Enhanced Desktop Recorder in .NET using C# and Windows Forms

Sateesh Arveti Updated date Jul 23, 2008

155.6k 50 6

DesktopRecorder.zip

Nowadays, Desktop or Application sharing is common to coordinate work properly in IT as well as other fields also. Sometimes, it may require to record the Operations done by us on Desktop for future reference. I have not seen many applications which support Desktop recording, sharing and broadcasting of it. I think it's better to design an application that will do above operations little bit easier. So, I design this application in VS.NET 2005 using C# and windows forms. I will explain features provided by this application, one sample scenario where we can use this application followed by its design and coding.

Features present in this application:

- It allows us to record the Desktop Operations.
- It allows us to view the recordings with an inbuilt Media Player.
- It allows us to view the List of recent recordings.
- Inbuilt Functionality to add Audio to the Desktop recordings.
- Inbuilt Functionality to broadcast Desktop recordings.
- Inbuilt Functionality to see remote Desktop.
- Inbuilt Functionality to play, pause and stop recordings.
- Inbuilt Functionality to start, pause and stop recording.
- Inbuilt Functionality to show recording Duration, number of users connected to broadcast (desktop sharing).
- Easy to use UI.
- Now, Desktop recording and sharing is just a click away from us.

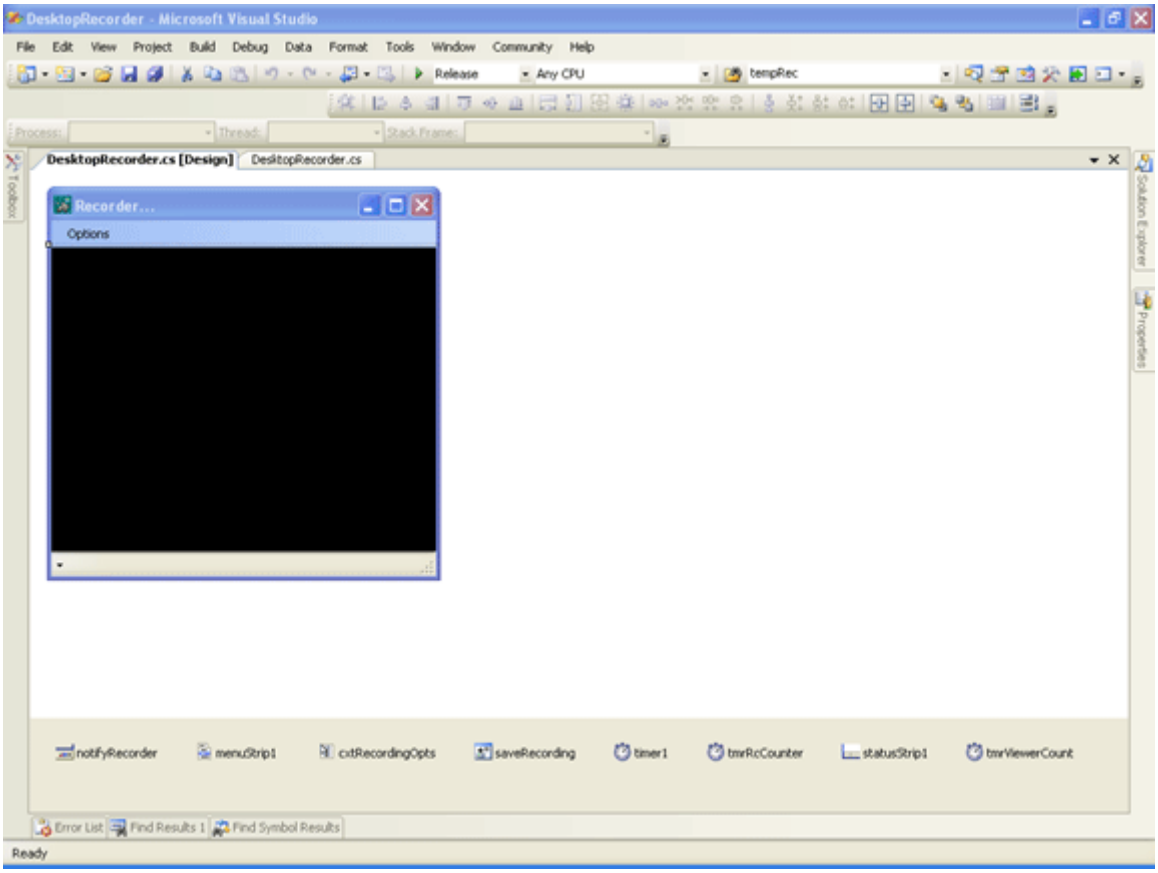
Sample Scenario for using this application:

1000's word description. So, why can't a tester record the steps for reproducing the bug in a video, instead of complex documents? Around of 50 - 60% of our effort in fixing the bug will be spent in analyzing, reproducing, get clarifications from testers regarding bug. So, in order to compensate it, the best solution is to record the bug's details in a video with audio support for extra information. In this kind of situations, this application will be very handy to use. There are other cases also, where this application will be useful like analyzing complex application's functionalities etc.

Prerequisites:

In order to use this application, we require Media Player Encoder 9. It is an plugin for Windows media player. It can be downloaded from here:

Now, create a new Windows Application using C# in VS.NET 2005 and name it as DesktopRecorder. Add controls to Main Form (DesktopRecorder) as shown below:



Now, I will outline functionality of few main controls in the application:

ContextMenu (cxtRecordingOpts):

This Control having few menu items for following operations:

- Start Recording --> To start recording the Desktop.
- Pause Recording --> To pause recording the Desktop.

- Save Current Recording --> To save current recording.
- Broadcast --> To enable broadcast, set Port Number for broadcasting.
- Recent Recordings --> To see list of recent recordings.

menuStrip1 (Menu Strip):

This control is having an Option Menu with following items in it:

- Open Broadcast URL Recording --> To Set URL of broadcast for viewing in Media player.
- Play, Pause and Stop menuitems are used in changing the state of the player.
- Show Media Player UI --> To show/hide inbuilt media player UI.
- Exit --> To Exit the application.

Apart from this, I added few other controls to improve UI & Functionality.

Now, add a reference to COM dll (WMEncoderLib).If not, you can access this dlls from bin folder of code attached here. This will allows us to access functionality provided by Windows Media Player Encoder.

The main functionality of this application is it will record all desktop operations in wmv format for future reference along with sharing capability at a time.

Now, I will explain coding part of this application.

On Click of Start Recording --> Here, we are creating an instance media player encoder and setting its properties like input audio, video and broadcast path etc. Finally, Encoder will be started to start recording. I will explain functionality of this event, little bit deeper. Since, this is the core for this application:

```
IWMEncProfile SelProfile;
IWMEncSource AudioSrc;
try
{
   if (DesktopEncoder != null)
   {
      //Checks Whether Encoder in is paused state or not.If it paused, it //will just starts it and
returns.
      if (DesktopEncoder.RunState ==  MENC_ENCODER_STATE.WMENC_ENCODER_PAUSED)
      {
         DesktopEncoder.Start();
         return;
      }
   }
   #region Default settings for Encoder.
   DesktopEncoderAppln = new WMEncoderApp();
   DesktopEncoder = DesktopEncoderAppln.Encoder;
```

```
    IWMEncVideoSource2 VideoSrc =
(IWMEncVideoSource2)SrcGroup.AddSource(WMENC_SOURCE_TYPE.WMENC_VIDEO);
    #endregion
    //Set Audio Source,if Add Audio Checkbox is checked.
    if (addAudio.Checked)
    {
        AudioSrc = SrcGroup.AddSource(WMENC_SOURCE_TYPE.WMENC_AUDIO);
        if (txtAudioFile.Text.Trim() != "")
        {
            if (File.Exists(txtAudioFile.Text.Trim()))
            {
                //Set audio file path to be used while encoding or recording.
                AudioSrc.SetInput(txtAudioFile.Text.Trim(), "", "");
            }
            else
            {
                //Set to use Default audio device as input,if file does not exist.
                AudioSrc.SetInput("Default_Audio_Device", "Device", "");
            }
        }
        else
        {
            AudioSrc.SetInput("Default_Audio_Device", "Device", "");
        }
    }
    //Set Video Source:Desktop.
    VideoSrc.SetInput("ScreenCapture1", "ScreenCap", "");
    IWMEncProfileCollection ProfileCollection = DesktopEncoder.ProfileCollection;
    ProfileCollection = DesktopEncoder.ProfileCollection;
    int lLength = ProfileCollection.Count;
    //Set Profile.
    if (toolstripEnableBroadcast.Checked && txtPortNbr.Text.Trim() != "")
    {
        IWMEncBroadcast broadcast = DesktopEncoder.Broadcast;
broadcast.set_PortNumber(WMENC_BROADCAST_PROTOCOL.WMENC_PROTOCOL_HTTP,
Convert.ToInt32(txtPortNbr.Text.Trim()));
        for (int i = 0; i <= lLength - 1; i++)
        {
            SelProfile = ProfileCollection.Item(i);
            //Set selected profile as Windows Media Video 8 for Local Area //Network,if broadcasting is
enabled.
```

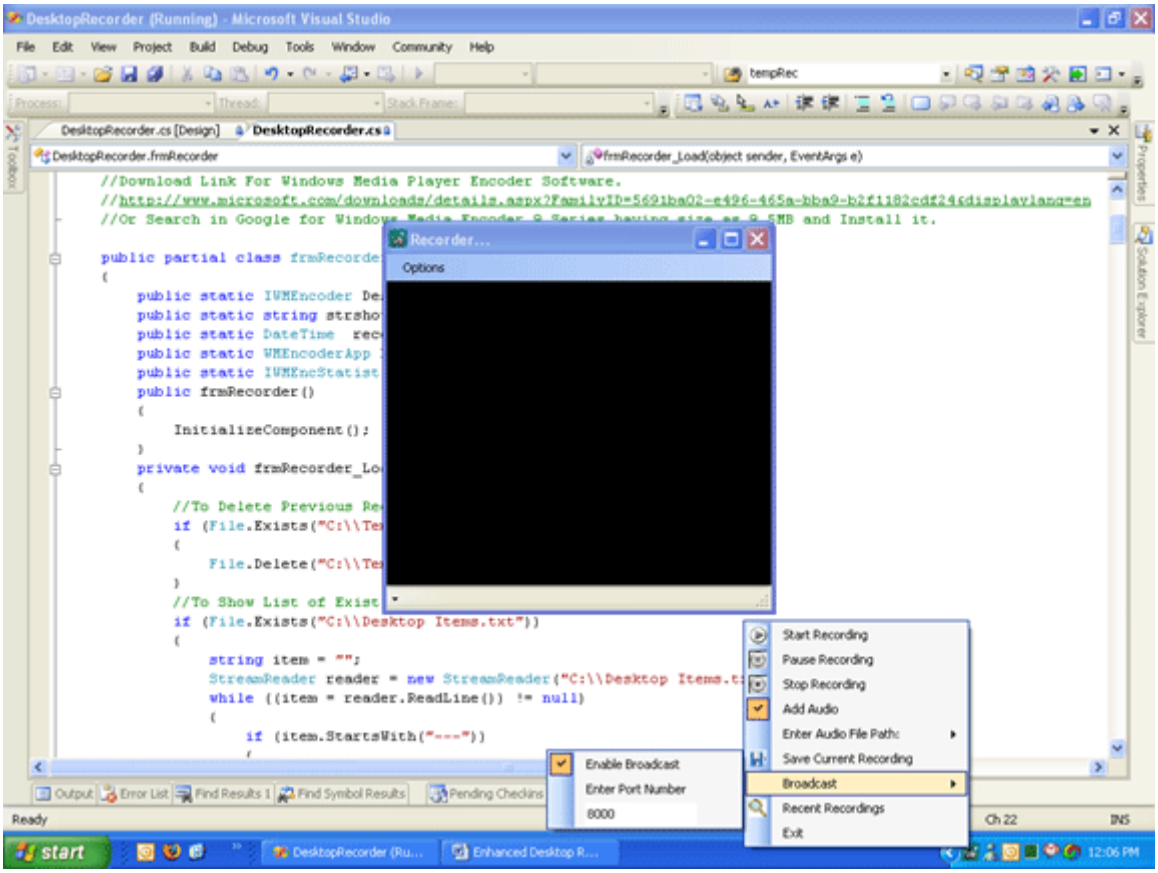
```
            SrcGroup.set_Profile((IWMEncProfile)SelProfile);
            break;
        }
    }
}
else
{
    for (int i = 0; i <= lLength - 1; i++)
    {
        SelProfile = ProfileCollection.Item(i);
        if (SelProfile.Name == "Screen Video/Audio High (CBR)")
        {
            SrcGroup.set_Profile((IWMEncProfile)SelProfile);
            break;
        }
    }
}
//Local File to Store Recording temporarily.
IWMEncFile inputFile = DesktopEncoder.File;
inputFile.LocalFileName = "C:\\TempRecording.wmv";
DesktopEncoder.PrepareToEncode(true);
DesktopEncoder.Start();
tmrRcCounter.Enabled = true;
recordStarttime = DateTime.Now;
if (toolstripEnableBroadcast.Checked && txtPortNbr.Text.Trim() != "")
{
    //Start Timer to Count Viewers connected to Broadcast.if broadcasting is enabled.
    tmrViewerCount.Enabled = true;
}
}
catch (Exception ex)
{
MessageBox.Show(ex.Message);
}
```

On Click of Stop Recording --> Here, Encoder will be stopped for stopping recording. Later, it will popup save dialog to store the recording to a file. The path of the file, where recording is saved will be added to Recent Recordings menu.

On Click of Exit (application closing) --> Here, all recent recording's path will be added to a file for loading it later.

1. Start the application, than click on Start Recording to start recording of desktop operations.
2. If you want to broadcast (to be viewed by others) this recording, than goto Broadcast menu and click Enable Broadcast, set port number to be used for broadcasting. It will use HTTP protocol for broadcasting. So, it won't be blocked by firewalls. Than, start the recording. So that, at a time you can record your desktop operations and broadcast it also.
3. If you want to add Audio to your recording, than goto Enter Audio File Path and set audio filename along with its path. Than, start the recording. So that, at a time you can record your desktop operations along with audio appended to it.
4. Then, Click on Stop Recording when you are done with it. It will popup a save dialog to select filename for saving the recording.
5. Then, you can play those recordings by going to Recent Recordings and selecting that filename from it.
6. If you want to watch broadcasting of the desktop, than goto Options Menu and select set URL of the Broadcast (like http://10.100.1.100:8000) and hit Enter. It will start playing the broadcast.
7. In Status bar, you can see recording duration followed by number of users connected to your broadcast.

Finally, I added some code to enhance UI of the application. And, the final output will be like this:



We can still enhance this application by improving UI, Broadcast Quality etc.

By using this application, we can record all your Desktop activities. I am attaching source code for further reference. I hope this code will be useful for all.

Enhanced Desktop Recorder in .NET using C# and Windows Forms | Desktop Recorder

Next Recommended Reading

Enhanced MSI Analyzer in .NET using C# and Windows Forms

Sateesh Arveti TOP 100

I hold Bachelors degree in Computer Science along with MCSD,MCTS and MVP for the year 2009. Areas of Interest: C#, WPF, WF, silverlight, ASP.NET, Oracle and SQL Server.

https://www.c-sharpcorner.com/members/sateesh-arveti

80 **5.5m** **4**

View Previous Comments

6 **50**

Type your comment here and press Enter Key (Minimum 10 characters)

Hello its not working windows 10....so please help me error '"{"Specified cast is not valid."}

bashiruddin Sheikh

Feb 09, 2019

NA **158** **0**

0 0 Reply

Hey i want to add text in video in dot net how can i do this

Pinki Yadav

Aug 28, 2018

NA **53** **0**

0 0 Reply

Can i integrate this application in web? if yes then please tell me how can i achieve that? I want to open this application using html javascript code and when recording is done th video file(as a blob) should be on html page. please help me thanks in advance.

Please update this article. The zip file is incomplete and there is a broken link to download the media player encoder 9

Vikash Kumar

1180 955 17.5k

Aug 22, 2017

0 0 Reply

i have error in using wmencoderlib;

Zubair Saif

NA 92 129

Apr 07, 2016

0 0 Reply

GOOD ONE

Sr Karthiga

188 12.1k 1.5m

Feb 18, 2016

0 0 Reply

GOOD ONE

Sr Karthiga

188 12.1k 1.5m

Feb 18, 2016

0 0 Reply

Recoding is not working. Only 0 byes is showing. What is the problem

Sundararaj N

NA 47 0

Sep 01, 2015

0 0 Reply

could you install Media Player Encoder 9 64bits verion on windows10 OS ?

Cheng chun yuan

NA 1 0

Aug 12, 2015

0 0 Reply

AudioSrc.SetInput("Default_Audio_Device", "Device", ""); this line recording audio from mic not from speakers. how to set speakers?

mursaleenfayyaz fayyazuddin

1973 34 3k

Jun 09, 2015

0 0 Reply

FEATURED ARTICLES

Easily Create SPA With .NET 6.0 And Angular 13 And Deploy To Azure

Easily Create Charts In Angular 13 with Dynamic Data

Implement Swagger UI In ASP.NET Web API Restful Service For Documentation Using Swashbuckle

5 Steps To Implement Event Call-Backs In Blazor

Getting Started with Azure File Storage

View All

TRENDING UP

01 Design The Full Load And Delta Load Patterns In SSIS

02 How Lazyloading In Blazor Can Increase Your Application Performance!