

Üniversite Kütüphane Yönetim Sistemi

Veritabanı Yönetim Sistemleri – Final Projesi Raporu -> **Habib Sultanı [22061001]**

Github linki: https://github.com/Habibsultani/LMS_project.git

Youtube linki: https://youtu.be/rGDiv6uhc_U

Bu projede, Kütüphane Yönetim Sistemi (Library Management System – LMS) geliştirilmiştir.

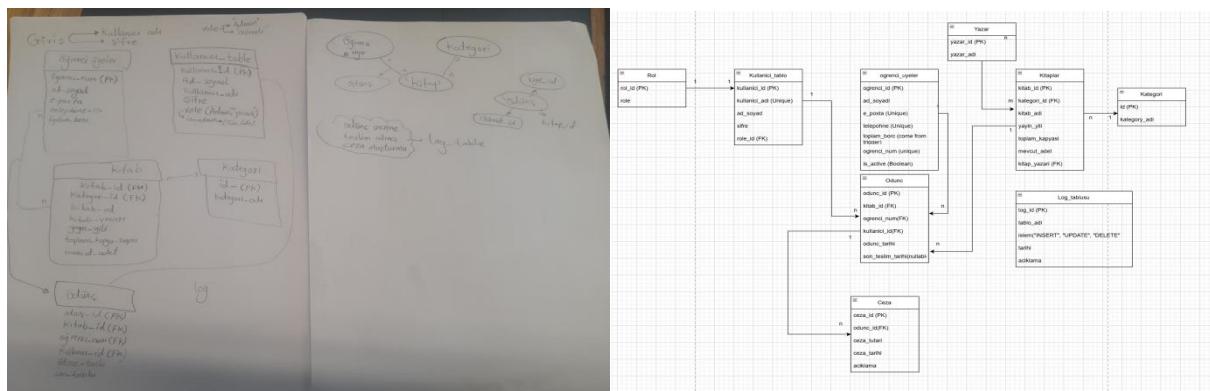
Projenin amacı; kitap, üye (öğrenci), ödünç verme, ceza ve raporlama işlemlerinin ilişkisel bir veritabanı üzerinde güvenli, tutarlı ve verimli bir şekilde yönetilmesini sağlamaktır.

Proje kapsamında:

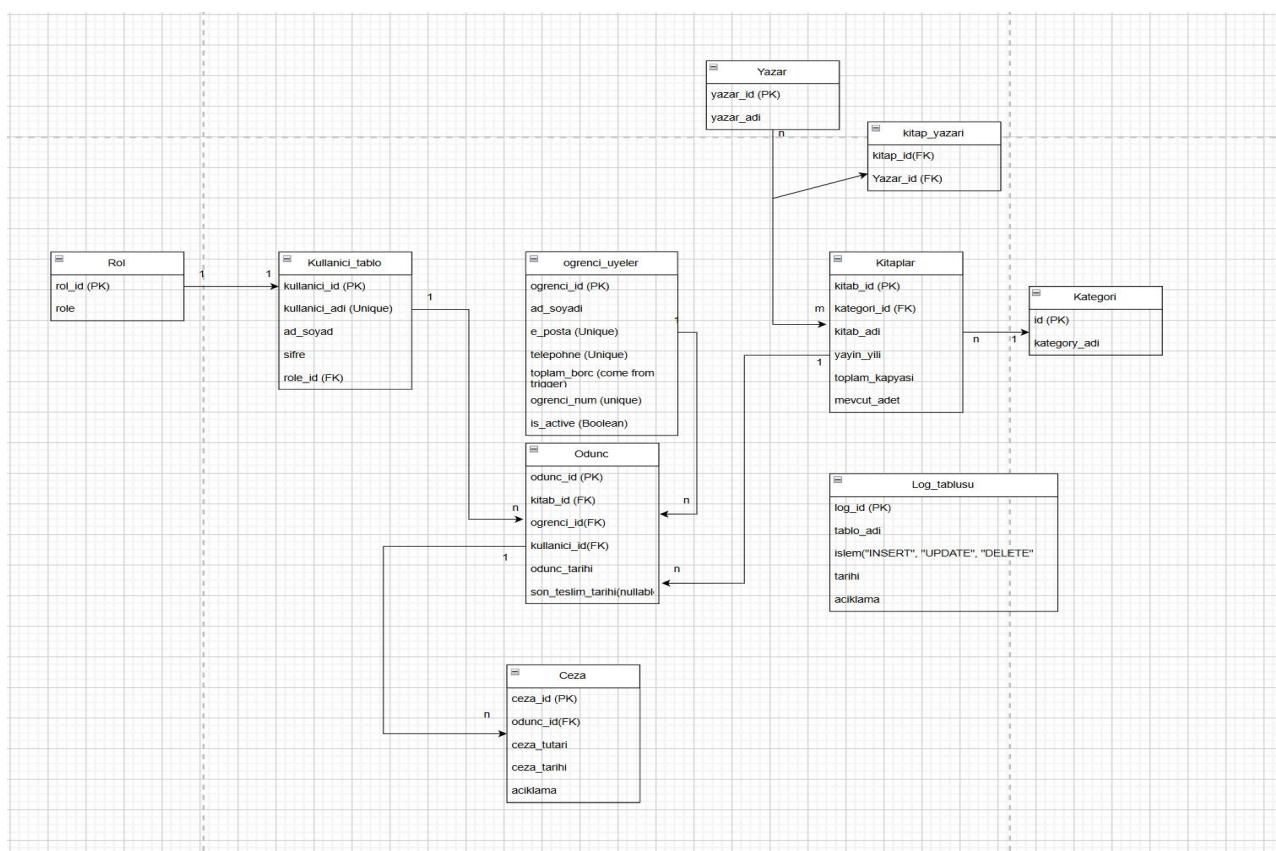
- İlişkisel veritabanı tasarımları yapılmış,
- Birincil anahtar (Primary Key) ve yabancı anahtar (Foreign Key) ilişkileri kurulmuş,
- Saklı yordamlar (Stored Procedures) ve tetikleyiciler (Triggers) kullanılmış,
- C# WinForms tabanlı bir masaüstü uygulaması geliştirilmiştir.

Proje geliştirme süreci aşamalı (stage-based) olarak ilerlemiştir. Her aşamada veritabanı tasarımları iyileştirilmiş ve akademik DBMS kurallarına uygun hale getirilmiştir.

1. Aşama – İlk Tasarım (Normalize Edilmemiş Model)



2. Aşama Final formatta normalizasyon sonrası tabular 3NF hali



3. Aşama İlişkisel şemaya dönüştürme ve veritabanı tablolarının oluşturulması için hazırlanması.

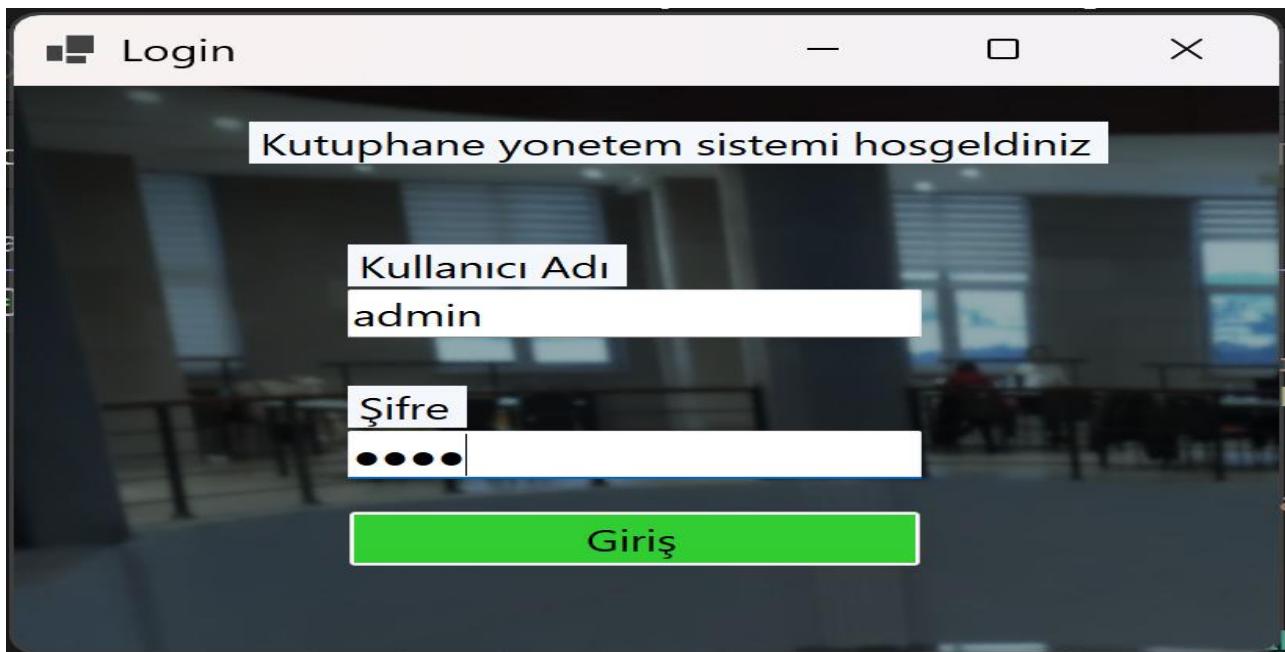
```
Rol(  
    rol_id INT PRIMARY KEY,  
    rol_adi VARCHAR(50) NOT NULL  
)  
  
KULLANICI_TABLO(  
    kullanici_id INT PRIMARY KEY,  
    kullanici_adi VARCHAR(50) NOT NULL UNIQUE,  
    ad_soyad VARCHAR(100) NOT NULL,  
    sifre VARCHAR(100) NOT NULL,  
    rol_id INT NOT NULL,  
    FOREIGN KEY (rol_id) REFERENCES ROL(rol_id)  
)  
  
OGRENCI_UYELER(  
    ogrenci_id INT PRIMARY KEY,  
    ogrenci_num VARCHAR(20) NOT NULL UNIQUE,  
    ad_soyadi VARCHAR(100) NOT NULL,  
    e_posta VARCHAR(100) UNIQUE,  
    telefon VARCHAR(20),  
    toplam_borc DECIMAL(10,2) DEFAULT 0,  
    is_active BOOLEAN DEFAULT TRUE  
)  
  
KATEGORI(  
    kategori_id INT PRIMARY KEY,  
    kategori_adi VARCHAR(100) NOT NULL UNIQUE  
)  
  
CEZA(  
    ceza_id INT PRIMARY KEY,  
    odunc_id INT NOT NULL,  
    ceza_tutari DECIMAL(10,2) NOT NULL,  
    ceza_tarihi DATE NOT NULL,  
    aciklama VARCHAR(255),  
    FOREIGN KEY (odunc_id) REFERENCES ODUNC(odunc_id)  
)  
  
KITAPLAR(  
    kitab_id INT PRIMARY KEY,  
    kategori_id INT NOT NULL,  
    kitab_adi VARCHAR(200) NOT NULL,  
    yayin_yili INT,  
    toplam_kopyasi INT NOT NULL,  
    mevcut_adet INT NOT NULL,  
    FOREIGN KEY (kategori_id) REFERENCES KATEGORI(kategori_id)  
)  
  
YAZAR(  
    yazar_id INT PRIMARY KEY,  
    yazar_adi VARCHAR(100) NOT NULL  
)  
  
KITAP_YAZARI(  
    kitab_id INT,  
    yazar_id INT,  
    PRIMARY KEY (kitab_id, yazar_id),  
    FOREIGN KEY (kitab_id) REFERENCES KITAPLAR(kitab_id),  
    FOREIGN KEY (yazar_id) REFERENCES YAZAR(yazar_id)  
)  
  
ODUNC(  
    odunc_id INT PRIMARY KEY,  
    kitab_id INT NOT NULL,  
    ogrenci_id INT NOT NULL,  
    kullanici_id INT NOT NULL,  
    odunc_tarihi DATE NOT NULL,  
    son_teslim_tarihi DATE NOT NULL,  
    teslim_tarihi DATE,  
    FOREIGN KEY (kitab_id) REFERENCES KITAPLAR(kitab_id),  
    FOREIGN KEY (ogrenci_id) REFERENCES OGRENCI_UYELER(ogrenci_id),  
    FOREIGN KEY (kullanici_id) REFERENCES KULLANICI_TABLO(kullanici_id)  
)  
  
LOG_TABLOSU(  
    log_id INT PRIMARY KEY,  
    tablo_adi VARCHAR(50) NOT NULL,  
    islem VARCHAR(10) NOT NULL,  
    tarih TIMESTAMP NOT NULL,  
    aciklama VARCHAR(255)  
)
```

4. aşamada veritabanı tasarımı tamamlanmış ve SQL dosyaları database dizini altında düzenli bir yapıda hazırlanmıştır.

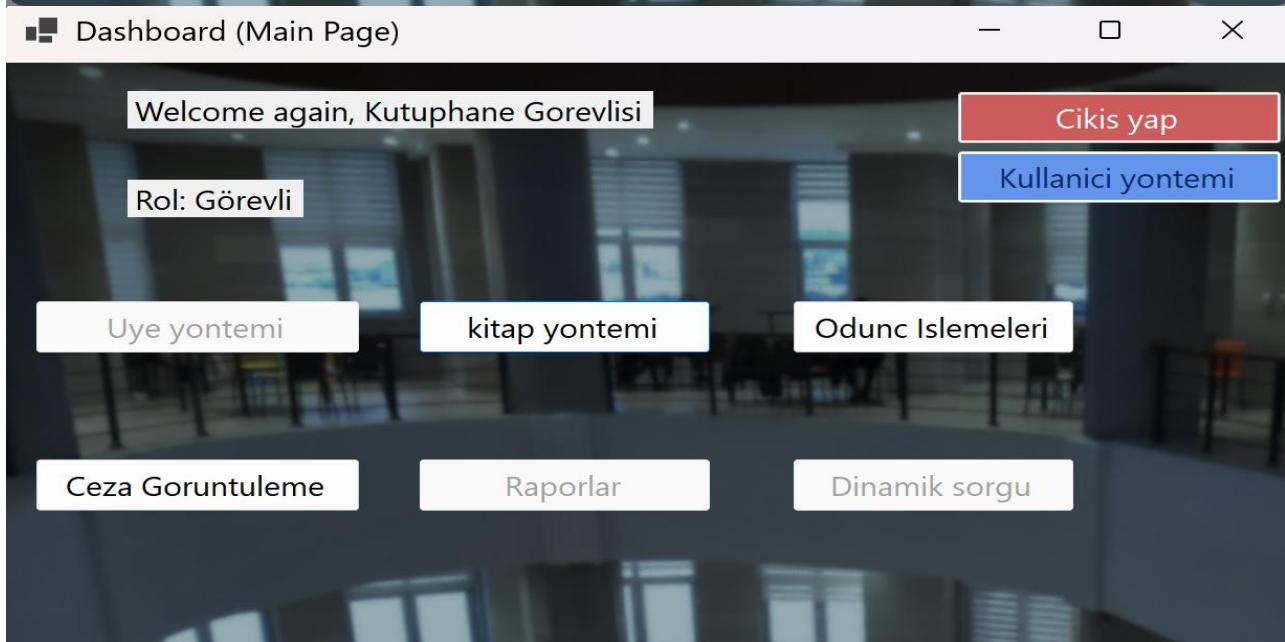
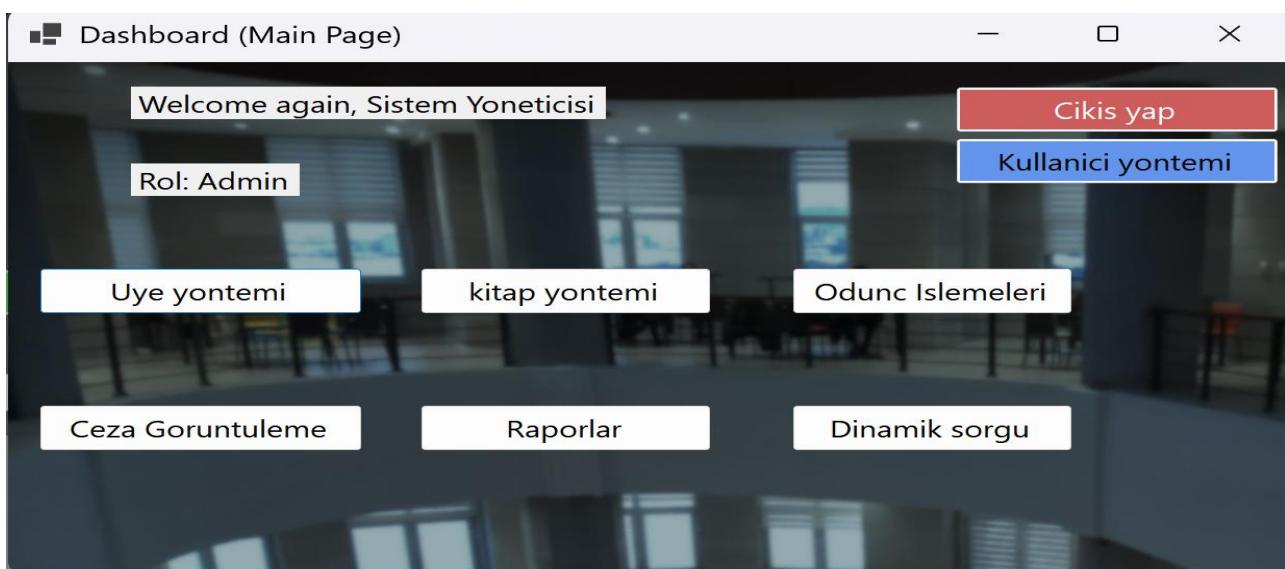
Bu kapsamda; 00_create_db.sql dosyası veritabanının oluşturulması için, 01_tables.sql dosyası tablo tanımları için, 02_procedures.sql dosyası saklı yordamlar için, 03_triggers.sql dosyası tetikleyiciler için ve 04_seed_data.sql dosyası sahte (seed) verilerin eklenmesi için kullanılmıştır.

4. aşamada veritabanı tasarlandıktan sonra C# WinForms kullanılarak uygulama geliştirilmiştir. Bu aşamadan sonra, uygulamaya ait her bir ekran görüntüsü için detaylı açıklamalar verilecektir.

5.1 Bu birinci sayfa, kullanıcı giriş (login) ekranıdır. Kullanıcılar, kendilerine ait kullanıcı ID ve şifre bilgilerini girerek uygulamaya erişim sağlayabilir.



5.2 Bu sayfa ana sayfa (**dashboard**) ekranıdır. **Admin** kullanıcılar tüm işlemleri yapabilirken, **görevli** kullanıcılar **kullanıcı yönetimi** ve **üye yönetimi** gibi bazı yetkili işlevlere erişemez.



5.3: Bu sayfa **kullanıcı yönetimi** sayfasıdır. Bu ekrana yalnızca **admin** yetkisine sahip kullanıcılar erişebilir; görevli kullanıcıların erişimi kısıtlanmıştır. **Admin** yetkisine sahip kullanıcılar bu sayfa üzerinden yeni kullanıcı ekleyebilir ve kullanıcıların rollerini (**admin/görevli**) belirleyebilir.

User management page

Yeni bir kullanıcı ekleme sayfa

Ana sayfaya

kullanici adi

Ad soyad

Sifre

Role

Admin

Ekle

Temizle

5.4: Bu sayfa **üye yönetimi (Student Management)** sayfasıdır. Bu ekrana yalnızca admin yetkisine sahip kullanıcılar erişebilir. Admin kullanıcılar yeni **üye (öğrenci)** ekleyebilir, öğrencileri **ad-soyad** ve **e-posta bilgilerine göre arayabilir**. Ayrıca, **toplam borcu sıfır olan** seçili öğrenciler **sistemden silinebilir**.

Student management page

User management page

Ogrenci Number

Ad soyadi

E-posta adresi

telephon numarası

Ekle

Guncelle

Sil

Temizle

Ad soyad

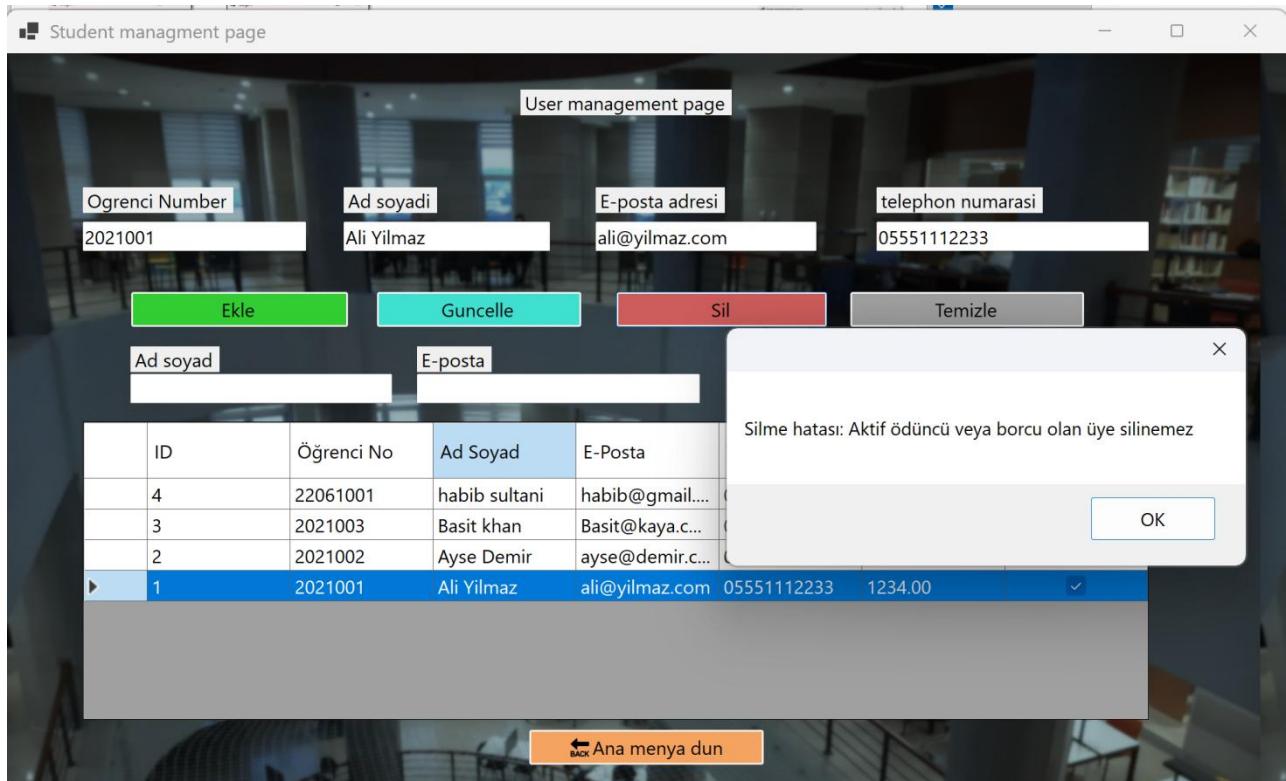
E-posta

Ogrenci ara

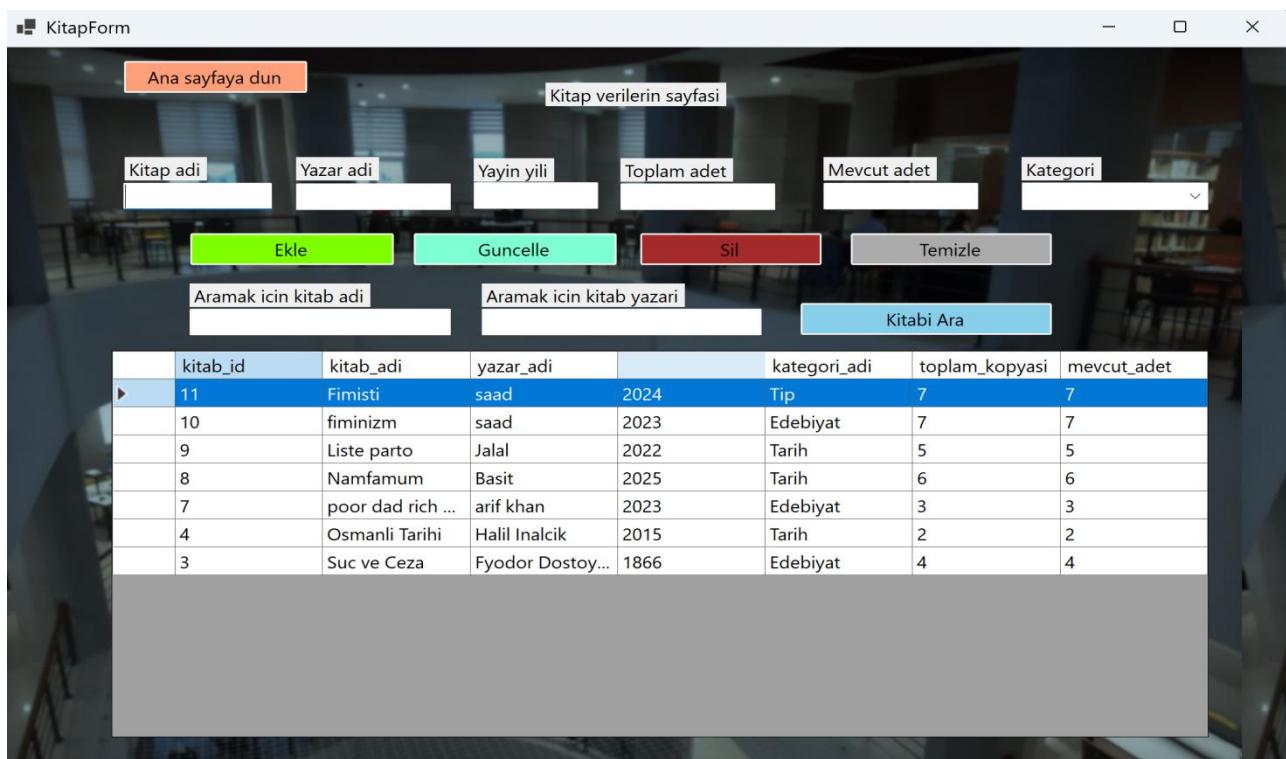
ID	Öğrenci No	Ad Soyad	E-Posta	Telefon	Toplam Borç	Aktif
4	22061001	habib sultani	habib@gmail....	055555555555	0.00	<input checked="" type="checkbox"/>
3	2021003	Basit khan	Basit@kaya.c...	05553334455	0.00	<input checked="" type="checkbox"/>
2	2021002	Ayse Demir	ayse@demir.c...	05552223344	0.00	<input checked="" type="checkbox"/>
1	2021001	Ali Yilmaz	ali@yilmaz.com	05551112233	1234.00	<input checked="" type="checkbox"/>

BACK Ana menya dun

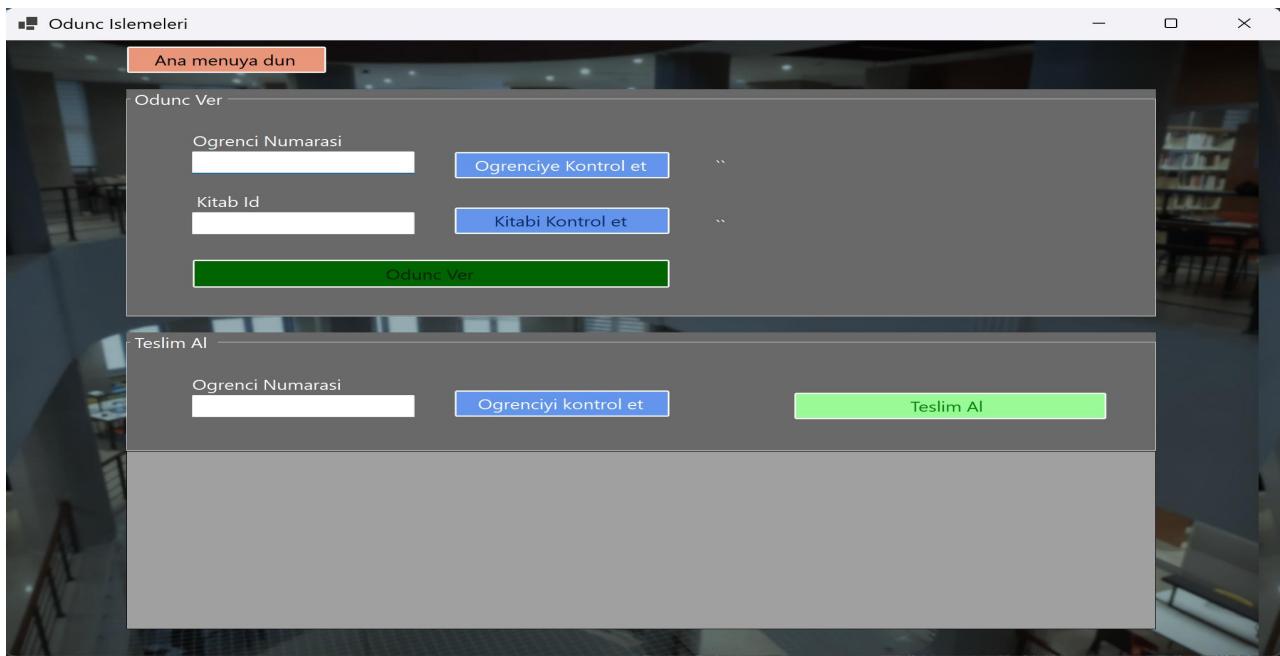
5.4.1:TR_UYE_DELETE_BLOCK trigger'ı burada çalıştı. Ali Yılmaz'ın borcu sıfırdan büyük olduğu için silme işlemi başarısız oldu.



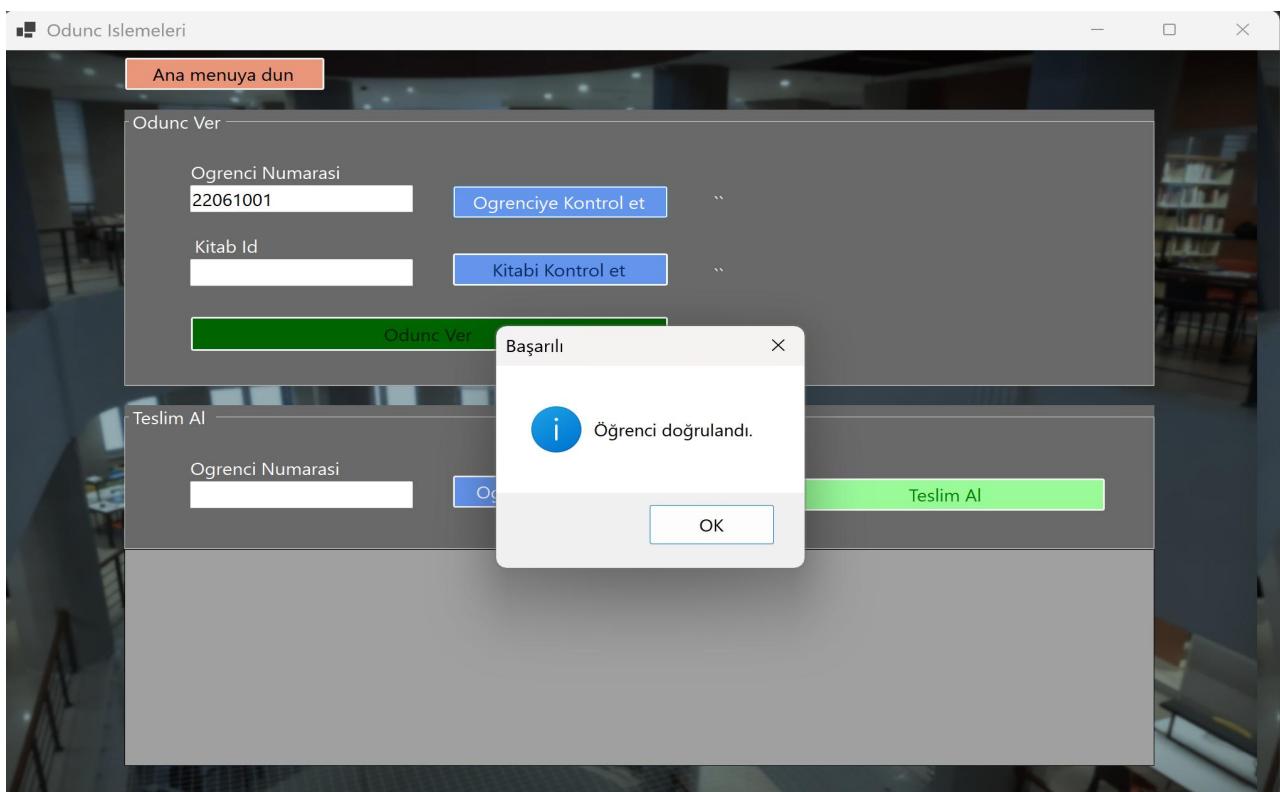
5.6:Bu sayfa **kitap yönetimi** sayfasıdır. Bu ekrana hem **admin** hem de **görevli** kullanıcılar erişebilir. Kullanıcılar bu sayfa üzerinden yeni **kitap ekleyebilir**, seçilen kitaplari **güncelleyebilir ve silebilir**. Kitap ekleme işlemi sırasında yazar, kategori ve kitap olmak üzere üç farklı tabloya veri aktarımı yapılmaktadır. Ayrıca kitaplar, **kitap adı** ve **yazar adına** göre **aranabilmektedir**. Bununla birlikte, yeni bir **ödünç verme** işlemi gerçekleştirildiğinde ilgili **stored procedure** çalışarak kitabın mevcut adet sayısını 1 azaltmaktadır.

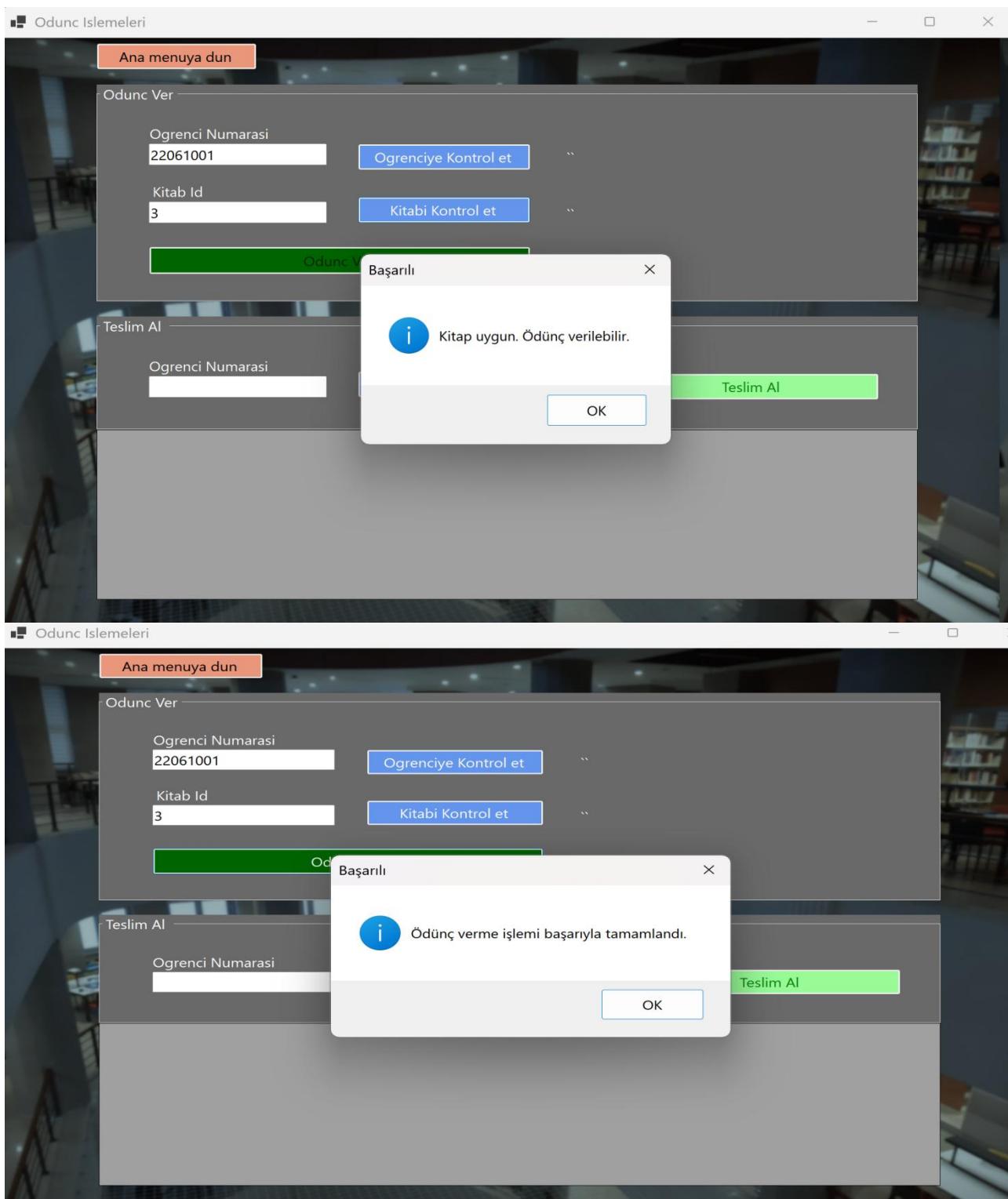


5.7: Ödünç verme sayfasında hem **ödünç verme** hem de **teslim alma** işlemleri gerçekleştirilmektedir.

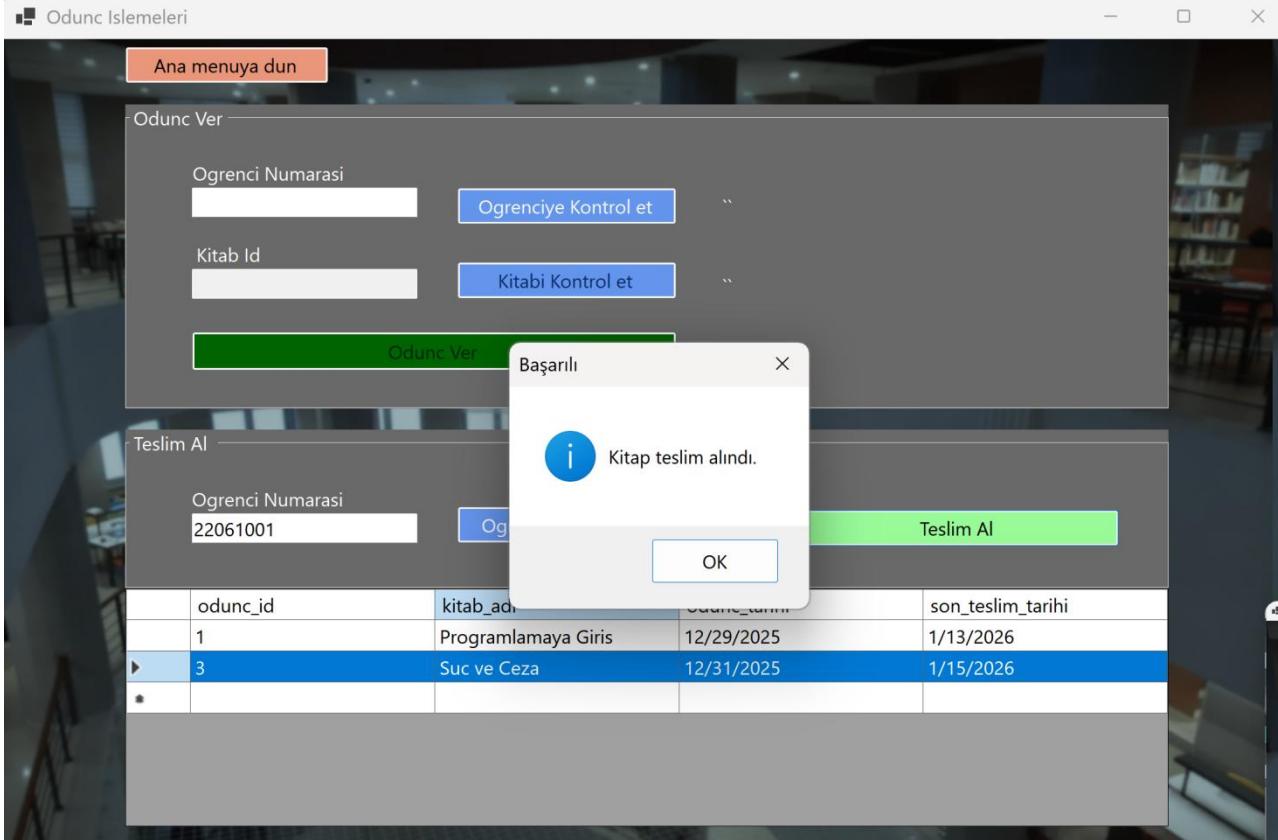
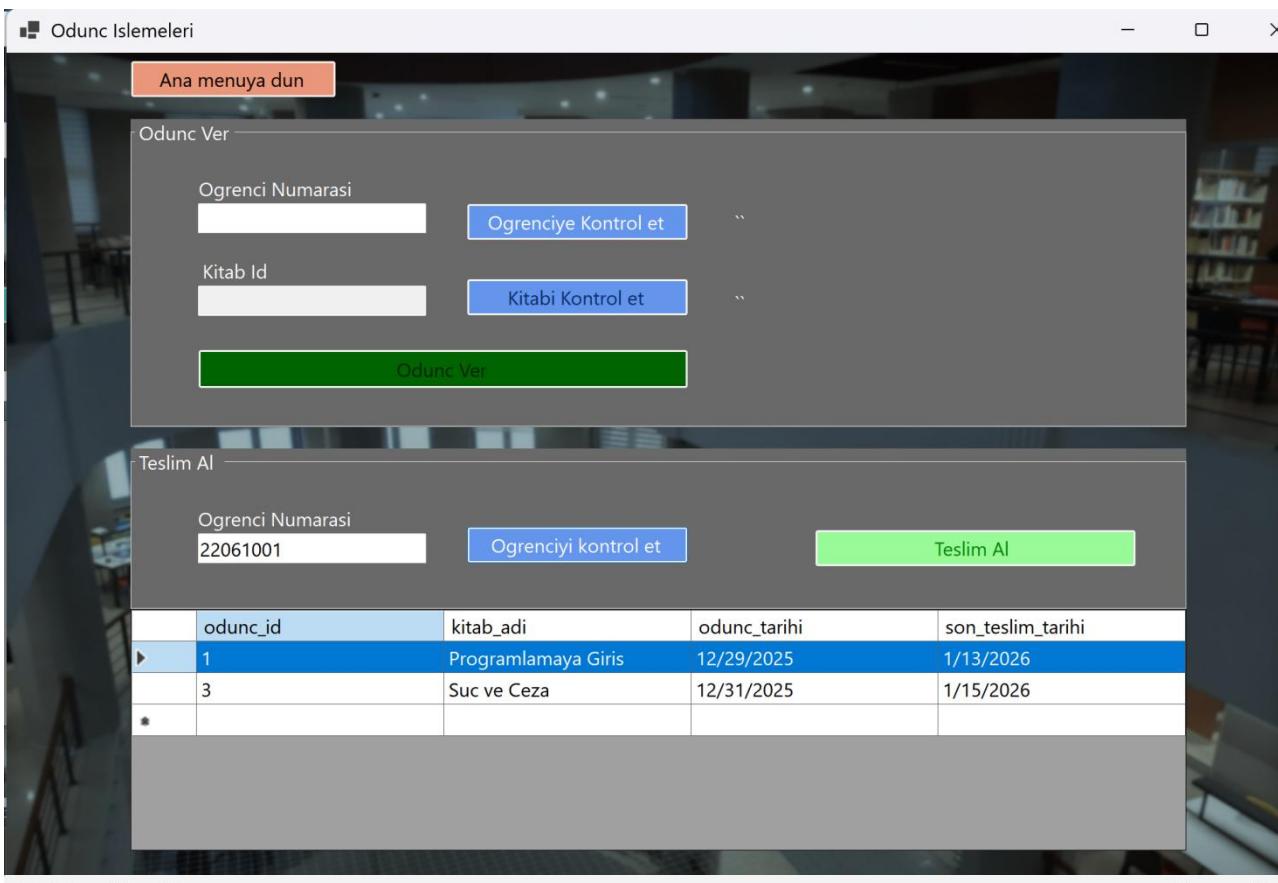


5.7.1:Ödünç verme işleminde ilk olarak öğrencinin durumu kontrol edilir. Öğrencinin sistemde kayıtlı ve aktif olup olmadığı sorgulanır; **aktif olmayan veya kaydı bulunmayan öğrencilere ödünç verme işlemi yapılmaz.** Daha sonra **kitap kontrolü** gerçekleştirilir ve stokta mevcut adet olup olmadığına bakılır. Kitap stokta mevcut ise '**Ödünç Ver**' butonu **aktif hale gelir** ve ödünç verme işlemi gerçekleştirilebilir. Aksi durumda sistem, **ilgili hata mesajlarını** kullanıcıya göstermektedir. Bu süreçte ödünç verme işlemi, **sp_YeniOduncVer** adlı stored procedure kullanılarak gerçekleştirilmektedir. ayrıca **TR_UDUNC_INSERT trigger'ı** devreye girerek gerekli güncellemleri gerçekleştirir.

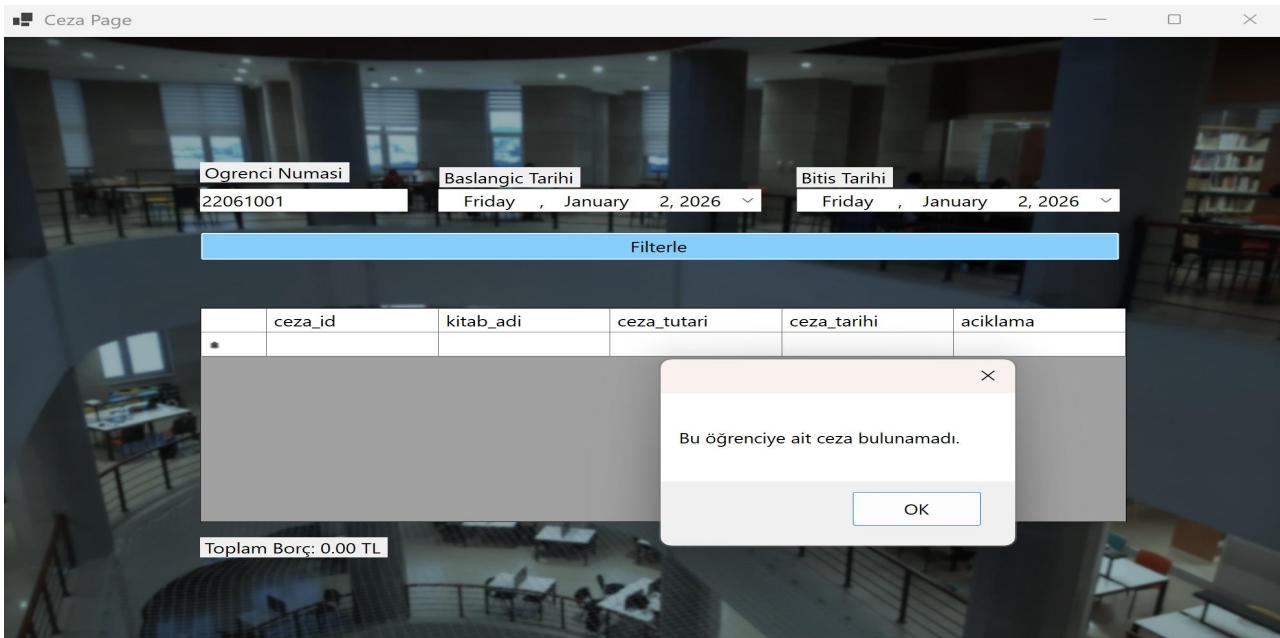




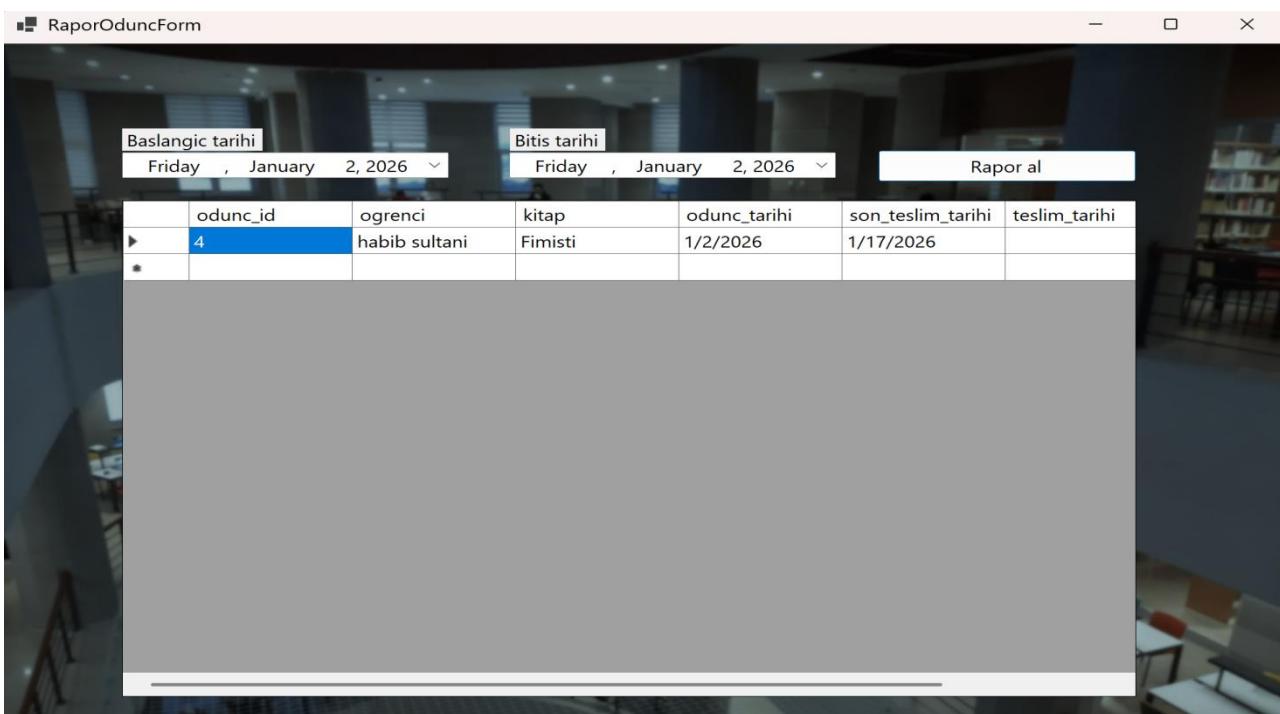
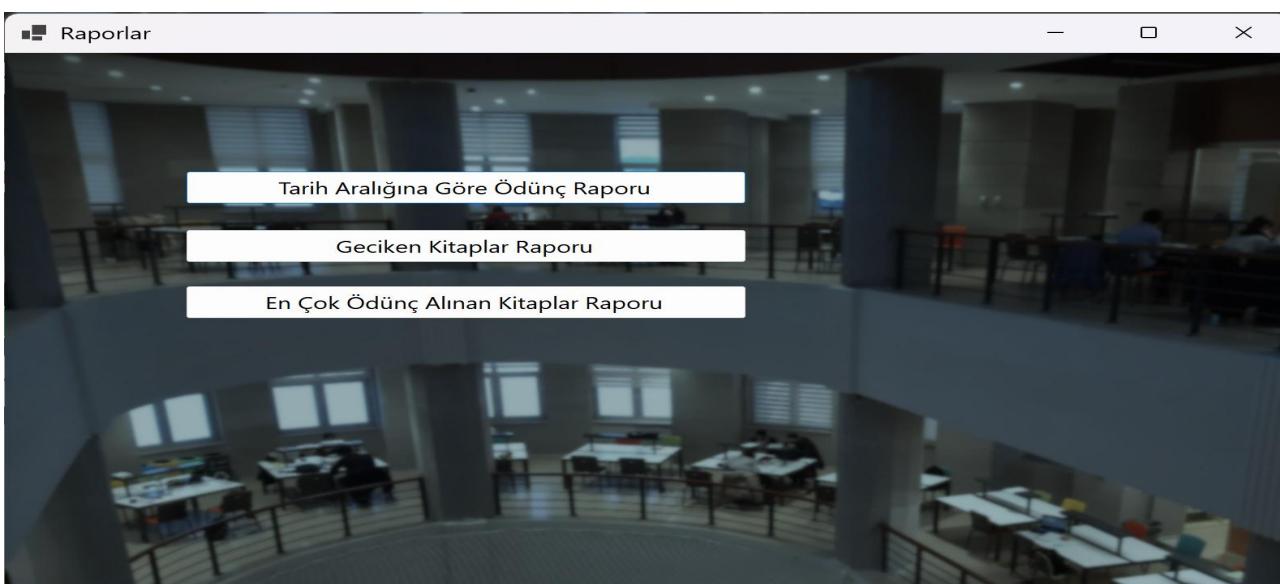
5.8: Teslim alma işleminde görevli, ilk olarak öğrencinin **öğrenci numarasını girer**. Sistem, öğrenciyi bularak kayıtlı ve aktif olması durumunda **öğrencinin almış olduğu ödünçleri listeler**. Listelenen ödünçler arasından seçilen ödünç için '**Teslim Al**' butonuna tıklanlığında **sp_KitapTeslimAl** adlı **stored procedure** çağrıılır ve teslim alma işlemi gerçekleştirilir. Öğrencinin aktif olmaması veya sistemde kayıtlı olmaması durumunda '**Teslim Al**' butonu aktif hale gelmez, işlem yapılmaz ve sistem ilgili hata mesajını üretir. Teslim alma işlemi sırasında ayrıca **TR_ODUNC_UPDATE_TESLIM trigger'ı** devreye girerek gerekli güncellemeleri gerçekleştirir.

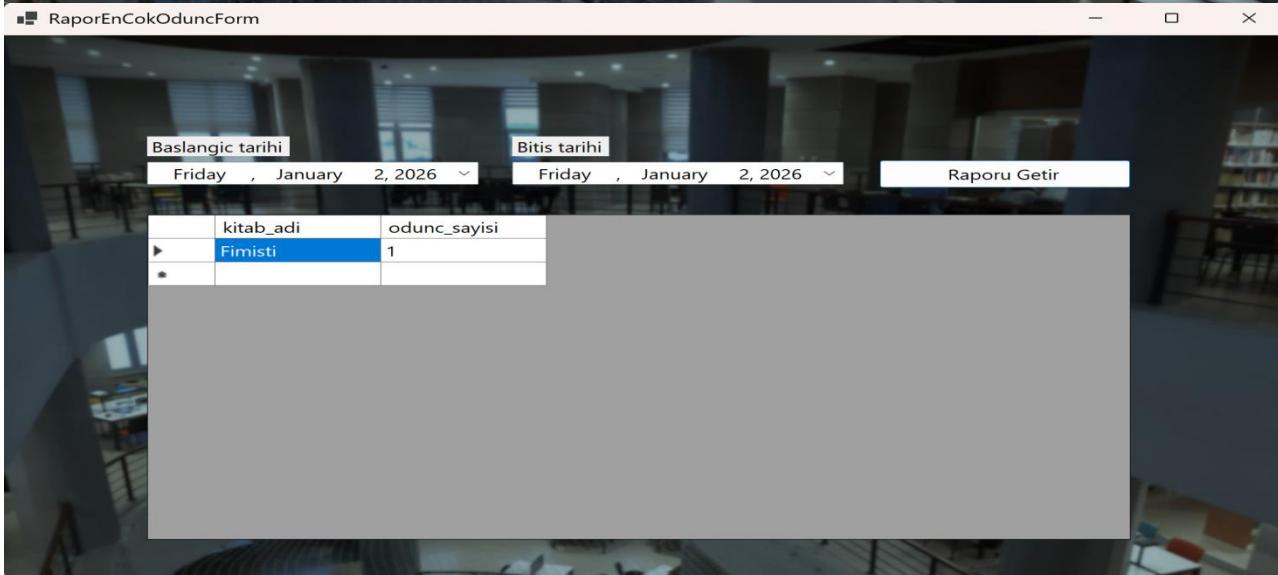
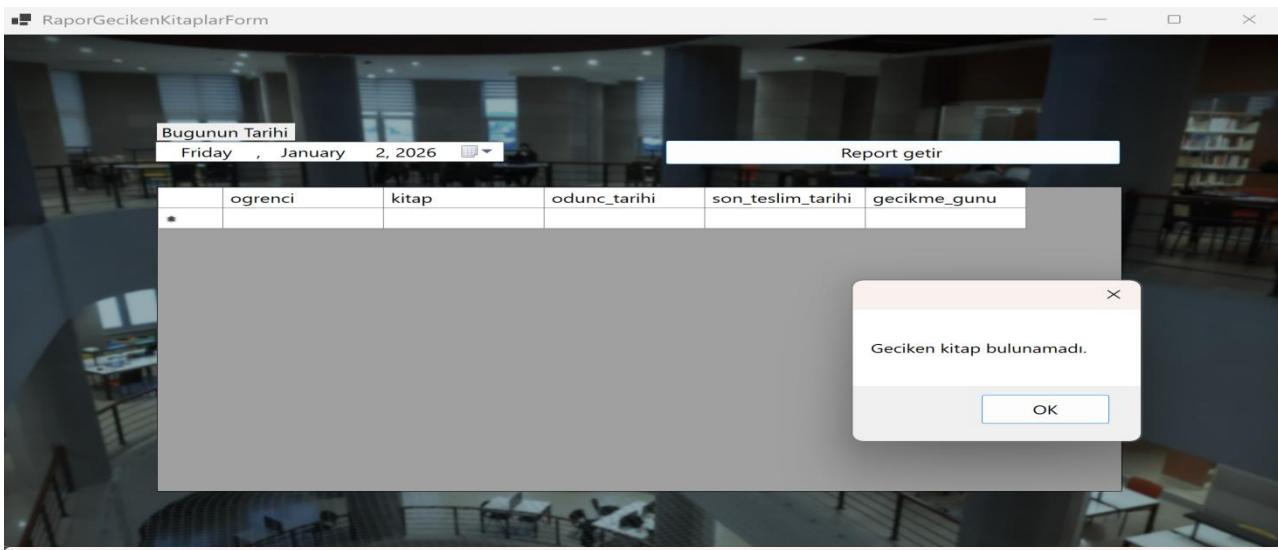


5.9:Bu sayfa **ceza görüntüleme** sayfasıdır. Öğrenci numarası girilerek ilgili öğrencinin cezaları görüntülenir. Cezalar, belirlenen tarih aralığına göre filtrelenebilir ve öğrencinin **toplam borç** tutarı sistem tarafından hesaplanarak ekranda gösterilir.

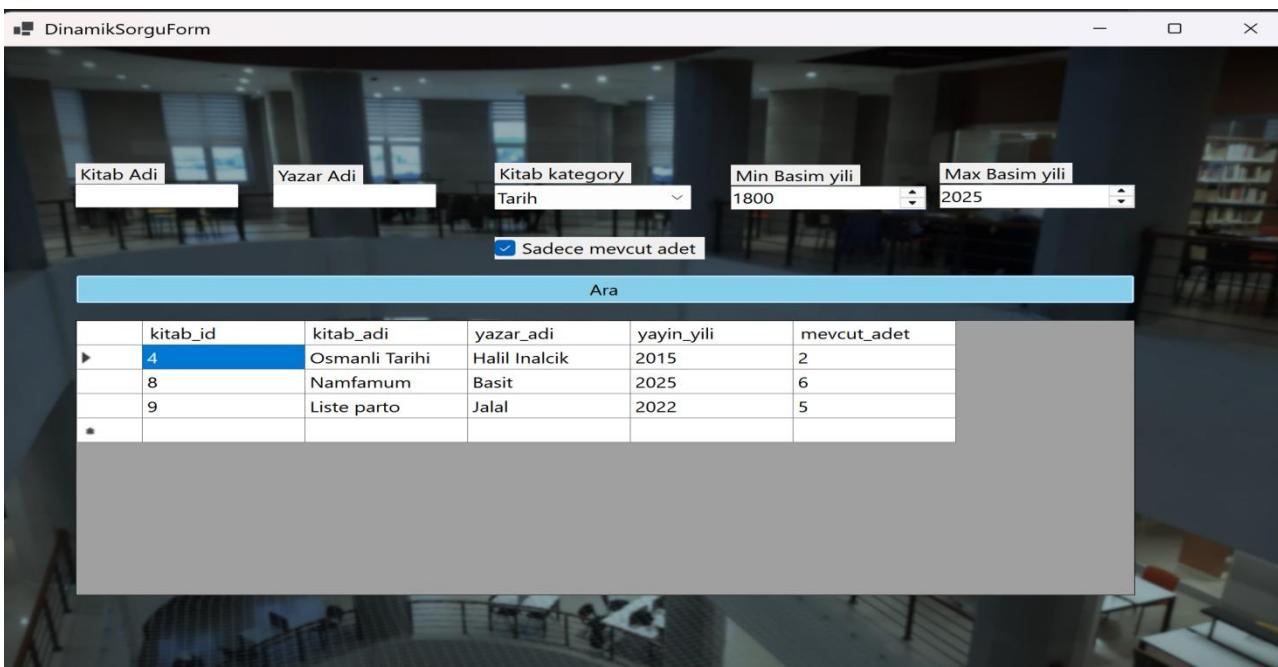


5.10: Raporlar sayfasında üç farklı rapor oluşturulmaktadır. Bunlar; tarih aralığına göre ödünç raporu, geciken kitaplar raporu ve en çok ödünç alınan kitaplar raporudur.





5.11: Dinamik Soru ekranı, kullanıcının **birden fazla filtreyi isteğe bağlı olarak seçebilmesine imkân tanıyan, parametreli ve **esnek** bir **sorgu** yapısı sunmaktadır. Boş bırakılan alanlar soruya dahil edilmezken, yalnızca doldurulan kriterler SQL sorgusuna dinamik olarak eklenmekte ve sonuçlar tablo üzerinde listelenmektedir.**



Bu projede log tablosu kullanılarak sistemde gerçekleşen tüm işlemler kayıt altına alınmaktadır. Trigger ve stored procedure'ler tarafından gerçekleştirilen tüm işlemler log tablosuna yazılarak izlenebilirlik ve denetim sağlanmaktadır.

		log_id	tablo_adi	islem	tarih	aciklama
<input type="checkbox"/>	Edit Copy Delete	1	ODUNC	INSERT	2025-12-29 12:41:02	Yeni ödünç verildi. OduncID=1
<input type="checkbox"/>	Edit Copy Delete	2	ODUNC	INSERT	2025-12-29 14:46:40	Yeni ödünç verildi. OduncID=2
<input type="checkbox"/>	Edit Copy Delete	3	ODUNC	UPDATE	2025-12-29 15:30:05	Kitap teslim alındı. OduncID=2
<input type="checkbox"/>	Edit Copy Delete	4	ODUNC	INSERT	2025-12-31 21:45:02	Yeni ödünç verildi. OduncID=3
<input type="checkbox"/>	Edit Copy Delete	5	ODUNC	UPDATE	2025-12-31 21:45:38	Kitap teslim alındı. OduncID=3
<input type="checkbox"/>	Edit Copy Delete	6	ODUNC	INSERT	2026-01-02 20:03:03	Yeni ödünç verildi. OduncID=4