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)
* [Network](http://docs.google.com/dir_b9ac88db2949395b3130dd4ffb1be4e1.htm)

TcpSocket.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\_TCPSOCKET\_HPP

26 #define SFML\_TCPSOCKET\_HPP

27

29 // Headers

31 #include <SFML/Network/Export.hpp>

32 #include <SFML/Network/Socket.hpp>

33 #include <SFML/System/Time.hpp>

34

35

36 namespace sf

37 {

38 class TcpListener;

39 class IpAddress;

40 class Packet;

41

[46](http://docs.google.com/classsf_1_1TcpSocket.htm) class SFML\_NETWORK\_API [TcpSocket](http://docs.google.com/classsf_1_1TcpSocket.htm) : public [Socket](http://docs.google.com/classsf_1_1Socket.htm)

47 {

48 public :

49

54  [TcpSocket](http://docs.google.com/classsf_1_1TcpSocket.htm)();

55

66  unsigned short getLocalPort() const;

67

79  [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm) getRemoteAddress() const;

80

92  unsigned short getRemotePort() const;

93

111  [Status](http://docs.google.com/classsf_1_1Socket.htm#a51bf0fd51057b98a10fbb866246176dc) connect(const [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm)& remoteAddress, unsigned short remotePort, [Time](http://docs.google.com/classsf_1_1Time.htm) timeout = [Time::Zero](http://docs.google.com/classsf_1_1Time.htm#a8db127b632fa8da21550e7282af11fa0));

112

122  void disconnect();

123

137  [Status](http://docs.google.com/classsf_1_1Socket.htm#a51bf0fd51057b98a10fbb866246176dc) send(const void\* data, std::size\_t size);

138

155  [Status](http://docs.google.com/classsf_1_1Socket.htm#a51bf0fd51057b98a10fbb866246176dc) receive(void\* data, std::size\_t size, std::size\_t& received);

156

169  [Status](http://docs.google.com/classsf_1_1Socket.htm#a51bf0fd51057b98a10fbb866246176dc) send([Packet](http://docs.google.com/classsf_1_1Packet.htm)& packet);

170

185  [Status](http://docs.google.com/classsf_1_1Socket.htm#a51bf0fd51057b98a10fbb866246176dc) receive([Packet](http://docs.google.com/classsf_1_1Packet.htm)& packet);

186

187 private:

188

189  friend class [TcpListener](http://docs.google.com/classsf_1_1TcpListener.htm);

190

195  struct PendingPacket

196  {

197  PendingPacket();

198

199  Uint32 Size;

200  std::size\_t SizeReceived;

201  std::vector<char> Data;

202  };

203

205  // Member data

207  PendingPacket m\_pendingPacket;

208 };

209

210 } // namespace sf

211

212

213 #endif // SFML\_TCPSOCKET\_HPP

214

215

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