



**T.C**  
**KOCAELİ SAĞLIK VE TEKNOLOJİ ÜNİVERSİTESİ**  
**MÜHENDİSLİK VE DOĞA BİLİMLERİ FAKÜLTESİ**  
**BİLGİSAYAR/YAZILIM MÜHENDİSLİĞİ**

**HAFTALIK DERS PROGRAMI PROJESİ**

**HAZIRLAYANLAR**

**AHMET EREN ŞENGÜL 220502036**

**BAŞAR BURAK ÜNAL 220501025**

**YUSUF USTAOĞLU 220502003**

**EFE ÖZTAŞ 220501014**

**ONUR KOCA 240502040**

**DERS SORUMLUSU:**  
**Dr. Öğr. Üyesi ELİF PINAR HACİBEYOĞLU**

**TARİH: 22.03.2025**

# 1 GİRİŞ

## 1.1 Projenin amacı

- Ders Programı Yönetim Sistemi, üniversite veya eğitim kurumlarında haftalık ders programlarının etkin ve pratik bir şekilde oluşturulması, yönetilmesi ve görüntülenmesi amacıyla geliştirilmiştir. Bu sistem, akademik takvim içerisinde ders programlamayı kolaylaştırarak, idari personel ve öğretim üyelerinin zaman ve kaynak yönetimini optimize etmeyi hedeflemektedir.

## 1.2 Projenin gerçekleştirilmesinde beklenenler:

- Ders, bölüm, derslik ve kullanıcı bilgilerinin veritabanında saklanması
- Farklı kullanıcı rollerine (admin, öğretim üyesi) göre erişim kontrolü
- Ders programı oluşturma ve çakışma kontrolü (aynı derslik veya aynı öğretim üyesi için)
- Derslere öğretim üyesi atama imkanı
- Ders programının Excel formatında dışa aktarılabilmesi
- Kullanıcı dostu arayüz ile kolay ders programı yönetimi
- Dönemlere göre sınıflandırılmış ders görüntüleme

# 2 GEREKSİNİM ANALİZİ

## 2.1 Arayüz gereksinimleri

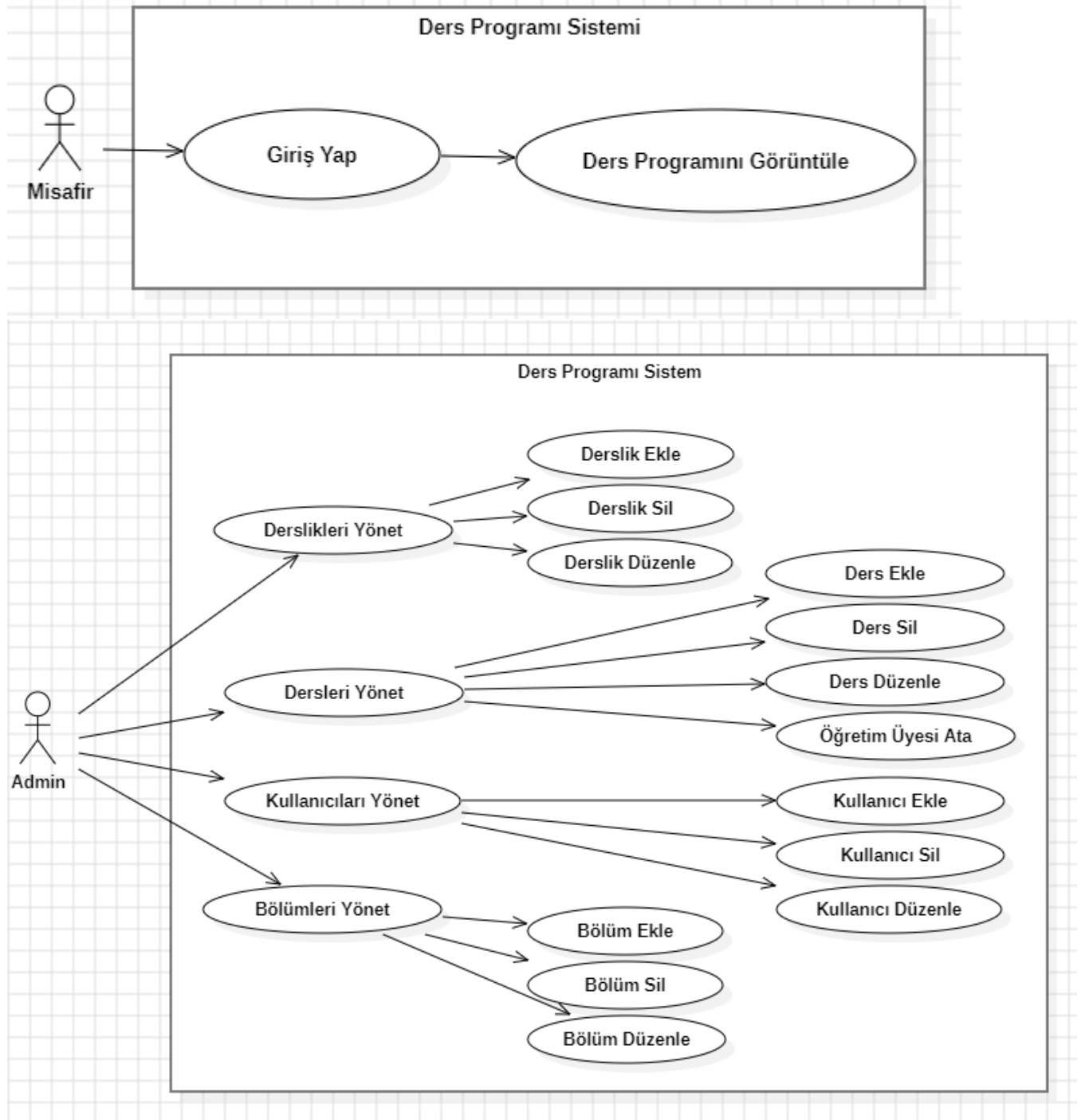
- Kullanım kolaylığını arttırmak için Karanlık Mod özelliği ekledik
- Basit ve sezgisel bir kullanıcı arayüzü
- Duyarlı (responsive) web tasarımı ile farklı cihazlarda (masaüstü, tablet, mobil) kullanılabilir olması
- Ders programının günlere ve saatlere göre tablo formunda görüntülenmesi
- Ekleme, silme ve düzenleme işlemleri için anlaşılır formlar
- İşlem başarısı veya hataları için kullanıcıya geribildirim sağlayan bildirimler
- Kullanıcı rolüne göre özelleştirilmiş görünüm ve işlevsellik
- Filtreleme ve arama özellikleri ile istenilen bilgilere kolay erişim

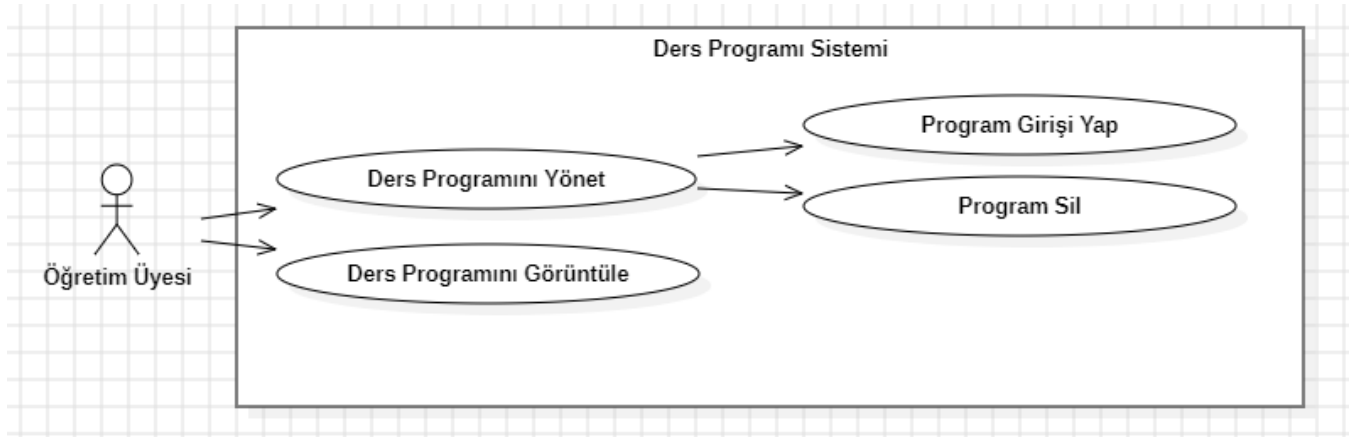
## 2.2 Fonksiyonel gereksinimler

- Sisteme kullanıcı ekleme, silme ve düzenleme
- Kullanıcı rolleri (admin, öğretim üyesi) belirleme
- Kullanıcı giriş kontrolü ve oturum yönetimi
- Bölüm ekleme, düzenleme ve silme
- Derslik ekleme, düzenleme ve silme
- Ders ekleme, düzenleme ve silme
- Derslere öğretim üyesi atama
- Seçilen gün ve saat dilimine ders ekleyebilme
- Var olan programdaki dersleri kaldırabilme
- Aynı dersliğe aynı saatte farklı ders eklenememe kontrolü
- Aynı öğretim üyesinin aynı saatte farklı derslere atanamaması kontrolü
- Ders programını Excel formatında dışa aktarma

- Profesyonel formatlanmış Excel raporu oluşturma
- Sınıflara (dönemlere) göre filtrelenmiş ders programı görüntüleme
- Haftalık ders programını tablo formatında görüntüleme

## 2.3 Use-Case diyagramı

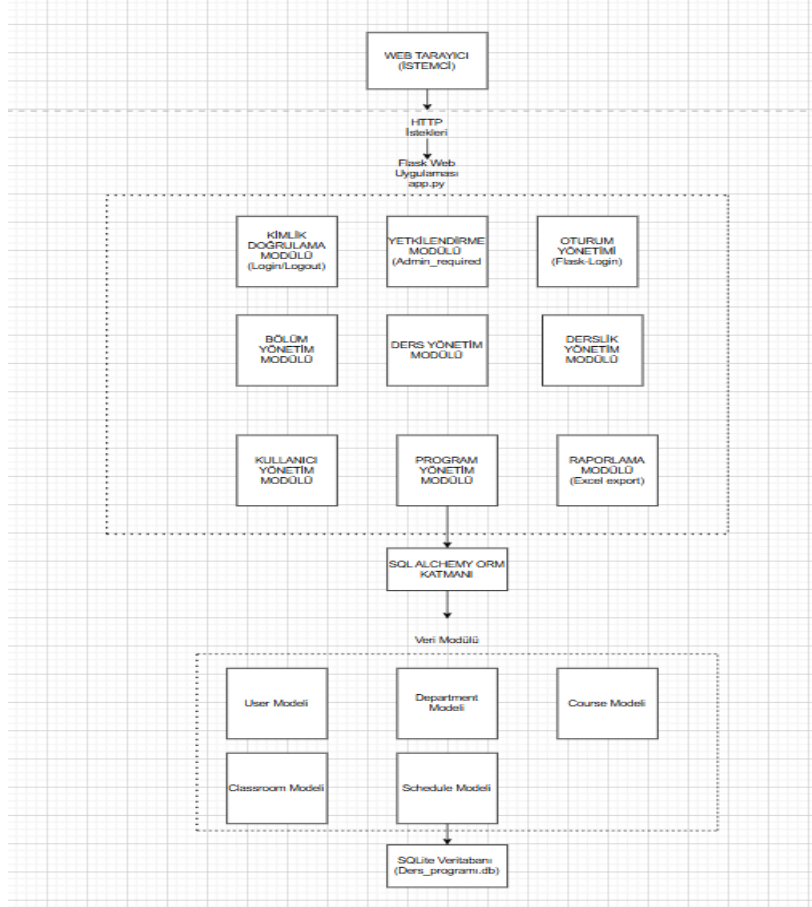




## 3 TASARIM

### 3.1 Mimari tasarım

- Veri Katmanı: SQLite veritabanı ile veri depolama ve erişim
- İş Mantığı Katmanı: Python Flask uygulamasında uygulama mantığının yönetimi
- Sunum Katmanı: HTML, CSS ve JavaScript kullanılarak oluşturulan web arayüzü
- Kullanıcı Yönetimi: Kullanıcı kimlik doğrulama ve yetkilendirme
- Veri Yönetimi: Bölüm, derslik, ders ve program verilerinin yönetimi
- Çakışma Kontrolü: Programda çakışmaları önleyen kontrol mekanizmaları
- Raporlama: Excel formatında dışa aktarma işlemleri



### 3.2 Kullanılacak teknolojiler

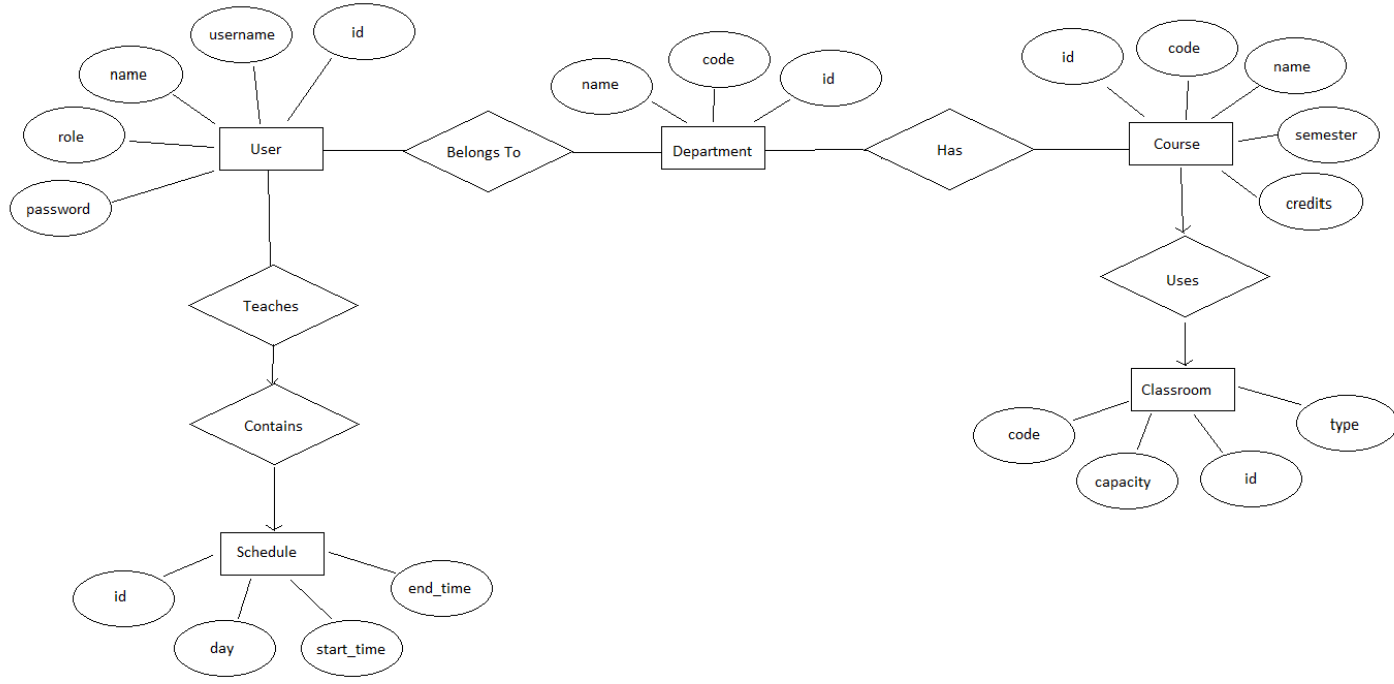
- Programlama Dili: Python 3
- Web Framework: Flask
- Veritabanı: SQLite
- ORM (Nesne İlişkisel Eşleştirici): SQLAlchemy
- Kullanıcı Arayüzü: HTML5, CSS3, JavaScript, Bootstrap 5
- Excel İşlemleri: OpenPyXL kütüphanesi
- Kimlik Doğrulama: Flask-Login
- Flask: Web uygulaması geliştirme
- Flask-Login: Kullanıcı oturum yönetimi
- Flask-SQLAlchemy: Veritabanı ORM işlemleri
- OpenPyXL: Excel dosyalarının oluşturulması ve düzenlenmesi
- Werkzeug: Güvenli parola hash'leme

### 3.3 Veri tabanı tasarımı

- Sistemde SQLite veritabanı kullanılmaktadır.
- User: Kullanıcı bilgileri (id, username, password\_hash, role, name)
- Department: Bölüm bilgileri (id, code, name)
- Classroom: Derslik bilgileri (id, code, capacity)
- Course: Ders bilgileri (id, code, name, department\_id, instructor\_id, semester)
- Schedule: Ders programı kayıtları (id, course\_id, classroom\_id, day, start\_time, end\_time)

•  
İlişkiler:

- Bir bölümün birden fazla dersi olabilir (1-n ilişkisi)
- Bir öğretim üyesi birden fazla ders verebilir (1-n ilişkisi)
- Bir ders programda birden fazla yere eklenebilir (1-n ilişkisi)
- Bir derslik birden fazla programda kullanılabilir (1-n ilişkisi)



### 3.4 Kullanıcı arayüzü tasarımı

Please log in to access this page.

Giriş Yap

Kullanıcı Adı

Şifre

Giriş Yap

(Giriş ekranı)

Ders Programına Ders Ekle

Ders

Derslik

Gün

Başlangıç Saati

Bitiş Saati

Ders Seçin

Derslik Seçin

Gün Seçin

--:--

--:--

Ders Ekle

Haftalık Ders Programı

Excel Olarak İndir

Gün / Sınıf	1. Sınıf	2. Sınıf	3. Sınıf	4. Sınıf
Pazartesi				
Salı				
Çarşamba				
Perşembe				
Cuma				

## (Ders Programı Ekranı)

- Üst Navigasyon Çubuğu: Kullanıcı bilgileri ve menü erişimi
- Yan Menü: Modüllere hızlı erişim
- İçerik Alanı: Formlar ve tablolar için ana görüntüleme alanı
- Bildirim Sistemi: İşlem sonuçları hakkında geri bildirim
- Giriş Ekranı: Kullanıcı kimlik doğrulama
- Ana Sayfa: Özet bilgiler ve hızlı erişim
- Bölümler Sayfası: Bölüm yönetimi
- Dersler Sayfası: Ders yönetimi ve öğretim üyesi atama
- Derslikler Sayfası: Derslik yönetimi
- Kullanıcılar Sayfası: Kullanıcı yönetimi
- Ders Programı: Haftalık ders programı görüntüleme ve yönetimi

## 4 UYGULAMA

### 4.1 Kodlanan bileşenlerin açıklamaları

- models.py: Veritabanı modelleri ve ORM sınıfları
- User: Kullanıcı bilgilerini temsil eden sınıf
- Department: Bölüm bilgilerini temsil eden sınıf
- Classroom: Derslik bilgilerini temsil eden sınıf
- Course: Ders bilgilerini temsil eden sınıf
- Schedule: Ders programı bilgilerini temsil eden sınıf
- 
- app.py: Ana uygulama modülü
- Uygulama başlatma ve konfigürasyon
- Route tanımlamaları ve istek işleme
- Kullanıcı kimlik doğrulama
- Veritabanı işlemleri
- Excel dışı aktarma işlevleri
- Çakışma kontrolü algoritmaları
- Şablonlar: Kullanıcı arayüzü HTML şablonları
- base.html: Ana şablon yapısı
- login.html: Giriş ekranı
- departments.html: Bölüm yönetimi
- courses.html: Ders yönetimi
- classrooms.html: Derslik yönetimi
- users.html: Kullanıcı yönetimi
- view\_schedule.html: Ders programı görüntüleme
- edit\_course.html: Ders düzenleme

### 4.2 Görev dağılımı

- Ekip olarak, haftalık ders programı projesinin her bir aşamasında birlikte ilerledik.
- Rapor ekip üyeleri tarafından iletişimi koparmadan birlikte hazırlanmıştır.

### 4.3 Karşılaşılan zorluklar ve çözüm yöntemleri

- Çakışma Kontrolü Sorunu: Ders ve öğretim üyelerinin çakışmaması için kontrol eden algoritmalar oluşturduk
- Excel Raporlama Formatı Oluşturma Sorunu: OpenPyXL kütüphanesi kullanarak Excel raporu oluşturulmasını sağladık.
- Öğretim Üyesi Ataması Sorunu: Course modeline (instructor\_id) alanı eklendi ve öğretim üyesi bilgileri ile ilişkilendirdik.

### 4.4 Proje isterlerine göre eksik yönler

- Eksik yön bulunmamaktadır.



## **5 TEST VE DOĞRULAMA**

### **5.1 Yazılımın test süreci**

- Yazılım test sürecinde testler proje süresi boyunca manuel olarak yapılmıştır.
- Herhangi bir test kodu geliştirilmemiştir.

### **5.2 Yazılımın doğrulanması**

- Kullanıcı giriş sistemi sorunsuz çalışmaktadır.
- Rol bazlı yetkilendirme sistemi (admin, öğretim üyesi) doğru şekilde işlemektedir.
- Yetkisiz erişim denemeleri başarılı bir şekilde engellenmektedir.
- Bölüm ekleme işlevi düzgün çalışmaktadır.
- Bölüm silme işlemi sorunsuz gerçekleşmektedir.
- Silinen bölümlerin veritabanından tamamen kaldırıldığı doğrulanmıştır.
- Ders ekleme işlevi hatasız çalışmaktadır.
- Ders düzenleme işlemleri eksiksiz gerçekleştirilebilmektedir.
- Derslere öğretim üyesi atama süreci doğru şekilde çalışmaktadır.
- Derslik ekleme işlemi sorunsuz tamamlanmaktadır.
- Derslik düzenleme ve kapasite güncelleme işlemleri başarılı bir şekilde gerçekleştirilmektedir.
- Aynı derslikte aynı saat diliminde farklı derslerin eklenmesi engellenmektedir.
- Aynı öğretim üyesinin aynı saatte farklı derslere atanması önlenmektedir.
- Excel dosyası oluşturma ve indirme işlevi sorunsuz çalışmaktadır.

## 6 KAYNAKÇA

<https://flask-login.readthedocs.io>

<https://docs.sqlalchemy.org/>

## 7 GİTHUB LİNKLERİ

Onur Koca <https://github.com/OnurKoca1>

Ahmet Eren Şengül <https://github.com/Eren1213>

Yusuf Ustaoglu <https://github.com/katlicia>

Efe Öztaş <https://github.com/EOztas>

Başar Burak Ünal <https://github.com/BasarBurakUnal>