

Usługi lokalizacji GPS

PUM

Zezwolenia – AndroidManifest.xml

1. Umieścić w pliku AndroidManifest.xml pozwolenie na korzystanie z usługi GPS
linie o kolorze czerwonym

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    package="com.example.gps_svalbard">

    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
```

Zezwolenia – MainActivity.java

2. Sprawdzenie czy aplikacja uzyskała od administratora urządzenia pozwolenie na korzystanie z – usługi lokalizacji GPS.

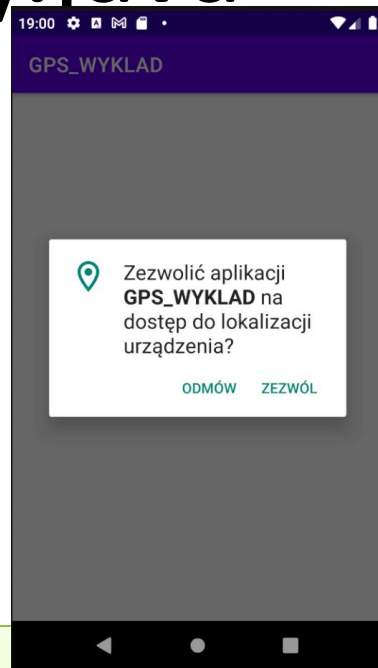
Jeśli brak zezwolenia aplikacja pyta o zezwolenie.

Wewnątrz MainActivity

```
private static final int MY_PERMISSIONS_REQUEST_LOCATION = 1;
```

Wewnątrz onCreate()

```
// Sprawdzenie uprawnień do korzystania z GPS (usługi lokalizacji)
if (ContextCompat.checkSelfPermission(this, android.Manifest.permission.ACCESS_FINE_LOCATION) !=
    PackageManager.PERMISSION_GRANTED) {
    ActivityCompat.requestPermissions(this, new String[]{android.Manifest.permission.ACCESS_FINE_LOCATION},
        MY_PERMISSIONS_REQUEST_LOCATION);
}
```



LocationManager

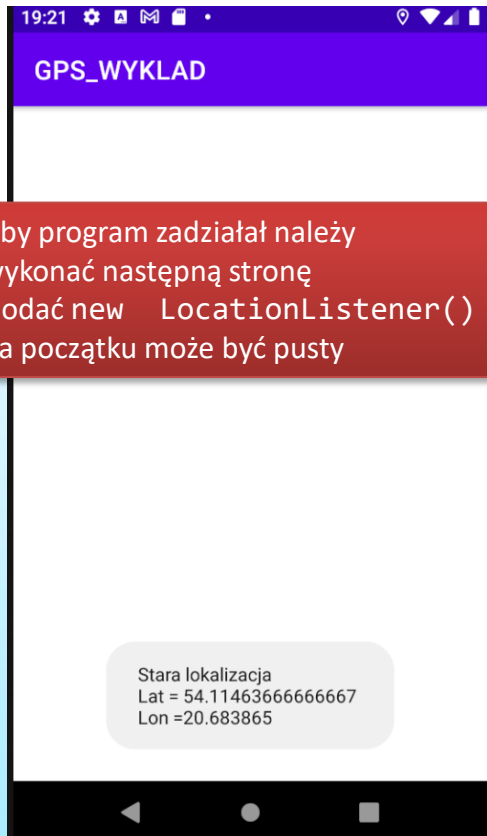
3. Utworzenie obiektów `locationManager` i `locationProvider` wewnątrz metody `onCreate()`

```
locationManager = (LocationManager) getSystemService(Context.LOCATION_SERVICE);
locationProvider = locationManager.getProvider(LocationManager.GPS_PROVIDER);

if (locationProvider != null) {
    try {
        locationManager.requestLocationUpdates(locationProvider.getName(),
            10000, // minimalny odstęp w czasie w ms między meldunkami o pozycji
            25,    // minimalna zmiana położenia w metrach
            this.locationListener);
    } catch (SecurityException e) {
        e.printStackTrace();
    }
}

Location location = locationManager.getLastKnownLocation(LocationManager.GPS_PROVIDER);
Toast.makeText(getApplicationContext(),
    "Stara lokalizacja \nLat = " + String.valueOf(location.getLatitude()) + "\nLon = "
        + String.valueOf(location.getLongitude()) ,
    Toast.LENGTH_LONG).show();
```

Aby program zadziałał należy wykonać następną stronę
Dodać `new LocationListener()`
na początku może być pusty



W podpowiedziach wybrać utworzenie zmiennych `locationManager` i `locationProvider` w `MainActivity`, jak na następnym slajdzie

Implementacja interfejsu

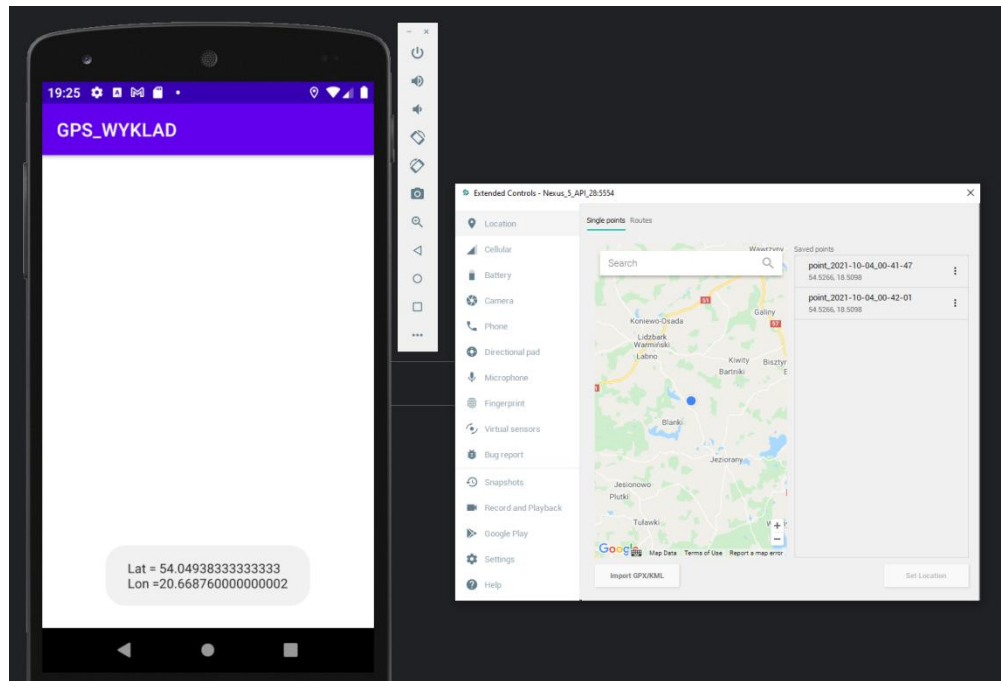
onLocationChanged(@NonNull Location location)

4. Odczytywanie meldunków o zmianie pozycji – kod w MainActivity

```
private static final int MY_PERMISSIONS_REQUEST_LOCATION = 324;
private LocationManager locationManager;
private LocationProvider locationProvider;
private LocationListener locationListener = new LocationListener() {
    @Override
    public void onLocationChanged(@NonNull Location location) {
        final double lat = location.getLatitude();
        final double lon = location.getLongitude();
        final long time = location.getTime();
        final double speed = location.getSpeed();

        Toast.makeText(getApplicationContext(),
            "Lat = " + String.valueOf(lat) + "\nLon = " + String.valueOf(lon) ,
            Toast.LENGTH_LONG).show();
    }
};
```

Zmiana lokalizacji



Zezwolenie na dostęp do internetu

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    package="com.example.gptest1">

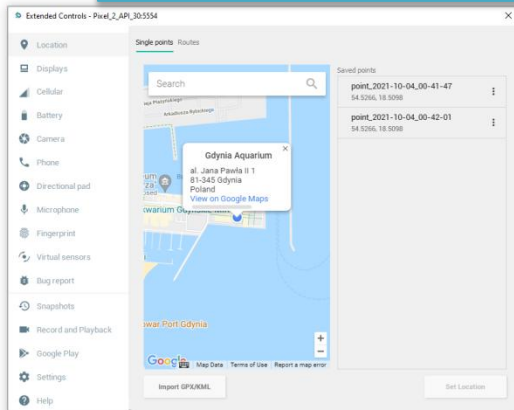
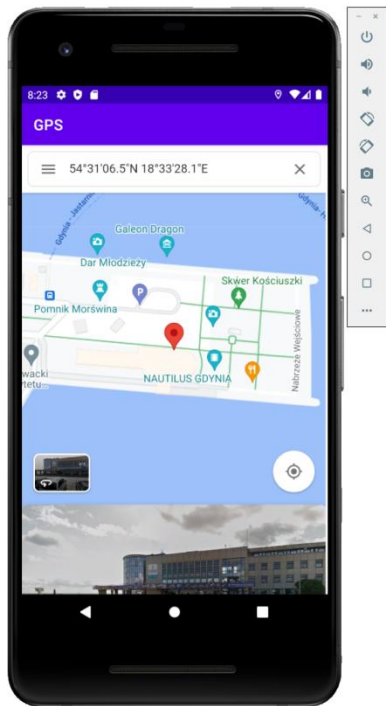
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
    <uses-permission android:name="android.permission.INTERNET" />

    <application
        android:usesCleartextTraffic="true"
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
```

Dodanie wyświetlania mapy z aktualną pozycją w WebView

Kod umieścić wewnątrz metody `onLocationChanged()`

```
WebView webView = (WebView) findViewById(R.id.webView);
webView.getSettings().setJavaScriptEnabled(true);
webView.getSettings().setDomStorageEnabled(true);
webView.getSettings().setAllowFileAccess(true);
webView.setWebViewClient(new WebViewClient());
webView.loadUrl("https://maps.google.com/maps?q=" + lat + "," + lon + "&t=m&z=13");
```



Przy pierwszym uruchomieniu przeglądarka wymaga zgody na pliki ciasteczek. Można ją uruchomić jako gość, bez zalogowania do konta Google. Na początku może wymagać kilkukrotnego uruchomienia aplikacji. Testowano na SDK 31.

Pixel 2 API 30

Wysyłanie danych o pozycji

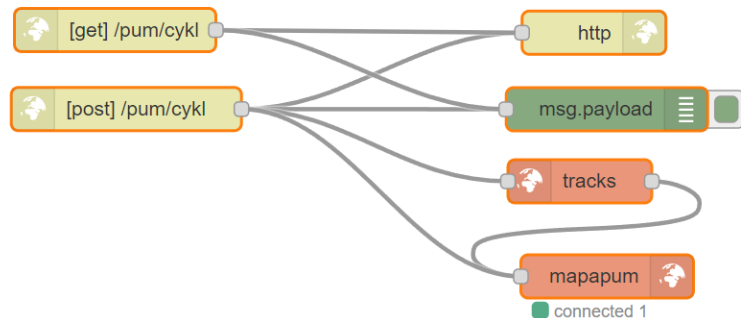
biblioteka Volley: <https://developer.android.com/training/volley/request>

```
RequestQueue queue = Volley.newRequestQueue(getApplicationContext());  
String url = "http://lukan.sytes.net:1880/pum/cykl";
```

```
JSONObject jsonObject = null;  
try {  
    jsonObject = new JSONObject();  
    jsonObject.put("name", "Andrzej");  
    jsonObject.put("lat", lat);  
    jsonObject.put("lon", lon);  
} catch (JSONException e) {  
    e.printStackTrace();  
}
```

```
JsonObjectRequest jsonObjectRequest = new JsonObjectRequest(Request.Method.POST, url,  
    jsonObject, new Response.Listener<JSONObject>() {  
        @Override  
        public void onResponse(JSONObject response) {  
        }  
    }, null);
```

```
queue.add(jsonObjectRequest);
```



Poprawianie błędu Volley

```
Volley.newRequestQueue(getApplicationContext());  
Luka  
: = n  
JSON  
name"  
.at",  
--"
```

- Create local variable 'Volley'
- Create field 'Volley'
- Create parameter 'Volley'
- Create class 'Volley'
- Create inner class 'Volley'
- Add dependency on com.android.volley:volley and import
- Rename reference

Trasa na mapie

