**KOCaELİ Sağlık ve teknoloji üniversitesi**

**Yazılım Laboratuvar Ders Ödevi**

**Konu: Makale Yönetim Sistemi**

**Hazırlayanlar: Sude DEMİRTAŞ / Selime Selin CAN**

**Bölüm: Bilgisayar Mühendisliği**

**Okul No:210501012 /210501005**

**20 KASIM 2023**

İçindekiler

[1 AMAÇ 3](#_Toc152426851)

[2 Proje Detayı 4](#_Toc152426852)

[3 Puan Listesi 5](#_Toc152426853)

[4 Program Dili 6](#_Toc152426854)

[5 Projeye Başlamadan Önce Neler Yaptık 7](#_Toc152426855)

[6 Proje Sayesinde Yeni Öğrendiğimiz Şeyler 8](#_Toc152426856)

[7 Projede Zorlandığımız Kısımlar 9](#_Toc152426857)

[8 Sonuç 10](#_Toc152426858)

[9 ER Diyagramı 11](#_Toc152426859)

[10 Kaynakça 13](#_Toc152426860)

# AMAÇ

Bu projenin amacı Kocaeli Sağlık ve Teknoloji Üniversitesi Dergisi bir araştırma dergisidir. Derginin kendisine gönderilen, editör değerlendirmesinde olan ve kabul edilen makaleleri takip edebilmesi için bir veri tabanı sistemine ihtiyacı vardır. Bu projedeki görevimiz bu dergi için bir Makale Yönetim Sistemi uygulaması geliştirmemiz.

# Proje Detayı

Sistemi kullanan bütün kullanıcıların sisteme giriş yapmaları gerekmektedir. Uygulama, her başlatıldığında “Üye Ol” ve “Giriş Yap” seçeneklerini kullanıcıya sunmalıdır. “Üye Ol” seçeneği ile kullanıcıdan aşağıdaki bilgiler alınmalıdır:

* Kullanıcı adı
* Şifre
* Yazar, Editör, Hakem seçeneklerinden biri

“Giriş Yap” seçeneği ile ise kullanıcının Kullanıcı adı ve Şifresi alınarak, yazar, editör ya da

hakem olmasına göre program devam etmelidir.

* Makaleler yazarlar tarafından sisteme yüklenmektedir.

Yazar giriş yaptığında, yeni bir makale yükleyebilmesi için gerekli bilgilerin girilebildiği ve

daha önce yüklemiş olduğu makalelerin bilgilerinin listelendiği bir ekran açılmalıdır.

Yazar yeni bir makale yüklemek için şu bilgileri girmelidir:

* Makale başlığı
* Yazarların ad ve soyadları (virgülle ayrılmış şekilde)
* Yükleme yapan yazarın e-posta adresi
* Yükleme yapan yazarın çalıştığı kurum
* Makalenin yükleme tarihi sistemden çekilmelidir.
* Makalenin mevcut durumu da “yüklendi” olarak kaydedilmelidir.

Her makalenin en az 1 yazarı olmalıdır. Sadece yükleme yapan yazarın bilgileri sistemde tutulur. Bir yazar birden fazla makale yüklemiş olabilir. Editör sisteme giriş yaptığında sistemde bulunan makale başlıklarının ve mevcut durumlarının listesini görmelidir.

* Editör, durumu “yüklendi” olan makaleler için sistemde kayıtlı olan hakemlerden birini atayabilmelidir.
* Bir hakeme atanan makalenin mevcut durumu “değerlendirmede” olarak değişmelidir.

Hakem sisteme giriş yaptığında değerlendirilmek üzere kendisine atanan makale başlıklarını görebilmelidir.

* Hakem kendisine atanan makalelere “kabul” ya da “ret” seçeneklerinden birini atayabilmelidir.
* Hakemin seçtiği sonuca göre makalenin mevcut durumu “kabul” ya da “ret” olarak değişecektir.

# Puan Listesi

* Kullanıcı girişlerinin gerçekleştirilmesi 15
* Yazar menüsü 15
* Editör menüsü 15
* Hakem menüsü 15
* ER diyagramları ve tabloların rapora eklenmesi 10
* Yazılan kodlara olan hakimiyet 30
* Bonus özellik (grafik ekran) 20

# Program Dili

Python, yüksek seviyeli, genel amaçlı bir programlama dilidir ve C++'ın temel özelliklerini benimser. C++'ın nesne yönelimli programlama (OOP) yeteneklerine benzer şekilde, Python da nesne yönelimli programlamayı destekler. Bu, veri ve işlevleri birleştirmek, yapıları modüler hale getirmek ve tekrar kullanılabilirlik sağlamak amacıyla sınıflar ve nesneler kullanmayı içerir.

Genişletilebilirlik açısından, Python, kullanıcıların kendi veri türlerini ve fonksiyonlarını tanımlamalarına izin veren bir dil olarak esnek bir yapı sunar. Bu da Python'ı çeşitli programlama ihtiyaçlarına uygun çözümler geliştirmek için uygun hale getirir.

Python'ın verimliliği, yüksek seviyeli bir dil olmasına rağmen hızlı çalışabilmesi ve bellek yönetiminde otomatikleştirmeye dayanabilmesi ile sağlanır.

Şablonlar konusunda, Python dilinde C++'taki gibi şablonlar yerine geniş bir standart kütüphane ve dinamik tür sistemine dayalı bir yaklaşım benimsenmiştir. Bu sayede, türden bağımsız fonksiyonlar ve yapılar kullanmak mümkündür.

Taşınabilirlik açısından, Python da platformdan bağımsız bir şekilde çalışabilir. Bu, Python'ın farklı işletim sistemleri ve donanım mimarileri üzerinde çalışabilme yeteneğini sağlar, bu da Python'ın taşınabilir ve çok platformlu uygulamalar geliştirmek için tercih edilen bir dil olmasını sağlar.

# Projeye Başlamadan Önce Neler Yaptık

İlk olarak projenin ana hedeflerini ve kapsamını netleştirmek için bir proje tanımı ve proje planı oluşturduk. Bu adımda, projenin neyi başarmayı hedeflediğini, hangi sonuçları elde etmek istediğimizi ve proje sürecinin nasıl ilerleyeceğini belirledik. Proje için gerekli kaynakları belirledik ve planladık. Projede yer alan tüm paydaşlar arasında etkili iletişimi sağlamak için bir iletişim planı oluşturduk. Bu plan, iletişim kanallarını, toplantılarını ve raporlama süreçlerini belirlemeyi içerir. Ayrıca paydaşların bilgilendirilmesi ve güncel tutulması için iletişim stratejileri geliştirdik. Proje sürecini yönetmek ve ilerlemeyi takip etmek için bir zaman çizelgesi oluşturduk. Bu zaman çizelgesi, projenin başlangıç ve bitiş tarihlerini, aşamalarını ve kilometre taşlarını içerir. Böylece proje ilerlemesini takip edebilir ve gerekirse planı güncelleyebiliriz. Projeye başlamadan önce gereken kaynakları temin ettik veya hazırladık. Bu kaynaklar, proje için gerekli olan kodumuzda kullanacağımız kütüphaneleri içerebilir.Kullanacağımız kütüphaneleri nerden indireceğimizi ve nasıl ekleyebileceğimize dair videolar bulup izledik.

# Proje Sayesinde Yeni Öğrendiğimiz Şeyler

Projemizin konusuyla ilgili olarak derinlemesine bir bilgi birikimi edindik. Araştırma yaparken, konuyla ilgili güncel trendleri, en iyi uygulamaları ve sektördeki gelişmeleri yakından takip ettik. Bu sayede, konu hakkında daha geniş bir perspektif kazandık ve uzmanlığımızı artırdık.

Bu projeyi yönetme sürecinde, proje yönetimi becerilerimizi geliştirdik. Zaman yönetimi, kaynak planlaması, risk yönetimi ve iletişim becerileri gibi önemli becerileri pratiğe döktük. Proje takvimini oluşturma, etkili iletişim kurma konularında daha yetkin hale geldik.

Projede kullanılan belirli teknikler veya araçlar üzerinde çalışarak yeni teknik yetkinlikler kazandık. Örneğin, proje yönetimi yazılımları, veri analizi araçları veya tasarım programları gibi özel araçlarla çalışarak bu alanlardaki becerilerimizi artırdık. Bu projenin gerektirdiği teknikleri öğrenerek, kendimizi teknolojik açıdan da geliştirdik.

Projede bir ekip olarak çalışarak takım çalışması becerilerimizi geliştirdik. Ekip üyeleriyle etkili iletişim kurma, görevleri koordine etme, işbirliği yapma ve çözüm odaklı çalışma gibi yetenekleri pekiştirdik. Birlikte çalışmanın önemini ve başarılı bir şekilde işbirliği yapmanın yollarını öğrendik.

Projenin farklı aşamalarında karşılaştığımız sorunlarla uğraşarak problem çözme yeteneklerimizi geliştirdik. Yaratıcı çözümler bulma, analitik düşünme, karar verme ve esneklik gibi becerilerimizi kullanarak zorlukları aştık. Bu deneyimler, gelecekteki projelerde karşılaşabileceğimiz sorunlara daha etkili bir şekilde yaklaşmamızı sağladı.

Projeyi tamamlarken, elde ettiğimiz sonuçları etkili bir şekilde sunma ve raporlama becerilerimizi geliştirdik.

# Projede Zorlandığımız Kısımlar

Bu projede, hiç bilmediğimiz bir dil olan python ile proje geliştirme sürecinde yaşadığımız sorunlar ve karşılaştığımız olumsuz deneyimleri paylaşmak istiyoruz. Proje için kullanmayı planladığımız grafik kütüphanesiyle ilgili yaşadığımız sorunlar oldukça can sıkıcıydı. Veri tabanını entegre etmekte zorluklar yaşadık yeterli kaynaklardan video izleyerek sorunlarımıza çözüm bulduk.

# Sonuç

Python, geniş bir kullanıcı kitlesine hitap eden, yüksek seviyeli bir programlama dilidir ve birçok avantaj sunar. Öncelikle, Python'ın okunabilir ve sade sözdizimi, kullanıcılarına kod yazmayı kolaylaştırır. Bu özellik, özellikle yeni başlayanlar için dilin öğrenilmesini hızlandırır ve programcıların kodlarını daha anlaşılır hale getirmelerini sağlar.

Python'ın geniş bir standart kütüphanesi bulunmaktadır. Bu kütüphane, birçok yaygın görevi kolayca gerçekleştirmeyi sağlar. Örneğin, dosya işlemleri, veri tabanı etkileşimleri, ağ programlaması ve daha pek çok konuda hazır modüller içerir. Bu durum, geliştiricilerin tekrarlayan işleri elle yazma ihtiyacını azaltarak zaman kazanmalarını sağlar.

Ayrıca, Python'ın geniş topluluğu ve büyük bir açık kaynak ekosistemi vardır. Bu durum, geliştiricilere zengin bir kaynak havuzu ve ücretsiz yardım ve destek sağlar. Python topluluğu, sorulara hızlı cevaplar, güncel kütüphaneler ve öğrenim kaynakları konusunda yardımcı olabilir.

Sonuç olarak, Python dilinin bu avantajları, geliştirme süreçlerini hızlandırdı,kod kalitesini artırdı ve daha etkili, sürdürülebilir yazılım projeleri oluşturmamıza yardımcı olmaktadır.

# ER Diyagramı

MAKALELER

KULLANİCİLAR

id (PK)

Eposta

id (PK)

Sifre

KullaniciAdi

Kurum

Makale\_Basligi

Rol

Tarih

Yazar

**SQL Tablo Yapıları:**

Kullanicilar Tablosu:

* id (Primary Key)
* KullaniciAdi
* Sifre
* Rol

CREATE TABLE IF NOT EXISTS Kullanicilar (

id INTEGER PRIMARY KEY,

KullaniciAdi VARCHAR(45),

Sifre VARCHAR(45),

Rol VARCHAR(10)

);

Makaleler Tablosu:

* id (Primary Key)
* Makale\_Basligi
* Yazar
* Eposta
* Kurum
* Tarih

CREATE TABLE IF NOT EXISTS Makaleler (

id INTEGER PRIMARY KEY,

Makale\_Basligi VARCHAR(45),

Yazar VARCHAR(45),

Eposta VARCHAR(45),

Kurum VARCHAR(45),

Tarih INT

);

# Kaynakça

**Python Programlama Dili:**

<https://sogrekci.com/ders-notu/python-ve-bilimsel-hesaplama/42-grafik-cizimi/>

<https://medium.com/datarunner/matplotlibkutuphanesi-1-99087692102b>

<https://kadakod.com/python-ile-grafik-arayuz-hazirlama-qt-designer/>

<https://dataiteam.com/python/>