



TC

Düzce Üniversitesi

Mühendislik Fakültesi

Bilgisayar Mühendisliği Bölümü

BM309- Veri Tabanı Yönetim Sistemleri

RAPOR

Proje - Grup No: 8

Proje Konusu: Bir online alışveriş mağazasının ürün, ürün satış, kargo ve ödeme kayıtlarını tutmak amacıyla veri tabanı projesi

Teslim Eden;

<u>No</u>	<u>İsim- Soyisim</u>
1. 211001086	ÇAĞLA ŞILAK
2. 211001087	YASİN YILMAZ
3. 221001442	MEHMET CANTDEMİR
4. 201001040	EREN YENİÇERİ

1. Senaryo

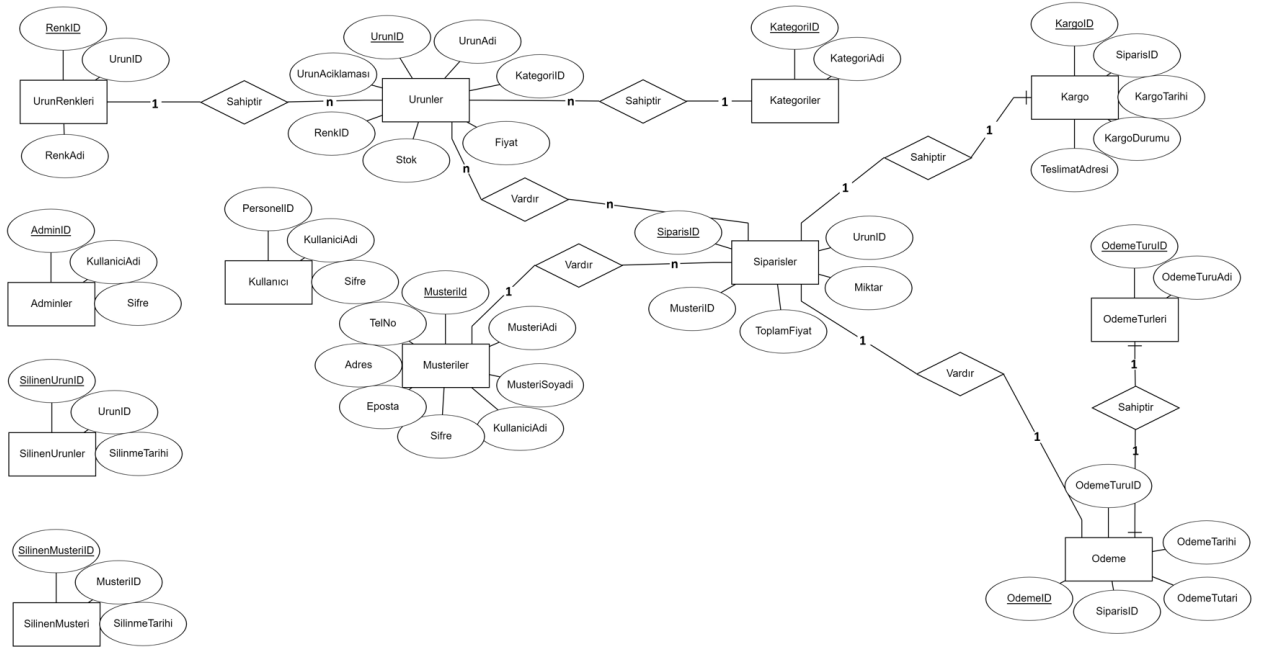
Bir online alışveriş mağazasının ürün, ürün satış, kargo ve ödeme kayıtlarını tutmak amacıyla veri tabanı projesi.

2. İş Kuralları

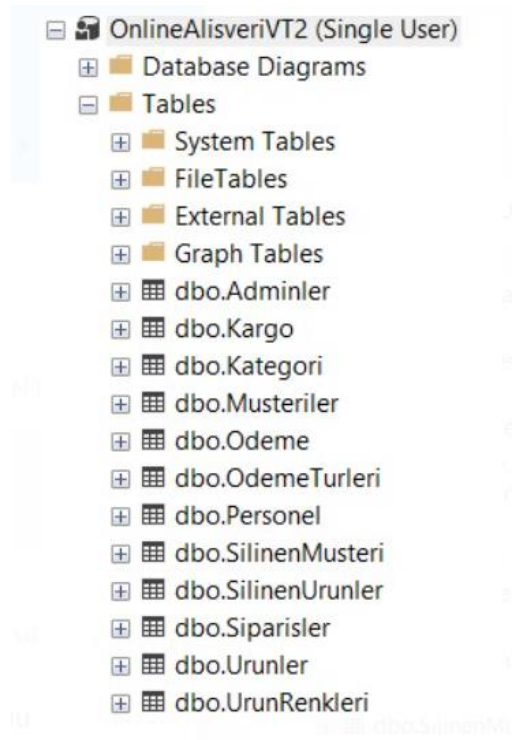
- Mağazada satılan her ürün için benzersiz bir Kimlik (ID) bulunmaktadır.
- Ürünlerin özellikleri arasında marka, kategori, fiyat, stok miktarı, açıklama ve resimler bulunur.
- Ürünler kategorize edilmiş olduğu bir sistem vardır.
- Her müşteri için benzersiz bir Kimlik (ID) mevcuttur.
- Müşterilerin adı, soyadı, e-posta adresi, teslimat adresleri, ödeme bilgileri ve sipariş geçmişi gibi bilgiler kaydedilir.
- Müşteri isteğe bağlı olarak kullanıcı profilleri oluşturabilirler.
- Müşteriler, sepete ürün ekleyebilir ve satın alma işlemini tamamlayabilirler.
- Farklı ödeme seçenekleri sunulur. (Kredi kartı, EFT/Havale, Nakit vb.)
- Müşteri kargo takibi için gerekli bilgiler sağlanır.
- Kolay bir şekilde alınacak ürün sepete eklenir ve sipariş tamamlandığında sipariş ver butonuna basılır sonrasında ödeme ekranına gider ödeme seçeneklerini seçtikten sonra onayla butonuna tıklandığında sipariş verilmiş olur.

3. ER DİYAGRAMLARI

Chen Diyagramı



Diyagram üzerinde ‘bire-bir, bire-çok, çok-a-çok’ ilişkiler ve PK/FK eşleştirmeleri gösterilmiştir.



4.2. VERİTABANI TABLOLARI

Online Alışveriş firması veri tabanında toplam 12 adet tablo oluşmaktadır. Bu tablolar queryler aracılığıyla oluşturulmuştur. Queryler ile tablolara PK/FK eşleştirilmesi, constrainte ve check kuralları eklenmesi gibi çeşitli özellikler eklenmiştir.

4.2.1. ANA TABLOLAR:

```
CREATE TABLE Musteriler (  
    MusteriID int PRIMARY KEY IDENTITY (1,1),  
    MusteriAdi varchar(25),  
    MusteriSoyadi varchar(25),  
    KullaniciAdi varchar(10),  
    Sifre varchar(7),  
    Adres varchar(50),  
    Eposta varchar(20),  
    Telno varchar(11)  
)
```

```
CREATE TABLE Kategori (  
    KategoriID int PRIMARY KEY IDENTITY (1,1),  
    KategoriAdi varchar(20)  
)
```

```
CREATE TABLE Odeme (  
    OdemeID int PRIMARY KEY IDENTITY (1,1),  
    SiparisID int,  
    OdemeTarihi date,  
    OdemeTutari decimal,  
    OdemeTuruID int,  
    FOREIGN KEY (SiparisID) REFERENCES Siparisler (SiparisID) ON DELETE  
CASCADE,  
    FOREIGN KEY (OdemeTuruID) REFERENCES OdemeTurleri (OdemeTuruID) ON  
DELETE CASCADE  
)
```

```
CREATE TABLE Personel(  
    PersonelID int PRIMARY KEY IDENTITY (1,1),  
    KullaniciAdi varchar(15),  
    Sifre varchar(15)
```

```
)
```

```
INSERT INTO Personel  
VALUES ('memo','1234')
```

```
CREATE TABLE SilinenMusteri (  
    SilinenMusteriID int PRIMARY KEY IDENTITY(1,1),  
    MusteriID int,  
    SilinmeTarihi date,  
    SilenAdmin varchar(15),  
    FOREIGN KEY (MusteriID) REFERENCES Musteriler (MusteriID) ON DELETE  
CASCADE  
)
```

```
DROP TABLE SilinenMusteri
```

```
CREATE TABLE Siparisler (  
    SiparisID int PRIMARY KEY IDENTITY (1,1),  
    Miktar int,  
    ToplamFiyat decimal,  
    MusteriID int,  
    UrunID int,  
    FOREIGN KEY (MusteriID) REFERENCES Musteriler(MusteriID) ON DELETE  
CASCADE,  
    FOREIGN KEY (UrunID) REFERENCES Urunler (UrunID) ON DELETE  
CASCADE  
)
```

```
CREATE TABLE Urunler (  
    UrunID int PRIMARY KEY IDENTITY (1,1),  
    UrunAdi varchar(25),  
    UrunAciklamasi varchar(50),  
    Fiyat money,  
    Stok int,
```

```

        KategoriID int ,
        RenkID int,
FOREIGN KEY (KategoriID) REFERENCES Kategori(KategoriID) ON DELETE
CASCADE,
FOREIGN KEY (RenkID) REFERENCES UrunRenkleri (RenkID) ON DELETE
CASCADE
    )

```

4.2.2. SİLİNİMİŞ KAYITLAR TABLOLARI:

```

CREATE TABLE SilinenMusteri (
        SilinenMusteriID int PRIMARY KEY IDENTITY(1,1),
        MusteriID int,
        SilinmeTarihi date,
        SilenAdmin varchar(15),
FOREIGN KEY (MusteriID) REFERENCES Musteriler (MusteriID) ON DELETE
CASCADE
    )
DROP TABLE SilinenMusteri

CREATE TABLE SilinenUrunler (
        SilinenUrunID int PRIMARY KEY IDENTITY(1,1),
        UrunID int,
        SilinmeTarihi date,
        AdminID int,
FOREIGN KEY (AdminID) REFERENCES Adminler(AdminID) ON DELETE
CASCADE
    )

```

4.3. TRIGGER

4.3.1. Müşteri tablosundan silinen kişinin bilgileri Silinen Müşteriler tablosuna aktarılır

```

CREATE TRIGGER MusteriSilmeK
on dbo.Musteriler
AFTER DELETE
AS
BEGIN

```

```
SET NOCOUNT ON;

INSERT INTO SilinenMusteri (MusteriID,SilinmeTarihi)

SELECT MusteriID , GETDATE()

FROM deleted

END;

SELECT * FROM sys.triggers;
```

4.3.2. Ürün tablosundan silinen ürünün bilgileri Silinen Ürünler tablosuna aktarılır

```
CREATE TRIGGER UrunSilmeK

on dbo.Urunler

AFTER DELETE

AS

BEGIN

set NOCOUNT ON;

INSERT INTO SilinenUrunler(UrunID,SilinmeTarihi)

SELECT UrunID,GETDATE()

FROM deleted

END;
```

4.3.3. Ürünler tablosundaki ürün eklendiğinde otomatik stok azaltılır

```
CREATE TRIGGER UpdateStok

ON Urunler

AFTER INSERT

AS

BEGIN

DECLARE @UrunID INT, @Stok INT;

SELECT @UrunID = i.UrunID, @Stok = i.Stok

FROM inserted i;
```



```
UPDATE Urunler  
SET Stok = Stok - @Stok  
WHERE UrunID = @UrunID;  
END;
```

4.4. STORED PROCEDURES

4.4.1. Silinen Müşterileri Gösteren Stored Procedures

```
CREATE PROCEDURE SilinenMusteriListele  
AS  
BEGIN  
SELECT * FROM SilinenMusteri;  
END;
```

4.4.2. Müşterileri Listeleyen Stored Procedures

```
CREATE PROCEDURE MusteriListele  
AS  
BEGIN  
SELECT * FROM Musteriler;  
END;
```

4.4.3. Uygulamamızın önyüzünde Ürün Eklendiğinde Ürünler tablosuna ekleyen Stored Procedures

```
CREATE PROCEDURE UrunEkleme  
@urunAdi varchar (20),  
@urunAciklamasi varchar(20),  
@fiyat int,  
@kategori int,  
@renk int  
AS  
BEGIN
```

```
INSERT INTO Urunler(UrunAdi,UrunAciklamasi,Fiyat,KategoriID,RenkID)
VALUES
    (@urunAdi,@urunAciklamasi,@fiyat,@kategori,@renk)
END
```

4.4.4. Uygulamanın önyüzünde girilen ürünID'sine göre Ürün silen Stored Procedures

```
CREATE PROCEDURE UrunSil
```

```
    @urunKodu int
```

```
AS
```

```
BEGIN
```

```
DELETE FROM Urunler
```

```
WHERE UrunID=@urunKodu
```

```
END
```

4.4.5. Uygulamanın önyüzünde UrunID'sine göre ürün güncelleyen Stored Procedures

```
CREATE PROCEDURE UrunGuncelleme
```

```
    @urunID int,
```

```
    @urunAdi varchar(20),
```

```
    @urunAciklamasi varchar(20),
```

```
    @fiyat int,
```

```
    @kategori int,
```

```
    @renk int
```

```
AS
```

```
BEGIN
```

```
UPDATE Urunler
SET
    UrunAdi=@urunAdi,
    UrunAciklamasi=@urunAciklamasi,
    Fiyat=@fiyat,
    KategoriID=@kategori,
    RenkID=@renk

WHERE
    UrunID=@urunID;

END
```

4.4.6. Ürünleri Görmek istediğimizde çağırılan ürünleri listeleyen Stored Procedures

```
CREATE PROCEDURE UrunListele
AS
BEGIN
    SELECT * FROM Urunler
END
```

4.4.7. Kategori tablosunda KategoriID'si 1 olan veriler Sweatshirt'e denk gelir. Aynı şekilde 2 ve 3 pantolon ve elbiseye denk gelir. KategoriID'sine göre ürünü listeleyen Stored Procedures

```
CREATE PROCEDURE sweatshirtListele
AS
BEGIN
    SELECT UrunAdi,UrunAciklamasi,Fiyat,Stok,KategoriID,RenkID
    FROM Urunler
    WHERE KategoriID = 1
```

```

END

CREATE PROCEDURE elbiseListele
AS
BEGIN

SELECT UrunAdi,UrunAciklamasi,Fiyat,Stok,KategoriID,RenkID
FROM Urunler
WHERE KategoriID = 2

END

CREATE PROCEDURE pantolonListele
AS
BEGIN

SELECT UrunAdi,UrunAciklamasi,Fiyat,Stok,KategoriID,RenkID
FROM Urunler
WHERE KategoriID = 3

END

```

4.4.8. Uygulamanın önyüzünde sipariş eklendiğinde siparişler tablosuna ekleyen Stored Procedures

```

CREATE PROCEDURE SiparisEkleme
    @miktar int ,
    @urunFiyati int,
    @musteriID int,
    @urunID int
AS
BEGIN

```

```
INSERT INTO Siparisler(Miktar,ToplamFiyat,MusteriID,UrunID)
VALUES
    (@miktar, @urunFiyati* @miktar ,@musteriID,@urunID)
END
```

4.4.9. Kargo sayfasında müşteri adı, müşteri soyadı ve müşterinin adresini alan Stored Procedures

```
CREATE PROCEDURE MusteriAdiSoyadiAdres
    @kullaniciAdi varchar
AS
BEGIN
    SELECT MusteriAdi , MusteriSoyadi ,Adres FROM Musteriler
    WHERE KullaniciAdi=@kullaniciAdi
END
select * from Adminler
```

4.4.10. Ödeme sayfasına gelindiğinde kargo bilgilerini getiren Stored Procedures

```
CREATE PROCEDURE kargoListele
AS
BEGIN
    SELECT SiparisID,KargoTarihi,KargoDurumu,TeslimatAdresi
    FROM Kargo
END
select * from Kargo
```

4.4.11. Uygulamanın önyüzünde Girilen verilere göre müşteri ekleyen Stored Procedures

```
CREATE PROCEDURE MusteriEkle
    @adi varchar (20),
```

```

        @soyadi varchar(20),
        @kullaniciadi varchar(20),
        @sifre varchar(20),
        @adres varchar(20),
        @eposta varchar(20),
        @telefon varchar(20)

AS

BEGIN

INSERT INTO
Musteriler(MusteriAdi,MusteriSoyadi,KullaniciAdi,Sifre,Adres,Eposta,Telno)

VALUES

        (@adi,@soyadi,@kullaniciadi,@sifre,@adres,@eposta,@telefon)

END

```

4.4.12. Uygulamanın önyüzünde Girilen verilere göre personel ekleyen Stored Procedures

```

CREATE PROCEDURE PersonelEkle

        @kullaniciAdi varchar (20),

        @sifre varchar(20)

AS

BEGIN

INSERT INTO Personel(KullaniciAdi,Sifre)

VALUES

        (@kullaniciAdi,@sifre)

END

```

4.4.13. KategoriID' ye göre Kategori Listeleyen Stored Procedures

```

CREATE PROCEDURE GetKategoriIDByAdi

```

```
        @KategoriAdi NVARCHAR(50)
AS
BEGIN
    SET NOCOUNT ON;

    SELECT KategoriID
    FROM Kategori
    WHERE KategoriAdi = @KategoriAdi;
END
```

4.4.14. Renk ismine göre renkleri listeleyen Stored Procedures

```
CREATE PROCEDURE GetRenkIDByAdi
    @RenkAdi NVARCHAR(50)
AS
BEGIN
    SET NOCOUNT ON;

    SELECT RenkID
    FROM urunRenkleri
    WHERE renkAdi = @RenkAdi;
END
```

4.4.15. Kategoriye göre Ürünleri listeleyen Stored Procedures

```
CREATE PROCEDURE GetUrunlerByKategoriID
    @KategoriID INT
AS
BEGIN
    SET NOCOUNT ON;
```

```
SELECT *  
  
FROM Urunler  
  
WHERE KategoriID = @KategoriID;  
  
END
```

4.4.16. Renge göre ürün listeleyen Stored Procedures

```
CREATE PROCEDURE GetUrunlerByRenkID  
  
    @RenkID INT  
  
AS  
  
BEGIN  
  
    SET NOCOUNT ON;  
  
  
    SELECT *  
  
    FROM urunler  
  
    WHERE RenkID = @RenkID;  
  
END
```

4.5. Kullanıcı Arayüzü

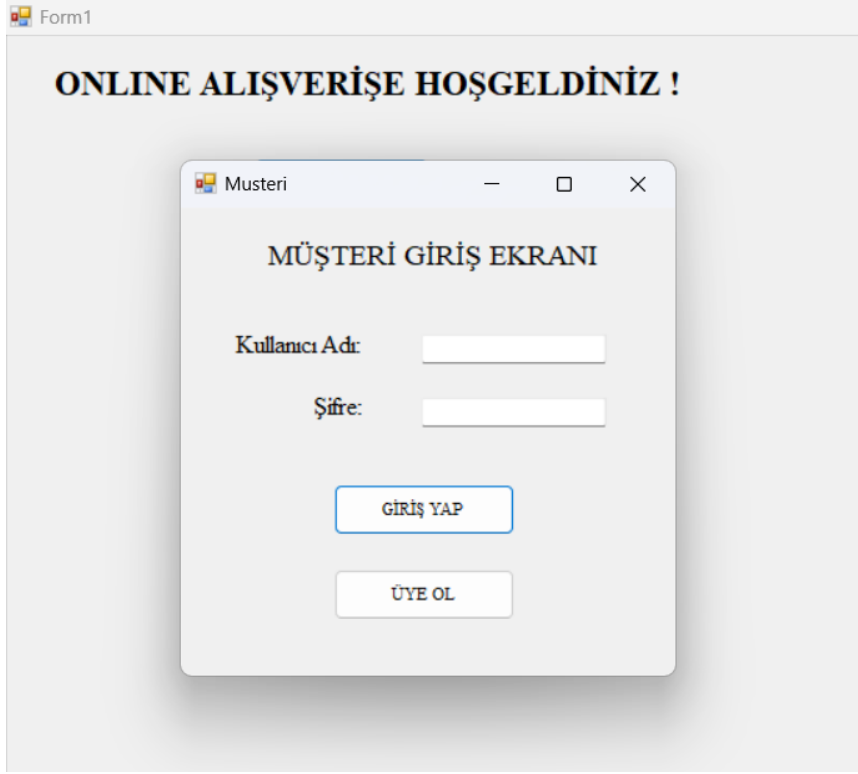
Uygulamanın ilk giriş ekranında bizi müşteri giriş, admin giriş ve personel giriş karşılıyor. Ürün satın almak isteniyorsa müşteri girişe, yetki verme ürün güncelleme vb. işlemler yapılmak isteniyorsa admin girişe, ürün silmek güncellemek veya eklemek vb. işlemler yapılmak isteniyorsa personel girişe tıklanılmalıdır.



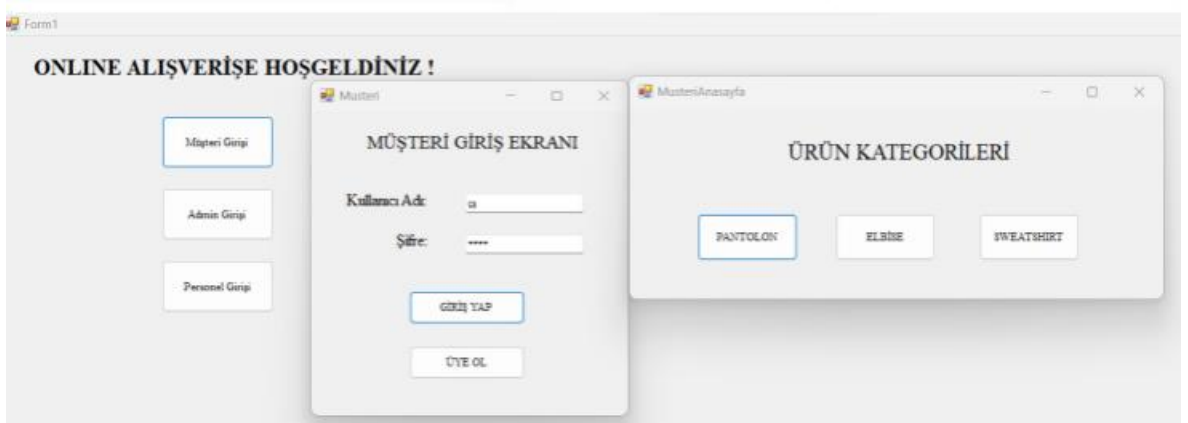
The screenshot shows a web application window titled 'Form1'. The main heading is 'ONLINE ALIŞVERİŞE HOŞGELDİNİZ !'. Below the heading, there are three buttons arranged vertically: 'Müşteri Girişi' (top), 'Admin Girişi' (middle), and 'Personel Girişi' (bottom). The buttons are light blue with black text and a thin blue border.

4.5.1 Müşteri Giriş

Müşteri giriş ekranına yönlendiriliyor. İlk aşamada kullanıcı adı ve şifre girişi yapılır. Sonrasında daha önceden üye olmuşsa giriş yap butonuna tıklanır eğer daha önceden üye olunmamışsa üye ol butonuna tıklanır.




Daha sonra ürün kategoriler sayfası karşımıza çıkar istediğimiz ürün örneğin pantolon butonuna tıklanır.



Pantolon butonuna tıklandığında istediğimiz ürünün adedini belirleyip sipariş ver butonuna tıklanır.


pantolon



Ürün Adı : jagger pantolon
Ürün Açıklaması : kışlık
Fiyatı : 400,0000
Stok : 30
Rengi : 1

Adet Belirleyin:


Sipariş Ver



Ürün Adı : KADİFE
Ürün Açıklaması : KISLIK
Fiyatı : 200,0000
Stok : 20
Rengi : 1

Adet Belirleyin:

Sipariş Ver




Ürün Adı : ?apkalo sweatshirt
Ürün Açıklaması : k??!?k
Fiyatı : 400,0000
Stok : 30
Rengi : 2

Adet Belirleyin:

Sipariş Ver

Sipariş ver butonuna tıklandıktan sonra ödeme ekranına yönlendirilir.


pantolon



Ürün Adı : jagger pantolon
Ürün Açıklaması : kışlık
Fiyatı : 400,0000
Stok : 30
Rengi : 1

Adet Belirleyin:

Sipariş Ver



Ürün Adı : KADİFE
Ürün Açıklaması : KISLIK
Fiyatı : 200,0000
Stok : 20
Rengi : 1

Adet Belirleyin:

Sipariş Ver

ÖdemeEkran

Ödeme Türü Seç: KREDİ KARTI

Seç

KARTIN:

Üzerindeki Ad Soyad:

Numarası:

Son Kullanma Tarihi:

CVV:

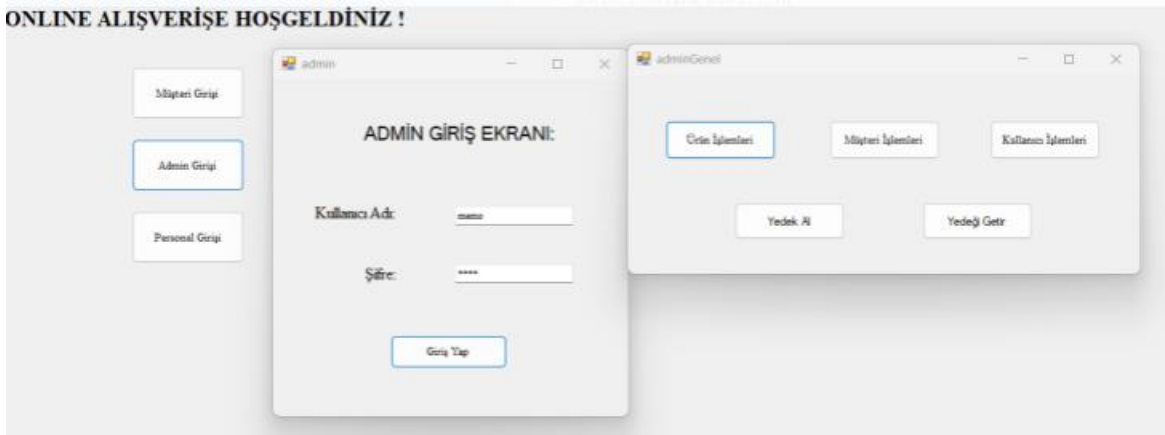
Onayla

Bu ekranda ödeme türü seçilir, kart bilgileri verilir ve onayla butonuna tıklandığında ödeme alınır ve sipariş tamamlanır. Kargo bilgileri müşteriye gösterilir.



4.5.2 Admin Giriş

Admin giriş ekranında kullanıcı adı ve şifre girildiğinde, Admin Genel sayfası açılır ve bu sayfada ürün işlemleri, müşteri işlemleri, kullanıcı işlemleri, yedek al işlemi, yedeği getir işlemi için butonlar bulunmaktadır. Buradaki işlemlerin hangisini gerçekleştirmek istiyorsak ona tıklanır.



Ürün işlemlerine erişmek istiyorsak, buradaki bilgileri doldururuz.

4.5.2.1. Ürün İşlemleri

adminÜrünEkranı

ÜRÜN İŞLEMLERİ

Ürün Ekle

Ürün ID:

Ürün Adı:

Ürün Açıklaması:

Fiyat:

Kategori ID:

Renk ID:

EKLE

Ürün Deyisi Sağ ve Kaydet

Ürün Güncelle

Ürün ID:

Ürün Adı:

Ürün Açıklaması:

Fiyat:

Kategori ID:

Renk ID:

GÜNCELLE

Ürün Sil

Ürün ID:

SİL

Ürünleri Listele

Silinen Ürünleri Listele

CSV Olarak Kaydet

PDF Olarak Kaydet

Ürünleri Listele butonuna tıkladığımızda eklediğimiz ürün burada gözüktür.

	ÜrünID	ÜrünAdı	ÜrünAçıklaması	Fiyat	Stok	KategoriID	RenkID
▶	1	Kısa Kollu Elbise	yazlık	700,0000	10	3	2
	2	Uzun Kollu elbise	kışlık	1000,0000	5	3	1
	3	jagger pantolon	kışlık	400,0000	30	2	1
	4	şapkalı sweatshirt	kışlık	400,0000	30	1	3
	6	KADİFE	KİSLİK	200,0000	20	2	1
	7	ŞAPKASIZ SWIT...	Kışlık	500,0000	40	1	5
	8	SWITSHIRT	Kışlık	500,0000	40	1	5

Ürünleri Listele

Silinen Ürünleri Listele

CSV Olarak Kaydet

PDF Olarak Kaydet

Ürün sildiğimizde ve Silinen Ürünleri Listele butonuna tıkladığımızda sildiğimiz ürünler listelenir.

	ÜrünID	SilinenÜrünID	SilinmeTarihi
▶	9	13	17.12.2023
	5	14	17.12.2023
	12	15	18.12.2023
	11	16	19.12.2023
	10	17	19.12.2023
	90	18	21.12.2023
*			

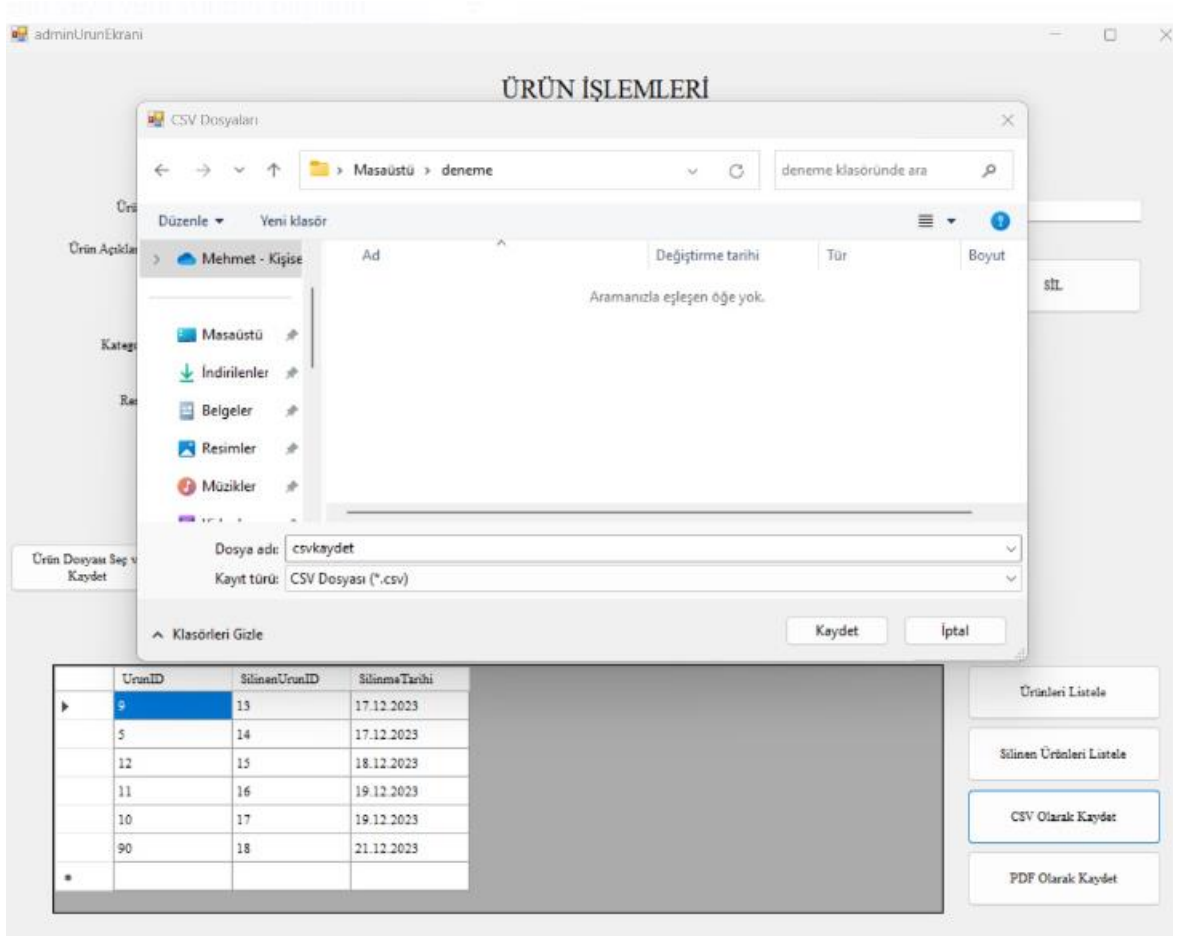
Ürünleri Listele

Silinen Ürünleri Listele

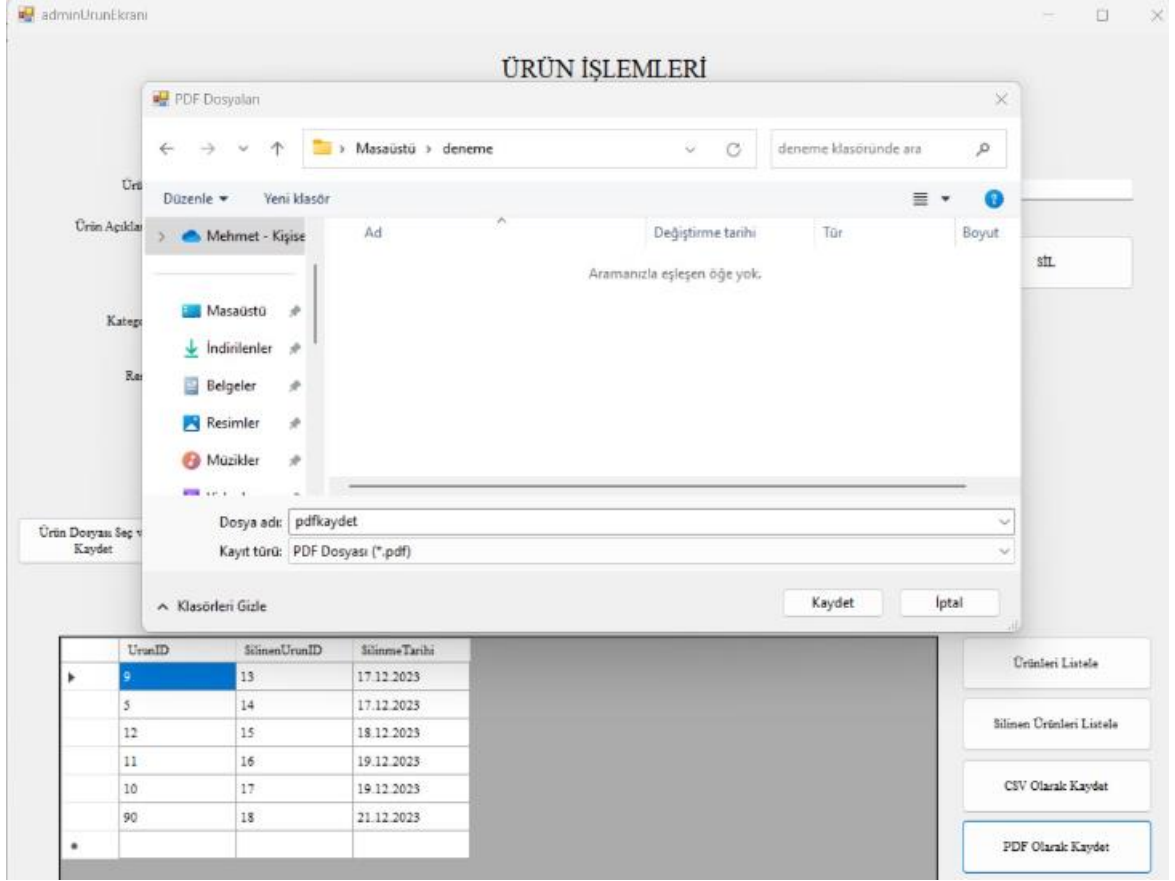
CSV Olarak Kaydet

PDF Olarak Kaydet

CSV Olarak Kaydet butonuna tıkladığımızda CSV olarak kaydeder.

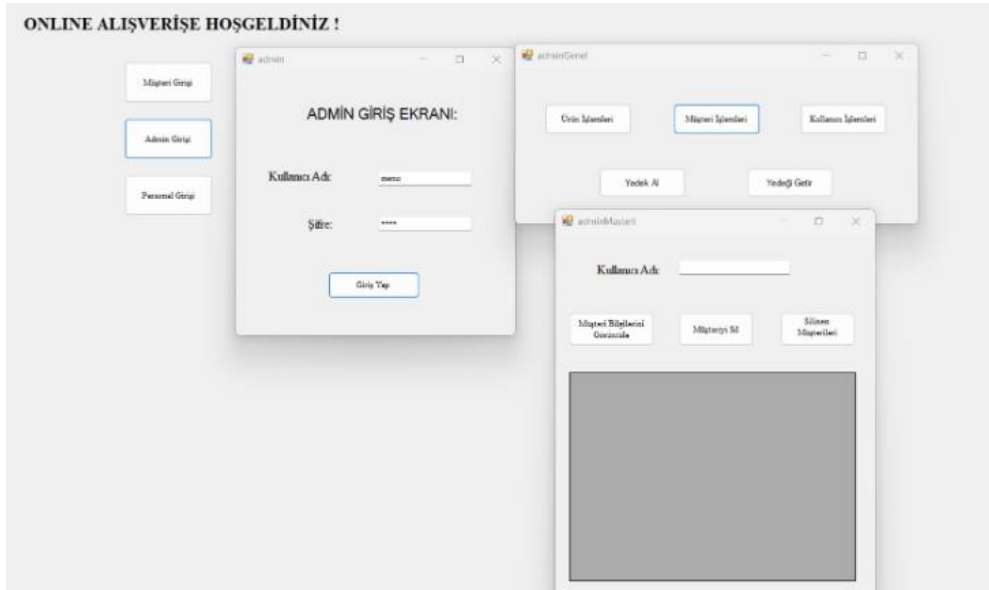


PDF Olarak Kaydet butonuna tıkladığımızda pdf olarak kaydeder.

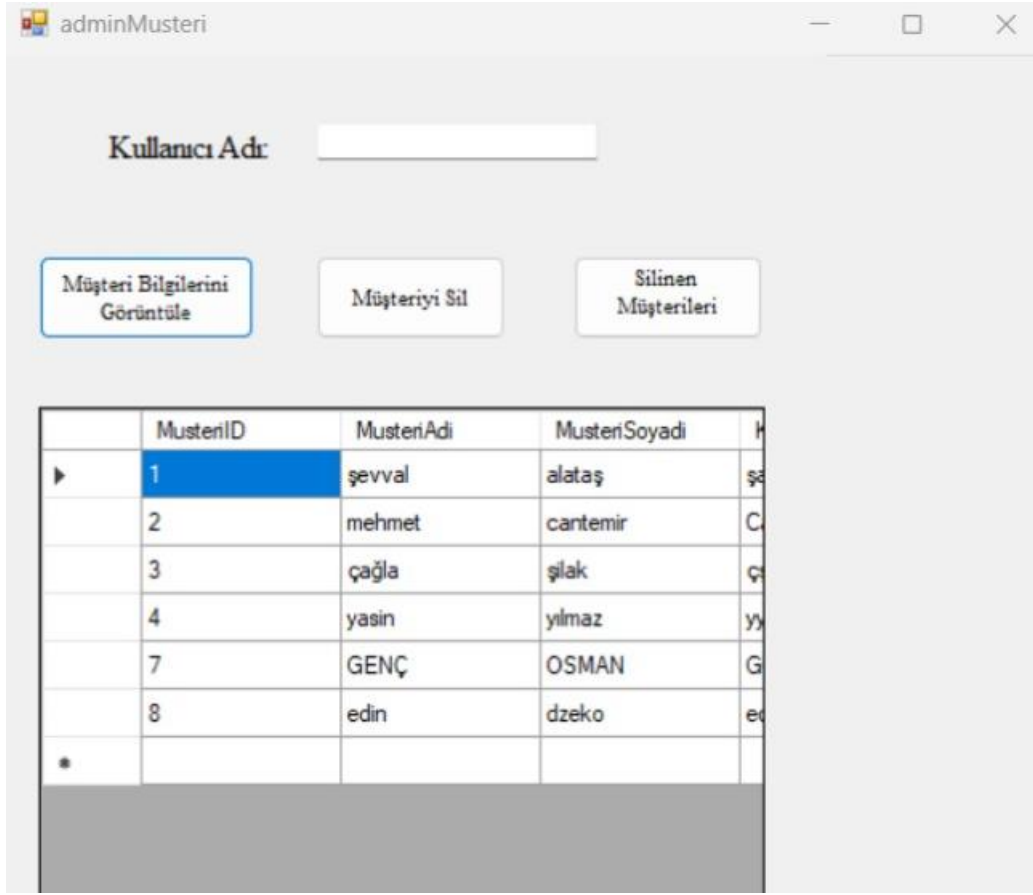


4.5.2.2. Müşteri İşlemleri

Müşteri işlemlerine tıkladığımızda kullanıcı adı ile giriş yapıp, Müşteri bilgilerini görüntüleyebilir, müşteriyi silebilir ve silinen müşterileri görebiliriz.



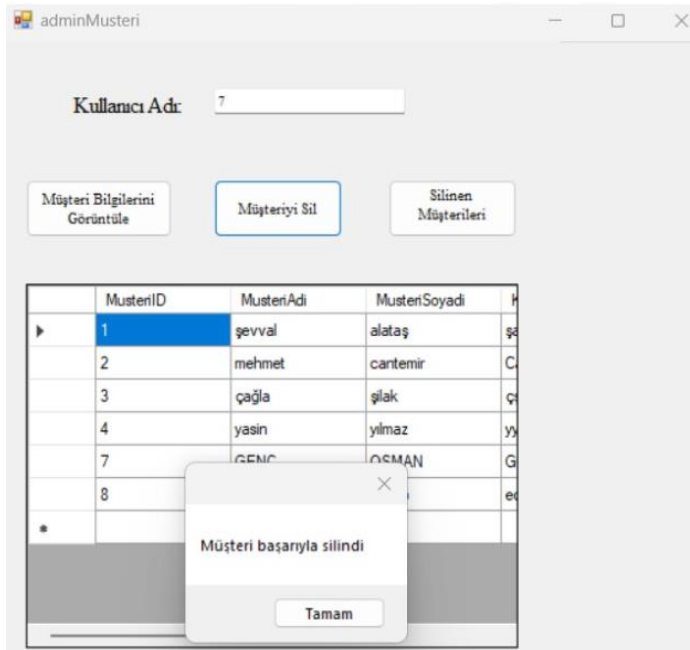
Müşteri Bilgilerini Görüntüle butonuna tıkladığımızda müşteri bilgileri görüntülenir.



The screenshot shows the 'adminMusteri' application window. At the top, there is a 'Kullanıcı Adı:' label followed by an empty text input field. Below this, there are three buttons: 'Müşteri Bilgilerini Görüntüle' (highlighted with a blue border), 'Müşteriyi Sil', and 'Silinen Müşterileri'. Below the buttons is a table with the following data:

	MusteriID	MusteriAdi	MusteriSoyadi	
▶	1	şevval	alataş	ş
	2	mehmet	cantemir	C
	3	çağla	şlak	Ç
	4	yasin	yılmaz	Y
	7	GENÇ	OSMAN	G
	8	edin	dzeko	e
*				

Müşteri sil butonuna tıkladığımızda müşteri silinir.



The screenshot shows the 'adminMusteri' application window. The 'Kullanıcı Adı:' field now contains the number '7'. The 'Müşteriyi Sil' button is highlighted with a blue border. Below the buttons is the same table as in the previous screenshot. A modal dialog box is open in the foreground with the text 'Müşteri başarıyla silindi' and a 'Tamam' button.

Silinen müşteri butonuna tıkladığımızda silinen müşteri bilgileri görüntülenir.

adminMusteri

Kullanıcı Adı: 7

Müşteri Bilgilerini Görüntüle

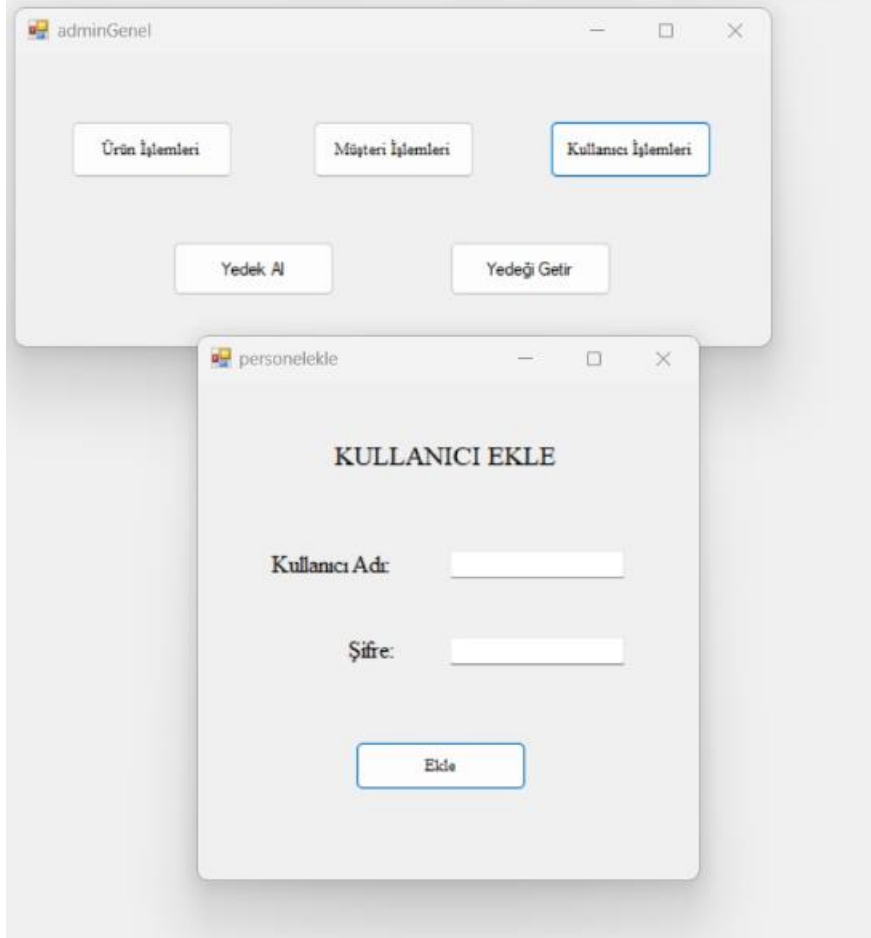
Müşteriyi Sil

Silinen Müşterileri

	MusteriID	SilinenMusteriID	Silinme Tarihi
►	6	1	18.12.2023
	5	2	24.12.2023
	7	3	24.12.2023
*			

4.5.2.3. Kullanıcı İşlemleri

Kullanıcı işlemleri butonuna tıkladığımızda Kullanıcı Ekle sayfası geliyor ve kullanıcı adı ile şifre girildiğinde ekle butonuna tıkladığımızda kullanıcı başarılı bir şekilde eklenir.



4.5.2.4. Yedek Al

Yedek Al butonuna tıkladığımızda BACKUP alınır. Bu işlemi sadece admin sayfasında adminler yapabilir.

Backup almanın SQL kodu,

```
BACKUP DATABASE [OnlineAlisveriVT2] TO DISK  
='C:\Users\mehme\Desktop\yedek.bak'
```

Backup almanın Stored Procedures'i,

```
CREATE PROCEDURE BackupAl
```

```
AS
```

```
BEGIN
```

```
    SET NOCOUNT ON;
```

```
    DECLARE @BackupPath NVARCHAR(255)
```

```
    SET @BackupPath = 'C:\Users\mehme\Desktop\yedek.bak'
```

```
    DECLARE @DatabaseName NVARCHAR(255)
```

```
    SET @DatabaseName = 'OnlineAlisveriVT2'
```

```
    DECLARE @BackupStatement NVARCHAR(MAX)
```

```
    SET @BackupStatement = 'BACKUP DATABASE [' + @DatabaseName + '] TO  
DISK =''' + @BackupPath + '''
```

```
    EXEC sp_executesql @BackupStatement
```

```
END
```

4.5.2.5. Yedeği Getir

Alınan BACKUP'ı getirir. Bu işlemi de sadece admin gerçekleştirir.

Backup geri yüklemenin SQL kodu,

```
USE master
```

```
GO
```

```
ALTER DATABASE [OnlineAlisveriVT2] set SINGLE_USER WITH ROLLBACK  
IMMEDIATE
```

```
RESTORE DATABASE [OnlineAlisveriVT2] FROM DISK =  
'C:\Users\mehme\Desktop\yedek.bak'
```

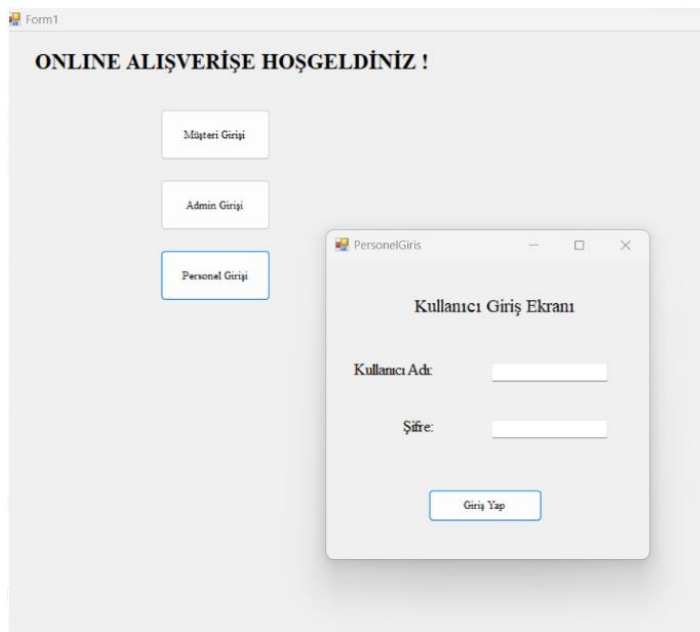
```
ALTER DATABASE [OnlineAlisveriVT2] set MULTI_USER
```

Backup geri yüklemenin Stored Procedures'i,

```
CREATE PROCEDURE dbo.usp_BackupDatabase  
  
AS  
  
BEGIN  
  
    SET NOCOUNT ON;  
  
    DECLARE @BackupPath NVARCHAR(255)  
  
    SET @BackupPath = 'C:\Users\mehme\Desktop\yedek.bak'  
  
    DECLARE @DatabaseName NVARCHAR(255)  
  
    SET @DatabaseName = 'OnlineAlisveriVT2'  
  
    DECLARE @BackupStatement NVARCHAR(MAX)  
  
    SET @BackupStatement = 'BACKUP DATABASE [' + @DatabaseName + '] TO  
DISK =''' + @BackupPath + ''''  
  
    EXEC sp_executesql @BackupStatement  
  
END
```

4.5.3 Personel Giriş

Personel Giriş butonuna tıkladığımızda Kullanıcı Giriş Ekranı gelir ve burada kullanıcı adını ve şifreyi girdiğimizde giriş yaparız.



The screenshot displays a web application interface. At the top, a banner reads "ONLINE ALIŞVERİŞE HOŞGELDİNİZ !". Below this, there are three buttons: "Müşteri Girişi", "Admin Girişi", and "Personel Girişi". The "Personel Girişi" button is highlighted with a blue border. A modal window titled "Kullanıcı Giriş Ekranı" is open in the foreground. It contains two input fields: "Kullanıcı Adı" and "Şifre", and a "Giriş Yap" button at the bottom.

Giriş Yap butonuna tıkladıktan sonra Personel Genel ekranı karşımıza çıkar. Burada Ürün işlemleri ve Müşteri işlemleri butonları bulunur.

The image shows a web application interface. The main window, titled 'Form1', displays a welcome message 'ONLINE ALIŞVERİŞE HOŞGELDİNİZ !' and three login buttons: 'Müşteri Girişi', 'Admin Girişi', and 'Personel Girişi'. The 'Personel Girişi' button is highlighted with a blue border. Overlaid on this is a smaller window titled 'PersonelGiris' with the subtitle 'Kullanıcı Giriş Ekranı'. This window contains two input fields: 'Kullanıcı Adı' (Username) with the text 'memo' and 'Şifre:' (Password) with four asterisks. A 'Giriş Yap' (Login) button is at the bottom. Below the 'PersonelGiris' window is another window titled 'PersonelGenel'. This window contains two buttons: 'Ürün İşlemleri' (Product Operations) and 'Müşteri İşlemleri' (Customer Operations).

Ürün işlemleri butonuna tıkladığımızda, admin sayfasında önümüze gelen işlemlerin aynısını yapar aynı şekilde müşteri işlemleri butonuna tıkladığımızda da admin sayfasında bulunan müşteri işlemlerinin aynısına personel de erişebilir.

Ek olarak admin personel ve müşteri girişlerinde veritabanında bulunan kullanıcı giriş ve şifrelerin girilmemesi durumunda hata mesajı alınır.

