

uniapp 安卓音频原生插件

目 录

使用说明	1
使用方法	1.1
更新日志	1.2
音频	2
存储权限	2.1
播放音频	2.2
设置播放位置	2.3
设置倍速	2.4
监听播放进度	2.5
获取当前播放位置	2.6
暂停播放	2.7
继续播放	2.8
停止播放	2.9
音频时长	2.10
获取当前播放属性	2.11
获取当前播放状态	2.12
录音	3
麦克风权限	3.1
开始录音	3.2
结束录音	3.3

使用方法

介绍

安卓音频原生插件，支持本地和网络音频播放进度监听，获取音频时长等，支持录音

联系作者

关注微信公众号可联系作者



插件地址

<https://ext.dcloud.net.cn/plugin?id=14803>

权限

1. 音频操作需要以下权限

- android.permission.INTERNET
- android.permission.ACCESS_NETWORK_STATE
- android.permission.READ_EXTERNAL_STORAGE
- android.permission.WRITE_EXTERNAL_STORAGE

2. 录音操作需要以下权限

- android.permission.RECORD_AUDIO
- android.permission.READ_EXTERNAL_STORAGE
- android.permission.WRITE_EXTERNAL_STORAGE

用法

1. 音频操作

在需要使用插件的页面加载以下代码

```
const module = uni.requireNativePlugin("leven-audio-AudioModule");
```

2. 录音

在需要使用插件的页面加载以下代码

```
const recordModule = uni.requireNativePlugin("leven-audio-RecordModule");
```

页面内容

```
<template>
  <view>
    <uni-card title="安卓音频操作原生插件—音频播放">
      <button type="primary" @click="requestPermission">存储权限</button>
      <button type="primary" @click="play">播放音频</button>
      <button type="primary" @click="setSeek">设置播放位置</button>
      <button type="primary" @click="onPlayListener">监听播放进度</button>
      <button type="primary" @click="getCurrentPosition">获取当前播放位置</button>
      <button type="primary" @click="pause">暂停播放</button>
      <button type="primary" @click="replay">继续播放</button>
      <button type="primary" @click="stop">停止播放</button>
      <button type="primary" @click="getDuration">音频时长</button>
    </uni-card>
    <uni-card title="安卓音频操作原生插件—录音">
      <button type="primary" @click="recorderPermission">麦克风权限</button>
      <button type="primary" @click="startRecorder">开始录音</button>
      <button type="primary" @click="stopRecorder">结束录音</button>
    </uni-card>
  </view>
</template>

<script>
  const module = uni.requireNativePlugin("leven-audio-AudioModule");
  const recordModule = uni.requireNativePlugin("leven-audio-RecordModule");
  export default {
    data() {
      return {
        // 当前播放音乐的索引
        playIndex: 0
      }
    },
  },
```

```

methods: {
  // 存储权限
  requestPermission() {
    module.requestPermission(res => {
      console.log(res)
    });
  },
  // 播放音频
  play() {
    let pathArr = [
      "/storage/emulated/0/Download/WeiXin/赵雷-成都.mp3",
      "/storage/emulated/0/Download/WeiXin/刀郎-西海情歌.mp3",
      "http://www.yeyuboke.com/svga/刘德华-孤星泪.mp3"
    ];
    let index = this.playIndex % pathArr.length;
    module.play({
      url: pathArr[index]
    }, res => {
      this.playIndex++;
      console.log(res)
      //这里可以监听播放进度
      // this.onPlayListener();
    });
  },
  // 设置播放位置
  setSeek() {
    module.setSeek({
      seek: 90
    }, res => {
      console.log(res)
    });
  },
  // 监听播放进度
  onPlayListener() {
    module.onPlayListener(res => {
      console.log(res)
    });
  },
  // 获取当前播放位置
  getCurrentPosition() {
    module.getCurrentPosition(res => {
      console.log(res)
    });
  },
  // 暂停播放
  pause() {
    module.pause(res => {
      console.log(res)
    });
  },
}

```

```

// 继续播放
replay() {
  module.replay(res => {
    console.log(res)
  });
},
// 停止播放
stop() {
  module.stop(res => {
    console.log(res)
  });
},
// 获取音频时长
getDuration() {
  module.getDuration({
    path: "/storage/emulated/0/Download/WeiXin/赵雷-成都.mp3"
  }, res => {
    console.log(res)
  });
},
// 麦克风权限
recorderPermission() {
  recordModule.requestPermission(res => {
    console.log(res)
  })
},
// 开始录音
startRecorder() {
  recordModule.start({
    saveDir: "/storage/emulated/0/leven_file"
  }, res => {
    console.log(res)
  })
},
// 结束录音
stopRecorder() {
  recordModule.stop(res => {
    console.log(res)
  })
}
}
}
</script>

<style>

</style>

```

更新日志

2023-10-16 v1.1.3

[优化]播放音频新增是否循环播放参数

2023-10-12 v1.1.2

[优化]优化音频播放

2023-10-11 v1.1.1

[新增]获取当前播放状态接口【[转到](#)】

2023-10-06 v1.1.0

[优化]播放音频新增倍速参数【[转到](#)】

[新增]设置倍速【[转到](#)】

[新增]获取当前播放属性【[转到](#)】

2023-09-26 v1.0.0

首次发布

存储权限

方法名

requestPermission

用法

- 用法如下：

```
module.requestPermission(res => {
  console.log(res)
});
```

- 参数说明

无

回调

- 示例

```
{
  "data": {},
  "message": "授权成功",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

播放音频

方法名

play

用法

- 用法如下：

```
let pathArr = [
  "/storage/emulated/0/Download/WeiXin/赵雷-成都.mp3",
  "/storage/emulated/0/Download/WeiXin/刀郎-西海情歌.mp3",
  "http://www.yeyuboke.com/svg/刘德华-孤星泪.mp3"
];
let index = this.playIndex % pathArr.length;
module.play({
  url: pathArr[index], //播放地址（支持本地音频和网络音频）
  speed: 1.0, //倍速
}, res => {
  this.playIndex++;
  console.log(res)
  //这里可以监听播放进度
  // this.onPlayListener();
});
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
url	String	是	无	要播放的音频绝对路径或网络地址
speed v1.1.0	Float	否	1.0	倍速
loop v1.1.3	Float	否	false	是否循环播放

回调

- 示例

```
{
  "data": {
    "status": "start",
    "duration": 328411
  },
  "message": "开始播放",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.status	String	播放状态，status：开始播放，finish：播放结束，error：播放出错
data.duration	Integer	音频总时长，单位：毫秒
code	Integer	返回类型，0.成功，其他：失败

设置播放位置

- 用法如下：

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
seek	Integer	是	无	要设置的播放位置，单位：秒

- 示例

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.position	Integer	当前播放位置，单位：毫秒
code	Integer	返回类型，0.成功，其他：失败

设置倍速

方法名

setSpeed

v1.1.0

用法

- 用法如下：

```
let speedArr = [0.5, 1.0, 1.5, 2.0];
let index = this.playSpeedIndex % speedArr.length;
module.setSpeed({
  speed: speedArr[index]
}, res => {
  this.playSpeedIndex++;
  console.log(res)
});
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
speed	Float	是	无	倍速

回调

- 示例

```
{
  "data": {},
  "message": "设置完成",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象

设置倍速

参数名	参数类型	参数描述
data.position	Integer	当前播放位置，单位：毫秒
code	Integer	返回类型，0.成功，其他：失败

监听播放进度

方法名

onPlayListener

用法

- 用法如下：

```
module.onPlayListener(res => {  
  console.log(res)  
});
```

- 参数说明

无

回调

- 示例

```
{  
  "data": {  
    "percent": 0.013805481982651363,  
    "position": 4738,  
    "duration": 343197  
  },  
  "message": "",  
  "code": 0  
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.percent	Float	当前播放的进度，<1
data.position	Integer	当前播放位置，单位：毫秒

参数名	参数类型	参数描述
data.duration	Integer	音频总时长，单位：毫秒
code	Integer	返回类型，0.成功，其他：失败

获取当前播放位置

方法名

getCurrentPosition

用法

- 用法如下：

```
module.getCurrentPosition(res => {  
  console.log(res)  
});
```

- 参数说明

无

回调

- 示例

```
{  
  "data": {  
    "position": 5905  
  },  
  "message": "",  
  "code": 0  
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.position	Integer	当前播放位置，单位：毫秒
code	Integer	返回类型，0.成功，其他：失败

暂停播放

方法名

pause

用法

- 用法如下：

```
module.pause(res => {
  console.log(res)
});
```

- 参数说明

无

回调

- 示例

```
{
  "data": {},
  "message": "操作完成",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

继续播放

方法名

replay

用法

- 用法如下：

```
module.replay(res => {  
  console.log(res)  
});
```

- 参数说明

无

回调

- 示例

```
{  
  "data": {},  
  "message": "操作完成",  
  "code": 0  
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

停止播放

方法名

stop

用法

- 用法如下：

```
module.stop(res => {
  console.log(res)
});
```

- 参数说明

无

回调

- 示例

```
{
  "data": {},
  "message": "操作完成",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

音频时长

方法名

getDuration

用法

- 用法如下：

```
module.getDuration({
  path: "/storage/emulated/0/Download/WeiXin/赵雷-成都.mp3"
}, res => {
  console.log(res)
});
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
path	Integer	是	无	本地绝对路径或网络地址

回调

- 示例

```
{
  "data": {
    "duration": 445518
  },
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象

音频时长

参数名	参数类型	参数描述
data.duration	Integer	音频总时长，单位：毫秒
code	Integer	返回类型，0.成功，其他：失败

获取当前播放属性

方法名

getAudioProperty

v1.1.0

用法

- 用法如下：

```
module.getAudioProperty(res => {
  console.log(res)
});
```

- 参数说明

无

回调

- 示例

```
{
  "data": {
    "speed": 1,
    "position": 33522,
    "duration": 328411
  },
  "message": "操作完成",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.speed	Float	当前播放倍速
data.position	Integer	当前播放位置，单位：毫秒

获取当前播放属性

参数名	参数类型	参数描述
data.duration	Integer	音频时长，单位：毫秒
code	Integer	返回类型，0.成功，其他：失败

获取当前播放状态

方法名

getPlayStatus

v1.1.1

用法

- 用法如下：

```
module.getPlayStatus(res => {
  console.log(res)
});
```

- 参数说明

无

回调

- 示例

```
{
  "data": {
    "playStatus": 1
  },
  "message": "操作完成",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.playStatus	Integer	当前播放状态，0：未播放，1：播放中，2：暂停播放
code	Integer	返回类型，0.成功，其他：失败

麦克风权限

方法名

requestPermission

用法

- 用法如下：

```
recordModule.requestPermission(res => {
  console.log(res)
});
```

- 参数说明

无

回调

- 示例

```
{
  "data": {},
  "message": "授权成功",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

开始录音

方法名

start

用法

- 用法如下：

```
recordModule.start({
  saveDir: "/storage/emulated/0/leven_file"
}, res => {
  console.log(res)
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
saveDir	String	是	无	要保存的目录

回调

- 示例

```
{
  "data": {},
  "message": "开始录制",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

结束录音

方法名

stop

用法

- 用法如下：

```
recordModule.stop(res => {  
  console.log(res)  
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
saveDir	String	是	无	要保存的目录

回调

- 示例

```
{  
  "data": {  
    "path": "/storage/emulated/0/leven_file/20230926140510.mp3",  
    "duration": 25124  
  },  
  "message": "录制结束",  
  "code": 0  
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.path	String	录音文件路径

结束录音

参数名	参数类型	参数描述
data.duration	String	录音时长，单位：毫秒
code	Integer	返回类型，0.成功，其他：失败