

Project Development Phase Sprint-1

Team ID	PNT2022TMID13325
Project Name	Containment zone alerting application

MainActivity.java

```
package com.example.client_containment;
```

```
import androidx.appcompat.app.AppCompatActivity;
```

```
import androidx.core.app.ActivityCompat;
```

```
import android.Manifest;
```

```
import android.app.PendingIntent;
```

```
import android.content.Intent;
```

```
import android.content.SharedPreferences;
```

```
import android.content.pm.PackageManager;
```

```
import android.os.Bundle;
```

```
import android.util.Log;
```

```
import android.widget.TextView;
```

```
import android.widget.Toast;
```

```
import com.example.client_containment.Service.MyLocationService;
```

```
import com.google.android.gms.location.FusedLocationProviderClient;
```

```
import com.google.android.gms.location.LocationRequest;
```

```
import com.google.android.gms.location.LocationServices;
```

```
import com.karumi.dexter.Dexter;
```

```
import com.karumi.dexter.PermissionToken;
```

```
import com.karumi.dexter.listener.PermissionDeniedResponse;
```

```
import com.karumi.dexter.listener.PermissionGrantedResponse;
```

```
import com.karumi.dexter.listener.PermissionRequest;
```

```
import com.karumi.dexter.listener.single.PermissionListener;
```

```

public class MainActivity extends AppCompatActivity {
    static MainActivity instance;
    LocationRequest locationRequest;
    TextView loc;
    FusedLocationProviderClient fusedLocationProviderClient;
    SharedPreferences sharedPreferences;

    public static MainActivity getInstance() {
        return instance;
    }

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        instance = this;
        loc = findViewById(R.id.location_text);
        sharedPreferences = getApplicationContext().getSharedPreferences("user_data", 0);
        Log.d("shared_pref", sharedPreferences.getString("name", "0"));
        Dexter.withActivity(this)
            .withPermission(Manifest.permission.ACCESS_FINE_LOCATION)
            .withListener(new PermissionListener() {
                @Override
                public void onPermissionGranted(PermissionGrantedResponse response) {
                    updateLocation();
                }

                @Override
                public void onPermissionDenied(PermissionDeniedResponse response) {
                    Toast.makeText(MainActivity.this, "No location",
                        Toast.LENGTH_LONG).show();
                }
            })
    }
}

```

```

        @Override
        public void onPermissionRationaleShouldBeShown(PermissionRequest
permission, PermissionToken token) {

            }

        }).check();

    }

    private void updateLocation() {
        buildLocationRequest();

        fusedLocationProviderClient = LocationServices.getFusedLocationProviderClient(this);

        if (ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED && ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_COARSE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {

            // TODO: Consider calling
            //    ActivityCompat#requestPermissions
            // here to request the missing permissions, and then overriding
            //    public void onRequestPermissionsResult(int requestCode, String[] permissions,
            //                                          int[] grantResults)
            // to handle the case where the user grants the permission. See the documentation
            // for ActivityCompat#requestPermissions for more details.

            return;
        }

        fusedLocationProviderClient.requestLocationUpdates(locationRequest,
getPendingIntent());
    }

    private PendingIntent getPendingIntent() {
        Intent intent = new Intent(this, MyLocationService.class);
        intent.setAction(MyLocationService.ACTION_PROCESS_UPDATE);

        return
PendingIntent.getBroadcast(this,0,intent,PendingIntent.FLAG_UPDATE_CURRENT);
    }

    private void buildLocationRequest(){

```

```

        locationRequest = new LocationRequest();
        locationRequest .setPriority(LocationRequest.PRIORITY_HIGH_ACCURACY);
        locationRequest.setInterval(3000);
        locationRequest.setFastestInterval(1000);
        locationRequest.setSmallestDisplacement(10f);
    }

    public void updateTextView(String location){
        MainActivity.this.runOnUiThread(new Runnable() {
            @Override
            public void run() {
                loc.setText(location);
            }
        });
    }
}

```

SignUp.java

```

package com.example.client_containment;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.content.SharedPreferences;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.EditText;

```

```

import com.android.volley.Request;
import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.VolleyError;

```

```
import com.android.volley.toolbox.JsonObjectRequest;
```

```
import com.android.volley.toolbox.Volley;
```

```
import org.json.JSONException;
```

```
import org.json.JSONObject;
```

```
public class SignUp extends AppCompatActivity {
```

```
    private EditText name;
```

```
    private EditText email;
```

```
    private EditText password;
```

```
    SharedPreferences sharedPreferences;
```

```
    @Override
```

```
    protected void onCreate(Bundle savedInstanceState) {
```

```
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_sign_up);
```

```
        name = findViewById(R.id.name);
```

```
        email = findViewById(R.id.email);
```

```
        password = findViewById(R.id.password);
```

```
        sharedPreferences = getApplicationContext().getSharedPreferences("user_data", 0);
```

```
        SharedPreferences.Editor editor = sharedPreferences.edit();
```

```
        editor.clear();
```

```
        editor.commit();
```

```
        if(sharedPreferences.getAll().size() >= 3){
```

```
            Intent intent = new Intent(this, MainActivity.class);
```

```
            startActivity(intent);
```

```
}
```

```
}
```

```
public void signUp(View view) {  
    if(!name.getText().equals("") || !email.getText().equals("") ||  
    !password.getText().equals("")){  
  
        postDataUsingVolley(name.getText().toString(),email.getText().toString(),password.getText().  
        toString());  
    }  
}  
  
private void postDataUsingVolley(String name, String email,String password) {  
    final RequestQueue queue = Volley.newRequestQueue(this);  
    String url = "http://192.168.43.234:5000/android_signup";  
  
    JSONObject postparams = new JSONObject();  
    try {  
        postparams.put("name", name);  
        postparams.put("email", email);  
        postparams.put("password", password);  
    } catch (JSONException e) {  
        e.printStackTrace();  
    }  
  
    JsonObjectRequest jsonObjReq = new JsonObjectRequest(Request.Method.POST, url,  
    postparams,  
        response -> {  
            Log.d("response",response.toString());  
  
            try {  
                int userId = response.getInt("id");  
  
                SharedPreferences.Editor editor = sharedPreferences.edit();
```

```
        editor.putString("name", name);
        editor.putString("email", email);
        editor.putInt("id", userId );
        editor.commit();
        System.out.println("The user id is : "+name+" "+email+" "+userId);
        Intent intent = new Intent(SignUp.this,MainActivity.class);
        startActivity(intent);

    } catch (JSONException e) {
        e.printStackTrace();
    }

},

error -> Log.d("error",error.toString());

queue.add(jsonObjReq);
}
}
```

Screenshots

