amblem, logo, ticari marka, simge, sembol içeren bir resim

Açıklama otomatik olarak oluşturuldu

# **T.C**

# **KOCAELİ SAĞLIK VE TEKNOLOJİ ÜNİVERSİTESİ**

# **MÜHENDİSLİK VE DOĞA BİLİMLERİ FAKÜLTESİ**

# **BİLGİSAYAR/YAZILIM MÜHENDİSLİĞİ**

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

**ÖĞRENCİ ADI: Umutcan Oğurlu**

**ÖĞRENCİ NUMARASI: 230501031**

**ÖĞRENCİ ADI: Berkay Ömer Aydemir**

**ÖĞRENCİ NUMARASI: 230501032**

**ÖĞRENCİ ADI: AHMET BERAT PALA**

**ÖĞRENCİ NUMARASI: 220501015**

**1. Giriş**

**1.1. Proje Amacı**

Bu projenin amacı, bir dersin çıktıları ile program çıktıları arasındaki ilişkiyi

değerlendirerek, öğrencilerin başarı oranlarını hesaplayan bir sistem geliştirmektir. Bu

sistem, kullanıcıdan alınan girdiler ve matematiksel hesaplamalarla ders çıktılarının ve

program çıktılarının başarı oranlarını belirler. Ayrıca, ilgili veriler bir Excel dosyasına

otomatik olarak aktarılır. Proje kapsamında ayrıca bir web kazıyıcı (crawler)

geliştirilerek, çevrimiçi ders verileri otomatik olarak toplanır ve sistemle entegre edilir.

**1.2. Projeden Beklenenler**

-Kullanıcıdan alınan girdilerle ders çıktılarının başarı oranlarını hesaplayabilen

bir sistem.

- Ders çıktıları ve program çıktıları arasındaki ilişkileri görsel olarak sunan bir

ilişki matrisi.

- Öğrencilerin ders başarısını değerlendiren ağırlıklı bir hesaplama.

- Hesaplanan verilerin Excel formatında otomatik olarak raporlanması.

- Ders bilgilerini toplamak için bir web kazıyıcı aracılığıyla çevrimiçi

platformlardan veri çekilmesi.

**2. Gereksinim Analizi**

**2.1. Kullanıcı Gereksinimleri**

- Kullanıcı, program çıktıları ile ders çıktıları arasındaki ilişkiyi bir matriste

belirtmelidir.

- Kullanıcı, dersin değerlendirme kriterlerine (Ödev1, Ödev2, Quiz, Vize, Final) ait

ağırlıkları girmelidir.

- Sistem, öğrenci bazında verilen notlara göre başarı oranlarını hesaplamalıdır.

- Kullanıcı, verilerin Excel formatında çıktısını alabilmelidir.

- Sistemin bir web kazıyıcı ile ders bilgilerini çevrimiçi platformlardan çekmesi

gerekmektedir.

## **2.2 Use-Case Diyagramı**

Use-case diyagramı temel akışları özetler:  
1. Kullanıcı, öğrenci notları ve değerlendirme kriterlerini içeren Excel dosyasını sağlar.  
2. Sistem, ders çıktıları ve değerlendirme kriterleri arasındaki ilişkiyi analiz eder.  
3. Öğrenci bazlı ve genel başarı oranları hesaplanır.  
4. Program çıktıları ve başarı oranları ile ilgili tablolar oluşturulur ve kaydedilir.

metin, diyagram, çizgi, çizim içeren bir resim

Açıklama otomatik olarak oluşturuldu

**2.2. Fonksiyonel Gereksinimler**

- İlişki matrisi, kullanıcı tarafından tanımlanabilir ve gösterilebilir.

- Değerlendirme kriterlerinin ağırlıkları girilebilir ve bu ağırlıklara göre ders

çıktılarının değerlendirilmesi yapılabilir.

- Öğrenciler için notlar girildiğinde, başarı oranları otomatik olarak hesaplanır.

- Hesaplanan veriler, sistem tarafından Excel formatında kaydedilebilir.

- Web kazıyıcı, ders bilgilerini internetten çekebilmeli ve kullanıcının belirlediği

formata uygun şekilde kaydedebilmelidir.

**2.3. Teknik Gereksinimler**

- Python programlama dili kullanılacaktır.

- Pandas kütüphanesi, veri işleme ve hesaplama için kullanılacaktır.

- Openpyxl, verilerin Excel dosyasına aktarılması için kullanılacaktır.

- BeautifulSoup ve Requests kütüphaneleri, web kazıyıcı geliştirilmesi için

kullanılacaktır.

 Kullanıcı arayüzü için basit bir komut satırı arayüzü tasarlanabilir.

**3. Tasarım**

**3.1. Sistem Tasarımı**

Proje, aşağıdaki ana bileşenlerden oluşmaktadır:

1. Kullanıcı Girdileri ve Hesaplama Modülü: Kullanıcıdan alınan verilerle ilişki

matrisi ve değerlendirme kriterleri hesaplanır.

2. Başarı Oranı Hesaplama: Öğrenci bazında başarı oranları, giriş verilerine göre

hesaplanır ve program çıktıları ile ilişkilendirilir.

3. Veri Aktarma Modülü: Hesaplanan veriler, Excel formatında sistem tarafından

dışa aktarılır.

4. Web Kazıyıcı Modülü: Ders bilgilerini çevrimiçi platformlardan otomatik olarak

çeker.

**3.2. Veri Yapıları**

- İlişki Matrisi: Program çıktıları ve ders çıktıları arasındaki ilişkiyi 0 ile 1 arasında

değerlerle tanımlar.

- Değerlendirme Kriterleri: Ödev1, Ödev2, Quiz, Vize, Final gibi kriterlerin

ağırlıkları bir veri yapısında saklanır.

- Başarı Oranı Hesaplama: Her öğrencinin başarı oranı, verilen notlar ve ağırlıklı

değerlendirme kriterlerine göre hesaplanır.

**3.3. Web Kazıyıcı Tasarımı**

Web kazıyıcı, hedef URL’den ders bilgilerini çeker ve bu veriyi kullanıcı tarafından

belirtilen formatta kaydeder. Kazıyıcı, HTML sayfa yapısını analiz eder ve istenilen

verileri belirler.

**4. Uygulama**

**4.1. Geliştirme Süreci**

1. Veri Girişi: Kullanıcıdan program çıktıları ile ders çıktıları arasındaki ilişki ve

değerlendirme kriterlerinin ağırlıkları alınır.

2. Hesaplama Modülü: Değerlendirme kriterlerine göre ders çıktıları ağırlıklı olarak

hesaplanır ve öğrenci başarı oranları belirlenir.

3. Excel Dosyasına Aktarım: Hesaplanan veriler Openpyxl kütüphanesiyle bir

Excel dosyasına aktarılır.

4. Web Kazıyıcı Entegrasyonu: Web kazıyıcı, ders bilgilerini çevrimiçi

platformlardan çeker ve veriyi uygun formatta kaydeder.

**4.2. Kullanılan Teknolojiler**

- Python: Temel programlama dili olarak kullanılmıştır.

- Pandas: Verilerin işlenmesi ve hesaplanması için kullanılmıştır.

- Openpyxl: Verilerin Excel dosyasına aktarılması için kullanılmıştır.

- BeautifulSoup ve Requests: Web kazıyıcıyı geliştirmek için kullanılmıştır.

**5. Test ve Doğrulama**

**5.1. Fonksiyonel Testler**

- İlişki Matrisi Testi: Kullanıcı tarafından girilen ilişki matrisinin doğru şekilde

görüntülenip görüntülenmediği kontrol edilir.

- Başarı Oranı Hesaplama Testi: Öğrencilerin notlarına göre başarı oranlarının

doğru hesaplanıp hesaplanmadığı test edilir.

- Excel Aktarım Testi: Hesaplanan verilerin doğru bir şekilde Excel dosyasına

aktarılması sağlanır.

- Web Kazıyıcı Testi: Web kazıyıcının doğru veriyi doğru formatta çekip

çekmediği test edilir.

**5.2. Performans Testi**

- Veri Hızı: Web kazıyıcının, ders verilerini hızlı bir şekilde çekip kaydetmesi

sağlanır.

- Hesaplama Süresi: Öğrenci başarı oranlarının hesaplanma süresi optimize

edilir.

**5.3. Hata Ayıklama**

- Programın her aşamasında, kullanıcı girdileri ve hesaplamalar üzerinden

yapılan testlerle hatalar ayıklanır ve sistemin doğru çalışması sağlanır.

**6. Sonuçlar**

Bu proje, bir dersin çıktılarının program çıktıları ile ilişkisini detaylı bir şekilde analiz

ederek her öğrenci için başarı oranlarını hesaplamaktadır. Kullanıcı, girdileriyle bu

başarı oranlarını görselleştirebilir ve verileri Excel dosyasına aktarabilir. Ek olarak,

web kazıyıcı sayesinde çevrimiçi ders bilgileri otomatik olarak toplanabilir ve sistemle

entegre edilebilir.

**7.KODLAR VE AÇIKLAMAR**

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

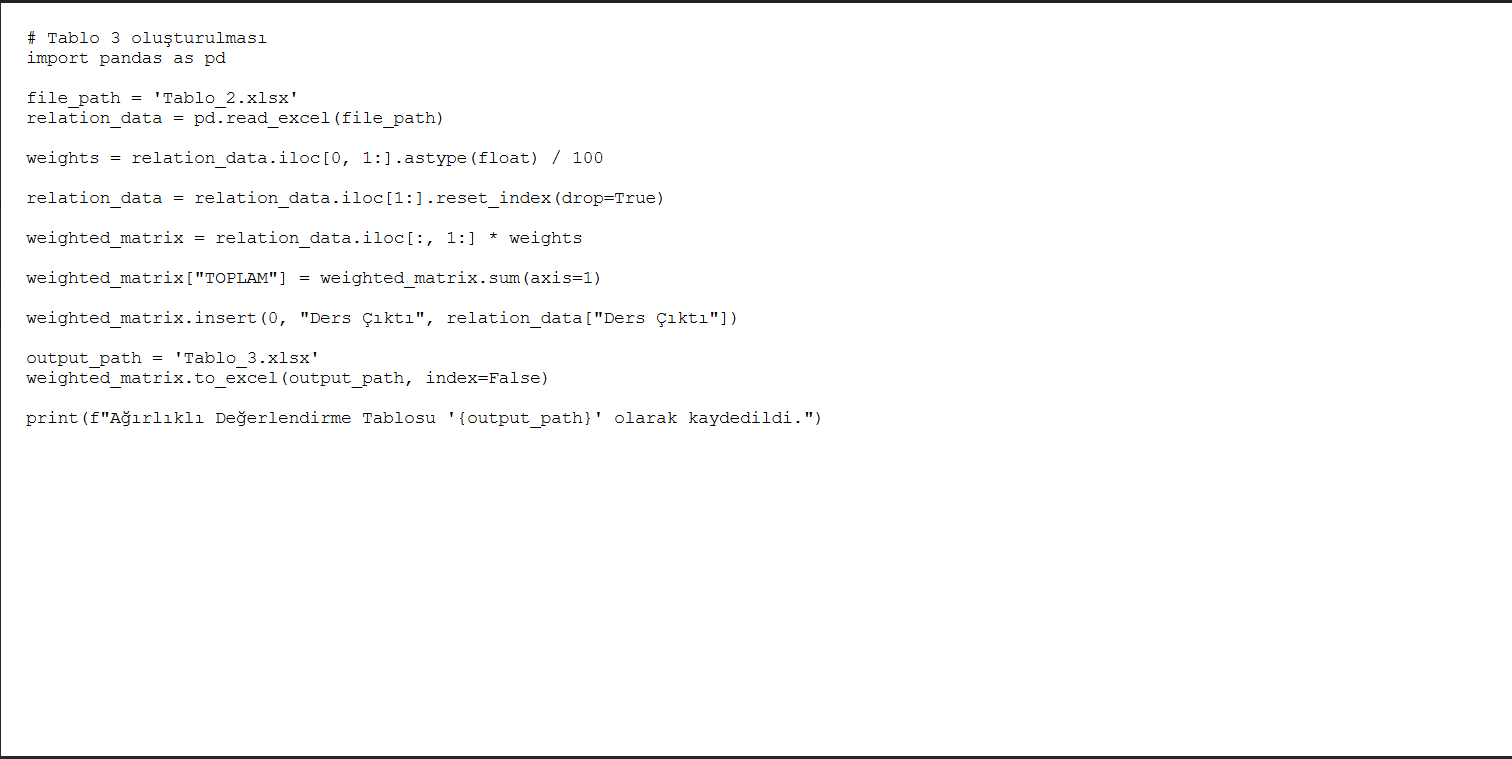
Açıklama otomatik olarak oluşturuldu**

**TABLO1:Bu kod, rastgele oluşturulan bir ilişki matrisiyle program çıktıları ve ders çıktıları arasındaki ilişkileri tabloya dökerek bir Excel dosyasına kaydeder.**

**metin, ekran görüntüsü, yazı tipi, mektup, harf içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**TABLO2:** **Bu kod, ders çıktıları ile değerlendirme kriterleri arasındaki ilişkileri rastgele bir matriste oluşturarak, ağırlıklarla birlikte birleştirip Excel dosyasına kaydeder.**

****

**TABLO3:** **Bu kod, Tablo 2'deki ilişki verilerini okur, ağırlıkları uygular ve her bir ders çıktısı için toplam ağırlıklı değerlendirmeyi hesaplayarak Tablo 3'ü Excel dosyasına kaydeder.**

****

**TABLO4: Bu kod, öğrencilerin notlarını ve ağırlıklı değerlendirme verilerini birleştirerek her bir öğrenci için ayrı ayrı başarı yüzdelerini hesaplar ve sonuçları Excel dosyalarına kaydeder.**

****

**TABLO5: Bu kod, Tablo 1 ve bir öğrenciye ait Tablo 4 verilerini birleştirerek program çıktıları için başarı oranlarını hesaplar ve sonuçları Tablo 5 olarak bir Excel dosyasına kaydeder.**

**8. Kaynakça**

 Python Documentation: https://docs.python.org

 Pandas Documentation: https://pandas.pydata.org

 Openpyxl Documentation: https://openpyxl.readthedocs.io

 BeautifulSoup Documentation:

https://www.crummy.com/software/BeautifulSoup/bs4/doc/

 Requests Documentation: <https://docs.python-requests.org/en/latest/>

**9.GİTHUB:**

Github.com/umutcanogurlu

Github.com/loren01

Github.com/aberatp