

# YAZILIM LABORATUVARI

## PROJE 2

Anıl Engin Keretli

Ahmet Burak Karkaç

Kocaeli Üniversitesi

Kocaeli Üniversitesi

Kocaeli/TÜRKİYE

Kocaeli/TÜRKİYE

230201128

220201173

### Akıllı Etkinlik Planlama Platformu

#### Özet

Bu raporda, modern bir etkinlik planlama platformu geliştirme süreci anlatılmaktadır. Proje kapsamında kullanıcılar, etkinlik oluşturabilir, bu etkinliklere katılabılır ve platformun sunduğu kişiselleştirilmiş önerilerden yararlanabilir. Ayrıca, bir harita entegrasyonu sayesinde kullanıcılar, etkinlik konumlarını görüntüleyebilir ve önerilen rotaları alabilir. Proje sürecinde .NET Framework ve PostgreSQL kullanılmış; kullanıcı deneyimini artırmak için dinamik bir frontend tasarımlı gerçekleştirilmiştir.

Platformun ana özellikleri arasında kullanıcı yönetimi, kişiselleştirilmiş öneriler, çakışma kontrolü, mesajlaşma sistemi ve oyunlaştırma unsurları bulunmaktadır. Bu proje, hem teknik becerilerin geliştirilmesine hem de ekip çalışması deneyiminin artırılmasına olanak sağlamıştır.

#### Program.cs

ASP.NET Core uygulamasının çalıştırılmasını sağlar. PostgreSQL veritabanı bağlantısı yapılır, ApplicationDbContext ile veritabanı işlemleri yönetilir. Servis katmanları (AuthenticationService, MessageService, vb.) DI konteynerine eklenir. Kimlik doğrulama ve yetkilendirme için AddAuthentication yapılandırılmıştır. Uygulama, hata yönetimi, yönlendirme, HTTPS yönlendirme ve statik dosya işlemlerini destekler. Varsayılan rota düzeni belirlenir ve app.Run() çağrılarıyla uygulama başlatılır.

#### Giriş

Etkinlik planlama ve yönetimi, hem bireysel hem de topluluklar için büyük bir ihtiyaçtır. Geleneksel yöntemlerin yetersiz kaldığı durumlarda dijital çözümler önem kazanmaktadır. Bu doğrultuda, "Akıllı Etkinlik Planlama Platformu" projesi, kullanıcılarla kişiselleştirilmiş deneyimler sunarak etkinlik oluşturma, katılım ve sosyal etkileşim ihtiyaçlarını karşılamayı hedeflemektedir.

Projenin amacı, kullanıcı dostu bir platform oluşturmak ve yazılım geliştirme süreçlerini uygulamalı olarak deneyimlemektir. Bu süreçte .NET Framework, PostgreSQL ve React.js gibi teknolojiler kullanılarak modern bir web uygulaması geliştirilmiştir.

Projenin temel hedefleri şunlardır:

Kullanıcıların ilgi alanlarına uygun etkinlik önerileri sunmak.

Etkinliklere katılım süreçlerini kolaylaştırmak ve organize etmek.

Harita ve rota planlama entegrasyonuyla kullanıcı deneyimini geliştirmek.

Veritabanı tasarımları ve API entegrasyonlarıyla sistemin dinamik çalışmasını sağlamak.

AdminService yönetici işlemleri için kullanılır. Veritabanı bağlamı (ApplicationDbContext) üzerinden etkinlik ve kullanıcı verilerini yönetir.

getAllEvents ve getAllUsers: Tüm etkinlikleri ve kullanıcıları liste olarak döner.

deleteUser ve deleteEvent: Belirtilen ID'ye göre kullanıcıyı veya etkinliği veritabanından siler. getEventById ve getUserId: Belirtilen ID'ye sahip bir etkinlik veya kullanıcıyı veritabanından bulur ve döner.

Bu sınıf, yöneticilerin kullanıcıları ve etkinlikleri yönetmesine olanak tanır.

## ADMİN PANELİ

The screenshot shows the Admin Panel interface with two main sections:

- Kullanıcılar** (Users): A table with columns: Kullanıcı Adı (Username), Email, İşlemler (Actions), Etkinlik Adı (Event Name), Başlangıç Tarihi (Start Date), and İşlemler (Actions). It lists two users: 'anil' and 'burak'.
- Etkinlikler** (Events): A table with columns: Etkinlik Adı (Event Name), Başlangıç Tarihi (Start Date), and İşlemler (Actions). It lists two events: 'dogumgunu' and 'yuzmek'.

## KAYIT İŞLEMLERİ

kullanıcı doğrulama ve kaydı için bir AuthenticationService sınıfını tanımlar. Kullanıcı işlemleri, veritabanı bağlamı (ApplicationDbContext) üzerinden gerçekleştirilir.

Authenticate: Kullanıcı adı ve şifreye göre kullanıcıyı kontrol eder. Eşleşen bir kullanıcı bulunursa, o kullanıcıyı döner. Aksi halde null döner.

Register: Yeni bir kullanıcı kaydı oluşturur. Kullanıcının bilgilerini alır, bir User nesnesi oluşturur ve veritabanına ekler. Varsayılan profil fotoğrafı URL'si tanımlıdır. Ayrıca, kullanıcının ilgi alanlarını liste olarak alır ve veritabanında saklamak için birleştirir.

Bu servis, kullanıcı giriş ve kayıt işlemlerinin sorunsuz şekilde gerçekleştirilebilmesini sağlar.

```
■CLASS AuthenticationService
  PRIVATE VARIABLE context AS
  ApplicationContext

  METHOD Constructor(context:
  ApplicationContext)
    SET this.context TO context
  END METHOD

  METHOD Authenticate(username: STRING,
  password: STRING) RETURNS User
    SEARCH for a user in context.Users
    WHERE
      Username MATCHES username AND
      PasswordHash MATCHES password

    IF user FOUND
      RETURN user
    ELSE
      RETURN NULL
    END IF
  END METHOD

  METHOD Register(username: STRING,
  password: STRING, email: STRING,
  createdDate: DATETIME,
  firstName: STRING, lastName: STRING,
  phoneNumber: STRING, location:
  STRING, aboutMe: STRING,
  birthDate: DATETIME, gender:
  STRING,
  profilePictureUrl: STRING, latitude:
  STRING,
  longitude: STRING,
  selectedInterests: LIST OF STRING)
    RETURNS User

    CREATE newUser AS User WITH
    PROPERTIES:
      Username = username
      PasswordHash = password
      Email = email
      CreatedAt = Convert createdDate TO
      Universal Time
      FirstName = firstName
      LastName = lastName
      PhoneNumber = phoneNumber
      Location = location
      AboutMe = aboutMe
      Interests = Concatenate selectedInterests
      INTO STRING
      BirthDate = Convert birthDate TO
      Universal Time
      Gender = gender
      ProfilePictureUrl = profilePictureUrl OR
      Default URL
      Latitude = latitude
      Longitude = longitude

    ADD newUser TO context.Users
    SAVE changes TO context

    RETURN newUser
  END METHOD
END CLASS.
```

## ETKİNLİKLER

Bu kod, etkinlik oluşturma, güncelleme, silme ve kullanıcıların etkinliklere katılımını yönetmek için bir EventService sınıfı sağlar. Fonksiyonlar, etkinlik çıkışmalarını kontrol eder, öneriler sunar, gamification puanlarını günceller ve kullanıcı etkinlik ilişkilerini yönetir. Ayrıca kullanıcı önerileri ve alternatif başlangıç tarihleri gibi özellikler içerir. Entity Framework kullanılarak veritabanıyla etkileşim sağlanır.

Etkinlik Oluştur

Etkinlik Adı  
Enter event name

Açıklama  
Bir açıklama yaz

Adres  
Adresi tari et

Başlangıç Zamanı  
gg aa yyyy -- --

Bitiş Zamanı  
gg aa yyyy -- --

Etkinlik Resim URL  
url

Kategori  
Kategori seçin

Etkinlik Konumu  
Harita

**Etkinlik oluşturma sayfası**

yuzmek

Açıklama  
Haydi yüzelim

Adres  
denizde

Başlangıç Zamanı  
3.12.2024

Bitiş Zamanı  
4.12.2024

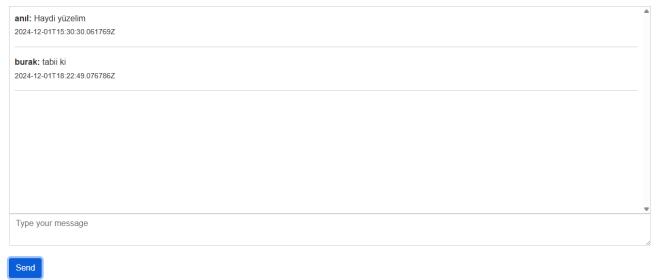
Etkinlik Sahibi  
anil

Etkinlik Səhərbelə Kəfə

**Etkinlik örneği**

## Mesajlar

**MessageController** mesajları görüntülemek için bir EventMessageViewModel sunar. MessageController, belirli bir etkinliğe ait mesajları alma (GetMessagesByEvent) ve mesaj gönderme (SendMessage) işlemlerini gerçekleştirir. EventMessageViewModel, seçili etkinlik, kullanıcı etkinlik listesi ve mesaj detaylarını bir arada sunar. Bu yapı, etkinlik tabanlı mesajlaşmayı destekleyerek RESTful API ile etkileşim sağlar. Kod, MessageService kullanarak iş mantığını soyutlar ve DTO ile sadeleştirilmiş veri aktarımı sağlar.



## PUAN HESABI

**GamificationService** puan hesaplamalarını yöneterek kullanıcıların etkinlik katılımını teşvik eder. Kullanıcılara etkinlik katılımı ("Participation"), oluşturma ("Creation") ve bonus aktiviteleri üzerinden puan kazandırır.

**Puan Yönetimi:** Kullanıcı yoksa, yeni bir UserPoints kaydı oluşturur ve etkinlik türüne göre ilgili puanları günceller. Toplam puan, katılım, oluşturma ve bonus puanlarının toplamıdır.

**Etkinlik Doğrulama:** Puan eklenmeden önce geçerli bir EventId kontrolü yapılır.

**Aktivite Geçmiş:** Her puan kazanımı, ActivityHistory tablosunda kaydedilerek kullanıcı aktivitelerinin izlenmesini sağlar.

**Kullanıcı Puanlarını Alma:** Kullanıcının mevcut puanları, GetUserPoints metodu ile sorgulanabilir. Bu sistem, kullanıcı bağlılığını artırmaya yönelik bir puanlama altyapısı sunar.

**UserService**, kullanıcı yönetimi işlemleri kolaylaştıran bir servis sınıfıdır.

**Kullanıcı Yönetimi:** Kullanıcıların listesini almak (getAllUsers), kullanıcı detaylarını kimlik veya kullanıcı adı ile sorgulamak (getUserDetailsById, getUserDetailsByUserName) gibi işlemleri destekler.

**Kullanıcı Oluşturma:** Kullanıcı adı benzersizse, verilen bilgilerle yeni bir kullanıcı oluşturur ve kaydeder (createUser). Varsayılan profil fotoğrafı URL'si belirlenmiştir.

**Kullanıcı Güncelleme:** Var olan bir kullanıcıyı günceller ve veritabanında değişiklikleri kaydeder (updateUser).

**Kullanıcı Silme:** Kullanıcıyla ilişkili tüm

kullanıcıyı siler (deleteUser).

Servis, kullanıcılar üzerinde temel CRUD (oluşturma, okuma, güncelleme, silme) işlemleri sağlamanın yanı sıra, kullanıcıyla ilgili bağımlı verilerin yönetimini de kapsar.

## KATKILAR

Buprojede Anıl, EventService, GamificationService, ve MessageService'in geliştirilmesinden sorumlu oldu. Ahmet ise UserService, veri modelleme, kullanıcı işlemleri, ve API entegrasyonlarını üstlenerek projeye katkıda bulundu. İkili, tasarım ve testlerde iş birliği yaptı.

**Entity Framework Core Documentation**  
<https://learn.microsoft.com/en-us/ef/core/>

**ASP.NET Core Fundamentals**  
<https://learn.microsoft.com/en-us/aspnet/core/fundamentals/>

**C# Programming Guide**  
<https://learn.microsoft.com/en-us/dotnet/csharp/>

**LINQ (Language Integrated Query)**  
<https://learn.microsoft.com/en-us/dotnet/csharp/programming-guide/concepts/linq/>

**Dependency Injection in ASP.NET Core**  
<https://learn.microsoft.com/en-us/aspnet/core/fundamentals/dependency-injection/>

The screenshot shows a user interface for managing events. At the top, there is a navigation bar with buttons for 'today', 'month', 'week', and 'day'. Below it, a calendar for December 2024 is displayed, with the 1st of December highlighted in green and labeled '15:29 yarın' (15:29 tomorrow). A tooltip 'Oluşturduğunuz Etkinlikler' (Your Created Events) points to a list of events on the right. The list includes an event titled 'dogumgunu' (Birthdays) from 'Tarih: 01.05.2025'. Below this, another section titled 'Katıldığınız / Başvurduğunuz Etkinlikler' (Your Attended / Applied Events) shows an event titled 'yuzmek' (Swimming) from 'Tarih: 03.12.2024'.

