

Sistema de Posicionamento Global GPS



Sistema de Posicionamento Global

- Sistema de posicionamento por satélite
- Criado para uso militar
- Vinculado ao Departamento de Defesa dos Estados Unidos da América
- Sistemas de posicionamento por satélite em operação ou em desenvolvimento:
 - GPS → Americano
 - GLONASS → Russo
 - GALILEO → União Européia
 - COMPASS → China

Sistema de Posicionamento Global

- Declarado operante desde 1995.
- Sistema para uso civil gratuito.
- Muito utilizado para levantamento topográfico
- Permite obter:
 - as coordenadas geográficas em Longitude e Latitude (WGS-84)
 - altitude em relação ao nível do mar
 - hora certa
 - precisão em metros

Sistema de Posicionamento Global

- Os satélites GPS ofertam serviço para uso civil e para uso militar.
 - O uso civil está disponível gratuitamente e continuamente em todo o globo terrestre.
 - O uso militar está disponível apenas para as forças armadas dos E.U.A. e aliados.
- Precisão:
 - O GPS de uso civil possui “erro” propositalmente adicionado ao sinal.
 - Precisão horizontal (lat/long) melhor do que a vertical (altitude)
 - GPS comum possui precisão em torno de 3,5 metros. GPS topográfico pode chegar a 1cm.

Sistema de Posicionamento Global

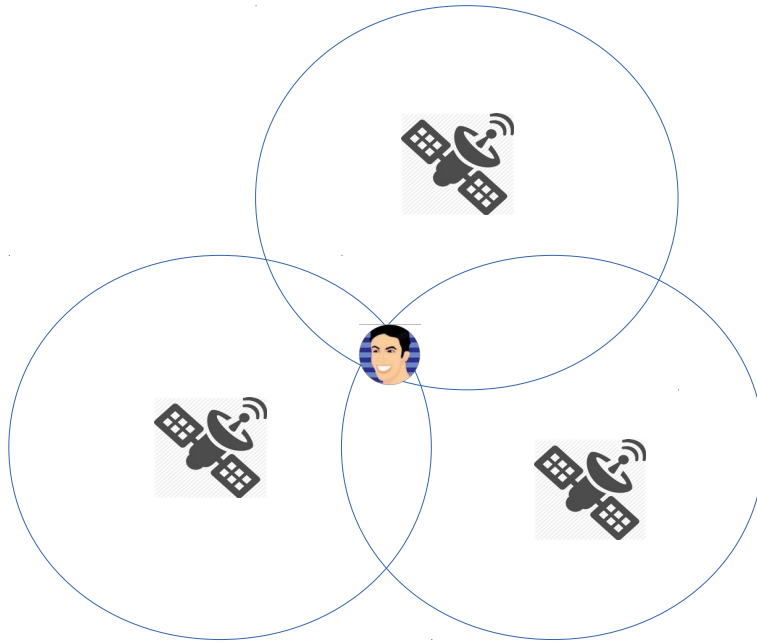
Funcionamento:

- Existem cerca de 24 satélites GPS em órbita na terra (estacionários)
- Em qualquer local da superfície do globo é possível ter visada com no mínimo 4 satélites
- Cada satélite envia um sinal, que é captado pela antena do receptor GPS.
- Através da hora do envio e do recebimento do pacote o aparelho GPS calcula a distância entre o aparelho e o satélite
- A localização geográfica é calculada por meio de triangulação entre o aparelho gps e os satélites

Sistema de Posicionamento Global

Funcionamento:

- Existem cerca de 24 satélites GPS em órbita na terra (estacionários)

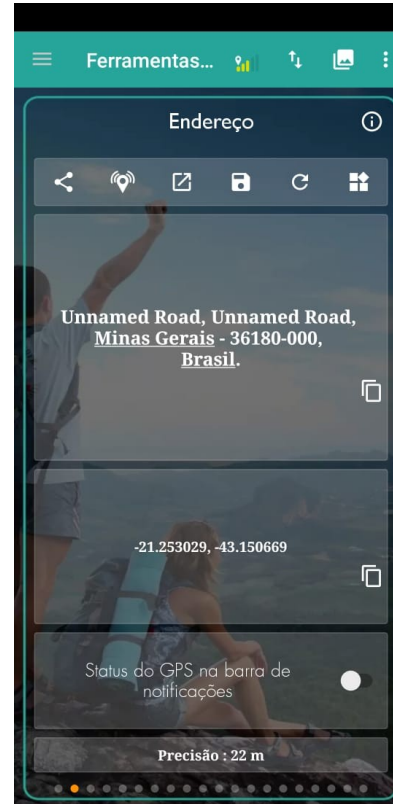


Neste exemplo, precisaria de mais 01 satélite para obter a altitude

Quanto mais satélites, melhor a precisão

Sistema de Posicionamento Global

- App: Ferramentas GPS



Sistema de Posicionamento Global

- Funcionamento não depende da internet
- Capacidade do sistema é ilimitada, pode receber quantos clientes forem
- O satélite não recebe nenhuma informação, apenas envia, já o cliente só recebe

Sistema de Posicionamento Global

- Em áreas abertas a precisão é maior, árvores, prédios e viadutos podem **bloquear** o sinal
- Um aparelho de geolocalização pode ter compatibilidade com diversas fontes (gps, glonass, etc..)
- Em caso de **guerra**, o sinal GPS pode ser **desligado**, mantendo-se apenas o sinal militar

Sistema de Posicionamento Global

Alternativas:

- Obtenção da localização através do endereço IP
 - Não é tão exato mas é possível determinar o país de origem e muitas vezes até a cidade de origem
- Obtenção do ID da Antena da Operadora de Celular
 - Cada antena de celular possui um ID público e sua geolocalização que permitem saber em qual antena o cliente está mais próximo (conectado).
- Triangularização entre antenas da rede de telefonia celular
 - Já foi utilizado para solucionar diversos crimes, mas depende de haver várias torres de celular na região. Funcionamento restrito à empresa de telefonia.

Sistema de Posicionamento Global

- Para utilizar GPS no aplicativo Android é necessário primeiramente obter a permissão de acesso, via **Android Manifest** e mais recentemente em tempo de execução.

Permissões:

- **ACCESS_COARSE_LOCATION**
 - Acessar localização de pouca precisão (exemplo: ip, torre de telefonia, wifi, etc...)
- **ACCESS_FINE_LOCATION**
 - Acessar localização precisa, geolocalização
 - Sempre solicitar as duas permissões

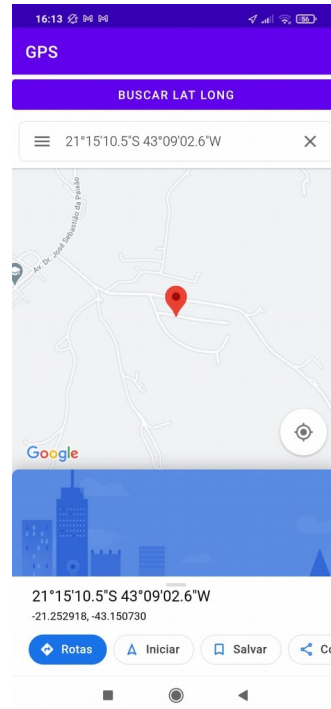
Sistema de Posicionamento Global

GPS NO ANDROID

- **LocationListener**
 - Recebe informações do **LocationManager** sobre mudanças na localização do dispositivo
- **onLocationChanged(Location location)**
 - Método da classe **LocationListener** acionado sempre que a localização é alterada

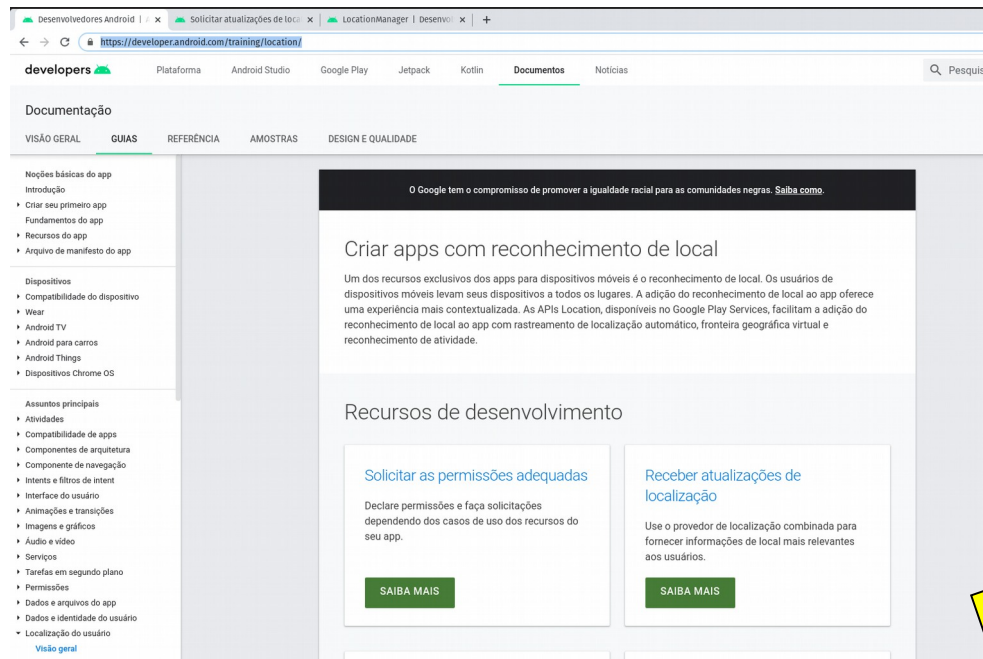
Atividade

- Criar uma aplicação Android que apresente em um mapa a geolocalização atual do usuário.



Atividade

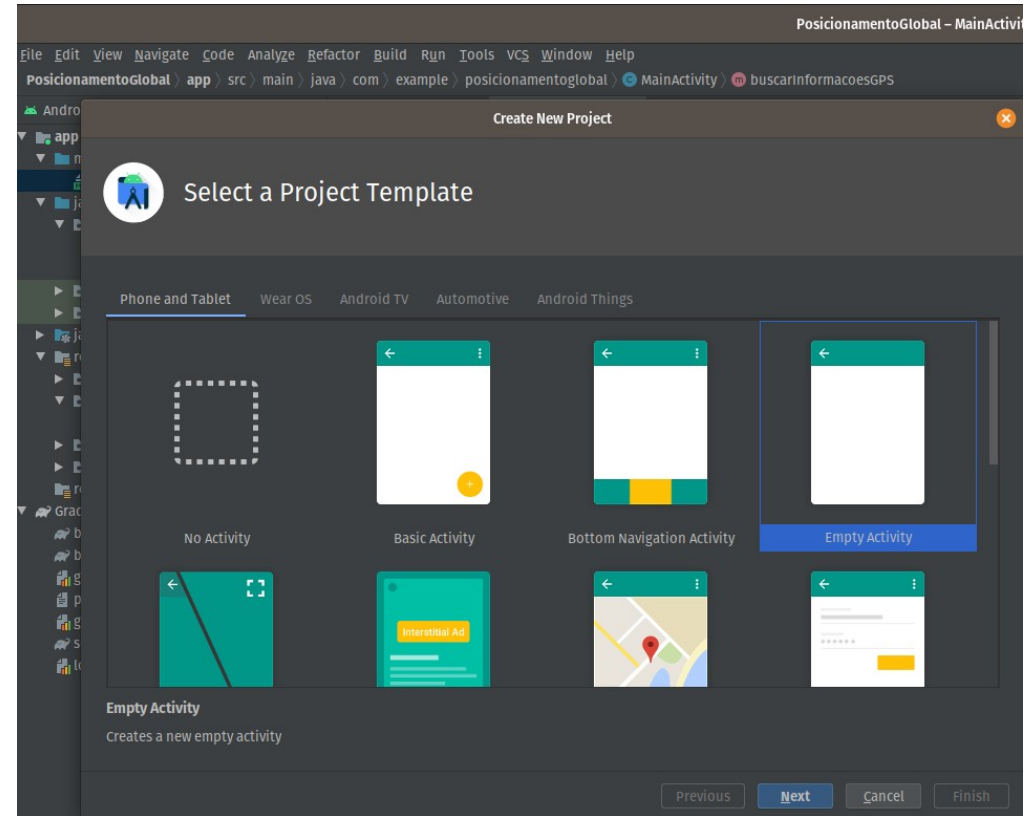
- Criar uma aplicação Android que apresente em um mapa a geolocalização atual do usuário.



<https://developer.android.com/training/location/>

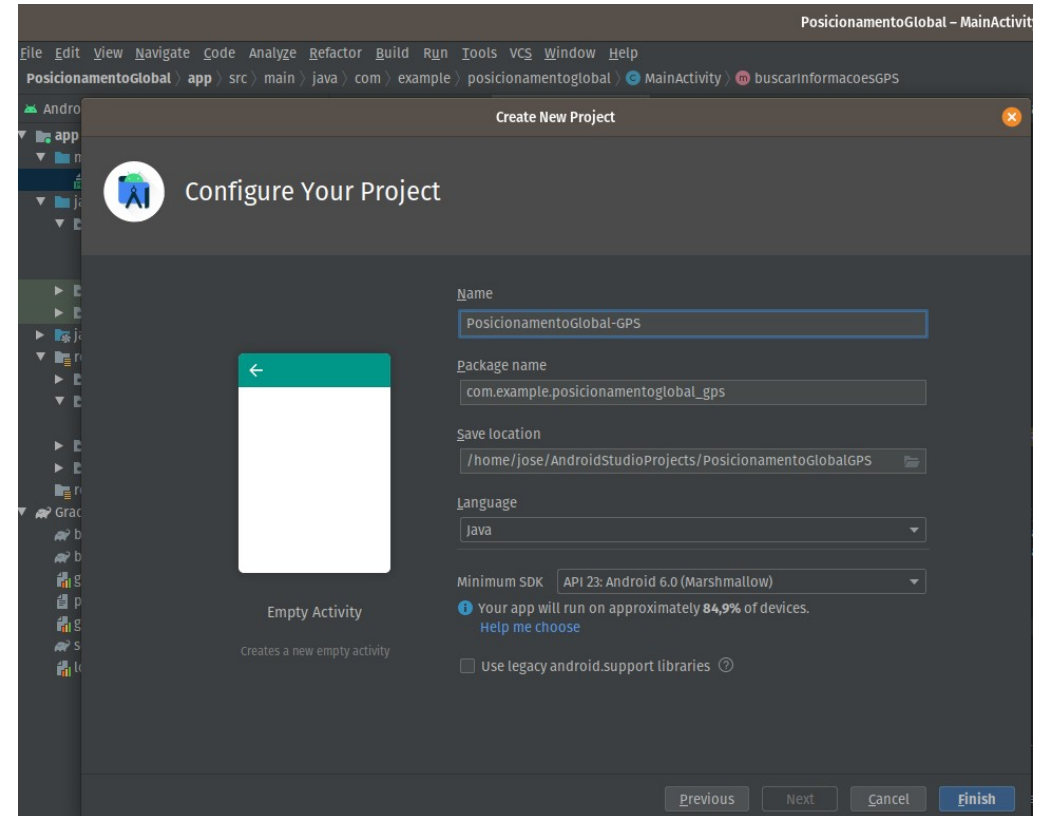
Atividade

- Criar uma aplicação Android que apresente em um mapa a geolocalização atual do usuário.
1. Abra o AndroidStudio
 2. Crie um novo projeto
 3. Escolha uma Activity em branco (EmptyActivity)



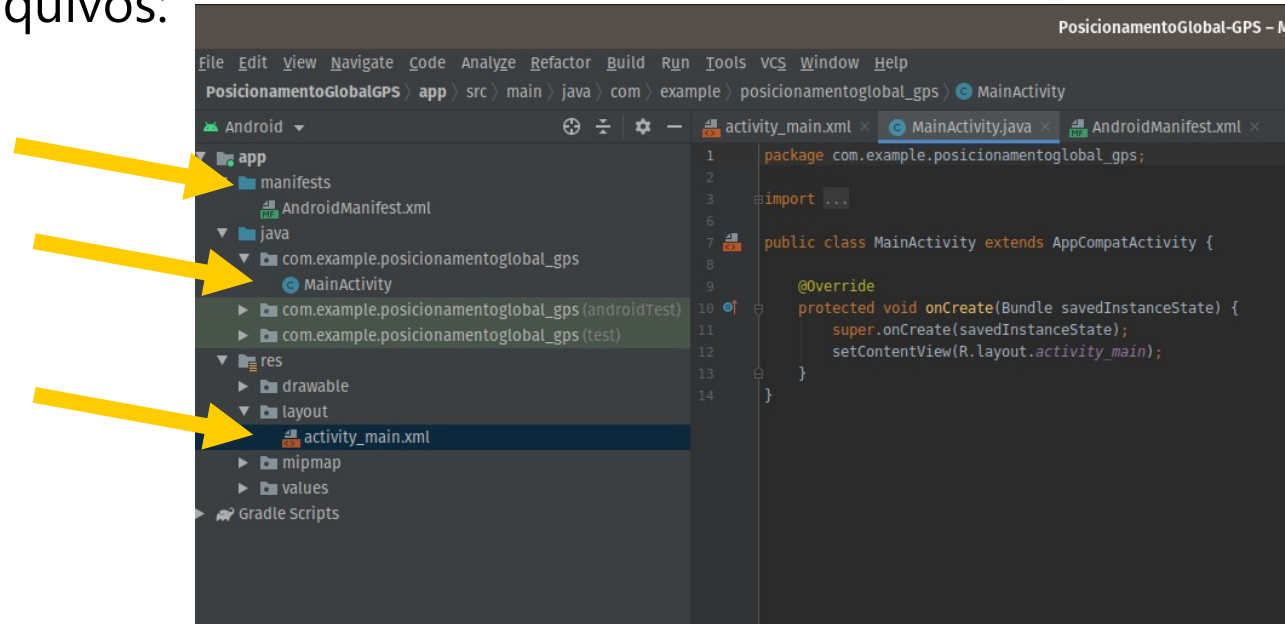
Atividade

- Criar uma aplicação Android que apresente em um mapa a geolocalização atual do usuário.
1. Abra o AndroidStudio
 2. Crie um novo projeto
 3. Escolha uma Activity em branco (EmptyActivity)
 4. Dê um nome para seu projeto
 - i. Ex: *PosicionamentoGlobal-GPS*
 5. Finish



Atividade

- Pronto!
- Projeto criado...
- Perceba os três principais arquivos:
 - AndroidManifest.xml
 - Configurações e permissões
 - MainActivity.java
 - programação
 - activity_main.xml
 - Interface gráfica



Atividade – Permissão para usar GPS e internet

- Via **Android Manifest** e na **função** que o chamará
- ACCESS_COARSE_LOCATION (Baixa precisão) ou ACCESS_FINE_LOCATION (Alta precisão)

AndroidManifest.xml

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

Atividade – Permissão para usar GPS e internet

- Via **Android Manifest** e na **função** que o chamará
- ACCESS_COARSE_LOCATION (Baixa precisão) ou ACCESS_FINE_LOCATION (Alta precisão)

AndroidManifest

<uses-permission
<uses-permission

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3     package="com.example.posicionamentoglobal_gps">
4
5     <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
6     <uses-permission android:name="android.permission.INTERNET" />
7     <application
8         android:allowBackup="true"
9         android:icon="@mipmap/ic_launcher"
10        android:label="PosicionamentoGlobal-GPS"
11        android:roundIcon="@mipmap/ic_launcher_round"
12        android:supportRtl="true"
13        android:theme="@style/Theme.PosicionamentoGlobalGPS">
14        <activity android:name=".MainActivity">
15            <intent-filter>
16                <action android:name="android.intent.action.MAIN" />
17
18                <category android:name="android.intent.category.LAUNCHER" />
19            </intent-filter>
20        </activity>
21    </application>
22
23 </manifest>
```

ACCESS_FINE_LOCATION" />
>

Atividade – Layout da aplicação

- Adicionar uma **webview**
 - Que carregara um link do google maps com as coordenadas X,Y.

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="match_parent"
    tools:context=".MainActivity">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">
        <Button
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="BUSCAR LAT LONG"
            android:onClick="buscarInformacoesGPS"/>
        <WebView
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:id="@+id/webv" />
    </LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

Atividade – Layout da aplicação

- Adicionar uma **webview**
 - Que carregara um link do google maps com as coordenadas X,Y.

Layout

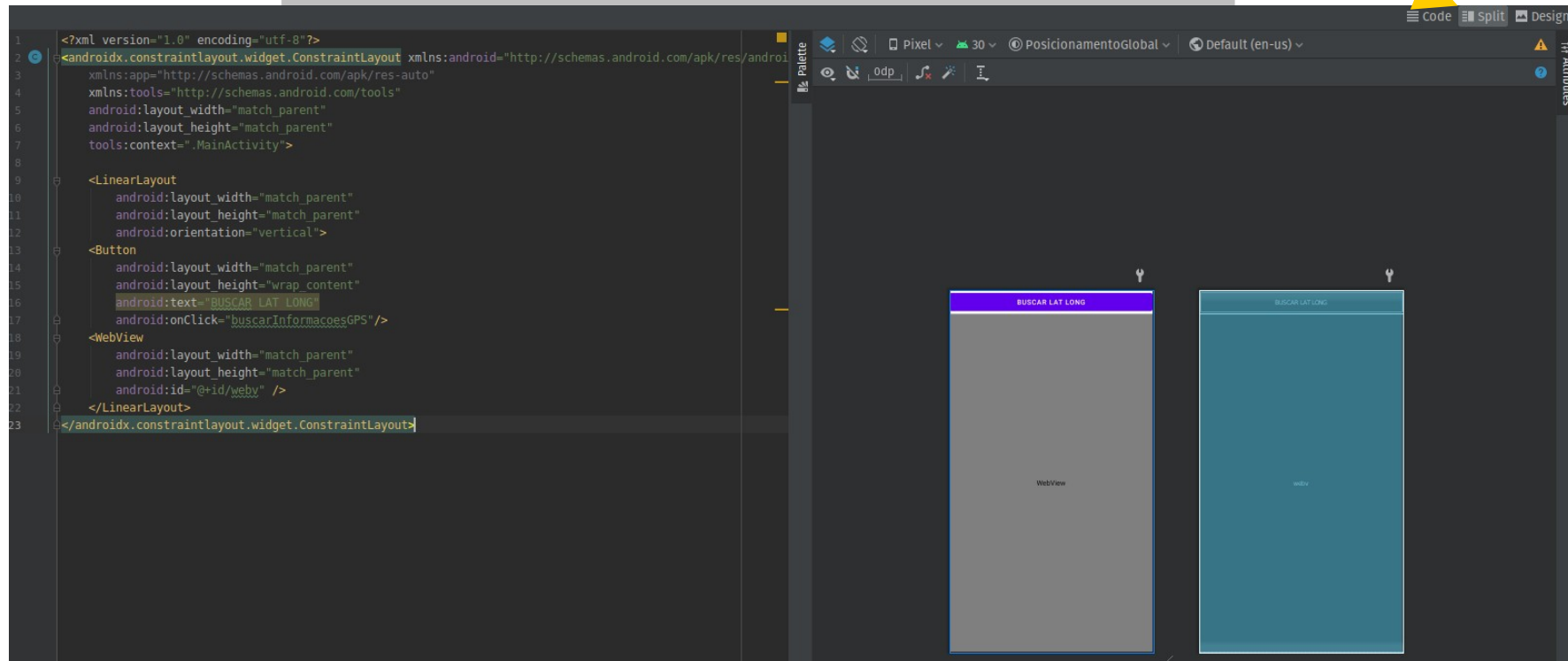
```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
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

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">
        <Button
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="BUSCAR LAT LONG"
            android:onClick="buscarInformacoesGPS"/>
        <WebView
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:id="@+id/webv" />
    </LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

E um botão para chamar a função!

Atividade – Layout da aplicação

Clicando em SPLIT você pode ter uma prévia de como esta ficando sua interface



Atividade - MinhaLocalizacaoListener

- Criar a classe: `MinhaLocalizacaoListener` que implements *LocationListener*
 - Que irá tratar os eventos de mudança da geolocalização.

Atividade - MinhaLocalizacaoListener

- Criar a classe: `MinhaLocalizacaoListener` que implements *LocationListener*
 - Que irá tratar os eventos de mudança da geolocalização.
- Adicionaremos dois atributos estáticos:
 - `static double` Latitude;
 - `static double` Longitude

Atividade - MinhaLocalizacaoListener

- Criar a classe: `MinhaLocalizacaoListener` que implements *LocationListener*
 - Que irá tratar os eventos de mudança da geolocalização.
- Adicionaremos dois atributos estáticos:
 - `static double` Latitude;
 - `static double` Longitude
- Esta classe será parâmetro da função **`requestLocationUpdates`**

Get Latitude e Longitude - LocationManager

- Classe **LocationManager**
- Método **requestLocationUpdates**

Get Latitude e Longitude - LocationManager

- Classe **LocationManager**
- Método **requestLocationUpdates**
- recebe como parâmetro:
 - Provedor : GPS
 - Intervalo de tempo de atualização: 0 ms
 - Intervalo em metros de atualização: 0 m
 - LocationListener: [MinhaLocalizacaoListener](#)

Get Latitude e Longitude - LocationManager

- Classe **LocationManager**
- Método **requestLocationUpdates**
- recebe como parâmetro:
 - Provedor : GPS
 - Intervalo de tempo de atualização: 0 ms
 - Intervalo em metros de atualização: 0 m
 - LocationListener: [MinhaLocalizacaoListener](#)

Com estes dados, a geolocalização será atualizada sempre que houver qualquer mudança, não havendo limite em metros ou intervalo de tempo

Classe *MyLocationListener*

MinhaLocalizacaoListener.java

```
package com.example.gps;
import android.location.Location;
import android.location.LocationListener;
import android.os.Bundle;

public class MinhaLocalizacaoListener implements LocationListener {
    public static double latitude;
    public static double longitude;

    @Override
    public void onLocationChanged(Location location) {
        location.getLatitude();
        location.getLongitude();
        this.latitude = location.getLatitude();
        this.longitude = location.getLongitude();
    }

    @Override
    public void onProviderDisabled(String provider) {
        // TODO Auto-generated method stub
    }

    @Override
    public void onProviderEnabled(String provider) {
        // TODO Auto-generated method stub
    }

    @Override
    public void onStatusChanged(String provider, int status, Bundle extras) {
        // TODO Auto-generated method stub
    }
}
```


Classe *MainActivity*

MainActivity

```
1 public void buscarInformacoesGPS(View v) {
2     LocationManager mLocManager = null;
3     LocationListener mLocListener;
4     mLocManager = (LocationManager) getSystemService(MainActivity.this.LOCATION_SERVICE);
5     mLocListener = new MinhaLocalizacaoListener();
6
7     mLocManager.requestLocationUpdates(LocationManager.GPS_PROVIDER, 0, 0, mLocListener);
8     if (mLocManager.isProviderEnabled(LocationManager.GPS_PROVIDER)) {
9         String texto = "Latitude: " + MinhaLocalizacaoListener.latitude + "\n" +
10             "Longitude: " + MinhaLocalizacaoListener.longitude + "\n";
11         Toast.makeText(MainActivity.this, texto, Toast.LENGTH_LONG).show();
12     } else {
13         Toast.makeText(MainActivity.this, "GPS DESABILITADO.", Toast.LENGTH_LONG).show();
14     }
15
16     this.mostrarGoogleMaps(MinhaLocalizacaoListener.latitude, MinhaLocalizacaoListener.longitude);
17 }
```

```
if (ActivityCompat.checkSelfPermission(this,
    Manifest.permission.ACCESS_FINE_LOCATION) != PackageManager.PERMISSION_GRANTED
    && ActivityCompat.checkSelfPermission(this,
    Manifest.permission.ACCESS_COARSE_LOCATION) != PackageManager.PERMISSION_GRANTED) {
    ActivityCompat.requestPermissions(MainActivity.this,
        new String[]{
            Manifest.permission.ACCESS_FINE_LOCATION
        }, 1);
    ActivityCompat.requestPermissions(MainActivity.this,
        new String[]{
            Manifest.permission.ACCESS_NETWORK_STATE
        }, 1);
    return;
}
```

Antes do requestLocationUpdates, tem que verificar se tem permissão para usar o GPS e internet



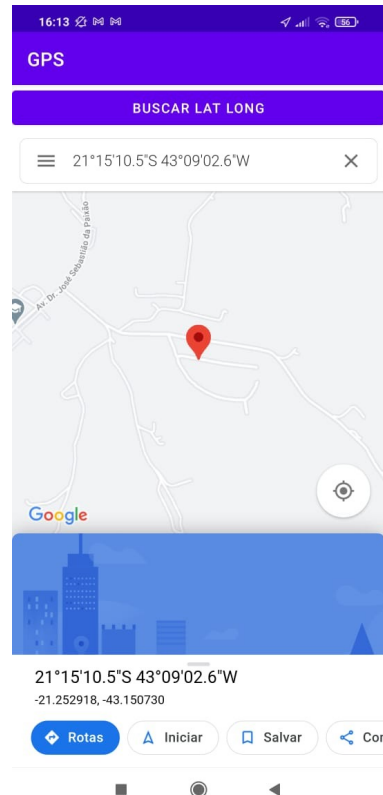
```
6
7 mLocManager.requestLocationUpdates(LocationManager.GPS_PROVIDER, 0, 0, mLocListener);
8 if (mLocManager.isProviderEnabled(LocationManager.GPS_PROVIDER)) {
9     String texto = "Latitude: " + MinhaLocalizacaoListener.latitude + "\n" +
10         "Longitude: " + MinhaLocalizacaoListener.longitude + "\n";
11     Toast.makeText(MainActivity.this, texto, Toast.LENGTH_LONG).show();
12 } else {
13     Toast.makeText(MainActivity.this, "GPS DESABILITADO.", Toast.LENGTH_LONG).show();
14 }
15
16 this.mostrarGoogleMaps(MinhaLocalizacaoListener.latitude, MinhaLocalizacaoListener.longitude);
17 }
```

Visualizando

- Abrir o link com as coordenadas conhecidas:
 - <https://www.google.com/maps/search/?api=1&query=latitude,longitude>
- Lembre-se de que é necessário solicitar permissões para acessar a internet.

```
1 public void mostrarGoogleMaps(double latitude, double longitude){  
2     WebView wv = findViewById(R.id.webv);  
3     wv.getSettings().setJavaScriptEnabled(true);  
4     wv.loadUrl("https://www.google.com/maps/search/?api=1&query=" +  
5         latitude + "," + longitude);  
7 }
```

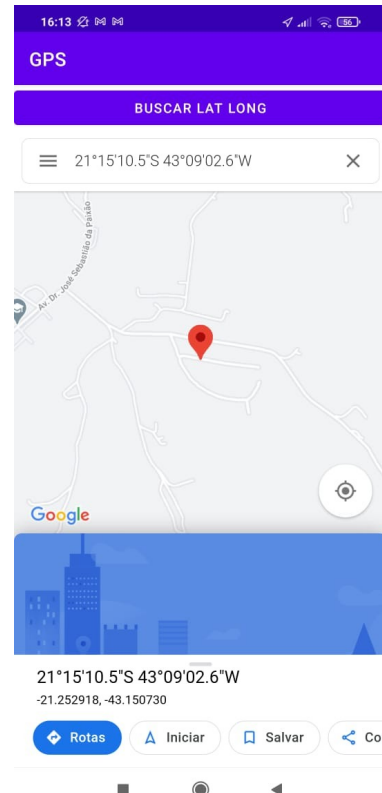

App funcionando ok!



App funcionando ok!

- Comentários

- Eu testei no meu celular físico
- Se você for testar no emulador, o GPS não vai funcionar dentro de casa
- Vai te reportar sempre Lat=0.0 e long=0.0
- Uma solução:
 - Troque o provedor para:
`locationManager.NETWORK_PROVIDER`
 - Linhas 7 e 8, do método `buscarInformacoesGPS`



Exercício de fixação

- Agora é a sua vez:
 - Tente fazer os passos aqui apresentados
 - Uma vez que você conseguiu replicar, tente alterar algumas coisas
 - Dê uma olhadinha na documentação da classe **locationManager**
 - <https://developer.android.com/reference/android/location/LocationManager>
 - Veja se dentre as funções disponíveis você tem alguma ideia de app
 - Treine a vontade!
 - Uma outra classe interessante é a **Location**
 - Com ela faremos a distância entre dois ponto(distanceBetween) na próxima aula
 - Fique a vontade para ir pensando na solução
 - <https://developer.android.com/reference/android/location/Location>