



**Çanakkale Onsekiz Mart Üniversitesi**  
**Mühendislik Fakültesi**  
**Bilgisayar Mühendisliği Bölümü**

**BLM-4002 Proje Dersi**

**Baha YOLAL – 200401031**

**Erlindi ISAJ – 200401114**

**Leonit SHABANI – 200401110**

**Öğr. Gör. İsmail KAHRAMAN**

# Öğrenci Bilgi Sistemi

## GİRİŞ BÖLÜMÜ

Rapor, bir üniversite öğrenci bilgi sisteminin tasarımına yönelik gereksinimleri ve kısıtları detaylı bir şekilde açıklamaktadır. Sistem, öğrencilerin ders seçiminden not girişine kadar çeşitli işlevleri içermekte ve farklı kullanıcı rollerine sahip olmaktadır. Her bir kullanıcı rolü, belirli işlevleri gerçekleştirebilmekte ve sistemin temel yapı taşlarını oluşturmaktadır. Ayrıca, kullanıcıların sisteme erişim ve işlevlerine ilişkin belirli kısıtlamalar ve gereksinimler de belirtilmiştir. Raporun en alt sayfasında projenin GitHub linki mevcuttur. Bu rapor, sistemin genel yapısını ve tasarım sürecindeki ana hedefleri açıklamak amacıyla hazırlanmıştır.

- **Sistem Üyelikleri ve Giriş:**

- ◊ Sistemde üyelik ekranı yoktur. Her kullanıcının sisteme kayıtlı birer üyeliği vardır ve sadece sistemde kayıtlı kullanıcılar oturum açabilmektedir.
- ◊ Kullanıcılar, sisteme kayıtlı TC Kimlik numaralarıyla ve parolalarıyla sisteme giriş yapabilirler.
- ◊ Her kullanıcıya sadece bir üyelik tipi atanmaktadır: Öğrenci, öğretim elemanı, danışman veya idareci.
- ◊ Her üyelik tipi kendisine atanmış üyelik sistemine giriş yapabilir. Örneğin; danışman 'Öğreti Elemanı Sistemine' öğretim elemanı ise 'Danışman Sistemin' giriş yapamaz.
- ◊ Üyelik tipleri sadece kendisine ait işlevleri gerçekleştirebilmektedir. Örneğin danışman not girişi yapamamaktadır.
- ◊ Bütün üyelik tipleri için hesap ayarları; cep telefonu bilgisi, kurumsal ve kişisel e-posta, ikamet adres ve şifre ayarlarını içerir. Bütün üyelik tipleri, hesap ayarları menüsünden hesap ayarlarını görüntüleyerek değiştirebilir.

- **Üyelik Tipleri İşlevleri, Kısıtlamaları ve Gereksinimleri:**

- **1. Öğrenci Bilgi Sistemi**

- ◊ Öğrenci Bilgi Sistemine giriş yapan kullanıcıların temel işlevleri yönetmeliğe uygun ders seçimi yaparlar ve aldıkları ders bilgilerini, notlarını ve detaylarını görüntüleyebilirler.
- ◊ Öğrenci Bilgi Sistemi, 'Derslerim', 'Ders Seçimi/Kayıt Yenileme', 'Belge Talebi', 'Görüntüleme işlemleri', 'Hesap Ayarları' menülerini içermektedir.
- ◊ Ders Seçimi/Kayıt Yenileme Menüsünde; öğrenciler, her dönem başında yönetmeliklere uygun olarak ders seçimi yaparlar. Öğrenciler ders seçiminde;

♣ Zorunlu dersler, alttan dersler(varsa), döneme göre sayısı değişen seçmeli dersler, GNO < 2.00 ise koşullu dersleri, GNO < 1.80 ise gereken yarıyıllardan ders alınamaz, önceki yıla ait dersleri seçmeleri zorunludur.

♣ Alttan dersleri yoksa ve GNO => 3.00 ise üst dönemden de ders seçebilirler.

♣ Notu “DD-DC” olan dersler GNO'ya bakılmaksızın yükseltmeye alınabilir, transkriptten silinmez. GNO 2.00'in üzerinde ise notu “CC” ve üzeri dersler yükseltmeye alınabilir.

♣ Seçmeli ders/derslerden başarısız (FD, FF, YS ve DS) olan öğrenci, bu ders/derslerin yerine danışmanı tarafından uygun görülen ve kredisi eşit olan başka bir seçmeli ders/dersler alabilir.

♣ Genel not ortalaması 3.00 ve üzeri öğrenciler seçmeli ders sınırlamalarından muaf olup, istedikleri seçmeli derslere kayıt olabilirler ve kontenjana dahil edilmezler.

♣ Öğrencilerin seçtiği dersler 45-48 (7. Ve 8. Yarıyıl İçin) AKTS'yi geçemez. Bir eğitim-öğretim yılında önlisans/lisans programları için ders ve uygulama kredisi toplamı 60 AKTS'dir. Önlisans programları toplam 120, dört yıllık lisans programlarında 240, beş yıllık lisans programlarında ise toplam 300 AKTS'dir.

◇ Derslerim Menüsünden; öğrenciler, ders ataması gerçekleşmişse aldıkları dersleri ve notlarını, aldıkları dersler ile ilgili genel bilgileri, hafta içeriklerini, ödevlerini ve ders bilgilerini görüntüleyebilirler.

◇ Belge Menüsünden; öğrenciler, öğrenci belgesi, transkript ve öğrenci ders içeriği raporunu görüntüleyebilirler.

◇ Görüntüleme İşlemleri Menüsünden; öğrenciler, ders programı, danışman bilgileri ve takvimi görüntüleyebilirler.

## • 2. Öğretim Elemanı Sistemi

◇ Öğretim elemanlarının temel görevleri, kendilerine atanmış derslerle ilgili işlemleri yönetmek, not girişi yapmak, devamsızlık girişi yapmak ve derslerin akademik içeriğini yönetmektir.

◇ Öğretim Elemanı Sistemi, ‘Derslerim’, ‘Görüntüleme işlemleri’ ve ‘Hesap Ayarları’ menülerini içermektedir.

◇ Derslerim menüsünden; öğretim elemanı, İdareci tarafından kendisine atanmış dersleri görüntüleyebilir ve atanan derslere ait işlemler gerçekleştirebilir. Öğretim elemanı işlemlerden; ders detaylarını, yoklama raporunu, sınıf listesini, devam listesini, vize/final sınavı yoklama listesini, vize/final sonuç raporunu, öğrenci not listesini ve bütünlümeye girmek isteyen öğrenci listesini görüntüleyebilir ve kendilerine atanan dersler için öğrencilerin sınav notlarını girerler.

♣ Not girişi her ders için bir ara sınav (%40) ve bir final sınavı (%60) notu olarak girmelidir. Ön tanımlı olarak ara sınav ve final sınav yüzdesi %40 - %60

olarak atanmıştır lakin öğretim elemanı ara sınav ve final sınavı yüzdesini değiştirebilir.

♣ Öğretim elemanı not girişini yaptıktan sonra sisteme notları kaydettiğinde sistem tarafından otomatik olarak yüzdelere göre yıl sonu not ortalaması ve harf notu hesaplamaları yapılır.

◇ Görüntüleme işlemleri menüsünden; öğretim elemanları, tüm öğrencileri, haftalık ders programlarını ve yoklama bilgilerini görüntüleyebilir. Aynı zamanda devamsızlık girişi de yapabilirler. (Öğrenciler teorik derslerin en az %70'ine ve uygulamalı derslerin en az %80'ine devam etmeleri zorunludur.)

### • 3. Danışman Sistemi

◇ Danışmanların temel görevi öğrenciler ile derslerin eşleştirilmesini kontrol edip kayıtlanma raporunu onaylamalarıdır.

◇ Danışman Sistemi, 'Danışmanlık İşlemleri' ve 'Hesap Ayarları' menülerini içermektedir.

◇ Danışmanlık İşlemleri Menüsünden; Danışman, danışman olduğu öğrencileri görüntüleyebilir ve danışman olduğu öğrencileri ait transkript, öğrenci bilgileri, ders kayıtlanma kontrol sayfası ve ders kayıtlanmasını görüntüleyebilir.

♣ Öğrenci ders seçimini danışman onayına göndermişse danışman, öğrencilerin ve derslerin eşleştirilmesini kontrol eder ve onaylar. Aynı zamanda öğrencinin seçili ders seçiminde değişiklik yapıp (ekle/sil) öğrenci onayına gönderebilirler ya da öğrenci onayına gönderilmeden de kayıtlanma tamamlanabilir.

### • 4. İdareci Sistemi

◇ İdarecinin temel görevi dersleri kontrol edip uygun öğretim elemanlarına belirli koşullara uyarak ders ataması yapmasıdır.

◇ İdareci Sistemi, 'İdareci İşlemleri' ve 'Hesap Ayarları' menülerini içermektedir.

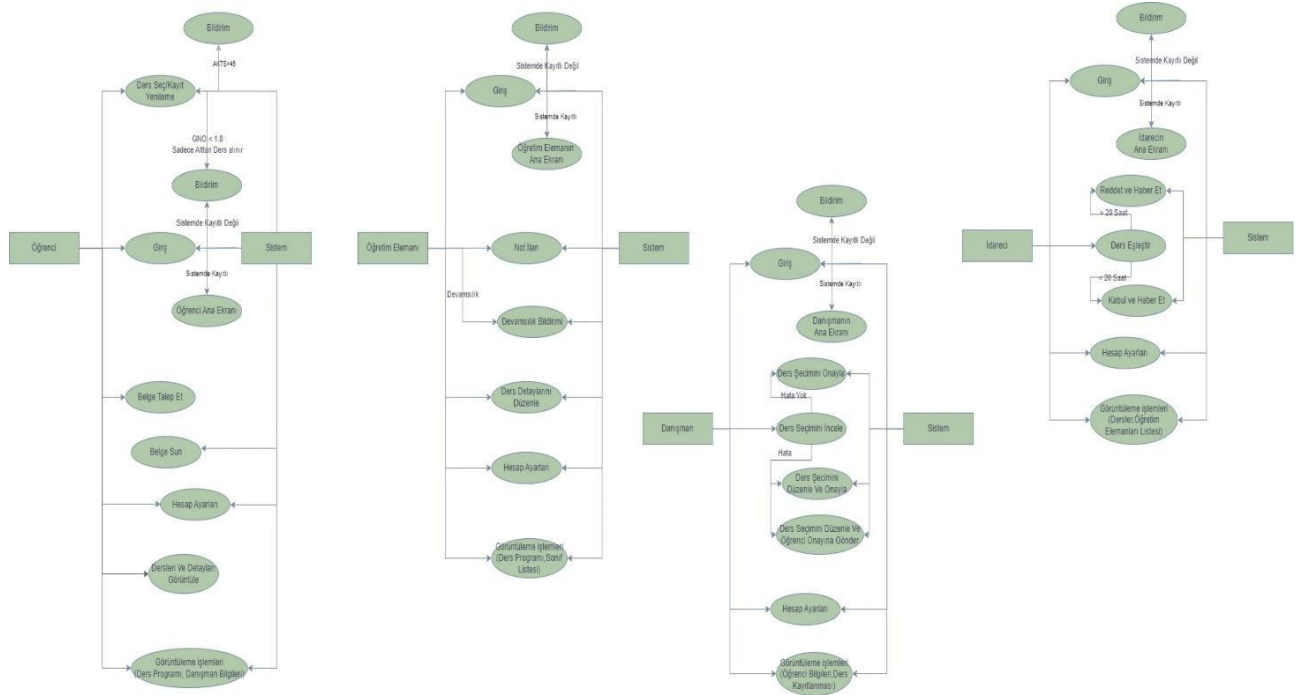
◇ İdareci İşlemleri Menüsünden; İdareci, Öğretim Elemanlarını ve dersleri görüntüleyebilir ve koşullara göre ders ataması gerçekleştirebilir.

♣ İdareci tarafından, bir öğretim elemanına haftalık en fazla 20 saatlik ders ve bir derse de sadece bir öğretim elemanı atanabilir.

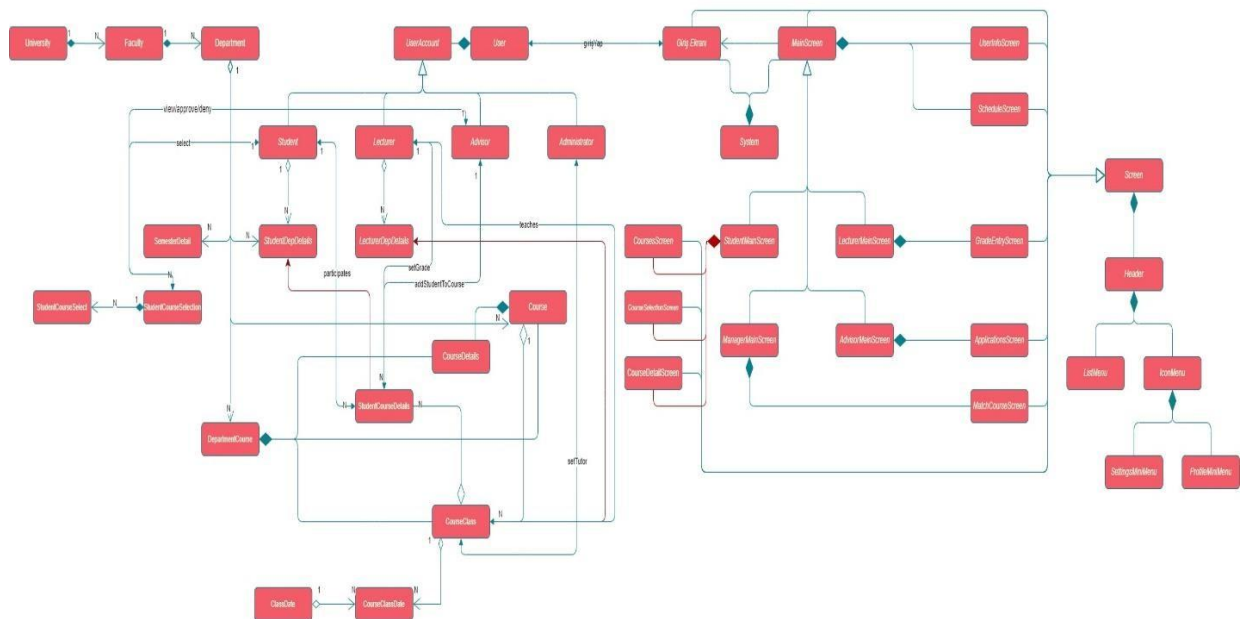
## TASARIM BÖLÜMÜ

Tasarımlar draw.io ve figma üzerinden yapılmış olup alınan görüntüler jpeg formatındadır. Bu sebepten dolayı görünmeyen tasarımlar için final raporu ile birlikte ek olarak klasör içerisinde tasarımların jpeg formatı sunulmuştur.

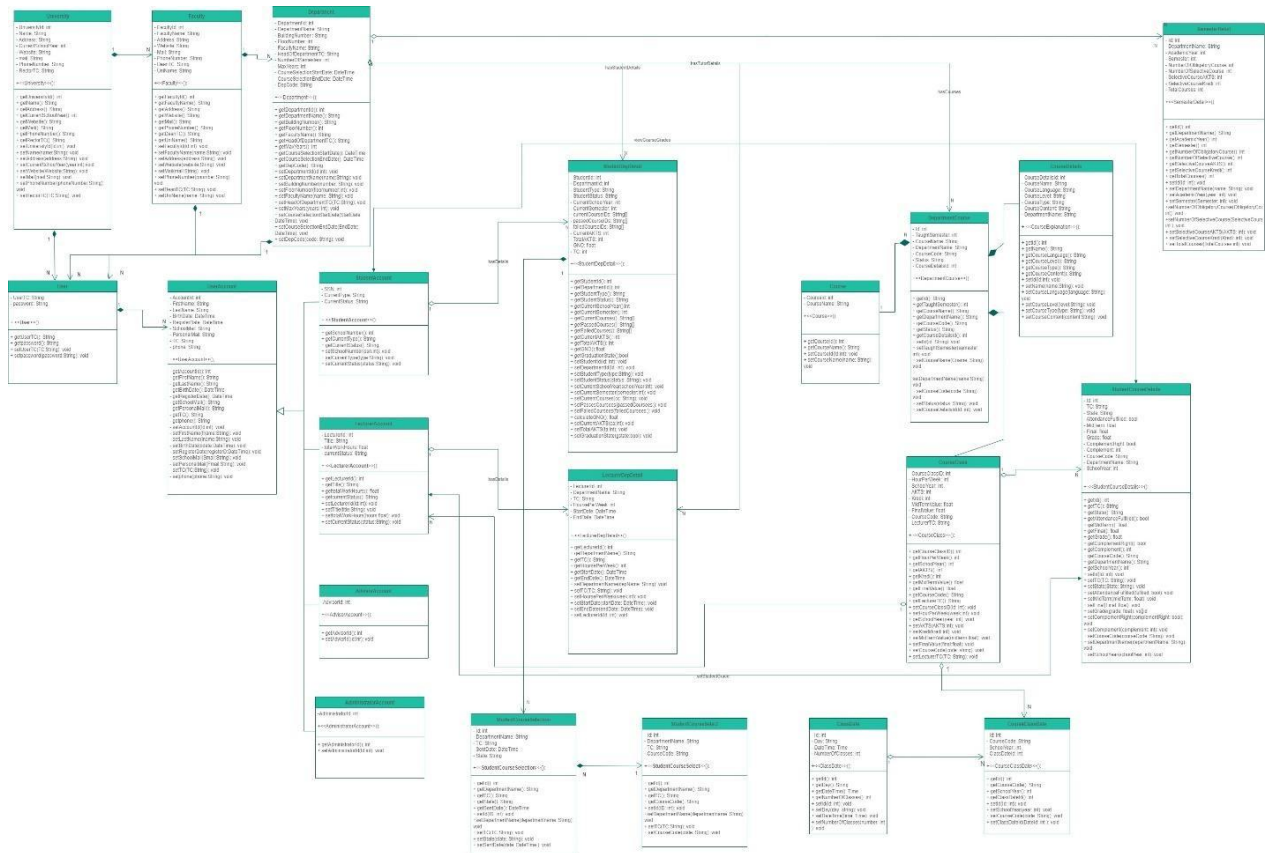
## 1- Special USE CASEs



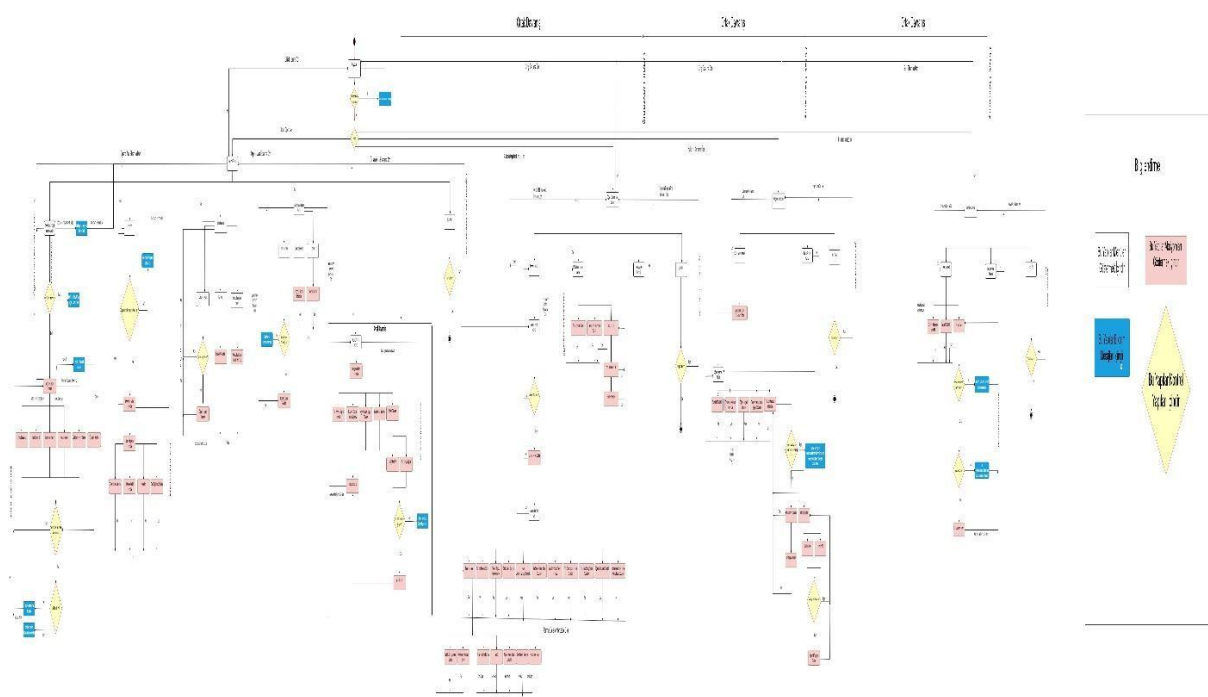
## 2- UML Concept Level



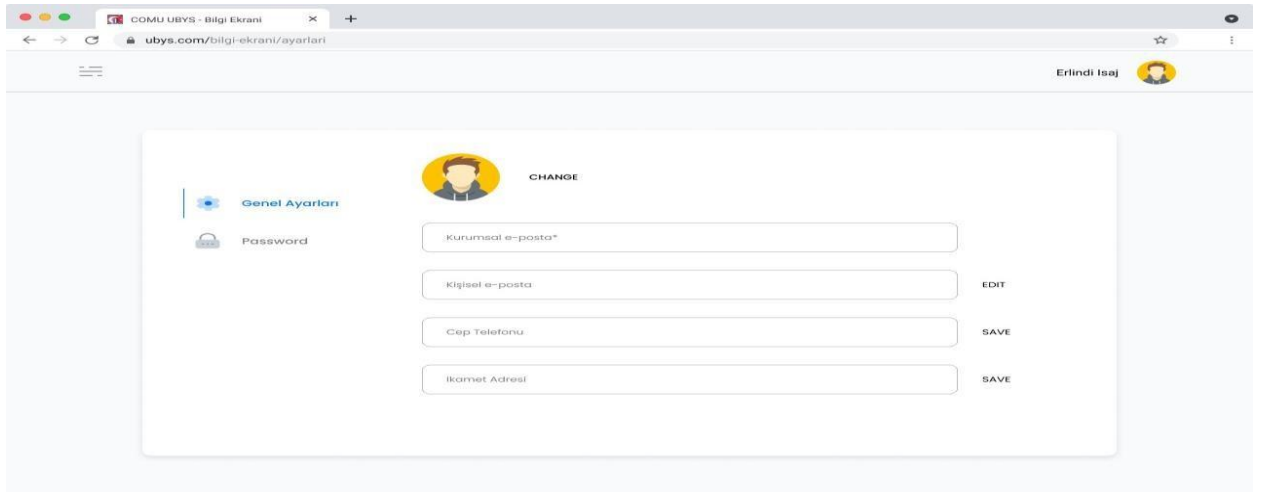
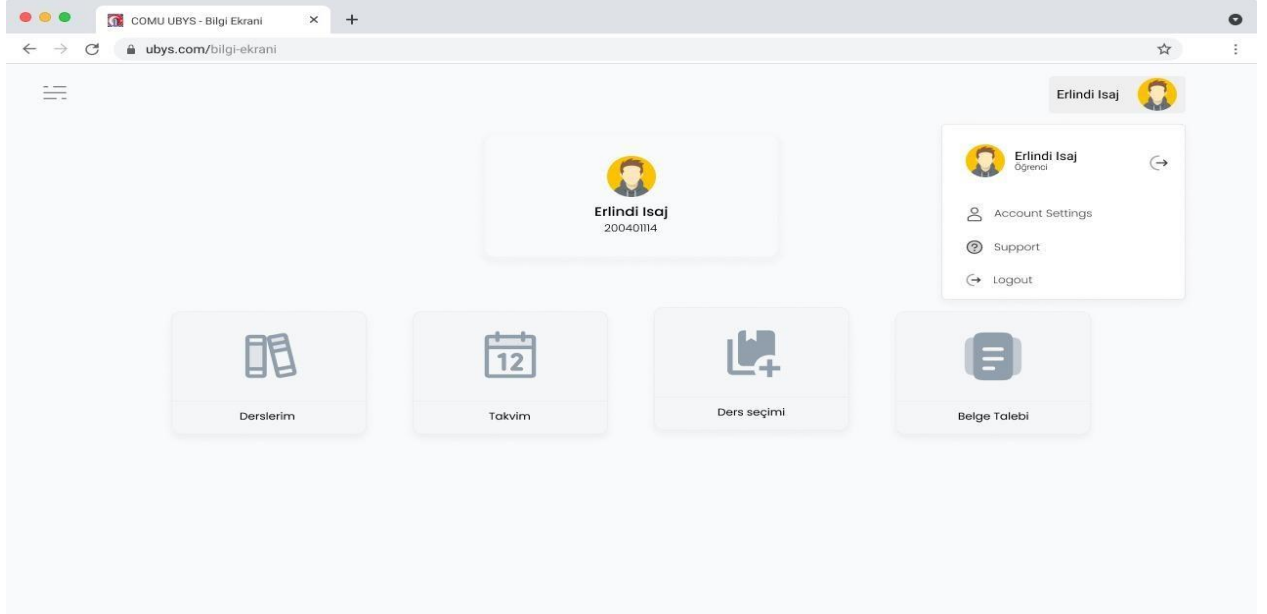
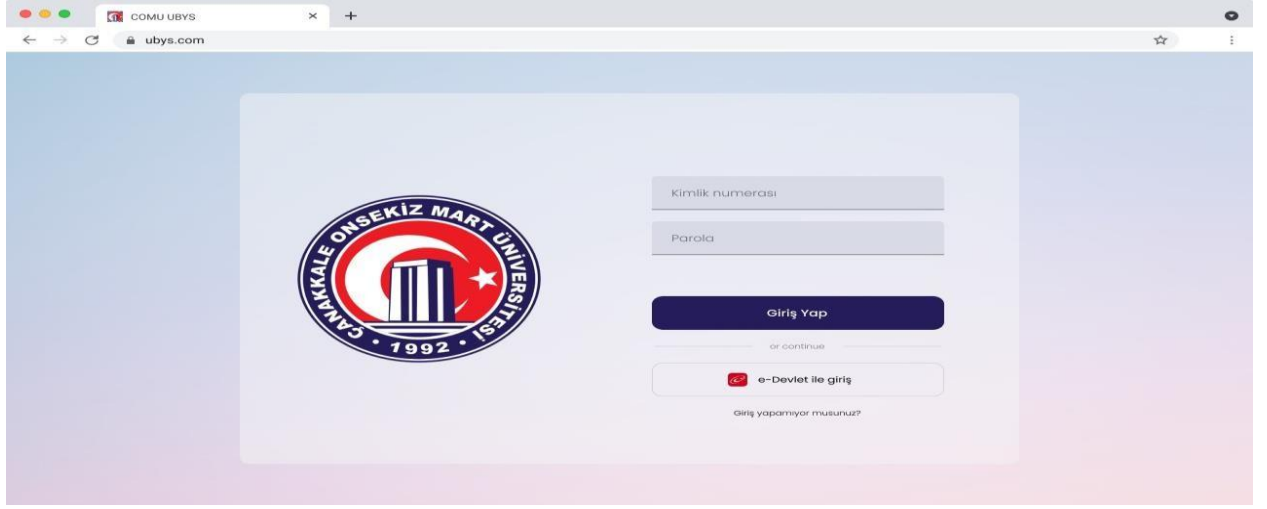
### 3- UML Implemtation Level



### 4- State Diagram



## 5- Ekran Tasarımları



COMU UBYS - Bilgi Ekranı

ubys.com/bilgi-ekrani/derssecimi

Erlindi Isaj

Danışman: Doç. Dr. Erlindi Isaj / Sınıf: 2 / Ders Dönemi: 3 / Durum: Ders Seçimi

Danışmana Gönder

Seçili Dersler

En Fazla AKTS	45	Seçili AKTS	10	Kalan AKTS	35
Sıra	Ders Kodu	Ders Adı	AKTS		
1	BLM2002	Programlama Dilleri Kavramları	5		
2	BLM3012	İleri Veritabanı Sistemleri	5		
2	BLM3012	İleri Veritabanı Sistemleri	5		
2	BLM3012	İleri Veritabanı Sistemleri	5		
2	BLM3012	İleri Veritabanı Sistemleri	5		
2	BLM3012	İleri Veritabanı Sistemleri	5		

Seçilmiş Dersler

Zorunlu Seçilmiş Dersler

Seçmeli Hakkında

Açıklama

Zorunlu Dersler

Üst Dönem Dersler

Başarılı Olunan Dersler

Seçmeli Dersler

Program Dışı Dersler

Burada bölümünüzde ait açmış veya daha önceki dönemlerinizde kaldığınız dersler yer almaktadır. (Önceki dönemde kalıp bu dönem açılmamışsa otomatik kayıtlıdır ve o dersleri burada göremiyorsunuz.) Mezun olabilmek için tamamlamanız zorunlu derslerdir.

Seç	Ders Kodu	Ders Adı	AKTS	Şube	Açıklama
+ Seç	BLM2002	Programlama Dilleri Kavramları	5	A - Doç. Dr. Erlindi Isaj	Alabilir
+ Seç	BLM2002	Programlama Dilleri Kavramları	5	A - Doç. Dr. Erlindi Isaj	Alabilir
+ Seç	BLM2002	Programlama Dilleri Kavramları	5	A - Doç. Dr. Erlindi Isaj	Alabilir
+ Seç	BLM2002	Programlama Dilleri Kavramları	5	A - Doç. Dr. Erlindi Isaj	Alabilir
+ Seç	BLM2002	Programlama Dilleri Kavramları	5	A - Doç. Dr. Erlindi Isaj	Alabilir
+ Seç	BLM2002	Programlama Dilleri Kavramları	5	A - Doç. Dr. Erlindi Isaj	Alabilir

COMU UBYS - Bilgi Ekranı

ubys.com/bilgi-ekrani/derslerim

Erlindi Isaj

2023 - Bahar

Ders Kodu	Ders Adı	Kredi	AKTS	Devam Durumu	Geçme Notu	HBN	Başarı Durumu
BLM-3002	Programlama Dilleri Kavramları	3,00	5,00	Devamlı	90,00	AA	Başarılı
				Vize: 90,00			
				Final: 90,00			
				Bütünleme: -			
BLM-3002	Programlama Dilleri Kavramları	3,00	5,00	Devamlı	90,00	AA	Başarılı
				Vize: 90,00			
				Final: 90,00			
				Bütünleme: -			
BLM-3002	Programlama Dilleri Kavramları	3,00	5,00	Devamlı	90,00	AA	Başarılı
				Vize: 90,00			
				Final: 90,00			
				Bütünleme: -			
BLM-3002	Programlama Dilleri Kavramları	3,00	5,00	Devamlı	90,00	AA	Başarılı
				Vize: 90,00			
				Final: 90,00			
				Bütünleme: -			

Geçmiş Dönem Derslerini Göster

2023 - Güz

Ders Kodu	Ders Adı	Kredi	AKTS	Dersin Koordinatörü	Devam Durumu	Geçme Notu	HBN	Başarı Durumu
-----------	----------	-------	------	---------------------	--------------	------------	-----	---------------



## SONUÇ BÖLÜMÜ

Analiz ve tasarıma uygun olarak yapmış olduğumuz sistemin gerçekleşmesi için kullandığımız frontend framework'ü React.js'dir ve bu framework'ü tercih etmemizin birkaç önemli sebebi bulunmaktadır. İlk olarak, bu framework'e olan aşinalığımız kodlama aşamasında işlerimizi kolaylaştırdı. React.js, JavaScript framework'leri arasında çok popülerdir ve bu sayede geniş bir topluluğa sahiptir. Bu geniş topluluk, StackOverflow gibi siteler üzerinden kolaylıkla yardım almamızı sağladı. Ayrıca, React.js ile kodlama yaparken Visual Studio Code platformunu tercih ettik. Visual Studio Code, kullanım kolaylığı ve çok sayıda eklentiye sahip olması nedeniyle tercih edildi. Bu eklentiler, isteğimize bağlı olarak kişiselleştirme yapmamızı sağladı.

Uygulamamızda karşılaştığımız en büyük sorunlardan biri verilerin uygulama içinde taşınmasını sağlamaktı. Bu sorunu çözmek için React Redux kullandık. React Redux ile verilerin yönetimini kolay bir hale getirdik ve uygulama içi veri akışını daha etkin bir şekilde kontrol edebildik.

Backend tarafında ise C# dilini kullandık ve .NET framework'ü tercih ettik. .NET, web uygulamaları geliştirme alanında çok popüler bir framework olup, bu alanda deneyim kazanmak istediğimiz için bu seçimi yaptık. Projemizin sonunda, .NET ile temel seviyede web uygulamaları geliştirmeyi öğrendik. Kodlama yaparken Visual Studio'yu tercih ettik çünkü .NET ile entegrasyonlu bir şekilde gelmekte ve web geliştirme için özel araçlar sunarak kodlamayı daha rahat hale getirmektedir.

Backend tarafında genel olarak çok sayıda Entity ve DTO ile çalıştığımız için bu nesneleri yönetmek başlangıçta zorlu oldu. Ancak, EntityFramework kullanarak bu işlemleri hızlandırıp basit bir hale getirdik. Ayrıca, uygulamamızda kimlik doğrulaması yaparak işlemleri yönlendirmek için JWT (JSON Web Token) kullandık. JWT, sektörde çok tercih edilen ve güvenilir bir yöntem olduğuna inandığımız için bu seçimi yaptık.

Özetle, projemizde frontend için React.js ve backend için .NET framework'lerini tercih ettik. React.js'nin popüleritesi ve geniş topluluğu, karşılaştığımız sorunları çözmemize yardımcı oldu. Visual Studio Code'un kişiselleştirilebilir yapısı, kodlama sürecimizi kolaylaştırdı. Backend tarafında .NET ve EntityFramework kullanarak web uygulamaları geliştirme ve veri yönetimini etkin bir şekilde gerçekleştirdik. JWT ile kimlik doğrulama, uygulamamıza güvenlik katmanı ekledi. Bu teknolojileri kullanarak, hem frontend hem de backend tarafında başarılı bir proje geliştirdik.

## KATKI BÖLÜMÜ

Projede değişen gereksinimler gibi önceden belirlenemeyen durumlara ve projeyi etkileyebilecek her tür değişikliğe karşı esneklik sağlayan Çevik yazılım geliştirme modeli benimsenmiş olup, ekip üyelerinin göstermiş olduğu özveri ve emek sonucunda projemizi tamamlamış bulunmaktayız. Projeye başlamadan önce görev dağılımları sağlanarak kimin hangi kısımlarda görev alacağı önceden belirlenmiş olup, her bir ekip üyesi görev dağılımına sadık kalmış ve gereken özveriyi göstermiştir.

GitHub Proje Link: <https://github.com/Bahayll/GraduationProject>