**MainActivity.kt**

package com.daumin.apsvietimas  
  
import android.support.v7.app.AppCompatActivity  
import android.os.Bundle  
import android.view.View  
import android.widget.\*  
import kotlinx.android.synthetic.main.activity\_main.\*  
import android.widget.Toast  
import android.bluetooth.BluetoothAdapter  
import android.content.Intent  
import android.bluetooth.BluetoothDevice  
import android.bluetooth.BluetoothSocket  
import android.os.AsyncTask  
import java.lang.Exception  
import java.util.\*  
import android.text.method.\*  
  
  
class MainActivity : AppCompatActivity() {  
  
 val DEVICE\_NAME = "BLUETOOTH SHIELD NAME"  
  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_main*)  
  
 val jungiklis: Switch = findViewById(R.id.*switch1*)  
 val spinner: Spinner = findViewById(R.id.*pasirinkimu\_spinner*)  
 val buttonas: Button = findViewById(R.id.*button*)  
 var sviesosLygis: SeekBar = findViewById(R.id.*seekBar1*)  
 var device: BluetoothDevice  
 var found: Boolean = false  
 logas.*movementMethod* = ScrollingMovementMethod();  
  
 jungiklis.*isEnabled* = false  
 spinner.*isEnabled* = false  
 sviesosLygis.*isEnabled* = false  
  
 ArrayAdapter.createFromResource(  
 this,  
 R.array.*pasirinkimai*,  
 android.R.layout.*simple\_spinner\_item* ).*also* **{** adapter **->** adapter.setDropDownViewResource(android.R.layout.*simple\_spinner\_dropdown\_item*)  
 pasirinkimu\_spinner.*adapter* = adapter  
 **}** jungiklis.setOnCheckedChangeListener **{** \_, isChecked **->** if (isChecked) {  
 spinner.*isEnabled* = true  
 if (spinner.*selectedItem*.equals("Rankinis valdymas"))  
 sviesosLygis.*isEnabled* = true  
 } else {  
 spinner.*isEnabled* = false  
 sviesosLygis.*isEnabled* = false  
 }  
 **}** buttonas.setOnClickListener **{** logas.*text* = ""  
 **}** spinner.*onItemSelectedListener* = object : AdapterView.OnItemSelectedListener {  
 override fun onNothingSelected(parent: AdapterView<\*>?) {  
 }  
  
 override fun onItemSelected(parent: AdapterView<\*>?, view: View?, position: Int, id: Long) {  
 if (pasirinkimu\_spinner.*selectedItem*.equals("Pagal apšvietimą")) {  
 sviesosLygis.*isEnabled* = false  
 } else if (pasirinkimu\_spinner.*selectedItem*.equals(("Pagal temperatūrą"))) {  
 sviesosLygis.*isEnabled* = false  
 } else if (pasirinkimu\_spinner.*selectedItem*.equals("Rankinis valdymas")) {  
 sviesosLygis.*isEnabled* = true  
 }  
 }  
 }  
  
 // Bluetooth device  
 val bluetoothAdapter = BluetoothAdapter.getDefaultAdapter()  
  
 try {  
 if (!bluetoothAdapter.*isEnabled*) {  
 val enableAdapter = Intent(BluetoothAdapter.*ACTION\_REQUEST\_ENABLE*)  
 startActivityForResult(enableAdapter, 0)  
 }  
  
 if (bluetoothAdapter.*isEnabled*) {  
 val bondedDevices = bluetoothAdapter.*bondedDevices* if (bondedDevices.isEmpty()) {  
 Toast.makeText(this@MainActivity, "Nėra suporuotų prietaisų", Toast.*LENGTH\_SHORT*).show()  
 } else {  
 for (iterator in bondedDevices) {  
 if (iterator.*name* == DEVICE\_NAME && !found) {  
 found = true  
 jungiklis.*isEnabled* = true  
 logas.append("Device name: " + iterator.*name* + "\n")  
  
 device = iterator //device is an object of type BluetoothDevice  
  
 DeviceHandler().execute(device)  
  
 break  
  
 }  
 }  
 }  
 }  
 } catch (ex: Exception) {  
 logas.append("Klaida: " + ex.toString() + "\n")  
 }  
  
 }  
  
 inner class DeviceHandler : AsyncTask<BluetoothDevice, Void, Void>() {  
  
 override fun doInBackground(vararg params: BluetoothDevice?): Void? {  
 try {  
 val socket =  
 params[0]!!.createRfcommSocketToServiceRecord(UUID.fromString("00001101-0000-1000-8000-00805f9b34fb"))  
 socket.connect()  
  
 OutHandler(socket).execute()  
 //InpHandler(socket).execute()  
 } catch (ex: Exception) {  
 logas.append(ex.toString() + "\n")  
 }  
 return null  
 }  
  
 }  
  
 inner class OutHandler(handler: BluetoothSocket) : AsyncTask<Void, String, Void?>() {  
 private val socket: BluetoothSocket = handler  
 private var jungiklis: Boolean = false  
 private var pasirinkimas: Int = 0  
 private var intensyvumas: Int = 0  
  
 override fun onPreExecute() {  
 super.onPreExecute()  
 }  
  
 override fun doInBackground(vararg params: Void): Void? {  
  
 val outputStream = socket.getOutputStream()  
 try {  
 while (true) {  
 var eilute: String = ""  
 if (switch1.*isChecked* == true) {  
  
 if (pasirinkimas == 2) {  
 eilute = "2." + intensyvumas.toString()  
 try {  
 outputStream.write(eilute.*toByteArray*(*charset*("UTF\_8")))  
 } catch (excep: Exception) {  
 logas.append("Klaida: " + excep.toString() + "\n")  
 }  
 } else {  
 eilute = pasirinkimas.toString()  
 try {  
 outputStream.write(eilute.*toByteArray*(*charset*("UTF\_8")))  
 } catch (excep: Exception) {  
 logas.append("Klaida: " + excep.toString() + "\n")  
 }  
 }  
  
 } else {  
 eilute = "-1"  
 outputStream.write(eilute.*toByteArray*(*charset*("UTF\_8")))  
 }  
 publishProgress(eilute)  
 Thread.sleep(1000)  
 }  
 } catch (ex: Exception) {  
 logas.append("Klaida: " + ex.toString() + "\n")  
 }  
 return null  
 }  
  
 override fun onProgressUpdate(vararg values: String) {  
 super.onProgressUpdate(\*values)  
 logas.append("Išsiųsta: " + values[0] + "\n")  
 pasirinkimas = pasirinkimu\_spinner.*selectedItemPosition* if (pasirinkimas == 2 && switch1.*isChecked* == true) {  
 intensyvumas = seekBar1.*progress* }  
 }  
 }  
  
  
 inner class InpHandler(handler: BluetoothSocket) : AsyncTask<Void, String, String>() {  
  
 private val socket: BluetoothSocket = handler  
  
 override fun doInBackground(vararg params: Void): String {  
 val inputStream = socket.getInputStream()  
 try {  
 while (false) { // PAKEISTI Į TRUE  
 var byteCount = inputStream.available()  
 var rawBytes: ByteArray = ByteArray(byteCount)  
 var input: String = ""  
  
 if (byteCount > 0) {  
 inputStream.read(rawBytes)  
 input = *String*(rawBytes, *charset*("UTF\_8"))  
 }  
 publishProgress(input)  
 }  
 } catch (ex: Exception) {  
 logas.append("Klaida: " + ex.toString() + "\n")  
 }  
 return ""  
 }  
  
 override fun onProgressUpdate(vararg values: String) {  
 super.onProgressUpdate(\*values)  
 try {  
 logas.append("Gauta: " + values[0] + "\n")  
 var intensyvumas = values[0].*toInt*()  
 if (intensyvumas >= 0 && intensyvumas <= 5) {  
 seekBar1.*progress* = intensyvumas  
 }  
 } catch (ex: Exception) {  
 logas.append("Klaida: " + ex.toString() + "\n")  
 }  
 }  
  
 }  
}

**activity\_main.xml**

<?xml version="1.0" encoding="utf-8"?>  
<android.support.constraint.ConstraintLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity">  
 <Switch  
 android:text="Apšvietimas"  
 android:layout\_width="355dp"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/switch1" app:layout\_constraintEnd\_toEndOf="parent" android:layout\_marginEnd="8dp"  
 android:layout\_marginRight="8dp" android:layout\_marginTop="32dp" app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent" android:layout\_marginLeft="8dp"  
 android:layout\_marginStart="8dp"/>  
 <Spinner  
 android:layout\_width="355dp"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/pasirinkimu\_spinner" android:layout\_marginTop="32dp"  
 app:layout\_constraintTop\_toBottomOf="@+id/switch1" app:layout\_constraintStart\_toStartOf="parent"  
 android:layout\_marginLeft="8dp" android:layout\_marginStart="8dp" app:layout\_constraintEnd\_toEndOf="parent"  
 android:layout\_marginEnd="8dp" android:layout\_marginRight="8dp"/>  
 <SeekBar  
 style="@style/Widget.AppCompat.SeekBar.Discrete"  
 android:layout\_width="288dp"  
 android:layout\_height="31dp"  
 android:max="5"  
 android:progress="0"  
 android:id="@+id/seekBar1" android:layout\_marginTop="44dp"  
 app:layout\_constraintTop\_toBottomOf="@+id/pasirinkimu\_spinner" android:layout\_marginStart="8dp"  
 app:layout\_constraintStart\_toStartOf="parent" android:layout\_marginLeft="8dp" android:layout\_marginEnd="8dp"  
 app:layout\_constraintEnd\_toEndOf="parent" android:layout\_marginRight="8dp" android:visibility="visible"  
 />  
 <TextView  
 android:layout\_width="357dp"  
 android:layout\_height="322dp"  
 android:id="@+id/logas" android:layout\_marginTop="8dp"  
 app:layout\_constraintTop\_toBottomOf="@+id/seekBar1" app:layout\_constraintStart\_toStartOf="parent"  
 android:layout\_marginLeft="8dp" android:layout\_marginStart="8dp" app:layout\_constraintEnd\_toEndOf="parent"  
 android:layout\_marginEnd="8dp" android:layout\_marginRight="8dp"  
 android:scrollbars="vertical" android:layout\_marginBottom="8dp"  
 app:layout\_constraintBottom\_toTopOf="@+id/button"/>  
 <Button  
 android:text="Išvalyti"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/button"  
 app:layout\_constraintBottom\_toBottomOf="parent" android:layout\_marginBottom="16dp"  
 app:layout\_constraintStart\_toStartOf="parent" android:layout\_marginLeft="146dp"  
 android:layout\_marginStart="146dp" app:layout\_constraintEnd\_toEndOf="parent"  
 android:layout\_marginEnd="177dp" android:layout\_marginRight="177dp"/>  
</android.support.constraint.ConstraintLayout>

**strings.xml**

<resources>  
 <string name="app\_name">Apšvietimo Valdymas</string>  
 <string-array name="pasirinkimai">  
 <item>Pagal apšvietimą</item>  
 <item>Pagal temperatūrą</item>  
 <item>Rankinis valdymas</item>  
 </string-array>  
</resources>

**AndroidManifest.xml**

<?xml version="1.0" encoding="utf-8"?>  
<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 package="com.daumin.apsvietimas">  
 <uses-permission android:name="android.permission.BLUETOOTH"></uses-permission>  
 <application  
 android:allowBackup="true"  
 android:icon="@mipmap/ic\_launcher"  
 android:label="@string/app\_name"  
 android:roundIcon="@mipmap/ic\_launcher\_round"  
 android:supportsRtl="true"  
 android:theme="@style/AppTheme">  
 <activity android:name=".MainActivity">  
 <intent-filter>  
 <action android:name="android.intent.action.MAIN"/>  
  
 <category android:name="android.intent.category.LAUNCHER"/>  
 </intent-filter>  
 </activity>  
 </application>  
  
</manifest>