



**T.C.  
BURSA ULUDAĞ ÜNİVERSİTESİ  
MÜHENDİSLİK FAKÜLTESİ  
Bilgisayar Mühendisliği Bölümü  
2022-2023 Eğitim Öğretim Yılı Bahar Yarıyılı  
BMB2006 - VERİ YAPILARI dersi Projesi**

**Proje raporu**

**Bilgisayar Mühendisliği Fakültesi yüksek lisans ve doktora öğrencilerinin ders kayıtlarını takip etmek amacıyla kullanılan bir yazılım projesi.**

**Hoca**  
Doç. Dr. Murtaza CİCİOĞLU

# Icerik

<b>ÖZET</b>	<b>3</b>
<b>GİRİŞ</b>	<b>4</b>
<b>AMAÇ</b>	<b>4</b>
❖ <b>Sistem Tasarımı</b>	<b>5</b>
❖ <b>Sistem Aktörleri ve Use Case Diyagramları:</b>	<b>5</b>
1. Sistem Aktörleri	5
2. Use Case Diyagramı	6
❖ <b>Kullanıcı Arayüzü Tasarımı</b>	<b>7</b>
1. Loading Ekranı ve Ana Sayfa:	7
2. Giriş Sayfası:	8
3. Öğrenci Arayüzü:	8
4. Öğretmen Arayüzü:	8
❖ <b>Veritabanı Tasarımı</b>	<b>9</b>
❖ <b>Backend Tasarımı</b>	<b>12</b>
	<b>15</b>

## ÖZET

Bilgisayar Mühendisliği Fakültesi yüksek lisans ve doktora öğrencilerinin ders kayıtlarının takibini yapmak amaçlı kullanılacak bir yazılım projesi geliştirildi. Proje, öğrencilerin kayıt olabilecekleri dersleri görüntülemelerine, ders kayıt işlemlerini gerçekleştirmelerine olanak tanıyor. Sistem, hem öğrenciler hem de akademik personel tarafından kullanılabilir.

## GİRİŞ

Bu rapor, Bilgisayar Mühendisliği Fakültesi yüksek lisans ve doktora öğrencilerinin ders kayıtlarının takibini yapmak amacıyla tasarlanan bir yazılım projesinin detaylarını içermektedir.

## AMAÇ

Bu proje raporu, Bilgisayar Mühendisliği Fakültesi'ndeki yüksek lisans ve doktora öğrencilerinin ders kayıtlarının takibini yapmak amacıyla oluşturulan Ders Takip Sistemi projesinin tasarımı, geliştirilmesi ve uygulanması hakkında detaylı bilgi vermektedir. Ayrıca proje kapsamında kullanılan teknolojiler ve yöntemler hakkında da bilgi verilmektedir.

## ❖ Sistem Tasarımı

Bu proje, Bilgisayar Mühendisliği Fakültesi'ndeki lisans, yüksek lisans ve doktora öğrencilerinin ders kayıtlarını takip etmek için tasarlanmıştır. Projenin temel amacı, öğrenci ve öğretmen gibi farklı kullanıcı türleri için uygun bir arayüz sağlamak, veritabanı tasarımı ile verilerin düzenli ve güvenli bir şekilde saklanması sağlamak ve backend tasarımı ile arayüz ve veritabanı arasındaki etkileşimi yönetmektir.

**Kullanıcı Arayüzü Tasarımı**, projenin en önemli bölümlerinden biridir. Bu bölümde, öğrenci, öğretmen ve personel için farklı arayüzler tasarlanacak ve kullanıcıların kolayca istedikleri bilgilere erişebilmesi sağlanacaktır. Arayüzlerin kullanımı kolay ve anlaşılır olacak, herhangi bir teknik bilgiye sahip olmayan kullanıcılar bile sistemi rahatlıkla kullanabileceklerdir.

**Veritabanı Tasarımı**, projenin diğer önemli bölümüdür. Bu bölümde, öğrenci, öğretmen ve personel gibi farklı kullanıcı türlerine ait verilerin düzenli ve güvenli bir şekilde saklanması sağlanacaktır. Veritabanı tasarımı, kullanıcıların verilerini kolayca arayabilmesi ve güncelleyebilmesi için optimize edilecektir.

**Backend Tasarımı**, projenin son önemli bölümüdür. Bu bölümde, Kullanıcı Arayüzü Tasarımı ve Veritabanı Tasarımı arasındaki etkileşim sağlanacaktır. Kullanıcıların arayüzler üzerinden yaptıkları istekler, backend tasarımı tarafından alınacak, gerekli işlemler gerçekleştirilecek ve sonuçlar kullanıcılara geri döndürülecektir. Bu bölüm, projenin tamamının düzgün ve verimli bir şekilde çalışmasını sağlayacak önemli bir unsurdur.

Sonuç olarak, bu proje, Bilgisayar Mühendisliği Fakültesi'ndeki öğrenci, öğretmen ve personel için ders takip sistemi sağlamak için üç ana kısımdan oluşan bir yazılım projesidir. Bu kısımların her biri, projenin başarılı bir şekilde hayata geçirilmesi için önemli bir role sahiptir.

## ❖ Sistem Aktörleri ve Use Case Diyagramları:

Sistemin işlevselliği, kullanım senaryoları ve aktörler arasındaki etkileşimleri gösteren Use Case diyagramı aşağıdaki gibidir:

### 1. Sistem Aktörleri

Sistemde 3 adet aktör bulunmaktadır:

**Kullanıcılar (Öğretmen/öğrenci):** Sistemi kullanacak olan temel aktördür. Kullanıcılar sisteme giriş yaparak ders kayıtlarını görüntüleyebilir ve akademik ilerlemelerini takip edebilirler.

**Sistem Sorumlusu:** Sistem sorumlusu, sistemdeki tüm işlemleri yönetir ve izler. Kullanıcıların erişim haklarını ve yetkilerini belirleyebilir, sistemin güvenliğini sağlayabilir ve veritabanı yönetimiyle ilgili işlemleri gerçekleştirebilir.

**Veri Tabanı:** Sistemdeki verilerin saklandığı ve yönetildiği temel aktördür. Veri tabanı, kullanıcıların kişisel ve akademik bilgilerini depolayacak, ders programlarını ve derslere kayıtlı öğrenci listelerini saklayacak ve bu verilere erişimi kontrol edecektir.

## 2. Use Case Diyagramı

**Kullanıcı Girişi:** Kullanıcıların sisteme giriş yapması ve hesaplarına erişim sağlaması için kullanılır. Kullanıcı, kullanıcı adı ve şifresini girerek sisteme giriş yapabilir.

**Ders Kaydı:** Kullanıcılar, mevcut dersler arasından seçim yaparak derslere kaydolabilirler. Bu use case, öğrencilerin ders seçme işlemlerini ve öğretmenlerin dersleri oluşturma ve güncelleme işlemlerini kapsar.

**Ders Programını Görüntüleme:** Kullanıcılar, kayıtlı oldukları derslerin programını görüntüleyebilirler. Bu use case, kullanıcıların ders saatlerini, sınıflarını ve öğretim görevlilerini görmelerine olanak tanır.

**Not Girişi:** Öğretmenler, öğrencilerin performansını değerlendirmek için not girişi yapabilirler. Bu use case, öğretmenlerin öğrencilerin sınav sonuçlarını, ödev notlarını veya proje puanlarını sisteme kaydetmelerini içerir.

**Akademik İlerlemeyi Görüntüleme:** Kullanıcılar, akademik ilerlemelerini kontrol etmek için notlarını ve ders geçme durumlarını görüntüleyebilirler. Bu use case, kullanıcılara derslerdeki performanslarını takip etme imkanı sunar.

**Yetkilendirme ve Güvenlik Yönetimi:** Sistem sorumlusu, kullanıcıların erişim haklarını ve yetkilerini yönetir. Bu use case, sisteme yeni kullanıcılar eklemeyi, kullanıcı hesaplarını yönetmeyi ve güvenlik önlemlerini uygulamayı içerir.

**Veri Tabanı Yönetimi:** Sistem sorumlusu, veritabanının yönetimini gerçekleştirir. Bu use case, veri tabanının güncellenmesini, yedeklenmesini ve veri bütünlüğünün korunmasını içerir.

## ❖ Kullanıcı Arayüzü Tasarımı

Kullanıcı Arayüzü Tasarımı, Ders Takip Sistemi'nin önemli bir parçasıdır. Bu kısım, öğrenci, ve öğretmen kullanıcıları için arayüzleri içerir. Arayüz, C# ve .NET programlama dilleri kullanılarak geliştirilmiştir.

### 1. Loading Ekranı ve Ana Sayfa:

Loading Ekranı ve Ana Sayfa, kullanıcıların sisteme hızlı ve kolay bir şekilde erişim sağlamalarını ve gerekli bilgilere erişerek işlemlerini gerçekleştirmelerini sağlamak amacıyla tasarlanmış önemli bileşenlerdir.

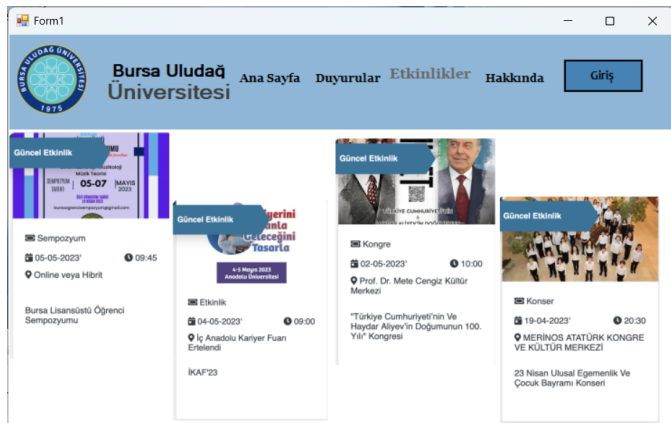
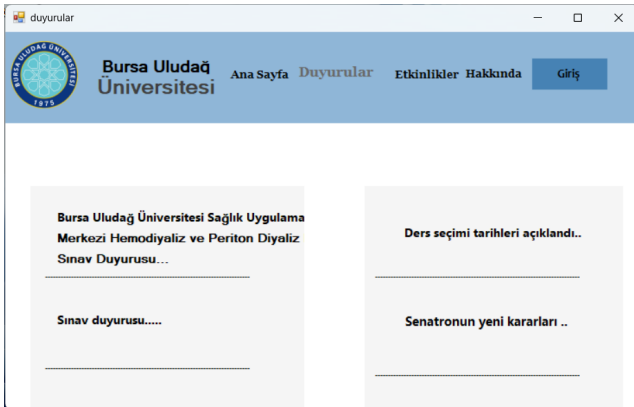
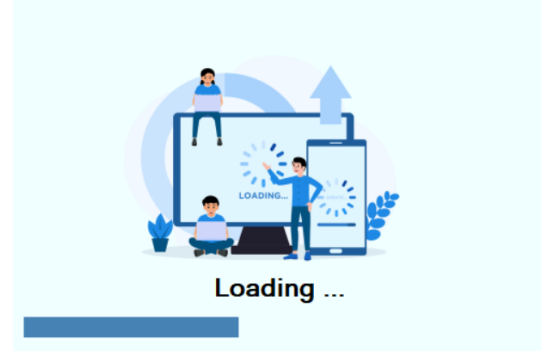
#### Loading Ekranı:

Loading ekranı, kullanıcının beklemesini engelleyerek sistemin yüklenme sürecini gösteren bir ekran tasarımıdır.

#### Ana Sayfa:

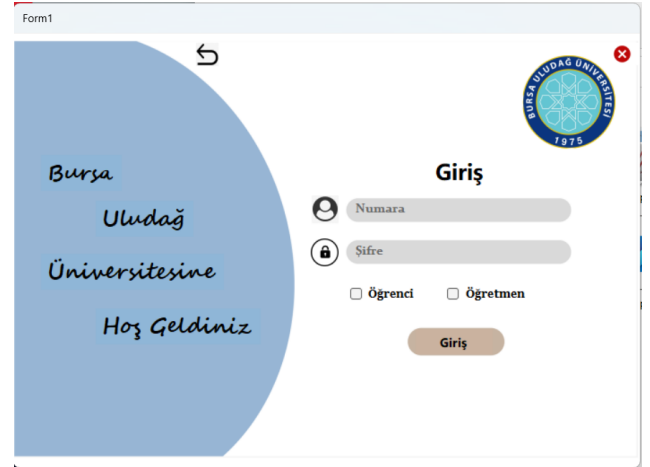
Ana Sayfa, Ders Takip Sistemi'nin başlangıç noktasıdır ve kullanıcıların karşılaştığı ilk sayfadır. Ana sayfa üzerinde kullanıcıları yönlendirmek için bir navbar (navigasyon çubuğu) bulunur. Bu navbar, kullanıcılara farklı sekmeler arasında geçiş yapma imkanı sunar. Örneğin, "Ana Sayfa" seçeneği kullanıcıyı ana sayfaya yönlendirirken, "Duyurular" seçeneği duyuruların bulunduğu sayfaya yönlendirir. Benzer şekilde, "Etkinlikler" seçeneği etkinliklerin bulunduğu sayfaya yönlendirir ve "Hakkında" seçeneği ise sistem hakkında bilgilendirme içeriği sunan bir sayfaya yönlendirir.

- Ana sayfa üzerinde bir navigasyon çubuğu (navbar) bulunmalıdır.
- Navbar, kullanıcılara ana sayfa, duyurular, etkinlikler, hakkında gibi sekmeleri seçme imkanı sunar.
- Giriş yapma işlemi için "Giriş" butonu vardır.



## 2. Giriş Sayfası:

Giriş yapma işlemi için "Giriş" butonu, kullanıcının sisteme erişim sağlaması için gereken bir adımdır. Kullanıcı, öğrenci veya öğretmen olarak sisteme giriş yapabilir. Giriş yapma butonuna tıklandığında kullanıcıya giriş yapma seçenekleri sunulur, kullanıcı numarası ve şifresini girerek sisteme giriş yapabilir. Bu şekilde kullanıcı, kendi hesabına erişerek öğrenci veya öğretmen arayüzüne yönlendirilir ve ilgili işlemleri gerçekleştirebilir.



Form1

Bursa  
Uludağ  
Üniversitesine  
Hoş Geldiniz

**Giriş**

Numara

Şifre

☐ Öğrenci ☐ Öğretmen

**Giriş**

## 3. Öğrenci Arayüzü:

Öğrenci arayüzü, öğrencilerin kişisel bilgilerini ve kayıtlı derslerini görüntülemelerini sağlar. Ayrıca, öğrenciler akademik ilerlemelerini takip edebilirler. Bu arayüz, öğrencilerin derslerinin detaylarını ve ders programlarını gösterir.



**KADİR CAN ÇELİK**

Ad : KADİR Soyad : ÇELİK  
Öğrenci id : 23 Cinsiyet : ERKEK  
Bölüm : BİLGİSAYAR MÜHENDİSLİĞİ Dönem : 1

**Profile**

**Dersler**

**Çıkış**



**KADİR CAN ÇELİK**

**Ders Alma**

Seç	Ders ID	Ders Adı	Ders Kredisi	Ders günü	Ders saati
<input checked="" type="checkbox"/>	12	DOKTORA UZMANLIĞI	5	SALI	12
<input type="checkbox"/>	13	TEZ DANIŞMANLIĞI	1	SALI	13
<input type="checkbox"/>	14	İLERİ ALGORİTMA T...	6	SALI	14
<input type="checkbox"/>	15	KRİPTOGRAFI	6	SALI	15
<input type="checkbox"/>	16	YAPAY ZEKANIN MA...	6	SALI	16
<input type="checkbox"/>	17	İLERİ ÖRÜNTÜ TANIL...	6	ÇARŞAMBA	9

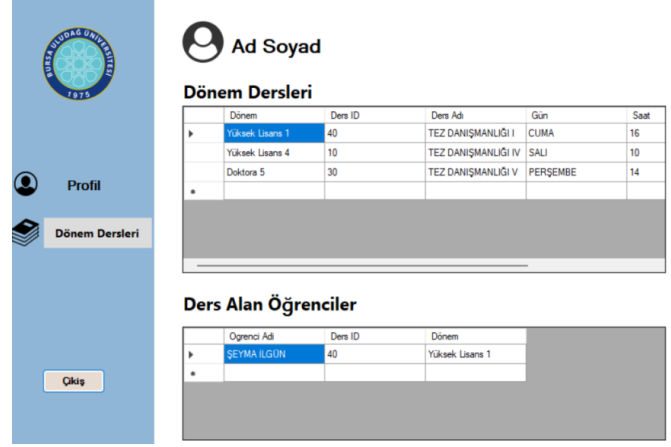
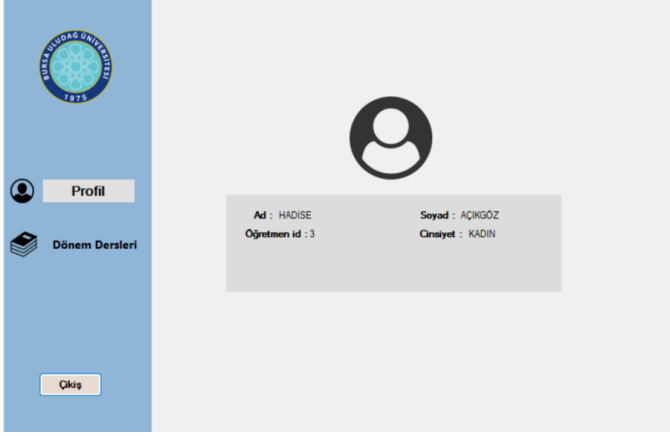
**Dönem Dersleri**

Ders ID	Ders Adı	Ders Kredisi	Ders günü	Ders saati
12	DOKTORA UZMANLIĞI	5	SALI	12

**Çıkış**

## 4. Öğretmen Arayüzü:

Ders Takip Sistemi'ne giriş yapan öğretmenlerin kişisel bilgilerini görüntüleyebildiği ve yönetebildiği bir bölümdür. Bu arayüzde öğretmenlerin adı, numarası, bölümü gibi kişisel bilgileri yer alır ve güncellenebilir. Ayrıca, öğretmenin verdiği derslerin bir listesi sunulur, bu listede ders adları, ders kodları, ders saatleri gibi ilgili bilgiler bulunur. Öğretmen arayüzünde ayrıca ders aldığı öğrencilerin listesi de görüntülenir, böylece öğretmenler hangi öğrencilerin dersine kayıtlı olduğunu takip edebilirler. Bu sayede öğretmenler, derslerini yönetebilir, öğrencilerin akademik performanslarını izleyebilir ve gerekli takip ve değerlendirmeleri yapabilirler.



## ❖ Veritabanı Tasarımı

### Veritabanı

Veritabanı, PostgreSQL veritabanı yönetim sistemi ve pgAdmin4 grafik kullanıcı arayüzü kullanılarak tasarlandı. Veritabanı şeması aşağıdaki gibidir:

- Öğrenci giriş tablosu: öğrenci id, kullanıcı adı ve şifre alanlarını içerir.
- Öğrenci bilgileri tablosu : Ad soyad , cinsiyet, bölüm, öğrenci id, dönem alanlarını içerir.
- Dönem dersleri tablosu: Dönem, ders id , ders adı, gün id, saat, öğretmen bilgilerini içerir.
- Öğretim görevlisi giriş tablosu: Kullanıcı adı, öğretim üyesi id, şifre bilgilerini içerir.
- Öğretim görevlisi bilgileri tablosu: ad soyad, cinsiyet, öğretmen bilgilerini içerir.
- Dersi alan öğrenciler tablosu: Kullanıcı adı, ders id, dönem bilgilerini içerir.
- Ders tablosu: Öğretim üyesi id, dersin adı, dersin kredisi, ders id, saat id, dönem id, gün id, ders türü bilgilerini içerir.

Aşağıda örnek bir tablo gösterilmiştir:



	ogretmen_id [PK] integer	ders_adi [PK] character varying[] (50)	ders_kredi [PK] integer	ders_id [PK] integer	saat_id [PK] integer	donem_id [PK] integer	gun_id [PK] character varying[] (20)	ders_turu [PK] character
1	1	{'ARAŞTIRMA TEKNİKLERİ VE YAYIN ETİĞİ'}	2	1	9	12	{PAZARTESİ}	{zorunlu}
2	1	{'PARALEL ALGORİTMALAR'}	6	41	9	11	{CUMA}	{seçmeli}
3	1	{'TEKNOLOJİ TRANSFERİ, AR-GE VE İNOVASYON...'}	2	21	13	22	{ÇARŞAMBA}	{zorunlu}
4	1	{'TEZ DANIŞMANLIĞI VI'}	25	31	15	26	{PERŞEMBE}	{zorunlu}
5	1	{'YÜKSEK LİSANS UZMANLIK ALAN dersi IV'}	5	11	11	14	{SALI}	{zorunlu}
6	2	{'BİLGISAYARLA GÖRME'}	6	42	10	11	{CUMA}	{seçmeli}
7	2	{'ÇOK DEĞİŞKENLİ İSTATİSTİKSEL VERİ ANALİZİ '}	6	22	14	22	{ÇARŞAMBA}	{seçmeli}
8	2	{'DOKTORA UZMANLIK ALAN dersi I '}	5	12	12	21	{SALI}	{zorunlu}
9	2	{'DOKTORA UZMANLIK ALAN dersi V'}	5	32	16	25	{PERŞEMBE}	{zorunlu}
10	2	{'TEZ DANIŞMANLIĞI II'}	1	2	10	12	{PAZARTESİ}	{zorunlu}
11	3	{'DOĞAL DİL İŞLEME'}	6	43	11	11	{CUMA}	{seçmeli}
12	3	{'DOKTORA UZMANLIK ALAN dersi VI'}	5	33	9	26	{CUMA}	{zorunlu}
13	3	{'İLERİ SES İŞLEME '}	6	23	15	22	{ÇARŞAMBA}	{seçmeli}
14	3	{'TEZ DANIŞMANLIĞI I'}	1	13	13	21	{SALI}	{zorunlu}
15	3	{'YÜKSEK LİSANS UZMANLIK ALAN dersi II'}	5	3	11	12	{PAZARTESİ}	{zorunlu}
16	4	{'DOKTORA UZMANLIK ALAN dersi VII'}	5	34	10	27	{CUMA}	{zorunlu}
17	4	{'İLERİ ALGORİTMA TASARIMI VE ANALİZİ '}	6	14	14	21	{SALI}	{zorunlu}

Bu tablolar oluşturulduktan sonra gerekli tablolar birbirleriyle ilişkilendirildi. Bu işlem, farklı tablolardaki verilerin bir araya getirilerek daha anlamlı bir şekilde kullanılabilmesini sağlar. Girilen verilerin kontrolü ve güvenilirliği sağlandı. Tabloların birbirleriyle ilişkilendirilmesi veri tabanı tasarımında oldukça önemlidir.

Foreign Key olarak belirlediğimiz tabloya girilen verilerin Foreign Key tablosundaki verilerle kontrol edilmesini sağlar.

Primary Key (Birincil Anahtar), bir tablodaki bir veya daha fazla sütunun benzersiz tanımlayıcı değerleri içerdiği bir veya daha fazla sütundan oluşan bir dizi veridir. Her satırın Primary Key değeri, o tablonun tekil bir tanımını oluşturur ve herhangi bir tekrarlayan veri olmamasını sağlar.

Foreign Key (Yabancı Anahtar), bir tablodaki sütunların diğer bir tablonun birincil anahtarına referans veren sütunlardır. Bu ilişki sayesinde, iki tablo arasında bir bağlantı kurulur.

## Sonuç

Fen Bilimleri Enstitüsü Ders Kayıt Takip Sistemi, öğrencilerin ders kayıtlarını takip etmelerine ve ders seçimlerini gerçekleştirmelerine olanak tanıyan, akademik personelin ders programlarını yönetmesine ve öğrenci notlarını takip olarak sağlayan bir yazılım projesidir. Projede kullanılan veritabanı, öğrenci bilgilerinin güvenli bir şekilde depolanmasını sağlamak için kullanılmaktadır.

Veri tabanının tasarımı, öğrencilerin kayıtlarının tutulmasını, derslere erişebilmesini, ders seçimi yapabilir ve notlarının takip edilmesini kolaylaştırmaktadır. Öğrencilerin ders seçimlerinin kaydedilmesi, ders programlarının oluşturulması için kullanıcı dostu bir arayüz tasarlanmıştır.

Bu proje, Fen Bilimleri Enstitüsü'nün Bilgisayar Mühendisliği öğrencilerinin ve akademik personelinin ders kayıtlarının takibi için önemli bir araç olacaktır. Projenin geliştirilmesinde emeği geçen tüm ekip üyelerine teşekkür ederiz.

## ❖ Backend Tasarımı

Kodun çalışma algoritması, Ders Takip Sistemi'nin en önemli parçasıdır. Bu kısım, veri tabanından alınan verileri Backend'de işleyip arayüze sunmayı içerir. Backend SQL, C# ve .NET programlama dilleri kullanılarak geliştirilmiştir.

Backend kısmında, dört farklı veri yapısı kullanılmıştır. Bunlar;

1 - Öğrenci ve öğretim görevlisi verilerinin tutulduğu AVL ağacıdır. Bu ağaç, veri tabanındaki “ogrenciBilgi” ve “ogrenciGiris” tablolarındaki verileri tutan “ogrenciAVL” ile “ogretimGorevlisiBilgi” ve “ogretimGorevlisiGiris” tablolarındaki verileri tutan “ogretimGorevlisiAVL” den oluşur.

2 - Seçilen derslerin tutulduğu “İkiliBagliListe” dir. “İkiliBagliListe”, öğrencinin seçtiği dersleri tutan “donemDersleri” tablosunu ve kayıt sırasında veri tabanına kaydedilene kadar elde tutulan dersleri içerir.

3 - Tüm derslerin tutulduğu “HashTablosu” dur. “HashTablosu”, derslerin içinde bulunan “donem\_id” anahtarını kullanarak verileri ilgili listede tutar.

4 - “HashTablosu” nun içinde “donem\_id” sine göre dersleri tutan “dersListe” dir. Bu yapı, aynı “donem\_id” sine sahip dersleri “HashTablosu” nda sıralı bir şekilde tutar.

### ● Veri Tabanının Backend’e Bağlanması

Backend’in, veri tabanına bağlanması üç aşamadan oluşur:

1- Bağlantının “NpgsqlConnection” ile kurulması

2- Verilerin “NpgsqlCommand” ile işlenmesi

3- Bağlantının “.Close()” metodu ile kapatılması

### ● Uygulama Arayüzünün Backend ile Entegrasyonu

1- İlgili arayüz elemanlarının “Text” lerine veri yapılarındaki verilerin yazılması

2- Öğrenciler için ders seçimi sırasında dönem derslerinin işaretli olup olmadığına göre verinin işlenmesi



Profil



Dersler

Çıkış



KADİR CAN ÇELİK

## Ders Alma

	Seç	Ders ID	Ders Adı	Ders Kredisi	Ders günü	Der
▶	<input checked="" type="checkbox"/>	12	DOKTORA UZMANLI...	5	SALI	12
	<input type="checkbox"/>	13	TEZ DANIŞMANLIĞI I	1	SALI	13
	<input type="checkbox"/>	14	İLERİ ALGORİTMA T...	6	SALI	14
	<input type="checkbox"/>	15	KRİPTOĞRAFI	6	SALI	15
	<input type="checkbox"/>	16	YAPAY ZEKANIN MA...	6	SALI	16
	<input type="checkbox"/>	17	İLERİ ÖRÜNTÜ TANI...	6	ÇARŞAMBA	9
*	<input type="checkbox"/>					

## Dönem Dersleri

Kaydet

	Ders ID	Ders Adı	Ders Kredisi	Ders günü	Ders saati
▶	12	DOKTORA UZMANLI...	5	SALI	12
*					



Profil



Dönem Dersleri

Çıkış



HADİSE AÇIKGÖZ

## Dönem Dersleri

	Dönem	Ders ID	Ders Adı	Gün	Saat
▶	Yüksek Lisans 1	43	DOĞAL DİL İŞLEME	CUMA	11
	Yüksek Lisans 2	3	YÜKSEK LİSANS UZ...	PAZARTESİ	11
	Doktora 1	13	TEZ DANIŞMANLIĞI I	SALI	13
	Doktora 2	23	İLERİ SES İŞLEME	ÇARŞAMBA	15
	Doktora 6	33	DOKTORA UZMANLI...	CUMA	9
*					

## Ders Alan Öğrenciler

	Öğrenci Adı	Ders ID	Dönem
*			

## Öğrencilerin Aldığı Görevlerin

Fen Bilimleri Enstitüsü yüksek lisans ve doktora öğrencilerinin ders kayıtlarının takibini yapmak amacıyla bir yazılım projesi tasarlanıp geliştirilmiştir. Bu proje, **Kullanıcı Arayüzü Tasarımı**, **Veritabanı Tasarımı** ve **Backend Tasarımı** olmak üzere üç ana kısımdan oluşmaktadır.

### **Kullanıcı Arayüzü Tasarımı**

Lokmane Boulbadaoui (032090090)

Chaimaa Nairi (031890119)

### **Veritabanı Tasarımı**

Şeyma İlgün (032190068)

Cansu Kocaoğlu (032090049)

Ayçanur Güç (032190055)

Beyza Tuncel (032190015)

Sevim Ulusoy (032190013)

Nisa Büyüктаş (032090058)

### **Backend Tasarımı**

Muhammet Güven (032190025)

Mehmet Kaan Genç (032190023)

Umut Can Oral (032190063)

Münir Aydemir (032190045)

## Önsöz

Bu çalışmanın hayata geçirilmesi sürecinde bilgi ve tecrübelerinden faydalandığımız ve bizi doğru yönde yönlendiren sayın *Doç. Dr. Murtaza CİCİOĞLU*'ya, sonsuz teşekkürlerimizi sunarız.

Bursa, 2023