



RPG0025 - Lidando com sensores em dispositivos móveis

Kervini Ribeiro da Silva - 202301206073

Polo Centro – Araraquara – SP

Lidando com sensores em dispositivos móveis – 9001 – 4º Semestre

Objetivo da Prática

Criar um App Wearable, utilizando os sensores e recursos físicos do aparelho. O app deve ter um caráter acessível, auxiliando pessoas com deficiência visual.

GITHUB: <https://github.com/Kervini/Mundo4-Nivel3.git>

Procedimentos

Análise

Foi desenvolvido o App para a empresa Doma, cujo o objetivo é aumentar a acessibilidade de seus funcionários através de um app que emita comunicados através de audio.

Conclusão

Devido a nunca ter tido contato com o desenvolvimento mobile, tive que me aprofundar no desenvolvimento mobile nativo para desenvolver esse projeto e ainda conhecer mais sobre o universo do android studio e tudo o que ele pode oferecer para desenvolver para android.

© AudioHelper.java ×

```
1 package com.example.projetoDoma;
2
3 > import ...
4
5
6
7
8
9 public class AudioHelper {
10     private AudioManager audioManager;
11     private final Context _context;
12
13     public AudioHelper(Context context) {
14         this._context = context;
15         if (_context != null) {
16             audioManager = (AudioManager) _context.getSystemService(Context.AUDIO_SERVICE);
17
18             audioManager.registerAudioDeviceCallback(new AudioDeviceCallback() {
19                 @Override
20                 public void onAudioDevicesAdded(AudioDeviceInfo[] addedDevices) {
21                     super.onAudioDevicesAdded(addedDevices);
22                     if(audioOutputAvailable(AudioDeviceInfo.TYPE_BLUETOOTH_A2DP)) {
23                         System.out.println("A Bluetooth headset has just been connected");
24                     }
25                 }
26
27                 @Override
28                 public void onAudioDevicesRemoved(AudioDeviceInfo[] removedDevices) {
29                     super.onAudioDevicesRemoved(removedDevices);
30                     if (!audioOutputAvailable(AudioDeviceInfo.TYPE_BLUETOOTH_A2DP)) {
31                         System.out.println("A Bluetooth headset is no longer connected");
32                     }
33                 }
34             }, handler: null);
35         }
36     }
37
38     public boolean audioOutputAvailable(int type) {
39         if (!_context.getPackageManager().hasSystemFeature(PackageManager.FEATURE_AUDIO_OUTPUT)) {
40             return false;
41         }
42
43         AudioDeviceInfo[] devices = audioManager.getDevices(AudioManager.GET_DEVICES_OUTPUTS);
44         for (AudioDeviceInfo device : devices) {
45             if (device.getType() == type) {
46                 return true;
47             }
48         }
49         return false;
50     }
51 }
```

```

19 public class MainActivity extends AppCompatActivity {
20     private ActivityMainBinding binding;
21     private AudioHelper audioHelper;
22     private MediaPlayer mp;
23     private TextToSpeech textToSpeech;
24
25     @Override
26     protected void onCreate(Bundle savedInstanceState) {
27         super.onCreate(savedInstanceState);
28         EdgeToEdge.enable(this);
29         binding = ActivityMainBinding.inflate(getLayoutInflater());
30         View view = binding.getRoot();
31         setContentView(view);
32         ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -> {
33             Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());
34             v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);
35             return insets;
36         });
37
38         textToSpeech = new TextToSpeech(this, new TextToSpeech.OnInitListener() {
39             @Override
40             public void onInit(int status) {
41
42                 if (status == TextToSpeech.SUCCESS) {
43                     System.out.println("TextToSpeech, Initialization Success");
44                 } else {
45                     System.out.println("TextToSpeech, Initialization Failed");
46                 }
47             }
48         });
49
50         textToSpeech.setLanguage(new Locale("pt", "BR"));
51
52         textToSpeech.speak("Testando um, dois, três.", TextToSpeech.QUEUE_FLUSH, null, "unique_id");
53
54         AudioHelper audioHelper = new AudioHelper(MainActivity.this);
55
56         if(audioHelper.audioOutputAvailable(AudioDeviceInfo.TYPE_BUILTIN_SPEAKER)){
57             System.out.println("Auto-falante disponível!");
58         }
59
60         if(audioHelper.audioOutputAvailable(AudioDeviceInfo.TYPE_BLUETOOTH_A2DP)){
61             System.out.println("Headset bluetooth conectado!");
62         }
63     }

```