

LAMPIRAN

1. Source Code Delphi Aplikasi Sistem Kendali Parkir

a. p_start

```
unit u_start;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics,  
Controls, Forms,
```

```
Dialogs, Menus, ExtCtrls, ComCtrls;
```

```
type
```

```
Tf_start = class(TForm)  
    m_utama: TMainMenu;  
    Timer1: TTimer;  
    m_popup: TPopupMenu;  
    menua: TMenuItem;  
    menuf: TMenuItem;  
    menuc: TMenuItem;  
    menud: TMenuItem;  
    menue: TMenuItem;  
    menuaa: TMenuItem;  
    menuab: TMenuItem;  
    N1: TMenuItem;  
    menuac: TMenuItem;  
    menufa: TMenuItem;  
    menufb: TMenuItem;  
    menuca: TMenuItem;  
    menucb: TMenuItem;  
    N2: TMenuItem;  
    menuda: TMenuItem;  
    menudb: TMenuItem;  
    menuea: TMenuItem;  
    N3: TMenuItem;  
    menueb: TMenuItem;  
    menub: TMenuItem;  
    menuba: TMenuItem;  
    menubb: TMenuItem;  
    menuec: TMenuItem;
```

```

menued: TMenuItem;
N4: TMenuItem;
menuef: TMenuItem;
Login2: TMenuItem;
Logout2: TMenuItem;
N5: TMenuItem;
Exit2: TMenuItem;
menueg: TMenuItem;
menuee: TMenuItem;
StatusBar1: TStatusBar;
menudc: TMenuItem;
menucc: TMenuItem;
Image1: TImage;
procedure Timer1Timer(Sender: TObject);
procedure FormShow(Sender: TObject);
procedure FormCloseQuery(Sender: TObject; var CanClose:
Boolean);
procedure FormCreate(Sender: TObject);
procedure menuacClick(Sender: TObject);
procedure Exit2Click(Sender: TObject);
procedure menuaaClick(Sender: TObject);
procedure menuabClick(Sender: TObject);
procedure JalankanProgram(Sender: TObject);
procedure menueaClick(Sender: TObject);
procedure menuebClick(Sender: TObject);
procedure menuecClick(Sender: TObject);
procedure menuedClick(Sender: TObject);
procedure menuefClick(Sender: TObject);
procedure menucaClick(Sender: TObject);
procedure menucbClick(Sender: TObject);
procedure menuccClick(Sender: TObject);
procedure menubaClick(Sender: TObject);
procedure menubbClick(Sender: TObject);
procedure menuegClick(Sender: TObject);
procedure menufaClick(Sender: TObject);
procedure menueeClick(Sender: TObject);
procedure menudaClick(Sender: TObject);
procedure menudbClick(Sender: TObject);
procedure menudcClick(Sender: TObject);
private
{ Private declarations }
t1,t2:longint; // t1&t2 : timer message

```

```

FDefProc : Pointer;
OnSynchronize:boolean;
DefaultLogoName,DefaultCaption,
DefaultVersi,WithBanner : string;
Procedure ClientWall(var M:TMessage);
Procedure EnableOtherButton(ok:boolean);
procedure HistoryUser(catatan: string);
procedure DefaultMenu(UserId: string);
//procedure DefaultMenu(FlagId: string);
function CariJendelaMDI(param:string;myform:tform):boolean;
procedure TampilkanForm(Sender: TObject ;fpara:tform);
procedure pilihform(Sender: TObject);
public
  { Public declarations }
  DefaultProgramName,DefaultServerName,DbPassword : String;
  DefaultAddrName1,DefaultAddrName2,DefaultAddrName3  :
String;
  procedure AfterLogin;
end;

var
  f_start: Tf_start;

implementation

uses u_publik, u_dmutama, u_blacklist, u_blacksearch, u_login,
u_member,
  u_parkin, u_parkout, u_parkpos, u_progress, u_reason,
u_registrasi,
  u_regsearch, u_setup, u_user, DB, u_md1prg, u_about, u_jenis,
u_rptglb,
  u_parkappr, u_rptadm, u_realtime, U_rptadmdetil,
U_rptadmglobal;

{$R *.dfm}

procedure Tf_start.Timer1Timer(Sender: TObject);
//package
//var
// txt, : string;
begin
  DefaultProgramName := "";

```

```

    if WithBanner=" then begin
        t1:=gettickcount();
        Timer1.Interval:=1000;
        StatusBar1.Panels[0].Text:=FormatDateTime('DD-MM-
YYYY HH:MM:SS',now());
// package
// txt:=DefaultCaption+' on '+formatdatetime('dd-mmm-yyyy
<hh:mm:ss am/pm>',now);
    end else begin
        t2:=gettickcount();
        Timer1.Interval:=200;
// package
// txt='Message: '"+WithBanner+"'";
DefaultProgramName := 'Message: '"+WithBanner+"'";
if t2-t1>15000 then begin
    if length(WithBanner)>0 then
        WithBanner:=copy(WithBanner,2,length(WithBanner))
    else
        WithBanner:="";
    end else
        if strtoint(formatdatetime('ss',now)) mod 5=0 then
f_start.Caption:="";
        end;
// package
// if txt="" then txt:=bacasetup('OWNER')+
'+bacasetup('VERSI');
// DefaultProgramName := txt;
// f_start.Caption:=txt;

    If DefaultProgramName = " Then DefaultProgramName :=
bacasetup('OWNER')+ ' +bacasetup('VERSI');
    If DefaultAddrName1 = " Then DefaultAddrName1 :=
bacasetup('BRANCH_ADDR1');
    If DefaultAddrName2 = " Then DefaultAddrName2 :=
bacasetup('BRANCH_ADDR2');
    If DefaultAddrName3 = " Then DefaultAddrName3 :=
bacasetup('BRANCH_ADDR3');
    f_start.Caption := DefaultProgramName;
end;

procedure Tf_start.FormShow(Sender: TObject);
var procwall:tfarproc;

```

```

begin
    if FileExists(DefaultLogoName) then begin
        procwall:=MakeObjectInstance(ClientWall);

FDefProc:=Pointer(SetwindowLong(ClientHandle,GWL_WNDP
ROC,longint(procwall)));
        end;
        dm_utama.UserId:="";
        dm_utama.FlagId:="";
        Timer1.Enabled:=true;
        t1:=gettickcount();
    end;

    procedure Tf_start.FormCloseQuery(Sender: TObject; var
CanClose: Boolean);
    begin
        if (f_start.FormStyle=fsMDIForm) and
(f_start.MDIChildCount>0) then CanClose:=false
        else if OnSynchronize then begin
            showmessage("Tunggu sesaat, Tidak bisa Logout/Exit karena
masih dalam proses synchronize!");
            CanClose:=false;
        end else begin
            //if dm_utama.UserId<>" then HistoryUser('Exit');
            CanClose:=true;
        end;
    end;

    procedure Tf_start.FormCreate(Sender: TObject);
    begin
        if FindComponent('dm_utama')=Nil then
dm_utama:=Tdm_utama.Create(application);
        AksesRegistry(HKEY_CURRENT_USER,'Software\Park
Soft\Entreprise','Versi',DefaultVersi,false);
        DbPassword:='mambu';
        DefaultServerName:="";
        DefaultLogoName:='c:\cps\logo.bmp';
        DefaultCaption:=DefaultProgramName+DefaultVersi;
    end;

    procedure Tf_start.ClientWall(var M: TMessage);
    var bmp:TBitmap;

```

```

begin
  if (M.Msg=WM_ERASEBKGD) then begin
    bmp:=TBitmap.Create;
    try
      bmp.Loadfromfile(DefaultLogoName);

stretchBlt(M.wParam,0,0,Width,Height,bmp.Canvas.Handle,
            0,0,bmp.Width,bmp.Height,SRCCOPY);
    finally
      bmp.Free;
    end
  end else

M.Result:=CallWindowProc(FDefProc,ClientHandle,M.Msg,M.w
Param,M.lParam);
end;

procedure Tf_start.EnableOtherButton(ok: boolean);
begin
  if dm_utama.UserLevel<>'E' then begin
    //ChangeServer.Enabled:=ok;
    //ChangeServer.Visible:=ok;
  end;
  //ChangePassword.Enabled:=ok; ChangePassword.Visible:=ok;
  //logout.Enabled:=ok;
end;

procedure Tf_start.HistoryUser(catatan: string);
begin
  with dm_utama.MyQuery do begin
    sql.clear;
    sql.add(format('INSERT                                T_USERTRX
VALUES("%s",NOW(),"%s")',[dm_utama.userid,catatan]));
    Application.ProcessMessages;
    try execsql;
    except end;
  end;
end;

procedure Tf_start.AfterLogin;
var txt:string;
begin

```

```

DefaultCaption:="";
//      login.Caption:='Login  <<'+DefaultServerName+'  :
'+dm_retail.Branch+'>>';
// package

if      lowercase(DefaultServerName)='localhost'      then
DefaultCaption:='<<localhost>>';

DefaultCaption:=DefaultCaption+DefaultProgramName+Default
Versi+'/'+dm_utama.UserId+'/'+dm_utama.UserLevel;
//      WithBanner:='Selamat Datang '+dm_retail.UserId+' di
'+DefaultProgramName+DefaultVersi;
// package
      WithBanner := "";

//HistoryUser('Logon');
txt:=copy(dm_utama.Branch+'
',1,6)+copy(dm_utama.UserId+stringofchar('
',10),1,10)+dm_utama.UserLevel+dm_utama.FlagId+dm_utama.P
os;

AksesRegistry(HKEY_CURRENT_USER,"Software\Microsoft\
Windows\CurrentVersion\RunOnce",'MyPark',txt,true);
// untuk proses sinkron data stock
end;

procedure Tf_start.menuacClick(Sender: TObject);
begin
      close;
end;

procedure Tf_start.Exit2Click(Sender: TObject);
begin
      close;
end;

procedure Tf_start.DefaultMenu(UserId: string);
//procedure Tf_start.DefaultMenu(FlagId: string);
var txt:string;
      i:integer;

```

```

begin
  with dm_utama.MyQuery do
  begin
    sql.Clear;
    {if FlagId='N' then
      sql.add('select * from t_mdprg where fc_flag="Y" ')
    else begin
      sql.add('select * from t_mdprg');
    end;}
    if UserId<>" then
    begin
      sql.add('select a.*,b.* from t_usermdl a,t_mdprg b');
      sql.add('where      a.fc_mdprg=b.fc_mdprg      and
a.fc_userid=:fc_userid');
      parambyname('fc_userid').AsString:=dm_utama.UserId;
      open;
      if isempty then
        showmessage('Anda belum diberi hak akses')
      else
        begin
          while not eof do
          begin

txt:=getspace('MENU'+fieldname('fc_mdprg').asString,10)+

getspace(fieldbyname('FC_NAMAPRG').asString,15)+
      fieldbyname('FC_DESCRIPTION').asString;
      try Item_menu(m_utama,txt,userId<>");
      except
      end;
      next;
    end;
  end;
end else
begin
  for i:=0 to m_utama.Items.Count-1 do
  begin
    if (copy(m_utama.Items[i].Name,5,1)='a')or
      (copy(m_utama.Items[i].Name,5,1)='f') then
      m_utama.Items[i].Visible:=true
    else m_utama.Items[i].Visible:=false;
  end;

```



```

    end;
  end;
end;

```

```

procedure Tf_start.menuaaClick(Sender: TObject);
begin
  if (FormStyle=fsMDIForm) and (MDIChildCount>0) then exit;
  if dm_utama.UserId<>" then menuab.Click;
  if dm_utama.UserId<>" then exit;
  f_login:=Tf_login.create(application);
  f_login.ShowModal;
  f_login.Free;
  if dm_utama.UserId="" then exit;
  DefaultMenu(dm_utama.UserId);
  menuab.Enabled:=true;{Enable Logout}
  AfterLogin;
end;

```

```

procedure Tf_start.menuabClick(Sender: TObject);
begin
  if (FormStyle=fsMDIForm) and (MDIChildCount>0) then exit;
  DefaultMenu("");
  dm_utama.MyData.Connected:=false;
  DefaultCaption:=DefaultProgramName+DefaultVersi;
  menuab.Enabled:=false;{Disable Logout}
end;

```

```

procedure Tf_start.JalankanProgram(Sender: TObject);
var txt:string;
begin
  txt:=TmenuItem(Sender).Hint;
  if copy(txt,1,4)='Menu' then
  begin
    TampilForm(Sender,f_start,'T'+copy(txt,11,15),false);
  end else showmessage('Menu tersebut tidak dapat diakses');
end;

```

```

procedure Tf_start.menueaClick(Sender: TObject);
var txt:string;
begin
  txt:='f_user';

```

```

    if not CariJendelaMDI(txt,f_user) then
    begin
        f_user:=Tf_user.Create(application);
        TampilkanForm(Sender,f_user);
    end;
end;

procedure Tf_start.pilihform(Sender: TObject);
var txt,nama,mdl:string;
    kd:Tform;
begin
    mdl:=TMenuItem(Sender).Hint;
    nama:=copy(mdl,5,5);
    with dm_utama.MyQuery do
    begin
        sql.clear;
        sql.add('Select fc_namaprg from t_mdprg where
fc_mdprg=:fc_mdprg ');
        parambyname('fc_mdprg').AsString:=nama;
        open;
    end;
    txt:=dm_utama.MyQuery.fieldbyname('fc_namaprg').AsString;
    kd:=TForm.Create(Application);
    kd.Name:=txt;
    //txt:='f_user';
    if not CariJendelaMDI(txt,kd) then
    begin
        f_REASON:=Tf_REASON.Create(application);
        TampilkanForm(Sender,kd);
    end;
    kd.Free;
end;

procedure Tf_start.TampilkanForm(Sender: TObject ;fpara:tform);
var title,txt:string;
begin
    //IsiModulPilih(TMenuItem(sender).name);
    title:=TMenuItem(Sender).Hint;
    txt:=copy(Title,15+10,length(title));
    if FormStyle=fsMDIForm then
        with fpara do
            begin

```

```

Caption:=txt;
BorderStyle:=bsNone;
Hint:=trim(copy(title,1,9));
visible:=false;
AutoSize:=true;
FormStyle:=fsMDIChild;
Show;
end
else
with fpara do
begin
Caption:="";
BorderStyle:=bsNone;
visible:=false;
AutoSize:=false;
FormStyle:=fsNormal;
Hint:=trim(copy(title,1,9));
Position:=poScreenCenter;
ShowModal;
end
end;

function
Tf_start.CariJendelaMDI(param:string;myform:tform):boolean;
var i,jumlah:integer;
t:pchar;
FormClassName : string;
memStat : TMemoryStatus;
begin
FormClassName:=param;
param:='T'+param;
result:=false;
if FormStyle=fsMDIForm then begin
jumlah:=MDIChildCount - 1;
for i:=jumlah downto 0 do
if MDIChildren[i].ClassName=param then begin
MDIChildren[i].Show;
result:=true;
end;
end else begin
t:=pchar(param);
if findwindow(t,pchar(copy(t,2,length(t))))<>>0 then begin

```

```

        result:=true;
        myform.setfocus;
    end;
end;
end;

procedure Tf_start.menuebClick(Sender: TObject);
var txt:string;
begin
    txt:='f_reason';
    if not CariJendelaMDI(txt,f_reason) then
        begin
            f_reason:=Tf_reason.Create(application);
            TampilkanForm(Sender,f_reason);
        end;
    end;
end;

procedure Tf_start.menuecClick(Sender: TObject);
var txt:string;
begin
    txt:='f_parkpos';
    if not CariJendelaMDI(txt,f_parkpos) then
        begin
            f_parkpos:=Tf_parkpos.Create(application);
            TampilkanForm(Sender,f_parkpos);
        end;
    end;
end;

procedure Tf_start.menuedClick(Sender: TObject);
var txt:string;
begin
    txt:='f_registrasi';
    if not CariJendelaMDI(txt,f_registrasi) then
        begin
            f_registrasi:=Tf_registrasi.Create(application);
            TampilkanForm(Sender,f_registrasi);
        end;
    end;
end;

procedure Tf_start.menuefClick(Sender: TObject);
var txt:string;
begin

```

```

txt:='f_setup';
if not CariJendelaMDI(txt,f_setup) then
begin
    f_setup:=Tf_setup.Create(application);
    TampilkanForm(Sender,f_setup);
end;
end;

procedure Tf_start.menucaClick(Sender: TObject);
var txt:string;
begin
    txt:='f_parkin';
    if not CariJendelaMDI(txt,f_parkin) then
    begin
        f_parkin:=Tf_parkin.Create(application);
        TampilkanForm(Sender,f_parkin);
    end;
end;

procedure Tf_start.menucbClick(Sender: TObject);
var txt:string;
begin
    txt:='f_parkout';
    if not CariJendelaMDI(txt,f_parkout) then
    begin
        f_parkout:=Tf_parkout.Create(application);
        TampilkanForm(Sender,f_parkout);
    end;
end;

procedure Tf_start.menuccClick(Sender: TObject);
var txt:string;
begin
    txt:='f_parkappr';
    if not CariJendelaMDI(txt,f_parkappr) then
    begin
        f_parkappr:=Tf_parkappr.Create(application);
        TampilkanForm(Sender,f_parkappr);
    end;
end;

procedure Tf_start.menubaClick(Sender: TObject);

```

```

var txt:string;
begin
  txt:='f_member';
  if not CariJendelaMDI(txt,f_member) then
  begin
    f_member:=Tf_member.Create(application);
    TampilkanForm(Sender,f_member);
  end;
end;

```

```

procedure Tf_start.menubbClick(Sender: TObject);
var txt:string;
begin
  txt:='f_blacklist';
  if not CariJendelaMDI(txt,f_blacklist) then
  begin
    f_blacklist:=Tf_blacklist.Create(application);
    TampilkanForm(Sender,f_blacklist);
  end;
end;

```

```

procedure Tf_start.menuegClick(Sender: TObject);
var txt:string;
begin
  txt:='f_mdprg';
  if not CariJendelaMDI(txt,f_mdprg) then
  begin
    f_mdprg:=Tf_mdprg.Create(application);
    TampilkanForm(Sender,f_mdprg);
  end;
end;

```

```

procedure Tf_start.menufaClick(Sender: TObject);
begin
  if FindComponent('f_about')=Nil then
  f_about:=Tf_about.create(application);
  f_about.ShowModal;
  f_about.Free;
end;

```

```

procedure Tf_start.menueeClick(Sender: TObject);
var txt:string;

```

```

begin
    txt:='f_jenis';
    if not CariJendelaMDI(txt,f_jenis) then
        begin
            f_jenis:=Tf_jenis.Create(application);
            TampilkanForm(Sender,f_jenis);
        end;
    end;

procedure Tf_start.menudaClick(Sender: TObject);
var txt:string;
begin
    txt:='f_rptglb';
    if not CariJendelaMDI(txt,f_rptglb) then
        begin
            f_rptglb:=Tf_rptglb.Create(application);
            TampilkanForm(Sender,f_rptglb);
        end;
    end;

procedure Tf_start.menudbClick(Sender: TObject);
var txt:string;
begin
    txt:='f_rptadmin';
    if not CariJendelaMDI(txt,f_rptadmin) then
        begin
            f_rptadmin:=Tf_rptadmin.Create(application);
            TampilkanForm(Sender,f_rptadmin);
        end;
    end;

procedure Tf_start.menudcClick(Sender: TObject);
var txt:string;
begin
    txt:='f_realtime';
    if not CariJendelaMDI(txt,f_realtime) then
        begin
            f_realtime:=Tf_realtime.Create(application);
            TampilkanForm(Sender,f_realtime);
        end;
    end;
end;

```

end.

b. p_user

```
unit u_user;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics,  
Controls, Forms,
```

```
Dialogs, StdCtrls, DBCtrls, Mask, Buttons, Grids, DBGrids,  
ExtCtrls, DB,
```

```
ZQuery, ZMySqlQuery;
```

```
type
```

```
Tf_user = class(TForm)
```

```
  Notebook1: TNotebook;
```

```
  Panel1: TPanel;
```

```
  DBGrid1: TDBGrid;
```

```
  Panel3: TPanel;
```

```
  Panel4: TPanel;
```

```
  Panel5: TPanel;
```

```
  Panel6: TPanel;
```

```
  a1: TBitBtn;
```

```
  a2: TBitBtn;
```

```
  a3: TBitBtn;
```

```
  a6: TBitBtn;
```

```
  a7: TBitBtn;
```

```
  Panel7: TPanel;
```

```
  Label1: TLabel;
```

```
  Label2: TLabel;
```

```
  Label3: TLabel;
```

```
  Label4: TLabel;
```

```
  Label5: TLabel;
```

```
  Label6: TLabel;
```

```
  aa2: TDBEdit;
```

```
  aa3: TDBEdit;
```

```
  aa1: TDBEdit;
```

```
  a4: TBitBtn;
```

```
  a5: TBitBtn;
```

```
  aa5: TDBComboBox;
```

```
  aa6: TDBComboBox;
```


DBNavigator2: TDBNavigator;
Panel8: TPanel;
GroupBox1: TGroupBox;
b2: TBitBtn;
b3: TEdit;
b4: TEdit;
rb1: TRadioButton;
rb2: TRadioButton;
DBNavigator1: TDBNavigator;
b1: TBitBtn;
t_user: TZMySqlQuery;
t_userfc_userid: TStringField;
t_userfv_username: TStringField;
t_userfc_flagadm: TStringField;
t_userfc_flagfo: TStringField;
ds_user: TDataSource;
t_userfc_password: TStringField;
aa4: TEdit;
t_userfc_hold: TStringField;
Label7: TLabel;
aa7: TDBComboBox;
Panel2: TPanel;
Panel9: TPanel;
Panel10: TPanel;
Panel11: TPanel;
Panel12: TPanel;
ha1: TBitBtn;
ha2: TBitBtn;
ha3: TBitBtn;
ha4: TBitBtn;
BitBtn5: TBitBtn;
DBGrid2: TDBGrid;
Label8: TLabel;
DBNavigator3: TDBNavigator;
Label9: TLabel;
DBGrid3: TDBGrid;
DBNavigator4: TDBNavigator;
t_usermdl: TZMySqlQuery;
ds_usermdl: TDataSource;
v_mdprg: TZMySqlQuery;
t_mdprg: TZMySqlQuery;
ds_mdprg: TDataSource;

```

t_usermdlfc_userid: TStringField;
t_usermdlfc_mdprg: TStringField;
t_mdprgfc_mdprg: TStringField;
t_mdprgfc_description: TStringField;
v_mdprgfc_mdprg: TStringField;
v_mdprgfc_description: TStringField;
t_usermdlnamamdl: TStringField;
BitBtn6: TBitBtn;
procedure a7Click(Sender: TObject);
procedure b1Click(Sender: TObject);
procedure a6Click(Sender: TObject);
procedure a1MouseMove(Sender: TObject; Shift: TShiftState;
X,
  Y: Integer);
procedure a1Click(Sender: TObject);
procedure FormClose(Sender: TObject; var Action:
TCloseAction);
procedure FormShow(Sender: TObject);
procedure a2Click(Sender: TObject);
procedure a3Click(Sender: TObject);
procedure a5Click(Sender: TObject);
procedure a4Click(Sender: TObject);
procedure rb1Click(Sender: TObject);
procedure b2Click(Sender: TObject);
procedure aa4Exit(Sender: TObject);
procedure t_userAfterScroll(DataSet: TDataSet);
procedure aa1Exit(Sender: TObject);
procedure aa1Enter(Sender: TObject);
procedure aa1KeyPress(Sender: TObject; var Key: Char);
procedure aa6KeyPress(Sender: TObject; var Key: Char);
procedure aa5KeyPress(Sender: TObject; var Key: Char);
procedure b3Enter(Sender: TObject);
procedure b3Exit(Sender: TObject);
procedure b3KeyPress(Sender: TObject; var Key: Char);
procedure aa7KeyPress(Sender: TObject; var Key: Char);
procedure BitBtn5Click(Sender: TObject);
procedure BitBtn6Click(Sender: TObject);
procedure ha1Click(Sender: TObject);
procedure ha4Click(Sender: TObject);
procedure t_usermdlBeforeOpen(DataSet: TDataSet);
procedure ha2Click(Sender: TObject);
procedure ha3Click(Sender: TObject);

```

```

    procedure DBGrid3Db1Click(Sender: TObject);
    procedure DBGrid2Db1Click(Sender: TObject);
    procedure t_md1prgBeforeOpen(DataSet: TDataSet);
private
    { Private declarations }
    Function CekData:boolean;
public
    { Public declarations }
end;

var
    f_user: Tf_user;

implementation

uses u_dmutama, u_publik;

{$R *.dfm}

procedure Tf_user.a7Click(Sender: TObject);
begin
    close;
end;

procedure Tf_user.b1Click(Sender: TObject);
begin
    Notebook1.ActivePage:='page1';
end;

procedure Tf_user.a6Click(Sender: TObject);
begin
    Notebook1.ActivePage:='page2';
    rb1Click(Sender);
end;

procedure Tf_user.a1MouseMove(Sender: TObject; Shift:
TShiftState; X,
    Y: Integer);
begin
    Murup(f_user,sender);
end;

```

```

procedure Tf_user.a1Click(Sender: TObject);
begin
    t_user.Append;
    ControlTag(f_user,[3,13],[2]);
    aa4.Text:="";
    aa1.SetFocus;
end;

```

```

procedure Tf_user.FormClose(Sender: TObject; var Action:
TCloseAction);
begin
    Action:=caFree;
end;

```

```

procedure Tf_user.FormShow(Sender: TObject);
begin
    OpenTable(f_user);
    Notebook1.ActivePage:='page1';
    ControlTag(f_user,[2],[13,3]);
    a1.SetFocus;
end;

```

```

procedure Tf_user.a2Click(Sender: TObject);
begin
    if t_user.IsEmpty then
    begin
        ShowMessage('Data masih kosong !');
        exit;
    end;
    try
        t_user.edit;
    except
    end;
    ControlTag(f_user,[3,13],[2]);
    aa1.Enabled:=false;
    aa4.Text:=t_userfc_password.AsString;
    aa2.SetFocus;
end;

```

```

procedure Tf_user.a3Click(Sender: TObject);
begin
    if t_user.IsEmpty then

```

```

begin
    ShowMessage('Data masih kosong !');
    exit;
end;
if    MessageDlg('Apakah yakin data akan
dihapus ?',mtConfirmation, [mbYes, mbNo], 0)=mrNo then exit;
try
    t_user.Delete;
finally
    t_user.Refresh;
end;
ControlTag(f_user,[2],[3,13]);
a7.SetFocus;
end;

procedure Tf_user.a5Click(Sender: TObject);
begin
    try
        t_user.cancel;
    except
        end;
    ControlTag(f_user,[2],[3,13]);
    aa4.Text:=t_userfc_password.AsString;
    a1.SetFocus;
end;

procedure Tf_user.a4Click(Sender: TObject);
begin
    if CekData=false then exit;
    if    MessageDlg('Apakah yakin data akan
disimpan ?',mtConfirmation, [mbYes, mbNo], 0)=mrNo then exit;
    try
        t_user.Post;
    except
        MessageDlg('Proses gagal / data sudah ada !',
mtError,[mbOk], 0);
        exit;
    end;
    t_user.Refresh;
    ControlTag(f_user,[2],[3,13]);
    a1.SetFocus;
end;

```

```

procedure Tf_user.rb1Click(Sender: TObject);
begin
    if rb1.Checked then
        begin
            b3.Enabled:=true;
            b4.Enabled:=false;
            b4.text="";
            b3.SetFocus;
        end else
        if rb2.Checked then
            begin
                b4.Enabled:=true;
                b3.Enabled:=false;
                b3.text="";
                b4.SetFocus;
            end;
        end;
end;

procedure Tf_user.b2Click(Sender: TObject);
begin
    if rb1.Checked then
        begin
            if b3.text<>" then
                if not t_user.Locate('fc_userid',b3.text,[]) then
                    begin
                        ShowMessage('Data tidak ada di server !');
                        b3.SetFocus;
                        exit;
                    end;
                end else
                if rb2.Checked then
                    begin
                        if b4.text<>" then
                            if not t_user.Locate('fv_username',b4.text,[lopartialkey])
                                then
                                    begin
                                        ShowMessage('Data tidak ada di server !');
                                        b4.SetFocus;
                                        exit;
                                    end;
                                end;
                            end;
                        end;
                    end;
end;

```

```

end;

Function Tf_user.CekData:boolean;
begin
    result:=true;
    if aa1.text="" then
        begin
            ShowMessage('UserID tidak boleh kosong !');
            result:=false;
        end else
        if aa2.text="" then
            begin
                ShowMessage('Username tidak boleh kosong !');
                result:=false;
            end else
            if aa3.text="" then
                begin
                    ShowMessage('Password tidak boleh kosong !');
                    result:=false;
                end else
                if aa4.text="" then
                    begin
                        ShowMessage('Ulangi Password tidak boleh kosong !');
                        result:=false;
                    end else
                    if aa5.text="" then
                        begin
                            ShowMessage('Flag Admin tidak boleh kosong !');
                            result:=false;
                        end else
                        if aa6.text="" then
                            begin
                                ShowMessage('Flag FO/BO tidak boleh kosong !');
                                result:=false;
                            end else
                            if (aa5.text='N')and(aa6.text='N') then
                                begin
                                    ShowMessage('Flag harus dipilih salah satu !');
                                    result:=false;
                                end;
                            end;
                        end;
                    end;
                end;
            end;
        end;
    end;
end;

```

```

procedure Tf_user.aa4Exit(Sender: TObject);
begin
    ColorOnExit(sender);
    if aa4.Text<>" then
    if aa3.Text<>aa4.Text then
    begin
        ShowMessage('Ulangi Password tidak sama !');
        aa4.Text:= "";
        aa4.SetFocus;
        exit;
    end;
end;

procedure Tf_user.t_userAfterScroll(DataSet: TDataSet);
begin
    aa4.Text:=t_userfc_password.AsString;
end;

procedure Tf_user.aa1Exit(Sender: TObject);
begin
    ColorOnExit(sender);
end;

procedure Tf_user.aa1Enter(Sender: TObject);
begin
    ColorOnEnter(sender);
end;

procedure Tf_user.aa1KeyPress(Sender: TObject; var Key: Char);
begin
    if key=#13 then sendmessage(handle,wm_nextdlgctl,0,0);
end;

procedure Tf_user.aa6KeyPress(Sender: TObject; var Key: Char);
begin
    if key=#13 then aa7.SetFocus;
end;

procedure Tf_user.aa5KeyPress(Sender: TObject; var Key: Char);
begin
    if key=#13 then aa6.SetFocus else
    sendmessage(handle,wm_nextdlgctl,0,0);
end;

```



```

end;

procedure Tf_user.b3Enter(Sender: TObject);
begin
    ColorOnEnter(sender);
end;

procedure Tf_user.b3Exit(Sender: TObject);
begin
    ColorOnExit(sender);
end;

procedure Tf_user.b3KeyPress(Sender: TObject; var Key: Char);
begin
    if key=#13 then b2.SetFocus;
end;

procedure Tf_user.aa7KeyPress(Sender: TObject; var Key: Char);
begin
    if key=#13 then a4.SetFocus;
end;

procedure Tf_user.BitBtn5Click(Sender: TObject);
begin
    Notebook1.ActivePage:='page1';
end;

procedure Tf_user.BitBtn6Click(Sender: TObject);
begin
    Notebook1.ActivePage:='modul';
end;

procedure Tf_user.ha1Click(Sender: TObject);
begin
    t_mdprg.First;
    while not t_mdprg.Eof do
        begin
            try
                t_usermdl.Append;
                t_usermdlfc_userid.AsString:=t_userfc_userid.AsString;
                t_usermdlfc_mdprg.AsString:=t_mdprgfc_mdprg.AsString;
            end;
        end;
    end;
end;

```

```

        t_usermdl.Post;
    Except
        showmessage('Modul sudah diberikan !');
        t_usermdl.CancelUpdates;
        exit;
    end;
    t_mdprg.next;
end;
requery(t_mdprg);
end;

procedure Tf_user.ha4Click(Sender: TObject);
begin
    if not t_usermdl.IsEmpty then
        begin
            with dm_utama.MyQuery do
                begin
                    sql.clear;
                    sql.add(' delete from t_usermdl where
fc_userid=:fc_userid');

ParamByName('fc_userid').AsString:=t_userfc_userid.AsString;
                try
                    ExecSql;
                except
                    end;
                end;
                requery(t_mdprg);
                requery(t_usermdl);
            end;
        end;
end;

procedure Tf_user.t_usermdlBeforeOpen(DataSet: TDataSet);
begin

t_usermdl.ParamByName('fc_userid').AsString:=t_userfc_userid.
AsString;
end;

procedure Tf_user.ha2Click(Sender: TObject);
begin
    try

```

```

        t_usermdl.Append;
        t_usermdlfc_userid.AsString:=t_userfc_userid.AsString;

t_usermdlfc_mdprg.AsString:=t_mdprgfc_mdprg.AsString;
        t_usermdl.Post;
    Except
        showmessage('Modul sudah diberikan !');
        t_usermdl.CancelUpdates;
        exit;
    end;
    requery(t_mdprg);
    t_mdprg.Next;

end;

procedure Tf_user.ha3Click(Sender: TObject);
begin
    if not t_usermdl.isempty then
        begin
            try
                t_usermdl.Delete;
                t_usermdl.ApplyUpdates;
            except
                end;
            requery(t_mdprg);
            t_usermdl.next;
        end;
    end;

procedure Tf_user.DBGrid3DbClick(Sender: TObject);
begin
    ha2Click(sender);
end;

procedure Tf_user.DBGrid2DbClick(Sender: TObject);
begin
    ha3Click(sender);
end;

procedure Tf_user.t_mdprgBeforeOpen(DataSet: TDataSet);
begin

```

```
t_mdIprg.ParamByName('fc_userid').AsString:=t_userfc_userid.AsString;
end;

end.
```