

FUNDAÇÃO INSTITUTO DE EDUCAÇÃO DE BARUERI

Instituto Técnico de Barueri “Brasília Flores de Azevedo”

Tecnologias Emergentes

Tutorial Geolocalização

versão 1.0

Prof. Adriano Domingues

2019

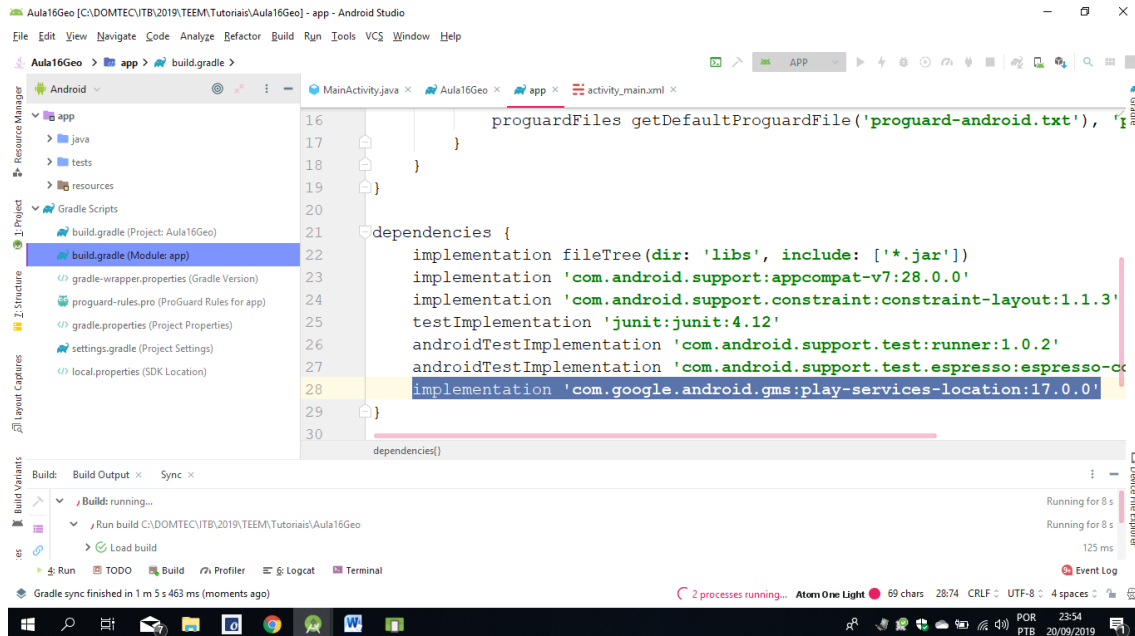
AULA 17 – Geolocalização Solicitando Permissão

Vamos continuar no projeto Aula16Geo:

Adicione a dependência abaixo no build.gradle (Module: app):

```
implementation 'com.google.android.gms:play-services-location:17.0.0'
```

Sincronize o projeto e aguarde o download da biblioteca no projeto:



Esse erro ocorrerá:

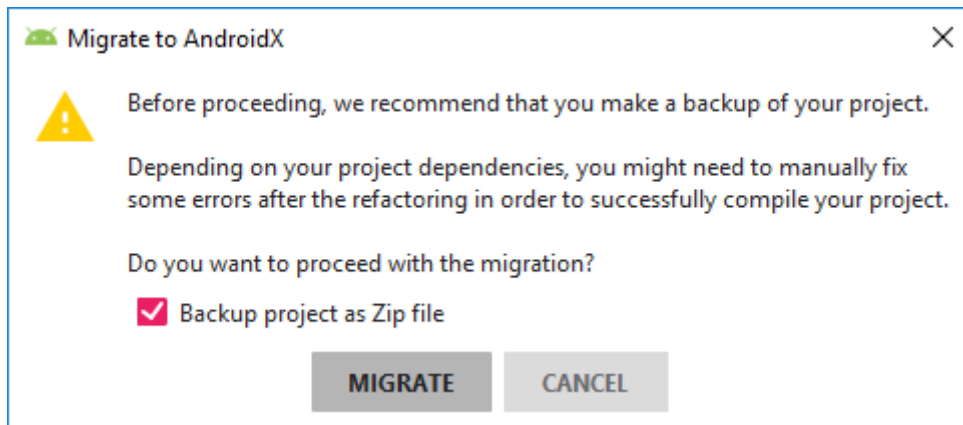
Manifest merger failed : Attribute application@appComponentFactory value=(android.support.v4.app.CoreComponentFactory) from [com.android.support:support-compat:28.0.0] AndroidManifest.xml:22:18-91

is also present at [androidx.core:core:1.0.0] AndroidManifest.xml:22:18-86 value=(androidx.core.app.CoreComponentFactory).

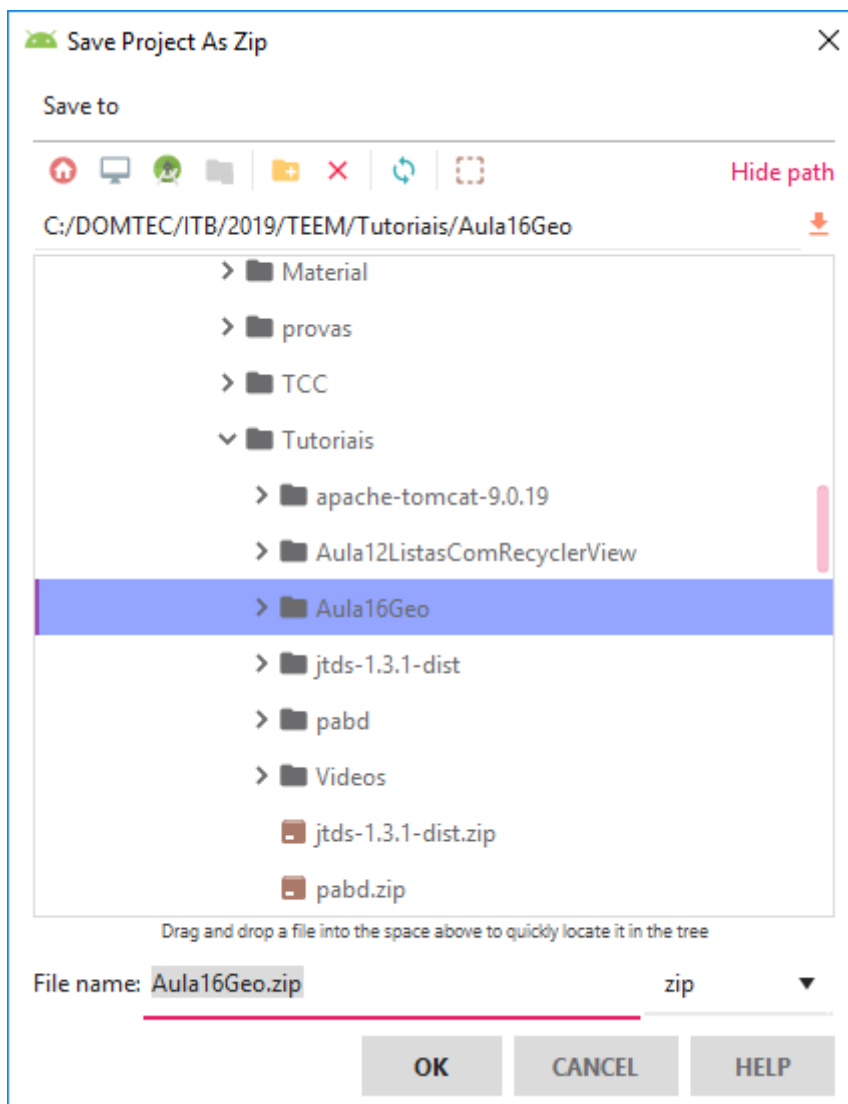
Suggestion: add 'tools:replace="android:appComponentFactory"' to <application> element at AndroidManifest.xml:9:5-23:19 to override.

Clique no menu Refactor > Migrate to AndroidX

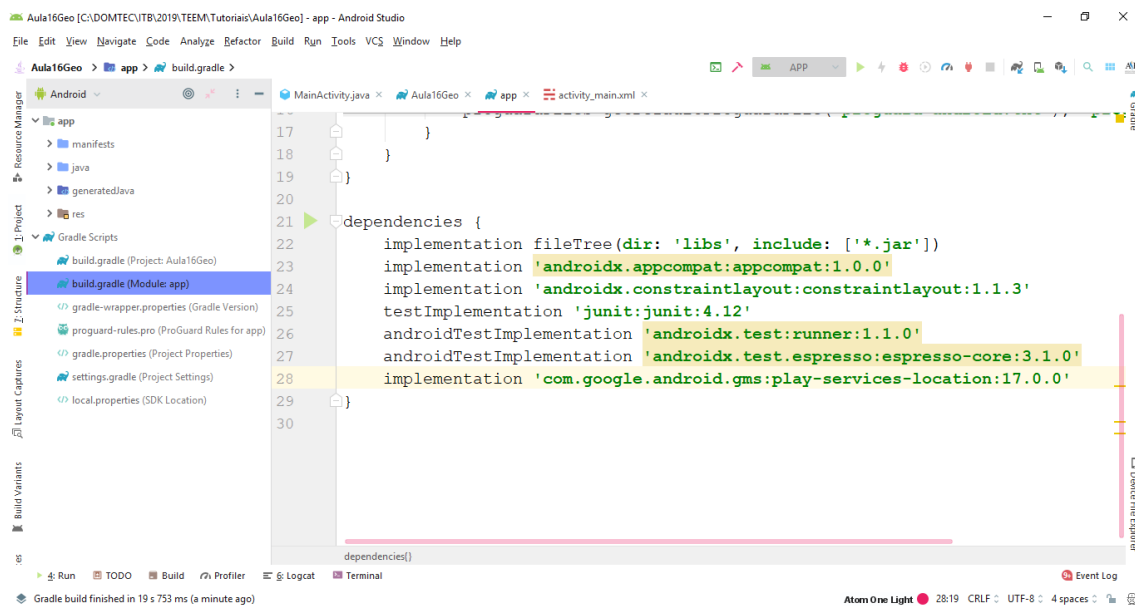
Confirme a mensagem abaixo para realizar uma cópia de segurança e iniciar a migração do sistema tradicional para o AndroidX.



Escolha o local de salvamento:

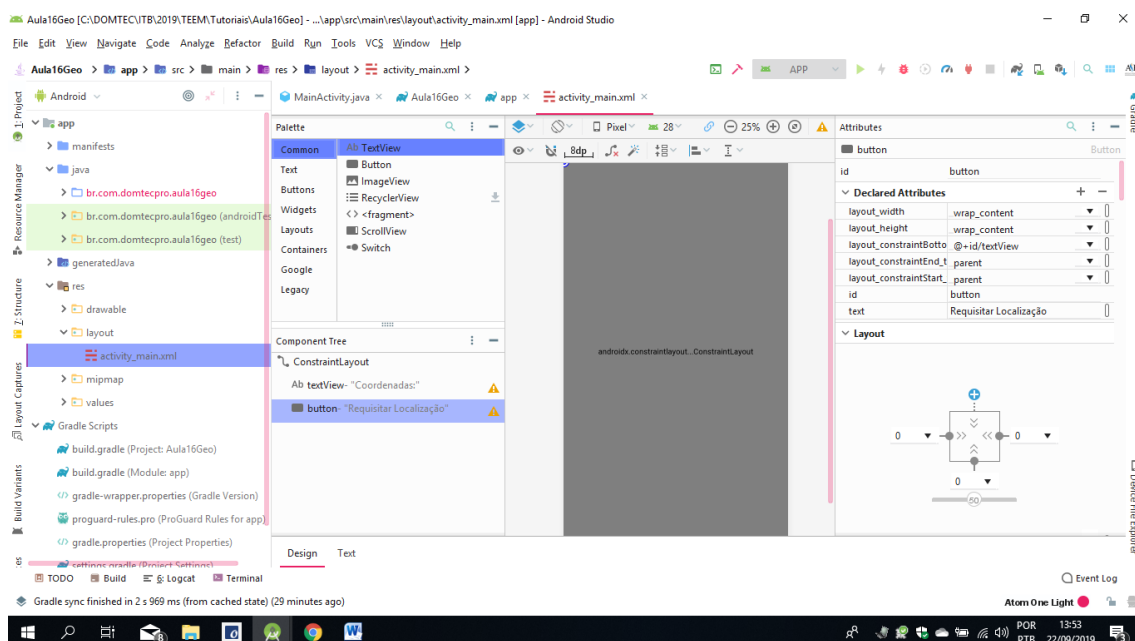


Após a inspeção do código clique no botão DO REFACTOR.

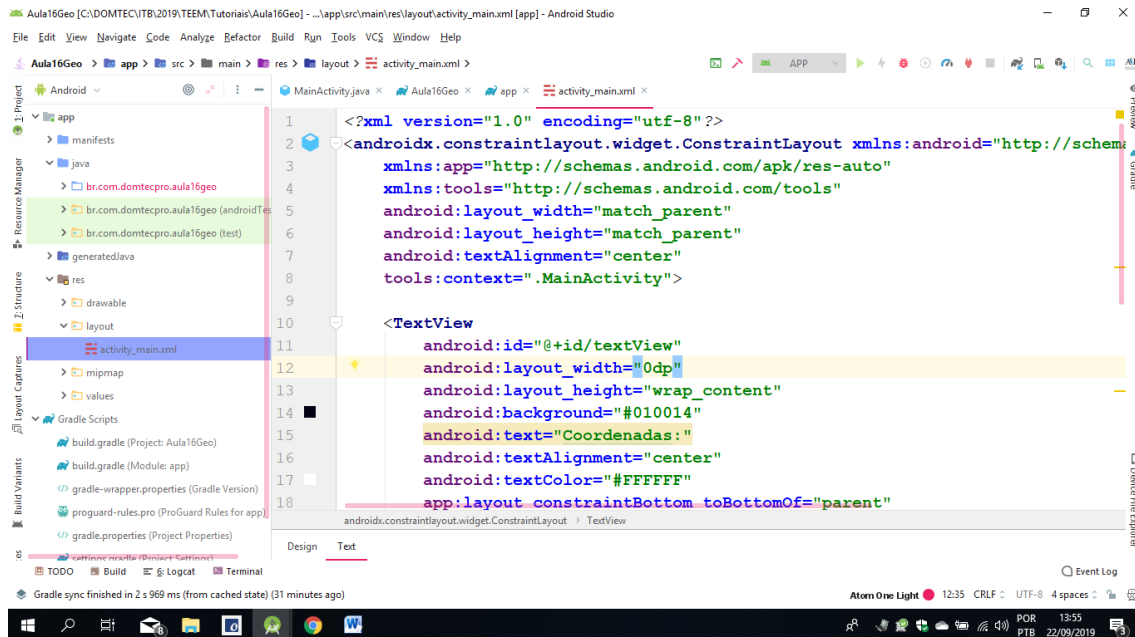


Compare as alterações realizadas, as bibliotecas androidx são as últimas atualizações do Google Android, e garante compatibilidade com os componentes anteriores, se você pretende utilizar alguma biblioteca nova provavelmente fará esta migração.

Vamos abrir o layout (activity_main.xml) para adicionar outro objeto:

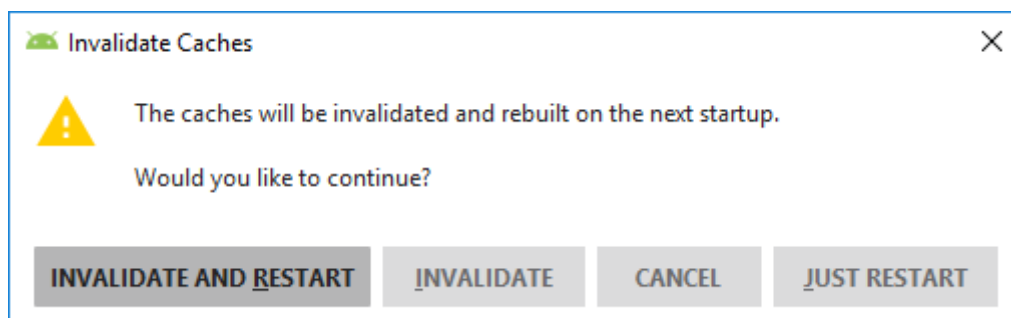


Observe que o layout não foi exibido adequadamente, então vamos abrir o código XML e efetuar alguns ajustes:



Na linha 12 substitua o valor 0dp por match_parent em android:layout_width.

Clique em File > Invalidate Caches / Restart



Clique em INVALIDATE AND RESTART e aguarde a reinicialização.

Na volta, clique em Build > Clean Project e ajuste o layout.

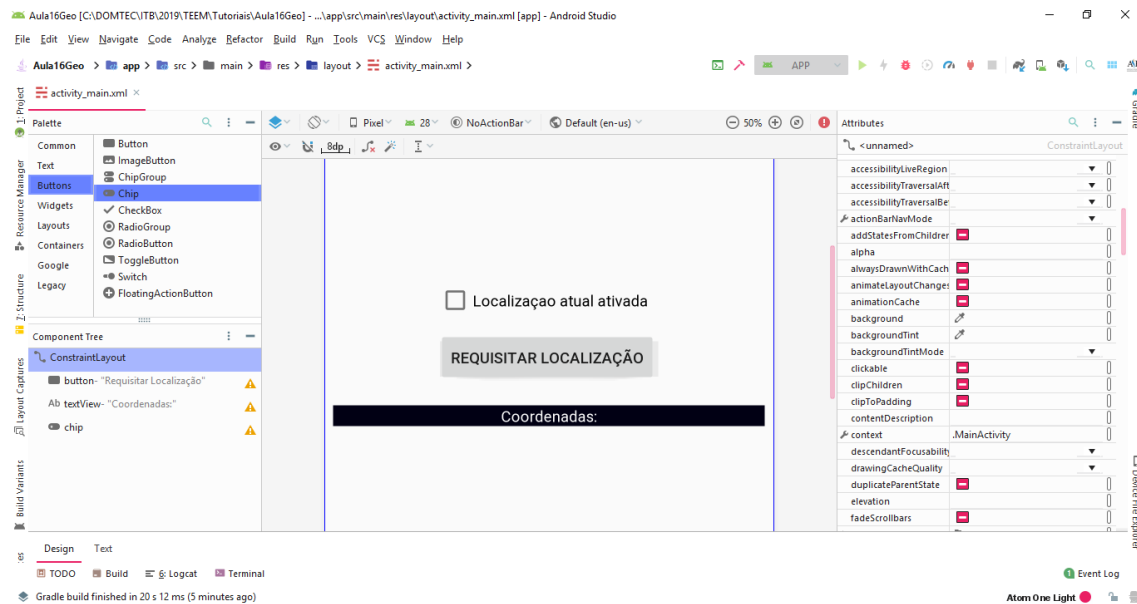
Adicione o objeto abaixo ao layout:

```

<com.google.android.material.chip.Chip
    android:id="@+id/chip"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginEnd="8dp"
    android:layout_marginBottom="12dp"
    android:text="Localização atual ativada"
    app:layout_constraintBottom_toTopOf="@+id/button"
    app:layout_constraintEnd_toEndOf="@+id/button" />

```

Ficando assim:



Na linha 26 adicione as linhas:

```
LocationRequest mLocationRequest;
private static final int REQUEST_CHECK_SETTINGS = 613;
Chip chip;
```

Para declaração dos objetos mLocationRequest, chip (para ativar localização atual) e da constante REQUEST_CHECK_SETTINGS

Abaixo da linha 37 vamos criar o objeto chip e preparar a ação de clique dele:

```
chip = findViewById(R.id.chip);
chip.setText("Localização Ativa");
chip.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

    }
});
```

Abaixo da linha 120, antes de fechar a chave (}) da classe insira o código do método abaixo:

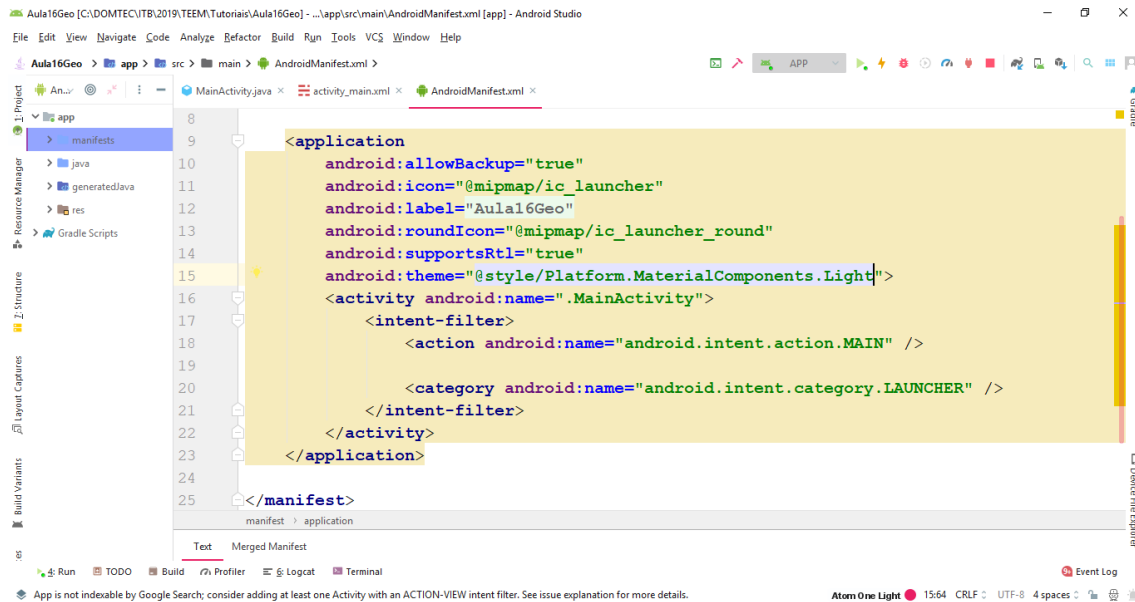
```
private void solicitarPermissao() {
    LocationSettingsRequest.Builder builder = new
    LocationSettingsRequest.Builder().addLocationRequest (mLocationReques
    t);
    SettingsClient client =
    LocationServices.getSettingsClient(this);
    Task<LocationSettingsResponse> task =
    client.checkLocationSettings(builder.build());

    task.addOnSuccessListener(new
    OnSuccessListener<LocationSettingsResponse>() {
        @Override
        public void onSuccess(LocationSettingsResponse
        locationSettingsResponse) {
            Toast.makeText(MainActivity.this, "Location is already
            on", Toast.LENGTH_SHORT).show();
        }
    });

    task.addOnFailureListener(new OnFailureListener() {
        @Override
        public void onFailure(@NonNull Exception e) {
            if (e instanceof ResolvableApiException) {
                try {
                    ResolvableApiException resolvable =
                    (ResolvableApiException) e;
                    resolvable.startResolutionForResult(MainActivity.this,
                    REQUEST_CHECK_SETTINGS);
                } catch (IntentSender.SendIntentException ignored) {
                }
            }
        }
    });
}
```

Agora de volta na linha 54 insira a chamada do método `solicitarPermissao()`

Troque o tema do layout (activity_main.xml) como descrito na linha 15 abaixo:



Na linha 160 adicione o método:

```
@Override
protected void onActivityResult(int requestCode, int resultCode,
Intent data) {
    if (requestCode == REQUEST_CHECK_SETTINGS) {
        switch (resultCode) {
            case Activity.RESULT_OK:
                Toast.makeText(this, "Location is now on",
                    Toast.LENGTH_SHORT).show();
                break;
            case Activity.RESULT_CANCELED:
                Toast.makeText(this, "User didn't allowed to change
location settings", Toast.LENGTH_SHORT).show();
                break;
        }
    }
}
```

E abaixo o método:

```
protected void createLocationRequest() {
    mLocationRequest = new LocationRequest();
    mLocationRequest.setInterval(10000);
    mLocationRequest.setFastestInterval(5000);

    mLocationRequest.setPriority(LocationRequest.PRIORITY_HIGH_ACCURACY);
}
```


Após um breve teste vamos alterar a chamada do objeto chip:

```
chip.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        if (!chip.isCheckedIconVisible()) {  
            createLocationRequest();  
            solicitarPermissao();  
        } else {  
            negarPermissao();  
        }  
    }  
});
```

Agora vamos criar o método `negarPermissao()`, que deve estar em vermelho acima:

```
private void negarPermissao() {  
    locationManager.removeUpdates(locationListener);  
}
```

Vamos agora alterar o método `configureButton()` da aula anterior, para carregar o chip como ativo, assim que a permissão for concedida:

```
private void configureButton() {  
    button.setOnClickListener(new View.OnClickListener() {  
        @Override  
        public void onClick(View v) {  
            try {  
                locationManager.requestLocationUpdates("gps", 5000,  
                    0, locationListener);  
                chip.setCheckedIconVisible(true);  
            } catch (SecurityException e) {  
                e.printStackTrace();  
            }  
        }  
    });  
}
```

Teste a aplicação, conceda as permissões e verifique que após solicitar as coordenadas, ao inativar o chip a captura da localização é interrompida.

Referência Bibliográfica

- [1] <http://developer.android.com>. Acessado em 28/04/2019.
- [2] https://www.youtube.com/watch?v=QNb_3QKSmMk. Acessado em 19/09/2019.
- [3] <https://imasters.com.br/android/requisitando-localizacao-usuario-da-forma-correta-no-android>. Acessado em 20/09/2019.