

Forum:[Hack Forums](#)

Thread: [Control Media Players using voice](#)

jmsgt9 - July 24, 2013, 1:41 pm

This project uses System.Speech.dll to recognize voice, then uses 'Declare Auto Function SendMessageW Lib "user32.dll" to simulate the pressing of media keys.

I made this program to experiment with voice recognition. Since I listen to large spotify playlists with my usb headset that usually contain songs I don't care for, I figured this would be a neat little tool. Here is the source and download links. Feel free to copy/paste no need to credit if you don't want to.

Commands and Action

Play - Simulates Play/Pause media key

Pause - Simulates Play/Pause media key

Next - Simulates NextTrack media key

Previous - Simulates PreviousTrack media key

backtrack - Simulates PreviousTrack media key twice, some music players will repeat current song if only pressed once.

Code 'Before declaring imports you must add

'C:\Program Files (x86)\Reference

Assemblies\Microsoft\Framework\NETFramework\v4.0\Profile\Client\System.Speech.dll
'as a reference.

Imports System.Speech

Imports System.Runtime.InteropServices

Public Class MainForm

'Declare Keyboard Hex Codes

Const APPCOMMAND_MEDIA_PLAY_PAUSE = &HE0000

Const APPCOMMAND_MEDIA_NEXTTRACK = &HB0000

Const APPCOMMAND_MEDIA_PREVIOUSTRACK = &HC0000

Const WM_APPCOMMAND = &H319

'Declare speech synthesizer, used to generate replies/acknowledgements kind of like TTS.

Dim synthesizer As New Synthesis.SpeechSynthesizer

Declare Auto Function SendMessageW Lib "user32.dll" Alias "SendMessageW" (

ByVal hWnd As Integer,

ByVal Msg As Integer,

ByVal wParam As Integer,

ByVal lParam As Integer) As Integer

'Declare speech recognizer.

Dim WithEvents recog As New Recognition.SpeechRecognitionEngine

Private Sub MainForm_Load(sender As Object, e As EventArgs) Handles MyBase.Load

'Tell the speech recognizer to use default audio device.

recog.SetInputToDefaultAudioDevice()

'Declaration of dependencies for the list of words to listen for.

Dim grammar As New Recognition.SrgsGrammar.SrgsDocument

Dim cmdRule As New Recognition.SrgsGrammar.SrgsRule("cmd")

```

'Set list of words to listen for.
Dim cmdlist As New Recognition.SrgsGrammar.SrgsOneOf("pause", "play", "next", "previous",
"backtrack")

'Link it all together...
cmdRule.Add(cmdlist)
grammar.Rules.Add(cmdRule)
grammar.Root = cmdRule

'Load cmdlist into speech recognizer.
recog.LoadGrammar(New Recognition.Grammar(grammar))

'Start listening for cmds.
recog.RecognizeAsync()
End Sub

'What to do after understanding a command...
Private Sub reco_RecognizeCompleted(ByVal sender As Object, ByVal e As
System.Speech.Recognition.RecognizeCompletedEventArgs) Handles recog.RecognizeCompleted
'Start Listening again...
recog.RecognizeAsync()
End Sub

'Acknowledge command with synthesized reply
Private Sub Talk(ByVal phrase As String)
synthesize.Speak(phrase)
End Sub

'How to handle a recognized phrase...
Private Sub reco_SpeechRecognized(ByVal sender As Object, ByVal e As
System.Speech.Recognition.RecognitionEventArgs) Handles recog.SpeechRecognized

'Specific events for each cmd.
Select Case e.Result.Text

    Case "play"
        Talk("Playing")
        SendMessageW(Handle, WM_APPCOMMAND, Handle,
APPCOMMAND_MEDIA_PLAY_PAUSE)

    Case "pause"
        Talk("Pausing")
        SendMessageW(Handle, WM_APPCOMMAND, Handle,
APPCOMMAND_MEDIA_PLAY_PAUSE)

    Case "next"
        Talk("Playing Next Track")
        SendMessageW(Handle, WM_APPCOMMAND, Handle,
APPCOMMAND_MEDIA_NEXTTRACK)

    Case "previous"
        Talk("Playing Previous Track")
        SendMessageW(Handle, WM_APPCOMMAND, Handle,

```

APPCOMMAND_MEDIA_PREVIOUSTRACK)

'Some Media Players will go back to beginning of current track on a previous track press, this overloads that by sending the key twice.

Case "backtrack"

Talk("Playing Previous Track")

SendMessageW(Handle, WM_APPCOMMAND, Handle, APPCOMMAND_MEDIA_PREVIOUSTRACK)

SendMessageW(Handle, WM_APPCOMMAND, Handle, APPCOMMAND_MEDIA_PREVIOUSTRACK)

End Select

End Sub

End Class

Here is a dl for just the exe:

[\[Virus Scan\]](#)

<https://www.dropbox.com/s/vn3rrm2ikdssl5/MusicVoiceControl.exe>

Here is a dl for the project:

<https://www.dropbox.com/s/1afox55gwrjdj1hu/MusicVoiceControl.zip>

*Note don't use this while talking or in a occupied area haha, it's not very accurate in deciphering words, "hey" can be recognized as "play" etc.

serubi - October 30, 2013, 9:24 pm

I'm working on a similar project. How did you get the appcommand values? For example: &HC0000