

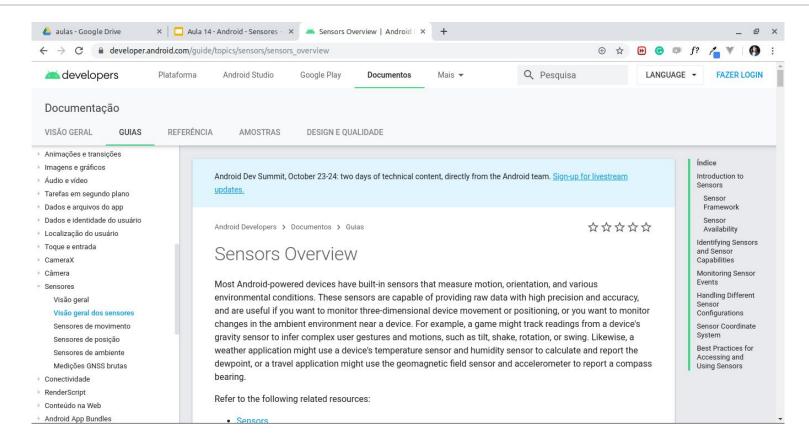
Desenvolvimento de Software para Dispositivos Móveis

Aula 14 - Sensores

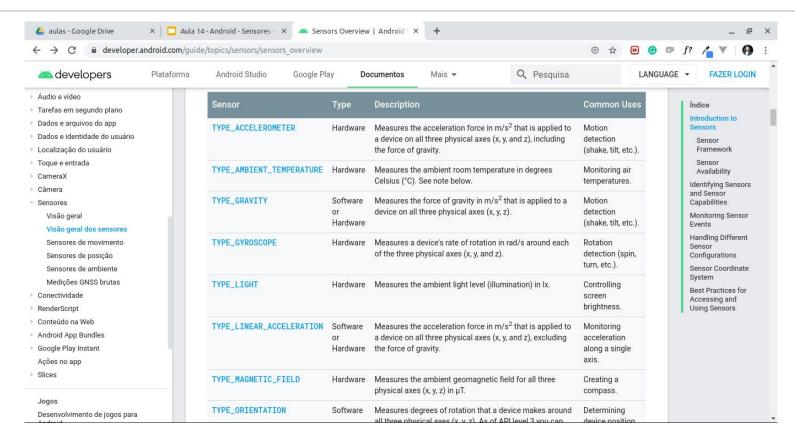


Professor: Anderson Almada

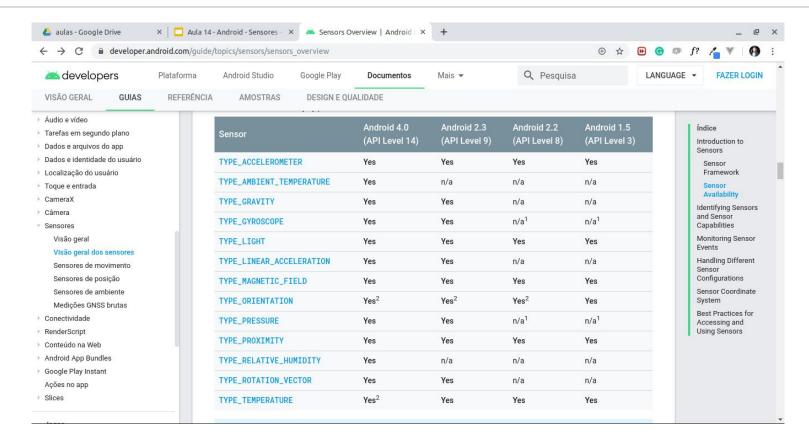
Introdução



Introdução



Introdução



- Vamos utilizar um gerenciador de sensores
 - SensorManager

Dentro do onCreate

```
sensorManager = (SensorManager) getSystemService(Context. SENSOR SERVICE);
```

 Para listar todos os sensores do dispositivo, basta chamar o método getSensorList.

```
List<Sensor> listSensors = sensorManager.getSensorList(Sensor. TYPE_ALL);
```

Usando um foreach, liste todos:

```
for (Sensor s : listSensors) {
   Log.i("Main", s.getStringType());
}
```

Utilizando o sensor de luminosidade

```
sensorLight = sensorManager.getDefaultSensor(Sensor. TYPE_LIGHT);
```

- Para iniciar a coleta de dados, implemente a interface de escuta do sensor
 - SensorEventListener

 Depois de adicionar os métodos, registre o escutador no onResume e desregistre o escutador no onPause

```
onResume()
sensorManager.registerListener(this, sensorLight, SensorManager.SENSOR_DELAY_NORMAL);
onPause()
sensorManager.unregisterListener(this);
```

 O método onSensorChanged sempre é chamado quando os dados são capturados pelos sensores. Logo, ele é o método a ser explorado

```
public void onSensorChanged(SensorEvent event) {
   Log.i("Main", ""+event.values[0]);
}
```

Exercício

 Sempre que a luminosidade for menor que 5, ligue o flash do dispositivo, caso contrário deixe desligado.

• Realize a importação do dexter

• Build.gradle

```
o implementation 'com.karumi:dexter:6.0.0'
```

Clique em sync

Adicione as permissões no manifest

```
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS COARSE LOCATION" />
```

No onCreate

}).check();

```
Dexter. with Activity (this)
       .withPermissions(
               Manifest.permission. ACCESS COARSE LOCATION,
               Manifest.permission. ACCESS FINE LOCATION
       ).withListener(new MultiplePermissionsListener() {
 @Override public void onPermissionsChecked(MultiplePermissionsReport report) {
             @Override
                                                public
                                                                               void
onPermissionRationaleShouldBeShown(List<PermissionRequest>
                                                                     permissions,
PermissionToken token) { /* ... */}
```

Crie um método para recuperar a última localização

```
location = LocationServices.getFusedLocationProviderClient(this).getLastLocation();
if (location != null) {
    location.addOnSuccessListener(new OnSuccessListener<Location>() {
        @Override
        public void onSuccess(Location location) {
            Log.i("Main", location.getLatitude() + " " + location.getLongitude());
        }
    });
}
```

 Crie um método para ser notificado quando a localização mudar. Para isso, configure como deseja ser notificado

```
private void startLocationUpdates() {
    // config
    locationRequest = new LocationRequest();
    locationRequest.setPriority(LocationRequest.PRIORITY_HIGH_ACCURACY);
    locationRequest.setInterval(UPDATE_INTERVAL);
    locationRequest.setFastestInterval(FASTEST_INTERVAL);
    // update
```

```
// update
  LocationServices. getFusedLocationProviderClient(this).requestLocationUpdates
  (locationRequest, new LocationCallback() {
     @Override
     public void onLocationResult(LocationResult locationResult) {
         if (locationResult == null) { return; }
         for (Location location : locationResult.getLocations()) {
             if (location != null) { }
  }, Looper.myLooper());
// fim do método startLocationUpdates
```

API Maps

https://www.google.com/maps/search/google+maps/@-3.8063156,-38.6036
 227,13z/data=!3m1!4b1

https://nominatim.openstreetmap.org/reverse?&format=json&lat=-3.806317&l
 on=-38.6036227

Link importante

https://developer.android.com/guide/topics/sensors/sensors_overview



Dúvidas??

E-mail: almada@crateus.ufc.br