Mídias
Executando Sons e Vídeos

Prof. Ilo Rivero

(ilo@pucminas.br)

O que vamos aprender nessa aula?

- Nessa aula vamos aprender executar sons e vídeos em nossos aplicativos com URLs ou arquivos (mp3 e mp4, por exemplo).

Dependência - AudioPlayers

audioplayers 0.19.0

Published May 13, 2021 • fireslime.xyz Null safety

FLUTTER ANDROID IOS MACOS WEB

841

Readme Changelog Example Installing Versions Scores

AudioPlayers

pub v0.19.0 build passing chat 246 online

A Flutter plugin to play multiple simultaneously audio files, works for Android, iOS

Contributing

We now have new rules for contributing!

All help is appreciated but if you have questions, bug reports, issues, feature requ
please first refer to our Contributing Guide.

Also, as always, please give us a star to help!

This will add a line like this to your package's pubspec.yaml (and run an implicit `dart pub get`):

```
dependencies:
  audioplayers: ^0.19.0
```

Alternatively, your editor might support `flutter pub get`. Check the docs for your editor to learn more.

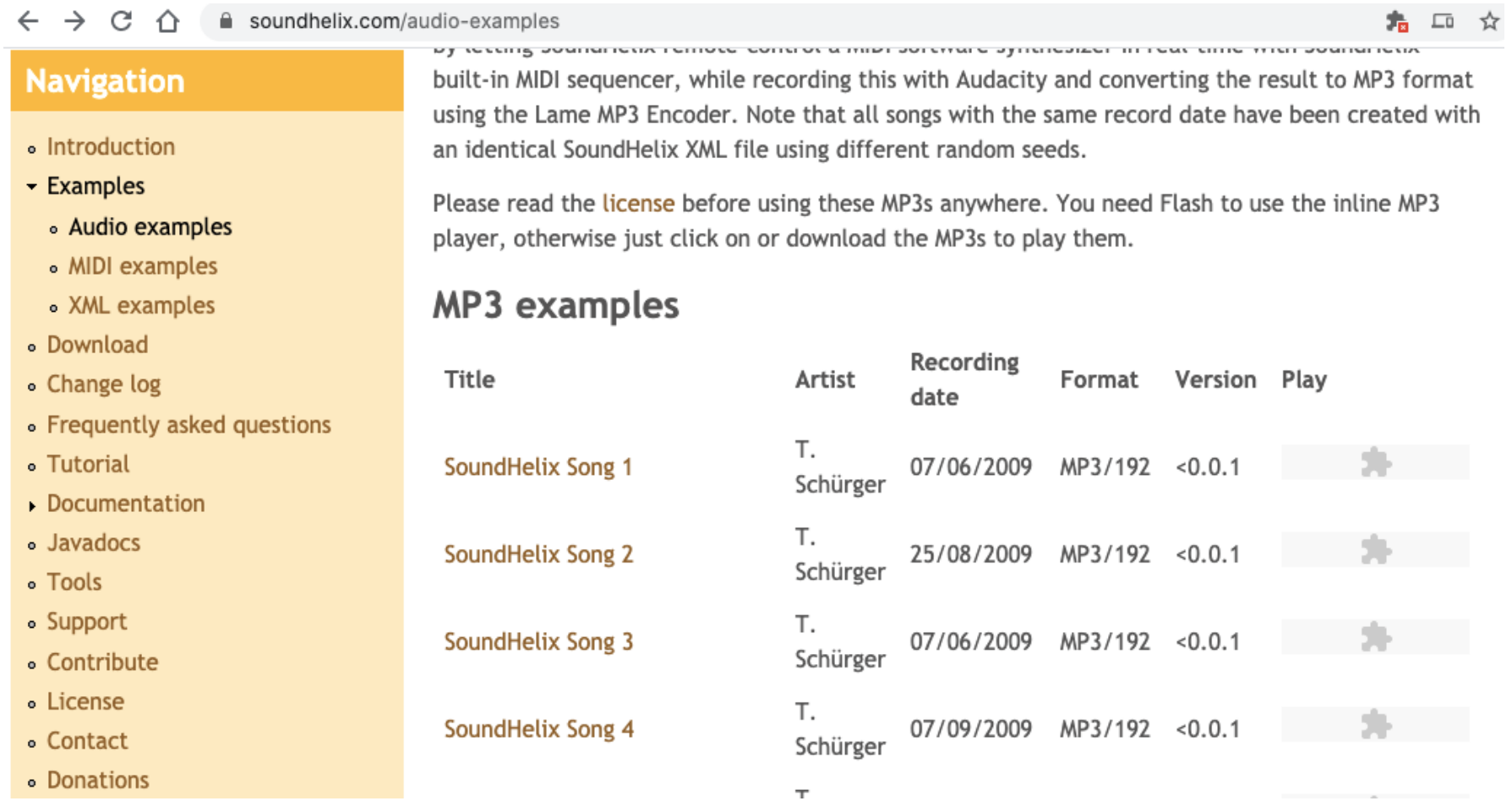
Import it

Now in your Dart code, you can use:

```
import 'package:audioplayers/audioplayers.dart';
```

Fonte: https://pub.dev/packages/audioplayers

Exemplos de MP3 Gratuitos



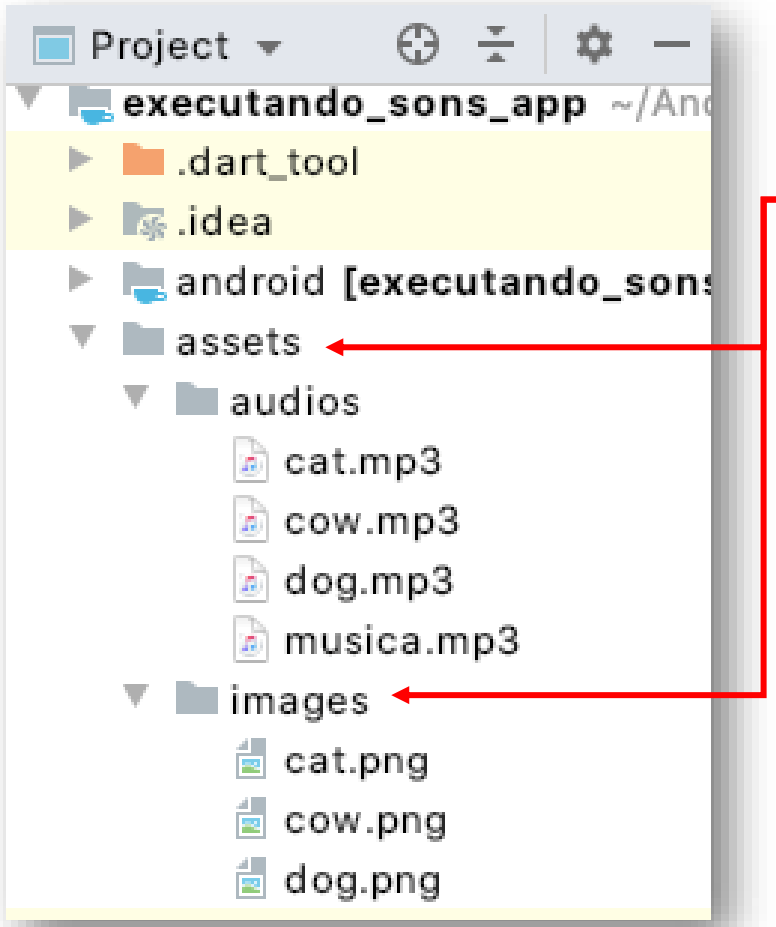
Fonte:

Executando um áudio com URL

```dart
class _HomeState extends State<Home> {
  AudioPlayer audioPlayer = AudioPlayer();
  String url = "https://www.soundhelix.com/examples/mp3/SoundHelix-Song-2.mp3";
  _executar() async{
    int resultado = await audioPlayer.play(url);
    if (resultado == 1) {
      // successo
    }
  }
```

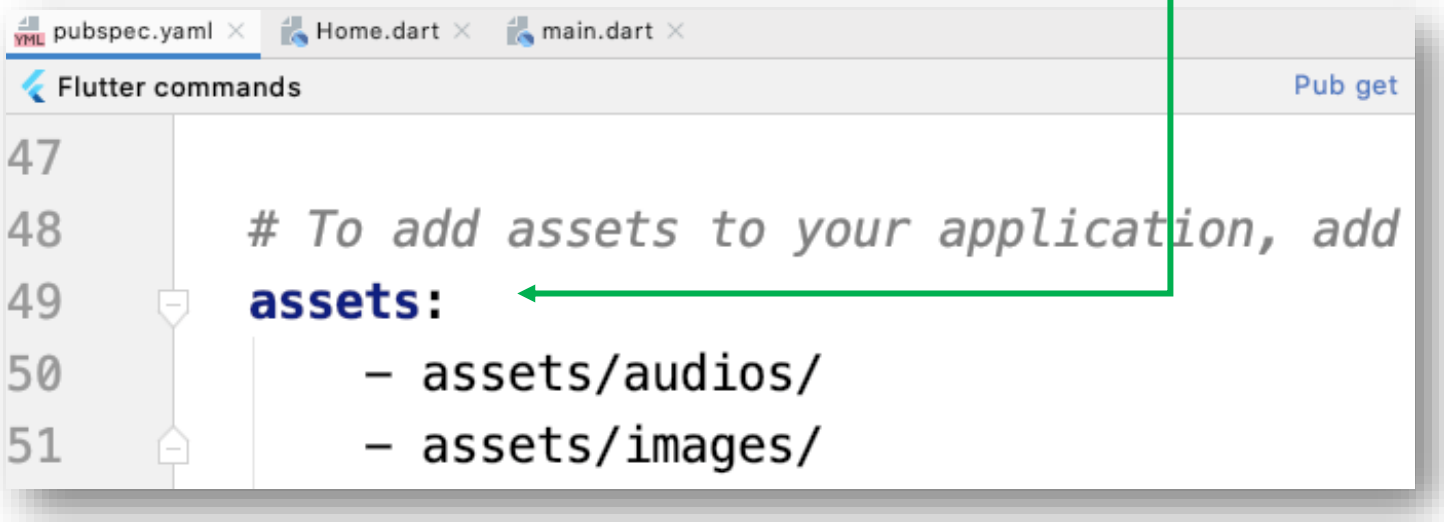
Não esquecer de incluir o import => **import 'package:audioplayers/audioplayers.dart';**

Executando um áudio de um arquivo MP3



Criar o diretório assets, e dentro dele, vou colocar outros diretrórios: audios e images

Tirar o comentário de assets no arquivo pbspec.yaml, e incluir uma linha para o diretório de imagens e o outro de audios, como a seguir



Executando um áudio de um arquivo MP3

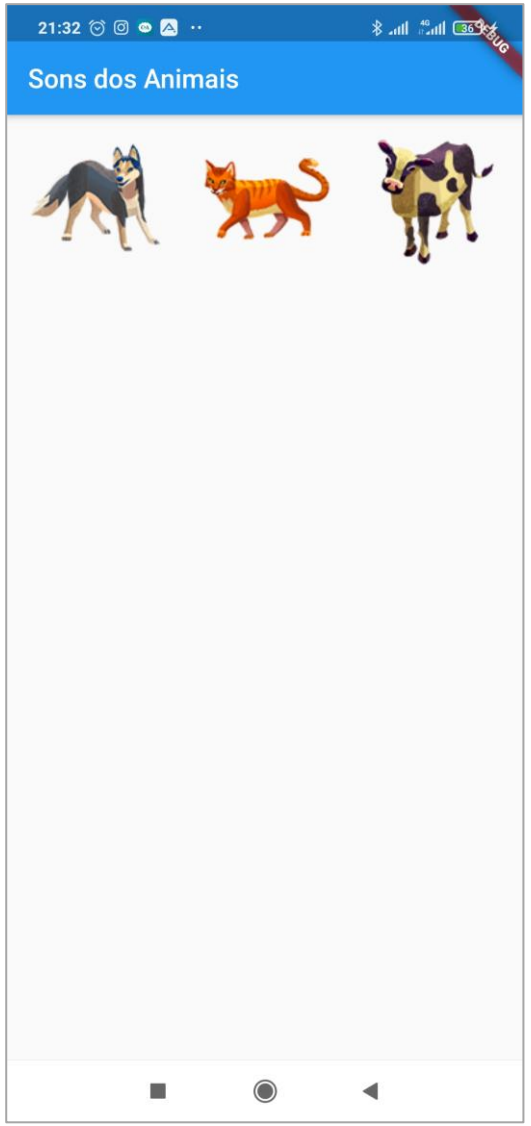
```dart
class _HomeState extends State<Home> {
  AudioPlayer audioPlayer = AudioPlayer();
  AudioCache audioCache = AudioCache(prefix: "assets/audios/");

  _executar() async{
    audioPlayer = await audioCache.play("dog.mp3");
  }
```

Para usar o AudioCache, é necessário importar o seguinte pacote: **import 'package:audioplayers/audioplayers.dart';**

o audioCache retorna um objeto audio player

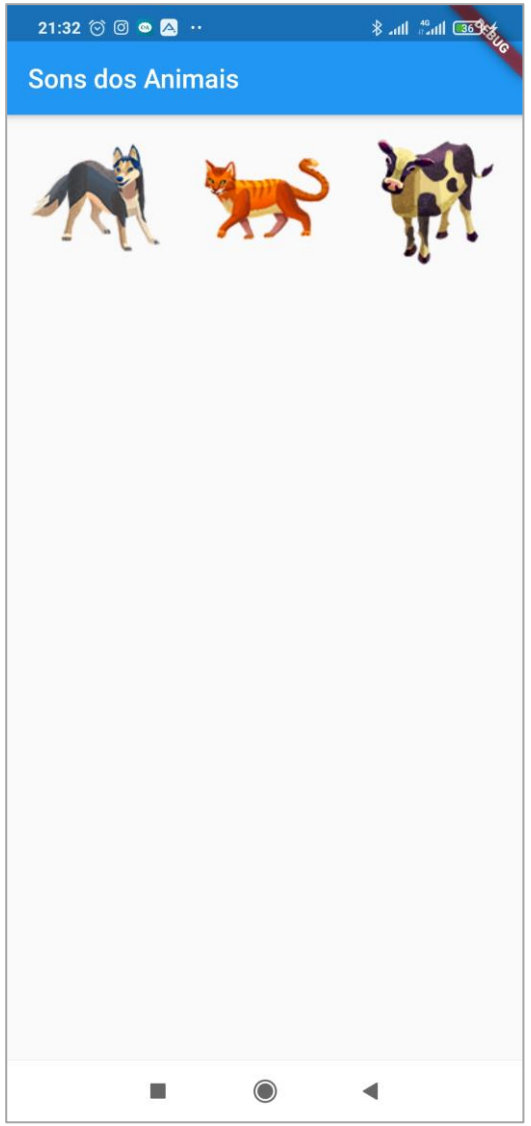
Exemplo - Sons dos Animais

```
_executarDog() async {
  audioPlayer = await audioCache.play("dog.mp3");
}
_executarCat() async {
  audioPlayer = await audioCache.play("cat.mp3");
}
_executarCow() async {
  audioPlayer = await audioCache.play("cow.mp3");
}
```



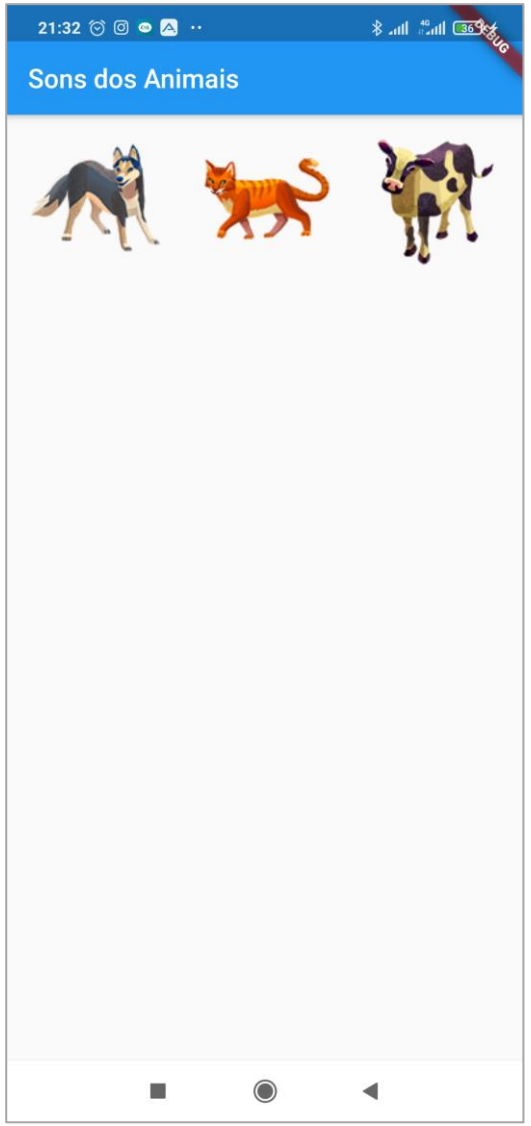
Exemplo - Sons dos Animais

```
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: Text("Sons dos Animais"),
    ), // AppBar
    body: Column(
      children: <Widget>[
        Row(
          mainAxisAlignment: MainAxisAlignment.center,
          children: <Widget>[
            Padding(
              padding: EdgeInsets.all(8),
              child: GestureDetector(
                child: Image.asset("assets/images/dog.png"),
                onTap: (){
                    _executarDog();
                },
              ), // GestureDetector
            ), // Padding
```



Exemplo - Sons dos Animais

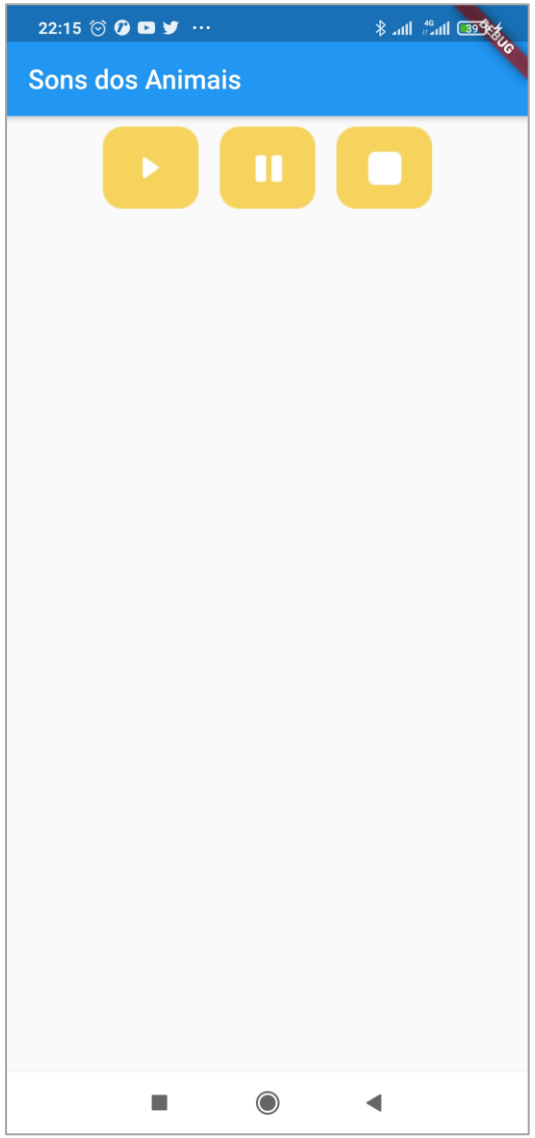
```
Padding(
  padding: EdgeInsets.all(8),
  child: GestureDetector(
    child: Image.asset("assets/images/cat.png"),
    onTap: (){
      _executarCat();
    },
  ), // GestureDetector
), // Padding
Padding(
  padding: EdgeInsets.all(8),
  child: GestureDetector(
    child: Image.asset("assets/images/cow.png"),
    onTap: (){
      _executarCow();
    },
  ), // GestureDetector
), // Padding
```



Exemplo - Executando Sons Player

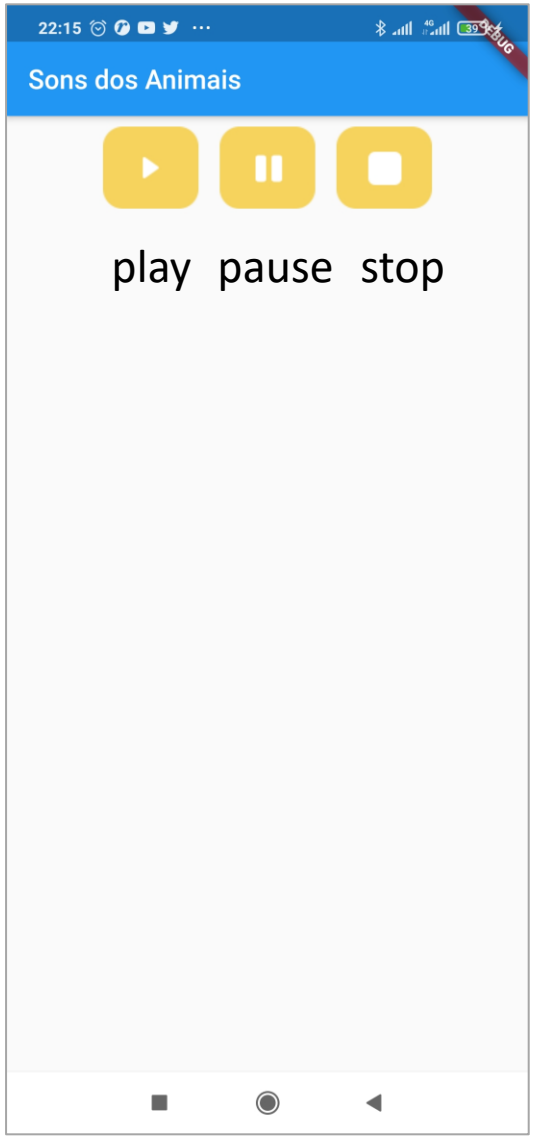
```
AudioPlayer audioPlayer = AudioPlayer();
AudioCache audioCache = AudioCache(prefix: "assets/audios/");
bool primeiraExecucao = true;

_executar() async {
  if(primeiraExecucao == true){
    audioPlayer = await audioCache.play("musica.mp3");
    primeiraExecucao = false;
  } else{
    audioPlayer.resume();
  }
}
```

O método resume() continua e executa a partir do ponto que foi pausado



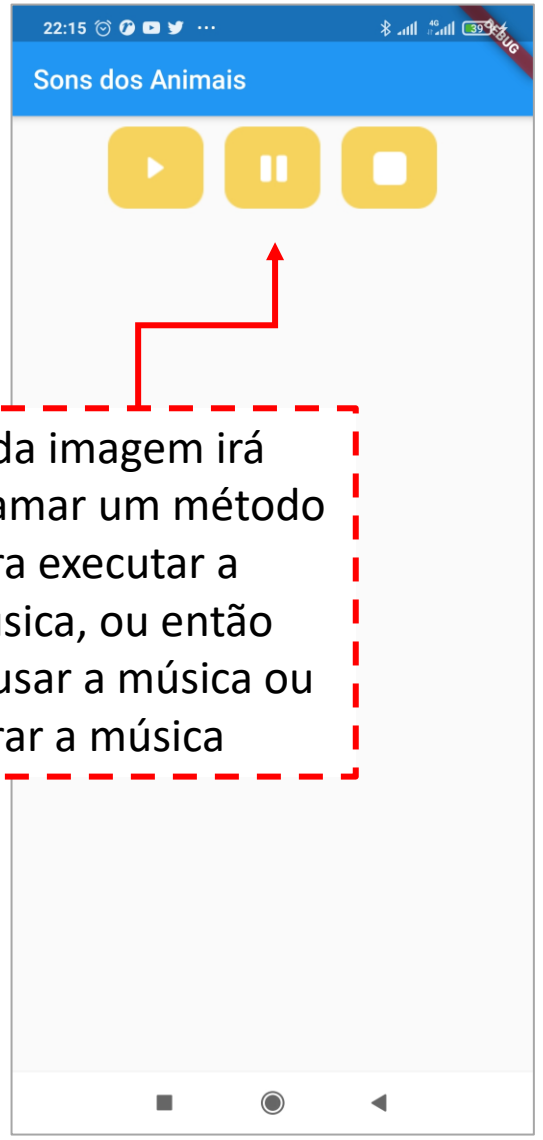
Exemplo - Executando Sons Player

```
_pausar() async{
  int resultado = await audioPlayer.pause();
  if(resultado == 1){
    //sucesso
  }
}

_parar() async{
  int resultado = await audioPlayer.stop();
  if(resultado == 1){
    //sucesso
  }
}
```



Exemplo - Executando Sons Player

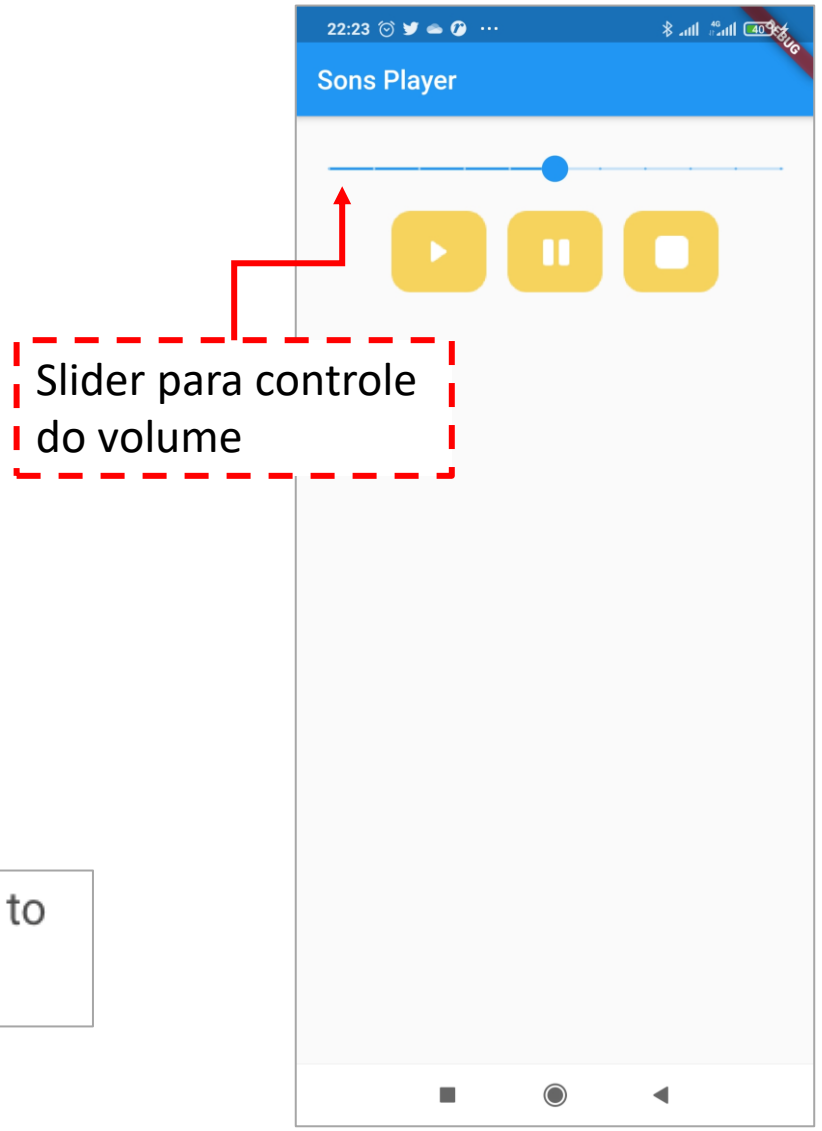
```
Padding(
  padding: EdgeInsets.all(8),
  child: GestureDetector(
    child: Image.asset("assets/images/executar.png"),
    onTap: (){
      _executar();
    },
  ), // GestureDetector
), // Padding
```



Cada imagem irá chamar um método para executar a música, ou então pausar a música ou parar a música

Controlando o volume

- Vamos usar o slider para realizar o controle de volume, mas na própria definição do áudio player, o valor zero é mudo e o valor 1 é o volume máximo



There is also an optional named `double volume` parameter, that defaults to `1.0`. It can go from `0.0` (mute) to `1.0` (max), varying linearly.

Fonte: https://pub.dev/packages/audioplayers

Controlando o volume

```
Slider(
      value: volume,
      min: 0,
      max: 1,
      divisions: 10,
      onChanged: (novoVolume){
        setState(() {
          volume = novoVolume;
        });
        audioPlayer.setVolume(novoVolume);
      }
  ), // Slider
```

```
double volume = 0.5;

_executar() async {
  audioPlayer.setVolume(volume);
  if(primeiraExecucao == true){
    audioPlayer = await audioCache.play("musica.mp3");
    primeiraExecucao = false;
  } else{
    audioPlayer.resume();
  }
}
```

Vamos fazer a primeira definição de volume ao dar play na música

Dependência - VideoPlayer

video_player 2.1.1

Published Apr 6, 2021 • flutter.dev Null safety

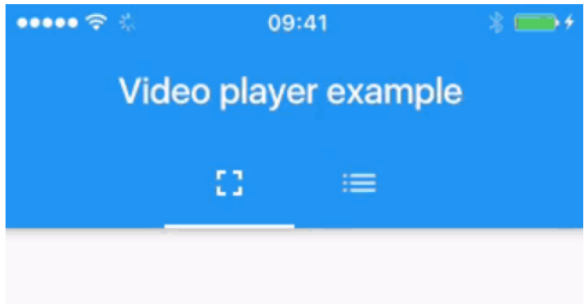
FLUTTER ANDROID IOS WEB

Readme Changelog Example Installing Versions Scores

Video Player plugin for Flutter

pub v2.1.1

A Flutter plugin for iOS, Android and Web for playing back video on a Widget su



This will add a line like this to your package's pubspec.yaml (and run an implicit `dart pub get`):

```
dependencies:
  video_player: ^2.1.1
```

Alternatively, your editor might support `flutter pub get`. Check the docs for your editor to learn more.

Import it

Now in your Dart code, you can use:

```
import 'package:video_player/video_player.dart';
```

Fonte: https://pub.dev/packages/video_player/install

Executando vídeos com URLs

iOS

Warning: The video player is not functional on iOS simulators. An iOS device must be used during development/testing.

Add the following entry to your *Info.plist* file, located in `<project root>/ios/Runner/Info.plist`:

```
<key>NSAppTransportSecurity</key>
<dict>
  <key>NSAllowsArbitraryLoads</key>
  <true/>
</dict>
```

Configurações que devem ser feitas, apenas para execuções de vídeos a partir de URLs

Android

Ensure the following permission is present in your Android Manifest file, located in `<project root>/android/app/src/main/AndroidManifest.xml`:

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

Executando vídeos com URLs

```
class _HomeState extends State<Home> {
  VideoPlayerController _videocontroller;

  @override
  void initState() {
    super.initState();
    _videocontroller = VideoPlayerController.network(
      "https://sample-videos.com/video123/mp4/720/big_buck_bunny_720p_1mb.mp4")
  ..initialize().then((_){});
  setState(() {
    _videocontroller.play();
  });
  }
```

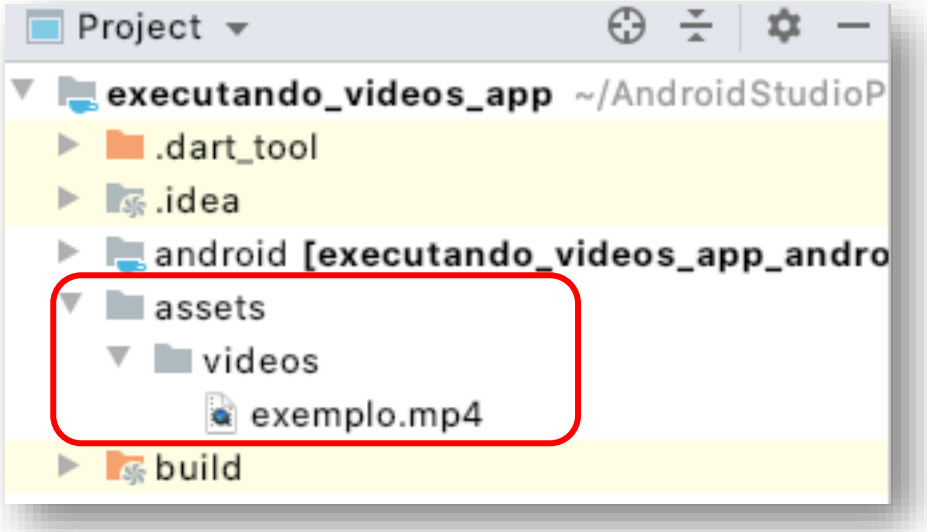
não será recuperado o retorno do initialize usando dois pontos

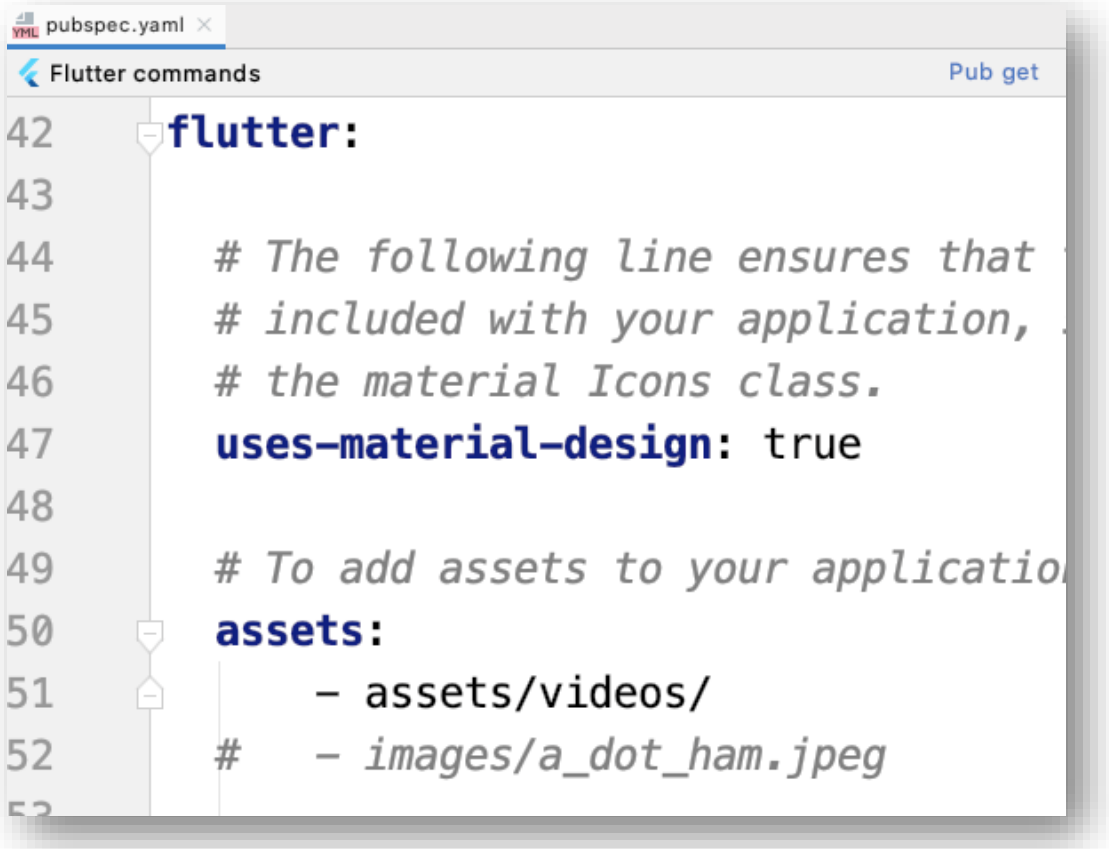
then é usado para fazer mais algo apos a inicialização do vídeo, e usamos o underline, pois não iremos passar nenhum parâmetro

Executando vídeos com URL

```
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: Center(
        child: AspectRatio(
            aspectRatio: _videocontroller.value.aspectRatio,
          child: VideoPlayer(_videocontroller),
        ), // AspectRatio
      ), // Center
    ); // Scaffold
  }
}
```

Executando vídeos com arquivo MP4



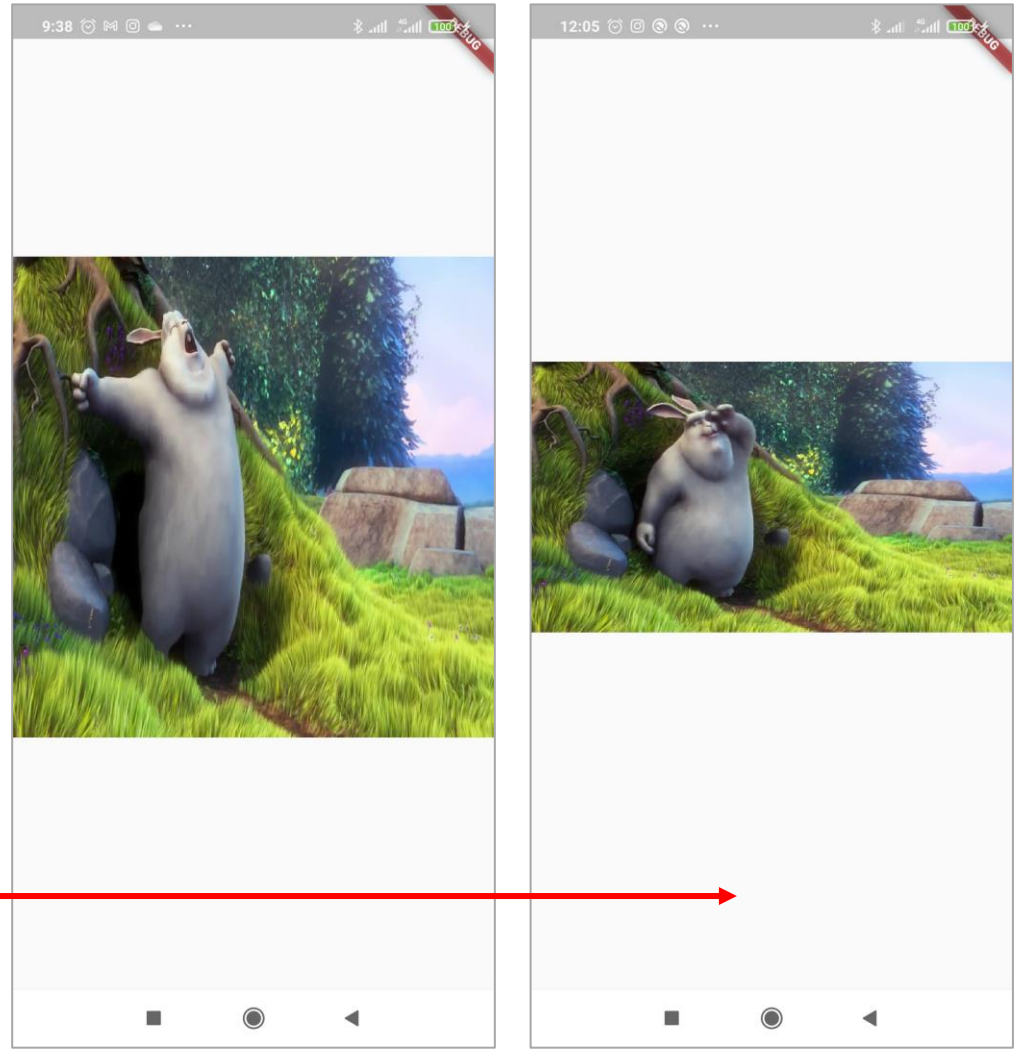


Exemplo - Executando vídeos com arquivo MP4

```
class _HomeState extends State<Home> {
  VideoPlayerController _videocontroller;

  @override
  void initState() {
    super.initState();
    _videocontroller = VideoPlayerController.asset(
        "assets/videos/exemplo.mp4"
    )
      ..setLooping(true)
      ..initialize().then((_) {
        _videocontroller.play();
      });
  }
```

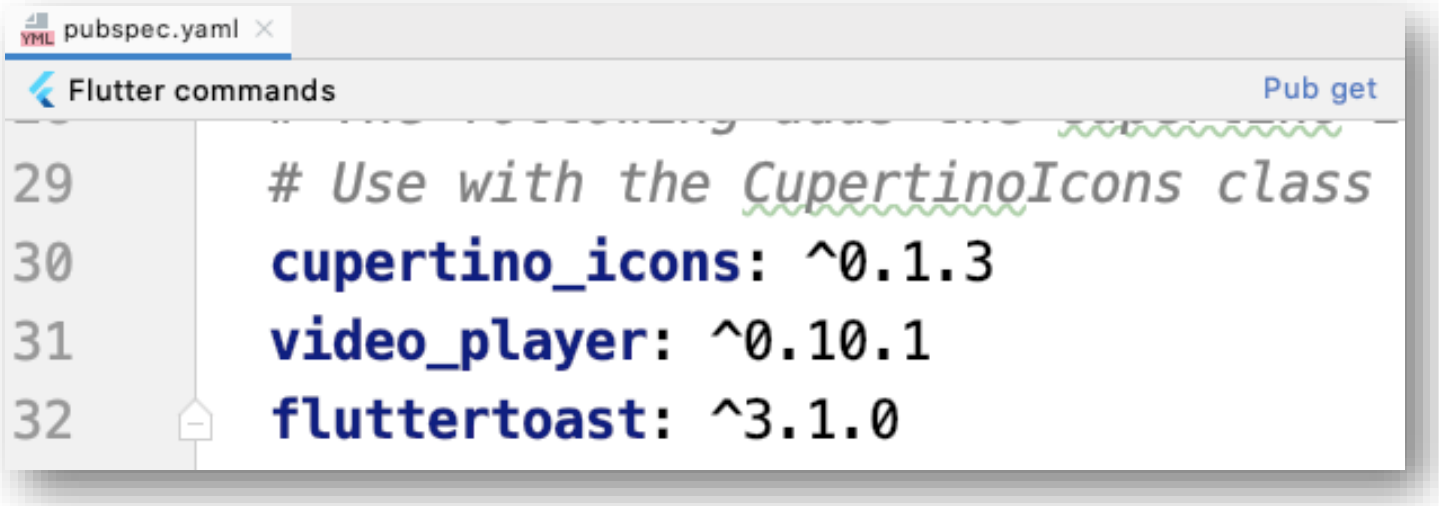
Exemplo - Executando vídeos com arquivo MP4

```
@override
Widget build(BuildContext context) {
  return Scaffold(
    body: Center(
      child: AspectRatio(
          aspectRatio: _videocontroller.value.aspectRatio,
        child: VideoPlayer(_videocontroller),
      ), // AspectRatio
    ), // Center
  ); // Scaffold
}
}
```



Erro - Compilação Javac

Erro:
“Execution failed for task 'fluttertoast:compileDebugJavaWithJavac'”



Referências Bibliográficas

- Curso da Udemy – **Flutter Essencial** do professor Ricardo Lecheta.

- Curso da Udemy - **Desenvolvimento Android e IOS com Flutter 2020 – Crie 15 Apps** do professor Jamilton Damasceno.

- https://www.soundhelix.com/audio-examples
- https://sample-videos.com/index.php#sample-mp4-video