Streaming audio between c# app and a website

★

I want to stream the system audio from my computer to another device in the same network.
For listening to my computer, I created a c# Windows console application (I hope the comments are fulfilling).

C# app:

```
using System.Threading;
using CSCore.SoundIn;

using WebSocketSharp;
using WebSocketSharp.Server;

namespace PBAConsole
{

    class Program
    {
        public static string currentAudioBytes = null;
        static void Main(string[] args)
        {

            //createweb socket
            try
            {
                Websocket();
            }catch(Exception e)
            {
                Console.WriteLine(e.Message);
            }

        }


        //open a web socket on the port 1111
        static void Websocket()
        {
            String connectionString = "ws://192.168.178.57:1111";
            WebSocketServer wss = new WebSocketServer(connectionString);

            wss.AddWebSocketService<Audio>("/Audio");
            wss.Start();
            Console.WriteLine("server started on :"+connectionString+"/Audio");

            GetAudioByteArray();
            Console.ReadKey();
            wss.Stop();
        }



        //gets the audio with by using the CSCore.SoundIn libary https://github.com
        public static void GetAudioByteArray()
        {
            WasapiCapture capture = new WasapiLoopbackCapture();
```

```
            capture.Start();
            //setup an eventhandler to receive the recorded data
            capture.DataAvailable += (s, e) =>
            {
                currentAudioBytes =Convert.ToBase64String(e.Data, e.Offset, e.ByteC
            };

        }
    }

    public class Audio : WebSocketBehavior
    {
        protected override void OnMessage(MessageEventArgs e)//Gets Message
        {
            base.OnMessage(e);
            Console.WriteLine("recieved message from client : " + e.Data);

            while (true)//constantly check if there is new audio data available**st
            {
                //Send the audio data, if there is some available
                if (Program.currentAudioBytes != null)
                {
                    try
                    {
                        Send(Program.currentAudioBytes);
                        Program.currentAudioBytes = null;
                    }
                    catch (Exception ex)
                    {
                        Console.WriteLine(ex.Message);
                    }

                }


            }

        }


    }
}
```

js code from the Website

```
function init() {
  audioCtx = new AudioContext();
}

//gets called by Clicking the button on the Website
function buildConnection() {


  var ws = new WebSocket("ws://192.168.178.57:1111/Audio");

  //Hey Server, I am listening
    waitForSocketConnection(ws, function(){
      console.log("message sent!!!");
      ws.send("Hey Server, I am listening");
    }, 0);

  //Gets bdase64 String from server
  ws.onmessage = function (event) {
    console.log("got message");
    var audioBytes = _base64ToArrayBuffer(event.data);
    playWave(audioBytes);
  }

}

// Make the function wait until the connection is made...
function waitForSocketConnection(socket, callback, tries){
  setTimeout(
      function () {
          if (socket.readyState === 1) {
              console.log("Connection is made")
              if (callback != null){
                  callback();
              }
          } else {
            tries++;
            if(tries>=2000){
              console.log("Cound't connect");
              return;
            }
              console.log("wait for connection...")
              waitForSocketConnection(socket, callback, tries);
          }

      }, 5); // wait 5 milisecond for the connection...
}
```

```
  var len = binary_string.length;
  var bytes = new Uint8Array(len);
  for (var i = 0; i < len; i++) {
      bytes[i] = binary_string.charCodeAt(i);
      console.log(bytes[i])
  }
  return bytes;
}

//play the audio
function playWave(byteArray) {
  var audioCtx = new (window.AudioContext || window.webkitAudioContext)();
  var samplerate = 44100;
  var myAudioBuffer = audioCtx.createBuffer(2, samplerate*0.015, samplerate);//roun
  var nowBuffering = myAudioBuffer.getChannelData(0);
  for (var i = 0; i < byteArray.length; i++) {
      nowBuffering[i] = byteArray[i];
  }

  var source = audioCtx.createBufferSource();
  source.buffer = myAudioBuffer;
  source.connect(audioCtx.destination);
  source.start();
}
```

HTML from the website

```
        <meta charset="utf-8">
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <title></title>
        <meta name="description" content="">
        <meta name="viewport" content="width=device-width, initial-scale=1">
        <!--<link rel="stylesheet" href="">-->
    </head>
    <body>
        <button style="padding:1%;" type="button" onclick="buildConnection()">send

        <script src="main.js"></script>
    </body>
</html>
```

Source: JavaSript – Stack Overflow

Share

September 28, 2021

Category : News

Tags: audio-streaming | c# | cscore | javascript

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment *

Name *

Email *

Website

☐ Save my name, email, and website in this browser for the next time I comment.