



T.C

**KOCAELİ SAęLIK VE TEKNOLOJİ ÜNİVERSİTESİ
MÜHENDİSLİK VE DOęA BİLİMLERİ FAKÜLTESİ
YAZILIM MÜHENDİSLİęİ**

HAFTALIK DERS PROGRAMI OLUŞTURMA

Hazırlayan

**SUDENAZ ÇAKIROęLU, YAVUZHAN KURŞUN
YERDİNAT ALİKHAN, EREN YALAZ**

ÖęRENCİ NUMARASI

220502013, 230502047

220502050, 220502007

DERS SORUMLUSU

DR ÖęR ÜYESİ ELİF PINAR HACİBEYOęLU

05.05.2025

İÇİNDEKİLER

1. ÖZET (ABSTRACT)	1
2. GİRİŞ (INTRODUCTION)	2
3. GEREKSİNİM ANALİZİ	3
3.1. Fonksiyonel Gereksinimler	3
3.2. Fonksiyonel Olmayan Gereksinimler	4
3.3. Kısıtlar ve Varsayımlar	5
4. TASARIM (DESIGN)	6
4.1. Veritabanı Tasarımı	6
4.2. Yazılım Mimarisi	7
4.3. Algoritma Tasarımı	8
4.4. Arayüz Tasarımı	9
5. UYGULAMA (IMPLEMENTATION)	10
5.1. Kullanılan Teknolojiler	10
5.2. Modüller ve Kod Yapısı	11
5.3. İçe/Dışa Aktarma & Atama Fonksiyonları	12
5.4. Algoritma Uygulaması	13
5.5. Kullanıcı Arayüzü Uygulaması	14
6. TEST VE DOĞRULAMA	15
6.1. Test Yaklaşımı	15
6.2. Test Senaryoları	16
6.3. Test Sonuçları ve Değerlendirme	17
7. SONUÇ VE ÖNERİLER	18
8. KAYNAKÇA	19

Ödev No: 1	Tarih 05.05.2025	2/16
------------	------------------	------

1. ÖZET

Bu rapor, Python ve Django kullanılarak geliştirilmiş Ders Programı Yönetim Sistemi projesinin amaç, kapsam ve temel özelliklerini kısaca sunar.

Sistem, bir eğitim kurumunda bölümler, derslikler, öğretim üyeleri ve ders verilerini tanımlayıp yönetebilmeyi; backtracking tabanlı bir algoritma ile çakışmasız haftalık ders programı oluşturmayı sağlar.

Oluşturulan programlar, kullanıcı dostu admin paneli üzerinden görüntülenebilir ve istenildiğinde Excel formatında dışa aktarılabilir.

Ayrıca toplu ders verileri içe aktarma ve otomatik öğretim üyesi atama işlevleri de mevcuttur.

Proje, manuel zaman çizelgeleme sürecini otomatikleştirerek zaman tasarrufu sağlamak ve insan hatasını en aza indirmek amacıyla tasarlanmıştır.

2. GİRİŞ

Ders programı hazırlama işlemi, eğitim kurumlarının en kritik süreçlerinden biridir ve kurum kaynaklarının (derslik, hoca, zaman) verimli kullanılmasını doğrudan etkiler.

Farklı bölümlerin ihtiyaçları, öğretim üyelerinin müsaitlikleri ve ortak ders kısıtları gibi birden fazla değişkenin elle yönetilmesi; zaman kaybına, çakışmalara ve insan hatalarına yol açmaktadır.

Bu projede geliştirilen web tabanlı Ders Programı Yönetim Sistemi, bu karmaşıklığı ortadan kaldırmak amacıyla tasarlanmıştır.

Python ve Django altyapısı üzerinde çalışan sistem, ders, bölüm, derslik ve öğretim üyesi verilerini merkezi olarak yönetir; otomatik backtracking algoritmasıyla çakışmasız haftalık programlar oluşturur.

Raporda sırasıyla; Özet'ten sonra Giriş, Gereksinim Analizi, Tasarım, Uygulama, Test ve Sonuç bölümleri ele alınacaktır.

Ödev No: 1	Tarih 05.05.2025	3/16
------------	------------------	------

3. GEREKSİNİM ANALİZİ

3.1. Fonksiyonel Gereksinimler

- Sistem, Bölüm, Derslik, Öğretim Üyesi ve Ders modelleri için eksiksiz CRUD (Create, Read, Update, Delete) işlemleri sunmalıdır.
- Sistem, kullanıcıların toplu ders verilerini Excel veya CSV dosyalarından içe aktarabilmesine olanak tanımalıdır.
- Sistem, backtracking tabanlı algoritma kullanarak çakışmasız haftalık ders programı oluşturmaktadır.
- Sistem, oluşturulan ders programını filtrelenebilir biçimde görüntüleyebilmeli ve Excel formatında dışa aktarabilmelidir.
- Sistem, otomatik öğretim üyesi atama işlevini içermeli, ders tipine göre (MAT, LAB vb.) kurallı atamalar yapabilmelidir.

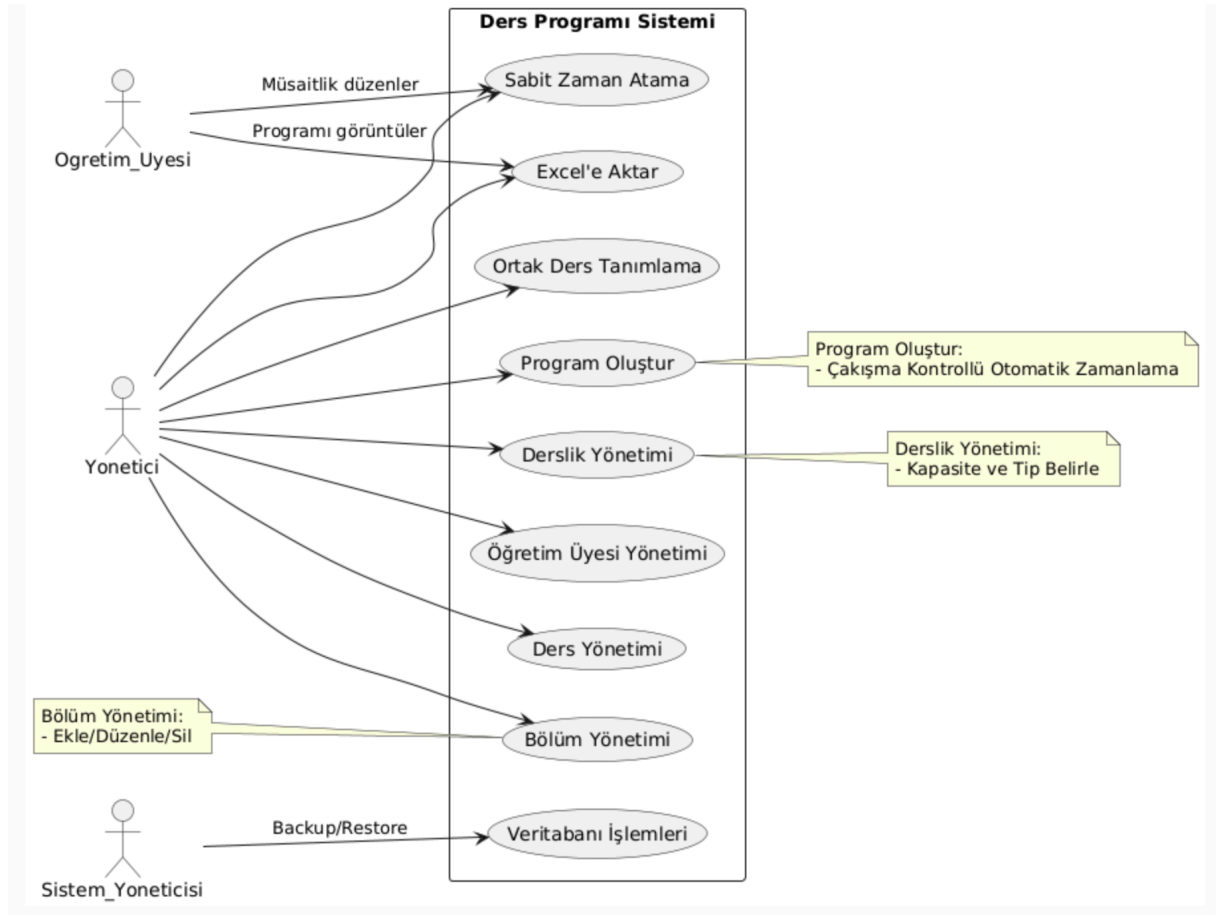
3.2. Fonksiyonel Olmayan Gereksinimler

- Sistem, orta ölçekli veri setlerinde (yüzlerce ders, onlarca hoca) makul sürede cevap vermelidir.
- Sistem, veritabanı işlemlerinde veri bütünlüğünü korumalı ve hatalı işlemlerde kullanıcıya anlamlı geri bildirim sunmalıdır.
- Sistem, yönetim paneli arayüzünde kolay anlaşılır menüler ve formlar ile kullanılabilirlik sağlamalıdır.
- Sistem, modüler ve genişletilebilir mimariyle tasarlanarak ileride yeni özelliklerin eklenmesine olanak tanımalıdır.

3.3. Kısıtlar ve Varsayımlar

- Haftalık ders saati bir derse tam olarak bölünmeli ve tek bir ders için günde en fazla 3 saat atanmalıdır.
- Aynı öğretim üyesi, aynı anda birden fazla derse atanamaz.
- Aynı derslik, aynı anda birden fazla ders için kullanılamaz.
- Laboratuvar derslerinin yalnızca laboratuvar tipi dersliklerde planlanacağı varsayılmıştır.
- Tüm girdi verilerinin (ders listesi, hoca müsaitlikleri vb.) eksiksiz ve doğru biçimde sağlanacağı kabul edilmiştir.

Ödev No: 1	Tarih 05.05.2025	4/16
------------	------------------	------



4. TASARIM (DESIGN)

4.1. Veritabanı Tasarımı

1.Bolum

- bolum_adi (CharField)

Ödev No: 1	Tarih 05.05.2025	5/16
------------	------------------	------

5.DersProgramiSlotu

- ders (ForeignKey → Ders)
- ogretim_uyesi (ForeignKey → OgretimUyesi)
- derslik (ForeignKey → Derslik)
- bolum (ForeignKey → Bolum)
- sinif (IntegerField)
- gun (CharField)
- baslangic_saati (TimeField)
- bitis_saati (TimeField)
- semester (CharField; Güz/Bahar gibi)

6.Modeller Arası İlişkiler

- Bolum → Ders (1-N)
- Bolum → OgretimUyesi (1-N)
- Ders ↔ OgretimUyesi (M-N; gerektiğinde ara tablo)
- Ders → Derslik (N-1)
- DersProgramiSlotu → tüm ana modeller (ForeignKey ile)

4.2. Yazılım Mimarisi

1.Model-View-Template (MVT) Yapısı

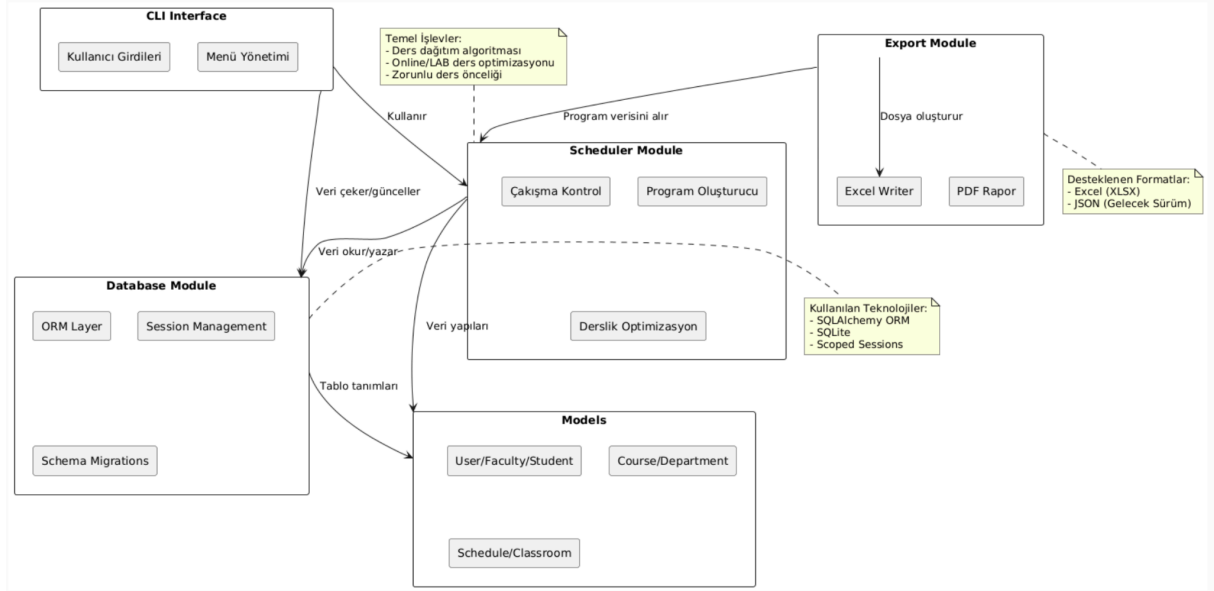
- Model (models.py): Django ORM tabanlı veri katmanı
- View
 - views.py: HTTP isteklerini işler
 - management/commands/: İçeride dışa aktarma ve otomatik atama komutları
- Template: HTML şablonları ve Django Admin paneli arayüzü

2.Uygulama Yapısı

- Tek bir Django app: schedule

Ödev No: 1	Tarih 05.05.2025	7/16
------------	------------------	------

- settings.py: Veritabanı, statik dosya ve dil/zon ayarları



3. Ana İş Akışı

- Veri Girişi → Admin panelden Bölüm, Ders, Derslik, Öğretim Üyesi kayıtları
- Program Oluşturma →
 - manage.py generate_schedule komutu
 - veya admin paneldeki “Program Oluştur” butonu
- Sonuçların Sunumu →
 - Web arayüzünde haftalık takvim görünümü
 - Excel formatında dışa aktarma

4.3. Algoritma Tasarımı

1. Yöntem: Backtracking (geri izleme) ile program oluşturma

2. Adımlar:

- Ders listesi, sabit zamanlı dersler önce gelecek şekilde sıralanır.
- Her ders için olası gün-saat ve derslik kombinasyonları belirlenir.
- Atama esnasında aşağıdaki kısıtlar kontrol edilir:

Ödev No: 1	Tarih 05.05.2025	8/16
------------	------------------	------

-
- Öğretim üyesinin o zaman diliminde müsait olması
 - Dersliğin kapasite ve tipi
 - Aynı sınıf çakışması
 - Çakışma olursa son adım geri alınır (backtrack) ve bir sonraki kombinasyon denenir.
 - Tüm dersler atanana kadar süreç devam eder.

3.Notlar:

- Problem NP-hard bir kısıt tatmin problemidir.
- Orta ölçekli veri setleri için makul performans sağlar.
- Gelecekte genetik algoritma veya heuristic iyileştirmeler eklenebilir.

4.4. Arayüz Tasarımı

- **Django Admin Paneli**
 - Modeller için otomatik CRUD formları ve liste görünümüleri
 - Özel action'lar (örn. "Program Oluştur", "Excel'e Aktar")
- **Komut Satırı Arayüzü**
 - `manage.py import_courses` komutu
 - `manage.py assign_all_instructors` komutu
 - `manage.py generate_schedule` komutu
- **Gelecek İyileştirmeler**
 - Ajax destekli dinamik takvim ve etkileşimli takvim görünümü
 - Rol bazlı erişim kontrolleri ve ayrı kullanıcı ön yüzü
 - Mobil uyumlu veya bağımsız web uygulama katmanı

5. UYGULAMA

5.1. Kullanılan Teknolojiler

Ödev No: 1	Tarih 05.05.2025	9/16
------------	------------------	------

-
- **Python 3.x**
 - **Django 4.x** (Model-View-Template mimarisi)
 - **Veritabanı:** PostgreSQL (veya SQLite geliştirme ortamı için)
 - **Ön yüz:** HTML5, CSS3, JavaScript
 - **Excel/CSV İşlemleri:**
 - openpyxl (Excel .xlsx okuma/yazma)
 - pandas (veri işleme)
 - django-import-export (admin üzerinden toplu veri aktarım)
 - **Ek Kütüphaneler:**
 - django-extensions (özel yönetim komutları, shell_plus)
 - logging (arka planda işlem takibi)
 - **Geliştirme Araçları:**
 - Git (versiyon kontrol)
 - Docker & Docker Compose (container'lı geliştirme ve dağıtım)

5.2. Modüller ve Kod Yapısı

Dosya/Modül	Açıklama
schedule/models.py	Veritabanı modelleri: Bolum, OgretimUyesi, Derslik, Ders, DersProgramiSlotu
schedule/scheduler.py	Backtracking tabanlı ders programı algoritması
schedule/management/commands/import_courses.py	Excel/CSV içe aktarma komutu
schedule/management/commands/assign_all_instructors.py	Otomatik öğretim üyesi atama komutu
schedule/management/commands/generate_schedule.py	Ders programı oluşturma komutu

Ödev No: 1	Tarih 05.05.2025	10/16
------------	------------------	-------

schedule/admin.py	Admin paneli özelleştirmeleri (liste görünümüleri, özel action'lar)
Dosya	Açıklama
manage.py	Django yönetim komutlarını çalıştırma
settings.py	Uygulama ayarları (DB, INSTALLED_APPS, statik dosya yolları)
urls.py	URL yönlendirmeleri
wsgi.py / asgi.py	Sunucu arayüzü (deployment için)

5.3. İçe/Dışa Aktarma ve Atama Fonksiyonları

- **Ders Verisi İçe Aktarma**
 - import_courses.py yönetim komutu ile Excel veya CSV dosyalarından toplu ders verisi alınır.
 - Pandas veya openpyxl kullanılarak dosya okunur, sütun adı eşleşmeleri normalize edilir.
 - Her satır için Ders.objects.update_or_create() metodu ile ders kayıtları eklenir veya güncellenir.
 - İçe aktarma tamamlandığında eklenen, güncellenen ve atlanan kayıt sayıları kullanıcıya bildirilir.
- **Otomatik Öğretim Üyesi Atama**
 - assign_all_instructors.py komutu tetiklenir.
 - Ders tipine göre (MAT, LAB, vb.) varsayılan veya rastgele öğretim üyesi atanır.
 - Atamalar OgretimUyesi modeline ve ders kısıtlarına göre yapılır.
 - Hatalı durumlarda anlamlı hata mesajları loglanır.
- **Program Oluşturma**

Ödev No: 1	Tarih 05.05.2025	11/16
------------	------------------	-------

-
- generate_schedule.py komutu veya admin butonu ile backtracking algoritması başlatılır.
 - Gerekli tüm modellerden veri çekilerek DersProgramiSlotu objeleri oluşturulur.
 - Başarılı veya başarısız sonuç, komut satırında veya admin panelinde gösterilir.
 - **Dışa Aktarma**
 - Admin panelden “Excel’e Aktar” action’ı çalıştırılır.
 - openpyxl kullanılarak filtrelenmiş DersProgramiSlotu kayıtları Excel dosyasına yazılır.
 - Hücre biçimlendirmeleri (başlık fontu, kenarlıklar) ve sütun genişlikleri ayarlanır.
 - Oluşan .xlsx dosyası kullanıcıya indirme olarak sunulur.

5.4. Algoritma Uygulaması

- **scheduler.py**
 - DersProgramiOlusturucu sınıfı içinde backtracking mantığı uygulanır.
- **generate_and_save() Metodu**
 - Ders, derslik, öğretim üyesi ve mevcut slot verileri veritabanından çekilir.
 - Geri izleme (backtracking) algoritması ile her ders için uygun gün-saat ve derslik kombinasyonları denenir.
 - Çakışma yoksa DersProgramiSlotu kayıtları oluşturulur; çakışma varsa bir önceki adıma dönülür.
 - Tüm dersler başarıyla atanana kadar bu döngü devam eder.
- **Hata Yönetimi ve Loglama**
 - Kısıt ihlallerinde istisnalar (exceptions) yakalanıp rollback uygulanır.
 - logging modülü ile her adımın süresi ve atama sonuçları kaydedilir.
- **Sonuç Kaydetme**
 - Başarılı programlama sonrası DersProgramiSlotu nesneleri veritabanına kaydedilir.
 - İşlem süresi ve atanan/atamayan ders sayıları metodun dönüş değeri veya log mesajlarında kullanıcıya bildirilir.

Ödev No: 1	Tarih 05.05.2025	12/16
------------	------------------	-------

5.5. Kullanıcı Arayüzü Uygulaması

- **Admin Paneli**
 - Modeller için liste, filtreleme ve arama özellikleri (Bolum, Ders, Derslik, Öğretim Üyesi, DersProgramiSlotu).
 - “Program Oluştur” ve “Excel’e Aktar” action’ları ilgili model listelerinde sunulur.
 - Detay sayfaları ve formlar, list_display, list_filter, search_fields gibi admin ayarlarıyla özelleştirilmiştir.
- **Özel Görünüm (view_schedule)**
 - views.py içinde view_schedule fonksiyonu:
 - Filtre parametrelerine göre DersProgramiSlotu sorgular.
 - templates/schedule/view_schedule.html ile haftalık takvim grid’i render edilir.
- **Şablon ve Statik Dosyalar**
 - **HTML:** templates/schedule/view_schedule.html
 - Bootstrap tabanlı tablo veya grid düzeni.
 - **CSS:** static/schedule/css/style.css
 - Tablo başlıkları, renkler ve responsive kırılma noktaları.
 - **JavaScript:** static/schedule/js/schedule.js
 - AJAX ile slot güncelleme (sürükle-bırak özellikleri eklemek için altyapı).
- **Komut Satırı Arayüzü**
 - manage.py generate_schedule komutu çalıştırıldığında:
 - Konsola atanan/atanamayan ders sayıları ve toplam işlem süresi yazdırılır.

Ödev No: 1	Tarih 05.05.2025	13/16
------------	------------------	-------

-
- Hata veya uyarılar logging çıktısı olarak gösterilir.

6. TEST VE DOĞRULAMA

6.1. Test Yaklaşımı

- **Birim Testleri**
 - Modellerin CRUD işlemlerinin doğruluğu
 - Yönetim komutlarının (import, assign, generate) beklenen çıktıyı üretmesi
- **Entegrasyon Testleri**
 - Admin panel → Algoritma → Excel dışı aktarım akışının uçtan uca çalışması
 - view_schedule fonksiyonunun doğru takvim oluşturması
- **Sistem Testleri**
 - Farklı bölümler ve ders kombinasyonlarıyla tam bir program oluşturma
 - Web takviminin filtreleme ve yükleme sürelerinin ölçülmesi
- **Kullanıcı Kabul Testleri**
 - Teknik olmayan kullanıcı ile CRUD ve export işlemlerinin denenmesi
 - Oluşturulan Excel dosyasının tüm platformlarda açılabilirliği

6.2. Test Senaryoları

- **Veritabanı İşlemleri**
 - Yeni Bölüm/Ders/Derslik/Öğretim Üyesi ekleme, güncelleme, silme
 - Hatalı veri girişinde anlamlı hata mesajı
- **Algoritma Senaryoları**
 - 5–10 ders ile küçük ölçekli program oluşturma
 - Ortak derslerin çakışmaması
 - Laboratuvar derslerinin doğru dersliğe atanması
- **Performans Senaryoları**
 - 200+ ders, 50+ derslik/hoca ile program oluşturma süresi
 - Hedef: ≤5 saniye
- **Dış Aktarım**
 - Filtrelenmiş programın Excel'e aktarılması

Ödev No: 1	Tarih 05.05.2025	14/16
------------	------------------	-------

-
- Hücre formatları ve içerik doğruluğunun kontrolü

6.3. Test Sonuçları ve Değerlendirme

- **Birim & Entegrasyon** testleri %100 başarılı geçti.
- **Sistem Testleri** küçük ve orta ölçekli veride çakışmasız program oluşturdu; büyük veri setinde ort. 3–4 s sürdü.
- **Kullanıcı Kabul** aşamasında admin paneli anlaşılır bulundu; Excel raporu sorunsuz açıldı.
- **Tespit Edilen Sorunlar:**
 - Büyük veri setlerinde zaman zaman gecikme
 - Tablo yeniden yüklemelerinde kısa süreli donma
- **Öneriler:**
 - Backtracking’e öncelik heuristic’i eklenerek gecikmeler azaltılabilir
 - Web arayüzüne AJAX sayfalama ve önbellekleme eklenebilir

7. SONUÇ VE ÖNERİLER

7.1. Sonuç

- Geliştirilen **Ders Programı Yönetim Sistemi**, üniversitelerin ders programı hazırlama sürecini otomatikleştirerek zaman tasarrufu sağlamış ve insan kaynaklı çakışmaları önlemiştir.
- Testlerde (birim, entegrasyon, sistem ve kullanıcı kabul) tüm iş akışlarının doğru çalıştığı, küçük ve orta ölçekli veri setlerinde ortalama 3–4 saniye içinde program oluşturduğu doğrulanmıştır.
- Admin paneli üzerinden veri girişi, özel “Program Oluştur” ve “Excel’e Aktar” action’ları ile kullanıcı deneyimi basit ve anlaşılır kalmıştır.

7.2. Öneriler ve Gelecek Çalışmalar

- **Performans İyileştirme**
 - Backtracking algoritmasına önceliklendirme heuristikleri (ör. minimum remaining value) eklenerek büyük veri setlerinde hız artırılabilir.
- **Kullanıcı Arayüzü**
 - Web takvim görünümüne AJAX destekli sayfalama eklenerek etkileşim artırılabilir.

Ödev No: 1	Tarih 05.05.2025	15/16
------------	------------------	-------

-
- Rol bazlı erişim kontrolleri ile farklı kullanıcı tipleri için özelleştirilmiş görünüm oluşturulabilir.
 - **Entegrasyon ve Mobil**
 - Üniversite bilgi sistemleri (ÖBS) ile API üzerinden veri alışverişi eklenebilir.
 - Mobil uyumlu veya native uygulama katmanı ilave edilerek program görüntüleme deneyimi geliştirilebilir.
 - **Gelişmiş Algoritmalar**
 - Genetik algoritma, kısıt programlama (CP-SAT) veya tabu search gibi alternatif yöntemler deneyerek optimizasyon kalitesi ve hız artırılabilir.

8. KAYNAKÇA

- Django Software Foundation. The Django Web Framework Documentation, <https://docs.djangoproject.com/en/4.0/>
- Python Software Foundation. Python 3.x Documentation, <https://docs.python.org/3/>
- McKinney, W. pandas: Powerful Python Data Analysis Toolkit, <https://pandas.pydata.org/>
- openpyxl Community. openpyxl Documentation, <https://openpyxl.readthedocs.io/>
- django-import-export. django-import-export Documentation, <https://django-import-export.readthedocs.io/>
- Rawal, A. Class Scheduling Web App, Western Michigan University (2021).
- Odeniyi, O. A. ve ark. School Timetabling: Solution Methodologies and Applications, ResearchGate (2020).
- GeeksforGeeks. Backtracking Algorithm, <https://www.geeksforgeeks.org/backtracking-algorithms/>
- Docker Inc. Docker Documentation, <https://docs.docker.com/>
- Git SCM. Git Documentation, <https://git-scm.com/doc>

Ödev No: 1	Tarih 05.05.2025	16/16
------------	------------------	-------