



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

**PROJE KONUSU: WEB ARAYÜZÜNDEN OTOMATİK DERS
PROGRAMI OLUŞTURMA**

**ÖĞRENCİ ADI: GÜREL BİLGİN, GİZEM YALÇIN, BERKAY
ARAS, ALİ AKSOY**

**ÖĞRENCİ NUMARASI: 220502041, 240501037, 220501033,
210501009**

DERS SORUMLUSU: ELİF PINAR HACİBEYOĞLU

TARİH: 07.05.2025

1 GİRİŞ

1.1 Projenin Amacı

Bu projenin amacı, okulumuz Kocaeli Sağlık ve Teknoloji Üniversitesi Mühendislik Fakültesi için sistematik haftalık ders programı oluşturma sistemi geliştirmektir. Sistem, Bilgisayar Mühendisliği (BLM) ve Yazılım Mühendisliği (YZM) bölümleri için belirli kısıtlar dahilinde ders çakışmalarını önleyen, derslik kapasitelerini dikkate alan ve kullanıcı dostu bir arayüz sunan bir yapıya sahiptir.

Projede Gerçekleştirilmesi Beklenenler

- Kullanıcı dostu veri giriş arayüzü oluşturulması
- DBMS kullanımı ve veri tabanının doğru tasarlanması
- Kullanıcı yönetimi (öğretim üyesi, öğrenci, yönetici ekleme-çıkarma)
- Bölüm yönetimi (bölüm ekleme-çıkarma)
- Ders yönetimi (ders ekleme-çıkarma)
- Derslik yönetimi (derslik ekleme-çıkarma)
- Haftalık program oluşturma (Pazartesi-Cuma, 09:00-17:00)
- Online dersler için saat düzenlemesi (17:00-21:00)
- Aynı sınıfa ait derslerin çakışmasının önlenmesi
- Öğretim üyelerinin uygunluk durumuna göre ders ataması
- Derslik kapasitesine uygun atama
- İki bölüme ortak verilen dersler için öncelikli yerleştirme
- Zorunlu ders saatleri olan dersler için uygun planlama
- Verilen şablonda ders programının Excel olarak oluşturulması
- Dinamik ve düzenlenebilir sistem mimarisi
- BLM ve YZM bölümleri için tüm kısıtlar dahilinde doğru sonuç üretilmesi

2 GEREKSİNİM ANALİZİ

2.1 Fonksiyonel Gereksinimler

- Öğretim üyesi, öğrenci ve yönetici hesaplarının oluşturulması ve yönetilmesi
- Kullanıcı rollerine göre erişim yetkilerinin belirlenmesi
- Kullanıcı giriş-çıkış işlemlerinin yönetilmesi

Bölüm Yönetimi

- Bölüm ekleme, düzenleme ve silme işlemleri
- Bölüm kodlarının tanımlanması (YZM, BLM)
- Bölümlere ait 8 yarıyıllık tüm derslerin listelenmesi

Ders Yönetimi

- Ders ekleme, düzenleme ve silme işlemleri
- Ders adı, kodu, haftalık saat sayısı, ders veren öğretim üyesi bilgilerinin tanımlanması
- İki bölümün ortak alabileceği derslerin belirlenmesi
- Zorunlu ders saatleri olan derslerin işaretlenmesi

Derslik Yönetimi

- Derslik ekleme, düzenleme ve silme işlemleri
- Derslik kapasitesi ve statüsünün (NORMAL, LAB) tanımlanması
- Dersliklerin kullanılabilirlik durumlarının takibi

Ders Programı Oluşturma

- Hafta içi günler (Pazartesi-Cuma) için program oluşturma
- Normal ders saatleri (09:00-17:00) ve online ders saatleri (17:00-21:00) için planlama
- Derslerin, belirlenen kısıtlamalar çerçevesinde otomatik yerleştirilmesi

Kısıtlama Yönetimi

- Aynı sınıfa ait derslerin çakışmasının engellenmesi
- Ortak öğrencisi olan derslerin çakışmasının engellenmesi
- Öğretim üyelerinin uygunluk durumlarına göre ders ataması yapılması
- Bir öğretim üyesinin aynı anda birden fazla ders vermesinin engellenmesi (ortak dersler hariç)
- Derslik kapasitesine uygun atamaların yapılması
- Zorunlu ders saatleri olan derslerin ilgili saatlere yerleştirilmesi

Veri Doğrulama ve Tutarlılık Kontrolü

- Veri girişlerinin doğruluğunun ve tutarlılığının kontrolü
- Çakışma ve kapasite aşımı durumlarında uyarı verilmesi
- Oluşturulan programın kısıtlamaları karşılayıp karşılamadığının kontrolü

Çıktı Yönetimi

- Oluşturulan ders programının verilmiş şablona uygun olarak Excel formatında dışa aktarılması
- Sınıf, bölüm veya öğretim üyesi bazında filtrelenmiş programların oluşturulması
- Program çıktılarının kaydedilmesi ve gerektiğinde tekrar kullanılabilmesi

Düzenleme ve Güncelleme

- Oluşturulan programın manuel olarak düzenlenebilmesi

- Belirli bir sınıf veya ders için program güncelleme imkanı
- Değişiklikler sonrası kısıtlama kontrollerinin tekrar yapılması

Veri Tabanı Entegrasyonu

- Tüm verilerin seçilen veri tabanı yönetim sistemi (MySQL) ile entegre çalışması
- Veri tabanı işlemlerinin güvenli ve verimli bir şekilde gerçekleştirilmesi
- Veri yedekleme ve geri yükleme fonksiyonlarının sağlanması

2.2 Arayüz Gereksinimleri

Bu projede geliştirilecek web arayüzü; kullanıcı dostu, erişilebilir ve rollerine göre özelleştirilmiş bir yapı sunmalıdır. Uygulamanın temel arayüz gereksinimleri aşağıdaki gibidir:

1. Giriş Sayfası

- Kullanıcıların sisteme giriş yapabilmesi için kullanıcı adı ve şifre alanları bulunmalıdır.
- Kullanıcı tipi seçimi (Program Koordinatörü, Öğretim Üyesi, Öğrenci) yapılmalıdır.
- Giriş bilgilerinin doğruluğu sunucu tarafında kontrol edilmelidir.

2. Program Koordinatörü Arayüzü

- Bölüm, öğretim üyesi, öğrenci, ders, derslik ekleme/silme/güncelleme işlemlerine uygun formlar sunulmalıdır.
- Ders programının otomatik olarak oluşturulmasına dair bir buton veya arayüz bileşeni bulunmalıdır.
- Herhangi bir derse ait alternatif saat ve sınıf seçenekleri görselleştirilmeli, kullanıcı bu seçeneklerden manuel seçim yapabilmelidir.
- Excel formatında veri yükleme/dışa aktarma özellikleri sunulmalıdır.

3. Öğretim Üyesi Arayüzü

- Kendi haftalık ders programını ayrıntılı şekilde görüntüleyebilmelidir.
- Diğer öğretim üyelerine ve sınıflara ait programları da sadece görüntüleme amaçlı erişebilmelidir.
- Kullanımı kolay, sade bir zaman çizelgesi arayüzü bulunmalıdır.

4. Öğrenci Arayüzü

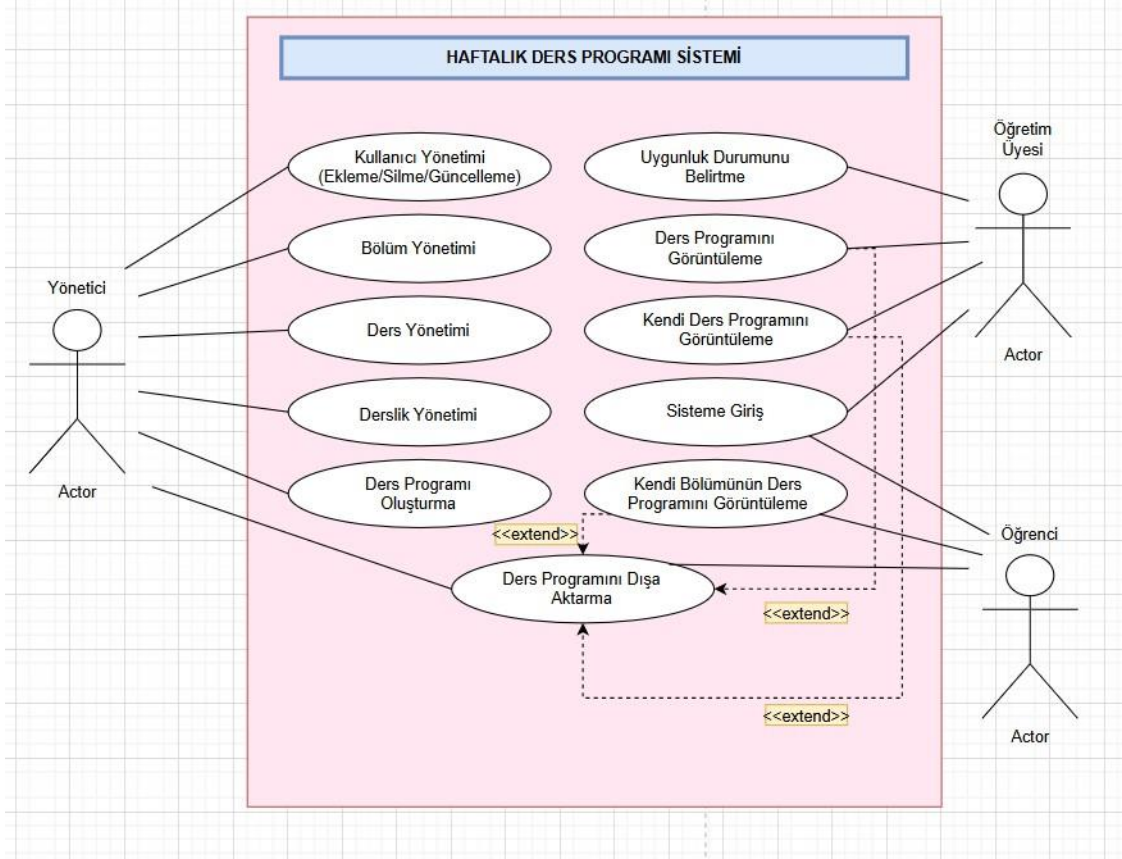
- Ders seçimi yapabileceği bir arayüz sunulmalıdır.
- Seçilen derslere göre haftalık program detaylı biçimde gösterilmelidir.
- Program görüntüsü Excel olarak indirilebilmelidir.

5. Genel Arayüz Özellikleri

- Tüm arayüzlerde responsive (mobil uyumlu) tasarım tercih edilmelidir.
- Form alanları hata kontrolü ile desteklenmeli, kullanıcıya açık geri bildirim verilmelidir.

- Renk, ikon ve buton tasarımlarıyla kullanıcı deneyimi artırılmalıdır.
- Tüm kullanıcı arayüzleri güvenli bağlantılarla (HTTPS) sunulmalıdır.

2.3 Use-Case Diyagramı



3 TASARIM

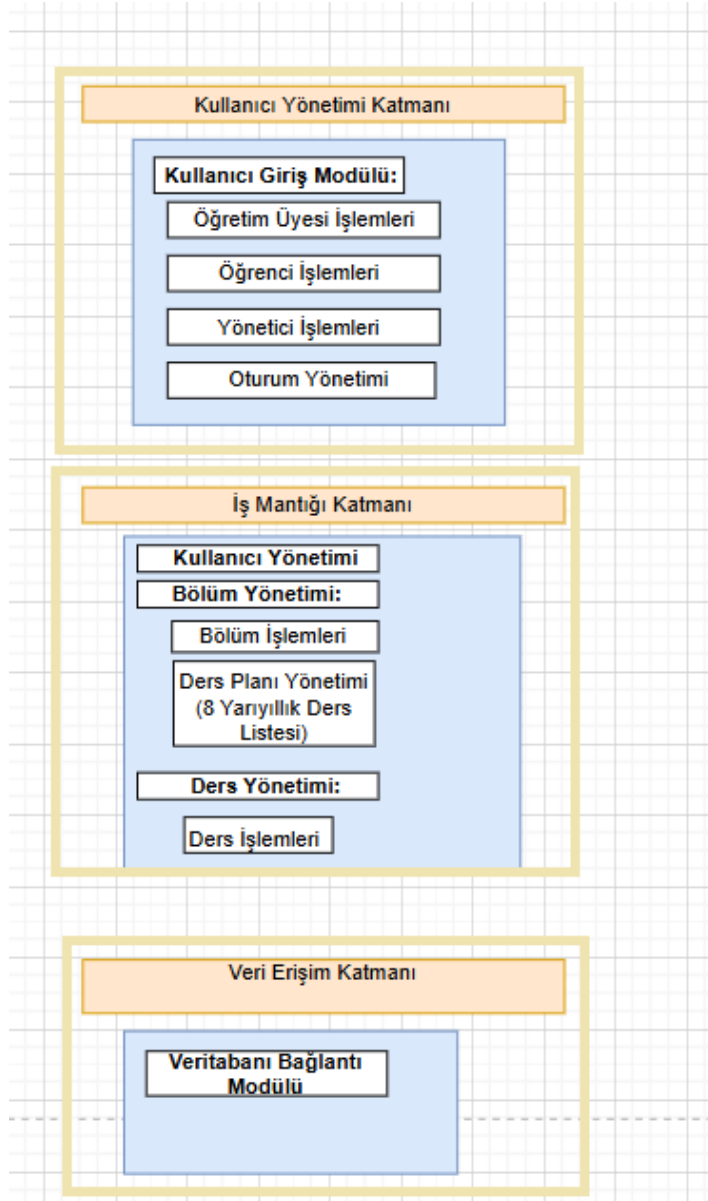
3.1 Mimari Tasarım

Projemizde belirli bir mimari tasarım kalıbı (örn. MVC, MVVM, Hexagonal) birebir uygulanmamıştır. Bununla birlikte, sistematik bir yapı ve düzen oluşturmak için modüler bir yaklaşım benimsenmiştir. Bu yaklaşım, uygulamanın farklı fonksiyonel bileşenlerini mantıksal olarak ayrılmış modüller halinde organize ederek, kod tabanının yönetilebilirliğini ve bakımını kolaylaştırmaktadır.

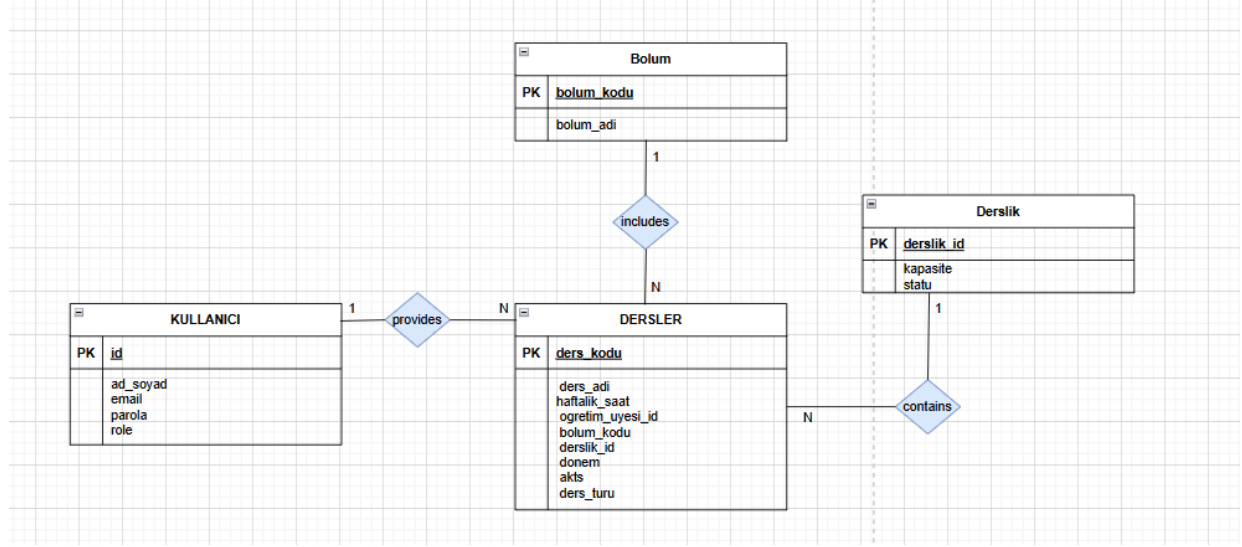
Uygulama içerisinde görev ve sorumluluklar belirli sınırlar çerçevesinde gruplandırılmış ve her bir modül kendi içinde tutarlı bir yapıya sahiptir. Kullanıcı yönetimi, bölüm yönetimi, ders yönetimi, derslik yönetimi gibi temel fonksiyonel birimler, belirli bir hiyerarşi ve düzen içerisinde çalışmaktadır.

Veri erişim katmanı ve iş mantığı katmanı arasında gevşek bir bağlantı sağlanarak, her bir bileşenin diğer bileşenlerden bağımsız olarak geliştirilmesi ve test edilmesi mümkün kılınmıştır.

Modül Diyagramı



ER Diyagramı:



3.2 Kullanılacak Teknolojiler

Ders programı oluşturma projemizin geliştirilmesinde çeşitli modern yazılım teknolojileri ve araçları bir araya getirilerek entegre bir çözüm oluşturulacaktır. Backend tarafında, veri yönetimi ve iş mantığı işlemleri için güçlü ve esnek bir yapı tercih edilmiştir. Frontend tarafında ise sade ve basit bir tasarım kullanılmıştır.

Proje Python dilinde yazılmıştır. Veritabanı yönetimi için MySQL tercih edilmiştir.

Excel işleme “openpyxl” ile yapılmıştır.

Web application framework olarak Bootstrap 5 ve Flask kullanılmıştır.

4 UYGULAMA

4.1 Kodlanan Bileşenlerin Açıklamaları

Bu projede, Flask tabanlı bir web uygulaması geliştirilmiş olup aşağıdaki yazılım bileşenleri kodlanmıştır:

1. Veritabanı Yönetim Bileşenleri

Database Sınıfı: MySQL ile bağlantı kuran, SQL komutlarını çalıştıran ve sonuçları geri döndüren merkezi sınıftır.

execute_query(query, params): Verilen SQL sorgusunu, opsiyonel parametrelerle birlikte çalıştırır. Veritabanına veri ekleme, silme ve güncelleme işlemleri bu fonksiyonla gerçekleştirilir.

fetch_all(query, params): SELECT sorgularını çalıştırarak sonuçları liste olarak döndürür.

2. Ders Programı Yönetim Bileşenleri

DersProgrami Sınıfı: Uygulamadaki temel iş mantığını kapsar. Derslerin eklenmesi, düzenlenmesi ve silinmesi işlemlerini içerir.

ders_ekle(data): Gelen JSON verisiyle yeni bir ders kaydını veritabanına ekler.

ders_duzenle(ders_kodu, data): Mevcut dersin bilgilerini günceller.

ders_sil(ders_kodu): Belirtilen ders koduna sahip kaydı siler.

3. Flask Web Bileşenleri (API Katmanı)

@app.route() Dekoratörleri: Her işlev için ayrı HTTP uç noktaları tanımlanmıştır.

POST /ders → Yeni ders ekler

PUT /ders/<ders_kodu> → Dersi günceller

DELETE /ders/<ders_kodu> → Dersi siler

GET /dersler → Tüm dersleri listeler

JSON Veri Yapısı: İstemciden alınan veriler JSON formatındadır ve sunucu yanıtları da JSON şeklindedir. Bu sayede modern istemci uygulamalarıyla uyumluluk sağlanır.

4. Kullanıcı Arayüzü Bileşenleri

Komut Satırı Arayüzü (Opsiyonel): Projenin erken sürümünde test amaçlı CLI menüsü kullanılmıştır. Sonraki sürümlerde web arayüzü veya harici istemci ile entegrasyon hedeflenmektedir.

Girdi Kontrolleri: input() ve form doğrulama işlemleriyle kullanıcıdan alınan veriler kontrol edilir.

5. Hata Yönetimi ve Doğrulama

Try-Except Blokları: Özellikle veritabanı ve kullanıcı işlemleri sırasında oluşabilecek hataları yakalar.

Yanıt Kodları: HTTP standartlarına uygun olarak 200 (başarılı), 400 (hatalı istek), 404 (bulunamadı) gibi yanıtlar döndürülür.

Kullanıcı Girdisi Doğrulama: Gelen JSON verisinin eksiksiz ve geçerli olup olmadığı kontrol edilir.

6. Program Akışı ve Kontrol Bileşenleri

Main Kontrolü (if name == "main"): Uygulamanın Flask sunucusu olarak başlatılmasını sağlar.

Debug Modu: app.run(debug=True) ile geliştirme sürecinde hata ayıklamayı kolaylaştırır.

Bu yapı, modern web uygulamalarında kullanılan RESTful API mimarisine uygun olarak geliştirilmiş ve genişletilebilir bir altyapı sunmaktadır.

4.2 Görev Dağılımı

Projenin tüm aşamalarında genel olarak beraber çalışılmış olup proje bitimi kod incelemesi topluca yapılmıştır. Rapor son kez beraber düzenlenmiştir.

Gürel Bilgin: Veritabanı tasarımı, temel CRUD işlemleri, main fonksiyonu yazımı ve menü sistemi geliştirilmesi, framework kullanarak web application arayüz yapımı.

Gizem Yalçın: Dokümantasyon ve yardım dosyalarının hazırlanması, DersProgrami sınıfının geliştirilmesi, rapor yazımı, arayüz yazımı.

Berkay Aras: Kısıtlama kontrolleri, kullanılabilirlik iyileştirmeleri, geribildirim mesajları kodlanması, test senaryoları hazırlanması, .

Ali Aksoy: Excel çıktı oluşturma modül yazımı, hata yönetimi mekanizmalarının oluşturulması, tüm proje kod review.

4.3 Karşılaşılan Zorluklar ve Çözüm Yöntemleri

Projede zaman yönetimini iyi ayarlayamamamızdan kaynaklanan kısıtlı vakit nedeniyle tüm proje isterlerini içeren bir proje ve raporu ortaya koyamadık. Kalan vakitte elimizden geldiğince eksikleri kapatmak için uğraştık.

Bu süreçte bizi en çok zorlayan konu kullanıcı rollerine göre özel arayüzler oluşturamamamızdı. Ayrıca bir önceki projeyle entegre etmede zorlandık. Hazır sistemi değiştirme konusunda çok sorun yaşadık. Kalan vakitte yapabildiğimiz şekilde bir web app ortaya koyuldu.

4.4 Proje İsterlerine Göre Eksik Yönler

- Kullanıcı rolüne göre özel arayüz
- Çakışmalar, derslik uygunluğu, öğretim üyesi uygunluğu gibi kurallar
- Derslik kapasitesi kontrolü
- Manuel müdahale

5. TEST VE DOĞRULAMA

Testler, manuel olarak yürütülmüştür.

Veritabanı işlemleri sonrasında yapılan doğrulamalarda, eklenen verilerin başarıyla kayıt altına alındığı, silinen verilerin sistemden kaldırıldığı ve düzenlenen verilerin doğru şekilde güncellendiği kontrol edilmiştir. Her işlem sonrası veritabanı yeniden sorgulanarak doğrulama yapılmıştır.

Ancak mevcut proje kapsamında kullanıcı oturumları ve rol bazlı giriş sistemi henüz entegre edilmediği için bu alanda test yapılmamıştır. Aynı şekilde, otomatik ders programı oluşturma ve çakışma kontrolü gibi gelişmiş algoritmalar henüz uygulanmadığından bu işlevlerin testleri de gerçekleştirilememiştir.

Program girdi alma konusunda yetersizdir. Bu sebeple test gerçekleştirilememiştir.

Grup Üyeleri GitHub Hesapları:

Gürel Bilgin: <https://github.com/GurelBilgin>

Gizem Yalçın: <https://github.com/gizemyalcinn>

Ali Aksoy: <https://github.com/aaksoy581>

Berkay Aras: <https://github.com/brkyaras>