

**T.C**

**KOCAELİ SAĞLIK VE TEKNOLOJİ ÜNİVERSİTESİ**

**MÜHENDİSLİK VE DOĞA BİLİMLERİ FAKÜLTESİ**

**BİLGİSAYAR/YAZILIM MÜHENDİSLİĞİ**

**PROJE KONUSU: PROGRAM DERS İLİŞKİ MATRİSİ PROJESİ**

**ÖĞRENCİELRİN ADI:**

Ali Can Kaplan  
Erhan Evren  
Emirhan Dursun  
Alperen Akın Işgın  
Kerem Savaş

**ÖĞRENCİLERİN NUMARASI:**

210501006  
230501034  
220502039  
220502034  
220502032

**DERS SORUMLUSU:**

**Dr. Öğr. Üyesi**

**Elif Pınar HACIBEYOĞLU**

**TARİH:25.12.2024**

1. **GİRİŞ**
   1. **Projenin amacı**

Bu projenin amacı, öğrenci performanslarını ders ve program çıktıları ile değerlendirme kriterleri üzerinden analiz ederek başarı oranlarını belirlemektir. Proje kapsamında, program çıktıları ve ders çıktıları arasındaki ilişkiyi gösteren matrisler oluşturulacak, öğrenci notları değerlendirme kriterlerine göre ağırlıklandırılarak analiz edilecek ve her bir öğrenci için başarı oranlarını içeren tablolar hazırlanacaktır. Bu analizler, ders ve program çıktıları ile değerlendirme kriterleri arasındaki ilişkileri görselleştirme ve öğrenci başarılarının daha iyi anlaşılmasını sağlamayı hedeflemektedir.

1. **GEREKSİNİM ANALİZİ**
   1. **Arayüz gereksinimleri**

**Kullanıcı Arayüzü Gereksinimleri:**

* Kullanıcı dostu bir arayüz ile ders, not, puan türü ve program çıktılarının yüklenmesi için farklı sekmeler tasarlanmıştır.
* Öğrenciler ve dersler arasındaki ilişkilerin görsel olarak sunulması için bir sekme yer alır.
* Her sekme içinde uygun giriş alanları, düğmeler ve seçim listeleri yer almaktadır.
* Excel dosyaları ile giriş ve çıkış verilerinin yüklenmesini sağlayan dosya seçici bileşenler bulunmaktadır.

**Donanım Arayüzü Gereksinimleri:**

* Kod, masaüstü kullanıcıları için bir Tkinter GUI arayüzü kullanmaktadır. Donanım olarak minimum bir fare ve klavye ile desteklenir.
* SQLite veritabanını kullanması nedeniyle, sabit diskte yer ve dosya erişim izinleri gereklidir.
  1. **Fonksiyonel gereksinimler**
* Kullanıcıların ders, not ve program çıktı bilgilerini manuel olarak yüklemesi.
* Program ve ders çıktıları ile ilgili tabloların otomatik olarak oluşturulması.
* Veritabanına ders, öğrenci ve not bilgilerinin kaydedilmesi.
* Ders çıktılarının ve program yeterliliklerinin bir web sitesinden alınarak Excel formatına aktarılması.
* Kullanıcıların belirli kriterlere göre verileri filtreleyip çıktı oluşturması.
  1. **Use-Case diyagramı**

çizgi, diyagram, ekran görüntüsü, daire içeren bir resim

Açıklama otomatik olarak oluşturuldu

* 1. **Modul diyagramı**
* Aşağıdaki görsel, uygulamanın modül diyagramıdır.

**metin, çizgi, diyagram, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

1. **TASARIM**
   1. **Mimari tasarım**

Uygulama, istemci tarafında Tkinter kullanılarak oluşturulmuş bir GUI içerir.  
Veritabanı işlemleri SQLite ile yönetilmektedir.  
Modüller:

* Ders Yönetimi: Derslerin eklenmesi ve listelenmesi.
* Program Çıktıları: Program çıktılarının Excel'e aktarılması.
* Not Yönetimi: Öğrenci notlarının kaydedilmesi ve görüntülenmesi.

Crawler: Web'den verilerin çekilmesi

* 1. **Kullanılacak teknolojiler**

Yazılım Dili: Python (Tkinter ve SQLite modülleri kullanılarak).

Harici Kütüphaneler:

* requests (HTTP istekleri için).
* BeautifulSoup (HTML parse etmek için).
* pandas (Veri işlemleri ve Excel dosya oluşturma için).
* openpyxl (Excel dosya yazma).
  1. **Veri tabanı tasarımı**

**er diagramı
**

**Tablolar:**

* Dersler: Ders adı ve kod bilgilerini tutar.
* Ogrenciler: Öğrenci numaralarını içerir.
* DersCikti: Ders çıktıları bilgilerini içerir ve ilgili dersle ilişkilidir.
* ProgramCikti: Program çıktıları bilgilerini içerir ve ilgili dersle ilişkilidir.
* Notlar: Öğrencilerin derslere ait not bilgilerini tutar. Notlar, öğrenci ve dersle ilişkilidir.
* PuanTuru: Her ders için puan türü ve yüzdeleri bilgilerinden oluşur.

**İlişkiler:**

* Dersler ve Ogrenciler arasında, Notlar tablosu aracılığıyla ilişki vardır.
* Dersler, hem DersCikti hem de ProgramCikti tabloları ile ilişkilidir.
* PuanTuru, ilgili olduğu dersle ilişkilidir**.**
  1. **Kullanıcı arayüzü tasarımı**

Uygulama, kullanıcı dostu bir arayüz sunmak amacıyla **Tkinter** kütüphanesi kullanılarak geliştirilmiştir. Arayüz, birden fazla sekmeden oluşan bir tasarım içerir ve her sekme farklı bir işlevi yerine getirmek üzere tasarlanmıştır.

**Veri Tabanına Yükleme Sekmesi:** Ders, öğrenci, not ve program çıktıları gibi bilgilerin veri tabanına yüklenmesi için gerekli alanlar içerir.

**Çıktı Oluşturma Sekmesi:** Ders ve program çıktılarının oluşturulması, çıktıların Excel dosyalarına aktarılması ve görselleştirilmesi için kullanılır.

**Uygulama Kullanımına Yönelik Açıklama Dokümantasyonu**

1. **Ders Ekleme ve Öğrenci Listesi sekmesi**

* Aşağıdaki görsel, uygulamanın "Ders Ekleme ve Öğrenci Listesi"ni göstermektedir. Bu ekran, verilerin yüklenmesi için kullanılmaktadır.

**metin, ekran görüntüsü, sayı, numara, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu**

Bu ekran, "Ders Planı" adlı bir uygulamanın ana penceresi olup, iki temel işlev sunmaktadır:

1. **Ders Yükleme**:
   * Kullanıcı, "Ders Adı" ve "Ders Kodu" alanlarına gerekli bilgileri girerek bir dersi sisteme yükleyebilir. "Dersi Yükle" düğmesi bu işlevi gerçekleştirir.
2. **Öğrencileri Yükleme**:
   * Kullanıcı, bir dosya seçerek öğrenci listesini yükleyebilir. "Dosya Seç" alanı, öğrenci bilgilerini içeren dosyanın seçilmesi içindir. Ardından "Öğrencileri Yükle" düğmesiyle işlem tamamlanır.
3. **Dersler Listesi**:
   * Bu bölümde yüklenen dersler bir liste olarak görüntülenmektedir. Şu an liste boş görünüyor.

Bu ekran, kullanıcıların dersleri ve öğrenci bilgilerini kolayca sisteme eklemesine olanak tanır.

**2. Program ve Ders Çıktıları sekmesi**

Aşağıdaki görsel, " Program ve Ders Çıktıları "nı göstermektedir. Bu ekran ders çıktılarını ve program çıktılarını yüklemek veya giriş yapmak için kullanılır

**metin, ekran görüntüsü, sayı, numara, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

Bu ekran, "Program ve Ders Çıktıları" sekmesine ait bir arayüzdür ve aşağıdaki işlevleri sunar:

1. **Ders Çıktıları Yükleme**:
   * Kullanıcı, "Ders Seç" menüsünden bir ders seçerek ve ilgili dosyayı yükleyerek ders çıktılarının sisteme eklenmesini sağlar. "Ders Çıktılarını Yükle" düğmesi bu işlemi tamamlar.
2. **Program Çıktıları Yükleme**:
   * Kullanıcı, "Ders Seç" menüsünden bir ders seçip ilgili dosyayı yükleyerek program çıktılarının yüklenmesini sağlar. "Program Çıktılarını Yükle" düğmesiyle işlem gerçekleşir.
3. **Ders Çıktıları Girişi**:
   * Kullanıcı, manuel olarak "Sayı" ve "Öğrenme Çıktısı" bilgilerini girerek bir ders çıktısı ekleyebilir. "Yeni Satır Ekle" butonu yeni bir satır oluştururken, "Verileri İşle" butonu ile bu bilgiler sisteme kaydedilir. "Sil" butonu ile gereksiz veriler kaldırılabilir.
4. **Program Çıktıları Girişi**:
   * Benzer şekilde, kullanıcı "Sıra" ve "Program Çıktısı" alanlarına veri girerek bir program çıktısı ekleyebilir. "Yeni Satır Ekle" ve "Verileri İşle" butonlarıyla işlem yapılır, "Sil" butonu ile gereksiz satırlar kaldırılır.

**3. Program ve Ders Çıktıları sekmesi**

* Aşağıdaki görsel, uygulamanın "Tabloları Ekleme Ekranı"nı göstermektedir. Bu ekran, tabloların yüklenmesi için kullanılmaktadır.

**metin, yazı tipi, sayı, numara, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

 **Ders Seçimi:**

* Kullanıcı, üstteki açılır menüden ("Ders Seç") bir ders seçebilir. Bu seçim, yükleme ve tablo işlemleri için temel bağlamı oluşturur.

 **Tablo1 Yükleme:**

* Kullanıcı, "Dosya Seç" alanını kullanarak bir dosya seçebilir ve ardından "Tablo1 Yükle" düğmesi ile bu dosyayı yükleyebilir.

 **Tablo2 Yükleme:**

* İkinci bir dosya için aynı işlem yapılabilir. Dosya seçildikten sonra, "Tablo2 Yükle" düğmesi ile yükleme tamamlanır.

 **Tablo Oluşturma:**

* Kullanıcı, ekranın alt kısmında bulunan üç düğme ("Tablo3 Oluştur", "Tablo4 Oluştur", "Tablo5 Oluştur") aracılığıyla yeni tablolar oluşturabilir. Bu düğmeler, tanımlı kriterlere veya yüklenen verilere göre tablo oluşturmayı tetikler.

1. **UYGULAMA**
   1. **Kodlanan bileşenlerin açıklamaları**

**Ekle Butonu**

**Fonksiyon**: ekle\_butonu\_action

**Açıklama**: Bu buton, ders ekleme işlemini başlatan bir fonksiyonu tetikler.

**Kod**:  
ekle\_butonu = ttk.Button(ders\_yukleme\_frame, text="Dersi Yükle", command=ekle\_butonu\_action)  
ekle\_butonu.grid(row=0, column=4, padx=10, pady=5)

**Öğrenme Çıktılarını Al Butonu**

**Fonksiyon**: ogrenme\_ciktilari\_al

**Açıklama**: Bu buton, öğrenme çıktılarının alınmasını başlatan fonksiyonu tetikler.

**Kod**:  
ogrenme\_ciktilari\_al\_butonu = ttk.Button(crawler\_frame, text="Öğrenme Çıktılarını Al", command=ogrenme\_ciktilari\_al)  
ogrenme\_ciktilari\_al\_butonu.grid(row=0, column=2, padx=10, pady=5)

**Ders Çıktılarını Yükle Butonu**

**Fonksiyon**: ders\_cikti\_ekle\_buton\_action

**Açıklama**: Bu buton, ders çıktılarını yükleyen bir fonksiyonu çağırır.

**Kod**:  
ders\_cikti\_ekle\_butonu = ttk.Button(ders\_ciktilari\_frame, text="Ders Çıktılarını Yükle", command=ders\_cikti\_ekle\_buton\_action)  
ders\_cikti\_ekle\_butonu.grid(row=0, column=5, padx=10, pady=5)

**Program Çıktılarını Yükle Butonu**

**Fonksiyon**: program\_cikti\_ekle\_buton\_action

**Açıklama**: Bu buton, program çıktılarını yükler.

**Kod**:  
program\_cikti\_ekle\_butonu = ttk.Button(program\_ciktilari\_yukleme\_frame, text="Program Çıktılarını Yükle", command=program\_cikti\_ekle\_buton\_action)  
program\_cikti\_ekle\_butonu.grid(row=0, column=5, padx=10, pady=5)

**Puan Türü Yükle Butonu**

**Fonksiyon**: puan\_turu\_ekle\_buton\_action

**Açıklama**: Bu buton, puan türlerini yüklemeye yarayan fonksiyonu tetikler.

**Kod**:  
puan\_turu\_ekle\_butonu = ttk.Button(puan\_turu\_yukle\_frame, text=" Yükle", command=puan\_turu\_ekle\_buton\_action)  
puan\_turu\_ekle\_butonu.grid(row=0, column=6, padx=10, pady=5)

**Notları Yükle Butonu**

**Fonksiyon**: notlari\_yukle\_buton\_action

**Açıklama**: Bu buton, notları yükler.

**Kod**:  
notlari\_yukle\_butonu = ttk.Button(notlar\_frame, text="Notları Yükle", command=notlari\_yukle\_buton\_action)  
notlari\_yukle\_butonu.grid(row=0, column=5, padx=10, pady=5)

**Tablo1 Yükle Butonu**

**Fonksiyon**: tablo1\_yukle\_buton\_action

**Açıklama**: Bu buton, Tablo1'i yükler.

**Kod**:  
tablo1\_yukle\_butonu = ttk.Button(tablo1\_frame, text="Tablo1 Yükle", command=tablo1\_yukle\_buton\_action)  
tablo1\_yukle\_butonu.grid(row=0, column=5, padx=10, pady=5)

**Tablo2 Yükle Butonu**

**Fonksiyon**: tablo2\_yukle\_buton\_action

**Açıklama**: Bu buton, Tablo2'yi yükler.

**Kod**:  
tablo2\_yukle\_butonu = ttk.Button(tablo2\_frame, text="Tablo2 Yükle", command=tablo2\_yukle\_buton\_action)  
tablo2\_yukle\_butonu.grid(row=0, column=5, padx=10, pady=5)

**Tablo3 Oluştur Butonu**

**Fonksiyon**: tablo3\_olustur\_butonu\_action

**Açıklama**: Bu buton, Tablo3'ü oluşturur.

**Kod**:  
tablo3\_olustur\_butonu = ttk.Button(Ciktilari\_olustur\_frame, text="Tablo3 Oluştur", command=tablo3\_olustur\_butonu\_action)  
tablo3\_olustur\_butonu.grid(row=0, column=1, padx=10, pady=5)

**Tablo4 Oluştur Butonu**

**Fonksiyon**: tablo4\_olustur\_butonu\_action

**Açıklama**: Bu buton, Tablo4'ü oluşturur.

**Kod**:  
tablo4\_olustur\_butonu = ttk.Button(Ciktilari\_olustur\_frame, text="Tablo4 Oluştur", command=tablo4\_olustur\_butonu\_action)  
tablo4\_olustur\_butonu.grid(row=0, column=2, padx=10, pady=5)

**Tablo5 Oluştur Butonu**

**Fonksiyon**: tablo5\_olustur\_butonu\_action

**Açıklama**: Bu buton, Tablo5'i oluşturur.

**Kod**:  
tablo5\_olustur\_butonu = ttk.Button(Ciktilari\_olustur\_frame, text="Tablo5 Oluştur", command=tablo5\_olustur\_butonu\_action)  
tablo5\_olustur\_butonu.grid(row=0, column=3, padx=10, pady=5)

* 1. **Görev dağılımı**

Web'den veri çekme ve işleme projesi, Ali Can Kaplan, Erhan Evren, Emirhan Dursun, Alperen Akın Işgın ve Kerem Savaş'ın ortak çalışmalarıyla tamamlanmıştır. Projenin tüm aşamaları, ekip üyelerinin iş birliği ve katkıları sayesinde başarılı bir şekilde gerçekleştirilmiştir.

* 1. **Karşılaşılan zorluklar ve çözüm yöntemleri**
     1. **Veritabanı Yapısının Tasarımı**
* **Zorluk:** Dersler, program çıktıları, öğrenciler ve notlar gibi farklı verilerin ilişkili bir şekilde saklanması gerektiğinden, tablo yapılarının belirlenmesi ve ilişkilerin doğru kurulması karmaşık hale geldi.
* **Çözüm:** SQLite kullanılarak her veri türü için ayrı tablolar oluşturuldu ve FOREIGN KEY ile tablolar arası ilişkiler sağlandı. Ayrıca, test aşamasında yanlış girişleri engellemek için veri doğrulama eklendi.
  + 1. **Tablo Oluşturmalarda İlişkili Verilerle Çalışma**
* **Zorluk:** Veritabanında dersler, öğrenciler ve notlar gibi tabloların birbirleriyle ilişkilendirilmesi gerektiğinden, doğru sıralamada ve referanslarda çalışmak karmaşık hale geldi.
* **Çözüm:** Tabloların öncelik sırasına göre oluşturulması sağlandı ve ilişkiler için ON DELETE CASCADE gibi kurallar kullanılarak veri tutarlılığı korundu. SQL sorguları düzenli olarak test edilerek sorunlar giderildi.
  + 1. **Web’den Veri Çekme (Crawler)**
* **Zorluk:** Web sayfasının HTML yapısının değişken olması, doğru tabloların bulunup işlenmesini zorlaştırdı.
* **Çözüm:** BeautifulSoup ile HTML öğelerinin seçimi esnek hale getirildi ve tablolar belirlenemediğinde kullanıcıya uyarı mesajları gösterildi.
  + 1. **Hata Yönetimi**
* **Zorluk:** Kullanıcı kaynaklı yanlış girişler veya bağlantı sorunları gibi durumlarda uygulamanın stabil çalışmasını sağlamak karmaşıktı.
* **Çözüm:** Tüm kritik işlevlerde try-except blokları kullanılarak uygulamanın çökmesi engellendi ve hata durumunda bilgilendirici mesajlar eklendi.
  1. **Proje isterlerine göre eksik yönler**

Web'den veri çekme ve işleme projesi, Ali Can Kaplan, Erhan Evren, Emirhan Dursun, Alperen Akın Işgın ve Kerem Savaş'ın ortak çalışmalarıyla tamamlanmıştır. Projenin tüm gereksinimleri karşılanmış ve eksik bir kısım kalmamıştır. Ekip üyelerinin iş birliği ve katkıları sayesinde projenin tüm aşamaları başarılı bir şekilde gerçekleştirilmiştir.

1. **TEST VE DOĞRULAMA**
   1. **Yazılımın test süreci**

* **Test Uygulaması Geliştirme**
* Yazılımın test süreci, tüm fonksiyonların tek tek çalıştırılmasını ve herhangi bir hata durumunda detaylı geri bildirim alınmasını sağlayacak bir test uygulamasının geliştirilmesiyle başlamıştır. Bu test uygulaması şu özelliklere sahiptir:
* **Tekrar Edilebilirlik**: Test uygulaması tekrar tekrar çalıştırılarak, yazılımda yapılan değişikliklerden sonra yeniden test edilmesini sağlar.
* **Her Fonksiyonun Test Edilmesi**: Yazılımın tüm bileşenlerini kapsamaktadır.
* **Hata Yakalama ve Bildirme**: Test sırasında hata oluşması durumunda, hata mesajları açıkça bildirilmektedir.
* **Başarı Mesajları**: Her fonksiyonun doğru şekilde çalıştığını belirten çıktılar üretilmektedir.
* **Test Edilen Bileşenler**
  + Aşağıdaki fonksiyonlar test edilmiştir:
  + dersleri\_listeye\_al
  + ogrenme\_ciktilari\_al
  + program\_cikti\_ekle\_buton\_action
  + notlari\_yukle\_buton\_action
  + tablo1\_yukle\_buton\_action
  + tablo1\_matrisi\_goruntule
  + ders\_cikti\_ekle\_buton\_action
  + puan\_turu\_ekle\_buton\_action
  + tablo2\_yukle\_buton\_action
  + tablo2\_matrisi\_goruntule
  + ekle\_butonu\_action
  + tablo3\_olustur\_butonu\_action
  + tablo4\_olustur\_butonu\_action
  + tablo5\_olustur\_butonu\_action
  + Bu fonksiyonlar yazılımın temel işlevlerini temsil etmektedir ve tamamı test kapsamına dahil edilmiştir.
  1. **Yazılımın doğrulanması**

**Test Sonuçları**

Aşağıda, test sonucunda her fonksiyon için elde edilen doğrulama sonuçları yer almaktadır:

**Başarıyla Çalışan Fonksiyonlar**

dersleri\_listeye\_al: Doğru şekilde çalıştı.

ogrenme\_ciktilari\_al: Doğru şekilde çalıştı.

program\_cikti\_ekle\_buton\_action: Doğru şekilde çalıştı.

notlari\_yukle\_buton\_action: Doğru şekilde çalıştı.

tablo1\_yukle\_buton\_action: Doğru şekilde çalıştı.

tablo1\_matrisi\_goruntule: Doğru şekilde çalıştı.

ders\_cikti\_ekle\_buton\_action: Doğru şekilde çalıştı.

puan\_turu\_ekle\_buton\_action: Doğru şekilde çalıştı.

tablo2\_yukle\_buton\_action: Doğru şekilde çalıştı.

tablo2\_matrisi\_goruntule: Doğru şekilde çalıştı.

ekle\_butonu\_action: Doğru şekilde çalıştı.

tablo3\_olustur\_butonu\_action: Doğru şekilde çalıştı.

tablo4\_olustur\_butonu\_action: Doğru şekilde çalıştı.

tablo5\_olustur\_butonu\_action: Doğru şekilde çalıştı.

**Eksik veya Hatalı Çalışan Fonksiyonlar**

Test uygulaması, yukarıdaki fonksiyonların tamamını hatasız şekilde çalıştırmıştır. Bu durumda herhangi bir eksik veya hatalı çalışan bileşen bulunmamaktadır.