Captcha

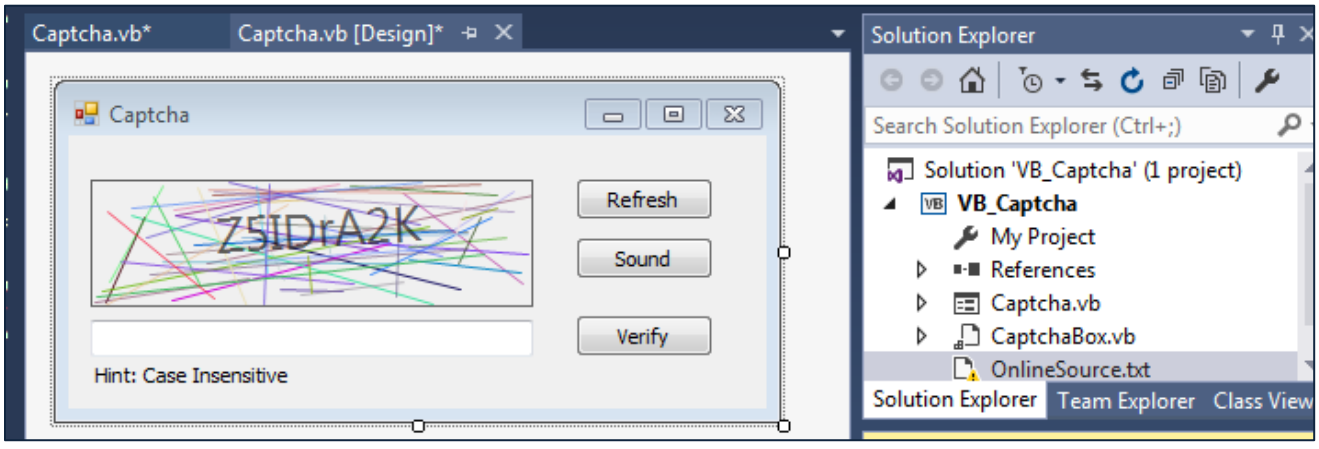
These guidelines show how implement Captcha feature in your applications using tools and services provided in VB.NET platform.
Time needed to accomplish this task: 20 - 30 minutes

GUI

- Create a new VB.NET project "Windows Form Application", change the project name and path according to your preferences.
- Create a new windows form (if it is not created by default), name it "frmCaptcha", and change its caption/text to "Captcha"
- In "frmCaptcha" form, create 1 textbox, 1 label, 1 CaptchaBox and 3 buttons and change their properties as follows:

Old Name	New Name	Caption/Text	Other Properties
TextBox1	txtInp		
Label1	lblHint	Hint: Case Insensitive	
CaptchaBox1	CaptchaBox1		
Button1	btnRefresh	Refresh	
Button2	btnSound	Sound	
Button3	btnVerify	Verify	

- By the end of this step you should have something looks like the following picture:



Now… Let's do some coding…

Coding

- At the beginning, let's do some initializations for the CaptchaBox1. Add these lines to the "frmCaptcha_Load" event sub:

```
CaptchaBox1.RandomCharacters = "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmopqrstuvwxyz1234567890"
CaptchaBox1.CaptchaTextLength = 5
```

Note: The above code allow for using the English alphabet, both capital and small characters as well as decimal digits to create the captcha text.
The second line limit the number of characters in the captcha text to 5 characters

Refreshing the Captcha:

To refresh the CaptchaBox1 and generate new Captcha text, write the following code in the "btnRefresh_Click" event sub.

```
CaptchaBox1.Refresh()
```

Playing the Captcha "Sound":

To make the user able to listen to the captcha text, a SAPI "spvoice" need to be created and configured. To do so, write the following code in the "btnSound_Click" event sub.

```
Dim SAPI
Dim captchaChar(CaptchaBox1.CaptchaTextLength - 1) As Char
Dim captchaText As String = CaptchaBox1.CaptchaText
Dim count As Integer
For count = 0 To captchaText.Length - 1
    captchaChar(count) = Convert.ToChar(captchaText(count))
Next

Try
    SAPI = CreateObject("SAPI.spvoice")
    For count = 0 To captchaText.Length - 1
        SAPI.speak(captchaChar(count).ToString)
    Next
Catch exc As Exception
    MessageBox.Show("An error has occurred: " & exc.Message)
Finally
End Try
```

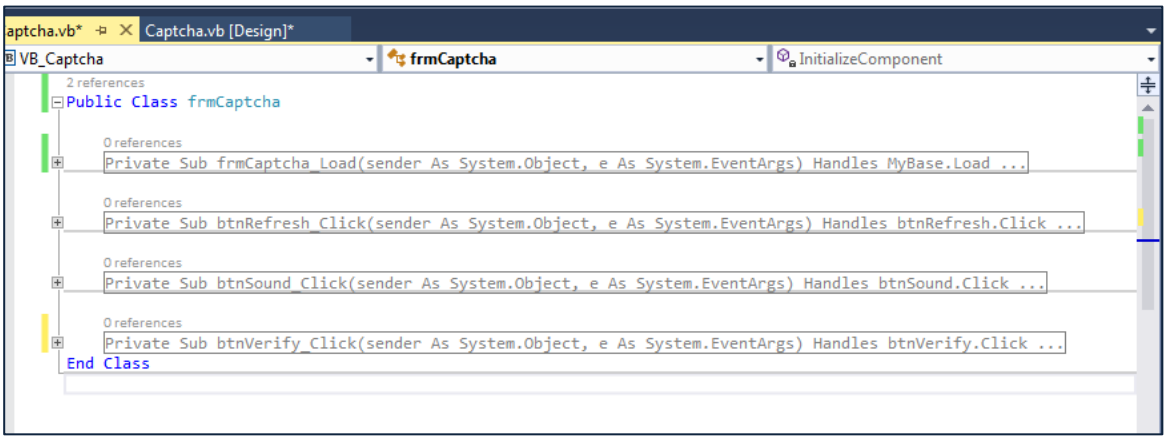
Verifying the Captcha:

To verify the captch text input by the user, you need to compare it with the captcha text created by the CaptchaBox1 object. The following code is used in the "btnVerify_Click" event sub.

```
If txtInp.Text.ToUpper.Equals(CaptchaBox1.CaptchaText.ToUpper) Then
    MessageBox.Show("Correct!")
Else
    MessageBox.Show("Incorrect!")
End If
```

Final Product:

At the end of this session you should have the following codes:



The "frmCaptcha_Load" sub should look like this:

```
Private Sub frmCaptcha_Load(sender As System.Object, e As System.EventArgs) Handles MyBase.Load
    CaptchaBox1.RandomCharacters = "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmopqrstuvwxyz1234567890"
    CaptchaBox1.CaptchaTextLength = 5
End Sub
```

The "btnRefresh_Click" sub should look like this:

```
Private Sub btnRefresh_Click(sender As System.Object, e As System.EventArgs) Handles btnRefresh.Click
    CaptchaBox1.Refresh()
End Sub
```

The "btnSound_Click" sub should look like this:

```
Private Sub btnSound_Click(sender As System.Object, e As System.EventArgs) Handles btnSound.Click
    Dim SAPI
    Dim captchaChar(CaptchaBox1.CaptchaTextLength - 1) As Char
    Dim captchaText As String = CaptchaBox1.CaptchaText
    Dim count As Integer
    For count = 0 To captchaText.Length - 1
        captchaChar(count) = Convert.ToChar(captchaText(count))
    Next

    Try
        SAPI = CreateObject("SAPI.spvoice")
        For count = 0 To captchaText.Length - 1

            SAPI.speak(captchaChar(count).ToString)
        Next
    Catch exc As Exception
        MessageBox.Show("An error has occurred: " & exc.Message)
    Finally

    End Try

End Sub
```

The “btnVerify_Click” sub should look like this:

```
Private Sub btnVerify_Click(sender As System.Object, e As System.EventArgs) Handles btnVerify.Click

    If txtInp.Text.ToUpper.Equals(CaptchaBox1.CaptchaText.ToUpper) Then
        MessageBox.Show("Correct!")
    Else
        MessageBox.Show("Incorrect!")
    End If
End Sub
```