

FINAL CODE

<i>Date</i>	<i>24 November 2022</i>
<i>Team ID</i>	<i>PNT2022TMID46284</i>
<i>Project Name</i>	<i>Project- IoT based safety gadget for child safety monitoring and notification</i>

Alert Notification code

```
package com.example.geofence; import
android.content.BroadcastReceiver; imp
ort android.content.Context;
import android.content.Intent; import
android.location.Location; import
android.os.CountDownTimer; imp
ort android.util.Log;
import android.widget.Toast;
import com.google.android.gms.location.Geofence; import
com.google.android.gms.location.GeofencingEvent; im
port java.util.List;
import android.os.Handler; public class GeofenceBroadcastReceiver extends BroadcastReceiver
{
private static final String TAG =
"GeofenceBroadcastReceiv"; @Override
public void onReceive( Context context, Intent intent)
{
//TODO: This method is called when the Broadcast Receiver is receiving
//an Intent broadcast
//.
/* Toast.makeText(context, "GEOFENCE_ENTERED",
Toast.LENGTH_SHORT).show(); final Toast mToastToShow;
intoastDurationInMilliseconds=1200000;
mToastToShow=Toast.makeText(context, "GEOFENCE_EXITED", Toast.LENGTH_LONG);
// Set the countdown to display the
```

```

toastCountDownTimertoastCountDown;

toastCountDown=newCountDownTimer(toastDurationInMilliseconds,100000)
{
    publicvoidonTick(longmillisUntilFinished)
    {
        mToastToShow.show();
    }
    publicvoidonFinish()
    {
        mToastToShow.cancel();
    }
};

// Show the toast and starts the
countdownmToastToShow.show();toast
CountDown.start();*/

NotificationHelper notificationHelper = new
NotificationHelper(context);notificationHelper.sendHighPriorityNotification("GEOFENCE_T
RANSITION_ENTER","",MapsActivity.class);

GeofencingEvent geofencingEvent =
GeofencingEvent.fromIntent(intent);if(geofencingEvent.hasError()){
    Log.d(TAG, "onReceive: Error receiving geofence
event...");return;
}

List<Geofence> geofenceList =
geofencingEvent.getTriggeringGeofences();for(Geofencegeofence:
geofenceList)
{
    Log.d(TAG, "onReceive:"+geofence.getRequestId());
}

// Location location =
geofencingEvent.getTriggeringLocation();int transitionType =
geofencingEvent.getGeofenceTransition();switch(transitio nType)

```

```

{
    case Geofence.GEOFENCE_TRANSITION_ENTER:
        notificationHelper.sendHighPriorityNotification("Entered the
Location", "", MapsActivity.class); break;

    case Geofence.GEOFENCE_TRANSITION_EXIT: notificationHelper.sendHi
ghPriorityNotification("Exited the Location ", "", MapsActivity.class); break;
}
}

```

Geofence:

Package

```

com.example.geofence; import
android.app.PendingIntent; impo
rt android.content.Context; import
android.content.ContextWrapper; impo
rt android.content.Intent;
import android.widget.Toast;
import com.google.android.gms.common.api.ApiException; i
mport com.google.android.gms.location.Geofence; import
com.google.android.gms.location.GeofenceStatusCodes; import
com.google.android.gms.location.GeofencingRequest; imp
ort com.google.android.gms.maps.model.LatLng;
public class GeofenceHelper extends ContextWrapper
{
    private static final String TAG =
"GeofenceHelper";
    PendingIntent pendingIntent;
    public GeofenceHelper(Context base)

```

```

{
    super(base);
}
public GeofencingRequest
getGeofencingRequest(Geofence geofence){
    return new GeofencingRequest.Builder()
        .addGeofence(geofence)

        .setInitialTrigger(GeofencingRequest.INITIAL_TRIGGER_ENTER)
        .build();
}

public Geofence getGeofence(String ID, LatLng latLng,
    float radius, int transitionTypes)
{
    return new Geofence.Builder()
        .setCircularRegion(latLng.latitude,
            latLng.longitude, radius)
        .setRequestId(ID)
        .setTransitionTypes(transitionTypes)
        .setLoiteringDelay(5000)
        .setExpirationDuration(Geofence.NEVER_EXPIRE)
        .build();
}

public PendingIntent getPendingIntent()
{
    if (pendingIntent != null)
    {
        return pendingIntent;
    }

    Intent intent = new
    Intent(this, GeofenceBroadcastR
        eceiver.class);

    pendingIntent = PendingIntent.getBroadcast(this,
        2607, intent, PendingIntent.FLAG_IMMUTABLE);
    return pendingIntent;
}

```

```

}

public String getErrorString(Exception e)
{
    if (e instanceof ApiException)
    {
        ApiException apiException =
            (ApiException)
                e; switch (apiException.getStatusCode())
        {
            case GeofenceStatusCodes
                .GEOFENCE_NOT_AVAILABLE:

                return "GEOFENCE_NOT_AVAILABLE";
            case GeofenceStatusCodes
                .GEOFENCE_TOO_MANY_GEOFENCES:
                return "GEOFENCE_TOO_MANY_GEOFENCES";
            case GeofenceStatusCodes
                .GEOFENCE_TOO_MANY_PENDING_INTENTS:
                return "GEOFENCE_TOO_MANY_PENDING_INTENTS";
        }
    }
    return e.getLocalizedMessage();
}
}

```