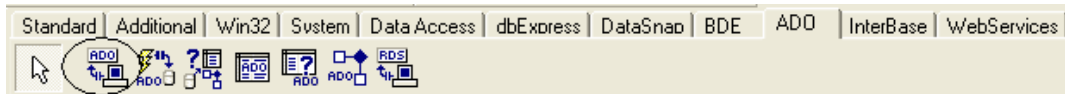


## 4. IMPLEMENTASI SISTEM

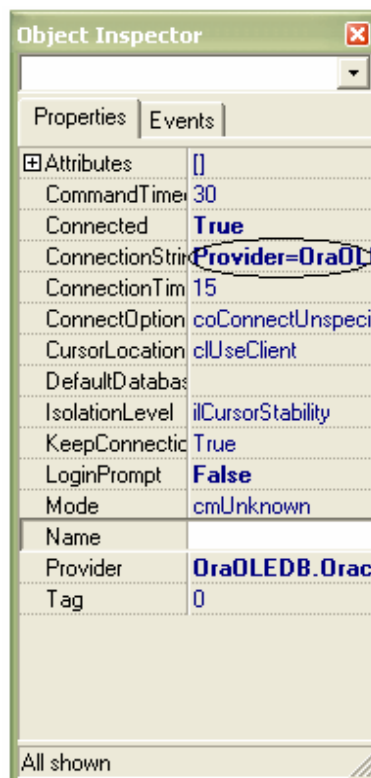
### 4.1. Setting Koneksi Database

Pembuatan *database* pada Borland Delphi 7 memerlukan penyetingan koneksi terlebih dahulu. Langkah pertama yaitu memilih komponen ADOConnection.



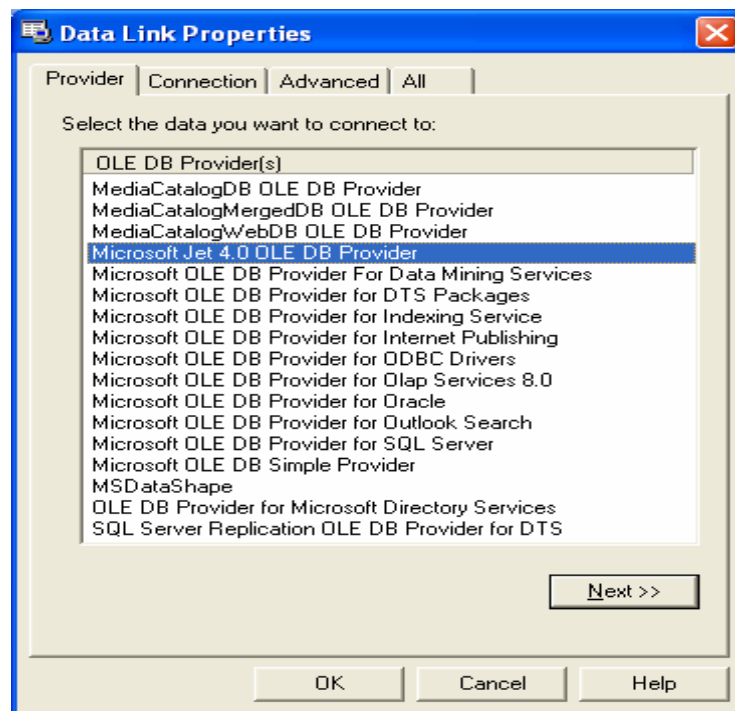
Gambar 4.1. Komponen ADOConnection

Komponen ADOConnection disimpan di DataModule. Setelah itu untuk melakukan koneksi *database*, pilih Connection String yang terletak di *Object Inspector* dari ADOConnection.



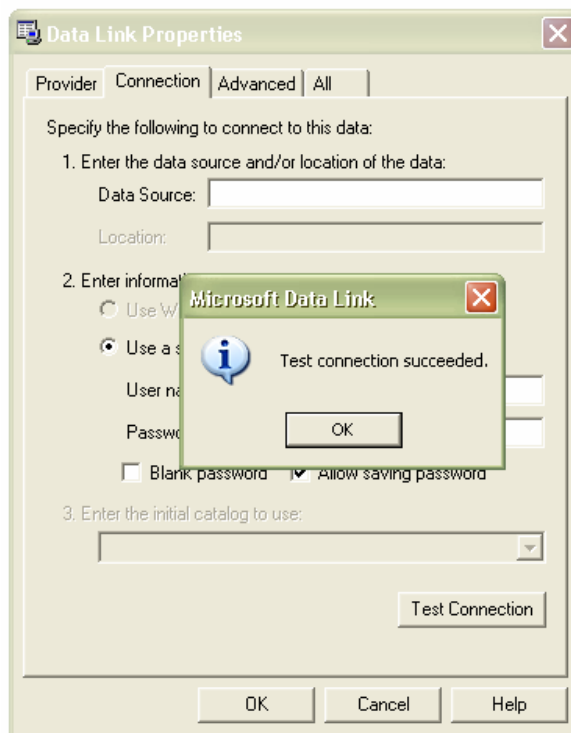
Gambar 4.2. Object Inspector ADOConnection

Setelah tombol `ConnectionString` ditekan, tekan tombol *”build”* pada *form* tersebut. Kemudian akan muncul *form* seperti Gambar 4.3. Setelah itu pilih *”Microsoft Jet 4.0 OLE DB Provider”*.



Gambar 4.3. OLE DB Provider

Setelah ditekan, akan muncul *form* seperti pada Gambar 4.4. .Kemudian pilih pada *”Test Connection”*. Bila muncul *message dialog* *”Test connection succeeded”* seperti pada Gambar 4.4., maka koneksi *database* sudah betul



Gambar 4.4. Data Link Properties

## 4.2. Implementasi Program

Pada Form Main terdapat 8 menu dan dari masing masing menu memiliki beberapa submenu. Masing-masing submenu menampilkan form tertentu. Tabel 4.1. akan menampilkan tiap submenu berasal dari form yang mana.

Tabel 4.1. Tabel Menu dari form main

Nama Menu	Sub Menu	Nama Form yang Digunakan
Menu	Login	FormLogin
	LogOut	
	Keluar	
Master	Supplier	FormSupplier
	Customer	FormCustomer
	Barang	FormBarang
	Mesin	FormMesin
	Tenaga Kerja	FormTenaKer

Tabel 4.1. Tabel Menu dari form main (lanjutan)

Nama Menu	Sub Menu	Nama Form yang Digunakan
Transaksi	Order Pembelian	FormOPembelian
	Order Penjualan	FormOPenjualan
	Pembelian	FormPembelian
	Penjualan	FormPenjualan
	Penjualan Barang Avalan	FormPAvalan
	Bayar Hutang	FormBayarHutang
	Bayar Piutang	FormBayarPiutang
Proses	Produksi	FormProduksi
Perhitungan	Perkiraan HPP	FormPerkiraanHPP
Setup	Harga	FormSetup
Laporan	Bahan Dasar	FormLaporanStokPerBahanDasar
	Barang Jadi	FormQRStokBarangJadi
	Barang Avalan	FormQRStokBarangAvalan
	Barang Krom	FormQRStokBarangKrom
	Order Pembelian	FormQROrderPembelian
	Order Penjualan	FormQROrderPenjualan
	Pembelian	FormLaporanPembelian
	Penjualan	FormLaporanPenjualan
	Penjualan Barang Avalan	FormLaporanPenjualanBarangAvalan
	Hutang Tiap Supplier	FormLaporanUtangTiapSupplier
	Piutang Tiap Customer	FormLaporanPiutangTiapCustomer
	Laba Rugi Per Bulan	FormLaporanLabaRugi

Tabel 4.2. Tabel *Procedure* dari tiap *form*

Nama Form	Procedure	Segmen
FormMain		
FormLogin		
FormBarang	procedure ClearEdit;	
	function ValidInput:boolean;	
	function ValidInputEdit:boolean;	
	procedure IsiCbSearch;	
	procedure BitBSaveClick(Sender: TObject);	
FormCustomer	procedure ClearEdit;	
	function ValidInput:boolean;	

Tabel 4.2. Tabel *Procedure* dari tiap *form* (lanjutan)

Nama Form	Procedure	Segmen
	procedure IsiCbSearch;	
	procedure BitBESaveClick(Sender: TObject);	
	function ValidInput:boolean;	
	function ValidInputEdit:boolean;	
	procedure IsiCbSearch;	
	procedure BitBESaveClick(Sender: TObject);	
FormSupplier	procedure ClearEdit;	
	function ValidInput:boolean;	
	function ValidInputEdit:boolean;	
	procedure IsiCbSearch;	
	procedure BitBESaveClick(Sender: TObject);	
FormTenaKer	function ValidInputEdit:boolean;	
	procedure BitBSaveClick(Sender: TObject);	
FormMesin	procedure ClearEdit;	
	function ValidInput:boolean;	
	function ValidInputEdit:boolean;	
	procedure IsiCbSearch;	
	procedure BitBSaveClick(Sender: TObject);	
	procedure BitBESaveClick(Sender: TObject);	
FormSetup	procedure BSaveClick(Sender: TObject);	
FormOPenjualan	procedure IsiCbSearchCustomer;	
	procedure IsiCbSearchEdit;	
	function IndexOfSgOrderJual(ABarangID:string):integer;	
	procedure ClearStringgrid;	
	procedure AddToSGOrderJual;	
	procedure BitBSaveClick(Sender: TObject);	4.1
FormOPembelian	procedure IsiCbSearchSupplier;	
	procedure IsiCbSearchEdit;	
	procedure IsiCBSerachOrder;	
	function IndexOfSgOrderBeli(ABarangID:string):integer;	
	procedure ClearStringgrid;	
	procedure AddToSGOrderBeli;	
	procedure BitBSaveClick(Sender: TObject);	
FormPembelian	procedure IsiCbSearchOrder;	
	procedure IsiCbSearchBeli;	
	function IndexOfSgBeli(ABarangID:string):integer;	
	procedure RemoveFromSG;	
	procedure ClearSG;	
	procedure AddToSG;	
	procedure BitBSaveClick(Sender: TObject);	

Tabel 4.2. Tabel *Procedure* dari tiap *form* (lanjutan)

Nama Form	Procedure	Segmen
FormProduksi	procedure IsiCbSearchOrder;	
	procedure BMProsesClick(Sender: TObject);	
	procedure BOProsesClick(Sender: TObject); procedure BTKProsesClick(Sender: TObject);	
	procedure BitBSaveClick(Sender: TObject);	4.2
FormPenjualan	procedure IsiCBSearchJual;	
	procedure IsiCBSearchProduksi;	
	function IndexOfSgJual(ABarangID:string):integer;	
	procedure ClearSG;	
	procedure AddToSG;	
	procedure ClearAll;	
FormPAvalan	procedure BitBSaveClick(Sender: TObject);	
	procedure IsiCBSearchCustomer;	
	procedure IsiCBSearchBarang;	
	function IndexOfSgJual(ABarangID:string):integer;	4.3
	procedure RemoveFromSG;	
	procedure ClearSG;	
FormBayarHutang	procedure AddToSG;	4.4
	procedure BitBSaveClick(Sender: TObject);	
	procedure IsiCBSearch;	
FormBayarPiutang	procedure DGPembelianDbClick(Sender: TObject);	
	procedure BitBSaveClick(Sender: TObject);	
	procedure DGPenjualanDbClick(Sender: TObject);	
FormPerkiraanHPP	procedure Button1Click(Sender: TObject);	4.5

Segmen 4.1. digunakan untuk menyimpan order penjualan yang telah diinputkan jika pemeriksaan yang ada sudah memenuhi syarat. Cara penyimpanannya dengan menyimpan di *database* dengan nama tabel OrderJual.

#### Segmen 4.1. procedure BitBSaveClick(Sender: TObject);

```

procedure TFormOPenjualan.BitBSaveClick(Sender: TObject);
var
  i:integer;
begin
  if SGOrderJual.RowCount<>2 then
  begin
    with Q do begin
      close;
      sql.Clear;
    end;
  end;
end;

```

#### Segmen 4.1. procedure BitBSaveClick (lanjutan)

```

sql.Text:='Insert into OrderJual([OrderJualID],[BarangID],
[CustomerID],[TglOrderJual],[TglShipment],[Jumlah],[Harga],
[Deleted])values (:ojid,:bi,:cid,:toj,:ts,:j,:h,:del)';
With parameters do begin
  ParamByName('ojid').Value:=EID.Text;
  ParamByName('bi').Value:=DTID.Caption;
  ParamByName('cid').Value:=LID.Text;
  ParamByName('toj').Value:=DToJual.DateTime;
  ParamByName('ts').Value:=DTSD.DateTime;
  ParamByName('j').Value:=StrToInt(EJumlah.Text);
  ParamByName('h').Value:=StrToCurr(EHarga.Text);
  ParamByName('del').Value:=false;
end;
ExecSQL;

//update status order jual
sql.Clear;
sql.Text:='update OrderJual set status=:s where
OrderJualID=:oji';
with Parameters do begin
  for i:=1 to SGOrderJual.RowCount-2 do begin
    ParamByName('oji').Value:=EID.Text;
    ParamByName('s').value:='Active';
    ExecSQL;
  end;
end;
end;
ShowMessage('Pemesanan penjualan berhasil');
ClearAll;
end else ShowMessage('Tidak ada pemesanan pembelian');
end;

```

Segmen 4.2. digunakan menyimpan data-data yang telah diinputkan ke dalam database pada tabel produksi serta melakukan update status pada pembelian dan juga pada order penjualan. Setelah itu juga memasukkan data-data pada detail-detail pada produksi.

#### Segmen 4.2. procedure BitBSaveClick(Sender: TObject);

```

procedure TFormProduksi.BitBSaveClick(Sender: TObject);
var
  totaldetail,i,j:integer;
  temp,hpp,g1,g2,g3,g4,btk,beratkarung:currency;
begin
  totaldetail:=0;
  if EOPID.Text='' then begin
    EOPID.SetFocus;
    ShowMessage('Order Penjualan ID masih kosong');
    exit;
  end;
end;

```

## Segmen 4.2. procedure BitBSaveClick (lanjutan)

```

if ESusut.Text='' then begin ESusut.SetFocus;
  ShowMessage('Penyusutan masih kosong');
  exit;
end;
if EBeras.Text='0' then begin
  EBeras.SetFocus;
end;
if EBekatul.Text='0' then begin
  EBekatul.SetFocus;
  ShowMessage('Jumlah Bekatul masih kosong');
  exit;
end;
if MessageDlg('Apakah anda yakin akan memasukkan data produksi',
mtConfirmation,[mbYes,mbNo], 0)= mrYes then begin
  //insert ke produksi
  with Q do
  begin
    close;
    sql.Clear;
    sql.Text:='Insert into
Produksi([ProduksiID],[BeliID],[OrderBeliID],[OrderJualID],
[TglProduksi],[Penyusutan],[JBeras],[JSekam],[JMenir],
[JBekatul],[Status])values
(:pid,:bi,:obid,:ojid,:tp,:ps,:jb,:js,:jm,:jbek,:st)';
    with parameters do
    begin
      ParamByName('pid').Value:=EPID.Text;
      ParamByName('bi').Value:=EBID.Text;

ParamByName('obid').Value:=QOrder.FieldValues['OrderBeliID'];
      ParamByName('ojid').Value:=EOPID.Text;
      ParamByName('tp').Value:=DTProduksi.DateTime;
      ParamByName('ps').Value:=ESusut.Text;
      ParamByName('jb').Value:=EBeras.Text;
      ParamByName('js').Value:=ESekam.Text;
      ParamByName('jm').Value:=EMenir.Text;
      ParamByName('jbek').Value:=EBekatul.Text;
      ParamByName('st').Value:=false;
    end;
    ExecSQL;
  end;

  //insert ke detail produksi barang
  with Q do
  begin
    Close;
    sql.Clear;
    sql.Text:='Insert into
DetailProduksiBarang([ProduksiID],[BarangID],[Berat],[Harga])
Values (:pid,:bi,:b,:h)';
    with parameters do
    begin
      ParamByName('pid').Value:=EPID.Text;
      ParamByName('bi').Value:=QOrder.FieldValues['b.BarangID'];

```

## Segmen 4.2. procedure BitBSaveClick (lanjutan)

```

        ParamByName('b').Value:=StrToFloat(EBerat.Text);
        ParamByName('h').Value:=StrToCurr(EHarga.Text);
    end;
    ExecSQL;
end;

//insert ke detail produksi tenaga kerja
if CBATurun.Checked then
begin
    with Q do
    begin
        close;
        sql.Clear;
        sql.Text:='select * from TenaKer where TenakerID=:tid';
        Parameters.ParamByName('tid').Value:='TK01';
        open;
        with Q1 do
        begin
            close;
            sql.Clear;
            sql.Text:='Insert into
            DetailProduksiTenaKer([ProduksiID],[TenaKerID],[Gaji])
            values (:pid,:tkid,:g)';
            with parameters do
            begin
                ParamByName('pid').Value:=EPID.Text;
                ParamByName('tkid').Value:=q.FieldValues['TenaKerID'];
                ParamByName('g').Value:=Q.FieldValues['Gaji'];
            end;
            ExecSQL;
            g1:=q.FieldValues['Gaji'];
        end;
    end;
end;
if CBJemur.Checked then
begin
    with Q do
    begin
        close;
        sql.Clear;
        sql.Text:='select * from TenaKer where TenakerID=:tid';
        Parameters.ParamByName('tid').Value:='TK02';
        open;
        with Q1 do
        begin
            close;
            sql.Clear;
            sql.Text:='Insert into
            DetailProduksiTenaKer([ProduksiID],[TenaKerID],[Gaji])
            values (:pid,:tkid,:g)';
            with parameters do
            begin
                ParamByName('pid').Value:=EPID.Text;
                ParamByName('tkid').Value:=Q.FieldValues['TenaKerID'];
                ParamByName('g').Value:=Q.FieldValues['Gaji'];
            end;
        end;
    end;
end;
end;

```

## Segmen 4.2. procedure BitBSaveClick (lanjutan)

```

        ExecSQL;
        g2:=q.FieldValues['Gaji'];
    end;
end;
end;
if CBGiling.Checked then
begin
    with Q do
    begin
    close;
    sql.Clear;
    sql.Text:='select * from TenaKer where TenakerID=:tid';
    Parameters.ParamByName('tid').Value:='TK03';
    open;
    with Q1 do
    begin
    close;
    sql.Clear;

    sql.Text:='Insert into
    DetailProduksiTenaKer([ProduksiID],[TenaKerID],[Gaji])
    Values (:pid,:tkid,:g)';
    with parameters do
    begin
        ParamByName('pid').Value:=EPID.Text;
        ParamByName('tkid').Value:=q.FieldValues['TenaKerID'];
        ParamByName('g').Value:=Q.FieldValues['Gaji'];
    end;
    ExecSQL;
    g3:=q.FieldValues['Gaji'];
    end;
    end;
end;
if CBANaik.Checked then
begin
    with Q do
    begin
    close;
    sql.Clear;
    sql.Text:='select * from TenaKer where TenakerID=:tid';
    Parameters.ParamByName('tid').Value:='TK04';
    open;
    with Q1 do
    begin
    close;
    sql.Clear;
    sql.Text:='Insert into
    DetailProduksiTenaKer([ProduksiID],[TenaKerID],[Gaji])
    values (:pid,:tkid,:g)';
    with parameters do
    begin
        ParamByName('pid').Value:=EPID.Text;
        ParamByName('tkid').Value:=q.FieldValues['TenaKerID'];
        ParamByName('g').Value:=Q.FieldValues['Gaji'];
    end;
    ExecSQL;

```

## Segmen 4.2. procedure BitBSaveClick (lanjutan)

```

    g4:=q.FieldValues['Gaji'];
  end;
end;
end;

//insert ke detail produksi overhead

with Q do begin
  close;
  sql.Clear;
  sql.Text:='select mesinid ,sum(harga/jangkawkt) as totalbagi
from mesin where Deleted=false group by mesinid';
  open;
  for i:=0 to recordcount-1 do begin
    temp:=q.FieldValues['totalbagi']/365/8*
    StrToInt(ELama.Text);
    with q1 do begin
      close;
      sql.Clear;
      sql.Text:='Insert into
DetailProduksiOH([ProduksiID],[MesinID],[Lama],[Nama],
[Biaya])values (:pid,:mid,:l,:n,:b)';
      with parameters do begin
        ParamByName('pid').Value:=EPID.Text;
        ParamByName('mid').Value:=q.FieldValues['MesinID'];
        ParamByName('l').Value:=ELama.Text;
        ParamByName('n').Value:='Penyusutan Mesin';
        ParamByName('b').Value:=temp;
      end;
      ExecSQL;
    end;
    Next;
  end;
end;

with q do begin
  close;
  sql.Clear;
  sql.Text:='Insert into
DetailProduksiOH([ProduksiID],[MesinID],[Lama],[Nama],
[Biaya]) values (:pid,:mid,:l,:n,:b)';
  with parameters do begin
    ParamByName('pid').Value:=EPID.Text;
    ParamByName('mid').Value:='';
    ParamByName('l').Value:=0;
    ParamByName('n').Value:='Solar';
    ParamByName('b').Value:=StrToCurr(ESolar.Text)
    *60/10000*StrToInt(EBerat.Text);
  end;
  ExecSQL;
end;
beratkarung:=StrToCurr(EBerat.Text)/50;
beratkarung:=RoundTo(beratkarung,1)*StrToCurr(EKarung.Text);
with q do begin
  close;
  sql.Clear;

```

## Segmen 4.2. procedure BitBSaveClick (lanjutan)

```

        sql.Text:='Insert into
        DetailProduksiOH([ProduksiID],[MesinID],[Lama],[Nama],
        [Biaya]) values (:pid,:mid,:l,:n,:b)';
with parameters do begin
    ParamByName('pid').Value:=EPID.Text;
    ParamByName('mid').Value:='';
    ParamByName('l').Value:=0;
    ParamByName('n').Value:='Packaging';
    ParamByName('b').Value:=beratkarung;
end;
    ExecSQL;
end;
//update status order jual
with q do
begin
    close;
    sql.Clear;
    sql.Text:='update OrderJual set status=:st where
    OrderJualID=:ojid';
    with Parameters do
    begin
        ParamByName('st').Value:='Produksi';
        ParamByName('ojid').Value:=EOPID.Text;
    end;
    ExecSQL;
end;

//update status pembelian
with q do
begin
    close;
    sql.Clear;
    sql.Text:='update Pembelian set Status=true
    where BeliID=:bi';
    Parameters.ParamByName('bi').Value:=EBID.Text;
    ExecSQL;
end;

//update jumlah barang pada Barang

//Padi
with q do
begin
    close;
    sql.Clear;
    sql.Text:='select * from Barang where BarangID=:bid';
    Parameters.ParamByName('bid').Value:='BM001';
    open;
    with q1 do
    begin
        close;
        sql.Clear;
        sql.Text:='update Barang set Jumlah=:j
        where BarangID=:bi';
        Parameters.ParamByName('j').Value:=q.
        FieldValues['Jumlah']-StrToFloat(EBerat.Text);
    end;
end;

```

## Segmen 4.2. procedure BitBSaveClick (lanjutan)

```

        Parameters.ParamByName('bi').Value:='BM001';
        ExecSQL;
    end;
end;
//Beras
with q do
begin
    close;
    sql.Clear;
    sql.Text:='select * from Barang where BarangID=:bid';
    Parameters.ParamByName('bid').Value:='BJ001';
    open;
    with ql do
    begin
        close;
        sql.Clear;
        sql.Text:='update Barang set Jumlah=:j
        where BarangID=:bi';
        Parameters.ParamByName('j').Value:=q.FieldValues
        ['Jumlah']+StrToFloat(EBeras.Text);
        Parameters.ParamByName('bi').Value:='BJ001';
        ExecSQL;
    end;
end;

//sekam
with q do
begin
    close;
    sql.Clear;
    sql.Text:='select * from Barang where BarangID=:bid';
    Parameters.ParamByName('bid').Value:='BA001';
    open;
    with ql do
    begin
        close;
        sql.Clear;
        sql.Text:='update Barang set Jumlah=:j
        where BarangID=:bi';
        Parameters.ParamByName('j').Value:=q.FieldValues
        ['Jumlah']+StrToFloat(ESekam.Text);
        Parameters.ParamByName('bi').Value:='BA001';
        ExecSQL;
    end;
end;

//menir
with q do
begin
    close;
    sql.Clear;
    sql.Text:='select * from Barang where BarangID=:bid';
    Parameters.ParamByName('bid').Value:='BA002';
    open;
    with ql do
    begin

```

### Segmen 4.2. procedure BitBSaveClick (lanjutan)

```

close;
sql.Clear;
sql.Text:='update Barang set Jumlah=:j
where BarangID=:bi';
    Parameters.ParamByName('j').Value:=q.FieldValues
    ['Jumlah']+StrToFloat (EMenir.Text);
    Parameters.ParamByName('bi').Value:='BA002';
    ExecSQL;
end;
end;

//bekatul
with q do
begin
    close;
    sql.Clear;
    sql.Text:='select * from Barang where BarangID=:bid';
    Parameters.ParamByName('bid').Value:='BA003';
    open;
    with ql do
    begin
        close;
        sql.Clear;
        sql.Text:='update Barang set Jumlah=:j where
        BarangID=:bi';
        Parameters.ParamByName('j').Value:=q.FieldValues['Jumlah']
        +StrToFloat(EBekatul.Text);
        Parameters.ParamByName('bi').Value:='BA003';
        ExecSQL;
    end;
end;
end;
ShowMessage('Proses Produksi Berhasil');
end;
ClearAll;
end;• ShowMessage('Jumlah Beras masih kosong');
exit;

```

Segmen 4.3. adalah segmen yang digunakan untuk melihat index dari *string grid* pada SGJual. Jika *string grid* masih kosong, maka hasil yang dikirimkan adalah -1.

### Segmen 4.3. function IndexOfSgJual(ABarangID:string):integer;

```

function TFormPAvalan.IndexOfSgJual(ABarangID: string): integer;
var
    i:integer;
begin
    result:=-1;
    for i:=1 to SGJual.RowCount-1 do begin
        if SGJual.Cells[0,i]=ABarangID then
            begin
                result:=i;
                exit;
            end;
    end; end; end;

```

Segmen 4.4. adalah segmen yang digunakan untuk memasukkan data yang telah dipilih ke dalam *string grid*. Dalam prosedur ini juga diperiksa apakah string grid masih kosong atau telah terisi. jika terisi akan diisi ke bawahnya.

#### Segmen 4.4. procedure AddToSG;

```

procedure TFormPAvalan.AddToSG;
var
  i,AIndex:integer;
  tot:Currency;
begin
  if EBID.Text='' then begin
    EBID.SetFocus;
    ShowMessage('Barang ID masih kosong');
    exit;
  end;
  if EBerat.Text='0' then begin
    EBerat.SetFocus;
    ShowMessage('Berat masih kosong');
    exit;
  end;
  if EBerat.Text>QBarang.FieldValues['Jumlah'] then begin
    EBerat.SetFocus;
    ShowMessage('Berat tidak mencukupi');
    exit;
  end;

  tot:=0;
  with SGJual do begin
    AIndex:=IndexOfSgJual(QBarang.FieldValues['BarangID']);
    if AIndex=-1 then begin
      rowcount := rowcount+1;
      cells[0,rowcount-2]:=QBarang.FieldValues['BarangID'];
      cells[1,rowcount-2]:=QBarang.FieldValues['Name'];
      cells[2,rowcount-2]:=EBerat.Text;
      cells[3,rowcount-2]:=EHarga.Text;
      cells[4,rowcount-2]:=CurrToStr(StrToInt(EBerat.Text)
        *StrToCurr (EHarga.Text));
    end else begin
      cells[2,AIndex]:=EBerat.Text;
      cells[3,AIndex]:=EHarga.text;
      cells[4,AIndex]:=CurrToStr(StrToInt(EBerat.Text)*
        StrToCurr (EHarga.Text));
    end;
    for i:=1 to RowCount-2 do
      begin
        tot:=tot+StrToCurr(Cells[4,i]);
      end;
    end;
    LTotal.Caption:=CurrToStr(tot);
  end;
end;

```

Segmen 4.5. adalah segmen yang digunakan untuk melakukan penghitungan hpp dari data-data yang telah diinputkan. Untuk bagian-bagian tertentu, harga telah otomatis diinputkan dengan cara mengambil dari database.

Segmen 4.5. procedure Button1Click(Sender: TObject);

```

procedure TFormPerkiraanHPP.Button1Click(Sender: TObject);
var
  i,lama:integer;
  g1,g2,g3,g4,bm,btk,boh,temp,total,hpp:currency;
  beras,ks,susut,padi,jk:Double;
begin
  lama:=6;
  with q do
  begin
    close;
    sql.Clear;
    sql.Text:='select * from TenaKer where TenaKerID="TK01"';
    open;
    g1:=q.FieldValues['Gaji'];
    close;
    sql.Clear;
    sql.Text:='select * from TenaKer where TenaKerID="TK02"';
    open;
    g2:=q.FieldValues['Gaji'];
    close;
    sql.Clear;
    sql.Text:='select * from TenaKer where TenaKerID="TK03"';
    open;
    g3:=q.FieldValues['Gaji'];
    close;
    sql.Clear;
    sql.Text:='select * from TenaKer where TenaKerID="TK04"';
    open;
    g4:=q.FieldValues['Gaji'];
    bm:=StrToInt(EBerat.Text)*StrToCurr(EHarga.Text);
    if strtoint(EAir.Text)<14 then
    begin
      ShowMessage('Padi terlalu kering');
      exit;
    end else
    begin
      ks:=(StrToInt(EAir.Text)-14)*13/10;
      susut:=ks*StrToInt(EBerat.Text)/100;
      padi:=StrToInt(EBerat.Text)-susut;
      beras:=56*StrToInt(EBerat.Text)/100;
      jk:=beras/50;
      btk:=StrToInt(EBerat.Text)*g1+StrToInt(EBerat.Text)*g2+
      StrToInt(EBerat.Text)*g3+beras*g4;
      total:=0;
      temp:=0;
      with Q do begin
        Connection:=DataModule1.ADOConnection;
        close;

```

## Segmen 4.5. procedure Button1Click (lanjutan)

```
sql.Clear;
sql.Text:='select mesinid ,sum(harga/jangkawkt) as
totalbagi from mesin group by mesinid';
open;
for i:=0 to recordcount-1 do begin
  temp:=q.FieldValues['totalbagi']/365/8*lama;
  total:=total+temp;
  Next;
end;
boh:=(total+(StrToInt(ESolar.Text)*0.006*padi+
StrToCurr(EKarung.Text)*jk));
end;
hpp:=(bm+btk+boh)/beras;
LHPP.Caption:=CurrToStr(hpp);
end; end;end;
```