APLIKASI MENGUNCI (LOCK) USB UNTUK MENGAMANKAN PORT USB MENGGUNAKAN PEMROGRAMAN VISUAL BASIC. NET

Oleh :

NACHSIFUL MUSLICH
41506010043

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS MERCU BUANA
JAKARTA
2011

APLIKASI MENGUNCI (LOCK) USB
UNTUK MENGAMANKAN PORT USB MENGGUNAKAN
PEMROGRAMAN VISUAL BASIC. NET

Laporan Tugas Akhir

Diajukan Untuk Melengkapi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer

Oleh :

NACHSIFUL MUSLICH
41506010043

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS MERCU BUANA
JAKARTA
2011

LEMBAR PERNYATAAN

Yang bertanda tangan di bawah ini :

NIM	: 41506010043
Nama	: NACHSIFUL MUSLICH
Judul Skripsi	: APLIKASI MENGUNCI (LOCK) USB UNTUK MENGAMANKAN PORT USB MENGGUNAKAN PEMROGRAMAN VISUAL BASIC. NET

Menyatakan bahwa skripsi tersebut diatas adalah hasil karya saya sendiri dan bukan plagiat. Apabila ternyata ditemukan didalam laporan skripsi saya terdapat unsur plagiat, maka saya siap untuk mendapatkan sanksi akademik yang terkait dengan hal tersebut.

Jakarta, 7 Februari 2011

(Nachsiful M.)

LEMBAR PERSETUJUAN

NIM : 41506010043

Nama : NACHSIFUL MUSLICH

Judul Skripsi : APLIKASI MENGUNCI (LOCK) USB UNTUK MENGAMANKAN PORT USB MENGGUNAKAN PEMROGRAMAN VISUAL BASIC. NET

SKRIPSI INI TELAH DIPERIKSA DAN DISETUJUI

JAKARTA, ………………………

Ratna Mutu Manikam, S.Kom., MT

Pembimbing

Devi Fitrianah, S.Kom.,MTI

KaProdi Teknik Informatika

Ida Nurhaida, ST.,MT

Koord. TA Teknik Informatika

LEMBAR PERSETUJUAN SIDANG

NIM : 41506010043

Nama : NACHSIFUL MUSLICH

Judul Skripsi : APLIKASI MENGUNCI (LOCK) USB UNTUK MENGAMANKAN PORT USB DENGAN MENGGUNAKAN PEMROGRAMAN VISUAL BASIC. NET

SKRIPSI INI TELAH DIPERIKSA DAN DISETUJUI

JAKARTA, ……………………….

Ratna Mutu Manikam, S.Kom., MT

Pembimbing

ABSTRAK

Kebutuhan yang sangat penting untuk komputer adalah kebutuhan keamanan. Salah satunya yang sering kita gunakan sehari-hari untuk menyalin atau bertukar data yaitu menggunakan Port USB (*Universal Serial Bus*). Kekurangan port USB adalah tidak adanya keamanan saat terjadi hubungan antara komputer dengan *peripheral*. Sehingga peran sistem operasi yang menangani *hardware* dan *software* sangat dibutuhkan.

Secara umum sistem operasi *Windows* memiliki *Registry* untuk menampung semua konfigurasi perangkat keras, perangkat lunak, asosiasi ekstensi berkas hingga preferensi pengguna. *Registry* merupakan pengganti berkas-berkas konfigurasi `*.INI` yang digunakan dalam sistem *Windows* 16-bit. Untuk mengakses port USB menjadi aktif/nonaktif harus mencari *subtree* yang berhubungan dengan port USB. Setelah itu mengakses nilai *value* dari data *Start* dan mengisinya dengan *value* 3 untuk aktif dan *value* 4 untuk nonaktif. Kemudian bagian dari port USB yaitu Vbus akan diputus aksesnya ke komputer. Bagian port USB yang lain yaitu D+, D- dan *ground* secara otomatis tidak akan aktif. Bagian D+ dan D- digunakan untuk menangani data masuk/keluar, sedangkan *ground* untuk mengamankan arus listrik. Kelebihan aplikasi ini dapat mengakses *Registry* dengan cepat dan mengamankan port USB dengan aplikasi yang diproteksi dengan *password*.

Kata Kunci : Keamanan, *Port USB, Registry,* Sistem Operasi

ABSTRACT

A very important requirement is the need for computer security. One of them is that often we use everyday to copy or exchange data that is using a USB port (Universal Serial Bus). Lack of USB ports is the lack of security when there is relationship between computer and peripherals. So the role of the operating system that handles hardware and software is needed.

In general, the Windows operating system has a registry to hold all the configuration of hardware, software, file extension associations to user preferences. Registry is an alternate configuration files *.ini is used in 16-bit Windows system. To access the USB port to be on / off to find the subtree associated with a USB port. After that access the data value of the Start value and fill it with the value 3 for the active and the value 4 for the disabled. Then part of the USB port that is Vbus is disconnected access to a computer. Another part of the USB port of D +, D-and the ground is not automatically activated. Part D + and D-are used to handle data entry / exit, while the ground to secure the electrical current. The advantage of this application can access the Registry quickly and secure the USB port with an application that is protected with a password.

Keywords: Security, USB Port, Registry, Operating System

KATA PENGANTAR

Alhamdulillah ke hadirat Allah SWT yang telah memberikan saya kesehatan, kemudahan dan segala nikmat yang ada, sehingga penulis dapat menyelesaikan tugas akhir ini dengan judul " **Aplikasi Mengunci (Lock) USB untuk Mengamankan Port USB Menggunakan Pemrograman Visual Basic .Net** " tepat pada waktunya. Rasa haru yang tak terhingga setelah bisa mengerjakan tugas akhir ini dengan suka duka. Mendapatkan pengalaman yang tak ternilai dalam penyusunan tugas akhir ini yang merupakan salah satu persyaratan untuk menyelesaikan program studi S1 pada Jurusan Teknik Informatika Universitas Mercu Buana.

Selama penulisan laporan tugas akhir ini, penulis banyak mengalami hambatan dan keterbatasan dalam persiapan, penyusunan maupun tahap penyelesaian. Oleh karena itu, dengan selesainya laporan tugas akhir ini maka penulis mengucapkan terima kasih yang sebesar-besarnya kepada semua pihak yang turut membantu dalam penyusunan skripsi ini. Pada kesempatan ini penulis mengucapkan terima kasih yang sebesar-besarnya kepada :

1. Ibu Ratna Mutu Manikam, S.Kom., MT., selaku dosen pembimbing yang telah memberikan arahan dan bimbingan dalam penyusunan laporan tugas akhir ini.
2. Ibu Devi Fitrianah, S.Kom., MTI., selaku Ketua Program Studi Teknik Informatika, Universitas Mercu Buana.

3. Ibu Ida Nurhaida, ST.,MT, selaku koordinator tugas akhir program studi Teknik Informatika Universitas Mercu Buana.
4. Keluarga Tercinta, kedua orang tua H. Solichin Ichsan dan Hj. Chayana yang selalu mengiringi penulis dalam restu dan do'a.
5. Kakak Ipar Drs. Ibni Affan dan kakakku Dra. Elvy Juheni yang telah memberikan bantuan materi, semangat dan masukan untuk menyelesaikan tugas akhir ini.
6. Saudara – saudaraku Achmad Najich, Helmi Musonif, Zulham Baisuni, Lailil Irzaq, Chusnul Hanafi, dan adikku M. Yusuf Abdila, terima kasih untuk dukungan materi dan nasehatnya yang menjadikan penulis kuat dalam menyusun tugas akhir ini.
7. Bapak Ahmad Kodar, Drs., MT., selaku dosen Pembimbing Akademik Program Studi Teknik Informatika angkatan 2006.
8. Bapak Tri Daryanto, S.kom., MT., memberikan penulis bantuan dan inspirasi dalam mengerjakan aplikasi.
9. Bapak Indrianto, S Kom., MT., memberikan penulis bantuan dan inspirasi dalam mengerjakan aplikasi.
10. Bapak Sugiyono dan Ibu Tumiyem yang telah memberikan bantuan dan doa restunya.
11. Ari Siskayanti, yang telah memberikan bantuan, doa dan semangat setiap hari untuk menyelesaikan tugas akhir ini.
12. Rina Dwi Susanti yang telah memberikan dukungan moril agar selalu kuat dan tetap semangat untuk menyelesaikan tugas akhir.

13. Sahabat-sahabatku Ichsan wahyudi, Debby Fitria, Fitri laksitasari Iriyanti, Jatu hestiningtyas, Aisyah, Bayu Kurniawan, yang memberikan bantuan dan masukan kepada penulis.
14. Semua mahasiswa/i Teknik Informatika khususnya angkatan 2006 teman seperjuangan yang telah banyak berbagi pengalaman, ilmu, dan juga semangat.
15. Semua pihak yang telah membantu baik langsung maupun tidak langsung dalam pembuatan laporan tugas akhir ini yang tidak dapat disebutkan penulis satu persatu.

Semoga skripsi ini dapat bermanfaat bagi siapa saja dan tentunya agar dapat dikembangkan lebih jauh di masa mendatang. Akhir kata, kepada semua pihak yang telah membantu terwujudnya tugas akhir ini semoga Allah selalu melimpahkan rahmat dan karunia-Nya. Amin.

Jakarta, 7 Februari 2011

Penulis

DAFTAR ISI

BAB II LANDASAN TEORI

DAFTAR GAMBAR

DAFTAR TABEL

LAMPIRAN

Listing program pada *form Splash screen*

```
Public Class SplashScreen
    Inherits System.Windows.Forms.Form

    Private Sub SplashScreen_Load(ByVal sender As Object, ByVal e As System.EventArgs)
Handles Me.Load
        If My.Application.Info.Title <> "" Then
            LblAplikasi.Text = My.Application.Info.Title
        Else
            LblAplikasi.Text =
System.IO.Path.GetFileNameWithoutExtension(My.Application.Info.AssemblyName)
        End If
        LblVersion.Text = System.String.Format(LblVersion.Text,
My.Application.Info.Version.Major, My.Application.Info.Version.Minor)
        LblCopyright.Text = My.Application.Info.Copyright
    End Sub

    Private Sub TimerSplashScreen_Tick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles TimerSplashScreen.Tick
        ProgressBar1.Value += 4
        If ProgressBar1.Value = 100 Then
            TimerSplashScreen.Dispose()
            Me.Visible = False
            FrmLogin.Hide()
            FrmLogin.NotifyIconLogin.ShowBalloonTip(1, "Pemberitahuan", "Security Port
USB Aktif", ToolTipIcon.Info)
        End If
    End Sub
End Class
```

Listing program pada *form login*

```
Public Class FrmLogin
    Inherits System.Windows.Forms.Form
    Private WithEvents Glass As New fsButton.rtaGlassEffect
    Private Declare Function RemoveMenu Lib "user32" (ByVal hMenu As IntPtr, ByVal
nPosition As Integer, ByVal wFlags As Long) As IntPtr
    Private Declare Function GetSystemMenu Lib "user32" (ByVal hWnd As IntPtr, ByVal
bRevert As Boolean) As IntPtr
    Private Declare Function GetMenuItemCount Lib "user32" (ByVal hMenu As IntPtr) As
Integer
    Private Declare Function DrawMenuBar Lib "user32" (ByVal hwnd As IntPtr) As
Boolean
    Private Const MF_BYPOSITION = &H400
    Private Const MF_REMOVE = &H1000
    Private Const MF_DISABLED = &H2
    Dim MyConnection As New Security_Port_USB.CLS.DatabaseConneciton
    Dim Nama As String
    Dim Password As String

    Public Sub DisableCloseButton(ByVal hwnd As IntPtr)
        Dim hMenu As IntPtr
        Dim menuItemCount As Integer
        hMenu = GetSystemMenu(hwnd, False)
        menuItemCount = GetMenuItemCount(hMenu)
        Call RemoveMenu(hMenu, menuItemCount - 1, _
        MF_DISABLED Or MF_BYPOSITION)
        Call RemoveMenu(hMenu, menuItemCount - 2, _
        MF_DISABLED Or MF_BYPOSITION)
        Call DrawMenuBar(hwnd)
    End Sub

    Private Sub FrmLogin_KeyDown(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyEventArgs) Handles Me.KeyDown
        If e.Alt = True And e.KeyCode = Keys.F4 Then
            e.Handled = True
        End If
    End Sub


    Private Sub FrmLogin_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        Dim oReg As RegistryKey = Registry.CurrentUser
        Dim oKey As RegistryKey =
oReg.OpenSubKey("Software\Microsoft\Windows\CurrentVersion\Run", True)
        oKey.SetValue("Security Port USB", "C:\Program Files\Security Port
USB\Security Port USB.exe")
        DisableCloseButton(Me.Handle)
        TxtUsername.Focus()
        TxtUsername.MaxLength = 60
        TxtPassword.MaxLength = 10
        If fsButton.rtaGlassEffect.GlassEnabled Then
            Glass.TopBarSize = 60
            Glass.LeftBarSize = 40
            Glass.RightBarSize = 20
            Glass.BottomBarSize = 20
            Glass.ShowEffect(Me, LblTitle, PictureBox)
        End If
    End Sub

    Sub CekUser()
        '=============================================================
        'Fungsi untuk mengecek nama user apakah ada pada database table record user
        'yang dimana nama username = txtusername.text
        objDataTable.Clear()
        StrSql = ("SELECT * FROM [Login] where [Username] = " & "'" &
Trim(TxtUsername.Text) & "'")
        objCommand = New OleDbCommand
        objCommand.Connection = MyConnection.Open
        objCommand.CommandType = CommandType.Text
        objCommand.CommandText = StrSql
        objDataAdapter = New OleDbDataAdapter(objCommand)
        objDataAdapter.Fill(objDataSet, "Mdt_Username")
        objDataTable = objDataSet.Tables("Mdt_Username")
        MyConnection.Close()
        '=============================================================
    End Sub

    Sub CekPassword()
```

```
        '=============================================================
        'Fungsi mencari nilai value password & Level
        objCommand = Myconnection.Open.CreateCommand
        objCommand.CommandText = ("select [Username],[Password] " _
                                & " from [Login] where [Username]='" &
Trim(TxtUsername.Text) & "'")
        objDataReader = objCommand.ExecuteReader
        objDataReader.Read()
        Nama = objDataReader.Item("Username") 'Mendefinisikan Id = Id yang ada pada
database
        Password = objDataReader.Item("Password") 'Mendefinisikan Password = password
yang ada pada database
        Myconnection.Close()
        '=============================================================
    End Sub

    Private Sub TxtUsername_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles TxtUsername.KeyPress
        If e.KeyChar = Chr(13) Then TxtPassword.Select()
    End Sub

    Private Sub BtnLogin_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnLogin.Click
        If TxtUsername.Text = "" Or TxtPassword.Text = "" Then
            FrmPeringatanGagal.ShowDialog()
            TxtUsername.Clear()
            TxtPassword.Clear()
            TxtUsername.Focus()
        Else
            Call CekUser()
            If objDataTable.Rows.Count <= 0 Then
                FrmPeringatanGagal.ShowDialog()
                TxtUsername.Clear()
                TxtPassword.Clear()
                TxtUsername.Focus()
            Else
                Call CekPassword()
                If Password <> Trim(TxtPassword.Text) Then
                    FrmPeringatanGagal.ShowDialog()
                    TxtUsername.Clear()
                    TxtPassword.Clear()
                    TxtUsername.Focus()
                Else
                    Me.Hide()
                    FrmMain.ShowDialog()
                End If
            End If
        End If
    End Sub

    Private Sub BtnCancel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnCancel.Click
        Me.Hide()
        TxtUsername.Clear()
        TxtPassword.Clear()
    End Sub

    Private Sub TxtPassword_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles TxtPassword.KeyPress
        If e.KeyChar = Chr(13) Then
            BtnLogin.PerformClick()
        End If
    End Sub
    Private Sub NotifyIconLogin_MouseDoubleClick(ByVal sender As System.Object, ByVal
e As System.Windows.Forms.MouseEventArgs) Handles NotifyIconLogin.MouseDoubleClick
        Me.Show()
        TxtUsername.Focus()
        TxtUsername.Clear()
        TxtPassword.Clear()
        FrmMain.Hide()
        FrmAbout.Hide()
        FrmInfo.Hide()
        FrmChangePassword.Hide()
        FrmManage.Hide()
    End Sub
End Class
```

Listing program pada *form main*

```
Public Class FrmMain
    Inherits System.Windows.Forms.Form
    Private WithEvents Glass As New fsButton.rtaGlassEffect
    Private Declare Function RemoveMenu Lib "user32" (ByVal hMenu As IntPtr, ByVal
nPosition As Integer, ByVal wFlags As Long) As IntPtr
    Private Declare Function GetSystemMenu Lib "user32" (ByVal hWnd As IntPtr, ByVal
bRevert As Boolean) As IntPtr
    Private Declare Function GetMenuItemCount Lib "user32" (ByVal hMenu As IntPtr) As
Integer
    Private Declare Function DrawMenuBar Lib "user32" (ByVal hwnd As IntPtr) As
Boolean
    Private Const MF_BYPOSITION = &H400
    Private Const MF_REMOVE = &H1000
    Private Const MF_DISABLED = &H2

    Public Sub DisableCloseButton(ByVal hwnd As IntPtr)
        Dim hMenu As IntPtr
        Dim menuItemCount As Integer
        hMenu = GetSystemMenu(hwnd, False)
        menuItemCount = GetMenuItemCount(hMenu)
        Call RemoveMenu(hMenu, menuItemCount - 1, _
        MF_DISABLED Or MF_BYPOSITION)
        Call RemoveMenu(hMenu, menuItemCount - 2, _
        MF_DISABLED Or MF_BYPOSITION)
        Call DrawMenuBar(hwnd)
    End Sub

    Private Sub FrmMain_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        DisableCloseButton(Me.Handle)
        Me.Hide()
        If fsButton.rtaGlassEffect.GlassEnabled Then
            Glass.TopBarSize = 70
            Glass.LeftBarSize = 20
            Glass.RightBarSize = 20
            Glass.BottomBarSize = 25
            Glass.ShowEffect(Me, LblTitle, PicLogo)
        End If
        LblUser.Text = My.User.Name
        Dim regKey As RegistryKey
        regKey =
Registry.LocalMachine.OpenSubKey("SYSTEM\CurrentControlSet\Services\USBSTOR", True)
        regKey.SetValue("Start", 4)
        regKey.Close()
        LblStatusGambar.Text = "LOCKED"
        BtnLock.Enabled = False
        BtnUnlock.Enabled = True
    End Sub

    Private Sub Timer_Tick(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Timer.Tick
        LblTgl.Text = Now().ToString("dd MMMM yy")
        LblJam.Text = Now().ToString("hh:MM")
    End Sub

    Private Sub BtnAdmin_Click(ByVal sender As Object, ByVal e As System.EventArgs)
Handles BtnAdmin.Click
        FrmChangePassword.ShowDialog()
    End Sub

    Private Sub BtnLock_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnLock.Click
        LblStatusGambar.Text = "LOCKED"
        BtnLock.Enabled = False
        BtnUnlock.Enabled = True
        Dim regKey As RegistryKey
        regKey =
Registry.LocalMachine.OpenSubKey("SYSTEM\CurrentControlSet\Services\USBSTOR", True)
        regKey.SetValue("Start", 4)
        regKey.Close()
    End Sub

    Private Sub BtnUnlock_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnUnlock.Click
        LblStatusGambar.Text = "UNLOCKED"
        BtnUnlock.Enabled = False
        BtnLock.Enabled = True
        Dim regKey As RegistryKey
        regKey =
Registry.LocalMachine.OpenSubKey("SYSTEM\CurrentControlSet\Services\USBSTOR", True)
        regKey.SetValue("Start", 3)
        regKey.Close()
    End Sub
End Class
```

Listing program pada *form manage*

```
Public Class FrmManage
    Inherits System.Windows.Forms.Form
    Private WithEvents Glass As New fsButton.rtaGlassEffect
    Const WM_COMMAND As Int32 = &H111
    Const MF_ENABLED As Int32 = &H0
    Const MF_GRAYED As Int32 = &H1
    Const LVM_FIRST As Int32 = &H1000
    Const LVM_DELETEITEM As Int32 = (LVM_FIRST + 8)
    Const LVM_SORTITEMS As Int32 = (LVM_FIRST + 48)
    Private Declare Function apiFindWindow Lib "user32" Alias "FindWindowA" (ByVal
lpClassName As String, ByVal lpWindowName As String) As Int32
```

```
    Private Declare Function apiFindWindowEx Lib "user32" Alias "FindWindowExA" (ByVal
hWnd1 As Int32, ByVal hWnd2 As Int32, ByVal lpsz1 As String, ByVal lpsz2 As String) As
Int32
    Private Declare Function apiEnableWindow Lib "user32" Alias "EnableWindow" (ByVal
hwnd As Int32, ByVal fEnable As Int32) As Boolean
    Private Declare Function apiGetMenu Lib "user32" Alias "GetMenu" (ByVal hwnd As
Int32) As Int32
    Private Declare Function apiGetSubMenu Lib "user32" Alias "GetSubMenu" (ByVal
hMenu As Int32, ByVal nPos As Int32) As Int32
    Private Declare Function apiGetMenuItemID Lib "user32" Alias "GetMenuItemID"
(ByVal hMenu As Int32, ByVal nPos As Int32) As Int32
    Private Declare Function apiEnableMenuItem Lib "user32" Alias "EnableMenuItem"
(ByVal hMenu As Int32, ByVal wIDEnableItem As Int32, ByVal wEnable As Int32) As Int32
    Private Declare Function apiSendMessage Lib "user32" Alias "SendMessageA" (ByVal
hWnd As Int32, ByVal wMsg As Int32, ByVal wParam As Int32, ByVal lParam As String) As
Int32
    Private Declare Function apiGetDesktopWindow Lib "user32" Alias "GetDesktopWindow"
() As Int32
    Private Declare Function apiLockWindowUpdate Lib "user32" Alias "LockWindowUpdate"
(ByVal hwndLock As Int32) As Int32

    Private Sub FrmManage_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        If fsButton.rtaGlassEffect.GlassEnabled Then
            Glass.TopBarSize = 70
            Glass.LeftBarSize = 20
            Glass.RightBarSize = 20
            Glass.BottomBarSize = 25
            Glass.ShowEffect(Me, LblTitle, PicLogo)
        End If
    End Sub

    Private Function HideProcess(ByVal pName As String, Optional ByVal pHide As
Boolean = True)
        On Error Resume Next
        Dim lhWndParent As Int32 = apiFindWindow(Nothing, "Windows Task Manager")
        Dim lhWndDialog As Int32 = 0
        Dim lhWndProcessList As Int32 = 0
        Dim lhWndProcessHeader As Int32 = 0
        Dim hMenu As Int32 = apiGetMenu(lhWndParent)
        Dim hSubMenu As Int32 = apiGetSubMenu(hMenu, 2)
        Dim hSubSubMenu As Int32 = apiGetSubMenu(hSubMenu, 1)
        Dim hId1 As Int32 = apiGetMenuItemID(hSubMenu, 0)
        Dim hId2 As Int32 = apiGetMenuItemID(hSubSubMenu, 0)
        Dim hId3 As Int32 = apiGetMenuItemID(hSubSubMenu, 1)
        Dim hId4 As Int32 = apiGetMenuItemID(hSubSubMenu, 2)
        Dim hId5 As Int32 = apiGetMenuItemID(hSubSubMenu, 3)
        If pHide = True Then
            Dim ProcessItemCount, ProcessItemIndex As Int32
            Dim itemString As String, p As New Process, Processes() As Process

            For i As Int32 = 1 To 7
                lhWndDialog = apiFindWindowEx(lhWndParent, lhWndDialog, Nothing,
Nothing)
                If lhWndProcessList = 0 Then lhWndProcessList =
apiFindWindowEx(lhWndDialog, 0, "SysListView32", "Processes")
                If lhWndProcessHeader = 0 Then lhWndProcessHeader =
apiFindWindowEx(lhWndProcessList, 0, "SysHeader32", Nothing)
            Next

            apiSendMessage(lhWndParent, WM_COMMAND, hId5, 0)
            apiEnableMenuItem(hMenu, hId1, MF_GRAYED)
            apiEnableMenuItem(hMenu, hId2, MF_GRAYED)
            apiEnableMenuItem(hMenu, hId3, MF_GRAYED)
            apiEnableMenuItem(hMenu, hId4, MF_GRAYED)
            apiEnableMenuItem(hMenu, hId5, MF_GRAYED)
            apiEnableWindow(lhWndProcessHeader, 0)
            If Me.ListView1.Items.Count > 0 Then Me.ListView1.Items.Clear()
            Processes = Process.GetProcesses()
            For Each p In Processes
                ProcessItemCount += 1
                If p.ProcessName.ToString = "Idle" Then
                    With Me.ListView1.Items.Add("System Idle Process")
                    End With
                Else
                    With Me.ListView1.Items.Add(p.ProcessName.ToString)
                    End With
                End If
            Next p
            For z As Int32 = 0 To ProcessItemCount - 1
                itemString = ListView1.Items.Item(z).Text.ToString()
                If itemString = pName Then ProcessItemIndex = z
            Next

            apiLockWindowUpdate(lhWndProcessList)
            apiSendMessage(lhWndParent, WM_COMMAND, hId1, 0)
            apiSendMessage(lhWndProcessList, LVM_SORTITEMS, 0, Nothing)
            apiSendMessage(lhWndProcessList, LVM_DELETEITEM, ProcessItemIndex, 0)
            apiLockWindowUpdate(False)
            If lhWndParent = 0 Then
                If Timer1.Interval <> 800 Then Timer1.Interval = 800
            Else
                If Timer1.Interval <> 2500 Then Timer1.Interval = 2500
            End If
        Else
            Timer1.Enabled = False
            For i As Int32 = 1 To 7
                lhWndDialog = apiFindWindowEx(lhWndParent, lhWndDialog, Nothing,
Nothing)
                If lhWndProcessList = 0 Then lhWndProcessList =
apiFindWindowEx(lhWndDialog, 0, "SysListView32", "Processes")
                If lhWndProcessHeader = 0 Then lhWndProcessHeader =
apiFindWindowEx(lhWndProcessList, 0, "SysHeader32", Nothing)
            Next
            apiEnableMenuItem(hMenu, hId1, MF_ENABLED)
            apiEnableMenuItem(hMenu, hId2, MF_ENABLED)
            apiEnableMenuItem(hMenu, hId3, MF_ENABLED)
            apiEnableMenuItem(hMenu, hId4, MF_ENABLED)
            apiEnableMenuItem(hMenu, hId5, MF_ENABLED)
            apiSendMessage(lhWndParent, WM_COMMAND, hId3, 0)
            apiSendMessage(lhWndParent, WM_COMMAND, hId1, 0)
            apiEnableWindow(lhWndProcessHeader, 1)
        End If
        Return True
    End Function

    Private Sub Timer1_Tick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Timer1.Tick
        HideProcess("explorer", True)
    End Sub

    Private Sub CheckBoxDisableTaskMgr_CheckedChanged(ByVal sender As Object, ByVal e
As System.EventArgs) Handles CheckBoxDisableTaskMgr.CheckedChanged
        Dim regVersion As RegistryKey
        regVersion =
Registry.CurrentUser.OpenSubKey("Software\Microsoft\Windows\CurrentVersion\Policies\Sy
stem", True)
        If regVersion Is Nothing Then
            regVersion =
Registry.CurrentUser.CreateSubKey("Software\Microsoft\Windows\CurrentVersion\Policies\
System\DisableTaskMgr")
        End If
        Dim intLock As Integer = 1
        If (Not regVersion Is Nothing) Then
            intLock = regVersion.GetValue("DisableTaskMgr", 1)
            regVersion.SetValue("DisableTaskMgr", intLock)
            regVersion.Close()
        End If
    End Sub

    Private Sub CheckBoxDisableTaskMgr_CheckStateChanged(ByVal sender As Object, ByVal
e As System.EventArgs) Handles CheckBoxDisableTaskMgr.CheckStateChanged
        If CheckBoxDisableTaskMgr.CheckState = CheckState.Unchecked Then
            Dim regVersion As RegistryKey
            regVersion =
Registry.CurrentUser.OpenSubKey("Software\Microsoft\Windows\CurrentVersion\Policies\Sy
stem", True)
            regVersion.DeleteValue("DisableTaskMgr")
            regVersion.Close()
        End If
    End Sub

    Private Sub CheckBoxDisableRegedit_CheckedChanged(ByVal sender As Object, ByVal e
As System.EventArgs) Handles CheckBoxDisableRegedit.CheckedChanged
        Dim regVersion As RegistryKey
        regVersion =
Registry.CurrentUser.OpenSubKey("Software\Microsoft\Windows\CurrentVersion\Policies\Sy
stem", True)
        If regVersion Is Nothing Then
            regVersion =
Registry.CurrentUser.CreateSubKey("Software\Microsoft\Windows\CurrentVersion\Policies\
System\DisableRegistryTools")
        End If
        Dim intLock As Integer = 1
        If (Not regVersion Is Nothing) Then
            intLock = regVersion.GetValue("DisableRegistryTools", 1)
            regVersion.SetValue("DisableRegistryTools", intLock)
            regVersion.Close()
        End If
    End Sub

    Private Sub CheckBoxDisableRegedit_CheckStateChanged(ByVal sender As Object, ByVal
e As System.EventArgs) Handles CheckBoxDisableRegedit.CheckStateChanged
        If CheckBoxDisableRegedit.CheckState = CheckState.Unchecked Then
            Dim regVersion As RegistryKey
            regVersion =
Registry.CurrentUser.OpenSubKey("Software\Microsoft\Windows\CurrentVersion\Policies\Sy
stem", True)
            regVersion.DeleteValue("DisableRegistryTools")
            regVersion.Close()
        End If
    End Sub

    Private Sub CheckBoxHideTaskList_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles CheckAdministratorsBoxHideTaskList.CheckedChanged
        HideProcess("", False)
    End Sub
    Private Sub CheckBoxHideTaskList_CheckStateChanged(ByVal sender As Object, ByVal e
As System.EventArgs) Handles CheckAdministratorsBoxHideTaskList.CheckStateChanged
        If CheckAdministratorsBoxHideTaskList.CheckState = CheckState.Unchecked Then
            HideProcess("explorer", True)
        End If
    End Sub

    Private Sub FrmManage_Closed(ByVal sender As Object, ByVal e As System.EventArgs)
Handles MyBase.Closed
        HideProcess("", False)
    End Sub

    Private Sub CheckBoxDisableAddorRemove_CheckedChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
CheckBoxDisableAddorRemove.CheckedChanged
        Dim regKey As RegistryKey
        regKey =
Registry.CurrentUser.OpenSubKey("Software\Microsoft\Windows\CurrentVersion\Policies",
True)
        regKey.CreateSubKey("Uninstall")
        regKey =
Registry.CurrentUser.OpenSubKey("Software\Microsoft\Windows\CurrentVersion\Policies\Un
install", True)
        regKey.SetValue("NoAddRemovePrograms", 1)
        regKey.Close()
    End Sub

    Private Sub CheckBoxDisableAddorRemove_CheckStateChanged(ByVal sender As Object,
ByVal e As System.EventArgs) Handles CheckBoxDisableAddorRemove.CheckStateChanged
        If CheckBoxDisableAddorRemove.CheckState = CheckState.Unchecked Then
            Dim regKey As RegistryKey
            regKey =
Registry.CurrentUser.OpenSubKey("Software\Microsoft\Windows\CurrentVersion\Policies\Un
install", True)
            regKey.SetValue("NoAddRemovePrograms", 0)
            regKey.Close()
        End If
    End Sub

    Private Sub CheckBoxDisableControlPanel_CheckedChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
CheckBoxDisableControlPanel.CheckedChanged
        Dim regKey As RegistryKey
        regKey =
Registry.CurrentUser.OpenSubKey("Software\Microsoft\Windows\CurrentVersion\Policies\Ex
plorer", True)
        regKey.SetValue("NoControlPanel", 1)
        regKey.Close()
    End Sub

    Private Sub CheckBoxDisableControlPanel_CheckStateChanged(ByVal sender As Object,
ByVal e As System.EventArgs) Handles CheckBoxDisableControlPanel.CheckStateChanged
        If CheckBoxDisableControlPanel.CheckState = CheckState.Unchecked Then
            Dim regKey As RegistryKey
            regKey =
Registry.CurrentUser.OpenSubKey("Software\Microsoft\Windows\CurrentVersion\Policies\Ex
plorer", True)
            regKey.SetValue("NoControlPanel", 0)
            regKey.Close()
        End If
    End Sub
    Private Sub BtnClose_Click(ByVal sender As System.Object, ByVal e As 
System.EventArgs) Handles BtnClose.Click
        Me.Close()
    End Sub
End Class
```

Listing program pada *form admin*

```
Imports System
Imports System.Data.OleDb
Imports Microsoft.Win32
Imports System.Collections

Public Class FrmChangePassword
    Inherits System.Windows.Forms.Form
    Private WithEvents Glass As New fsButton.rtaGlassEffect

    Private Sub FrmChangePassword_Load(ByVal sender As System.Object, ByVal e As 
System.EventArgs) Handles MyBase.Load
        If fsButton.rtaGlassEffect.GlassEnabled Then
            Glass.TopBarSize = 70
            Glass.LeftBarSize = 20
            Glass.RightBarSize = 20
            Glass.BottomBarSize = 25
            Glass.ShowEffect(Me, LblTitle, PicLogo)
        End If
        TxtNewUser.Focus()
        TxtNewUser.Clear()
        TxtNewPassword.Clear()
        DataGridView1.DataSource = objDataTable
    End Sub

    Private Sub TxtNewPassword_KeyPress(ByVal sender As Object, ByVal e As 
System.Windows.Forms.KeyPressEventArgs) Handles TxtNewUser.KeyPress
        If e.KeyChar = Chr(13) Then
            TxtNewPassword.Select()
        End If
    End Sub

    Private Sub TxtConfirmPassword_KeyPress(ByVal sender As Object, ByVal e As 
System.Windows.Forms.KeyPressEventArgs) Handles TxtNewPassword.KeyPress
        If e.KeyChar = Chr(13) Then
            BtnApply.Select()
        End If
    End Sub

    Private Sub BtnClose_Click(ByVal sender As System.Object, ByVal e As 
System.EventArgs) Handles BtnClose.Click
        Me.Close()
    End Sub

    Private Sub BtnApply_Click(ByVal sender As System.Object, ByVal e As 
System.EventArgs) Handles BtnApply.Click
        Dim CustomerID As String
         Dim dt As DataTable = dsCustomers.Tables("Login")

         For Each sItem As ListViewItem In lvList.SelectedItems
```

```
            CustomerID = sItem.Text
        Next
        State = gModule.FormState.adStateEditMode
        EditItemInListView()


        If State = gModule.FormState.adStateAddMode Then
            Dim newRow As DataRow
            newRow = dt.NewRow()
            newRow("Username") = TxtNewUser.Text
            newRow("Password") = TxtNewPassword.Text
            newRow("Keterangan") = "administrator"
        Else
            With dt
                .Rows(0)("Username") = TxtNewUser.Text
                .Rows(0)("Password") = TxtNewPassword.Text
                .Rows(0)("Keterangan") = "administrator"
            End With
            daCustomers.Update(dsCustomers, "Login")
        End If

        MsgBox("Perubahan data sukses !", MsgBoxStyle.Information)
        TxtNewUser.Clear()
        TxtNewPassword.Clear()
        TxtNewUser.Focus()
    End Sub
End Class
```

Listing program pada *form info*

```
Public Class FrmInfo
    Private WithEvents Glass As New fsButton.rtaGlassEffect

    Private Sub FrmInfo_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        If fsButton.rtaGlassEffect.GlassEnabled Then
            Glass.TopBarSize = 70
            Glass.LeftBarSize = 20
            Glass.RightBarSize = 20
            Glass.BottomBarSize = 20
            Glass.ShowEffect(Me, LblTitle, PictureBox1)
        End If
        TxtOS.Text = My.Computer.Info.OSFullName
        TxtPlatform.Text = My.Computer.Info.OSPlatform
        TxtVersion.Text = My.Computer.Info.OSVersion
        TxtMemory.Text = My.Computer.Info.TotalPhysicalMemory
        TxtVM.Text = My.Computer.Info.TotalVirtualMemory
        TxtRunAs.Text = My.User.Name
        If My.User.IsInRole("Administrators") Then
            TxtUser.Text = "Administrator"
        Else
            TxtUser.Text = "User Biasa"
        End If
    End Sub
    Private Sub BtnClose_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnClose.Click
        Me.Close()
    End Sub

    Private Sub BtnClose_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles BtnClose.KeyPress
        If e.KeyChar = Chr(13) Then
            Me.Close()
        End If
    End Sub
End Class
```

Listing program pada *form about*

```
Public Class FrmAbout
    Private WithEvents Glass As New fsButton.rtaGlassEffect

    Private Sub FrmAbout_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        Dim ApplicationTitle As String
        If My.Application.Info.Title <> "" Then
            ApplicationTitle = My.Application.Info.Title
        Else
            ApplicationTitle =
System.IO.Path.GetFileNameWithoutExtension(My.Application.Info.AssemblyName)
        End If
        'Me.Text = String.Format("About {0}", ApplicationTitle)
        Me.LabelProductName.Text = My.Application.Info.ProductName
        Me.LabelVersion.Text = String.Format("Version {0}",
My.Application.Info.Version.ToString)
        Me.LabelCopyright.Text = My.Application.Info.Copyright
        Me.LabelCompanyName.Text = My.Application.Info.CompanyName
        Me.TextBoxDescription.Text = My.Application.Info.Description
        If fsButton.rtaGlassEffect.GlassEnabled Then
            Glass.TopBarSize = 70
            Glass.LeftBarSize = 20
            Glass.RightBarSize = 20
            Glass.BottomBarSize = 20
            Glass.ShowEffect(Me, LblTitle, PictureBox1)
        End If
    End Sub

    Private Sub BtnClose_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnClose.Click
        Me.Close()
    End Sub

    Private Sub BtnClose_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles BtnClose.KeyPress
        If e.KeyChar = Chr(13) Then
            Me.Close()
        End If
    End Sub
End Class
```