Public Class screencapture

Dim pictureForm As New Form

Dim picturebox As New PictureBox

Dim c = 1

Dim seconds

Dim timer As Boolean = False

Dim TimerInterval As Integer

Dim AutoInterval As Integer

Friend Function ScreenShot(ByVal Screen\_Capture As Boolean, ByVal Area\_Capture As Boolean, ByVal Timer As Boolean, ByVal Auto As Boolean)

If Screen\_Capture = True Then

Area\_Capture = False

Timer = False

Auto = False

ElseIf Area\_Capture = True Then

Screen\_Capture = False

Timer = False

Auto = False

AreaCaptureToolStripMenuItem.PerformClick()

Timer1.Enabled = False

ElseIf Timer = True Then

Screen\_Capture = False

Area\_Capture = False

Auto = False

ElseIf Auto = True Then

Screen\_Capture = False

Area\_Capture = False

Timer = False

End If

If Area\_Capture = False Then

If My.Settings.PlaySound = True And Auto = False Then My.Computer.Audio.Play(Application.StartupPath + "\Camera.wav")

Dim bounds As Rectangle

Dim screenshot\_image As System.Drawing.Bitmap

Dim graph As Graphics

bounds = Screen.PrimaryScreen.Bounds

screenshot\_image = New System.Drawing.Bitmap(bounds.Width, bounds.Height, My.Settings.PixelFormat)

graph = Graphics.FromImage(screenshot\_image)

graph.CopyFromScreen(bounds.X, bounds.Y, 0, 0, bounds.Size, CopyPixelOperation.SourceCopy)

Neutrino\_Screen\_Capture\_Image\_Viewer.PictureBox1.Image = screenshot\_image

If Auto = True Then

Timer1.Enabled = True

If CheckBox2.Checked = True Then Me.Opacity = 100 Else If My.Settings.Trasparent = True Then Me.Opacity = 0 Else Me.Opacity = My.Settings.Transparency

Try

screenshot\_image.Save((My.Settings.Default\_Directory + "\" + My.Settings.FileName + (My.Settings.AutomaticNmber).ToString + "." + (My.Settings.FileFormat).ToString).ToString, My.Settings.FileFormat)

Catch ex As Exception

MsgBox("Sorry ! An error occured to save the image file. Please try again.", MsgBoxStyle.OkOnly + MsgBoxStyle.Critical, "Error")

End Try

My.Settings.AutomaticNmber = My.Settings.AutomaticNmber + 1

Else

Timer1.Enabled = False

Me.Opacity = 100

Me.ShowInTaskbar = True

If My.Settings.MessageShow = True Then MsgBox("Neutrino Screen Capture captured your screen." + vbCrLf + "File Name : " + My.Settings.FileName + vbCrLf + "Folder : " + (My.Settings.Default\_Directory).ToString, MsgBoxStyle.Information, "Neutrino Screen Capture")

Neutrino\_Screen\_Capture\_Image\_Viewer.WindowState = FormWindowState.Maximized

If My.Settings.ImageViewer = True Then Neutrino\_Screen\_Capture\_Image\_Viewer.Show()

Me.ShowInTaskbar = True

If My.Settings.DefaultDir\_Bool = True Then

Try

If My.Settings.autonum = True Then screenshot\_image.Save((My.Settings.Default\_Directory + "\" + My.Settings.FileName + (My.Settings.AutomaticNmber).ToString + "." + (My.Settings.FileFormat).ToString).ToString, My.Settings.FileFormat) Else screenshot\_image.Save((My.Settings.Default\_Directory + "\" + My.Settings.FileName + "." + (My.Settings.FileFormat).ToString).ToString, My.Settings.FileFormat)

Catch ex As Exception

MsgBox("Sorry ! An error occured to save the image file. Please try again.", MsgBoxStyle.OkOnly + MsgBoxStyle.Critical, "Error")

End Try

End If

End If

End If

If My.Settings.NotificationShow = True Then

NotifyIcon1.BalloonTipTitle = "Screen Captured"

NotifyIcon1.BalloonTipText = "Neutrino Screen Capture captured your screen."

NotifyIcon1.ShowBalloonTip(My.Settings.NotificationDuration)

End If

If My.Settings.ClipBoardCopy = True Then Clipboard.SetImage(Neutrino\_Screen\_Capture\_Image\_Viewer.PictureBox1.Image)

If My.Settings.autonum = True Then My.Settings.AutomaticNmber = My.Settings.AutomaticNmber + 1

Return 0

End Function

Private Sub ScreenCaptureToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

'If timer = True And My.Settings.NotificationShow = True Then

'Timer2.Enabled = True

'Timer2.Interval = 800

' ElseIf timer = False Then

' Timer2.Enabled = False

' End If

'If My.Settings.Trasparent = True Then

'Me.Opacity = 0

' ElseIf My.Settings.Trasparent = False Then

' Me.Opacity = My.Settings.Transparency

' End If

'If My.Settings.taskbar = True Then Me.ShowInTaskbar = True Else Me.ShowInTaskbar = False

'Timer1.Enabled = True

' Timer\_Capture.Close()

'Area\_Capture.Close()

'Me.ShowInTaskbar = False

ScreenShot(True, False, False, False)

End Sub

Private Sub NotifyIcon1\_Click(ByVal sender As Object, ByVal e As System.EventArgs)

End Sub

Private Sub NotifyIcon1\_MouseDoubleClick(ByVal sender As System.Object, ByVal e As System.Windows.Forms.MouseEventArgs)

Me.Show()

End Sub

Private Sub ExitToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

End

End Sub

Private Sub AboutNeutrinoScreenCaptureToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

About\_Neutrino\_Screen\_Capture.Show()

End Sub

Private Sub Timer1\_Tick(ByVal sender As System.Object, ByVal e As System.EventArgs)

If radCapScr.Checked = True Then

radCapAra.Checked = False

CheckBox3.Checked = False

CheckBox4.Checked = False

ScreenShot(True, False, False, False)

ElseIf radCapAra.Checked = True Then

radCapScr.Checked = False

CheckBox3.Checked = False

CheckBox4.Checked = False

ScreenShot(False, True, False, False)

ElseIf CheckBox3.Checked = True Then

radCapScr.Checked = False

radCapAra.Checked = False

CheckBox4.Checked = False

ScreenShot(False, False, True, False)

ElseIf CheckBox4.Checked = True Then

radCapScr.Checked = False

radCapAra.Checked = False

CheckBox3.Checked = False

ScreenShot(False, False, False, True)

If CheckBox1.Checked = True Then Timer3.Enabled = True Else Timer3.Enabled = False

End If

End Sub

Private Sub SaveAsToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

Dim savefile As New SaveFileDialog

Try

savefile.Title = "Save File"

savefile.FileName = "Save As..."

savefile.Filter = "JPEG |\*.jpeg |GIF |\*.gif |BMP |\*.bmp |PNG |\*.png |TIFF |\*.tiff |ICO |\*.ico |EMF |\*.emf |EXIF |\*.exif |WMF |\*.wmf"

savefile.InitialDirectory = My.Settings.Default\_Directory

If savefile.ShowDialog = DialogResult.OK Then

Neutrino\_Screen\_Capture\_Image\_Viewer.PictureBox1.Image.Save(savefile.FileName, System.Drawing.Imaging.ImageFormat.Bmp)

End If

Catch ex As Exception

MsgBox("Sorry ! An error occured to save the image file. Please try again.", MsgBoxStyle.OkOnly + MsgBoxStyle.Critical, "Error")

End Try

End Sub

Private Sub AreaCaptureToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

Area\_Capture.Show()

Area\_Capture.Opacity = 100

Area\_Capture.FormBorderStyle = Windows.Forms.FormBorderStyle.Sizable

End Sub

Private Sub screencapture\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)

ComboBox1.SelectedItem = "Seconds"

ComboBox2.SelectedItem = "Seconds"

ComboBox3.SelectedItem = "Seconds"

If CheckBox3.Checked = True Then GroupBox2.Enabled = True Else GroupBox2.Enabled = False

If CheckBox4.Checked = True Then GroupBox3.Enabled = True Else GroupBox3.Enabled = False

If My.Settings.ImageViewer = True Then chkImageViewer.Checked = True

If My.Settings.NotificationShow = True Then

chkNotification.Checked = True

lblDuration1.Visible = True

numNotifyDuration.Visible = True

cmbTimeOptions1.Visible = True

chkMessage.Location = New Point(chkMessage.Location.X, 86)

ElseIf My.Settings.NotificationShow = False Then

chkNotification.Checked = False

lblDuration1.Visible = False

numNotifyDuration.Visible = False

cmbTimeOptions1.Visible = False

chkMessage.Location = New Point(chkMessage.Location.X, 60)

End If

If My.Settings.MessageShow = True Then

chkMessage.Checked = True

ElseIf My.Settings.MessageShow = False Then

chkMessage.Checked = False

End If

If My.Settings.PlaySound = True Then chkSound.Checked = True Else chkSound.Checked = False

If My.Settings.Trasparent = True Then chkTransparent.Checked = True Else chkTransparent.Checked = False

If chkTransparent.Checked = True Then

lblTrans.Visible = False

txtTrans.Visible = False

txtTrans.Text = My.Settings.Transparency

ElseIf chkTransparent.Checked = False Then

lblTrans.Visible = True

txtTrans.Visible = True

End If

If My.Settings.DefaultDir\_Bool = True Then chkAutoSave.Checked = True Else chkAutoSave.Checked = False

If My.Settings.taskbar = True Then chktask.Checked = True Else chktask.Checked = False

If My.Settings.ClipBoardCopy = True Then chkImageClip.Checked = True Else chkImageClip.Checked = False

cmbFileFormat.SelectedItem = (My.Settings.FileFormat).ToString

Select Case My.Settings.PixelFormat

Case Imaging.PixelFormat.Alpha

cmbPixelFormat.SelectedItem = "Alpha"

Case Imaging.PixelFormat.PAlpha

cmbPixelFormat.SelectedItem = "PAlpha"

Case Imaging.PixelFormat.Extended

cmbPixelFormat.SelectedItem = "Extended"

Case Imaging.PixelFormat.Format16bppArgb1555

cmbPixelFormat.SelectedItem = "Argb (16 bpp) 1555"

Case Imaging.PixelFormat.Format32bppArgb

cmbPixelFormat.SelectedItem = "Argb (32 bpp)"

Case Imaging.PixelFormat.Format64bppArgb

cmbPixelFormat.SelectedItem = "Argb (64 bpp)"

Case Imaging.PixelFormat.Format16bppGrayScale

cmbPixelFormat.SelectedItem = "Gray Scale (16 bpp)"

Case Imaging.PixelFormat.Format16bppRgb555

cmbPixelFormat.SelectedItem = "Rgb (16 bpp) 555"

Case Imaging.PixelFormat.Format16bppRgb565

cmbPixelFormat.SelectedItem = "Rgb (16 bpp) 565"

Case Imaging.PixelFormat.Format24bppRgb

cmbPixelFormat.SelectedItem = "Rgb (24 bpp)"

Case Imaging.PixelFormat.Format32bppRgb

cmbPixelFormat.SelectedItem = "Rgb (32 bpp)"

Case Imaging.PixelFormat.Format48bppRgb

cmbPixelFormat.SelectedItem = "Rgb (48 bpp)"

Case Imaging.PixelFormat.Indexed

cmbPixelFormat.SelectedItem = "Indexed"

Case Imaging.PixelFormat.Format1bppIndexed

cmbPixelFormat.SelectedItem = "Indexed (1 bpp)"

Case Imaging.PixelFormat.Format4bppIndexed

cmbPixelFormat.SelectedItem = "Indexed (4 bpp)"

Case Imaging.PixelFormat.Format8bppIndexed

cmbPixelFormat.SelectedItem = "Indexed (8 bpp)"

Case Imaging.PixelFormat.Format32bppPArgb

cmbPixelFormat.SelectedItem = "PArgb (32 bpp)"

Case Imaging.PixelFormat.Format64bppPArgb

cmbPixelFormat.SelectedItem = "PArgb (64 bpp)"

Case Imaging.PixelFormat.DontCare

cmbPixelFormat.SelectedItem = "Don't Care"

End Select

cmbPixelFormat.SelectedItem = (My.Settings.PixelFormat).ToString

cmbTimeOptions1.SelectedItem = "Seconds"

txtFileName.Text = My.Settings.FileName

txtFolder.Text = My.Settings.Default\_Directory

Me.WindowState = FormWindowState.Normal

Me.StartPosition = FormStartPosition.CenterScreen

End Sub

Private Sub ChangeDefaultDirectoryToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

Dim defaultdir As New FolderBrowserDialog

Try

If defaultdir.ShowDialog = DialogResult.OK Then

My.Settings.Default\_Directory = defaultdir.SelectedPath

txtFolder.Text = My.Settings.Default\_Directory

End If

Catch ex As Exception

MsgBox("An error occured. Please try again", MsgBoxStyle.Critical, "Error")

End Try

My.Settings.Save()

End Sub

Private Sub ConvertImageToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

Convert\_Image.Show()

End Sub

Private Sub CaptureScreenToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

ScreenCaptureToolStripMenuItem.PerformClick()

End Sub

Private Sub CaptureAreaToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

AreaCaptureToolStripMenuItem.PerformClick()

End Sub

Private Sub ImageConverterToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

ConvertImageToolStripMenuItem.PerformClick()

End Sub

Private Sub ExitToolStripMenuItem1\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

ExitToolStripMenuItem.PerformClick()

End Sub

Private Sub HelpToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

Help.Show()

End Sub

Private Sub TimerCaptureToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

Timer\_Capture.Show()

radCapScr.Checked = False

radCapAra.Checked = False

CheckBox3.Checked = False

CheckBox4.Checked = False

End Sub

Private Sub MinimizeToSystemTrayToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

Me.Hide()

Area\_Capture.Close()

Timer\_Capture.Close()

Help.Close()

About\_Neutrino\_Screen\_Capture.Close()

End Sub

Private Sub HelpToolStripMenuItem1\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

Help.Show()

End Sub

Private Sub AboutToolStripMenuItem1\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

About\_Neutrino\_Screen\_Capture.Show()

End Sub

Private Sub ToolStripMenuItem11\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

MinimizeToSystemTrayToolStripMenuItem.PerformClick()

End Sub

Private Sub ResizeToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

Me.Show()

End Sub

Private Sub ToolStripMenuItem6\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

ScreenCaptureToolStripMenuItem.PerformClick()

End Sub

Private Sub ToolStripMenuItem7\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

AreaCaptureToolStripMenuItem.PerformClick()

End Sub

Private Sub ToolStripMenuItem24\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

ConvertImageToolStripMenuItem.PerformClick()

End Sub

Private Sub ToolStripMenuItem25\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

Help.Show()

End Sub

Private Sub ToolStripMenuItem26\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

About\_Neutrino\_Screen\_Capture.Show()

End Sub

Private Sub ToolStripMenuItem27\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

MinimizeToSystemTrayToolStripMenuItem.PerformClick()

End Sub

Private Sub ToolStripMenuItem28\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

End

End Sub

Private Sub TimerCaptureToolStripMenuItem1\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

End Sub

Private Sub Timer2\_Tick(ByVal sender As System.Object, ByVal e As System.EventArgs)

If My.Settings.NotificationShow = True Then

Me.NotifyIcon1.BalloonTipTitle = "Timer Capture started"

Me.NotifyIcon1.BalloonTipText = seconds & " Seconds remaining..."

Me.NotifyIcon1.ShowBalloonTip(970)

seconds = seconds - 1

If seconds <= -1 Then

Timer1.Enabled = False

End If

End If

End Sub

Private Sub Button1\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

'If My.Settings.Trasparent = True Then Me.Opacity = 0

'If My.Settings.Trasparent = False Then Me.Opacity = My.Settings.Transparency

'If radCapScr.Checked = True Then

''Timer1.Interval = 1

'ScreenCaptureToolStripMenuItem.PerformClick()

'ElseIf radCapAra.Checked = True Then

'AreaCaptureToolStripMenuItem.PerformClick()

' ElseIf CheckBox3.Checked = True Then

'ScreenCaptureToolStripMenuItem.PerformClick()

'ElseIf CheckBox4.Checked = True Then

'If CheckBox1.Checked = True Then Timer3.Enabled = True

'If CheckBox1.Checked = False And CheckBox2.Checked = True Then

'Timer3.Enabled = False

'Me.Opacity = 100

'End If

'If My.Settings.taskbar = False Then Me.ShowInTaskbar = False Else Me.ShowInTaskbar = True

'Timer1.Enabled = True

'End If

If radCapAra.Checked = False And CheckBox2.Checked = False Then

If My.Settings.Trasparent = True Then Me.Opacity = 0 Else Me.Opacity = My.Settings.Transparency

If My.Settings.taskbar = True Then Me.ShowInTaskbar = True Else Me.ShowInTaskbar = False

Else

Me.Opacity = 100

Me.ShowInTaskbar = True

End If

Timer1.Enabled = True

End Sub

Private Sub CheckBox3\_CheckedChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

If CheckBox3.Checked = True Then

GroupBox2.Enabled = True

radCapScr.Checked = False

radCapAra.Checked = False

CheckBox3.Font = New System.Drawing.Font(CheckBox3.Font, FontStyle.Bold)

End If

If CheckBox3.Checked = False Then

GroupBox2.Enabled = False

CheckBox3.Font = New System.Drawing.Font(CheckBox3.Font, FontStyle.Regular)

End If

End Sub

Private Sub CheckBox4\_CheckedChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

If CheckBox4.Checked = True Then

GroupBox3.Enabled = True

radCapScr.Checked = False

radCapAra.Checked = False

CheckBox3.Checked = False

CheckBox4.Font = New System.Drawing.Font(CheckBox3.Font, FontStyle.Bold)

End If

If CheckBox4.Checked = False Then

GroupBox3.Enabled = False

CheckBox4.Font = New System.Drawing.Font(CheckBox3.Font, FontStyle.Regular)

End If

End Sub

Private Sub NumericUpDown3\_ValueChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

If Val(NumericUpDown3.Value) <> vbNull Then

Dim time = 1

Select Case ComboBox1.SelectedItem

Case "Seconds"

time = Val(NumericUpDown3.Value) \* 1000

Case "Centiseconds"

time = Val(NumericUpDown3.Value) \* 100

Case "Miliseconds"

time = Val(NumericUpDown3.Value)

End Select

TimerInterval = time

Me.Timer1.Interval = TimerInterval

seconds = Me.Timer1.Interval / 1000

timer = True

Else

MsgBox("Please enter a valid time.", , "Error!")

End If

End Sub

Private Sub NumericUpDown1\_ValueChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

If Val(NumericUpDown1.Value) <> vbNull Then

Dim time = 1

Select Case ComboBox1.SelectedItem

Case "Seconds"

time = Val(NumericUpDown1.Value) \* 1000

Case "Centiseconds"

time = Val(NumericUpDown1.Value) \* 100

Case "Miliseconds"

time = Val(NumericUpDown1.Value)

End Select

Me.Timer1.Interval = time

seconds = Me.Timer1.Interval / 1000

Else

MsgBox("Please enter a valid time.", , "Error!")

End If

If CheckBox1.Checked = False Then CheckBox2.Checked = True

End Sub

Private Sub CheckBox1\_CheckedChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

If CheckBox1.Checked = True Then

CheckBox2.Checked = False

CheckBox1.Font = New System.Drawing.Font(CheckBox1.Font, FontStyle.Bold)

End If

If CheckBox1.Checked = False Then

CheckBox2.Checked = True

CheckBox1.Font = New System.Drawing.Font(CheckBox1.Font, FontStyle.Regular)

End If

End Sub

Private Sub NumericUpDown2\_ValueChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

Select Case ComboBox3.SelectedItem

Case "Seconds"

Timer3.Interval = NumericUpDown2.Value \* 1000

Case "Centiseconds"

Timer3.Interval = NumericUpDown2.Value \* 100

Case "Miliseconds"

Timer3.Interval = NumericUpDown2.Value

End Select

End Sub

Private Sub Timer3\_Tick(ByVal sender As System.Object, ByVal e As System.EventArgs)

My.Computer.Audio.Play(Application.StartupPath + "\Camera.wav")

Timer1.Enabled = False

Timer3.Enabled = False

If My.Settings.ImageViewer = True Then

Neutrino\_Screen\_Capture\_Image\_Viewer.Show()

Neutrino\_Screen\_Capture\_Image\_Viewer.WindowState = FormWindowState.Maximized

End If

If My.Settings.MessageShow = True Then MsgBox("Neutrino Screen Capture captured your screen." + vbCrLf + "File Name : " + My.Settings.FileName + vbCrLf, MsgBoxStyle.Information, "Neutrino Screen Capture")

Me.Opacity = 100

Me.ShowInTaskbar = True

End Sub

Private Sub CheckBox2\_CheckedChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

If CheckBox2.Checked = True Then CheckBox2.Font = New System.Drawing.Font(CheckBox2.Font, FontStyle.Bold)

If CheckBox2.Checked = False Then CheckBox2.Font = New System.Drawing.Font(CheckBox2.Font, FontStyle.Regular)

End Sub

Private Sub radCapScr\_CheckedChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

If radCapScr.Checked = True Then radCapScr.Font = New System.Drawing.Font(radCapScr.Font, FontStyle.Bold)

If radCapScr.Checked = False Then radCapScr.Font = New System.Drawing.Font(radCapScr.Font, FontStyle.Regular)

End Sub

Private Sub radCapAra\_CheckedChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

If radCapAra.Checked = True Then radCapAra.Font = New System.Drawing.Font(radCapAra.Font, FontStyle.Bold)

If radCapAra.Checked = False Then radCapAra.Font = New System.Drawing.Font(radCapAra.Font, FontStyle.Regular)

End Sub

Private Sub Button2\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

My.Computer.Audio.Play(Application.StartupPath + "\Camera.wav")

Timer1.Enabled = False

Timer3.Enabled = False

If My.Settings.ImageViewer = True Then

Neutrino\_Screen\_Capture\_Image\_Viewer.Show()

Neutrino\_Screen\_Capture\_Image\_Viewer.WindowState = FormWindowState.Maximized

End If

If My.Settings.MessageShow = True Then MsgBox("Neutrino Screen Capture captured your screen." + vbCrLf + "File Name : " + My.Settings.FileName + vbCrLf, MsgBoxStyle.Information, "Neutrino Screen Capture")

Me.Opacity = 100

Me.ShowInTaskbar = True

End Sub

Private Sub LinkLabel1\_LinkClicked(ByVal sender As System.Object, ByVal e As System.Windows.Forms.LinkLabelLinkClickedEventArgs)

File\_\_Name\_Editor.Show()

End Sub

Private Sub ComboBox4\_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

Select Case cmbFileFormat.SelectedItem

Case "JPEG"

My.Settings.FileFormat = System.Drawing.Imaging.ImageFormat.Jpeg

Case "PNG"

My.Settings.FileFormat = System.Drawing.Imaging.ImageFormat.Png

Case "BMP"

My.Settings.FileFormat = System.Drawing.Imaging.ImageFormat.Bmp

Case "GIF"

My.Settings.FileFormat = System.Drawing.Imaging.ImageFormat.Gif

Case "TIFF"

My.Settings.FileFormat = System.Drawing.Imaging.ImageFormat.Tiff

Case "EMF"

My.Settings.FileFormat = System.Drawing.Imaging.ImageFormat.Emf

Case "EXIF"

My.Settings.FileFormat = System.Drawing.Imaging.ImageFormat.Exif

Case "WMF"

My.Settings.FileFormat = System.Drawing.Imaging.ImageFormat.Wmf

End Select

My.Settings.Save()

End Sub

Private Sub cmbPixelFormat\_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

Select Case cmbPixelFormat.SelectedItem

Case "Alpha"

My.Settings.PixelFormat = System.Drawing.Imaging.PixelFormat.Alpha

Case "PAlpha"

My.Settings.PixelFormat = System.Drawing.Imaging.PixelFormat.PAlpha

Case "Canonical"

My.Settings.PixelFormat = System.Drawing.Imaging.PixelFormat.Canonical

Case "Extended"

My.Settings.PixelFormat = System.Drawing.Imaging.PixelFormat.Extended

Case "Argb (16 bpp) 1555"

My.Settings.PixelFormat = System.Drawing.Imaging.PixelFormat.Format16bppArgb1555

Case "Argb (32 bpp)"

My.Settings.PixelFormat = System.Drawing.Imaging.PixelFormat.Format32bppArgb

Case "Argb (64 bpp)"

My.Settings.PixelFormat = System.Drawing.Imaging.PixelFormat.Format64bppArgb

Case "Gray Scale (16 bpp)"

My.Settings.PixelFormat = System.Drawing.Imaging.PixelFormat.Format16bppGrayScale

Case "Rgb (16 bpp) 555"

My.Settings.PixelFormat = System.Drawing.Imaging.PixelFormat.Format16bppRgb555

Case "Rgb (16 bpp) 565"

My.Settings.PixelFormat = System.Drawing.Imaging.PixelFormat.Format16bppRgb565

Case "Rgb (24 bpp)"

My.Settings.PixelFormat = System.Drawing.Imaging.PixelFormat.Format24bppRgb

Case "Rgb (32 bpp)"

My.Settings.PixelFormat = System.Drawing.Imaging.PixelFormat.Format32bppRgb

Case "Rgb (48 bpp)"

My.Settings.PixelFormat = System.Drawing.Imaging.PixelFormat.Format48bppRgb

Case "Indexed"

My.Settings.PixelFormat = System.Drawing.Imaging.PixelFormat.Indexed

Case "Indexed (1 bpp)"

My.Settings.PixelFormat = System.Drawing.Imaging.PixelFormat.Format1bppIndexed

Case "Indexed (1 bpp)"

My.Settings.PixelFormat = System.Drawing.Imaging.PixelFormat.Format1bppIndexed

Case "Indexed (4 bpp)"

My.Settings.PixelFormat = System.Drawing.Imaging.PixelFormat.Format4bppIndexed

Case "Indexed (8 bpp)"

My.Settings.PixelFormat = System.Drawing.Imaging.PixelFormat.Format8bppIndexed

Case "PArgb (32 bpp)"

My.Settings.PixelFormat = System.Drawing.Imaging.PixelFormat.Format32bppPArgb

Case "PArgb (64 bpp)"

My.Settings.PixelFormat = System.Drawing.Imaging.PixelFormat.Format64bppPArgb

Case "Don't Care"

My.Settings.PixelFormat = System.Drawing.Imaging.PixelFormat.DontCare

End Select

My.Settings.Save()

End Sub

Private Sub GroupBox4\_Enter(ByVal sender As System.Object, ByVal e As System.EventArgs)

End Sub

Private Sub Button3\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

Dim openFolder As New FolderBrowserDialog

If openFolder.ShowDialog = DialogResult.OK Then

My.Settings.Default\_Directory = openFolder.SelectedPath

txtFolder.Text = My.Settings.Default\_Directory

End If

My.Settings.Save()

End Sub

Private Sub CheckBox5\_CheckedChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

If chkAutoSave.Checked = True Then

My.Settings.DefaultDir\_Bool = True

chkAutoSave.Font = New System.Drawing.Font(chkAutoSave.Font, FontStyle.Bold)

ElseIf chkAutoSave.Checked = False Then

My.Settings.DefaultDir\_Bool = False

chkAutoSave.Font = New System.Drawing.Font(chkAutoSave.Font, FontStyle.Regular)

End If

End Sub

Private Sub txtFileName\_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

My.Settings.FileName = txtFileName.Text

My.Settings.Save()

End Sub

Private Sub CheckBox6\_CheckedChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

If chkImageViewer.Checked = True Then

My.Settings.ImageViewer = True

chkImageViewer.Font = New Font(chkImageViewer.Font, FontStyle.Bold)

Else

My.Settings.ImageViewer = False

chkImageViewer.Font = New Font(chkImageViewer.Font, FontStyle.Regular)

End If

My.Settings.Save()

End Sub

Private Sub chkNotification\_CheckedChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

Dim DurationPoint As New Point

DurationPoint = New System.Drawing.Point(133, 60)

If chkNotification.Checked = True Then

My.Settings.NotificationShow = True

chkNotification.Font = New Font(chkNotification.Font, FontStyle.Bold)

lblDuration1.Visible = True

numNotifyDuration.Visible = True

cmbTimeOptions1.Visible = True

chkMessage.Location = New Point(chkMessage.Location.X, 86)

ElseIf chkNotification.Checked = False Then

My.Settings.NotificationShow = False

chkNotification.Font = New Font(chkNotification.Font, FontStyle.Regular)

lblDuration1.Visible = False

numNotifyDuration.Visible = False

cmbTimeOptions1.Visible = False

chkMessage.Location = New Point(chkMessage.Location.X, 60)

End If

My.Settings.Save()

End Sub

Private Sub numNotifyDuration\_ValueChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

Select Case cmbTimeOptions1.SelectedItem

Case "Seconds"

My.Settings.NotificationDuration = numNotifyDuration.Value \* 1000

Case "Centiseconds"

My.Settings.NotificationDuration = numNotifyDuration.Value \* 100

Case "Miliseconds"

My.Settings.NotificationDuration = numNotifyDuration.Value

End Select

My.Settings.Save()

End Sub

Private Sub chkMessage\_CheckedChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

If chkMessage.Checked = True Then

My.Settings.MessageShow = True

chkMessage.Font = New Font(chkMessage.Font, FontStyle.Bold)

ElseIf chkMessage.Checked = False Then

My.Settings.MessageShow = False

chkMessage.Font = New Font(chkMessage.Font, FontStyle.Regular)

End If

My.Settings.Save()

End Sub

Private Sub chkSound\_CheckedChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

If chkSound.Checked = True Then

My.Settings.PlaySound = True

chkSound.Font = New Font(chkSound.Font, FontStyle.Bold)

Else

My.Settings.PlaySound = False

chkSound.Font = New Font(chkSound.Font, FontStyle.Regular)

End If

My.Settings.Save()

End Sub

Private Sub chkTransparent\_CheckedChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

If chkTransparent.Checked = True Then

My.Settings.Trasparent = True

chkTransparent.Font = New Font(chkTransparent.Font, FontStyle.Bold)

lblTrans.Visible = False

txtTrans.Visible = False

ElseIf chkTransparent.Checked = False Then

My.Settings.Trasparent = False

chkTransparent.Font = New Font(chkTransparent.Font, FontStyle.Regular)

lblTrans.Visible = True

txtTrans.Visible = True

End If

My.Settings.Save()

End Sub

Private Sub txtTrans\_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

If txtTrans.Text > 100 Then

MsgBox("Invalid Transparency. Transparency must be less than 100.", MsgBoxStyle.Exclamation, "Error")

txtTrans.Text = 100

ElseIf txtTrans.Text < 100 Then

My.Settings.Transparency = 100 - txtTrans.Text

My.Settings.Save()

End If

End Sub

Private Sub TextBox1\_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

If txtTrans.Text > 0 Then

My.Settings.Transparency = txtTrans.Text

Else

MsgBox("Enter a valid value. Transparency must be greater than 0.", MsgBoxStyle.Exclamation, "Error")

End If

My.Settings.Save()

End Sub

Private Sub TabPage1\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

End Sub

Private Sub MenuStrip1\_ItemClicked(ByVal sender As System.Object, ByVal e As System.Windows.Forms.ToolStripItemClickedEventArgs)

End Sub

Private Sub Button3\_Click\_1(ByVal sender As System.Object, ByVal e As System.EventArgs)

Dim defaultdir As New FolderBrowserDialog

Try

If defaultdir.ShowDialog = DialogResult.OK Then

My.Settings.Default\_Directory = defaultdir.SelectedPath

txtFolder.Text = My.Settings.Default\_Directory

End If

Catch ex As Exception

MsgBox("An error occured. Please try again", MsgBoxStyle.Critical, "Error")

End Try

My.Settings.Save()

End Sub

Private Sub chkAutoNum\_CheckedChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

If chkAutoNum.Checked = True Then

lblAuto.Visible = True

numAuto.Visible = True

My.Settings.autonum = True

My.Settings.FileName = My.Settings.FileName + (My.Settings.AutomaticNmber).ToString

txtFileName.Text = My.Settings.FileName

chkAutoNum.Font = New System.Drawing.Font(chkAutoNum.Font, FontStyle.Bold)

ElseIf chkAutoNum.Checked = False Then

lblAuto.Visible = False

numAuto.Visible = False

My.Settings.autonum = True

My.Settings.FileName = My.Settings.FileName.Replace(My.Settings.AutomaticNmber, "")

txtFileName.Text = My.Settings.FileName

chkAutoNum.Font = New System.Drawing.Font(chkAutoNum.Font, FontStyle.Regular)

End If

My.Settings.Save()

End Sub

Private Sub CheckBox6\_CheckedChanged\_1(ByVal sender As System.Object, ByVal e As System.EventArgs)

If chktask.Checked = True Then

My.Settings.taskbar = True

chktask.Font = New Font(chktask.Font, FontStyle.Bold)

Else

My.Settings.taskbar = False

chktask.Font = New Font(chktask.Font, FontStyle.Regular)

End If

My.Settings.Save()

End Sub

Private Sub numAuto\_ValueChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

My.Settings.AutomaticNmber = numAuto.Value

My.Settings.Save()

End Sub

Private Sub chkImageClip\_CheckedChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

If chkImageClip.Checked = True Then

chkImageClip.Font = New Font(chkImageClip.Font, FontStyle.Bold)

My.Settings.ClipBoardCopy = True

Else

chkImageClip.Font = New Font(chkImageClip.Font, FontStyle.Regular)

My.Settings.ClipBoardCopy = False

End If

End Sub

Private Sub screencapture\_Load\_1(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

End Sub

Private Sub ToolStripMenuItem27\_Click\_1(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles ToolStripMenuItem27.Click

End Sub

End Class