

Lampiran A

Listing Program di Form Utama

```
unit UProgramSms;

interface

uses

    Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
    Dialogs, AdvSmoothMenu, ExtCtrls, ComCtrls, StdCtrls, AdvStyleIF, jpeg,
    AdvSmoothLabel, AdvSmoothButton, AdvSmoothListBox,
    AdvSmoothComboBox,
    AdvSmoothToggleButton, AdvPanel, AdvWiiProgressBar, SMSComp, WinXP,
    XPMan, DB, ADODB, ActnList, ActnMan, ToolWin, StdStyleActnCtrls, Spin,
    AdvGlassButton, AdvSelectors, GDIPPictureContainer, Grids, BaseGrid,
    AdvGrid, DBAdvGrid, DBCtrls, Mask, AdvSmoothPanel,
    Menus, AdvSmoothSplashScreen, Buttons, DBGrids, RpCon,
    RpConDS, RpDefine, RpRave;

type

    TfrmMenuUtama = class(TForm)
        AdvSmoothMenu: TAdvSmoothMenu;
        PageControl: TPageControl;
        TabSheet2: TTabSheet;
        TabSheet4: TTabSheet;
        TabSheet5: TTabSheet;
        StatusBar: TStatusBar;
        TabSheet1: TTabSheet;
        GroupBox4: TGroupBox;
        Panel1: TPanel;
        Adcdbase: TADOConnection;
```

Oxygencomponent: TOxygenSMS;
WinXP1: TWinXP;
Timer1: TTimer;
XPManifest: TXPManifest;
Timer2: TTimer;
Panel2: TPanel;
btnkonek: TAdvSmoothButton;
btndiskonek: TAdvSmoothButton;
GroupBox1: TGroupBox;
DataSourceStok: TDataSource;
ADOTableStok: TADOTable;
TabSheet6: TTabSheet;
GroupBox5: TGroupBox;
ADOTableInbox: TADOTable;
DataSourceInbox: TDataSource;
ADOTableInboxNo_Hp: TWideStringField;
ADOTableInboxNama_Toko: TWideStringField;
ADOTableInboxPesan: TWideStringField;
ADOTableInboxTanggal_Masuk: TDateTimeField;
Timer3: TTimer;
GroupBox7: TGroupBox;
DBEdit1: TDBEdit;
DBMemo1: TDBMemo;
Label1: TLabel;
Label2: TLabel;
DBEdit2: TDBEdit;
TabSheet7: TTabSheet;
GroupBox8: TGroupBox;

AdvSmoothLabel1: TAdvSmoothLabel;
GroupBox10: TGroupBox;
GroupBox9: TGroupBox;
GroupBox11: TGroupBox;
btnnewcust: TAdvSmoothButton;
btnditcust: TAdvSmoothButton;
btndelcust: TAdvSmoothButton;
btnrefcust: TAdvSmoothButton;
btnprintcust: TAdvSmoothButton;
GroupBox12: TGroupBox;
GroupBox13: TGroupBox;
GroupBox6: TGroupBox;
GroupBox14: TGroupBox;
btngetinbox: TAdvSmoothButton;
btnkirimpesan: TAdvSmoothButton;
GroupBox15: TGroupBox;
AdvSmoothPanel1: TAdvSmoothPanel;
btNewStock: TAdvSmoothButton;
btnditstock: TAdvSmoothButton;
btndeletestock: TAdvSmoothButton;
btnsearchstok: TAdvSmoothButton;
btnRefreshstock: TAdvSmoothButton;
btnprintstock: TAdvSmoothButton;
Panel3: TPanel;
AdvWiiProgressBar3: TAdvWiiProgressBar;
ADOTableCustomer: TADOTable;
DataSourceCustomer: TDataSource;
ADOTableCustomerID_Cust: TWideStringField;

ADOTableCustomerNama_Toko: TWideStringField;
ADOTableCustomerContact_Person: TWideStringField;
ADOTableCustomerAlamat: TWideStringField;
ADOTableCustomerNo_Tlp: TWideStringField;
ADOTableCustomerNo_Hp: TWideStringField;
ADOTableCustomerKeterangan: TWideStringField;
GroupBox2: TGroupBox;
Panel4: TPanel;
lbConnection: TAdvSmoothLabel;
GroupBox3: TGroupBox;
AdvSmoothPanel2: TAdvSmoothPanel;
lbcomport: TAdvSmoothLabel;
spedcomport: TSpinEdit;
lbsmsc: TLabel;
lbsmscenter: TAdvSmoothLabel;
lbtipe: TLabel;
lbtype: TAdvSmoothLabel;
lbhardver: TAdvSmoothLabel;
lbhv: TLabel;
GroupBox16: TGroupBox;
AdvSmoothPanel3: TAdvSmoothPanel;
lbsv: TLabel;
lbsoftver: TAdvSmoothLabel;
lbsoftverdate: TAdvSmoothLabel;
lbsvd: TLabel;
lbime: TLabel;
lbime: TAdvSmoothLabel;
lbstatus: TAdvSmoothLabel;

lpsc: TLabel;
Image4: TImage;
Image1: TImage;
Panel5: TPanel;
AdvWiiProgressBar1: TAdvWiiProgressBar;
AdvSmoothButton6: TAdvSmoothButton;
queryid: TADOQuery;
Panel6: TPanel;
ADOTableReport: TADOTable;
DataSourceReport: TDataSource;
Panel7: TPanel;
GroupBox17: TGroupBox;
dbgridreport: TDBAdvGrid;
AdvSmoothButton1: TAdvSmoothButton;
AdvSmoothButton2: TAdvSmoothButton;
Panel8: TPanel;
Panel9: TPanel;
GroupBox18: TGroupBox;
GroupBox19: TGroupBox;
TabSheet3: TTabSheet;
TabSheet8: TTabSheet;
GroupBox20: TGroupBox;
DataSourceDraft: TDataSource;
ADOTableDraft: TADOTable;
ADOTableDraftNo_Hp: TWideStringField;
ADOTableDraftNama_Toko: TWideStringField;
ADOTableDraftPesan: TWideStringField;
ADOTableDraftTanggal_Kirim: TWideStringField;

ADOTableDraftJam: TWideStringField;
GroupBox21: TGroupBox;
GroupBox22: TGroupBox;
AdvSmoothButton3: TAdvSmoothButton;
ADOTableInboxNama: TWideStringField;
ADOTableReportNo_Hp: TWideStringField;
ADOTableReportNama_Toko: TWideStringField;
ADOTableReportStatus: TWideStringField;
ADOTableReportTanggal_Kirim: TDateTimeField;
ADOTableStokJenis_Barang: TWideStringField;
ADOTableStokMerk_Barang: TWideStringField;
ADOTableStokType_Barang: TWideStringField;
ADOTableStokStock: TSmallintField;
ADOTableStokMasa_Garansi: TSmallintField;
ADOTableStokJenis_Garansi: TWideStringField;
ADOTableStokHarga_Beli: TIntegerField;
ADOTableStokHarga_Jual: TIntegerField;
ADOTableStokKeterangan: TWideStringField;
DBAdvGrid6: TDBAdvGrid;
querypoid: TADOQuery;
GroupBox23: TGroupBox;
AdvSmoothButton4: TAdvSmoothButton;
DataSourcePo: TDataSource;
ADOTablePo: TADOTable;
ADOTablePoNomer_PO: TWideStringField;
ADOTablePoID_Cust: TWideStringField;
ADOTablePoTanggal_Pesan: TWideStringField;
ADOTablePoKeterangan: TWideStringField;

ADOTablePoTotal: TIntegerField;
GroupBox24: TGroupBox;
AdvSmoothButton5: TAdvSmoothButton;
ADOTableReportNamaToko: TWideStringField;
DBAdvGrid2: TDBAdvGrid;
RvProjectCust: TRvProject;
RvCCust: TRvDataSetConnection;
DBAdvGrid3: TDBAdvGrid;
Label3: TLabel;
Label4: TLabel;
DBEdit3: TDBEdit;
DBEdit4: TDBEdit;
DBMemo2: TDBMemo;
AdvWiiProgressBar4: TAdvWiiProgressBar;
AdvWiiProgressBar2: TAdvWiiProgressBar;
AdvWiiProgressBar5: TAdvWiiProgressBar;
DBAdvGrid4: TDBAdvGrid;
AdvSmoothLabel3: TAdvSmoothLabel;
AdvSmoothLabel2: TAdvSmoothLabel;
AdvSmoothLabel4: TAdvSmoothLabel;
AdvSmoothLabel5: TAdvSmoothLabel;
DBAdvGrid5: TDBAdvGrid;
AdvSmoothLabel6: TAdvSmoothLabel;
GroupBox25: TGroupBox;
Image2: TImage;
GroupBox26: TGroupBox;
Label5: TLabel;
Label6: TLabel;

Label7: TLabel;
Label8: TLabel;
Label9: TLabel;
Label10: TLabel;
Image3: TImage;
Label11: TLabel;
Label12: TLabel;
Image5: TImage;
Image6: TImage;
ADOTablePoLkCustomer: TStringField;
ADOTablePoLkAlamat: TStringField;
AdvSmoothSplashScreen1: TAdvSmoothSplashScreen;
GDIPPictureContainer1: TGDIPPictureContainer;
btnDelrpt: TAdvSmoothButton;
AdvWiiProgressBar7: TAdvWiiProgressBar;
RvStock: TRvDataSetConnection;
RvProjectStock: TRvProject;
DBAdvGrid1: TDBAdvGrid;
AdvSmoothLabel7: TAdvSmoothLabel;
AdvSmoothButton7: TAdvSmoothButton;
AdvSmoothButton8: TAdvSmoothButton;
AdvSmoothButton9: TAdvSmoothButton;
AdvSmoothButton10: TAdvSmoothButton;
btnExit: TAdvSmoothButton;
AdvSmoothButton11: TAdvSmoothButton;
AdvSmoothButton12: TAdvSmoothButton;
AdvSmoothButton13: TAdvSmoothButton;
AdvSmoothButton14: TAdvSmoothButton;


```
procedure FormCreate(Sender: TObject);
procedure AdvSmoothMenuItemClick(Sender: TObject; ItemIndex: Integer);
procedure Timer1Timer(Sender: TObject);
procedure Timer2Timer(Sender: TObject);
procedure btnkonekClick(Sender: TObject);
procedure ConnState(st: boolean);
procedure btndiskonekClick(Sender: TObject);
procedure FormClose(Sender: TObject; var Action: TCloseAction);
procedure spedcomportChange(Sender: TObject);
procedure btNewStockClick(Sender: TObject);
procedure btndeditstockClick(Sender: TObject);
procedure btndeletestockClick(Sender: TObject);
procedure dbgridstokDbClick(Sender: TObject);
procedure btnRefreshstockClick(Sender: TObject);
procedure btnsearchstokClick(Sender: TObject);
procedure FormShow(Sender: TObject);
procedure btngetinboxClick(Sender: TObject);
procedure Timer3Timer(Sender: TObject);
procedure OxygencomponentSMSReportReceived(Index: Integer;
    Time: TDateTime; Send: String; Delivery: Integer);
procedure OxygencomponentSMSMessageReceived(Index: Integer;
    Time: TDateTime; Text, Send: String; Pict: TBitmap);
procedure btnnewcustClick(Sender: TObject);
procedure btndelcustClick(Sender: TObject);
procedure btnrefcustClick(Sender: TObject);
procedure btndeditcustClick(Sender: TObject);
procedure ADOTableCustomerNewRecord(DataSet: TDataSet);
procedure dbgridcustDbClickCell(Sender: TObject; ARow, ACol: Integer);
```

```

procedure btnkirimpesanClick(Sender: TObject);
procedure AdvSmoothButton6Click(Sender: TObject);
procedure DeleteMessages2Click(Sender: TObject);
procedure DeleteMessages1Click(Sender: TObject);
procedure ForwardMessages1Click(Sender: TObject);
procedure btndelrptClick(Sender: TObject);
procedure AdvSmoothButton3Click(Sender: TObject);
procedure AdvSmoothButton4Click(Sender: TObject);
procedure AdvSmoothButton5Click(Sender: TObject);
procedure btnprintcustClick(Sender: TObject);
procedure Button1Click(Sender: TObject);
procedure btnprintstockClick(Sender: TObject);
procedure btnExitClick(Sender: TObject);

private
  { Private declarations }

public
  { Public declarations }

  Function LoadMessage(ox : TOxygenSMS;location : integer) : String;
  procedure UpdateProgress(Percentage: integer);

end;

var
  frmMenuUtama: TfrmMenuUtama;
  Title: String ;

implementation

uses DateUtils , StrUtils, UInputStock, UInputStockCari, UNewCust,
  UNewMessage, RptCustomer , UPoDetail, USendtoMany;

{$R *.dfm}

```

{Fungsi ini digunakan ReadInbox berfungsi untuk membaca seluruh data dari dalam Inbox Handset, fungsi ini akan memanggil fungsi LoadMessage }

Function ReadInBox(ox : TOxygenSMS; var vCount, vBusy : String) : TStringList;

var SlotList : TStringList;

l, Count, Busy : Integer;

Locations,X : String;

Begin

Result := TStringList.Create;

Screen.Cursor := crHourGlass;

SlotList := TStringList.Create;

ox.GetInboxInfo(Count, Busy, Locations);

vCount := IntToStr(Count);

vBusy := IntToStr(Busy);

SlotList.CommaText := ox.GetInboxLocations;

Screen.Cursor := crDefault;

For l := 0 To SlotList.Count-1 do

x := frmMenuUtama.LoadMessage(ox, StrToInt(SlotList.Strings[l]));

SlotList.Free;

end;

//Menu untuk mengotrol page control

procedure TfrmMenuUtama.AdvSmoothMenuItemClick(Sender: TObject; ItemIndex: Integer);

begin

PageControl.ActivePageIndex := ItemIndex;

end;

procedure TfrmMenuUtama.FormCreate(Sender: TObject);

var

```

i: integer;

begin
    DoubleBuffered := true;    //Untuk style advsmoothmenu
    AdvSmoothMenu.SetComponentStyle(tsOffice2007Luna);

    Application.Title:='Smart Sms Software      '; //Untuk Mengatur Judul
Agar Bergerak Kiri Kanan
    Title:=Application.Title;

//program splash screen begin

    //AdvSmoothSplashScreen1.Show;

    //AdvSmoothSplashScreen1.BeginUpdate;

    //Sleep(200);

    //AdvSmoothSplashScreen1.BasicProgramInfo.ProgramVersion.Text :=
    'Version: ';

    //AdvSmoothSplashScreen1.BasicProgramInfo.ProgramVersion.Text :=
    'Version: ';

    AdvSmoothSplashScreen1.BasicProgramInfo.ProgramVersion.Location :=
    ilCustom;

    AdvSmoothSplashScreen1.BasicProgramInfo.ProgramVersion.PosX := 250;
    AdvSmoothSplashScreen1.BasicProgramInfo.ProgramVersion.PosY := 135;

    AdvSmoothSplashScreen1.EndUpdate;

    AdvSmoothSplashScreen1.ProgressBar.Top := 250;
    AdvSmoothSplashScreen1.ProgressBar.Left := 150;

    Sleep(100);

    UpdateProgress(10);

    Sleep(100);

    UpdateProgress(20);

    Sleep(100);

    UpdateProgress(30);

    Sleep(100);

    UpdateProgress(40);

```

```

Sleep(100);
UpdateProgress(50);
I := 50;
while AdvSmoothSplashScreen1.ProgressBar.Position < 100 do
begin
    Sleep(1);
    UpdateProgress(I);
    Inc(I);
end;
//program splash screen end ;
end;
procedure TfrmMenuUtama.UpdateProgress(Percentage: integer);
begin
    AdvSmoothSplashScreen1.ProgressBar.Position := Percentage;
end;
//Timer untuk mengatur animasi judul
procedure TfrmMenuUtama.Timer1Timer(Sender: TObject);
var
    Temp:String;
begin
    Temp:= Application.Title;
    Temp:= Temp+Temp[1];
    Temp:= Copy(Temp,2,length(Temp)-1);
    Application.Title:= Temp;
    frmMenuUtama.Caption:= Temp;
end;
//Timer untuk menampilkan tanggal dan jam di status bar
procedure TfrmMenuUtama.Timer2Timer(Sender: TObject);

```

```

begin
StatusBar.Panels[3].Text := FormatDateTime('dd mmmm yyyy',Now);
StatusBar.Panels[4].Text := FormatDateTime('hh:mm:ss',Now);
end;

//Program untuk melakukan koneksi

procedure TfrmMenuUtama.btnkonekClick(Sender: TObject);
begin
    Oxygencomponent.Close;
    if Oxygencomponent.Open then
    begin
        if not ADOTableInbox.Active then ADOTableInbox.Open;
        Timer3.Enabled := false;
        StatusBar.Panels[0].Text:= 'Connected' ;
        StatusBar.Panels[1].Text:= 'Phone type : ' + Oxygencomponent.PhoneType;
        StatusBar.Panels[2].Text:= 'Imei : '+Oxygencomponent.IMEI;
        lbsmsc.Caption:= Oxygencomponent.GetSMSCNumber;
        lbtipe.Caption:= Oxygencomponent.PhoneType;
        //lbsinyal.Caption:= ' 100 %';
        //lbbatre.Caption := ' 100 %';
        lbhv.Caption:= Oxygencomponent.HW;
        lbsv.Caption:= Oxygencomponent.SW;
        lbsvd.Caption:= Oxygencomponent.SWDate;
        lbime.Caption:= Oxygencomponent.IMEI;
        lbsc.Caption:= 'Connected' ;
        Image1.Visible := true ;
        Image4.Visible := false;
        Timer3.Enabled := True;
    end
end

```

```
else ShowMessage(' Maaf Port Yang Anda Pilih Tidak Sesuai, Coba Ulangi Lagi !
');
```

```
end;
```

```
procedure TfrmMenuUtama.ConnState(st: boolean);
```

```
begin
```

```
  spedcomport.Enabled:= not st ;
```

```
end;
```

```
//Program untuk menghentikan koneksi
```

```
procedure TfrmMenuUtama.btndiskonekClick(Sender: TObject);
```

```
begin
```

```
  Oxygencomponent.Close;
```

```
  Timer3.Enabled := false;
```

```
  StatusBar.Panels[0].text := 'Not Connected';
```

```
  StatusBar.Panels[1].text := 'Phone type: none';
```

```
  StatusBar.Panels[2].text := 'IMEI:';
```

```
  lbsmsc.caption := ' Disconnected ';
```

```
  lbtype.Caption := ' Disconnected ';
```

```
  //lbsinyal.Caption := 'Disconnected';
```

```
  //lbbatre.Caption := 'Disconnected';
```

```
  lbhv.Caption := 'Disconnected';
```

```
  lbsv.Caption := 'Disconnected';
```

```
  lbsvd.Caption := 'Disconnected';
```

```
  lbime.Caption := 'Disconnected';
```

```
  lbsc.Caption := 'Disconnected';
```

```
  Image4.Visible := true ;
```

```
  Image1.Visible := false;
```

```
end;
```

```
//Perintah pada saat keluar dari program untuk menghentikan koneksi ke  
com. port
```

```
procedure TfrmMenuUtama.FormClose(Sender: TObject; var Action: TCloseAction);
```

```
begin
```

```
Oxygencomponent.Close;
```

```
Oxygencomponent.Free;
```

```
ADOTablePo.Close;
```

```
end;
```

```
//Perintah untuk mengatur com. port yang digunakan
```

```
procedure TfrmMenuUtama.spedcomportChange(Sender: TObject);
```

```
begin
```

```
Oxygencomponent.ComNumber := StrToInt(spedcomport.Text);
```

```
end;
```

```
procedure TfrmMenuUtama.btNewStockClick(Sender: TObject);
```

```
begin
```

```
ADOTableStok.Append;
```

```
FrmInputStock.Show;
```

```
frmInputStock.DBEditmasagaransi.Text:= '0';
```

```
frmInputStock.DBEdithb.Text := '0';
```

```
frmInputStock.DBEdithj.Text := '0';
```

```
frmInputStock.DBEditstok.Text := '0';
```

```
end;
```

```
procedure TfrmMenuUtama.btneditstockClick(Sender: TObject);
```

```
begin
```

```
frmInputStock.ShowModal;
```

```
end;
```

```
procedure TfrmMenuUtama.btndeletestockClick(Sender: TObject);
```

```
begin
```

```
ADOTableStok.Delete;
```


end;

procedure TfrmMenuUtama.dbgridstokDbClick(Sender: TObject);

begin

frmInputStok.ShowModal;

end;

procedure TfrmMenuUtama.btnRefreshstockClick(Sender: TObject);

begin

ADOTableStok.Refresh;

ADOTableStok.IndexFieldNames := 'Jenis_Barang';

end;

procedure TfrmMenuUtama.btnsearchstokClick(Sender: TObject);

begin

frmsearchstok.ComboBoxSearchStok.Items.Clear;

frmsearchstok.ComboBoxSearchStok.Items.Add('Jenis_Barang');

frmsearchstok.ComboBoxSearchStok.Items.Add('Merk_Barang');

frmsearchstok.Editsearchstok.Clear;

frmsearchstok.ShowModal;

If (frmsearchstok.ComboBoxSearchStok.Text<>") AND
(frmsearchstok.Editsearchstok.Text<>") Then
ADOTableStok.Locate(frmsearchstok.ComboBoxSearchStok.Text,frmsearchstok.
Editsearchstok.Text,[loPartialKey]);

end;

{Fungsi ini digunakan untuk membaca satu SMS dalam salah satu slot/lokasi dalam InBox}

Function ReadMessageFromLocation(ox : TOxygenSMS;

location : integer) : TStringList;

var IsMsg: WordBool;

Time: TDateTime;

Text, Send : string;

Deliv: integer;

```

    Pict : TBitmap;
Begin
    Result := TStringList.Create;
    If ox.ReadSMSMessage( Location, isMsg,
        Time, Text, Send, Deliv, Pict) Then
        Begin
            Result.Add('Message Location ' + Format('%2.2d', [Location] ));
            Result.Add('Date ' + FormatDateTime('dd/mm/yyyy hh:nn:ss', Time) );
            Result.Add("");
            Result.Add('Message');
            Result.Add(Text);
            Result.Add("");
            Result.Add('Sender' + Send);
        End
    Else Result.Add('Slot is empty');
End;

{Fungsi ini digunakan untuk untuk menampilkan ringkasan singkat dari SMS dalam inbox }

Function TfrmMenuUtama.LoadMessage(ox : TOxygenSMS;location : integer) : String;
var IsMsg: WordBool;
    Time: TDateTime;
    Text, Send, vlist : string;
    Deliv: integer;
    Pict : TBitmap;
Begin
    If ox.ReadSMSMessage( Location, isMsg,Time, Text, Send, Deliv, Pict) Then
        Begin
            ADOTableInbox.Append;

```

```

ADOTableInbox.FieldByName('Tanggal_Masuk').AsDateTime := Now;
ADOTableInbox.FieldByName('No_HP').AsString := Send;
ADOTableInbox.FieldByName('Pesan').AsString := Text;
ADOTableInbox.Post;
Oxygencomponent.DeleteSMSMessage(location);

End

Else vlist := 'Slot is empty';

Result := vlist;

End;

procedure TfrmMenuUtama.FormShow(Sender: TObject);
begin
Timer3.Enabled := false;
ADOTablePo.Open;
end;

procedure TfrmMenuUtama.btngetinboxClick(Sender: TObject);
Var vCount,vBusy : String;
begin
If Oxygencomponent.Open then
begin
Timer3.Enabled := false;
ReadInBox(Oxygencomponent,vCount,vBusy);
Timer3.Enabled := true;
end
end;

procedure TfrmMenuUtama.Timer3Timer(Sender: TObject);
begin
if not Oxygencomponent.CheckConnection then
begin

```

```

If Oxygencomponent.Open then
begin
    if not ADOTableInbox.Active then ADOTableInbox.Open;
    Timer3.Enabled := false;
    Timer3.Enabled := true;
end;
end
end;

procedure TfrmMenuUtama.OxygencomponentSMSReportReceived(Index: Integer; // Program untuk mengatur komonen oxygen dengan database
Time: TDateTime; Send: String; Delivery: Integer);
    //Var vCount,vBusy : String;
Begin
    if not ADOTableReport.Active then ADOTableReport.Open;
    ADOTableReport.Append;
    ADOTableReport.FieldName('No_Hp').AsString := Send;
    ADOTableReport.FieldName('Tanggal_Kirim').AsDateTime := Now;
    Case Delivery Of
    0 : ADOTableReport.FieldName('Status').AsString := 'Ter kirim';
    1 : ADOTableReport.FieldName('Status').AsString := 'Failed';
    2 : ADOTableReport.FieldName('Status').AsString := 'Pending';
    3 : ADOTableReport.FieldName('Status').AsString := 'Failed';
    End;
    ADOTableReport.Post;
    Oxygencomponent.DeleteSMSMessage(Index);
end;

procedure
TfrmMenuUtama.OxygencomponentSMSMessageReceived(Index: Integer;
Time: TDateTime; Text, Send: String; Pict: TBitmap);

```

```

Var vCount,vBusy : String;
Begin
  ReadInBox(Oxygencomponent,vCount,vBusy);
end;
procedure TfrmMenuUtama.btnnewcustClick(Sender: TObject);
begin
ADOTableCustomer.Append;
frmNewCust.Show;
end;
procedure TfrmMenuUtama.btndelcustClick(Sender: TObject);
begin
  MessageDlg(' Are You Sure To Delete This Record ? ',mtWarning,[mbOK],0);
ADOTableCustomer.Delete;
end;
procedure TfrmMenuUtama.btnrefcustClick(Sender: TObject);
begin
ADOTableCustomer.Refresh;
end;
procedure TfrmMenuUtama.btncustClick(Sender: TObject);
begin
frmNewCust.ShowModal;
end;
procedure TfrmMenuUtama.ADOTableCustomerNewRecord(DataSet: TDataSet);
begin
  queryid.Close;
  queryid.Open;
  if queryid.Fields[0].Value = NULL then
    ADOTableCustomerID_Cust.Value:='C-'+FormatFloat('000', 1)

```

```

else

  ADOTableCustomerID_Cust.Value:='C-'+FormatFloat('000',
queryid.Fields[0].Value);

end;

procedure TfrmMenuUtama.dbgridcustDbClickCell(Sender: TObject;
ARow,
  ACol: Integer);
begin
frmNewCust.ShowModal;
end;

procedure TfrmMenuUtama.btnkirimpesanClick(Sender: TObject);
begin
ADOTableDraft.Append;
frmNewMessage.Show;
frmNewMessage.Memomes.Clear;
frmNewMessage.Memomes.SetFocus;
end;

procedure TfrmMenuUtama.AdvSmoothButton6Click(Sender: TObject);
begin
ADOTableInbox.Delete;
end;

procedure TfrmMenuUtama.DeleteMessages2Click(Sender: TObject);
begin
ADOTableInbox.DeleteRecords(arCurrent);
end;

procedure TfrmMenuUtama.DeleteMessages1Click(Sender: TObject);
begin
ADOTableDraft.Append;
frmNewMessage.Show;

```

```
frmNewMessage.Memomes.Clear;
end;
procedure TfrmMenuUtama.ForwardMessages1Click(Sender: TObject);
begin
frmNewMessage.Show;
end;
procedure TfrmMenuUtama.btndelrptClick(Sender: TObject);
begin
frmMenuUtama.ADOTableReport.DeleteRecords(arCurrent);
end;
procedure TfrmMenuUtama.AdvSmoothButton3Click(Sender: TObject);
begin
ADOTableDraft.Delete;
end;

procedure TfrmMenuUtama.AdvSmoothButton4Click(Sender: TObject);
begin
frmPoDetail.show;
frmPoDetail.ADOTablePoDetil.Append;
end;
procedure TfrmMenuUtama.AdvSmoothButton5Click(Sender: TObject);
begin
frmSendtomany.ShowModal;
frmSendtomany.DBMemoSentomany.Clear;
//frmSendtomany.DBMemoSentomany.SetFocus;
end;
procedure TfrmMenuUtama.btnprintcustClick(Sender: TObject);
begin
```

RvProjectCust.Execute;

end;

procedure TfrmMenuUtama.Button1Click(Sender: TObject);

begin

frmMenuUtama.Refresh;

frmSendtomany.Refresh;

end;

procedure TfrmMenuUtama.btnprintstockClick(Sender: TObject);

begin

RvProjectStock.Execute;

end;

procedure TfrmMenuUtama.btnExitClick(Sender: TObject);

begin

Application.Terminate;

end;

end.

Lampiran B

Listing Program Setiap *Form*

➤ Program Input Stok

```
unit UInputStock;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, DB, ADODB, AdvSmoothButton, StdCtrls, Mask, DBCtrls,
  AdvSmoothMessageDialog;
type
  TfrmInputStock = class(TForm)
    GroupBox1: TGroupBox;
    lbbarang: TLabel;
    lbmerk: TLabel;
    lbtype: TLabel;
    lbmasagaransi: TLabel;
    lbjenisgaransi: TLabel;
    lbhb: TLabel;
    lbhj: TLabel;
    lbket: TLabel;
    DBEdittipe: TDBEdit;
    DBEditmasagaransi: TDBEdit;
    DBEdithb: TDBEdit;
    DBEdithj: TDBEdit;
    DBEditstok: TDBEdit;
    DBEditket: TDBEdit;
```

```

Label1: TLabel;
DBComboBoxjenisbarang: TDBComboBox;
DBComboBoxmerkbarang: TDBComboBox;
DBComboBox1: TDBComboBox;
GroupBox2: TGroupBox;
btninputok: TAdvSmoothButton;
btninputcancel: TAdvSmoothButton;
Label2: TLabel;
procedure btninputokClick(Sender: TObject);
procedure btninputcancelClick(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
End;
var
  frmInputStock: TfrmInputStock;
implementation
uses UProgramSms;
{$R *.dfm}
procedure TfrmInputStock.btninputokClick(Sender: TObject);
begin
IF frmMenuUtama.ADOTableStokType_Barang.Value<>"then
  begin
    IF frmMenuUtama.DataSourceStok.State In [dsInsert,dsEdit]then
    frmMenuUtama.ADOTableStok.Post;
  end;
  MessageDlg('Apakah Data Yang Ada Sudah Diisi Dengan
Lengkap?',mtWarning,[mbOK],0);

```

```
    Close;
end;
procedure TfrmInputStock.btninputcancelClick(Sender: TObject);
begin
    frmMenuUtama.ADOTableStok.Cancel;
    frmInputStock.close;
end; end.
```

➤ **Program *Input Customer***

```
unit UNewCust;
interface
uses
    Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
    Dialogs, ExtCtrls, StdCtrls, AdvSmoothButton, Mask, DBCtrls, DB, ADODB;
type
    TfrmNewCust = class(TForm)
        GroupBox1: TGroupBox;
        Panel1: TPanel;
        LblCust: TLabel;
        Label1: TLabel;
        Label2: TLabel;
        Label3: TLabel;
        Label4: TLabel;
        Label5: TLabel;
        Label6: TLabel;
        DBEdit1: TDBEdit;
        DBEdit2: TDBEdit;
        DBEdit3: TDBEdit;
    end;
end.
```

```

DBEdit4: TDBEdit;
DBEdit5: TDBEdit;
DBEdit6: TDBEdit;
DBEdit7: TDBEdit;
GroupBox2: TGroupBox;
AdvSmoothButton1: TAdvSmoothButton;
AdvSmoothButton2: TAdvSmoothButton;
procedure AdvSmoothButton1Click(Sender: TObject);
procedure AdvSmoothButton2Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;
var
  frmNewCust: TfrmNewCust;
implementation
uses UProgramSms;
{$R *.dfm}
procedure TfrmNewCust.AdvSmoothButton1Click(Sender: TObject);
begin
IF frmMenuUtama.ADOTableCustomerID_Cust.Value<>"then
  begin
    IF frmMenuUtama.DataSourceCustomer.State In [dsInsert,dsEdit]then
      frmMenuUtama.ADOTableCustomer.Post;
  end;
  MessageDlg('Apakah Data Yang Ada Sudah Diisi Dengan
  Lengkap?',mtWarning,[mbOK],0);
  Close;

```

```

end;

procedure TfrmNewCust.AdvSmoothButton2Click(Sender: TObject);

begin

frmMenuUtama.ADOTableCustomer.Cancel;

frmMenuUtama.ADOTableCustomer.CancelUpdates;

frmNewCust.Close;

end;

end.

```

➤ **Program New Messages**

```

unit UNewMessage;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, DB, ADODB, Grids, BaseGrid, AdvGrid, DBAdvGrid, StdCtrls,
  AdvSmoothButton, DBCtrls, Mask, AdvWiiProgressBar, AdvSmoothLabel;
type
  TfrmNewMessage = class(TForm)
    GroupBox1: TGroupBox;
    dbgriddatacust: TDBAdvGrid;
    DataSourceCustinNewM: TDataSource;
    ADOTableCustinNewM: TADOTable;
    ADOTableCustinNewMNama_Toko: TWideStringField;
    ADOTableCustinNewMAalamat: TWideStringField;
    ADOTableCustinNewMNo_Hp: TWideStringField;
    GroupBox2: TGroupBox;
    GroupBox3: TGroupBox;
    Label2: TLabel;
    GroupBox4: TGroupBox;
    Label1: TLabel;
    dbdtnamatoko: TDBEdit;
    Label3: TLabel;
    Label4: TLabel;
    dbedtalamat: TDBEdit;
    dbedtnomerhp: TDBEdit;
    Label5: TLabel;
    GroupBox5: TGroupBox;
    AdvSmoothButton1: TAdvSmoothButton;
    AdvSmoothButton2: TAdvSmoothButton;
    Memomes: TDBMemo;
    DBEdit1: TDBEdit;
    DBEdit2: TDBEdit;
    DBEdit3: TDBEdit;
    DBEdit4: TDBEdit;

```

```

AdvWiiProgressBar5: TAdvWiiProgressBar;
AdvSmoothLabel2: TAdvSmoothLabel;
procedure AdvSmoothButton1Click(Sender: TObject);
procedure AdvSmoothButton2Click(Sender: TObject);
procedure dbdtnamatokoChange(Sender: TObject);
procedure FormShow(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  frmNewMessage: TfrmNewMessage;
implementation

uses UProgramSms;
{$R *.dfm}
procedure TfrmNewMessage.AdvSmoothButton1Click(Sender: TObject);
begin
  // tambah nama form tempat oxygen frm.oxygen....
  if
    frmMenuUtama.Oxygencomponent.SendSMSMessage(dbedtnomerhp.Text,M
    emomes.Text,167,true,false,nil) then
    begin
      ShowMessage('SMS sudah dikirim')
      //FSMS.SaveMessage;
    end
  else
    ShowMessage('Pengiriman SMS Gagal, silahkan ulang kembali !');
  IF frmMenuUtama.ADOTableDraftPesan.Value<>"then
  begin
    IF frmMenuUtama.DataSourceDraft.State In [dsInsert,dsEdit]then
    frmMenuUtama.ADOTableDraft.Post;
    end;
    //MessageDlg('Apakah Data Yang Ada Sudah Diisi Dengan
    Lengkap?',mtWarning,[mbOK],0);
    //Close;
    end;
procedure TfrmNewMessage.AdvSmoothButton2Click(Sender: TObject);
begin
  Memomes.Clear;
  frmMenuUtama.ADOTableDraft.Cancel;
  frmMenuUtama.ADOTableDraft.CancelUpdates;
  frmNewMessage.Close;
  end;
procedure TfrmNewMessage.dbdtnamatokoChange(Sender: TObject);
begin
  DBEdit1.Text := dbdtnamatoko.Text;
  DBEdit2.Text := dbedtnomerhp.Text;

```

```

end;
procedure TfrmNewMessage.FormShow(Sender: TObject);
begin
DBEdit1.Visible := false;
DBEdit2.Visible := false;
DBEdit3.Visible := false;
DBEdit4.Visible := false;
DBEdit1.Text := dbdtnamatoko.Text;
DBEdit2.Text := dbedtnomerhp.Text;
DBEdit3.Text := FormatDateTime('dd mmmm yyyy',Now);
DBEdit4.Text := FormatDateTime('hh:mm:ss',Now);
Memomes.Clear;
end;
end.

```

➤ Program Po Detail

```

unit UPoDetail;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, Grids, BaseGrid, AdvGrid, DBAdvGrid, Buttons, Mask,
  DBCtrls, DB, ADODB, AdvSmoothButton, ComCtrls, DateUtils, DBGrids;
type
  TfrmPoDetail = class(TForm)
    LblCust: TLabel;
    Label1: TLabel;
    DBEdit1: TDBEdit;
    DBEdit2: TDBEdit;
    Label2: TLabel;
    Label3: TLabel;
    Label4: TLabel;
    SpeedButton1: TSpeedButton;
    DBEdit3: TDBEdit;
    DBEdit4: TDBEdit;
    Label5: TLabel;
    DBEdit5: TDBEdit;
    Label6: TLabel;
    GroupBox1: TGroupBox;
    AdvSmoothButton1: TAdvSmoothButton;
    AdvSmoothButton2: TAdvSmoothButton;
    DataSourcePoDetil: TDataSource;
    ADOTablePoDetil: TADOTable;
    ADOTablePoDetilNomer_PO: TWideStringField;
    ADOTablePoDetilID_Cust: TWideStringField;
    ADOTablePoDetilMerk_Barang: TWideStringField;
    ADOTablePoDetilType_Barang: TWideStringField;
    ADOTablePoDetilQty: TIntegerField;

```

```

ADOTablePoDetilHarga: TIntegerField;
ADOTablePoDetilPotongan: TIntegerField;
ADOTablePoDetilTotal: TIntegerField;
DateTimePicker1: TDateTimePicker;
Label11: TLabel;
DBEdit9: TDBEdit;
DBGrid1: TDBGrid;
ADOTablePoDetilLkTypeBarang: TStringField;
procedure SpeedButton1Click(Sender: TObject);
procedure DateTimePicker1Change(Sender: TObject);
procedure AdvSmoothButton1Click(Sender: TObject);
procedure FormShow(Sender: TObject);
procedure FormClose(Sender: TObject; var Action: TCloseAction);
procedure ADOTablePoDetilBeforeInsert(DataSet: TDataSet);
procedure ADOTablePoDetilMerk_BarangChange(Sender: TField);
procedure ADOTablePoDetilType_BarangChange(Sender: TField);
private
  { Private declarations }
public
  { Public declarations }
end;
var
  frmPoDetail: TfrmPoDetail;
implementation
uses UProgramSms, UPoDetailPelanggan;
{$R *.dfm}
procedure TfrmPoDetail.SpeedButton1Click(Sender: TObject);
begin
frmPoDetailPelanggan.show;
end;
procedure TfrmPoDetail.DateTimePicker1Change(Sender: TObject);
begin
  frmMenuUtama.ADOTablePo.Edit;
  frmMenuUtama.ADOTablePoTanggal_Pesan.AsDateTime := DateOf
(DateTimePicker1.Date);
end;
procedure TfrmPoDetail.AdvSmoothButton1Click(Sender: TObject);
begin
  if DataSourcePoDetil.State IN [dsInsert, dsEdit] Then ADOTablePoDetil.Post;
end;
procedure TfrmPoDetail.FormShow(Sender: TObject);
begin
ADOTablePoDetil.Open;
frmMenuUtama.ADOTableStok.Open;
frmMenuUtama.ADOTableCustomer.Open;
end;
procedure TfrmPoDetail.FormClose(Sender: TObject;
  var Action: TCloseAction);
begin
ADOTablePoDetil.Close;

```



```

frmMenuUtama.ADOTableStok.Close;
frmMenuUtama.ADOTableCustomer.Close;
end;
procedure TfrmPoDetail.ADOTablePoDetilBeforeInsert(DataSet:
TDataSet);
begin
if frmMenuUtama.DataSourcePo.State IN [dsInsert, dsEdit] Then
frmMenuUtama.ADOTablePo.Post;
end;
procedure TfrmPoDetail.ADOTablePoDetilMerk_BarangChange(Sender:
TField);
begin
ADOTablePoDetilHarga.AsFloat :=
frmMenuUtama.ADOTableStokHarga_Jual.AsFloat;
end;
procedure TfrmPoDetail.ADOTablePoDetilType_BarangChange(Sender:
TField);
begin
ADOTablePoDetilMerk_Barang.AsString :=
frmMenuUtama.ADOTableStokMerk_Barang.AsString;
end;
end.

```

➤ Program Po Detail Pelanggan

```

unit UPoDetailPelanggan;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, Grids, BaseGrid, AdvGrid, DBAdvGrid;

type

TfrmPoDetailPelanggan = class(TForm)

DBAdvGrid1: TDBAdvGrid;

procedure DBAdvGrid1DbClick(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

```

```

    end;
var
    frmPoDetailPelanggan: TfrmPoDetailPelanggan;
implementation
uses UProgramSms;
{$R *.dfm}
procedure TfrmPoDetailPelanggan.DBAdvGrid1DbClick(Sender: TObject);
begin
    frmMenuUtama.ADOTablePo.Edit;
    frmMenuUtama.ADOTablePoID_Cust.AsString :=
        frmMenuUtama.ADOTableCustomerID_Cust.AsString;
end;
end.

```

➤ **Program Queue Messages**

```

unit UQueueSms;
interface
uses
    Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
    Dialogs, Grids, BaseGrid, AdvGrid, DBAdvGrid, AdvSmoothButton, DBGrids,
    StdCtrls, AdvSmoothLabel, AdvWiiProgressBar;
type
    Tfrmqueue = class(TForm)
        DBAdvGrid1: TDBAdvGrid;
        GroupBox1: TGroupBox;
        AdvSmoothButton2: TAdvSmoothButton;
        AdvSmoothButton1: TAdvSmoothButton;
        AdvWiiProgressBar5: TAdvWiiProgressBar;
    end;

```

```

AdvSmoothLabel2: TAdvSmoothLabel;
AdvSmoothButton3: TAdvSmoothButton;
procedure AdvSmoothButton1Click(Sender: TObject);
procedure Button1Click(Sender: TObject);
procedure AdvSmoothButton2Click(Sender: TObject);
procedure AdvSmoothButton3Click(Sender: TObject);

private
  { Private declarations }

public
  { Public declarations }

end;

var
  frmqueue: Tfrmqueue;

implementation

uses UProgramSms, USendtoMany;
{$R *.dfm}

procedure Tfrmqueue.AdvSmoothButton1Click(Sender: TObject);
begin
  frmSendtomany.ADOTableStoMany.Delete;
end;

procedure Tfrmqueue.Button1Click(Sender: TObject);
begin
  while not frmSendtomany.ADOTableStoMany.Eof do
    begin
      if
Length(frmSendtomany.ADOTableStoMany.FieldName('Pesan').AsString)>0
then
        begin
          if
frmMenuUtama.Oxygencomponent.SendSMSMessage(frmSendtomany.ADOTab

```

```
leStoManyNo_Hp.AsString,frmSendtomany.DBMemoSentomany.Text,167,true,fa  
lse,nil) then
```

```
    frmSendtomany.ADOTableStoMany.Delete
```

```
else
```

```
    frmSendtomany.ADOTableStoMany.Next;
```

```
end
```

```
else
```

```
    frmSendtomany.ADOTableStoMany.Next;
```

```
end;
```

```
ShowMessage('SMS sudah dikirim');
```

```
end;
```

```
procedure Tfrmqueue.AdvSmoothButton2Click(Sender: TObject);
```

```
begin
```

```
while not frmSendtomany.ADOTableStoMany.Eof do
```

```
    begin
```

```
        if
```

```
Length(frmSendtomany.ADOTableStoMany.FieldByName('Pesan').AsString)>0  
then
```

```
    begin
```

```
        if
```

```
frmMenuUtama.Oxygencomponent.SendSMSMessage(frmSendtomany.ADOTab  
leStoManyNo_Hp.AsString,frmSendtomany.DBMemoSentomany.Text,167,true,fa  
lse,nil) then
```

```
            frmSendtomany.ADOTableStoMany.Delete
```

```
        else
```

```
            frmSendtomany.ADOTableStoMany.Next;
```

```
        end
```

```
    else
```

```
        frmSendtomany.ADOTableStoMany.Next;
```

```
    end;
```

```

    ShowMessage('SMS sudah dikirim');
end;
procedure Tfrmqueue.AdvSmoothButton3Click(Sender: TObject);
begin
frmqueue.Close;
end;
end.

```

➤ **Program Cari Data (Pelanggan)**

```

unit UInputStockCari;
interface
uses
    Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
    Dialogs, AdvSmoothButton, ExtCtrls, StdCtrls;
type
    Tfrmsearchstok = class(TForm)
        GroupBox1: TGroupBox;
        lbfieldsearch: TLabel;
        ComboBoxSearchStok: TComboBox;
        Editsearchstok: TEdit;
        Panel1: TPanel;
        AdvSmoothButton2: TAdvSmoothButton;
        AdvSmoothButton1: TAdvSmoothButton;
        procedure AdvSmoothButton2Click(Sender: TObject);
        procedure AdvSmoothButton1Click(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
    end;
var
    frmsearchstok: Tfrmsearchstok;
implementation
uses UProgramSms;
{$R *.dfm}
procedure Tfrmsearchstok.AdvSmoothButton2Click(Sender: TObject);
begin
    Close ;
end;
procedure Tfrmsearchstok.AdvSmoothButton1Click(Sender: TObject);
begin

```

```
Close;  
end;  
end.
```

➤ **Program *Send To Many***

```
unit USendtoMany;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, Grids, BaseGrid, AdvGrid, DBAdvGrid, AdvSmoothButton, StdCtrls,  
DBCtrls, Mask, DB, ADODB, DBGrids, AdvWiiProgressBar, AdvSmoothLabel;
```

```
type
```

```
TfrmSendtomany = class(TForm)
```

```
  GroupBox1: TGroupBox;
```

```
  dbgriddatacust: TDBAdvGrid;
```

```
  GroupBox2: TGroupBox;
```

```
  Label2: TLabel;
```

```
  GroupBox4: TGroupBox;
```

```
  Label1: TLabel;
```

```
  Label3: TLabel;
```

```
  dbdtnamatoko: TDBEdit;
```

```
  dbedtnomerhp: TDBEdit;
```

```
  GroupBox3: TGroupBox;
```

```
  Label5: TLabel;
```

```
  GroupBox5: TGroupBox;
```

```
  AdvSmoothButton2: TAdvSmoothButton;
```

```
  DataSourceCustinNewSto: TDataSource;
```

```
  ADOTableCustinNewSto: TADOTable;
```

```
  ADOTableCustinNewStoNama_Toko: TWideStringField;
```

```
  ADOTableCustinNewStoAlamat: TWideStringField;
```

```
ADOTableCustinNewStoNo_Hp: TWideStringField;
ADOTableStoMany: TADOTable;
DataSourceStoMany: TDataSource;
ADOTableStoManyNo_Hp: TWideStringField;
ADOTableStoManyNama_Toko: TWideStringField;
ADOTableStoManyPesan: TWideStringField;
AdvSmoothButton1: TAdvSmoothButton;
AdvSmoothButton4: TAdvSmoothButton;
DBMemoSentomany: TDBMemo;
DBEdit1: TDBEdit;
DBEdit2: TDBEdit;
AdvWiiProgressBar5: TAdvWiiProgressBar;
AdvSmoothLabel2: TAdvSmoothLabel;
Edit1: TEdit;
procedure AdvSmoothButton3Click(Sender: TObject);
procedure AdvSmoothButton2Click(Sender: TObject);
procedure DBEdit1Change(Sender: TObject);
procedure AdvSmoothButton1Click(Sender: TObject);
procedure Button1Click(Sender: TObject);
procedure AdvSmoothButton4Click(Sender: TObject);
procedure FormShow(Sender: TObject);
procedure dbdtnamatokoChange(Sender: TObject);
procedure FormCreate(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;
```

```

var
    frmSendtomany: TfrmSendtomany;
implementation
uses UProgramSms, UQueueSms;
{$R *.dfm}

procedure TfrmSendtomany.AdvSmoothButton3Click(Sender: TObject);
begin
    DBMemoSentomany.Clear;
end;

procedure TfrmSendtomany.AdvSmoothButton2Click(Sender: TObject);
begin
    frmSendtomany.DBMemoSentomany.Clear;
    frmSendtomany.ADOTableStoMany.Cancel;
    frmSendtomany.ADOTableStoMany.CancelUpdates;
    frmSendtomany.Close;
end;

procedure TfrmSendtomany.DBEdit1Change(Sender: TObject);
begin
    //dbtokos.Text := DBEdit1.Text;
    //DBEdit3.text := DBEdit2.Text;
end;

procedure TfrmSendtomany.AdvSmoothButton1Click(Sender: TObject);
begin
    IF frmSendtomany.ADOTableStoManyNo_Hp.Value<>" then
        begin
            If frmSendtomany.DataSourceStoMany.State In [dsInsert, dsEdit] Then
                frmSendtomany.ADOTableStoMany.Post;
        end;
end;

```



```

//ADOTableStoMany.Append;

//MessageDlg('Apakah Data Yang Ada Sudah Diisi Dengan
Lengkap?',mtWarning,[mbOK],0);

//Close;

end;

procedure TfrmSendtomany.Button1Click(Sender: TObject);
begin
ADOTableStoMany.UpdateRecord;
end;

procedure TfrmSendtomany.AdvSmoothButton4Click(Sender: TObject);
begin
ADOTableStoMany.Append;
//ADOTableStoMany.Open;
frmqueue.showmodal;
end;

procedure TfrmSendtomany.FormShow(Sender: TObject);
begin
DBEdit1.Text := dbdtnamatoko.Text;
DBEdit2.Text := dbedtnomerhp.Text;
end;

procedure TfrmSendtomany.dbdtnamatokoChange(Sender: TObject);
begin
DBEdit1.Text := dbdtnamatoko.Text;
DBEdit2.Text := dbedtnomerhp.Text;
end;

procedure TfrmSendtomany.FormCreate(Sender: TObject);
begin
frmSendtomany.dbdtnamatoko.Visible := false;
frmSendtomany.dbedtnomerhp.Visible := false;

```

```
frmSendtomany.AdvSmoothButton1.Visible := false;
```

```
end;
```

```
end.
```

Lampiran C

Data Pengamatan Pengiriman dan Penerimaan Pesan

No.	Waktu Pengiriman Pesan	Operator 1	Operator 2	Laporan Pengiriman	Waktu Penerimaan Pesan	Status
1	3 Juni 2010 00:42:24 AM	Telkomsel	Telkomsel	Terkirim	3 Juni 2010 00:43:55 AM	Berhasil
2	18 Juni 2010 8:21:16 AM	Telkomsel	Telkomsel	Tertunda / <i>Pending</i>	18 Juni 2010 8:25:05 AM	Berhasil
3	18 Juni 2010 9:45:25 AM	Telkomsel	XL	Tertunda / <i>Pending</i>	18 Juni 2010 9:50:38 AM	Berhasil
4	18 Juni 2010 17:30:46 AM	Telkomsel	XL	Terkirim	18 Juni 2010 17:32:06 AM	Berhasil
5	29 Juli 2010 10:28:19 PM	Telkomsel	Indosat	Terkirim	29 Juli 2010 10:30:10 PM	Berhasil
6	29 Juli 2010 22:28:19 PM	Telkomsel	Indosat	Terkirim	Tidak Diterima	Gagal
7	29 Juli 2010 23:51:49PM	Telkomsel	Indosat	Terkirim	Tidak Diterima	Gagal
8	30 Juli 2010 11:38:19 AM	Telkomsel	Telkomsel	Terkirim	30 Juli 2010 11:40:29 AM	Berhasil
9	31 Juli 2010 10:46:19 AM	Telkomsel	Indosat	Terkirim	31 Juli 2010 10:55:36 AM	Berhasil