

Institiúid Teicneolaíochta Cheatharlach



INSTITUTE *of*
TECHNOLOGY

CARLOW

At the Heart of South Leinster

Department of Computing & Networking.

***Bachelor of Science (Hons) in Software
DevelopmentCW238***

Design and develop of a suitable activity monitor for
Children with Autism.

Technical Document.

Author: Dylan Scott

Supervisor: Dr Oisin Cawley.

Contents

Java Code	3
Session Activity	3
Settings Activity.....	12
btActivity.....	14
SelectDeviceActivity.....	19
DeviceListAdapter.....	21
DeviceInfoModel.....	23
Reference Declaration.....	23
dbHelper.....	24
ViewDataActivity.....	26
BarChartActivity.....	30
LineChartActivity.....	33
SessionModel.....	36
SearchDataAdapter.....	37
SplashScreenActivity.....	38
XML.....	39
activity_session.....	39
activity_settings.....	43
activity_main.....	46
activity_select_device.....	48
device_info_layout.....	49
activity_view_data.....	50
search_result_layout.....	52
dialog_layout.....	53
activity_bar_chart.....	54
activity_line_chart.....	56
activity_splash_screen.....	58
Arduino Code.....	59
Reference Declaration.....	62
Plagiarism Declaration.....	63

Java Code.

Session Activity

```
package com.example.steps;
import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.content.ContextCompat;
import android.Manifest;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.hardware.Sensor;
import android.hardware.SensorEvent;
import android.hardware.SensorEventListener;
import android.hardware.SensorManager;
import android.os.Bundle;
import android.view.View;
import android.view.WindowManager;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.Locale;
import java.util.Timer;
import java.util.TimerTask;

/**
 * This work was completed under the GNU GENERAL PUBLIC LICENSE
 * Copyright (C) 2007 Free Software Foundation, Inc. <https://fsf.org/>
 * Everyone is permitted to copy and distribute verbatim copies
 * of this license document, but changing it is not allowed.
 */
```

Final Year Project Technical Document

```
* Author Dylan Scott C00242050
* This file contains my projects "Session Activity"
*/

public class SessionActivity extends AppCompatActivity implements SensorEventListener {
    private TextView textViewStepCounter, dateTV,
        textViewCals, textViewDistance, textViewTimer, textViewTarget, welcomeTxt;
    private Button settingsBtn, homeBtn, stopStartBtn, saveBtn, calBtn, searchBtn;
    private SensorManager sensorManager;
    private Sensor stepCounter;
    private boolean timerStarted;
    int weight, stepCount, target, caloriesBurned;
    float height;
    float velocity = 0.9f;
    float distanceWalked = 0.0f;
    Timer timer;
    TimerTask timerTask;
    Double time = 0.0;
    private String firstName;
    private String lastName;
    private String weightN;
    private String heightN;
    private DBHelper dbHelper;

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

        setContentView(R.layout.activity_session);
        init();

        if (getIntent().hasExtra( name: "firstname")) {
            firstName = getIntent().getStringExtra( name: "firstname");
            lastName = getIntent().getStringExtra( name: "lastname");
            weightN = getIntent().getStringExtra( name: "weight");
            heightN = getIntent().getStringExtra( name: "height");
            String targetS = getIntent().getStringExtra( name: "target");
            target = Integer.parseInt(targetS);
            textViewTarget.setText(String.valueOf("Target: " + target));
            welcomeTxt.setText("Welcome " + firstName + " \n " + "Press start to begin");
        }
        else if (getIntent().getBooleanExtra( name: "EXIT" , defaultValue: false)){
            finish();
            System.exit( status: 0);
        }

        if (ContextCompat.checkSelfPermission( context: this,
            Manifest.permission.ACTIVITY_RECOGNITION) == PackageManager.PERMISSION_DENIED) {
            requestPermissions(new String[]{Manifest.permission.ACTIVITY_RECOGNITION}, requestCode: 0);
        }
        // keeps screen on while counting steps
        getWindow().addFlags(WindowManager.LayoutParams.FLAG_KEEP_SCREEN_ON);
        sensorManager = (SensorManager) getSystemService(Context.SENSOR_SERVICE);
    }
}
```

```
boolean isCounterSensor;
if (sensorManager.getDefaultSensor(Sensor.TYPE_STEP_COUNTER) != null) {
    stepCounter = sensorManager.getDefaultSensor(Sensor.TYPE_STEP_COUNTER);
    isCounterSensor = true;
} else {
    textViewStepCounter.setText("Sensor not present.");
    isCounterSensor = false;
}

// Button to open settings activity
settingsBtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent = new Intent( packageContext: SessionActivity.this, SettingsActivity.class);
        startActivity(intent);
    }
});

// Button to save session data
saveBtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        double d = (double)distanceWalked;
        if (dbHelper.insertSessionData(firstName, lastName, stepCount, caloriesBurned, d)) {
            Toast.makeText( context: SessionActivity.this, text: "Data saved successfully",
                Toast.LENGTH_SHORT).show();
        }
    }
});
```

```
// Button to open view data activity
searchBtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        startActivity(new Intent( packageContext: SessionActivity.this , ViewDataActivity.class));
    }
});

// Button to call calculate functions
calBtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        caloriesBurned = calCalories(weightN, heightN);
        textViewCals.setText(String.valueOf("Burned: " + caloriesBurned));
        distanceWalked = calDistance();
        textViewDistance.setText(String.valueOf("Distance: " + distanceWalked));
    }
});

@Override
public void onSensorChanged(SensorEvent sensorEvent) {
    if (sensorEvent.sensor == stepCounter) {
        stepCount = (int) sensorEvent.values[0];
        textViewStepCounter.setText("Steps: " + stepCount );
        if (stepCount == target) {
            AlertDialog.Builder resetAlert = new AlertDialog.Builder( context: this);
            resetAlert.setTitle("Target Reached for Today, Well done \" " + firstName);
            resetAlert.setMessage("Would like to save session and reset ?");
        }
    }
}
```

```
resetAlert.setPositiveButton( text: "Save & Reset", new DialogInterface.OnClickListener() {  
    @Override  
    public void onClick(DialogInterface dialogInterface, int i) {  
        double d = (double)distanceWalked;  
        caloriesBurned = calCalories(weightN, heightN);  
        textViewCals.setText(String.valueOf("Burned: " + caloriesBurned));  
        d = calDistance();  
        textViewDistance.setText(String.valueOf("Distance: " + distanceWalked));  
        if (dbHelper.insertSessionData(firstName, lastName, stepCount, caloriesBurned, d)) {  
            Toast.makeText( context: SessionActivity.this, text: "Data saved successfully: ",  
                Toast.LENGTH_SHORT).show();  
        }  
        if (timerTask != null) {  
            timerTask.cancel();  
            buttonUI( start: "START", R.color.green);  
            time = 0.0;  
            timerStarted = false;  
            textViewTimer.setText(formatTime( seconds: 0, minutes: 0, hours: 0));  
            stepCount = 0;  
            caloriesBurned = 0;  
            distanceWalked = 0.0f;  
            textViewDistance.setText(String.valueOf("Distance Walked"));  
            textViewCals.setText(String.valueOf("Calories Burned"));  
            welcomeTxt.setText(String.valueOf("Welcome"));  
            textViewTarget.setText(String.valueOf("Target: "));  
            stepCount = stepCount - stepCount;  
            textViewStepCounter.setText(String.valueOf("Steps : 0"));  
        }  
    }  
});
```

```
        resetAlert.setNeutralButton( text: "Continue", new DialogInterface.OnClickListener() {  
            @Override  
            public void onClick(DialogInterface dialogInterface, int i) {  
                //do nothing  
            }  
        });  
        resetAlert.show();  
    }  
}  
  
@Override  
public void onAccuracyChanged(Sensor sensor, int accuracy) {  
}  
  
@Override  
protected void onResume() {  
    super.onResume();  
    if (sensorManager.getDefaultSensor(Sensor.TYPE_STEP_COUNTER) != null) {  
        sensorManager.registerListener( listener: this, stepCounter, sensorManager.SENSOR_DELAY_NORMAL);  
    }  
}  
  
@Override  
protected void onPause() {  
    super.onPause();  
    if (sensorManager.getDefaultSensor(Sensor.TYPE_STEP_COUNTER) != null) {  
        sensorManager.unregisterListener( listener: this, stepCounter);  
    }  
}  
}
```

```
// Function to initialise activity UI  
public void init() {  
    textViewStepCounter = findViewById(R.id.stepCounterText);  
    textViewCals = findViewById(R.id.calorieText);  
    textViewTimer = findViewById(R.id.timeText);  
    textViewTarget = findViewById(R.id.targetText);  
    welcomeTxt = findViewById(R.id.welcomeText);  
    stopStartBtn = findViewById(R.id.startStopButton);  
    textViewDistance = findViewById(R.id.distanceText);  
    settingsBtn = findViewById(R.id.settingsButton);  
    saveBtn = findViewById(R.id.saveButton);  
    searchBtn = findViewById(R.id.searchRecordBtn);  
    calBtn = findViewById(R.id.calBtn);  
    timer = new Timer();  
    dbHelper = new DBHelper( context: this);  
    dbHelper.getWritableDatabase();  
    String dateText = new SimpleDateFormat( pattern: "MMM dd, yyyy", Locale.getDefault()).format(new Date());  
    dateTV = findViewById(R.id.dateText);  
    dateTV.setText(dateText);  
}  
  
// Function which calculates total calories burned during the session  
public int calCalories (String weightN , String heightN){  
    int cals = 0;  
    weight = Integer.parseInt(weightN);  
    height = Integer.parseInt(heightN);  
    float heightInMtrs = height / 10;  
    int mins = getMins();  
    cals = (int) ((int) (0.035 * weight) + ((velocity*velocity) / heightInMtrs) * (0.029) * (weight));  
}
```



```
    cals = cals * mins;
    return cals;
}

// Function to calculate total distance walked from the number of steps
public float calDistance() {
    float heightInches = (float) (height * 0.393701);
    float strideLen = (float) (heightInches * 0.435);
    float dis = strideLen * stepCount;
    float miles = dis / 63360;
    float km = (float) (miles * 1.60934);
    return km;
}

// Function that resets session variables in the reset button has been pressed
public void resetPressed(View view) {
    AlertDialog.Builder resetAlert = new AlertDialog.Builder( context: this);
    resetAlert.setTitle("Reset Timer");
    resetAlert.setMessage("Are you sure you want to reset?");
    resetAlert.setPositiveButton( text: "Reset", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialogInterface, int i) {
            if (timerTask != null) {
                timerTask.cancel();
                buttonUI( start: "Start", R.color.green);
                time = 0.0;
                timerStarted = false;
                textViewTimer.setText(formatTime( seconds: 0, minutes: 0, hours: 0));
                stepCount = 0;
                caloriesBurned = 0;
            }
        }
    });
}
```

```
        distanceWalked = 0.0f;
        textViewDistance.setText(String.valueOf("Distance Walked"));
        textViewCals.setText(String.valueOf("Calories Burned"));
        textViewStepCounter.setText(String.valueOf("Steps: 0"));
        textViewTarget.setText(String.valueOf("Target: "));
    }
}
});
resetAlert.setNeutralButton(text: "Cancel", new DialogInterface.OnClickListener() {
    @Override
    public void onClick(DialogInterface dialogInterface, int i) {
    }
});
resetAlert.show();
}
// Function to check whether Stop / Start Button has been pressed
// sets text and color of button depending on status
public void startStopPressed(View view) {
    if (!timerStarted) {
        timerStarted = true;
        buttonUI(start: "Stop", R.color.red);
        startTimer();
        boolean reset = false;
    } else {
        timerStarted = false;
        buttonUI(start: "Start", R.color.green);
        timerTask.cancel();
    }
}
}
```

```
//Function to update button UI depending on whether it is started or stopped
private void buttonUI(String start, int color) {
    stopStartBtn.setText(start);
    stopStartBtn.setTextColor(ContextCompat.getColor(context: this, color));
}

//Function to start timer, increments timer and updates timer text every second
private void startTimer() {
    timerTask = new TimerTask() {
        @Override
        public void run() {
            runOnUiThread(new Runnable() {
                @Override
                public void run() {
                    time++;
                    textViewTimer.setText(getTime());
                }
            });
        }
    };
};
timer.scheduleAtFixedRate(timerTask, delay: 0, period: 1000);
}
```

```
//Function to return total time of session
private String getTime() {
    int rounded = (int) Math.round(time);
    int seconds = ((rounded % 86400) % 3600) % 60;
    int minutes = ((rounded % 86400) % 3600) / 60;
    int hours = ((rounded % 86400) / 3600);
    return formatTime(seconds, minutes, hours);
}

//Function to return mins to calculate calories burned
private int getMins() {
    int rounded = (int) Math.round(time);
    int minutes = ((rounded % 86400) % 3600) / 60;
    return minutes;
}

// Function to format time as Hours: Minutes: Seconds: to be displayed on app screen
private String formatTime(int seconds, int minutes, int hours) {
    return String.format("%02d", hours) + " : " + String.format("%02d", minutes) + " : "
        + String.format("%02d", seconds);
}
}
```

Settings Activity.

```
package com.example.stepc;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

/**
 * This work was completed under the GNU GENERAL PUBLIC LICENSE
 * Copyright (C) 2007 Free Software Foundation, Inc. <https://fsf.org/>
 * Everyone is permitted to copy and distribute verbatim copies
 * of this license document, but changing it is not allowed.
 *
 * Author Dylan Scott C00242050
 * This file contains my projects "Settings Activity" & is where the application starts
 */
public class SettingsActivity extends AppCompatActivity {

    EditText firstName, lastName, weight, height, target;
    String firstNameString, lastNameString, weightString, heightString, targetString;
    int weightInt, heightInt, targetInt;
    Button saveSet, backBtn, btButton;
    DBHelper DB;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_settings);
        firstName = (EditText) findViewById(R.id.cFirstName);
```

```
lastName = (EditText) findViewById(R.id.cLastName);
weight = (EditText) findViewById(R.id.cWeight);
height = (EditText) findViewById(R.id.cHeight);
target = (EditText) findViewById(R.id.cTarget);
saveSet = (Button) findViewById(R.id.saveSettingsButton);
backBtn = (Button) findViewById(R.id.settingsBackButton);
btButton = (Button) findViewById(R.id.settingsBTButton);
DB = new DBHelper( context: this);
DB.getWritableDatabase();

// Button to save session settings
saveSet.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent = new Intent( packageContext: SettingsActivity.this, SessionActivity.class);
        firstNameString = firstName.getText().toString();
        lastNameString = lastName.getText().toString();
        weightString= weight.getText().toString();
        weightInt = Integer.parseInt(weightString);
        heightString = height.getText().toString();
        heightInt = Integer.parseInt(heightString);
        targetString = target.getText().toString();
        targetInt = Integer.parseInt(targetString);
        intent.putExtra( name: "firstname" , firstNameString);
        intent.putExtra( name: "lastname" , lastNameString);
        intent.putExtra( name: "weight", weightString);
        intent.putExtra( name: "height", heightString);
        intent.putExtra( name: "target", targetString);
        startActivity(intent);
    }
});
```

```
});

// Button to open session activity
backBtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        // Move to adapter list
        Intent intent = new Intent( packageContext: SettingsActivity.this, SessionActivity.class);
        startActivity(intent);
    }
});

// Button to open bt activity to select Bluetooth Device
btButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent = new Intent( packageContext: SettingsActivity.this, btActivity.class);
        startActivity(intent);
    }
});
}
```

btActivity.

```
package com.example.stepc;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import android.bluetooth.BluetoothAdapter;
import android.bluetooth.BluetoothDevice;
import android.bluetooth.BluetoothSocket;
import android.content.Intent;
import android.os.Bundle;
import android.os.Handler;
import android.os.Looper;
import android.os.Message;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.ProgressBar;
import android.widget.Toast;
import java.io.IOException;
import java.io.InputStream;
import java.io.OutputStream;
import java.util.UUID;
import static android.content.ContentValues.TAG;
/**
 * This work was completed under the GNU GENERAL PUBLIC LICENSE
 * Copyright (C) 2007 Free Software Foundation, Inc. <https://fsf.org/>
 * Everyone is permitted to copy and distribute verbatim copies
 * of this license document, but changing it is not allowed.
 * Author Dylan Scott C00242050
 * This file contains my projects "Session Activity"
 */
```

```
public class btActivity extends AppCompatActivity{

    private String deviceName = null;
    public static Handler handler;
    public static BluetoothSocket socket;
    public static ConnectedThread connectedThread;
    public static CreateConnectThread createConnectThread;
    private final static int CONNECTION_STATUS = 1;
    private Boolean btEnabled;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        final Button buttonConnect = findViewById(R.id.buttonConnect);
        final Button toggleConnect = findViewById(R.id.buttonToggleOff);
        final Button close = findViewById(R.id.closeApp);
        final Button buttonStartSession = findViewById(R.id.buttonSettings);
        final Toolbar toolbar = findViewById(R.id.toolbar);
        final ProgressBar connectionProgress = findViewById(R.id.progressBar);
        connectionProgress.setVisibility(View.GONE);

        deviceName = getIntent().getStringExtra( name: "deviceName");
        if (deviceName != null){

            // Get the MAC address to make BT Connection
            String MACAddress = getIntent().getStringExtra( name: "deviceAddress");

            toolbar.setSubtitle("Connecting to " + deviceName + "...");
            connectionProgress.setVisibility(View.VISIBLE);
            buttonConnect.setEnabled(false);
            BluetoothAdapter bluetoothAdapter = BluetoothAdapter.getDefaultAdapter();
            createConnectThread = new CreateConnectThread(bluetoothAdapter, MACAddress);
            createConnectThread.start();
        }

        // Handler to update UI depending on the message it receives
        handler = new Handler(Looper.getMainLooper()) {
            @Override
            public void handleMessage(Message msg){
                if (msg.what == CONNECTION_STATUS) {
                    switch (msg.arg1) {
                        case 1:
                            toolbar.setSubtitle("Connected to " + deviceName);
                            connectionProgress.setVisibility(View.GONE);
                            break;
                        case -1:
                            toolbar.setSubtitle("Unable to connect");
                            connectionProgress.setVisibility(View.GONE);
                            break;
                    }
                }
            }
        };
    }
};
```

```
// // Button that allows user select a bluetooth device to pair with when pressed
buttonConnect.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent = new Intent( packageContext: btActivity.this, SelectDeviceActivity.class);
        startActivity(intent);
    }
});

// Button that brings user to Session Activity when pressed
buttonStartSession.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent = new Intent( packageContext: btActivity.this, SessionActivity.class);
        startActivity(intent);
    }
});

// Button to toggle bluetooth
toggleConnect.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if (!btEnabled) {
            Toast.makeText( context: btActivity.this, text: "Bluetooth Enabled", Toast.LENGTH_SHORT)
                .show();
            btEnabled = true;
        }
    }
});
```

```
        }else {
            Toast.makeText( context: btActivity.this, text: "Bluetooth Disabled",
                Toast.LENGTH_SHORT).show();
            btEnabled = false;
        }
    }
});

// Button to close app
close.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent close = new Intent(getApplicationContext(), SettingsActivity.class);
        close.setFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
        close.putExtra( name: "EXIT", value: true);
        startActivity(close);
        finish();
        System.exit( status: 0);
    }
});
}
```



```
public static class CreateConnectThread extends Thread {
    public CreateConnectThread(BluetoothAdapter bluetoothAdapter, String address) {
        BluetoothDevice bluetoothDevice = bluetoothAdapter.getRemoteDevice(address);
        BluetoothSocket tmp = null;
        UUID uuid = bluetoothDevice.getUuids()[0].getUuid();
        try {
            tmp = bluetoothDevice.createInsecureRfcommSocketToServiceRecord(uuid);
        } catch (IOException e) {
            Log.e(TAG, "Socket's create() method failed", e);
        }
        socket = tmp;
    }
    public void run() {
        BluetoothAdapter bluetoothAdapter = BluetoothAdapter.getDefaultAdapter();
        bluetoothAdapter.cancelDiscovery();
        try {
            socket.connect();
            Log.e(tag: "Status", msg: "Connected");
            handler.obtainMessage(CONNECTION_STATUS, arg1: 1, arg2: -1).sendToTarget();
        } catch (IOException connectException) {
            try {
                socket.close();
                Log.e(tag: "Status", msg: "Cannot connect to device");
                handler.obtainMessage(CONNECTION_STATUS, arg1: -1, arg2: -1).sendToTarget();
            } catch (IOException closeException) {
                Log.e(TAG, msg: "Could not close the client socket", closeException);
            }
        }
        return;
    }
}
```

```
// New connected thread is created
connectedThread = new ConnectedThread(socket);
connectedThread.run();
}

// Closes the client socket and causes the thread to finish.
public void cancel() {
    try {
        socket.close();
    } catch (IOException e) {
        Log.e(TAG, msg: "Could not close", e);
    }
}
}
```

```
public static class ConnectedThread extends Thread {
    private final BluetoothSocket socket;
    public ConnectedThread(BluetoothSocket socket) {
        this.socket = socket;
        InputStream tempIn = null;
        OutputStream tempOut = null;
        try {
            tempIn = socket.getInputStream();
            tempOut = socket.getOutputStream();
        } catch (IOException ignored) {}
    }
    public void run() {

    }
    public void write(String input) {

    }
    // shuts down the connection
    public void cancel() {
        try {
            socket.close();
        } catch (IOException e) {}
    }
}
}
```

SelectDeviceActivity.

```
package com.example.stepc;
import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.DefaultItemAnimator;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;
import android.bluetooth.BluetoothAdapter;
import android.bluetooth.BluetoothDevice;
import android.os.Bundle;
import android.view.View;
import com.google.android.material.snackbar.Snackbar;
import java.util.ArrayList;
import java.util.List;
import java.util.Set;

/**
 * This work was completed under the GNU GENERAL PUBLIC LICENSE
 * Copyright (C) 2007 Free Software Foundation, Inc. <https://fsf.org/>
 * Everyone is permitted to copy and distribute verbatim copies
 * of this license document, but changing it is not allowed.
 *
 * Author Dylan Scott 000242050
 * This file contains my projects "Select Device Activity"
 */
```

```
public class SelectDeviceActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_select_device);

        BluetoothAdapter bluetoothAdapter = BluetoothAdapter.getDefaultAdapter();
        Set<BluetoothDevice> pairedDevices = bluetoothAdapter.getBondedDevices();
        List<Object> deviceList = new ArrayList<>();
        if (pairedDevices.size() > 0) {
            // Get the name and address of each paired device.
            for (BluetoothDevice device : pairedDevices) {
                String deviceName = device.getName();
                String macAddress = device.getAddress(); // MAC address
                DeviceInfoModel deviceInfoModel = new DeviceInfoModel(deviceName, macAddress);
                deviceList.add(deviceInfoModel);
            }
        }
        RecyclerView recyclerView = findViewById(R.id.recyclerViewDevice);
        recyclerView.setLayoutManager(new LinearLayoutManager( context: this));
        DeviceListAdapter deviceListAdapter = new DeviceListAdapter( context: this, deviceList);
        recyclerView.setAdapter(deviceListAdapter);
        recyclerView.setItemAnimator(new DefaultItemAnimator());
    }
}
```

```
    } else {  
        View view = findViewById(R.id.recyclerViewDevice);  
        Snackbar snackbar = Snackbar.make(view, text: "Make sure device is within range",  
            Snackbar.LENGTH_INDEFINITE);  
        snackbar.setAction(text: "OK", new View.OnClickListener() {  
            @Override  
            public void onClick(View view) { }  
        });  
        snackbar.show();  
    }  
}  
}
```

DeviceListAdapter.

```
package com.example.stepc;
import android.content.Intent;
import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.LinearLayout;
import android.widget.TextView;
import androidx.recyclerview.widget.RecyclerView;
import java.util.List;

/**
 * This work was completed under the GNU GENERAL PUBLIC LICENSE
 * Copyright (C) 2007 Free Software Foundation, Inc. <https://fsf.org/>
 * Everyone is permitted to copy and distribute verbatim copies
 * of this license document, but changing it is not allowed.
 *
 * Author Dylan Scott C00242050
 * This file contains my projects DeviceListAdapter clasee
 */

public class DeviceListAdapter extends RecyclerView.Adapter<RecyclerView.ViewHolder> {

    private Context context;
    private List<Object> deviceList;

    public static class ViewHolder extends RecyclerView.ViewHolder {
        TextView textViewName, textViewAddress;
        LinearLayout linearLayout;

        public ViewHolder(View v) {
            super(v);
            textViewName = v.findViewById(R.id.textViewDeviceName);
            textViewAddress = v.findViewById(R.id.textViewDeviceAddress);
            linearLayout = v.findViewById(R.id.linearLayoutDeviceInfo);
        }
    }

    public DeviceListAdapter(Context context, List<Object> deviceList) {
        this.context = context;
        this.deviceList = deviceList;
    }
}
```

```
public RecyclerView.ViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {
    View v = LayoutInflater.from(parent.getContext()).inflate(R.layout.device_info_layout,
        parent, attachToRoot: false);
    ViewHolder vh = new ViewHolder(v);
    return vh;
}

@Override
public void onBindViewHolder(final RecyclerView.ViewHolder holder, final int position) {
    ViewHolder itemHolder = (ViewHolder) holder;
    final DeviceInfoModel deviceInfoModel = (DeviceInfoModel) deviceList.get(position);
    itemHolder.textViewName.setText(deviceInfoModel.getDeviceName());
    itemHolder.textViewAddress.setText(deviceInfoModel.getMacAddress());
    itemHolder.linearLayout.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent intent = new Intent(context, btActivity.class);
            intent.putExtra( name: "deviceName", deviceInfoModel.getDeviceName());
            intent.putExtra( name: "deviceAddress", deviceInfoModel.getMacAddress());
            context.startActivity(intent);
        }
    });
}

@Override
public int getItemCount() {
    int dataCount = deviceList.size();
    return dataCount;
}
}
```

DeviceInfoModel.

```
package com.example.stepc;

/**
 * This work was completed under the GNU GENERAL PUBLIC LICENSE
 * Copyright (C) 2007 Free Software Foundation, Inc. <https://fsf.org/>
 * Everyone is permitted to copy and distribute verbatim copies
 * of this license document, but changing it is not allowed.
 *
 * Author Dylan Scott C00242050
 * This file contains my projects DeviceInfoModel class
 */

public class DeviceInfoModel {

    private final String deviceName;
    private final String macAddress;

    public DeviceInfoModel(String deviceName, String deviceHardwareAddress){
        this.deviceName = deviceName;
        this.macAddress = deviceHardwareAddress;
    }

    public String getDeviceName(){return deviceName;}

    public String getMacAddress(){return macAddress;}

}
```

Reference Declaration.

The Bluetooth section of my project was aided by following the following tutorial.

<https://medium.com/swlh/create-custom-android-app-to-control-arduino-board-using-bluetooth-ff878e998aa8>

dbHelper.

```
package com.example.stepc;
import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

/**
 * This work was completed under the GNU GENERAL PUBLIC LICENSE
 * Copyright (C) 2007 Free Software Foundation, Inc. <https://fsf.org/>
 * Everyone is permitted to copy and distribute verbatim copies
 * of this license document, but changing it is not allowed.
 *
 * Author Dylan Scott 000242050
 * This file contains my projects dbHelper class
 */

public class DBHelper extends SQLiteOpenHelper {

    public static final String DBNAME = "project.db";

    public DBHelper(Context context) {
        super(context, DBNAME, factory: null, version: 1);
    }

    @Override
    public void onCreate(SQLiteDatabase MyDB) {
        MyDB.execSQL("create Table sessionTable(id INTEGER PRIMARY KEY, fname TEXT, " +
            "lname TEXT, steps INTEGER, caloriesburned INTEGER, distancewalked DOUBLE)");
    }

    @Override
    public void onUpgrade(SQLiteDatabase MyDB, int i, int i1) {
        ;
        MyDB.execSQL("drop Table if exists sessionTable");
        onCreate(MyDB);
    }
}
```



```
public Boolean insertSessionData(String firstName, String lastName, int steps,
                                int caloriesburned, double distancewalked){
    SQLiteDatabase MyDB = this.getWritableDatabase();
    ContentValues userValues = new ContentValues();
    userValues.put("fname", firstName);
    userValues.put("lname", lastName);
    userValues.put("steps", steps);
    userValues.put("caloriesburned", caloriesburned);
    userValues.put("distancewalked", distancewalked);
    long result = MyDB.insert( table: "sessionTable", nullColumnHack: null, userValues);
    if (result == -1) return false;
    else
        return true;
}

public Cursor getAllData(String f , String l){
    SQLiteDatabase MyDB = this.getWritableDatabase();
    Cursor cursor = MyDB.rawQuery( sql: "Select * from sessionTable where fname = '"+f+"' " +
        "and lname = '"+l+"' " , selectionArgs: null);
    return cursor;
}

public Cursor getAvg(String f , String l) {
    SQLiteDatabase MyDB = this.getWritableDatabase();
    Cursor cursor = MyDB.rawQuery( sql: "Select * from sessionTable where fname = '"+f+"' " +
        "and lname = '"+l+"' " , selectionArgs: null);
    return cursor;
}
}
```

ViewDataActivity.

```
package com.example.steps;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.recyclerview.widget.LinearLayoutManager;  
import androidx.recyclerview.widget.RecyclerView;  
import android.app.Dialog;  
import android.content.Intent;  
import android.database.Cursor;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.TextView;  
import java.util.ArrayList;  
import java.util.List;  
  
/**  
 * This work was completed under the GNU GENERAL PUBLIC LICENSE  
 * Copyright (C) 2007 Free Software Foundation, Inc. <https://fsf.org/>  
 * Everyone is permitted to copy and distribute verbatim copies  
 * of this license document, but changing it is not allowed.  
 *  
 * Author Dylan Scott C00242050  
 * This file contains my projects "View Data Activity"  
 */
```

```
public class ViewDataActivity extends AppCompatActivity {

    private EditText firstName, lastName;
    private Button btnSearch, btnAvg, btnGraph;
    String firstNameString, lastNameString;
    private RecyclerView rvSearchData;
    private DBHelper dbHelper;
    private TextView noRecordFound;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_view_data);
        init();

        btnAvg.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                String first, last;
                int avg;
                first = firstName.getText().toString().trim();
                last = lastName.getText().toString().trim();
                Cursor cursor = dbHelper.getAvg(first, last);
                List<SessionModel> strList = new ArrayList<>();
```

```
if (cursor.moveToNext()) {
    do {
        String step = cursor.getString( columnIndex: 3);
        String cals = cursor.getString( columnIndex: 4);
        String distance = cursor.getString( columnIndex: 5);
        SessionModel sessionModel =
            new SessionModel(Integer.parseInt(step),
                Integer.parseInt(cals), Double.parseDouble(distance));
        strList.add(sessionModel);
    } while (cursor.moveToNext());
}
double totalSteps = 0, totalCals = 0, totalDis = 0;
for (int i = 0; i < strList.size(); i++) {
    totalSteps = totalSteps + strList.get(i).getSteps();
    totalCals = totalCals + strList.get(i).getCalories();
    totalDis = totalDis + strList.get(i).getDistance();
}
double avgSteps, avgCals, avgDis;
avgSteps = totalSteps / strList.size();
avgCals = totalCals / strList.size();
avgDis = totalDis / strList.size();

final Dialog dialog = new Dialog( context: ViewDataActivity.this);
dialog.setContent(R.layout.dialog);

TextView avgStp = (TextView) dialog.findViewById(R.id.avgSteps);
TextView avgCal = (TextView) dialog.findViewById(R.id.avgCalories);
TextView avgDistance = (TextView) dialog.findViewById(R.id.avgDistance);
```

```
avgStp.setText("Average steps : " + avgSteps);
avgCal.setText("Average calories burned : " + avgCals);
avgDistance.setText("Average distance : " + avgDis);
Button okButton = (Button) dialog.findViewById(R.id.btnOK);
okButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        dialog.dismiss();
    }
});
dialog.show();
});
```

```
btnSearch.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        String first, last;
        first = firstName.getText().toString().trim();
        last = lastName.getText().toString().trim();
        Cursor cursor = dbHelper.getAllData(first, last);
        List<SessionModel> strList = new ArrayList<>();
        if (cursor.moveToNext()) {
            do {
                String step = cursor.getString( columnIndex: 3);
                String cal = cursor.getString( columnIndex: 4);
                String distance = cursor.getString( columnIndex: 5);
                SessionModel sessionModel =
                    new SessionModel(Integer.parseInt(step),
                        Integer.parseInt(cal), Double.parseDouble(distance));
                strList.add(sessionModel);
            } while (cursor.moveToNext());
        }
        if (strList.isEmpty() || strList.size() == 0) {
            noRecordFound.setVisibility(View.VISIBLE);
            rvSearchData.setVisibility(View.GONE);
        }
    }
});
```

```
        else {
            noRecordFound.setVisibility(View.GONE);
            rvSearchData.setVisibility(View.VISIBLE);
            SearchDataAdapter searchDataAdapter = new SearchDataAdapter( context: ViewDataActivity.this,
                strList);
            rvSearchData.setAdapter(searchDataAdapter);
        }
    }
});

// Buttons which brings user to Line Chart Activity
btnGraph.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent = new Intent( packageContext: ViewDataActivity.this, LineChartActivity.class);
        firstNameString = firstName.getText().toString();
        lastNameString = lastName.getText().toString();
        intent.putExtra( name: "firstname" , firstNameString);
        intent.putExtra( name: "lastname" , lastNameString);
        startActivity(intent);
    }
});
}
```

```
public void init() {
    dbHelper = new DBHelper( context: this);
    dbHelper.getWritableDatabase();
    firstName = findViewById(R.id.firstNameSearch);
    noRecordFound = findViewById(R.id.norecordFound);
    lastName = findViewById(R.id.lastNameSearch);
    btnSearch = findViewById(R.id.searchData);
    btnAvg = findViewById(R.id.btnBChart);
    btnGraph = findViewById(R.id.graphData);
    rvSearchData = findViewById(R.id.recyclerViewSearchData);
    rvSearchData.setLayoutManager(new LinearLayoutManager( context: this));
}
}
```

BarChartActivity.

```
package com.example.stepc;
import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;
import android.app.Dialog;
import android.content.Intent;
import android.database.Cursor;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import java.util.ArrayList;
import java.util.List;

/**
 * This work was completed under the GNU GENERAL PUBLIC LICENSE
 * Copyright (C) 2007 Free Software Foundation, Inc. <https://fsf.org/>
 * Everyone is permitted to copy and distribute verbatim copies
 * of this license document, but changing it is not allowed.
 *
 * Author Dylan Scott C00242050
 * This file contains my projects "View Data Activity"
 */

public class ViewDataActivity extends AppCompatActivity {

    private EditText firstName, lastName;
    private Button btnSearch, btnAvg, btnGraph;
    String firstNameString, lastNameString;
    private RecyclerView rvSearchData;
    private DBHelper dbHelper;
    private TextView noRecordFound;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_view_data);
        init();

        btnAvg.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                String first, last;
                int avg;
                first = firstName.getText().toString().trim();
                last = lastName.getText().toString().trim();
                Cursor cursor = dbHelper.getAvg(first, last);
                List<SessionModel> strList = new ArrayList<>();
            }
        });
    }
}
```

```

if (cursor.moveToNext()) {
    do {
        String step = cursor.getString( columnIndex: 3);
        String cals = cursor.getString( columnIndex: 4);
        String distance = cursor.getString( columnIndex: 5);
        SessionModel sessionModel =
            new SessionModel(Integer.parseInt(step),
                Integer.parseInt(cals), Double.parseDouble(distance));
        strList.add(sessionModel);
    } while (cursor.moveToNext());
}
double totalSteps = 0, totalCals = 0, totalDis = 0;
for (int i = 0; i < strList.size(); i++) {
    totalSteps = totalSteps + strList.get(i).getSteps();
    totalCals = totalCals + strList.get(i).getCalories();
    totalDis = totalDis + strList.get(i).getDistance();
}
double avgSteps, avgCals, avgDis;
avgSteps = totalSteps / strList.size();
avgCals = totalCals / strList.size();
avgDis = totalDis / strList.size();

final Dialog dialog = new Dialog( context: ViewDataActivity.this);
dialog.setContent(R.layout.dialog);

TextView avgStp = (TextView) dialog.findViewById(R.id.avgSteps);
TextView avgCal = (TextView) dialog.findViewById(R.id.avgCalories);
TextView avgDistance = (TextView) dialog.findViewById(R.id.avgDistance);

```

```

avgStp.setText("Average steps : " + avgSteps);
avgCal.setText("Average calories burned : " + avgCals);
avgDistance.setText("Average distance : " + avgDis);
Button okButton = (Button) dialog.findViewById(R.id.btnOK);
okButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        dialog.dismiss();
    }
});
dialog.show();
});

```

```
btnSearch.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        String first, last;
        first = firstName.getText().toString().trim();
        last = lastName.getText().toString().trim();
        Cursor cursor = dbHelper.getAllData(first, last);
        List<SessionModel> strList = new ArrayList<>();
        if (cursor.moveToNext()) {
            do {
                String step = cursor.getString( columnIndex: 3);
                String cal = cursor.getString( columnIndex: 4);
                String distance = cursor.getString( columnIndex: 5);
                SessionModel sessionModel =
                    new SessionModel(Integer.parseInt(step),
                                      Integer.parseInt(cal), Double.parseDouble(distance));
                strList.add(sessionModel);
            } while (cursor.moveToNext());
        }
        if (strList.isEmpty() || strList.size() == 0) {
            noRecordFound.setVisibility(View.VISIBLE);
            rvSearchData.setVisibility(View.GONE);
        }
    }
});
```

```
        else {
            noRecordFound.setVisibility(View.GONE);
            rvSearchData.setVisibility(View.VISIBLE);
            SearchDataAdapter searchDataAdapter = new SearchDataAdapter( context: ViewDataActivity.this,
                                                                           strList);
            rvSearchData.setAdapter(searchDataAdapter);
        }
    }
});

// Buttons which brings user to Line Chart Activity
btnGraph.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent = new Intent( packageContext: ViewDataActivity.this, LineChartActivity.class);
        firstNameString = firstName.getText().toString();
        lastNameString = lastName.getText().toString();
        intent.putExtra( name: "firstname" , firstNameString);
        intent.putExtra( name: "lastname" , lastNameString);
        startActivity(intent);
    }
});
}
```



```
public void init() {
    dbHelper = new DBHelper( context: this);
    dbHelper.getWritableDatabase();
    firstName = findViewById(R.id.firstNameSearch);
    noRecordFound = findViewById(R.id.norecordFound);
    lastName = findViewById(R.id.lastNameSearch);
    btnSearch = findViewById(R.id.searchData);
    btnAvg = findViewById(R.id.btnBChart);
    btnGraph = findViewById(R.id.graphData);
    rvSearchData = findViewById(R.id.recyclerViewSearchData);
    rvSearchData.setLayoutManager(new LinearLayoutManager( context: this));
}
}
```

LineChartActivity.

```
package com.example.stepc;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.database.Cursor;
import android.graphics.Color;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import com.androidplot.xy.BoundaryMode;
import com.androidplot.xy.LineAndPointFormatter;
import com.androidplot.xy.SimpleXYSeries;
import com.androidplot.xy.StepMode;
import com.androidplot.xy.XYPlot;
import com.androidplot.xy.XYSeries;
import java.util.ArrayList;
import java.util.Arrays;
import java.util.List;

/**
 * This work was completed under the GNU GENERAL PUBLIC LICENSE
 * Copyright (C) 2007 Free Software Foundation, Inc. <https://fsf.org/>
 * Everyone is permitted to copy and distribute verbatim copies
 * of this license document, but changing it is not allowed.
 *
 * Author Dylan Scott C00242050
 * This file contains my projects "Line Chart Activity"
 */
```

```
public class LineChartActivity extends AppCompatActivity {

    private EditText firstName, lastName;
    private Button btnSearch, btnBarChart, btnGraph;
    private DBHelper dbHelper;
    private XYPlot lineChart;
    private String fName;
    private String lName;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_line_chart);
        init();

        if (getIntent().hasExtra("firstname")) {
            fName = getIntent().getStringExtra("firstname");
            lName = getIntent().getStringExtra("lastname");
            firstName.setText(String.valueOf(fName));
            lastName.setText(String.valueOf(lName));
        }

        Cursor cursor = dbHelper.getAllData(fName, lName);
        List<String> yAxisVals = new ArrayList<>();
        if (cursor.moveToNext()) {
            do {
                String data = cursor.getString(columnIndex: 3);
                yAxisVals.add(data);
            } while (cursor.moveToNext());
        }
    }
}
```

```
Cursor cursor2 = dbHelper.getAllData(fName, lName);
List<Integer> xValsLen = new ArrayList<>();
int i = 1;
if (cursor2.moveToNext()) {
    do {
        xValsLen.add(i);
        i++;
    } while (cursor2.moveToNext());
}

Number[] yVals = convert(yAxisVals);
Number[] xVals = convertToLen(xValsLen);
XYSeries series = new SimpleXYSeries(Arrays.asList(xVals), Arrays.asList(yVals), title: "Line Graph");
LineAndPointFormatter seriesFormat = new LineAndPointFormatter(Color.RED, Color.BLUE,
    fillColor: null, plf: null);
```

```
lineChart.addSeries(series, seriesFormat);
lineChart.setRangeStep(StepMode.INCREMENT_BY_VAL, value: 400);
lineChart.setDomainStep(StepMode.INCREMENT_BY_VAL, value: 1);
lineChart.setRangeBoundaries( lowerBoundary: 0 , upperBoundary: 4000, BoundaryMode.FIXED );
lineChart.setDomainBoundaries( lowerBoundary: 0 , upperBoundary: yAxisVals.size() + 1, BoundaryMode.FIXED);

// Button to open Bar Chart Activity
btnBarChart.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {

        Intent intent = new Intent( packageContext: LineChartActivity.this, BarChartActivity.class);
        intent.putExtra( name: "firstname" , fName);
        intent.putExtra( name: "lastname" , lName);
        startActivity(intent);
    }
});
```

```
// Button to open View Data Activity
btnSearch.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent = new Intent( packageContext: LineChartActivity.this, ViewDataActivity.class);
        startActivity(intent);
    }
});

}

public static Number[] convert(List<String> strList)
{
    Number[] stepsArray= new Number[strList.size()];
    for(int i=0;i<strList.size();i++) {
        stepsArray[i]= Integer.parseInt(strList.get(i));
    }
    return stepsArray;
}

public static Number[] convertToLen(List<Integer> strList)
{
    Number[] xValAxisLen = new Number[strList.size()];
    for(int i=0;i<strList.size();i++)
    {
        xValAxisLen[i]= Integer.parseInt(String.valueOf(strList.get(i)));
    }
    return xValAxisLen;
}
}
```

```
public void init() {
    dbHelper = new DBHelper( context: this);
    dbHelper.getWritableDatabase();
    firstName = findViewById(R.id.firstNameSearch);
    lastName = findViewById(R.id.lastNameSearch);
    btnSearch = findViewById(R.id.searchData);
    btnBarChart = findViewById(R.id.btnBChart);
    btnGraph = findViewById(R.id.graphData);
    lineChart = findViewById(R.id.xyplot);
}
}
```

SessionModel.

```
package com.example.steps;  
  
/**  
 * This work was completed under the GNU GENERAL PUBLIC LICENSE  
 * Copyright (C) 2007 Free Software Foundation, Inc. <https://fsf.org/>  
 * Everyone is permitted to copy and distribute verbatim copies  
 * of this license document, but changing it is not allowed.  
 *  
 * Author Dylan Scott C00242050  
 * This file contains my projects SessionModel class  
 */  
  
public class SessionModel {  
    private int steps;  
    private int calories;  
    private double distance;
```

```
    public SessionModel(int steps, int calories, double distance) {  
        this.steps = steps;  
        this.calories = calories;  
        this.distance = distance;  
    }  
  
    public void setCalories(int calories) {  
        this.calories = calories;  
    }  
  
    public void setDistance(double distance) {  
        this.distance = distance;  
    }  
  
    public void setSteps(int steps) {  
        this.steps = steps;  
    }  
  
    public double getDistance() {  
        return distance;  
    }  
  
    public int getCalories() {  
        return calories;  
    }  
  
    public int getSteps() {  
        return steps;  
    }  
}
```

SearchDataAdapter.

```
package com.example.stepc;
import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.TextView;
import java.util.List;
import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

/**
 * This work was completed under the GNU GENERAL PUBLIC LICENSE
 * Copyright (C) 2007 Free Software Foundation, Inc. <https://fsf.org/>
 * Everyone is permitted to copy and distribute verbatim copies
 * of this license document, but changing it is not allowed.
 *
 * Author Dylan Scott C00242050
 * This file contains my projects SearchDataAdapter class
 */

public class SearchDataAdapter extends RecyclerView.Adapter<SearchDataAdapter.MyViewHolder> {
    private Context context;
    private List<SessionModel> list;

    public SearchDataAdapter(Context context, List<SessionModel> list) {
        this.context = context;
        this.list = list;
    }

    @NonNull
    @Override
    public MyViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
        View view = LayoutInflater.from(context).inflate(R.layout.search_layout, parent, attachToRoot: false);
        return new MyViewHolder(view);
    }

    @Override
    public void onBindViewHolder(@NonNull MyViewHolder holder, int position) {
        holder.txtSteps.setText("Steps: " + list.get(position).getSteps());
        holder.txtCal.setText("Calories : " + list.get(position).getCalories());
        holder.txtDistance.setText("Distance: " + list.get(position).getDistance());
    }

    @Override
    public int getItemCount() {
        return list.size();
    }

    public class MyViewHolder extends RecyclerView.ViewHolder {
        private TextView txtSteps, txtCal, txtDistance;

        public MyViewHolder(@NonNull View itemView) {
            super(itemView);
            txtSteps = itemView.findViewById(R.id.txtStep);
            txtCal = itemView.findViewById(R.id.txtCal);
            txtDistance = itemView.findViewById(R.id.txtDistance);
        }
    }
}
```

SplashScreenActivity.

```
package com.example.stepc;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.os.Handler;

/**
 * This work was completed under the GNU GENERAL PUBLIC LICENSE
 * Copyright (C) 2007 Free Software Foundation, Inc. <https://fsf.org/>
 * Everyone is permitted to copy and distribute verbatim copies
 * of this license document, but changing it is not allowed.
 *
 * Author Dylan Scott C00242050
 * This file contains my projects "Splash Screen Activity"
 */

public class SplashScreenActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_splash_screen);

        getSupportActionBar().hide();
        new Handler().postDelayed(new Runnable(){
            @Override
            public void run() {
                Intent intent = new Intent( packageContext: SplashScreenActivity.this, SettingsActivity.class);
                startActivity(intent);
                finish();
            }
        }, delayMillis: 3500);
    }
}
```

XML.

activity_session.

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    tools:context=".SessionActivity">

    <androidx.appcompat.widget.Toolbar
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@color/grey" />

    <com.google.android.material.button.MaterialButton
        app:cornerRadius="28dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:background="@null"
        android:onClick="resetPressed"
        android:text="Reset"
        android:textAllCaps="false"
        android:layout_gravity="start"
        android:textColor="@color/red"
        android:textSize="13sp" />
```

```
<TextView
    android:id="@+id/timeText"
    android:layout_width="150dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="137dp"
    android:layout_marginTop="5dp"
    android:layout_centerHorizontal="true"
    android:layout_marginEnd="20dp"
    android:layout_marginBottom="1dp"
    android:gravity="center"
    android:text="00 : 00 : 00"
    android:textColor="@color/black"
    android:textSize="20sp" />

<com.google.android.material.button.MaterialButton
    app:cornerRadius="28dp"
    android:id="@+id/startStopButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="end"
    android:layout_marginStart="1dp"
    android:layout_toEndOf="@id/timeText"
    android:background="@null"
    android:onClick="startStopPressed"
    android:text="Start"
    android:textAllCaps="false"
    android:textColor="@color/green"
    android:textSize="13sp" />
```

```
<TextView
    android:id="@+id/dateText"
    android:layout_width="322dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="50dp"
    android:layout_marginTop="70dp"
    android:layout_marginEnd="50dp"
    android:layout_marginBottom="5dp"
    android:gravity="center"
    android:textSize="25sp"
    tools:ignore="MissingConstraints" />
```

```
<TextView
    android:id="@+id/welcomeText"
    android:layout_width="322dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="50dp"
    android:layout_marginTop="7dp"
    android:layout_marginEnd="50dp"
    android:layout_marginBottom="5dp"
    android:layout_below="@id/dateText"
    android:gravity="center"
    android:text="Welcome"
    android:textSize="25sp"
    tools:ignore="MissingConstraints" />
```

```
<TextView
    android:id="@+id/targetText"
    android:layout_width="150dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="5dp"
    android:layout_marginTop="50dp"
    android:layout_marginEnd="50dp"
    android:layout_marginBottom="5dp"
    android:layout_below="@+id/welcomeText"
    android:layout_toRightOf="@id/calorieText"
    android:gravity="center"
    android:text="Target"
    android:textSize="25sp"
    tools:ignore="MissingConstraints" />
```

```
<TextView
    android:id="@+id/stepCounterText"
    android:layout_width="150dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="5dp"
    android:layout_marginTop="50dp"
    android:layout_marginEnd="50dp"
    android:layout_marginBottom="5dp"
    android:gravity="center"
    android:layout_below="@+id/welcomeText"
    android:text="Step Count"
    android:textSize="25sp"
    tools:ignore="MissingConstraints" />
```



```
<TextView
    android:id="@+id/calorieText"
    android:layout_width="150dp"
    android:layout_height="wrap_content"
    android:layout_below="@+id/stepCounterText"
    android:layout_marginStart="5dp"
    android:layout_marginTop="50dp"
    android:layout_marginEnd="50dp"
    android:layout_marginBottom="50dp"
    android:gravity="center"
    android:text="Calories:"
    android:textSize="25sp"
    tools:ignore="MissingConstraints" />
```

```
<TextView
    android:id="@+id/distanceText"
    android:layout_width="150dp"
    android:layout_height="wrap_content"
    android:layout_below="@id/stepCounterText"
    android:layout_toRightOf="@id/calorieText"
    android:layout_marginStart="1dp"
    android:layout_marginTop="50dp"
    android:layout_marginEnd="5dp"
    android:layout_marginBottom="50dp"
    android:gravity="center"
    android:text="Distance:"
    android:textSize="25sp"
    tools:ignore="MissingConstraints" />
```

```
<com.google.android.material.button.MaterialButton
    app:cornerRadius="28dp"
    android:id="@+id/settingsButton"
    android:layout_width="160dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="30dp"
    android:layout_below="@+id/distanceText"
    android:layout_marginTop="17dp"
    android:layout_marginBottom="20dp"
    android:layout_marginEnd="5dp"
    app:icon="@drawable/ic_baseline_settings_24"
    app:iconGravity="textStart"
    app:iconPadding="0dp"
    android:textAllCaps="false"
    android:textColor="@android:color/background_light"/>
```

```
<com.google.android.material.button.MaterialButton
    app:cornerRadius="28dp"
    android:id="@+id/saveButton"
    android:layout_width="160dp"
    android:layout_height="wrap_content"
    android:layout_marginLeft="5dp"
    android:layout_below="@+id/distanceText"
    android:layout_marginEnd="5dp"
    android:layout_marginTop="17dp"
    android:layout_marginBottom="20dp"
    android:layout_toRightOf="@+id/settingsButton"
    app:icon="@drawable/ic_baseline_save_alt_24"
    app:iconGravity="textStart"
    app:iconPadding="0dp"
    android:textAllCaps="false"
    android:textColor="@android:color/background_light" />
```

```
<com.google.android.material.button.MaterialButton
    app:cornerRadius="28dp"
    android:id="@+id/searchRecordBtn"
    android:layout_width="160dp"
    android:layout_height="wrap_content"
    android:layout_marginEnd="5dp"
    android:layout_marginTop="17dp"
    android:layout_below="@+id/settingsButton"
    android:layout_marginStart="30dp"
    app:icon="@drawable/ic_baseline_search_24"
    app:iconGravity="textStart"
    app:iconPadding="0dp"
    android:textAllCaps="false" />
```

```
<com.google.android.material.button.MaterialButton
    app:cornerRadius="28dp"
    android:id="@+id/calBtn"
    android:layout_marginEnd="5dp"
    android:layout_marginTop="17dp"
    android:layout_width="160dp"
    android:layout_height="wrap_content"
    android:layout_marginLeft="50dp"
    android:layout_below="@id/saveButton"
    android:layout_toRightOf="@+id/searchRecordBtn"
    app:icon="@drawable/ic_baseline_calculate_24"
    app:iconGravity="textStart"
    app:iconPadding="0dp"
    android:textAllCaps="false" />
</RelativeLayout>
```

activity_settings.

```
<?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"
    tools:context=".SettingsActivity">

    <ImageView
        android:layout_width="600dp"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_centerHorizontal="true"
        android:adjustViewBounds="true"
        android:scaleType="fitCenter"
        android:contentDescription="TODO" />

    <TextView
        android:id="@+id/settings"
        android:layout_width="322dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="50dp"
        android:layout_marginTop="50dp"
        android:layout_marginEnd="50dp"
        android:layout_marginBottom="50dp"
        android:gravity="center"
        android:text="Update Settings"
        android:textSize="25dp" />
```

```
<EditText
    android:id="@+id/cFirstName"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/settings"
    android:layout_marginStart="10dp"
    android:layout_marginTop="1dp"
    android:layout_marginEnd="1dp"
    android:layout_marginBottom="1dp"
    android:drawablePadding="2dp"
    android:hint="First Name"
    android:inputType="text"
    android:padding="20dp" />

<EditText
    android:id="@+id/cLastName"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/cFirstName"
    android:layout_marginStart="1dp"
    android:layout_marginTop="1dp"
    android:layout_marginEnd="1dp"
    android:layout_marginBottom="1dp"
    android:drawablePadding="2dp"
    android:hint="Second Name"
    android:padding="20dp"
    android:inputType="text" />
```

```
<EditText
    android:id="@+id/cWeight"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/cLastName"
    android:layout_marginStart="1dp"
    android:layout_marginTop="1dp"
    android:layout_marginEnd="1dp"
    android:layout_marginBottom="1dp"
    android:drawablePadding="2dp"
    android:hint="Weight in KG"
    android:inputType="number"
    android:padding="20dp" />
```

```
<EditText
    android:id="@+id/cHeight"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/cWeight"
    android:layout_marginStart="1dp"
    android:layout_marginTop="1dp"
    android:layout_marginEnd="1dp"
    android:layout_marginBottom="1dp"
    android:drawablePadding="2dp"
    android:hint="Height in CM"
    android:padding="20dp"
    android:inputType="number" />
```

```
<EditText
    android:id="@+id/cTarget"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/cHeight"
    android:layout_marginStart="1dp"
    android:layout_marginTop="1dp"
    android:layout_marginEnd="1dp"
    android:layout_marginBottom="1dp"
    android:drawablePadding="2dp"
    android:hint="Target Steps"
    android:padding="20dp"
    android:inputType="number" />
```

```
<com.google.android.material.button.MaterialButton
    app:cornerRadius="28dp"
    android:id="@+id/saveSettingsButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/cTarget"
    android:layout_marginStart="20dp"
    android:layout_marginLeft="41dp"
    android:layout_marginTop="20dp"
    android:layout_marginEnd="20dp"
    android:layout_marginRight="41dp"
    android:layout_marginBottom="20dp"
    app:icon="@drawable/ic_baseline_save_alt_24"
    app:iconGravity="textStart"
    app:iconPadding="0dp" />
```

```
<com.google.android.material.button.MaterialButton
    android:id="@+id/settingsBackButton"
    app:cornerRadius="28dp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/cTarget"
    android:layout_centerHorizontal="true"
    android:layout_marginStart="20dp"
    android:layout_marginLeft="41dp"
    android:layout_marginRight="41dp"
    android:layout_marginTop="20dp"
    android:layout_marginEnd="20dp"
    android:layout_marginBottom="20dp"
    android:layout_toRightOf="@id/saveSettingsButton"
    app:icon="@drawable/ic_baseline_home_24"
    app:iconGravity="textStart"
    app:iconPadding="0dp" />
```

```
<com.google.android.material.button.MaterialButton
    android:id="@+id/settingsBTButton"
    app:cornerRadius="28dp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/cTarget"
    android:layout_centerHorizontal="true"
    android:layout_marginStart="20dp"
    android:layout_marginTop="19dp"
    android:layout_marginEnd="20dp"
    android:layout_marginBottom="41dp"
    android:layout_toRightOf="@id/settingsBackButton"
    app:icon="@drawable/ic_sharp_bluetooth_24"
    app:iconGravity="textStart"
    app:iconPadding="0dp" />
```

```
</RelativeLayout>
```

activity_main.

```
<?xml version="1.0" encoding="utf-8"?>
<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:id="@+id/constraintLayout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@android:color/background_light"
    tools:context=".btActivity">

    <androidx.appcompat.widget.Toolbar
        android:id="@+id/toolbar"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="?attr/colorPrimary"
        android:minHeight="?attr/actionBarSize"
        android:theme="?attr/actionBarTheme"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:subtitleTextColor="@android:color/background_light"
        app:title="Bluetooth Connection"
        app:titleTextColor="@android:color/background_light" />
```

```
<ImageView
    android:id="@+id/imageView"
    android:layout_width="200dp"
    android:layout_height="300dp"
    android:layout_marginTop="24dp"
    android:contentDescription="Visual LED"
    android:background="@drawable/austracker"
    android:scaleType="centerCrop"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/toolbar"
    tools:ignore="VectorDrawableCompat" />

<com.google.android.material.button.MaterialButton
    app:cornerRadius="28dp"
    android:id="@+id/buttonConnect"
    style="@style/Widget.AppCompat.Button.Borderless"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Connect"
    android:textColor="@android:color/background_light"
    app:layout_constraintBottom_toBottomOf="@+id/toolbar"
    app:layout_constraintEnd_toEndOf="@+id/toolbar"
    app:layout_constraintTop_toTopOf="@+id/toolbar" />
```

```
<com.google.android.material.button.MaterialButton
    app:cornerRadius="28dp"
    android:id="@+id/buttonToggleOff"
    android:layout_width="200dp"
    android:layout_height="wrap_content"
    android:layout_marginTop="400dp"
    app:icon="@drawable/ic_sharp_bluetooth_24"
    app:iconGravity="textStart"
    app:iconPadding="0dp"
    android:textSize="24sp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
```

```
<com.google.android.material.button.MaterialButton
    android:id="@+id/closeApp"
    android:layout_width="200dp"
    android:layout_height="wrap_content"
    android:layout_below="@+id/buttonSettings"
    android:layout_marginTop="552dp"
    app:icon="@drawable/ic_baseline_close_24"
    app:iconGravity="textStart"
    app:iconPadding="0dp"
    android:textSize="24sp"
    app:cornerRadius="28dp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.502"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
```

```
<ProgressBar
    android:id="@+id/progressBar"
    style="?android:attr/progressBarStyle"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
```

```
<TextView
    android:id="@+id/textViewInfo"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="16dp"
    android:layout_marginTop="16dp"
    android:layout_marginEnd="16dp"
    android:textAlignment="center"
    android:textStyle="italic"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/closeApp" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

activity_select_device

```
<?xml version="1.0" encoding="utf-8"?>
<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=".SelectDeviceActivity">

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recyclerViewDevice"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```


device_info_layout

```
<?xml version="1.0" encoding="utf-8"?>
<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="wrap_content">

    <LinearLayout
        android:id="@+id/linearLayoutDeviceInfo"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginTop="16dp"
        android:layout_marginEnd="16dp"
        android:orientation="vertical"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent">
```

```
        <TextView
            android:id="@+id/textViewDeviceName"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:textSize="18sp"
            android:textStyle="bold" />
        <TextView
            android:id="@+id/textViewDeviceAddress"
            android:layout_width="match_parent"
            android:layout_height="wrap_content" />
    </LinearLayout>
```

```
    <View
        android:id="@+id/divider"
        android:layout_width="0dp"
        android:layout_height="1dp"
        android:layout_marginStart="16dp"
        android:layout_marginTop="16dp"
        android:layout_marginEnd="16dp"
        android:background="?android:attr/listDivider"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/linearLayoutDeviceInfo" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

activity_view_data.

```
<?xml version="1.0" encoding="utf-8"?>
<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=".ViewDataActivity">

    <RelativeLayout
        android:id="@+id/linearlayout"
        app:layout_constraintTop_toTopOf="parent"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="12dp"
        android:orientation="vertical"
        android:padding="12dp">

        <EditText
            android:id="@+id/firstNameSearch"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Enter First Name" />
```

```
<EditText
    android:id="@+id/lastNameSearch"
    android:layout_below="@+id/firstNameSearch"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Enter Second Name" />

<com.google.android.material.button.MaterialButton
    app:cornerRadius="28dp"
    android:id="@+id/searchData"
    android:layout_below="@+id/lastNameSearch"
    android:layout_width="100dp"
    android:layout_height="wrap_content"
    android:layout_marginEnd="20dp"
    app:icon="@drawable/ic_baseline_search_24"
    app:iconGravity="textStart"
    app:iconPadding="0dp"
    android:layout_marginTop="12dp"
    android:textAllCaps="false" />
```

```
<com.google.android.material.button.MaterialButton
    app:cornerRadius="28dp"
    android:id="@+id/btnBChart"
    android:layout_width="100dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="1dp"
    app:icon="@drawable/ic_baseline_average_24"
    app:iconGravity="textStart"
    app:iconPadding="0dp"
    android:layout_toRightOf="@+id/searchData"
    android:layout_below="@+id/lastNameSearch"
    android:layout_marginTop="12dp"
    android:textAllCaps="false"/>
```

```
<com.google.android.material.button.MaterialButton
    app:cornerRadius="28dp"
    android:id="@+id/graphData"
    android:layout_width="100dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="10dp"
    app:icon="@drawable/ic_baseline_insert_chart_24"
    app:iconGravity="textStart"
    app:iconPadding="0dp"
    android:layout_toRightOf="@+id/btnBChart"
    android:layout_below="@+id/lastNameSearch"
    android:layout_marginTop="12dp"
    android:textAllCaps="false"/>
```

```
</RelativeLayout>
```

```
<androidx.recyclerview.widget.RecyclerView
    android:id="@+id/recyclerViewSearchData"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    app:layout_constraintTop_toBottomOf="@id/linearlayout"
    android:layout_margin="12dp"/>
```

```
<TextView
    android:id="@+id/norecordFound"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="No Record Found !"
    android:visibility="gone"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    android:textSize="20sp"/>
```

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

search_result_layout.

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="5dp"
    android:padding="5dp">

    <TextView
        android:id="@+id/txtStep"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="steps" />

    <TextView
        android:id="@+id/txtCal"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="12dp"
        android:layout_toRightOf="@id/txtStep"
        android:text="Calories" />
```

```
    <TextView
        android:id="@+id/txtDistance"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="12dp"
        android:layout_toRightOf="@id/txtCal"
        android:text="Distance" />
</RelativeLayout>
```

dialog_layout.

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:layout_centerInParent="true"
        android:padding="12dp"
        android:layout_margin="12dp">

        <TextView
            android:id="@+id/avgSteps"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Average steps : " />

        <TextView
            android:id="@+id/avgCalories"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="12dp"
            android:text="Average calories burned : " />

        <TextView
            android:id="@+id/avgDistance"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="12dp"
            android:text="Average distance : " />

        <Button
            android:id="@+id/btnOk"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_alignParentBottom="true"
            android:text="OK"
            android:layout_marginTop="15dp" />
    </LinearLayout>
</RelativeLayout>
```

activity_bar_chart.

```
<?xml version="1.0" encoding="utf-8"?>
<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=".BarChartActivity">

    <RelativeLayout
        android:id="@+id/linearlayout"
        app:layout_constraintTop_toTopOf="parent"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="12dp"
        android:orientation="vertical"
        android:padding="12dp">

        <EditText
            android:id="@+id/firstNameSearch"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Enter First Name" />
```

```
<EditText
    android:id="@+id/lastNameSearch"
    android:layout_below="@+id/firstNameSearch"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Enter Second Name" />

<com.google.android.material.button.MaterialButton
    app:cornerRadius="28dp"
    android:id="@+id/searchData"
    android:layout_below="@+id/lastNameSearch"
    android:layout_width="100dp"
    android:layout_height="wrap_content"
    android:layout_marginEnd="20dp"
    app:icon="@drawable/ic_baseline_search_24"
    app:iconGravity="textStart"
    app:iconPadding="0dp"
    android:layout_marginTop="12dp"
    android:textAllCaps="false"/>
```

```
<com.google.android.material.button.MaterialButton
    app:cornerRadius="28dp"
    android:id="@+id/btnBChart"
    android:layout_width="100dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="1dp"
    app:icon="@drawable/ic_baseline_bar_chart_24"
    app:iconGravity="textStart"
    app:iconPadding="0dp"
    android:layout_toRightOf="@+id/searchData"
    android:layout_below="@+id/lastNameSearch"
    android:layout_marginTop="12dp"
    android:textAllCaps="false" />
```

```
<com.google.android.material.button.MaterialButton
    app:cornerRadius="28dp"
    android:id="@+id/graphData"
    android:layout_width="100dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="10dp"
    app:icon="@drawable/ic_baseline_line_chart_24"
    app:iconGravity="textStart"
    app:iconPadding="0dp"
    android:layout_toRightOf="@+id/btnBChart"
    android:layout_below="@+id/lastNameSearch"
    android:layout_marginTop="12dp"
    android:textAllCaps="false" />
```

```
<com.androidplot.xy.XYPlot
    android:id="@+id/barChart"
    style="@style/APDefacto"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/graphData"
    android:layout_marginTop="12dp"
    app:lineLabelRotationBottom="-45"
    app:lineLabelRotationLeft="-45"
    app:lineLabels="left|bottom"
    app:title="Title" />
```

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

activity_line_chart.

```
<?xml version="1.0" encoding="utf-8"?>
<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=".LineChartActivity">

    <RelativeLayout
        android:id="@+id/linearlayout"
        app:layout_constraintTop_toTopOf="parent"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="12dp"
        android:orientation="vertical"
        android:padding="12dp">

        <EditText
            android:id="@+id/firstNameSearch"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="First Name" />
```

```
<EditText
    android:id="@+id/lastNameSearch"
    android:layout_below="@+id/firstNameSearch"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Second Name" />

<com.google.android.material.button.MaterialButton
    app:cornerRadius="28dp"
    android:id="@+id/searchData"
    android:layout_below="@+id/lastNameSearch"
    android:layout_width="100dp"
    android:layout_height="wrap_content"
    android:layout_marginEnd="20dp"
    app:icon="@drawable/ic_baseline_search_24"
    app:iconGravity="textStart"
    app:iconPadding="0dp"
    android:layout_marginTop="12dp"
    android:textAllCaps="false" />
```



```
<com.google.android.material.button.MaterialButton
    app:cornerRadius="28dp"
    android:id="@+id/btnBChart"
    android:layout_width="100dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="1dp"
    app:icon="@drawable/ic_baseline_bar_chart_24"
    app:iconGravity="textStart"
    app:iconPadding="0dp"
    android:textSize="12sp"
    android:layout_toRightOf="@+id/searchData"
    android:layout_below="@+id/lastNameSearch"
    android:layout_marginTop="12dp"
    android:textAllCaps="false"/>
<com.google.android.material.button.MaterialButton
    app:cornerRadius="28dp"
    android:id="@+id/graphData"
    android:layout_width="100dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="10dp"
    app:icon="@drawable/ic_baseline_line_chart_24"
    app:iconGravity="textStart"
    app:iconPadding="0dp"
    android:textSize="11sp"
    android:layout_toRightOf="@+id/btnBChart"
    android:layout_below="@+id/lastNameSearch"
    android:layout_marginTop="12dp"
    android:textAllCaps="false"/>
```

```
<com.androidplot.xy.XYPlot
    android:id="@+id/lineChart"
    style="@style/APDefacto"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/graphData"
    android:layout_marginTop="12dp"
    app:lineLabelRotationBottom="-45"
    app:lineLabelRotationLeft="-45"
    app:lineLabels="left|bottom"
    app:title="Title" />
</RelativeLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

activity_splash_screen.

```
<?xml version="1.0" encoding="utf-8"?>
<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=".SplashScreenActivity"
    android:background="@drawable/autracker">

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

Arduino Code.

```
#include <math.h>
#include <Wire.h>
#include <SoftwareSerial.h>
SoftwareSerial BTSerial(2, 3);

#define SENDING_INTERVAL 1000
#define SENSOR_READ_INTERVAL 50
unsigned long prevSensoredTime = 0;
unsigned long curSensoredTime = 0;
#define ACCEL_BUFFER_COUNT 125
byte aAccelBuffer[ACCEL_BUFFER_COUNT];
int iAccelIndex = 2;

/* MPU-6050 sensor */
#define MPU6050_ACCEL_XOUT_H 0x3B
#define MPU6050_FWR_MGMT_1 0x6B
#define MPU6050_FWR_MGMT_2 0x6C
#define MPU6050_WHO_AM_I 0x75
#define MPU6050_I2C_ADDRESS 0x68

typedef union accel_t_gyro_union {
  struct {
    uint8_t x_accel_h;
    uint8_t x_accel_l;
    uint8_t y_accel_h;
    uint8_t y_accel_l;
    uint8_t z_accel_h;
    uint8_t z_accel_l;
    uint8_t t_h;
    uint8_t t_l;
    uint8_t x_gyro_h;
    uint8_t x_gyro_l;
    uint8_t y_gyro_h;
    uint8_t y_gyro_l;
    uint8_t z_gyro_h;
    uint8_t z_gyro_l;
  } reg;

  struct {
    int x_accel;
    int y_accel;
    int z_accel;
    int temperature;
    int x_gyro;
    int y_gyro;
    int z_gyro;
  } value;
};
```

Final Year Project Technical Document

```
void setup() {
    int error;
    uint8_t c;

    Serial.begin(9600);
    Wire.begin();
    BTSerial.begin(9600);

    // Clear the 'sleep' bit to start the sensor.
    MPU6050_write_reg (MPU6050_PWR_MGMT_1, 0);

    initBuffer();
}

void loop() {
    curSensoredTime = millis();

    // Read from sensor
    if(curSensoredTime - prevSensoredTime > SENSOR_READ_INTERVAL) {
        readFromSensor(); // Read from sensor
        prevSensoredTime = curSensoredTime;

        if(iAccelIndex >= ACCEL_BUFFER_COUNT - 3) {
            sendToRemote();
            initBuffer();
        }
    }
}

void sendToRemote() {

    BTSerial.write( "accel" );
    // Write accel data
    BTSerial.write( (char*)aAccelBuffer );
}

void readFromSensor() {
    int error;
    double dI;
    accel_t_gyro_union accel_t_gyro;

    error = MPU6050_read (MPU6050_ACCEL_XOUT_H, (uint8_t *) &accel_t_gyro, sizeof(accel_t_gyro));
    if(error != 0) {
        Serial.print(F("Read accel, temp and gyro, error = "));
        Serial.println(error,DEC);
    }

    uint8_t swap;
    #define SWAP(x,y) swap = x; x = y; y = swap
    SWAP (accel_t_gyro.reg.x_accel_h, accel_t_gyro.reg.x_accel_l);
    SWAP (accel_t_gyro.reg.y_accel_h, accel_t_gyro.reg.y_accel_l);
    SWAP (accel_t_gyro.reg.z_accel_h, accel_t_gyro.reg.z_accel_l);
    SWAP (accel_t_gyro.reg.t_h, accel_t_gyro.reg.t_l);
    SWAP (accel_t_gyro.reg.x_gyro_h, accel_t_gyro.reg.x_gyro_l);
    SWAP (accel_t_gyro.reg.y_gyro_h, accel_t_gyro.reg.y_gyro_l);
    SWAP (accel_t_gyro.reg.z_gyro_h, accel_t_gyro.reg.z_gyro_l);
}
```

Final Year Project Technical Document

```
// Print the raw acceleration values
Serial.print(accel_t_gyro.value.x_accel, DEC);
Serial.print(F(", "));
Serial.print(accel_t_gyro.value.y_accel, DEC);
Serial.print(F(", "));
Serial.print(accel_t_gyro.value.z_accel, DEC);
Serial.print(F(", at "));
Serial.print(iAccelIndex);
Serial.println(F(""));

if(iAccelIndex < ACCEL_BUFFER_COUNT && iAccelIndex > 1) {
  int tempX = accel_t_gyro.value.x_accel;
  int tempY = accel_t_gyro.value.y_accel;
  int tempZ = accel_t_gyro.value.z_accel;

  char temp = (char)(tempX >> 8);
  if(temp == 0x00)
    temp = 0x7f;
  aAccelBuffer[iAccelIndex] = temp;
  iAccelIndex++;
  temp = (char)(tempX);
  if(temp == 0x00)
    temp = 0x01;
  aAccelBuffer[iAccelIndex] = temp;
  iAccelIndex++;

  temp = (char)(tempY >> 8);
  if(temp == 0x00)
    temp = 0x7f;
  aAccelBuffer[iAccelIndex] = temp;
  iAccelIndex++;
  temp = (char)(tempY);
  if(temp == 0x00)
    temp = 0x01;
  aAccelBuffer[iAccelIndex] = temp;
  iAccelIndex++;

  temp = (char)(tempZ >> 8);
  if(temp == 0x00)
    temp = 0x7f;
  aAccelBuffer[iAccelIndex] = temp;
  iAccelIndex++;
  temp = (char)(tempZ);
  if(temp == 0x00)
    temp = 0x01;
  aAccelBuffer[iAccelIndex] = temp;
  iAccelIndex++;
}
}
```

Final Year Project Technical Document

```
int MPU6050_read(int start, uint8_t *buffer, int size)
{
    int i, n, error;

    Wire.beginTransaction(MPU6050_I2C_ADDRESS);
    n = Wire.write(start);
    if (n != 1)
        return (-10);

    n = Wire.endTransmission(false); // hold the I2C-bus
    if (n != 0)
        return (n);
    .
    Wire.requestFrom(MPU6050_I2C_ADDRESS, size, true);
    i = 0;
    while(Wire.available() && i<size)
    {
        buffer[i++]=Wire.read();
    }
    if ( i != size)
        return (-11);
    return (0); // return : no error
}

int MPU6050_write(int start, const uint8_t *pData, int size)
{
    int n, error;
    Wire.beginTransaction(MPU6050_I2C_ADDRESS);
    n = Wire.write(start); // write the start address
    if (n != 1)
        return (-20);
    n = Wire.write(pData, size); // write data bytes
    if (n != size)
        return (-21);
    error = Wire.endTransmission(true); // release the I2C-bus
    if (error != 0)
        return (error);
    return (0); // return : no error
}

int MPU6050_write_reg(int reg, uint8_t data)
{
    int error;
    error = MPU6050_write(reg, &data, 1);
    return (error);
}

void initBuffer() {
    iAccelIndex = 2;
    for(int i=iAccelIndex; i<ACCEL_BUFFER_COUNT; i++) {
        aAccelBuffer[i] = 0x00;
    }
    aAccelBuffer[0] = 0xfe;
    aAccelBuffer[1] = 0xfd;
    aAccelBuffer[122] = 0xfd;
    aAccelBuffer[123] = 0xfe;
    aAccelBuffer[124] = 0x00;
}
```

Reference Declaration.

The Arduino section of my project was aided by following the following tutorial.

<https://www.instructables.com/Make-your-own-activity-tracker/>

Plagiarism Declaration.

- I declare that all material in this submission e.g., Thesis/essay/project/assignment is entirely my/our own work except where duly acknowledged.
- I have cited the sources of all quotations, paraphrases, summaries of information, tables, diagrams, or other material; including software and other electronic media in which intellectual property rights may reside
- I have provided a complete bibliography of all works and sources used in the preparation of this submission.
- I understand that failure to comply with the Institute's regulations governing plagiarism constitutes a serious offense.

Student Name: Dylan Scott

Student Number: C00242050

Signature: