

Nome: **Saulo Henrique dos Santos**

Matrícula: **202208545378**

Curso: **Desenvolvimento Full Stack**

Campus: **POLO INDAIATUBA - SP**

Github: **<https://github.com/SauloHenriqueSantos/M4N3>**



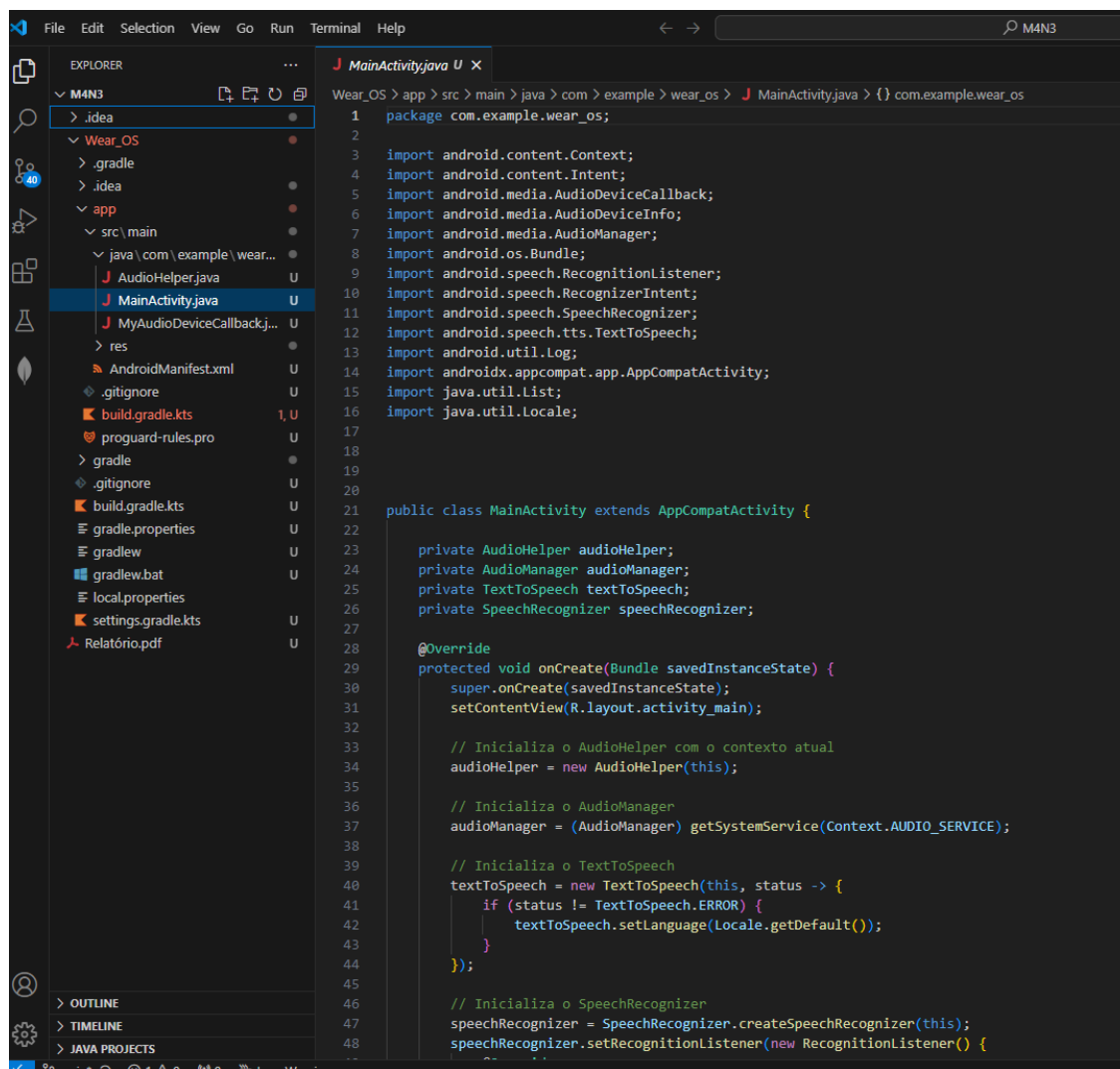
Missão Prática | Mundo 4 | Nível 3

RPG0025 - Lidando com sensores em dispositivos móveis

Objetivo:

1. Instalação do Android Studio e do emulador;
2. Criar um app para Wear OS;
3. Executar um app no emulador;
4. Fazer capturas de telas no Android Studio;
5. Fazer capturas de telas com app complementar.

Resultado Final :

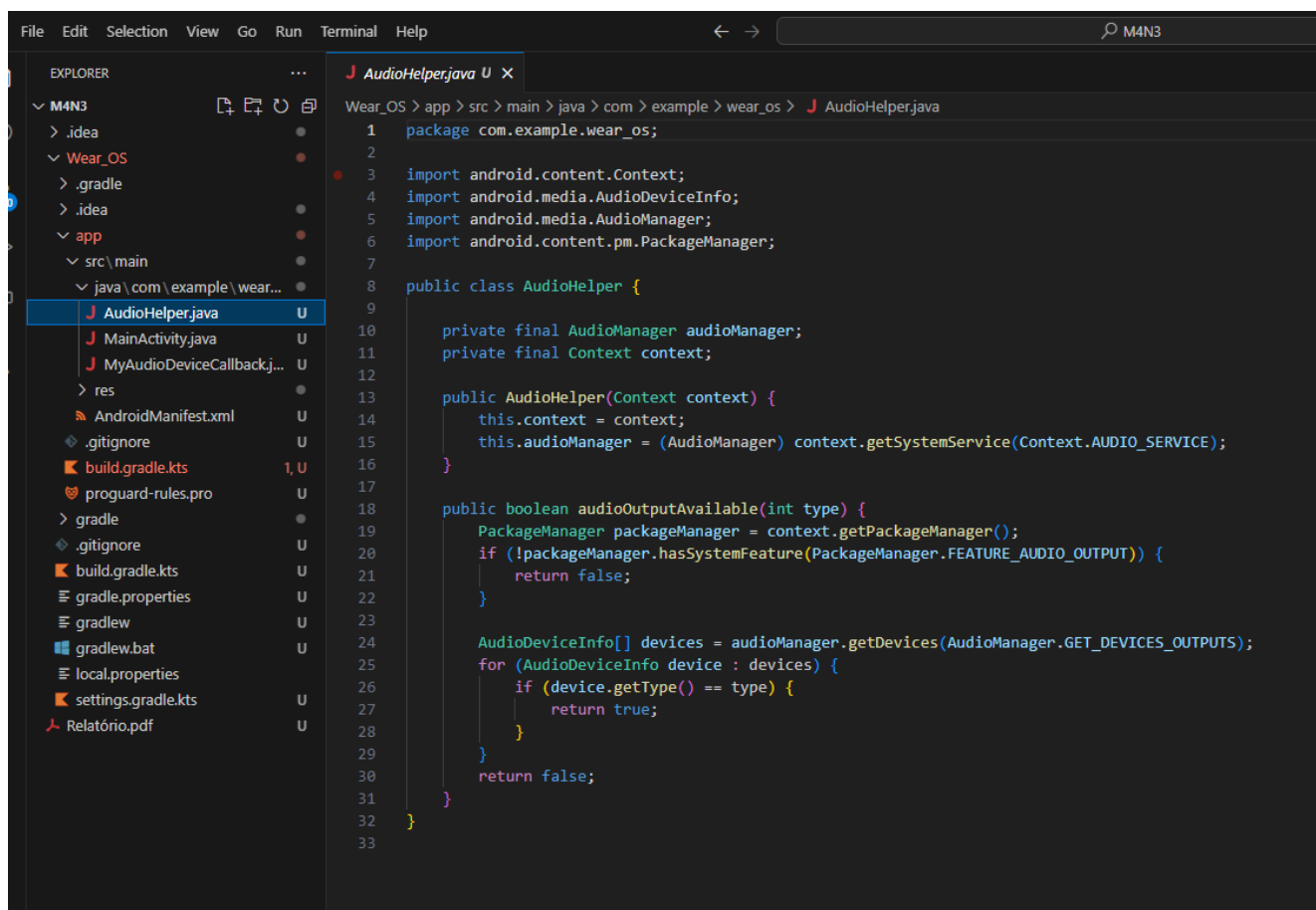


```
File Edit Selection View Go Run Terminal Help
M4N3

EXPLORER
M4N3
  .idea
  Wear_OS
  .gradle
  .idea
  app
    src/main
      java\com\example\wear...
        AudioHelper.java U
        MainActivity.java U
        MyAudioDeviceCallbackj... U
      res
        AndroidManifest.xml U
        .gitignore U
        build.gradle.kts 1, U
        proguard-rules.pro U
      gradle
        .gitignore U
        build.gradle.kts U
        gradle.properties U
        gradlew U
        gradlew.bat U
        local.properties U
        settings.gradle.kts U
      Relatório.pdf U
    OUTLINE
    TIMELINE
    JAVA PROJECTS

J MainActivity.java U X
Wear_OS > app > src > main > java > com > example > wear_os > J MainActivity.java > {} com.example.wear_os

1 package com.example.wear_os;
2
3 import android.content.Context;
4 import android.content.Intent;
5 import android.media.AudioDeviceCallback;
6 import android.media.AudioDeviceInfo;
7 import android.media.AudioManager;
8 import android.os.Bundle;
9 import android.speech.RecognitionListener;
10 import android.speech.RecognizerIntent;
11 import android.speech.SpeechRecognizer;
12 import android.speech.tts.TextToSpeech;
13 import android.util.Log;
14 import androidx.appcompat.app.AppCompatActivity;
15 import java.util.List;
16 import java.util.Locale;
17
18
19
20
21 public class MainActivity extends AppCompatActivity {
22
23     private AudioHelper audioHelper;
24     private AudioManager audioManager;
25     private TextToSpeech textToSpeech;
26     private SpeechRecognizer speechRecognizer;
27
28     @Override
29     protected void onCreate(Bundle savedInstanceState) {
30         super.onCreate(savedInstanceState);
31         setContentView(R.layout.activity_main);
32
33         // Inicializa o AudioHelper com o contexto atual
34         audioHelper = new AudioHelper(this);
35
36         // Inicializa o AudioManager
37         audioManager = (AudioManager) getSystemService(Context.AUDIO_SERVICE);
38
39         // Inicializa o TextToSpeech
40         textToSpeech = new TextToSpeech(this, status -> {
41             if (status != TextToSpeech.ERROR) {
42                 textToSpeech.setLanguage(Locale.getDefault());
43             }
44         });
45
46         // Inicializa o SpeechRecognizer
47         speechRecognizer = SpeechRecognizer.createSpeechRecognizer(this);
48         speechRecognizer.setRecognitionListener(new RecognitionListener() {
```



```
File Edit Selection View Go Run Terminal Help
M4N3

EXPLORER
M4N3
  .idea
  Wear_OS
  .gradle
  .idea
  app
    src/main
      java\com\example\wear...
        AudioHelper.java U
        MainActivity.java U
        MyAudioDeviceCallbackj... U
      res
        AndroidManifest.xml U
        .gitignore U
        build.gradle.kts 1, U
        proguard-rules.pro U
      gradle
        .gitignore U
        build.gradle.kts U
        gradle.properties U
        gradlew U
        gradlew.bat U
        local.properties U
        settings.gradle.kts U
      Relatório.pdf U
    OUTLINE
    TIMELINE
    JAVA PROJECTS

J AudioHelper.java U X
Wear_OS > app > src > main > java > com > example > wear_os > J AudioHelper.java

1 package com.example.wear_os;
2
3 import android.content.Context;
4 import android.media.AudioDeviceInfo;
5 import android.media.AudioManager;
6 import android.content.pm.PackageManager;
7
8 public class AudioHelper {
9
10     private final AudioManager audioManager;
11     private final Context context;
12
13     public AudioHelper(Context context) {
14         this.context = context;
15         this.audioManager = (AudioManager) context.getSystemService(Context.AUDIO_SERVICE);
16     }
17
18     public boolean audioOutputAvailable(int type) {
19         PackageManager packageManager = context.getPackageManager();
20         if (!packageManager.hasSystemFeature(PackageManager.FEATURE_AUDIO_OUTPUT)) {
21             return false;
22         }
23
24         AudioDeviceInfo[] devices = audioManager.getDevices(AudioManager.GET_DEVICES_OUTPUTS);
25         for (AudioDeviceInfo device : devices) {
26             if (device.getType() == type) {
27                 return true;
28             }
29         }
30         return false;
31     }
32 }
33
```

