

DCN ASSIGNMENT – 3

Submitted By – Rishi Raj Chaurasia

Course – MCA 2nd SEM

Roll no. – 31

Objective :

To Find the devices near your phone/device.

Introduction :

Bluetooth

We're all use to wireless communication by now, even if we don't always realize it. Radio receivers and television sets pick up programs beamed in radio waves hundreds (possibly even thousands) of km/miles through the air. Bluetooth is a similar radio-wave technology, but it's mainly designed for communicating over short distances less than about 10m or 30ft. Most electronic gadgets that we use these days have built-in radio antennas (transmitters and receivers) so they can simultaneously send and receive wireless signals to other Bluetooth gadgets. Older gadgets can be converted to work with Bluetooth using plug-in adapters.

Bluetooth devices automatically detect and connect to one another. It can be used to detect and connect to other nearby Bluetooth devices.

App Demo :

When the app starts

1. Checks Location, Storage and Bluetooth Permissions are granted or not.
2. Gets Location and Bluetooth Permissions, if not granted.
3. Checks if Bluetooth and GPS is enabled or not.
4. Enable Bluetooth and GPS if not enabled.

App Frontend

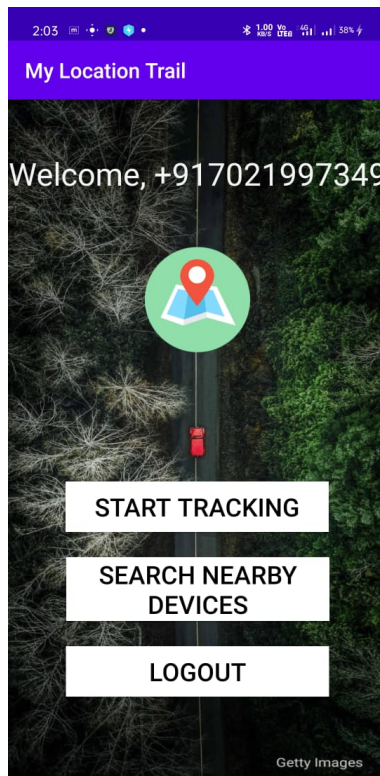
1. On app start you are required to register if not already registered.
2. You are asked to enter mobile phone number.
 1. OTP is sent.
 2. OTP is read automatically or can be manually entered.
 3. User logs in.

It uses following permissions

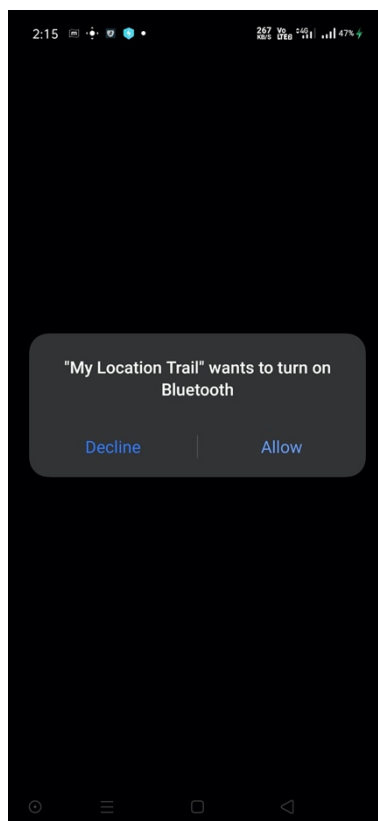
1. - android.permission.BLUETOOTH
2. - android.permission.BLUETOOTH_ADMIN
3. - android.permission.ACCESS_COARSE_LOCATION
4. - android.permission.ACCESS_FINE_LOCATION
5. - android.permission.WRITE_EXTERNAL_STORAGE

For nearby device detection

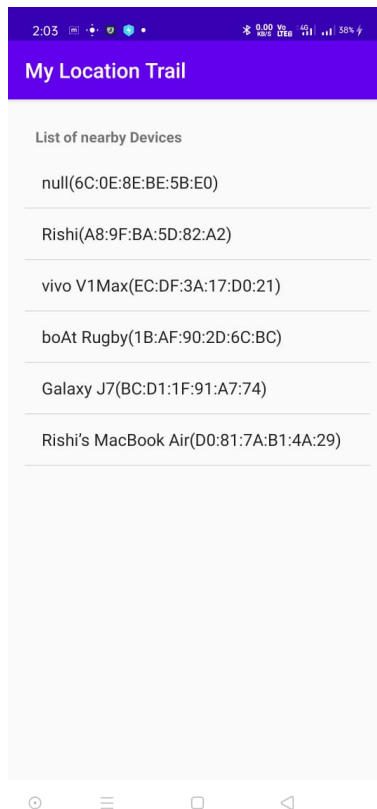
1. Click on “Search nearby devices” button on home screen.



2. App asks to turn on Bluetooth if not already enabled.



3. On clicking on “Allow” app starts to detect nearby Bluetooth devices which are discoverable.



4. List of devices along with their name and device MAC Address is displayed to uniquely identify each device in case of similar names.
5. MAC Addresses of nearby devices can be stored and used to trace out the location and contact information of users from the database who have come in vicinity of the host user.

Code Snippet :

BluetoothActivity.java

```
import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;

import android.bluetooth.BluetoothAdapter;
import android.bluetooth.BluetoothDevice;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.content.IntentFilter;
import android.os.Bundle;
```

```

import android.widget.AdapterView;
import android.widget.ListView;
import android.widget.Toast;

import java.util.ArrayList;

public class BluetoothActivity extends AppCompatActivity {

    private static final int REQUEST_ENABLE_BT =1 ;
    BluetoothAdapter mBluetoothAdapter;
    ListView mListView;
    ArrayList<Device> arrayList;
    ArrayAdapter<Device> arrayAdapter;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_bluetooth);
        mBluetoothAdapter =BluetoothAdapter.getDefaultAdapter();
        mListView= findViewById(R.id.listDevices);
        arrayList=new ArrayList<>();
        checkBluetoothEnabled();
        discoverDevices();

    }
    private void checkBluetoothEnabled(){
        if(!mBluetoothAdapter.isEnabled())
        {
            Intent intent=new Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);
            startActivityForResult(intent,REQUEST_ENABLE_BT);
        }
    }
    @Override
    protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {
        super.onActivityResult(requestCode, resultCode, data);
        if (requestCode == REQUEST_ENABLE_BT) {
            if (resultCode == RESULT_OK)
            {
                Toast.makeText(this, "Bluetooth is Enabled", Toast.LENGTH_LONG).show();
            }
            else if(resultCode == RESULT_CANCELED)
            {
                Toast.makeText(this, "Bluetooth Enabling Failed", Toast.LENGTH_LONG).show();
            }
        }
    }
}

```

```

class Device{
    String deviceName;
    String deviceAddress;

    Device(String deviceName, String deviceAddress){
        this.deviceName = deviceName;
        this.deviceAddress = deviceAddress;
    }

    @NonNull
    @Override
    public String toString() {
        return deviceName + "(" + deviceAddress + ")";
    }
}

private void discoverDevices(){
    mBluetoothAdapter.startDiscovery();
    IntentFilter intentFilter=new IntentFilter(BluetoothDevice.ACTION_FOUND);
    registerReceiver(myReceiver, intentFilter);
    arrayAdapter=new ArrayAdapter<>(getApplicationContext(),
                                   android.R.layout.simple_list_item_1, arrayList);
    mListView.setAdapter(arrayAdapter);
}
BroadcastReceiver myReceiver= new BroadcastReceiver() {
    @Override
    public void onReceive(Context context, Intent intent) {
        String action= intent.getAction();
        if(BluetoothDevice.ACTION_FOUND.equals(action))
        {
            BluetoothDevice device=
                intent.getParcelableExtra(BluetoothDevice.EXTRA_DEVICE);
            Device newDevice = new Device(device.getName(), device.getAddress());
            arrayList.add(newDevice);
            arrayAdapter.notifyDataSetChanged();
        }
    }
};
}

```

Note: All other code is exactly same as submitted in previous assignment.

-----X-----