

Source: C# Corner ([www.c-sharpcorner.com](http://www.c-sharpcorner.com))

PRINT

## Article



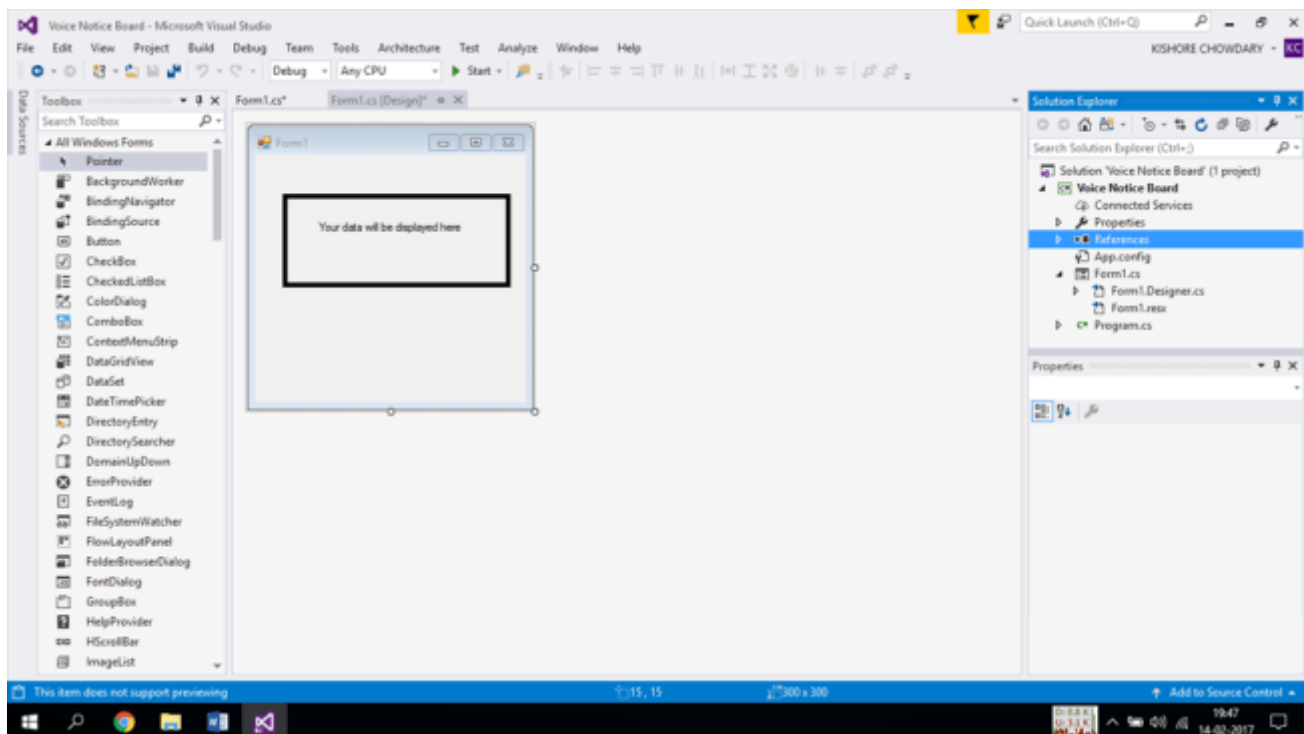
### Voice Notice Board Using Windows Forms

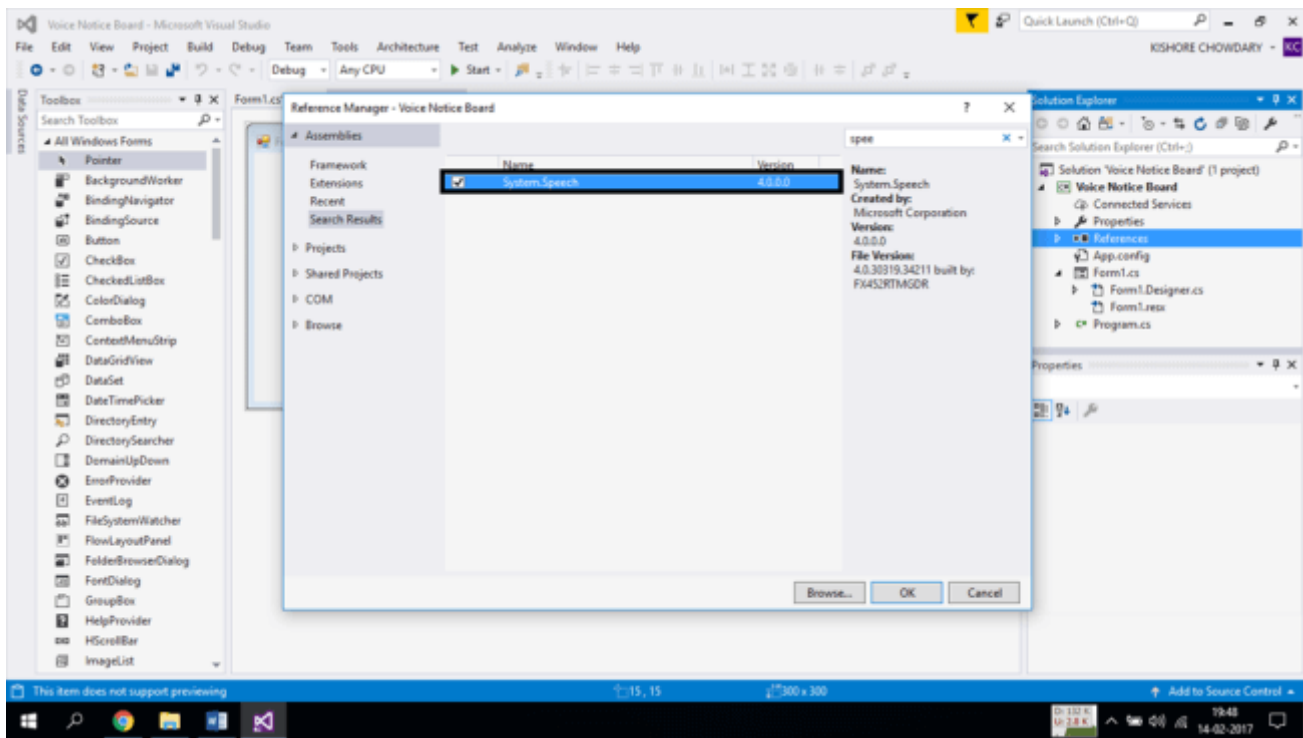
By **Kishore Chowdary** on **Feb 17 2017**

This article will help you in creating a Windows form application in which you can say something to display in the forms through the labels, which I call a voice notice board. For this purpose, you will be creating a Windows form application and you are going to write the speech synthesis code. Now, you will be giving a set of predefined phrases, which you would like to display in the form. You can connect the monitor for an output and then display the notices there just by saying them out loud.

#### Creating the form

As usual, go to Visual Studio and open up Visual Studio. Create a new Windows forms project. As soon as you create the project, add the *System.Speech* reference file. This is going to help you for speech synthesis. Add a label to the form. This is where your message will be displayed when you speak. Once you finish creating the design and adding the reference file, now it is time to code.





## Writing Code

To write the code for speech synthesis, double click on the form, which you have opened. This is where you are going to write the speech synthesis code. In the code, you can find something called Slist in which an array of strings will be declared. This is where you should declare all the messages, which you would like to display in the forms by speaking. The string which is given in this list will only be recognized. Thus, enter all the strings, which you will be speaking as messages. Add the code given below in the Form and change the phrases for your messages.

```

1. using System;
2. using System.Speech.Recognition;
3. using System.Speech.Synthesis;
4. using System.Windows.Forms;
5.
6. namespace Voice_Notice_Board
7. {
8.     public partial class Form1 : Form
9.     {
10.         public Form1()
11.         {
12.             InitializeComponent();
13.         }
14.
15.         public void label1_Click(object sender, EventArgs e)
16.         {
17.
18.         }
19.         SpeechSynthesizer speechSynthesizerObj;
20.         public void Form1_Load(object sender, EventArgs e)
21.         {
22.
23.
24.             speechSynthesizerObj = new SpeechSynthesizer();
25.             speechSynthesizerObj.SpeakAsync("Please speak");
26.             SpeechRecognitionEngine sRecognize = new SpeechRecognitionEngine();

```

```
27.     Choices sList = new Choices();
28.     //you can add more messages and sentences in here to speak out while the program runs.
29.     sList.Add(new string[] {"Hello", "Good Morning", "Hai", "Bye", "Good Night"});
30.     Grammar gr = new Grammar(new GrammarBuilder(sList));
31.     try
32.     {
33.         sRecognize.RequestRecognizerUpdate();
34.         sRecognize.LoadGrammar(gr);
35.         sRecognize.SpeechRecognized += sRecognize_SpeechRecognized;
36.         sRecognize.SetInputToDefaultAudioDevice();
37.         sRecognize.RecognizeAsync(RecognizeMode.Multiple);
38.         sRecognize.Recognize();
39.     }
40.     catch
41.     {
42.         return;
43.     }
44. }
45.
46. public void sRecognize_SpeechRecognized(object sender, SpeechRecognizedEventArgs e)
47. {
48.     label1.Text = e.Result.Text;
49. }
50. }
51. }
```

## Deploying

Thus, you have created an application which will work as a notice board with just the voice commands. Save your project and click on the start. This will deploy your project. Now, speak out the phrases, which you would like to display in the form. This will display the phrases, which you say if they are already written in the SList of the program. I hope you like this concept of a Voice Notice board. Thank you for reading.

---

Thank you for using C# Corner