
    //cek data serial from bluetooth android App
    if( mySerial.available() >0 ) {
        val = mySerial.read();
        Serial.println(val);
    }
    if( val == '1' ) {
        digitalWrite(tb1,HIGH); statustb1="1"; }
    else if( val == '2' ) {
        digitalWrite(tb2,HIGH); statustb2="2"; }
    else if( val == 'A' ) {
        digitalWrite(tb1,LOW); statustb1="A"; }
    else if( val == 'B' ) {
        digitalWrite(tb2,LOW); statustb2="B"; }

    else if( val == 'S' ) {
        delay(500);
        mySerial.println(statustb1+statustb2+"J"); //delay(500);
        val=' ';
    }
}
```



- Nguyễn Hữu Trung
- 0908617108
- [trungnh@hcmute.edu.vn](mailto:trungnh@hcmute.edu.vn)
- [utex.hcmute.edu.vn](http://utex.hcmute.edu.vn)