SFML

* [Main Page](http://docs.google.com/index.htm)
* [Modules](http://docs.google.com/modules.htm)
* [Classes](http://docs.google.com/annotated.htm)
* [Files](http://docs.google.com/files.htm)
* [File List](http://docs.google.com/files.htm)
* [include](http://docs.google.com/dir_f3190241575fd2bd132a392ae6942f4a.htm)
* [SFML](http://docs.google.com/dir_692f376662c82a26cfe4cfa3aceebe24.htm)
* [Audio](http://docs.google.com/dir_3c6cd700807083a0c327f22d879e95e3.htm)

SoundRecorder.hpp

1

2 //

3 // SFML - Simple and Fast Multimedia Library

4 // Copyright (C) 2007-2013 Laurent Gomila (laurent.gom@gmail.com)

5 //

6 // This software is provided 'as-is', without any express or implied warranty.

7 // In no event will the authors be held liable for any damages arising from the use of this software.

8 //

9 // Permission is granted to anyone to use this software for any purpose,

10 // including commercial applications, and to alter it and redistribute it freely,

11 // subject to the following restrictions:

12 //

13 // 1. The origin of this software must not be misrepresented;

14 // you must not claim that you wrote the original software.

15 // If you use this software in a product, an acknowledgment

16 // in the product documentation would be appreciated but is not required.

17 //

18 // 2. Altered source versions must be plainly marked as such,

19 // and must not be misrepresented as being the original software.

20 //

21 // 3. This notice may not be removed or altered from any source distribution.

22 //

24

25 #ifndef SFML\_SOUNDRECORDER\_HPP

26 #define SFML\_SOUNDRECORDER\_HPP

27

29 // Headers

31 #include <SFML/Audio/Export.hpp>

32 #include <SFML/System/Thread.hpp>

33 #include <vector>

34

35

36 namespace sf

37 {

[42](http://docs.google.com/classsf_1_1SoundRecorder.htm) class SFML\_AUDIO\_API [SoundRecorder](http://docs.google.com/classsf_1_1SoundRecorder.htm)

43 {

44 public :

45

50  virtual ~[SoundRecorder](http://docs.google.com/classsf_1_1SoundRecorder.htm)();

51

67  void start(unsigned int sampleRate = 44100);

68

75  void stop();

76

87  unsigned int getSampleRate() const;

88

100  static bool isAvailable();

101

102 protected :

103

110  [SoundRecorder](http://docs.google.com/classsf_1_1SoundRecorder.htm)();

111

123  virtual bool onStart();

124

139  virtual bool onProcessSamples(const Int16\* samples, std::size\_t sampleCount) = 0;

140

150  virtual void onStop();

151

152 private :

153

161  void record();

162

171  void processCapturedSamples();

172

179  void cleanup();

180

182  // Member data

184  [Thread](http://docs.google.com/classsf_1_1Thread.htm) m\_thread;

185  std::vector<Int16> m\_samples;

186  unsigned int m\_sampleRate;

187  bool m\_isCapturing;

188 };

189

190 } // namespace sf

191

192

193 #endif // SFML\_SOUNDRECORDER\_HPP

194

195

Copyright � Laurent Gomila  ::  Documentation generated by [doxygen](http://www.doxygen.org/)  ::