

# VERİ TABANI YÖNETİM SİSTEMLERİ FİNAL PROJESİ

PROJE ADI: ŞİRKET OTOMASYOMU

15060007 Kübra Nur SARUHAN

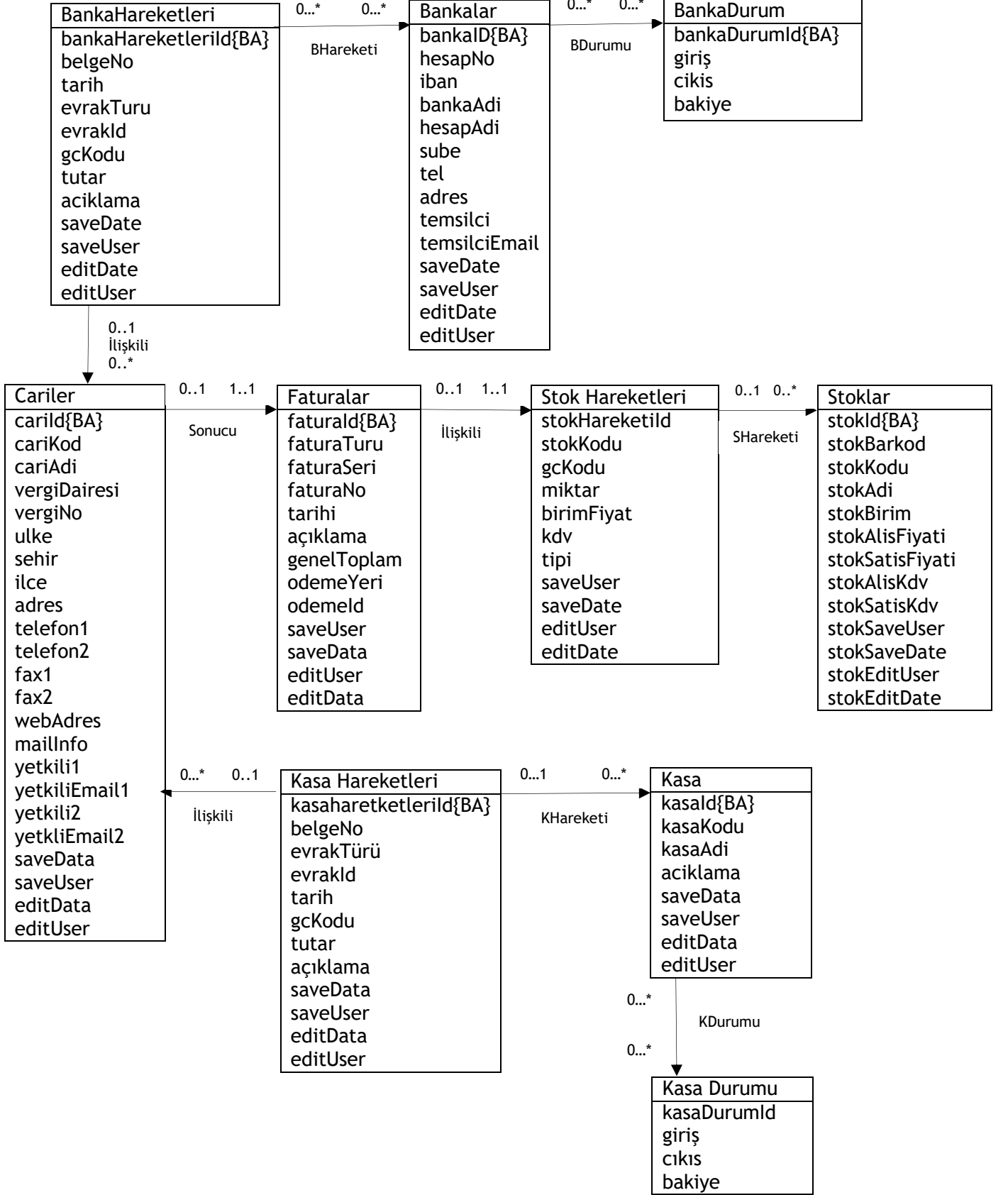
15060765 Ebubekir DOĞAN

# PROJE SENARYOSU

Bir şirket veri tabanı oluşturulması isteniyor. Aşağıda belirtilen açıklamalar ışığında (açıklamalarda yer almayan konular için uygun varsayımlar gerekiyorsa, uygun varsayımlar yaparak) hizmet üreten bir otomasyon tasarlanmamız isteniyor.

- Her yeni gelen stok grupları için grup kodunun, adının, kaydedildiği tarihi vardır.
- Her stok kodun ve adının, grup kodu ve adının, alış satış fiyatının, alış satış KDV'lerinin, birim barkodu vardır.
- Her stok hareketi için gc kodu, miktarı, birim fiyatı, kdv ve tipi vardır.
- Her cari grubu için grup adının, kodunun, kaydedildiği tarihi vardır.
- Her carinin kodu, adı, vergi dairesi, vergi numarası, ülkesi, şehri, ilçesi, adresi, telefon1, telefon2, fax1, fax2, web adresi, maili, yetkili1, yetkili email1, yetkili2, yetkili email2, kaydedildiği tarih vardır.
- Her bankanın adı, şubesi, adresi, telefonu, temsilci, temsilci email, hesap numara, hesap adı, ibanı, kaydedildiği tarih vardır.
- Her bankandan işlem yapıldığı giriş, çıkış ve bakiye kaydedilir.
- Her banka hareketi için banka numarası, belge numarası, tarihi, evrak türü, evrak numarası, gc kodu, tutar, açıklama ve kaydedildiği tarih vardır.
- Her faturanın numarası, türü, tarihi, irsaliye numarası, irsaliye tarihi, açıklama, genel toplam, ödeme yeri, ödeme çeşidi, kaydedildiği tarih vardır.
- Her kasanın kodu, adı, açıklaması ve kaydedildiği tarih vardır.
- Her kasa hareketinin kasa numarası, cari numarası, belge numarası, evrak türü, evrak numarası, tarih, gc kodu, tutarı, açıklaması, ve kaydedildiği tarih vardır.
- Her kasadan giriş çıkış ve bakiyesi vardır.

# UML DİYAGRAMI



# İLİŞKİ ŞEMALARI

Bankalar (bankaID, hesapNo, iban, bankaAdi, hesapAdi, sube, tel, adres, temsilci, temsilciEmail, saveDate, saveUser, editDate, editUser)

BankaDurum (bankaDurumId, giriş, cikis, bakiye)

BDurumu (bankaID, bankaDurumId)

BankaHareketleri (bankaHareketlerId, belgeNo, tarih, evrakTuru, evrakId, gcKodu, tutar, aciklama, saveDate, saveUser, editDate, editUser)

BHareketi (bankaID, bankaHareketlerId)

Cariler (carId, cariKod, cariAdi, vergiDairesi, vergiNo, ulke, sehir, ilce, adres, telefon1, telefon2, fax1, fax2, webAdres, mailInfo, yetkili1, yetkiliEmail1, yetkili2, yetkliEmail2, saveData, saveUser, editData, editUser, bankaHareketlerId, kasaharetketlerId)

Faturalar (faturald, faturaTuru, faturaSeri, faturaNo, tarihi, açıklama, genelToplam, ödemeYeri, odemeld, saveUser, saveData, editUser, editData, carId)

StokHareketleri (stokHareketId, stokKodu, gcKodu, miktar, birimFiyat, kdv, tipi, saveUser, saveDate, editUser, editDate, faturald)

Stoklar (stokId, stokBarkod, stokKodu, stokAdi, stokBirim, stokAlisFiyati, stokSatisFiyati, stokAlisKdv, stokSatisKdv, stokSaveUser, stokSaveDat, stokEditUser, stokEditDate, stokHareketId)

Kasa Hareketleri (kasaharetketlerId, belgeNo, evrakTürü, evrakId, tarih, gcKodu, tutar, aciklama, saveData, saveUser, editData, editUser)

Kasa (kasald, kasaKodu, kasaAdi, aciklama, saveData, saveUser, editData, editUser, kasaharetketlerId)

KasaDurumu (kasaDurumId, giriş, cıkıs, bakiye)

KDurumu (kasald, kasaDurumId)


# OTOMASYON TASARIMI


İlk olarak login girişimiz açılıyor. Kullanıcının girmiş olduğu ad ve şifreye göre programımızda yönetici veya normal olarak giriş yapıyor. Normal kullanıcıların kullanıcı yönetimine erişimine izni bulunmaktadır.

Login

✕

User Login





Ana Form

1 2 3 4 5 6

Stok Kartı Stok Listesi Stok Grupları

Stok İşlemleri

Stok

Stok Kartı

Stok Listesi

Stok Grupları

1) Kullanıcı Yönetim

2) Menüler

3) Alt Menüler

4) Sol Açılır Menü

5) MasterPage

6) Aktif Kullanıcı Bilgisi

Stok

Cari

Banka

Kasa

Fatura

Genel Otomasyon Yazılımı Kullanıcı : admin Çıkış Yap 02.07.2016

Tasarım olarak sadece stok bölümlerini raporumuzda kullanıyoruz. Diğer bölümlerdeki tasarımlarımız benzerdir.

## 1)Stok Gruplar

Stok grupları yeni bir kategorimiz geldiğinde eklememiz gereken bölümdür. Stok grup ve kodunu kaydettiğimiz yerdir.

The screenshot shows the 'Stok Listesi - Ana Form' window. On the left, there is a sidebar with icons for 'Stok', 'Cari', 'Banka', 'Kasa', and 'Fatura'. The main area displays a list of stock items with columns for 'STOK KODU' and 'BARKOD'. A 'Stok Grupları' dialog box is open in the center, allowing users to add new stock groups. The dialog box has a 'Grup Bilgisi' section with fields for 'Grup Kodu' and 'Grup Adı'. Below this, there is a table with columns 'GRUP KODU' and 'GRUP ADI' containing the following data:

GRUP KODU	GRUP ADI
00026	İçecek
1234	Giyim

At the bottom of the dialog box, there are buttons for 'Kaydet' (Save), 'Sil' (Delete), and 'Kapat' (Close). The background window also has a search bar and buttons for 'Ara', 'Temizle', 'Listeyi Yazdır (PDF)', and 'Listeyi Yazdır (XLS)'. The status bar at the bottom indicates 'Genel Otomasyon Yazılımı', 'Kullanıcı : admin', 'Çıkış Yap', and the date '02.07.2016'.

## 2) Stok Açılış Kartı

Stok açılış kartımız; stok grubumuza ait yeni ve farklı ürün bilgilerini kaydettiğimiz yerdir. Örneğin grup adı giyim kategorisi olan ürünlerin çocuk giyim için bilgilerini kaydedebileceğimiz yerdir.

The screenshot shows the 'Stok Açılış Kartı' dialog box. It has a 'Ana Stok Bilgileri' section with fields for 'Stok Kodu' (containing '0000002') and 'Stok Adı'. There is a 'Stok Resim Seç' button next to the 'Stok Adı' field. Below this, there is a 'Genel Stok Bilgileri' section with two columns: 'Grup Bilgileri' and 'Fiyat Bilgileri'. The 'Grup Bilgileri' column has fields for 'Grup Kodu' and 'Grup Adı'. The 'Fiyat Bilgileri' column has fields for 'Alış Fiyatı' and 'Satış Fiyatı'. Below these, there is a 'Barkod ve Birim' section with a dropdown for 'Birim' (set to 'ADET') and a field for 'Barkod'. To the right of this is a 'KDV Bilgileri (%)' section with fields for 'Alış KDV' and 'Satış KDV'. At the bottom of the dialog box, there are buttons for 'Kaydet' (Save), 'Sil' (Delete), and 'Kapat' (Close). The background window shows the same sidebar and main area as the first screenshot. The status bar at the bottom indicates 'Genel Otomasyon Yazılımı', 'Kullanıcı : admin', 'Çıkış Yap', and the date '02.07.2016'. A small message box in the bottom right corner says 'Stok Açılış Kartı Formu Açıldı.'

### 3)Stok Listesi

Stok açılış kartı ile kaydettiğimiz bilgileri listeleyebileceğimiz yerdir. Stok kodu, stok adı veya barkod kısımlarından istediğimizi kullanarak listede arama yapabiliriz.

### Database İşlemleri İçin Oluşturduğunuz Sınıflarımız

Sadece stok bölümündeki sınıflarını raporumuzda belirtiyoruz. Diğer bölümlerdeki sınıflarımız ile benzerdir.

#### 1) Stok Grupları Sınıfları

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Data.Linq;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using DevExpress.XtraEditors;
```

```
namespace Otomasyon.Modul_Stok
{
```

```
    public partial class DaoStokGruplari : DevExpress.XtraEditors.XtraForm
    {
```

```
        Fonksiyonlar.DatabaseDataContext DB = new Fonksiyonlar.DatabaseDataContext();
```

```

Fonksiyonlar.Mesajlar Mesajlar = new Fonksiyonlar.Mesajlar();
public bool Secim = false;
int SecimID = -1;
bool Edit = false;

public DaoStokGruplari()
{
    InitializeComponent();
}

private void frmStokGruplari_Load(object sender, EventArgs e)
{
    Listele();
}

void Listele()
{
    var lst = from s in DB.TBL_STOKGRUPLARIs
              select s;
    Liste.DataSource = lst;
}

void Temizle()
{
    txtGrupAdi.Text = "";
    txtGrupKodu.Text = "";
    Edit = false;
    Listele();
}

void YeniKaydet()
{
    try
    {
        Fonksiyonlar.TBL_STOKGRUPLARI Grup = new
Fonksiyonlar.TBL_STOKGRUPLARI();
        Grup.GRUPADI = txtGrupAdi.Text;
        Grup.GRUPKODU = txtGrupKodu.Text;
        Grup.GRUPSAVEDATE = DateTime.Now;
        Grup.GRUPSAVEUSER = AnaForm.UserID;
        DB.TBL_STOKGRUPLARIs.InsertOnSubmit(Grup);
        DB.SubmitChanges();
        Mesajlar.YeniKayit("Yeni Grup Kaydı Oluşturuldu.");
        Temizle();
    }
    catch (Exception e)
    {
        Mesajlar.Hata(e);
    }
}

void Guncelle()
{

```



```

        try
        {
            Fonksiyonlar.TBL_STOKGRUPLARI Grup = DB.TBL_STOKGRUPLARIs.First(s => s.ID
== SecimID);
            Grup.GRUPKODU = txtGrupKodu.Text;
            Grup.GRUPADI = txtGrupAdi.Text;
            Grup.GRUPEDITUSER = AnaForm.UserID;
            Grup.GRUPEDITDATE = DateTime.Now;
            DB.SubmitChanges();
            Mesajlar.Guncelle(true);
            Temizle();
        }
        catch (Exception e)
        {
            Mesajlar.Hata(e);
        }
    }

    void Sil()
    {
        try
        {
            DB.TBL_STOKGRUPLARIs.DeleteOnSubmit(DB.TBL_STOKGRUPLARIs.First(s => s.ID
== SecimID));
            DB.SubmitChanges();
            Listele();
        }
        catch (Exception e)
        {
            Mesajlar.Hata(e);
        }
    }

    private void BtnKapat_Click(object sender, EventArgs e)
    {
        this.Close();
    }

    private void BtnSil_Click(object sender, EventArgs e)
    {
        if (Edit && SecimID > 0 && Mesajlar.Sil() == DialogResult.Yes) Sil();
    }

    private void BtnKaydet_Click(object sender, EventArgs e)
    {
        if (Edit && SecimID > 0 && Mesajlar.Guncelle() == DialogResult.Yes) Guncelle();
        else YeniKaydet();
    }

    void Sec()
    {
        try
        {

```

```

        Edit = true;
        SecimID = int.Parse(gridView1.GetFocusedRowCellValue("ID").ToString());
        txtGrupAdi.Text = gridView1.GetFocusedRowCellValue("GRUPADI").ToString();
        txtGrupKodu.Text =
gridView1.GetFocusedRowCellValue("GRUPKODU").ToString();
    }
    catch (Exception)
    {
        Edit = false;
        SecimID = -1;
    }
}
private void Liste_DoubleClick(object sender, EventArgs e)
{
    Sec();
    if(Secim && SecimID > 0)
    {
        AnaForm.Aktar = SecimID;
        this.Close();
    }
}
}
}
}

```

## 2) Stok Açılış Kartı Sınıfları

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Linq;
using System.Threading.Tasks;
using System.Windows.Forms;
using DevExpress.XtraEditors;

namespace Otomasyon.Modul_Stok
{
    public partial class DaoStokKarti : DevExpress.XtraEditors.XtraForm
    {
        Fonksiyonlar.DatabaseDataContext DB = new Fonksiyonlar.DatabaseDataContext();
        Fonksiyonlar.Mesajlar Mesajlar = new Fonksiyonlar.Mesajlar();
        Fonksiyonlar.Numara Numaralar = new Fonksiyonlar.Numara();
        Fonksiyonlar.Formlar Formlar = new Fonksiyonlar.Formlar();
        Fonksiyonlar.Resimleme Resimleme = new Fonksiyonlar.Resimleme();

        bool Edit = false;
        bool Resim = false;
        OpenFileDialog Dosya = new OpenFileDialog();
        int GrupID = -1;
        int StokID = -1;
    }
}

```

```

public DaoStokKarti()
{
    InitializeComponent();
}

private void BtnKapat_Click(object sender, EventArgs e)
{
    this.Close();
}

private void frmStokKarti_Load(object sender, EventArgs e)
{
    txtStokKodu.Text = Numaralar.StokKodNumarasi();
    Mesajlar.FormAcilis(this.Text);
}

void Temizle()
{
    txtStokKodu.Text = Numaralar.StokKodNumarasi();
    txtStokAdi.Text = "";
    txtSatisFiyat.Text = "0";
    txtSatisKdv.Text = "0";
    txtGrupKodu.Text = "";
    txtGrupAdi.Text = "";
    txtBirim.SelectedIndex = 0;
    txtBarkod.Text = "";
    txtAlisKdv.Text = "0";
    txtAlisFiyat.Text = "0";
    Edit = false;
    Resim = false;
    GrupID = -1;
    StokID = -1;
    AnaForm.Aktar = -1;
    pictureBox1.Image = null;
}

void ResimSec()
{
    Dosya.Filter = "Jpg(*.jpg)|*.jpg|Jpeg(*.jpeg)|*.jpeg";
    if(Dosya.ShowDialog() == DialogResult.OK)
    {
        pictureBox1.ImageLocation = Dosya.FileName;
        Resim = true;
    }
}

private void btnResimSec_Click(object sender, EventArgs e)
{
    ResimSec();
}

public void Ac(int ID)
{

```

```

Edit = true;
StokID = ID;
Fonksiyonlar.TBL_STOKLAR Stok = DB.TBL_STOKLARs.First(s => s.ID == StokID);
GrupAc(Stok.STOKGRUPID.Value); //value olarak değeri aldık.
if (Stok.STOKRESIM != null) pictureBox1.Image =
Resimleme.ResimGetirme(Stok.STOKRESIM.ToArray()); //array olarak resimleri alıyoruz.
else pictureBox1.Image = null;
txtAlisFiyat.Text = Stok.STOKALISFIYAT.ToString();
txtAlisKdv.Text = Stok.STOKALISKDV.ToString();
txtBarkod.Text = Stok.STOKBARKOD;
txtBirim.Text = Stok.STOKBIRIM;
txtSatisFiyat.Text = Stok.STOKSATISFIYAT.ToString();
txtSatisKdv.Text = Stok.STOKSATISKDV.ToString();
txtStokAdi.Text = Stok.STOKADI;
txtStokKodu.Text = Stok.STOKKODU;
}

void GrupAc(int ID)
{
GrupID = ID;
txtGrupAdi.Text = DB.TBL_STOKGRUPLARIs.First(s => s.ID == GrupID).GRUPADI;
txtGrupKodu.Text = DB.TBL_STOKGRUPLARIs.First(s => s.ID == GrupID).GRUPKODU;
}

void YeniKaydet()
{
try
{
Fonksiyonlar.TBL_STOKLAR Stok = new Fonksiyonlar.TBL_STOKLAR();
Stok.STOKADI = txtStokAdi.Text;
Stok.STOKALISFIYAT = decimal.Parse(txtAlisFiyat.Text);
Stok.STOKALISKDV = decimal.Parse(txtAlisKdv.Text);
Stok.STOKBARKOD = txtBarkod.Text;
Stok.STOKBIRIM = txtBirim.Text;
Stok.STOKGRUPID = GrupID;
Stok.STOKKODU = txtStokKodu.Text;
if (Resim)Stok.STOKRESIM = new
System.Data.Linq.Binary(Resimleme.ResimYukleme(pictureBox1.Image));
Stok.STOKSATISFIYAT = decimal.Parse(txtSatisFiyat.Text);
Stok.STOKSATISKDV = decimal.Parse(txtSatisKdv.Text);
Stok.STOKSAVEDATE = DateTime.Now;
Stok.STOKSAVEUSER = AnaForm.UserID;
DB.TBL_STOKLARs.InsertOnSubmit(Stok);
DB.SubmitChanges();
Mesajlar.YeniKayit("Yeni Stok Kaydı Oluşturuldu.");
Temizle();
}
catch (Exception e)
{
Mesajlar.Hata(e);
}
}

```

```

    }

    void Guncelle()
    {
        try
        {
            Fonksiyonlar.TBL_STOKLAR Stok = DB.TBL_STOKLARs.First(s => s.ID == StokID);
            Stok.STOKADI = txtStokAdi.Text;
            Stok.STOKALISFIYAT = decimal.Parse(txtAlisFiyat.Text);
            Stok.STOKALISKDV = decimal.Parse(txtAlisKdv.Text);
            Stok.STOKBARKOD = txtBarkod.Text;
            Stok.STOKBIRIM = txtBirim.Text;
            Stok.STOKGRUPID = GrupID;
            Stok.STOKKODU = txtStokKodu.Text;
            if (Resim) Stok.STOKRESIM = new
System.Data.Linq.Binary(Resimleme.ResimYukleme(pictureBox1.Image));
            Stok.STOKSATISFIYAT = decimal.Parse(txtSatisFiyat.Text);
            Stok.STOKSATISKDV = decimal.Parse(txtSatisKdv.Text);
            Stok.STOKEDITDATE = DateTime.Now;
            Stok.STOKEDITUSER= AnaForm.UserID;
            DB.SubmitChanges();
            Mesajlar.Guncelle(true);
            Temizle();
        }
        catch (Exception e)
        {
            Mesajlar.Hata(e);
        }
    }

    void Sil()
    {
        try
        {
            DB.TBL_STOKLARs.DeleteOnSubmit(DB.TBL_STOKLARs.First(s => s.ID == StokID));
        }
        catch (Exception e)
        {
            Mesajlar.Hata(e);
        }
    }

    private void txtStokKodu_ButtonClick(object sender,
DevExpress.XtraEditors.Controls.ButtonPressedEventArgs e)
    {
        int ID = Formlar.StokListesi(true);
        if(ID > 0 )
        {
            Ac(ID);
        }
        AnaForm.Aktar = -1;
    }

```

```

private void txtGrupKodu_ButtonClick(object sender,
DevExpress.XtraEditors.Controls.ButtonPressedEventArgs e)
{
    int Id = Formlar.StokGruplari(true);
    if(Id > 0 )
    {
        GrupAc(Id);
    }
    AnaForm.Aktar = -1;
}

private void BtnSil_Click(object sender, EventArgs e)
{
    if (Edit && StokID > 0 && Mesajlar.Sil() == DialogResult.Yes) Sil();
}

private void BtnKaydet_Click(object sender, EventArgs e)
{
    if(Edit && StokID > 0 && Mesajlar.Guncelle() == DialogResult.Yes)
    {
        Guncelle();
    }
    else
    {
        YeniKaydet();
    }
}

private void frmStokKarti_FormClosing(object sender, FormClosingEventArgs e)
{
    Mesajlar.FormKapanis(this.Text);
}
}
}

```

### 3)Stok Listesi Sınıfları

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Data.Linq;
using System.Linq;
using System.Threading.Tasks;
using System.Windows.Forms;
using DevExpress.XtraEditors;

namespace Otomasyon.Modul_Stok
{
    public partial class frmStokListesi : DevExpress.XtraEditors.XtraForm
    {
        Fonksiyonlar.DatabaseDataContext DB = new Fonksiyonlar.DatabaseDataContext();
    }
}

```

```

public bool secim = false;
int SecimID = -1;

public frmStokListesi()
{
    InitializeComponent();
}

private void frmStokListesi_Load(object sender, EventArgs e)
{
    Listele();
}

void Listele()
{
    var lst = from s in DB.TBL_STOKLARs
              where s.STOKADI.Contains(txtStokAdi.Text) &&
s.STOKBARKOD.Contains(txtBarkod.Text) && s.STOKKODU.Contains(txtStokKodu.Text)
              select s;
    Liste.DataSource = lst;
}

private void BtnAra_Click(object sender, EventArgs e)
{
    Listele();
}

private void BtnSil_Click(object sender, EventArgs e)
{
    txtStokKodu.Text = "";
    txtStokAdi.Text = "";
    txtBarkod.Text = "";
}

void Sec()
{
    try
    {
        SecimID = int.Parse(gridView1.GetFocusedRowCellValue("ID").ToString());
    }
    catch(Exception)
    {
        SecimID = -1;
    }
}

private void gridView1_DoubleClick(object sender, EventArgs e)
{
    Sec();
    if (secim && SecimID > 0)
    {
        AnaForm.Aktar = SecimID;
        this.Close();
    }
}

```

```

private void btnPrintToPdf_Click(object sender, EventArgs e)
{
    using (SaveFileDialog sf = new SaveFileDialog())
    {
        sf.Filter = "Acrobat Reader|*.pdf";
        if (sf.ShowDialog() == DialogResult.OK)
        {
            Liste.ExportToPdf(sf.FileName);
        }
    }
}

private void btnPrintToXls_Click(object sender, EventArgs e)
{
    using (SaveFileDialog sf = new SaveFileDialog())
    {
        sf.Filter = "Excel|*.xls";
        if (sf.ShowDialog() == DialogResult.OK)
        {
            Liste.ExportToXls(sf.FileName);
        }
    }
}
}
}

```

## VTYS'DEKİ İLİŞKİ DİYAGRAMLARI

LAPTOP-UK1FELK....VW\_BANKADURUM\* x

Column	Alias	Table	Output	Sort Type	Sort Order	Group By	Filter	Or...	Or...	Or...
ID		TBL_BANK...	<input checked="" type="checkbox"/>			Group By				
CASE WHEN ...	GIRIS		<input checked="" type="checkbox"/>			Sum				
CASE WHEN ...	CIKIS		<input checked="" type="checkbox"/>			Sum				
SUM(CASE W...	BAKIYE		<input checked="" type="checkbox"/>			Expression				

```

SELECT dbo.TBL_BANKALAR.ID, SUM(CASE WHEN dbo.TBL_BANKAHAREKETLERI.GCKODU = 'G' THEN dbo.TBL_BANKAHAREKETLERI.TUTAR ELSE 0 END) AS GIRIS,
SUM(CASE WHEN dbo.TBL_BANKAHAREKETLERI.GCKODU = 'C' THEN dbo.TBL_BANKAHAREKETLERI.TUTAR ELSE 0 END) AS CIKIS,
SUM(CASE WHEN dbo.TBL_BANKAHAREKETLERI.GCKODU = 'G' THEN dbo.TBL_BANKAHAREKETLERI.TUTAR ELSE 0 END)
- SUM(CASE WHEN dbo.TBL_BANKAHAREKETLERI.GCKODU = 'C' THEN dbo.TBL_BANKAHAREKETLERI.TUTAR ELSE 0 END) AS BAKIYE
FROM   dbo.TBL_BANKAHAREKETLERI LEFT OUTER JOIN
      dbo.TBL_BANKALAR ON dbo.TBL_BANKAHAREKETLERI.BANKAID = dbo.TBL_BANKALAR.ID
GROUP BY dbo.TBL_BANKALAR.ID

```



LAPTOP-UK1FELKG...ANKAHAREKETLERI\* x

TBL\_BANKAHAREKETLERI

☐ \* (All Columns)

☒ ID

☒ BANKAID

☒ CARIID

☒ BELGENO

☒ TARİH

☒ EVRAKTURU

☒ EVRAKID

☐ GCKODU

☒ TUTAR

☒ ACIKLAMA

☐ SAVEDATE

☐ SAVEUSER

☐ EDITDATE

☐ EDITUSER

TBL\_BANKALAR

☐ \* (All Columns)

☐ ID

☒ HESAPNO

☐ IBAN

☐ BANKAADI

☒ HESAPADI

☐ SUBE

☐ TEL

☐ ADRES

☐ TEMSILCI

☐ TEMSILCIEMAIL

☐ SAVEDATE

☐ SAVEUSER

☐ EDITDATE

☐ EDITUSER

Column	Alias	Table	Output	Sort Type	Sort Order	Filter	Or...	Or...	Or...
ID		TBL_BANK...	<input checked="" type="checkbox"/>						
BANKAID		TBL_BANK...	<input checked="" type="checkbox"/>						
CARIID		TBL_BANK...	<input checked="" type="checkbox"/>						
BELGENO		TBL_BANK...	<input checked="" type="checkbox"/>						

```

SELECT dbo.TBL_BANKAHAREKETLERI.ID, dbo.TBL_BANKAHAREKETLERI.BANKAID, dbo.TBL_BANKAHAREKETLERI.CARIID, dbo.TBL_BANKAHAREKETLERI.BELGENO, dbo.TBL_BANKAHAREKETLERI.TARİH,
dbo.TBL_BANKAHAREKETLERI.EVRAKTURU, dbo.TBL_BANKAHAREKETLERI.EVRAKID, dbo.TBL_BANKAHAREKETLERI.TUTAR, dbo.TBL_BANKAHAREKETLERI.ACIKLAMA,
(CASE WHEN dbo.TBL_BANKAHAREKETLERI.GCKODU = 'G' THEN dbo.TBL_BANKAHAREKETLERI.TUTAR ELSE 0 END) AS GIRIS,
(CASE WHEN dbo.TBL_BANKAHAREKETLERI.GCKODU = 'C' THEN dbo.TBL_BANKAHAREKETLERI.TUTAR ELSE 0 END) AS CIKIS, dbo.TBL_BANKALAR.HESAPNO, dbo.TBL_BANKALAR.HESAPADI
FROM   dbo.TBL_BANKAHAREKETLERI LEFT OUTER JOIN
        dbo.TBL_BANKALAR ON dbo.TBL_BANKAHAREKETLERI.BANKAID = dbo.TBL_BANKALAR.ID

```

LAPTOP-UK1FELKG...VW\_BANKALISTESI\* x

TBL\_BANKALAR

☐ \* (All Columns)

☒ ID

☒ HESAPNO

☒ IBAN

☒ BANKAADI

☒ HESAPADI

☐ SUBE

☐ TEL

☐ ADRES

☐ TEMSILCI

☐ TEMSILCIEMAIL

☐ SAVEDATE

☐ SAVEUSER

☐ EDITDATE

☐ EDITUSER

VW\_BANKADURUM

☐ \* (All Columns)

☐ ID

☒ GIRIS

☒ CIKIS

☒ BAKIYE

Column	Alias	Table	Output	Sort Type	Sort Order	Filter	Or...	Or...	Or...
ID		TBL_BANKALAR	<input checked="" type="checkbox"/>						
HESAPNO		TBL_BANKALAR	<input checked="" type="checkbox"/>						
IBAN		TBL_BANKALAR	<input checked="" type="checkbox"/>						
BANKAADI		TBL_BANKALAR	<input checked="" type="checkbox"/>						
HESAPADI		TBL_BANKALAR	<input checked="" type="checkbox"/>						
GIRIS		VW_BANKADURUM	<input checked="" type="checkbox"/>						
CIKIS		VW_BANKADURUM	<input checked="" type="checkbox"/>						
BAKIYE		VW_BANKADURUM	<input checked="" type="checkbox"/>						

```

SELECT dbo.TBL_BANKALAR.ID, dbo.TBL_BANKALAR.HESAPNO, dbo.TBL_BANKALAR.IBAN, dbo.TBL_BANKALAR.BANKAADI, dbo.TBL_BANKALAR.HESAPADI, dbo.VW_BANKADURUM.GIRIS, dbo.VW_BANKADURUM.CIKIS,
        dbo.VW_BANKADURUM.BAKIYE
FROM   dbo.TBL_BANKALAR LEFT OUTER JOIN
        dbo.VW_BANKADURUM ON dbo.TBL_BANKALAR.ID = dbo.VW_BANKADURUM.ID

```

LAPTOP-UK1FELKG...bo.VW\_FATURALAR\* x

TBL\_CARILER

☐ \* (All Columns)

☐ ID

☐ CARIKOD

☒ CARIADI

☒ VERDIDAIRESI

☒ VERGINO

☐ GRUPID

☐ ULKE

☒ SEHIR

☒ ILCE

☒ ADRES

☐ TELEFON1

☐ TELEFON2

☐ FAX1

☐ FAX2

☐ WEBADRES

☐ MAILINFO

☐ YETKILI1

☐ YETKILIEMAIL1

☐ YETKILI2

☐ YETKILIEMAIL2

☐ SAVEDATE

☐ SAVEUSER

☐ EDITDATE

☐ EDITUSER

TBL\_FATURALAR

☐ \* (All Columns)

☐ ID

☐ FATURATURU

☐ FATURASERI

☒ FATURANO

☐ IRSALIYEID

☒ CARIKODU

☒ TARIHI

☒ ACIKLAMA

☒ GENELTOPLAM

☐ ODEMEYERI

☐ ODEMEYERIID

☐ SAVEUSER

☐ SAVEDATE

☐ EDITDATE

VW\_KALEMLER

☐ \* (All Columns)

☐ ID

☐ STOKID

☐ FATURAID

☐ IRSALIYEID

☒ STOKKODU

☒ STOKADI

☒ STOKBARKOD

☒ STOKBIRIM

☒ BIRIMFIYAT

☒ KDV

☒ MIKTAR

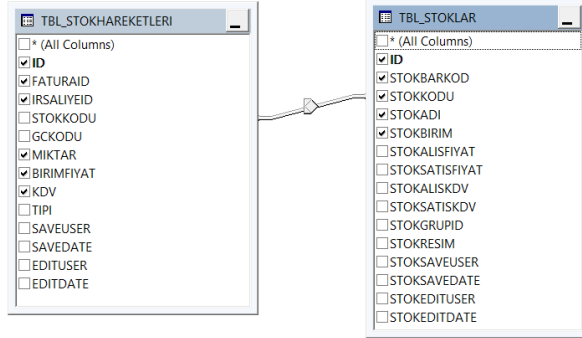
☒ TOPLAM

Column	Alias	Table	Output	Sort Type	Sort Order	Filter	Or...	Or...	Or...
CARIKODU		TBL_FATURALAR	<input checked="" type="checkbox"/>						
CARIADI		TBL_CARILER	<input checked="" type="checkbox"/>						
VERDIDAIRESI		TBL_CARILER	<input checked="" type="checkbox"/>						
VERGINO		TBL_CARILER	<input checked="" type="checkbox"/>						
SEHIR		TBL_CARILER	<input checked="" type="checkbox"/>						
ILCE		TBL_CARILER	<input checked="" type="checkbox"/>						
ADRES		TBL_CARILER	<input checked="" type="checkbox"/>						
FATURANO		TBL_FATURALAR	<input checked="" type="checkbox"/>						
STOKKODU		VW_KALEMLER	<input checked="" type="checkbox"/>						
STOKADI		VW_KALEMLER	<input checked="" type="checkbox"/>						
STOKBARKOD		VW_KALEMLER	<input checked="" type="checkbox"/>						
STOKBIRIM		VW_KALEMLER	<input checked="" type="checkbox"/>						
BIRIMFIYAT		VW_KALEMLER	<input checked="" type="checkbox"/>						
KDV		VW_KALEMLER	<input checked="" type="checkbox"/>						
MIKTAR		VW_KALEMLER	<input checked="" type="checkbox"/>						
TOPLAM		VW_KALEMLER	<input checked="" type="checkbox"/>						

```

SELECT dbo.TBL_FATURALAR.CARIKODU, dbo.TBL_CARILER.CARIADI, dbo.TBL_CARILER.VERDIDAIRESI, dbo.TBL_CARILER.VERGINO, dbo.TBL_CARILER.SEHİR, dbo.TBL_CARILER.İLCE, dbo.TBL_CARILER.ADRES,
        dbo.TBL_FATURALAR.FATURANO, dbo.TBL_FATURALAR.TARIHI, dbo.TBL_FATURALAR.ACIKLAMA, dbo.TBL_FATURALAR.GENELTOPLAM, dbo.VW_KALEMLER.STOKKODU, dbo.VW_KALEMLER.STOKADI,
        dbo.VW_KALEMLER.STOKBARKOD, dbo.VW_KALEMLER.STOKBIRIM, dbo.VW_KALEMLER.BIRIMFIYAT, dbo.VW_KALEMLER.KDV, dbo.VW_KALEMLER.MIKTAR, dbo.VW_KALEMLER.TOPLAM
FROM   dbo.VW_KALEMLER INNER JOIN
        dbo.TBL_FATURALAR ON dbo.VW_KALEMLER.FATURAID = dbo.TBL_FATURALAR.ID LEFT OUTER JOIN
        dbo.TBL_CARILER ON dbo.TBL_FATURALAR.CARIKODU = dbo.TBL_CARILER.CARIKOD

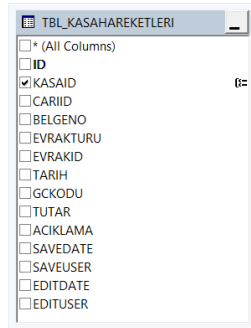
```



```

SELECT dbo.TBL_STOKHAREKETLERI.ID, dbo.TBL_STOKLAR.ID AS STOKID, dbo.TBL_STOKHAREKETLERI.FATURAID, dbo.TBL_STOKHAREKETLERI.IRSALIYEID, dbo.TBL_STOKLAR.STOKKODU, dbo.TBL_STOKLAR.STOKADI,
       dbo.TBL_STOKLAR.STOKBARKOD, dbo.TBL_STOKLAR.STOKBIRIM, dbo.TBL_STOKHAREKETLERI.BIRIMFIYAT, dbo.TBL_STOKHAREKETLERI.KDV, dbo.TBL_STOKHAREKETLERI.MIKTAR,
FROM   dbo.TBL_STOKHAREKETLERI LEFT OUTER JOIN
       dbo.TBL_STOKLAR ON dbo.TBL_STOKHAREKETLERI.STOKKODU = dbo.TBL_STOKLAR.STOKKODU

```

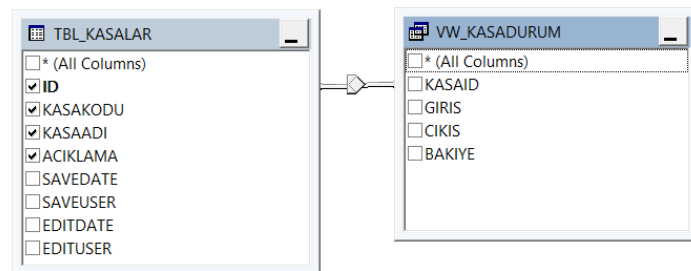


Column	Alias	Table	Output	Sort Type	Sort Order	Group By	Filter	Or...	Or...	Or...
KASAAID		TBL_KASA...	<input checked="" type="checkbox"/>			Group By				
CASE WHEN ...	GIRIS		<input checked="" type="checkbox"/>			Sum				
CASE WHEN ...	CIKIS		<input checked="" type="checkbox"/>			Sum				
SUM(CASE W...	BAKIYE		<input checked="" type="checkbox"/>			Expression				

```

SELECT KASAAID, SUM(CASE WHEN GCKODU = 'G' THEN TUTAR ELSE 0 END) AS GIRIS, SUM(CASE WHEN GCKODU = 'C' THEN TUTAR ELSE 0 END) AS CIKIS, SUM(CASE WHEN GCKODU = 'G' THEN TUTAR ELSE 0 END)
- SUM(CASE WHEN GCKODU = 'C' THEN TUTAR ELSE 0 END) AS BAKIYE
FROM   dbo.TBL_KASAHAREKETLERI
GROUP BY KASAAID

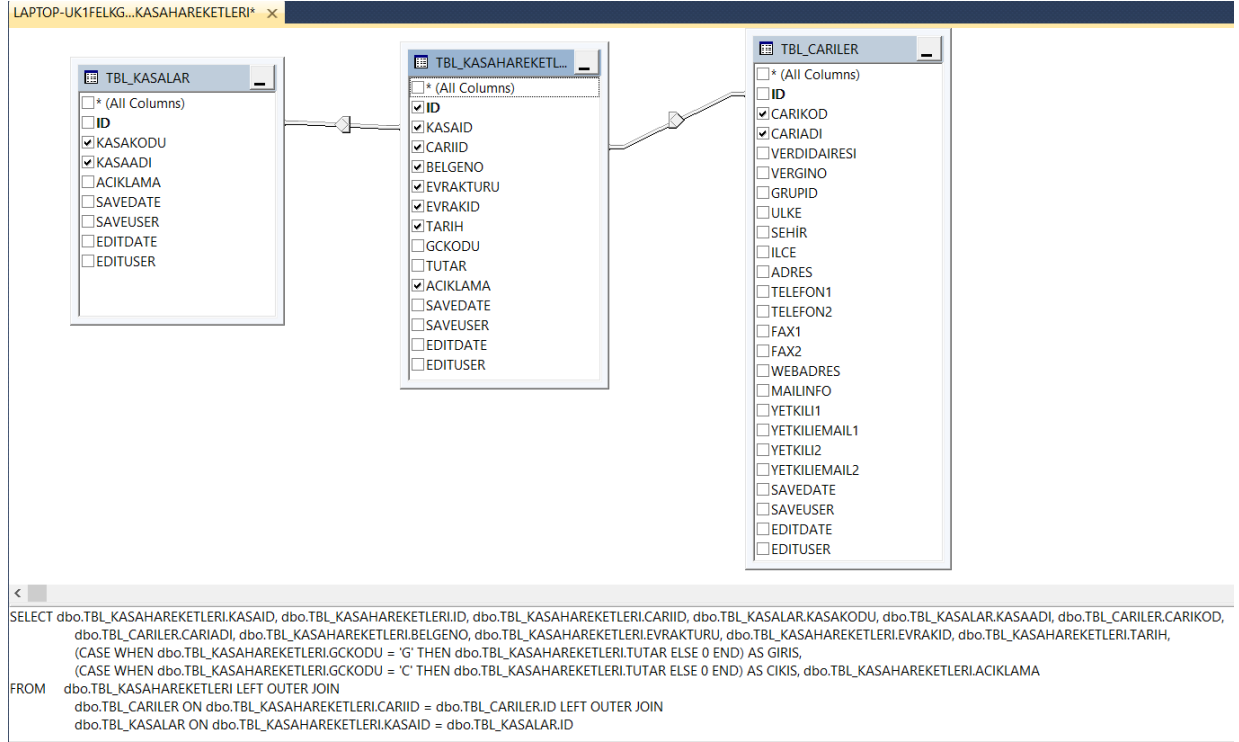
```



```

SELECT dbo.TBL_KASALAR.ID, dbo.TBL_KASALAR.KASAKODU, dbo.TBL_KASALAR.KASAADI, dbo.TBL_KASALAR.ACIKLAMA, (CASE WHEN dbo.VW_KASADURUM.GIRIS IS NULL
THEN 0 ELSE dbo.VW_KASADURUM.GIRIS END) AS GIRIS, (CASE WHEN dbo.VW_KASADURUM.CIKIS IS NULL THEN 0 ELSE dbo.VW_KASADURUM.CIKIS END) AS CIKIS,
(CASE WHEN dbo.VW_KASADURUM.BAKIYE IS NULL THEN 0 ELSE dbo.VW_KASADURUM.BAKIYE END) AS BAKIYE
FROM   dbo.TBL_KASALAR LEFT OUTER JOIN
       dbo.VW_KASADURUM ON dbo.TBL_KASALAR.ID = dbo.VW_KASADURUM.KASAAID

```



**NOT:** Otomasyonun tasarım kısmında Dev Express programı kullanılmıştır.