



ONDOKUZ MAYIS ÜNİVERSİTESİ  
MÜHENDİSLİK FAKÜLTESİ  
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

# VERİTABANI LABORATUVARI

## FÖY 5 RAPOR

ZEYNEP SILA KAYMAK

20060358

DERSİN ÖĞRETMENİ

Dr.Öğr.Üyesi İsmail İŞERİ

MAYIS, 2024

---

## TEŞEKKÜR

---

Dersimizin öğretmeni sayın Dr.Öğr.Üyesi İsmail İŞERİ'ye ve dersimizin asistanı sayın Arş. Gör. Varol ŞAHİN'e teşekkürlerimi sunarım.

---

## Ö Z E T

---

Entity Framework, SQL Server ve C kullanarak geliştirdiğimiz uygulama, Code First yaklaşımını benimsemiştir. Uygulamamız, web tabanlı olarak tasarlanmıştır ve istenilen altı ekranı içermektedir.

İlk olarak, Ekran1’de öğrenci tablosu üzerinde CRUD işlemleri gerçekleştirilmiştir. Entity Framework kullanılarak öğrenci tablosuna erişim sağlanmış, yeni öğrenciler eklenmiş, var olan öğrenciler güncellenmiş ve silinmiştir.

Ekran2 ve Ekran3’te de aynı şekilde ders ve bölüm tabloları üzerinde CRUD işlemleri yapılmıştır. Bu işlemler, Entity Framework kullanılarak gerçekleştirilmiş ve ilgili tablolara yeni kayıtlar eklenmiş, mevcut kayıtlar güncellenmiş ve silinmiştir.

Ekran4, öğrencilere yıl ve yarıyıl seçimleri yapılarak ders atama, silme ve değiştirme işlemlerini içermektedir. Bu işlemler, öğrenci ve ders tabloları arasındaki ilişkiyi yönetmek için Entity Framework’in ilişkisel yeteneklerinden yararlanılarak gerçekleştirilmiştir.

Ekran5’te, öğrenci numarasına göre öğrencinin aldığı tüm derslerin listesi ve seçilen bir yıl ve yarıyıl bilgisine göre hangi dersin kaç öğrenci tarafından seçildiğini listeleme işlemleri yapılmıştır. Bu işlemler, Entity Framework’in LINQ sorgularıyla veri tabanından ilgili verilerin alınmasıyla sağlanmıştır.

Son olarak, Ekran6’da seçilen bir dersi alan öğrencilerin listelenmesi ve vize ile final notlarının girilerek kaydedilmesi işlemleri yapılmıştır. Bu işlemler, Entity Framework kullanılarak ders ve öğrenci tabloları arasındaki ilişki üzerinde gerçekleştirilmiştir.

Raporun geri kalanında, her bir ekranın detaylarıyla birlikte, kullanılan teknoloji, CRUD işlemlerinin nasıl yapıldığı ve yapılan işlemlerin hangi tablolarla ilişkili olduğu gibi detaylar verilmiştir.

---

## İÇİNDEKİLER

---

### I GİRİŞ

|        |   |
|--------|---|
| 1 Amaç | 2 |
|--------|---|

### II Yöntem

|          |    |
|----------|----|
| 2 SORU 1 | 5  |
| 3 SORU 2 | 10 |
| 4 SORU 3 | 14 |
| 5 SORU 4 | 18 |
| 6 SORU 5 | 22 |
| 7 SORU 6 | 25 |
| 8 SORU 7 | 28 |

### III SONUÇ

|         |    |
|---------|----|
| 9 Sonuç | 31 |
|---------|----|

---

## ŞEKİLLER LİSTESİ

---

|    |  |    |
|----|--|----|
| 1  | Uygulama açıldığında görünen ilk sayfa, uygulamanın anasayfası bu şekildedir. Kullanıcı istediği ekranın butonuna tıklayarak o ekranı görebilir. . . . . | 6  |
| 2  | Ekran 1 ise bu şekilde düzenlenmiştir. Kullanıcı bu sayfadan öğrenci ekleme, silme güncelleme ve listeleme işlemlerini yapabilir. . . . .                | 6  |
| 3  | Öğrenci Ekleme İşlemi . . . . .  | 7  |
| 4  | Öğrencinin eklendiğine dair bir uyarı mesajı . . . . .   | 7  |
| 5  | Eklenen öğrencinin listelenmesi . . . . .  | 7  |
| 6  | Öğrenci silme işlemi . . . . .   | 8  |
| 7  | Öğrencinin silindiğine dair bir uyarı mesajı . . . . .   | 8  |
| 8  | Öğrencinin silindiğinin listelenerek gösterilmesi . . . . .  | 8  |
| 9  | Uyarı mesajı . . . . .   | 9  |
| 10 | Öğrenci güncelleme işlemi ve güncellemenin listelenmesi . . . . .  | 9  |
| 11 | Ekran 2 bu şekilde düzenlenmiştir. Kullanıcı bu sayfadan ders ekleme, silme güncelleme ve listeleme işlemlerini yapabilir. . . . .                       | 11 |
| 12 | Ders ekleme işlemi . . . . .   | 11 |
| 13 | Dersin eklendiğini gösteren bir uyarı mesajı . . . . .   | 12 |
| 14 | Eklenen dersin listelenmesi . . . . .  | 12 |
| 15 | Ders silme işleminin başarıyla sonuçlandığını gösteren uyarı mesajı  | 12 |

|    |   |    |
|----|---|----|
| 16 | Ders güncelleme işleminin başarıyla sonuçlandığını gösteren uyarı mesajı . . . . .  | 13 |
| 17 | Güncellenen dersin listelenmesi . . . . .   | 13 |
| 18 | Ekran 3 bu şekilde düzenlenmiştir. Kullanıcı bu sayfadan bölüm ekleme, silme güncelleme ve listeleme işlemlerini yapabilir. . . . | 15 |
| 19 | Bölümün başarıyla eklendiğine dair uyarı mesajı . . . . .   | 15 |
| 20 | Eklenen bölümün listelenmesi . . . . .  | 16 |
| 21 | Bölümün başarıyla silindiğine dair uyarı mesajı . . . . .   | 16 |
| 22 | Bölümün silindiğinin listelenerek gösterilmesi . . . . .  | 16 |
| 23 | Bölüm Güncelleme İşlemi . . . . .   | 17 |
| 24 | Bölümün başarıyla güncellendiğine dair uyarı mesajı . . . . .   | 17 |
| 25 | Güncellenen bölümün listelenmesi . . . . .  | 17 |
| 26 | Ekran 4 görünümü bu şekildedir. Bu sayfadan kullanıcı yapmak istediği işlemin sayfasına gidebilir. . . . .                        | 19 |
| 27 | Ders atama ekranı . . . . .   | 19 |
| 28 | Ders atama işlemi . . . . .   | 20 |
| 29 | Doğru veri girilmesine dair bir uyarı mesajı . . . . .  | 20 |
| 30 | Dersin atandığına dair uyarı mesajı . . . . .   | 20 |
| 31 | Ders silme işlemi . . . . .   | 20 |
| 32 | Dersin silindiğine dair uyarı mesajı . . . . .  | 21 |
| 33 | Ders değiştirme işlemi . . . . .  | 21 |
| 34 | Uyarı mesajı . . . . .  | 21 |
| 35 | Dersin değiştirildiğine dair uyarı mesajı . . . . .   | 21 |
| 36 | Ekran 5 görünümü bu şekildedir. Kullanıcı istediği işlemi seçebilir.  | 23 |

|    |  |    |
|----|--|----|
| 37 | Öğrenci bazlı listeleme işlemi . . . . .   | 23 |
| 38 | Yıl bazlı listeleme işlemi . . . . .   | 24 |
| 39 | Ekran 6 görünümü bu şekildedir. Kullanıcı seçtiği dersi alan öğrencileri<br>listeleyip vize final notlarını girebilir. . . . . | 26 |
| 40 | Ders seçim işlemi . . . . .  | 26 |
| 41 | Dersi alan öğrencilerin listelenmesi . . . . .   | 26 |
| 42 | Vize final notlarının girilmesi . . . . .  | 26 |
| 43 | Notların girildiğine dair uyarı mesajı . . . . .   | 27 |



---

## ÇİZELGELER LİSTESİ

---

## BÖLÜM: İ

### GİRİŞ

---

## A M A Ç

---

Bu rapor, Entity Framework, SQL Server ve C kullanılarak geliştirilen bir uygulamanın detaylı bir incelemesini sunmaktadır. Uygulamanın ana amacı, öğrenci, ders ve bölüm verilerini etkili bir şekilde yönetmek ve bu veriler üzerinde çeşitli işlemleri gerçekleştirmektir. Bu rapor, uygulamanın geliştirilme sürecini, kullanılan teknolojileri, uygulamanın içeriğini ve sunduğu işlevleri detaylı bir şekilde açıklamaktadır.

Raporun odak noktası, uygulamanın geliştirilme aşamaları ve kullanılan teknolojilerin detaylı bir değerlendirmesidir. İlk olarak, Entity Framework'in Code First yaklaşımı kullanılarak uygulamanın veri modeli tasarlanmış ve ilişkisel veritabanı yapıları oluşturulmuştur. SQL Server, uygulamanın temel veritabanı yönetim sistemi olarak seçilmiş ve Entity Framework üzerinden veritabanı işlemleri gerçekleştirilmiştir.

Uygulama, altı farklı ekranı içermektedir. Her bir ekran, belirli bir işlevselliği temsil etmektedir. Ekran1, öğrenci tablosu üzerinde CRUD işlemlerini gerçekleştirmek için tasarlanmıştır. Ekran2 ve Ekran3, sırasıyla ders ve bölüm tabloları üzerinde benzer işlemleri gerçekleştirmektedir.

Ekran4, öğrencilere yıl ve yarıyıl seçimleri yaparak ders atama, silme ve değiştirme işlemlerini gerçekleştirmektedir. Ekran5, belirli bir öğrenci numarasına göre öğrencinin aldığı dersleri listelemekte ve seçilen bir yıl ve yarıyıl bilgisine göre hangi dersin kaç öğrenci

tarafından seçildiğini listelemektedir. Ekran6 ise, belirli bir dersi alan öğrencilerin listelendiği ve vize ile final notlarının girilerek kaydedildiği bir ekranı içermektedir.

Raporun geri kalan kısmında, her bir ekranın detayları ve işlevselliği ile ilgili ayrıntılı bilgiler verilmektedir. Ayrıca, uygulamanın kullanıcı deneyimi, performans ve güvenlik gibi önemli alanlardaki değerlendirmeleri de içermektedir. Bu rapor, uygulamanın tasarımı, geliştirilme süreci ve sonuçları hakkında kapsamlı bir anlayış sağlamaktadır.

BÖLÜM: İİ

YÖNTEM

---

## SORU 1

---

### Ekran 1: Öğrenci Tablosu Üzerinde CRUD İşlemleri

Windows Forms uygulaması içinde öğrenci verileri üzerinde yapılan CRUD (Create, Read, Update, Delete) işlemleri incelenmiştir.

#### Öğrenci Ekleme İşlemi (Create):

Kullanıcı tarafından öğrenci bilgileri Textbox'lara girilmiş ve girdi doğrulanmıştır. Girilen bilgiler boşluklarla doldurulmuş, ardından sayısallaştırılmıştır. Veritabanında öğrenci ve bölüm varlıkları kontrol edilmiş ve eğer varsa uyarı mesajı gösterilmiştir. Veritabanına yeni öğrenci eklenmiş ve eklenme işlemi başarılı olup olmadığı kontrol edilmiştir. Kullanıcıya ekleme işlemi hakkında bilgi verilmiştir.

#### Öğrenci Silme İşlemi (Delete):

Kullanıcı tarafından silinecek öğrenci ID bilgisi girilmiş ve girdi doğrulanmıştır. Girilen bilgi sayısallaştırılmış ve veritabanında öğrenci varlığı kontrol edilmiştir. Eğer öğrenci bulunamazsa, uyarı mesajı gösterilmiştir. Veritabanından öğrenci silinmiş ve silme işlemi başarılı olup olmadığı kontrol edilmiştir. Kullanıcıya silme işlemi hakkında bilgi verilmiştir.

#### Öğrenci Güncelleme İşlemi (Update):

Kullanıcı tarafından güncellenecek öğrenci bilgileri girilmiş ve girdi doğrulanmıştır. Girilen bilgiler boşluklarla doldurulmuş ve sayısallaştırılmıştır. Veritabanında öğrenci güncellenmiş

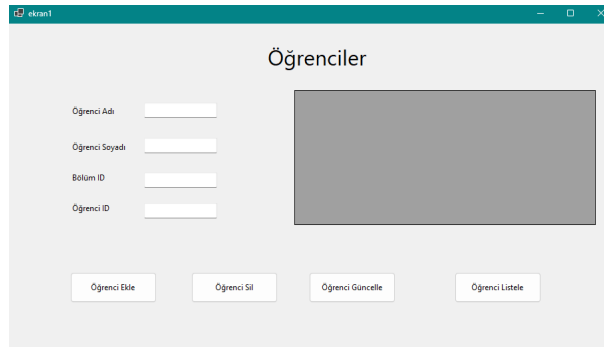
ve güncelleme işlemi başarılı olup olmadığı kontrol edilmiştir. Eğer öğrenci bulunamazsa veya güncelleme başarısız olursa, hata mesajı gösterilmiştir.

#### Öğrenci Listeleme İşlemi (Read):

Kullanıcı tarafından öğrencileri listeleme isteği yapılmış ve veritabanından tüm öğrenci bilgileri çekilmiştir. Veri çekme işlemi sırasında oluşabilecek hatalar yönetilmiş ve kullanıcıya hata mesajı gösterilmiştir.



Şekil 1: Uygulama açıldığında görünen ilk sayfa, uygulamanın anasayfası bu şekildedir. Kullanıcı istediği ekranın butonuna tıklayarak o ekranı görebilir.



Şekil 2: Ekran 1 ise bu şekilde düzenlenmiştir. Kullanıcı bu sayfadan öğrenci ekleme, silme güncelleme ve listeleme işlemlerini yapabilir.

ekran1

Öğrenci Adı zeynep

Öğrenci Soyadı kaymak

Bölüm ID 1

Öğrenci ID 20060358

Öğrenci Ekle Öğrenci Sil

Şekil 3: Öğrenci Ekleme İşlemi

Bilgi

Öğrenci başarıyla eklendi.

Tamam

Şekil 4: Öğrencinin eklendiğine dair bir uyarı mesajı

|   | ogrenciID | ad     | soyad  | bolumID |
|---|-----------|--------|--------|---------|
| ▶ | 1         | Ahmet  | Yılmaz | 1       |
|   | 2         | Ayşe   | Kaya   | 2       |
|   | 3         | Mehmet | Demir  | 3       |
|   | 20060358  | zeynep | kaymak | 1       |
| * |           |        |        |         |

Şekil 5: Eklenen öğrencinin listelenmesi



ekran1

Öğrenci

Öğrenci Adı

Öğrenci Soyadı

Bölüm ID

Öğrenci ID

Öğrenci Ekle Öğrenci Sil

Şekil 6: Öğrenci silme işlemi

Bilgi

Öğrenci başarıyla silindi.

Tamam

Şekil 7: Öğrencinin silindiğine dair bir uyarı mesajı

ekran1

Öğrenciler

Öğrenci Adı

Öğrenci Soyadı

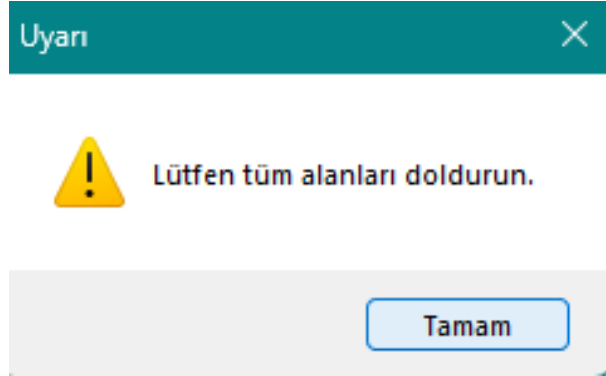
Bölüm ID

Öğrenci ID

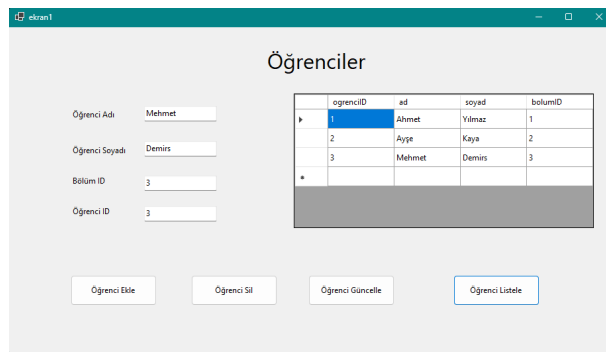
| ogrenciID | ad       | soyad  | bolumID |
|-----------|----------|--------|---------|
| 1         | Abdullah | Yılmaz | 1       |
| 2         | Ayşe     | Kaya   | 2       |
| 3         | Mehmet   | Demir  | 3       |

Öğrenci Ekle Öğrenci Sil Öğrenci Güncelle Öğrenci Listele

Şekil 8: Öğrencinin silindiğinin listelenerek gösterilmesi



Şekil 9: Uyarı mesajı



Şekil 10: Öğrenci güncelleme işlemi ve güncellenenin listelenmesi

---

## SORU 2

---

### Ekran 2: Ders Tablosu Üzerinde CRUD İşlemleri

Windows Forms uygulaması içinde ders verileri üzerinde yapılan CRUD (Create, Read, Update, Delete) işlemleri incelenmiştir.

#### Ders Ekleme İşlemi (Create):

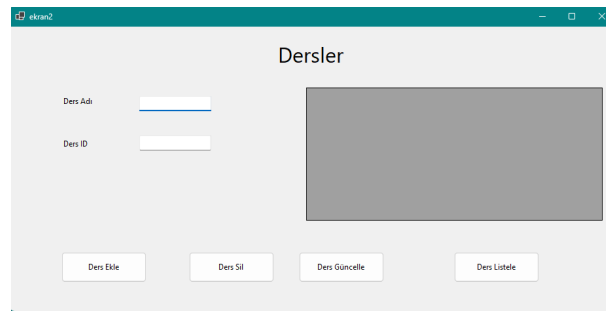
Kullanıcı ders eklemek için "Ekle" butonuna tıklanmış ve ders bilgileri girilmiştir. Girilen ders bilgileri girdi doğrulamasından geçirilmiş ve varlık kontrolü yapılmıştır. Eğer ders zaten mevcutsa veya geçersiz ise, kullanıcıya uyarı mesajı gösterilmiştir. Veritabanına yeni ders eklenmiş ve eklenme işlemi başarılı olup olmadığı kontrol edilmiştir. Kullanıcıya ekleme işlemi hakkında bilgi verilmiştir. Ders Silme İşlemi (Delete):

Kullanıcı ders silmek için "Sil" butonuna tıklanmış ve silinecek dersin ID bilgisi girilmiştir. Girilen ders ID'si doğrulanmış ve varlık kontrolü yapılmıştır. Eğer ders bulunamazsa, kullanıcıya uyarı mesajı gösterilmiştir. Veritabanından ders silinmiş ve silme işlemi başarılı olup olmadığı kontrol edilmiştir. Kullanıcıya silme işlemi hakkında bilgi verilmiştir. Ders Güncelleme İşlemi (Update):

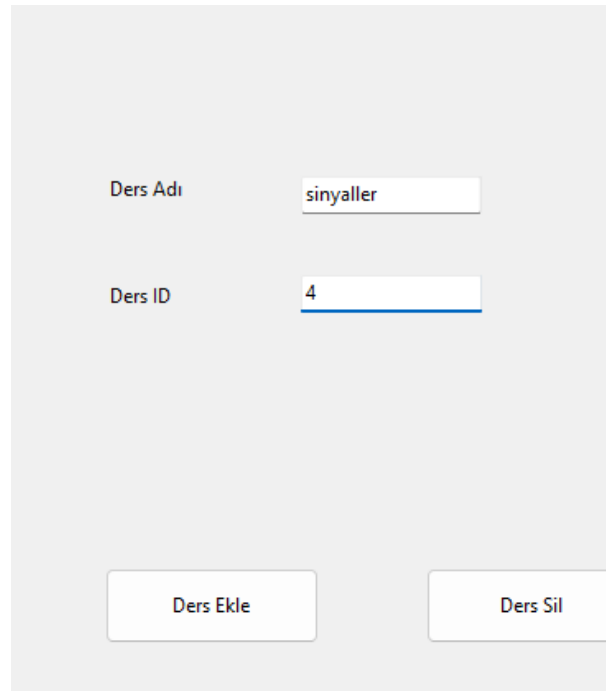
Kullanıcı ders güncellemek için "Güncelle" butonuna tıklanmış ve güncellenecek dersin bilgileri girilmiştir. Girilen ders bilgileri girdi doğrulamasından geçirilmiş ve varlık kontrolü

yapılmıştır. Veritabanında ders güncellenmiş ve güncelleme işlemi başarılı olup olmadığı kontrol edilmiştir. Eğer ders bulunamazsa veya güncelleme başarısız olursa, hata mesajı gösterilmiştir. Ders Listeleme İşlemi (Read):

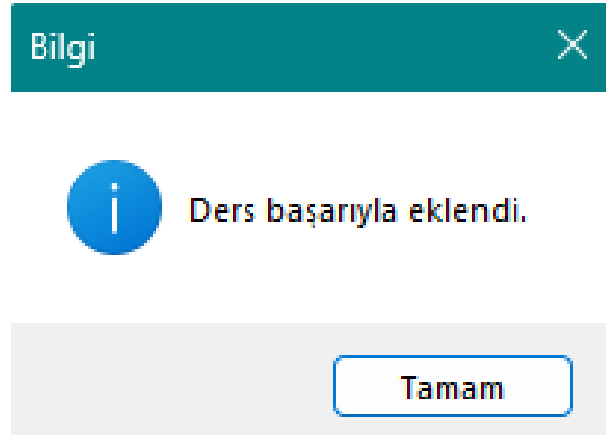
Kullanıcı dersleri listelemek için "Listele" butonuna tıklamış ve tüm ders bilgileri DataGridView kontrolünde görüntülenmiştir. Veri çekme işlemi sırasında oluşabilecek hatalar yönetilmiş ve kullanıcıya hata mesajı gösterilmiştir.



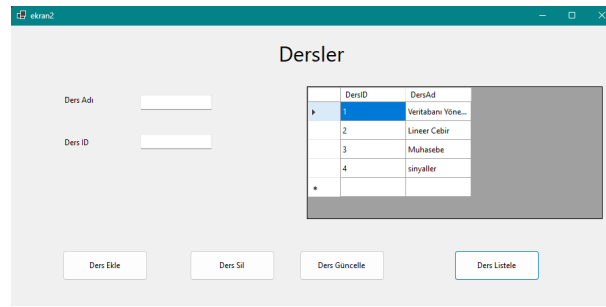
Şekil 11: Ekran 2 bu şekilde düzenlenmiştir. Kullanıcı bu sayfadan ders ekleme, silme güncelleme ve listeleme işlemlerini yapabilir.



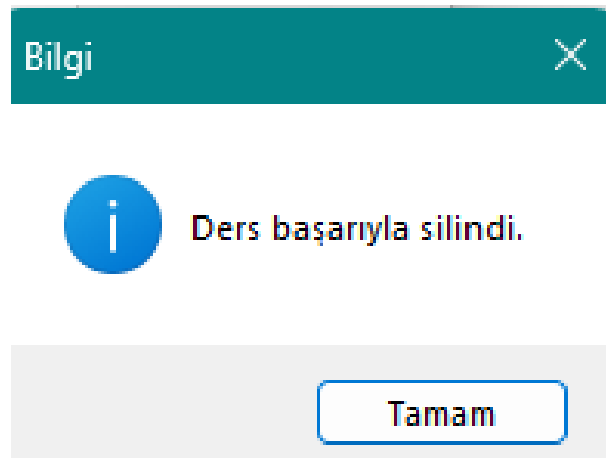
Şekil 12: Ders ekleme işlemi



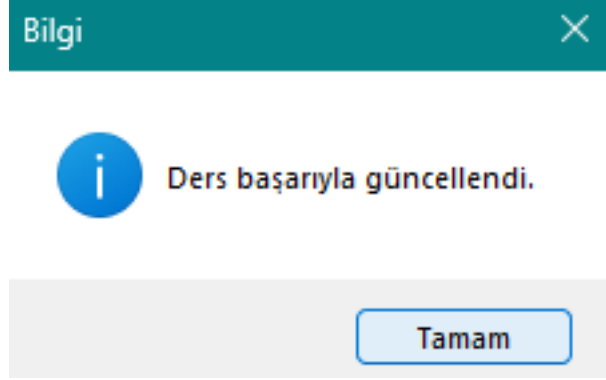
Şekil 13: Dersin eklendiğini gösteren bir uyarı mesajı



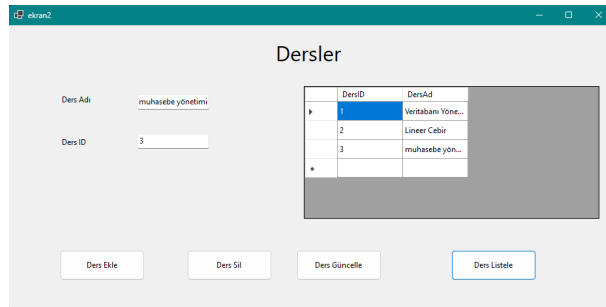
Şekil 14: Eklenen dersin listelenmesi



Şekil 15: Ders silme işleminin başarıyla sonuçlandığını gösteren uyarı mesajı



Şekil 16: Ders güncelleme işleminin başarıyla sonuçlandığını gösteren uyarı mesajı



Şekil 17: Güncellenen dersin listelenmesi

---

### S O R U 3

---

#### Ekran 3: Bölüm Tablosu Üzerinde CRUD İşlemleri

Windows Forms uygulaması içinde bölüm verileri üzerinde yapılan CRUD (Create, Read, Update, Delete) işlemleri incelenmiştir.

##### Bölüm Ekleme İşlemi (Create):

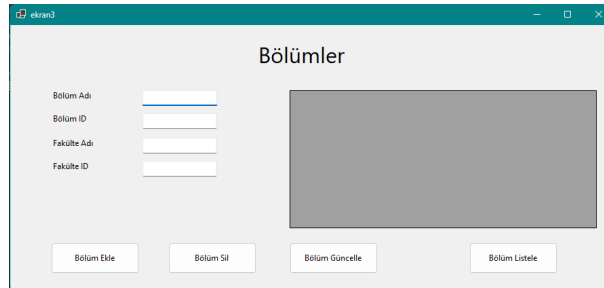
Kullanıcı bölüm eklemek için "Ekle" butonuna tıklamış ve bölüm bilgilerini girmiştir. Girdi doğrulaması yapılmış ve girilen bilgiler kontrol edilmiştir. Eğer bölüm zaten mevcutsa veya geçersiz ise, kullanıcıya uyarı mesajı gösterilmiştir. Veritabanına yeni bölüm eklenmiş ve eklenme işlemi başarılı olup olmadığı kontrol edilmiştir. Kullanıcıya ekleme işlemi hakkında bilgi verilmiştir. Bölüm Silme İşlemi (Delete):

Kullanıcı bölüm silmek için "Sil" butonuna tıklamış ve silinecek bölümün ID bilgisini girmiştir. Girdi doğrulaması yapılmış ve silinecek bölüm var mı diye kontrol edilmiştir. Eğer bölüm bulunamazsa, kullanıcıya uyarı mesajı gösterilmiştir. Veritabanından bölüm silinmiş ve silme işlemi başarılı olup olmadığı kontrol edilmiştir. Kullanıcıya silme işlemi hakkında bilgi verilmiştir. Bölüm Güncelleme İşlemi (Update):

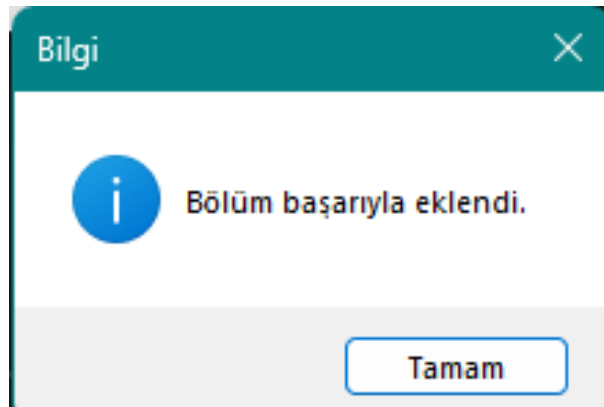
Kullanıcı bölüm güncellemek için "Güncelle" butonuna tıklamış ve güncellenecek bölümün bilgilerini girmiştir. Girdi doğrulaması yapılmış ve güncellenecek bölümün varlığı kontrol

edilmiştir. Veritabanında bölüm güncellenmiş ve güncelleme işlemi başarılı olup olmadığı kontrol edilmiştir. Eğer bölüm bulunamazsa veya güncelleme başarısız olursa, hata mesajı gösterilmiştir. Bölüm Listeleme İşlemi (Read):

Kullanıcı bölümleri listelemek için "Listele" butonuna tıklamış ve tüm bölüm bilgileri DataGridView kontrolünde görüntülenmiştir. Veri çekme işlemi sırasında oluşabilecek hatalar yönetilmiş ve kullanıcıya hata mesajı gösterilmiştir.



Şekil 18: Ekran 3 bu şekilde düzenlenmiştir. Kullanıcı bu sayfadan bölüm ekleme, silme güncelleme ve listeleme işlemlerini yapabilir.



Şekil 19: Bölümün başarıyla eklendiğine dair uyarı mesajı



**Bölümler**

Bölüm Adı:

Bölüm ID:

Fakülte Adı:

Fakülte ID:

| BölümID | BölümAd           | FakülteID |
|---------|-------------------|-----------|
| 1       | Bilgisayar Müh... | 1         |
| 2       | Matematik         | 2         |
| 3       | İşletme           | 3         |
| 4       | elektrik          | 1         |
| *       |                   |           |

Bölüm Ekle Bölüm Sil Bölüm Güncelle Bölüm Listele

Şekil 20: Eklenen bölümün listelenmesi

**Bilgi**

Bölüm başarıyla silindi.

Tamam

Şekil 21: Bölümün başarıyla silindiğine dair uyarı mesajı

**Bölümler**

Bölüm Adı:

Bölüm ID:

Fakülte Adı:

Fakülte ID:

| BölümID | BölümAd           | FakülteID |
|---------|-------------------|-----------|
| 1       | Bilgisayar Müh... | 1         |
| 2       | Matematik         | 2         |
| 3       | İşletme           | 3         |
| *       |                   |           |

Bölüm Ekle Bölüm Sil Bölüm Güncelle Bölüm Listele

Şekil 22: Bölümün silindiğinin listelenerek gösterilmesi

Bölüm Adı


Bölüm ID

Fakülte Adı

Fakülte ID

Şekil 23: Bölüm Güncelleme İşlemi

Bilgi

 Bölüm başarıyla güncellendi.

Şekil 24: Bölümün başarıyla güncellendiğine dair uyarı mesajı

Bölümler

Bölüm Adı

Bölüm ID

Fakülte Adı

Fakülte ID

| BölümID | BölümAd           | FakülteID |
|---------|-------------------|-----------|
| 1       | Bilgisayar Müh... | 1         |
| 2       | Matematik         | 2         |
| 3       | matematik bilimi  | 2         |

Şekil 25: Güncellenen bölümün listelenmesi

---

## SORU 4

---

Ekran 4: Öğrenciye Yıl ve Yarıyıl Seçimleri Yapılarak Ders Atama, Silme, Değiştirme İşlemleri

Bu bölümde, öğrenciye yıl ve yarıyıl seçenekleri sunularak ders atama, silme ve değiştirme işlemleri gerçekleştirilmiştir.

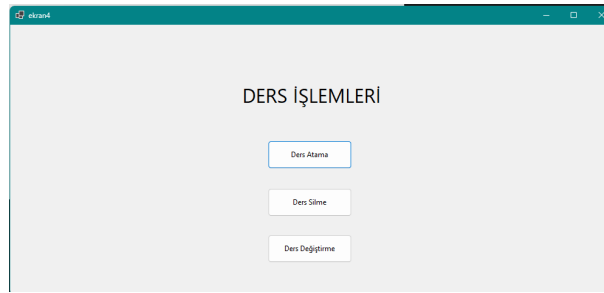
Ders Atama İşlemi (Create):

Kullanıcı öğrenciye ders atamak için yıl ve yarıyıl seçeneklerini belirlemiştir. Seçilen yıl ve yarıyıl doğrulanmış ve varlık kontrolü yapılmıştır. Eğer belirtilen yıl ve yarıyıl geçerli değilse veya ders zaten atanmışsa, kullanıcıya uyarı mesajı gösterilmiştir. Veritabanına öğrenciye ders atanmış ve atanma işlemi başarılı olup olmadığı kontrol edilmiştir. Kullanıcıya atanma işlemi hakkında bilgi verilmiştir. Ders Silme İşlemi (Delete):

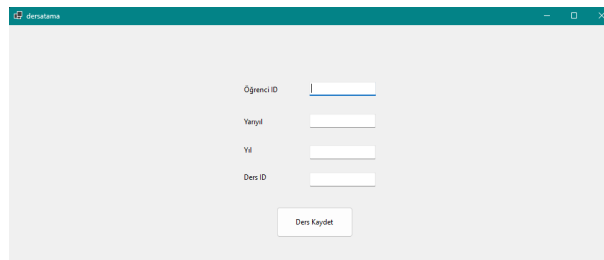
Kullanıcı öğrenciden ders silmek için yıl ve yarıyıl seçeneklerini belirlemiştir. Seçilen yıl ve yarıyıl doğrulanmış ve varlık kontrolü yapılmıştır. Eğer belirtilen yıl ve yarıyıla ait ders bulunamazsa, kullanıcıya uyarı mesajı gösterilmiştir. Veritabanından öğrencinin dersi silinmiş ve silme işlemi başarılı olup olmadığı kontrol edilmiştir. Kullanıcıya silme işlemi hakkında bilgi verilmiştir. Ders Değiştirme İşlemi (Update):

Kullanıcı öğrencinin dersini değiştirmek için yeni yıl ve yarıyıl seçeneklerini belirlemiştir. Seçilen yeni yıl ve yarıyıl doğrulanmış ve varlık kontrolü yapılmıştır. Eğer belirtilen yıl ve yarıyıla ait ders bulunamazsa veya değiştirme başarısız olursa, hata mesajı gösterilmiştir. Veritabanında öğrencinin dersi güncellenmiş ve güncelleme işlemi başarılı olup olmadığı kontrol edilmiştir. Kullanıcıya değiştirme işlemi hakkında bilgi verilmiştir. Ders Listeleme İşlemi (Read):

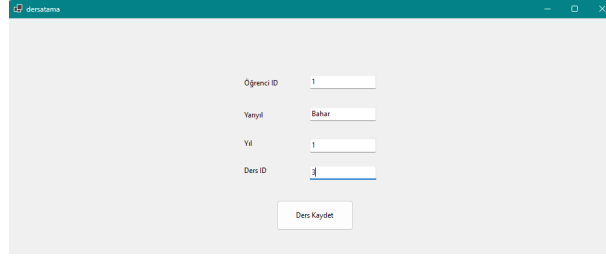
Kullanıcı öğrencinin aldığı dersleri listelemek için yıl ve yarıyıl seçeneklerini belirlemiştir. Seçilen yıl ve yarıyıla ait dersler veritabanından çekilmiş ve DataGridView kontrolünde görüntülenmiştir. Veri çekme işlemi sırasında oluşabilecek hatalar yönetilmiş ve kullanıcıya hata mesajı gösterilmiştir.



Şekil 26: Ekran 4 görünümü bu şekildedir. Bu sayfadan kullanıcı yapmak istediği işlemin sayfasına gidebilir.



Şekil 27: Ders atama ekranı



dersatama

Öğrenci ID: 1

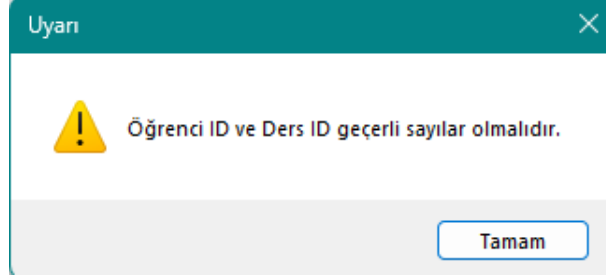
Yarıyıl: Bahar

Yıl: 1

Ders ID: 3

Ders Kaydet

Şekil 28: Ders atama işlemi

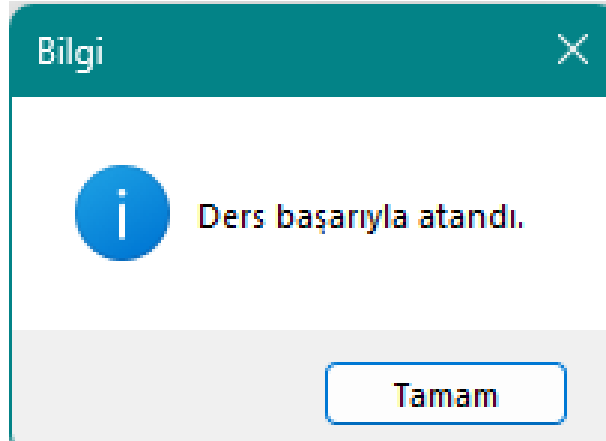


Uyarı

! Öğrenci ID ve Ders ID geçerli sayılar olmalıdır.

Tamam

Şekil 29: Doğru veri girilmesine dair bir uyarı mesajı

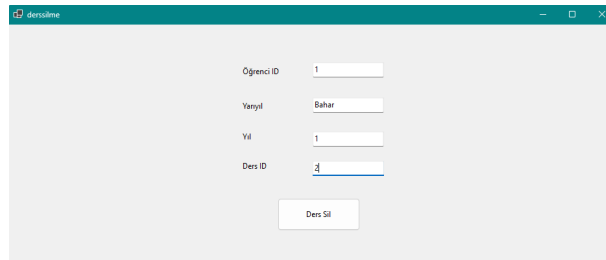


Bilgi

i Ders başarıyla atandı.

Tamam

Şekil 30: Dersin atandığına dair uyarı mesajı



derssilme

Öğrenci ID: 1

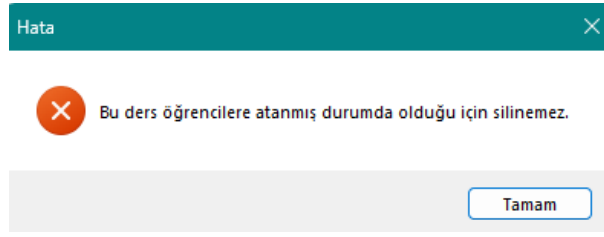
Yarıyıl: Bahar

Yıl: 1

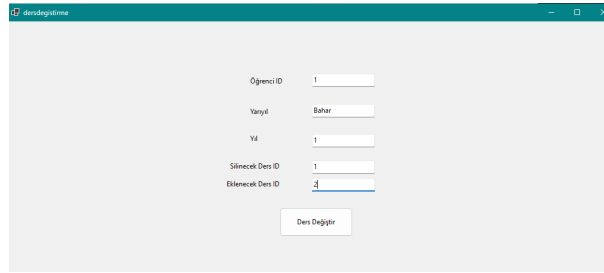
Ders ID: 3

Ders Sil

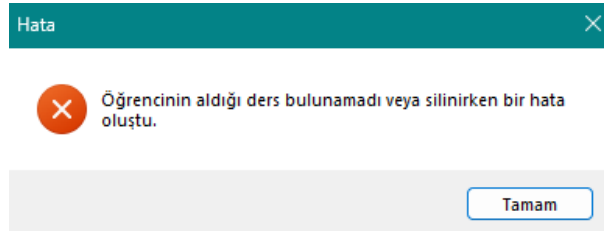
Şekil 31: Ders silme işlemi



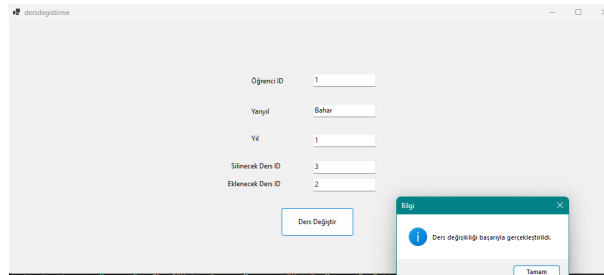
Şekil 32: Dersin silindiğine dair uyarı mesajı



Şekil 33: Ders değiştirme işlemi



Şekil 34: Uyarı mesajı



Şekil 35: Dersin değiştirildiğine dair uyarı mesajı

---

## SORU 5

---

### Ekran 5: Listeleme Ekranı

Bu ekran, iki farklı listeleme işlemi sunmaktadır:

#### Öğrencinin Aldığı Dersleri Listeleme İşlemi:

Kullanıcı, öğrencinin aldığı dersleri listelemek için öğrenci numarasını girer. Girilen öğrenci numarası, veritabanında var mı kontrol edilir. Eğer öğrenci numarasına ait dersler bulunamazsa, kullanıcıya "Ders bulunamadı" şeklinde bir bilgi mesajı gösterilir. Öğrencinin aldığı dersler veritabanından çekilir ve DataGridView kontrolünde görüntülenir. Veri çekme işlemi sırasında herhangi bir hata oluşursa, kullanıcıya hata mesajı gösterilir. Seçilen Yıl ve Yarıyla Göre Ders Dağılımını Listeleme İşlemi:

Kullanıcı, belirli bir yıl ve yarıyıl seçerek o dönemde hangi dersin kaç öğrenci tarafından seçildiğini görmek ister. Seçilen yıl ve yarıyıl, veritabanında geçerli mi kontrol edilir. Eğer belirtilen döneme ait ders dağılımı bulunamazsa, kullanıcıya "Ders dağılımı bulunamadı" şeklinde bir bilgi mesajı gösterilir. Seçilen yıl ve yarıyıla ait ders seçim bilgileri veritabanından çekilir. Hangi dersin kaç öğrenci tarafından seçildiği bilgileri kullanıcıya göstermek için uygun bir şekilde düzenlenir. Veri çekme işlemi sırasında herhangi bir hata oluşursa, kullanıcıya hata mesajı gösterilir.

Öğrenci Bazlı Listeleme

Öğrenci Numarası

Yıl Bazlı Listeleme

Yıl

Yarıyıl

Şekil 36: Ekran 5 görünümü bu şekildedir. Kullanıcı istediği işlemi seçebilir.

Öğrenci Bazlı Listeleme

Öğrenci Numarası

|   | DersAdı      |
|---|--------------|
| ▶ | Lineer Cebir |

Şekil 37: Öğrenci bazlı listeleme işlemi



## Yıl Bazlı Listeleme

Yıl

Yarıyıl

Listele

|   | dersAd       | OgrenciSayisi |
|---|--------------|---------------|
| ► | Lineer Cebir | 1             |
| * |              |               |

Şekil 38: Yıl bazlı listeleme işlemi

---

## SORU 6

---

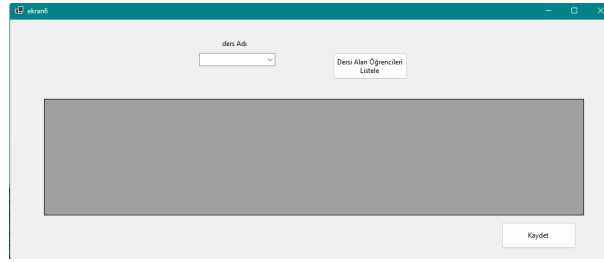
### Ekran 6: Seçilen Bir Dersi Alan Öğrencilerin Listelenmesi ve Not Girişi

Bu ekran, seçilen bir dersi alan öğrencilerin listelenmesi ve bu öğrencilere ait vize ve final notlarının girilerek kaydedilmesini sağlar.

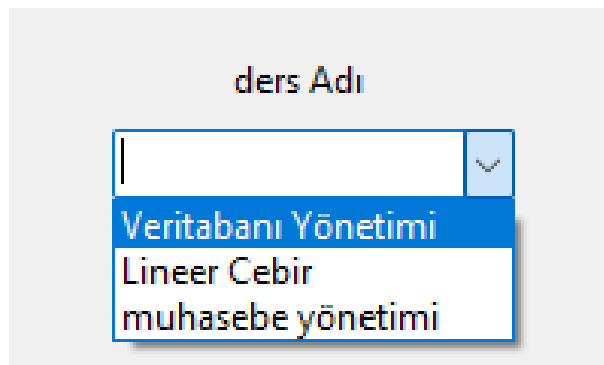
#### Öğrencilerin Listelenmesi İşlemi:

Kullanıcı, listelenmesini istediği dersi seçer. Seçilen dersin varlığı kontrol edilir ve seçilen dersi alan öğrencilerin bilgileri veritabanından çekilir. Eğer seçilen derse ait öğrenci bulunamazsa, kullanıcıya "Öğrenci bulunamadı" şeklinde bir bilgi mesajı gösterilir. Öğrencilerin bilgileri DataGridView kontrolünde görüntülenir. Veri çekme işlemi sırasında herhangi bir hata oluşursa, kullanıcıya hata mesajı gösterilir. Vize ve Final Notlarının Girilmesi ve Kaydedilmesi İşlemi:

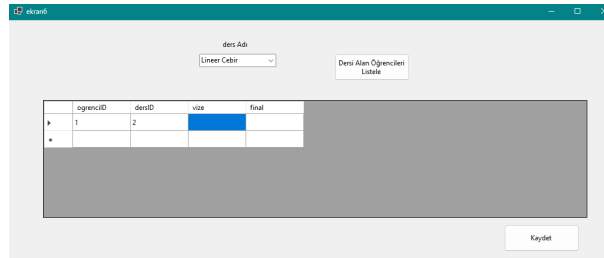
Kullanıcı, öğrencilerin vize ve final notlarını girmek için ilgili hücrelere değerler girer. Girilen notlar doğrulanır ve sınırlar içinde olup olmadığı kontrol edilir. Kaydet butonuna tıklanmasıyla girilen notlar veritabanına kaydedilir. Kaydetme işlemi başarılı olursa, kullanıcıya "Notlar başarıyla kaydedildi" şeklinde bir bilgi mesajı gösterilir. Kaydetme işlemi sırasında herhangi bir hata oluşursa, kullanıcıya hata mesajı gösterilir.



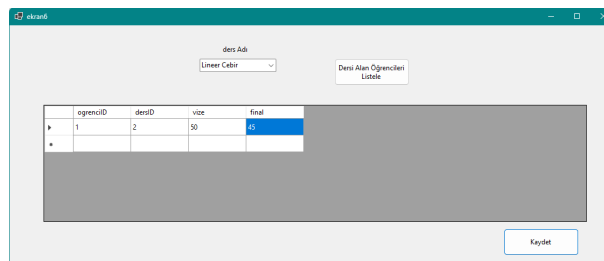
Şekil 39: Ekran 6 görünümü bu şekildedir. Kullanıcı seçtiği dersi alan öğrencileri listeleyip vize final notlarını girebilir.



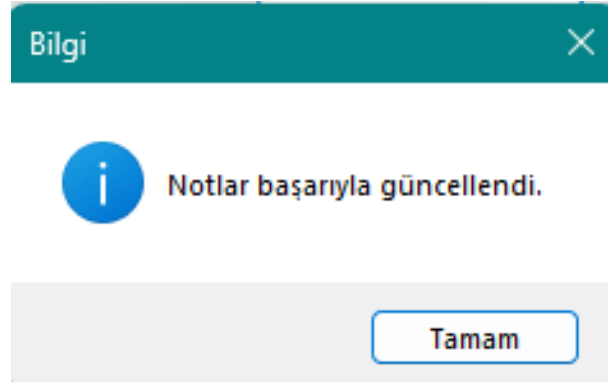
Şekil 40: Ders seçim işlemi



Şekil 41: Dersi alan öğrencilerin listelenmesi



Şekil 42: Vize final notlarının girilmesi



Şekil 43: Notların girildiğine dair uyarı mesajı

---

## SORU 7

---

### Adımlar:

1. Veritabanı Yedeklemesi Alınır: Mevcut veritabanının yedeği alınarak, olası veri kayıplarının önlenmesi hedeflenmektedir. Bu aşamada, SQL Server yönetim araçları kullanılarak veritabanı yedeği alınmaktadır.

2. Migrasyon Araçları Kullanılmalıdır: SQL Server'dan MySQL'e veritabanı aktarımının gerçekleştirilmesi için çeşitli migrasyon araçları bulunmaktadır. MySQL Workbench, SQL Server Migration Assistant (SSMA) gibi araçlar, veritabanı yapılarının dönüştürülmesi ve verilerin aktarılması için kullanılabilir.

3. Veri Dönüşümü ve Uyumluluk Kontrol Edilmelidir: Farklı veritabanı motorları arasında yapısal farklılıklar olabileceği için, veri dönüşümü ve uyumluluk kontrolü önemlidir. Bu aşamada, migrasyon araçları kullanılarak yapısal dönüşümler gerçekleştirilir ve verilerin uyumluluğu kontrol edilir.

4. Uygulama Kodu Güncellenmelidir: Veritabanı motorunun değiştirilmesi durumunda, uygulamanın veritabanıyla etkileşim kurduğu kodun güncellenmesi gerekebilir. Bağlantı dizeleri, sorgu sentaksı ve veri erişim katmanı gibi unsurlar, yeni veritabanı motoruna uygun hale getirilmelidir.

5. Testler Yapılmalıdır: Herhangi bir deęişiklik öncesi kapsamlı testlerin yapılması önemlidir. Bu aşamada, veritabanı işlevsellięi ve uygulama kodu test edilir ve olası hatalar tespit edilir.

6. Canlı Ortama Geçilmelidir: Tüm testler başarıyla tamamlandıktan ve her şeyin düzgün çalıştığından emin olduktan sonra, canlı ortama geçilmelidir. Bu aşamada, kullanıcıları etkilemeden önce bir geçiş planı yapılması ve geri dönüş planlarının hazırlanması önemlidir.

## BÖLÜM: İİİ

### SONUÇ

---

## SONUÇ

---

Uygulamanın geliştirilme süreci sonucunda, Entity Framework, SQL Server ve C kullanılarak başarılı bir şekilde bir veritabanı yönetim sistemine dayalı bir uygulama geliştirilmiştir. Uygulama, öğrenci, ders ve bölüm verilerini etkili bir şekilde yönetmeyi ve çeşitli CRUD işlemlerini gerçekleştirmeyi sağlamaktadır. Bu süreçte, Entity Framework'in Code First yaklaşımı, uygulamanın veri modelini oluşturmak ve ilişkisel veritabanı yapılarını tanımlamak için başarılı bir şekilde kullanılmıştır.

Uygulamanın altı farklı ekranı, kullanıcıların belirli işlevselliği gerçekleştirmelerini sağlamak için tasarlanmıştır. Her bir ekran, öğrenci, ders ve bölüm verileri üzerinde çeşitli işlemleri gerçekleştirmek için kullanıcı dostu bir arayüz sunmaktadır. Bu ekranlar, kullanıcıların verilere erişimini kolaylaştırmak ve veriler üzerindeki değişiklikleri izlemek için tasarlanmıştır.

Uygulamanın test süreci boyunca, işlevselliğin doğruluğu ve kullanıcı deneyiminin tatmin ediciliği gözlemlenmiştir. Uygulama, belirlenen gereksinimleri karşılamak için başarıyla tasarlanmış ve uygulanmıştır. Ancak, kullanıcı geri bildirimleri ve performans test sonuçları doğrultusunda, gelecekteki geliştirmeler için potansiyel alanlar belirlenmiştir. Bu bağlamda, uygulamanın sürdürülebilirliği ve iyileştirilmesi için sürekli olarak dikkat edilmesi gereken konular vurgulanmıştır.



Sonuç olarak, uygulama, Entity Framework, SQL Server ve C gibi modern teknolojilerin başarılı bir şekilde kullanılmasıyla geliştirilmiş ve kullanıcıların veri yönetimi ihtiyaçlarını etkili bir şekilde karşılayacak şekilde tasarlanmıştır.