

**ANKARA ÜNİVERSİTESİ**

**Bilgisayar Mühendisliği Bölümü**

**(BLM4531) Ağ Tabanlı Teknolojiler Ve Uygulamaları**

**PROJE: Etkinlik Yönetim Sistemi (Event  
Management System)**

**İŞİM-SOYAD:** SARDORKHON KHAMIDOV

**BÖLÜM:** BİLGİSAYAR MÜHENDİSLİĞİ

**Ö.N:** 20291221

*Proje Kaynak Kodu (GitHub):*

<https://github.com/Projects-uni/EventManagementSystem>

*Ptoje Anlatim Videosu:*

[https://drive.google.com/file/d/1nkqCRYGHJAgcKqyurl2SMpZ6sPhrT\\_oO/view?usp=sharing](https://drive.google.com/file/d/1nkqCRYGHJAgcKqyurl2SMpZ6sPhrT_oO/view?usp=sharing)

## 1. GİRİŞ

Bu proje kapsamında, web tabanlı bir **Etkinlik Yönetim Sistemi (Event Management System)** geliştirilmiştir. Geliştirilen sistem, etkinliklerin planlanması, organize edilmesi, görevlerin atanması ve katılımcıların yönetilmesi gibi süreçleri tek bir platform üzerinden gerçekleştirmeyi amaçlamaktadır.

Dönem başında teslim edilen proje dokümanında sistemin kapsamı, hedefleri ve planlanan modüller detaylı şekilde açıklanmıştır. Dönem sonunda ise bu plan doğrultusunda **tüm temel modüller başarıyla tamamlanmış**, buna ek olarak sistemin kullanılabilirliğini ve profesyonellliğini artırmak amacıyla **Swagger UI entegrasyonu** ve **bildirim mekanizması** projeye eklenmiştir.

Bu rapor, projenin **nihai hâlini**, geliştirilen tüm bileşenleri ve sistemin çalışma mantığını detaylı şekilde açıklamak amacıyla hazırlanmıştır.

*Proje Kaynak Kodu (GitHub):*

<https://github.com/Projects-uni/EventManagementSystem>

*Ptoje Anlatim Videosu:*

[https://drive.google.com/file/d/1nkqCRYGHJAgcKqyurl2SMpZ6sPhrT\\_oO/view?usp=sharing](https://drive.google.com/file/d/1nkqCRYGHJAgcKqyurl2SMpZ6sPhrT_oO/view?usp=sharing)

---

## 2. SİSTEMİN GENEL MİMARİSİ

Geliştirilen uygulama, **ASP.NET Core MVC** mimarisi kullanılarak oluşturulmuştur. Bu mimari sayesinde sistem;

- Modüler
- Genişletilebilir
- Bakımı kolay

The screenshot shows the 'Etkinlik Yönetim Sistemi' (Event Management System) interface. At the top, there's a navigation bar with links for 'Etkinliklerim', 'Davetler', and '+ Yeni Etkinlik Oluştur'. Below the navigation, there are two tabs: 'Kategori Yönetimi' (Category Management) and 'Konum Yönetimi' (Location Management), with 'Kategori Yönetimi' being active. The main content area displays a grid of event cards. Each card contains the event title, category, start and end dates, location, and organizer. Buttons for 'Detayları Gör' (View Details) and 'Organizatör' (Organizer) are also present. The events listed are:

- test event 2232322332** (Ongoing)  
Category: test 2  
Start: 30 Dec 2025 10:35  
End: 30 Dec 2025 10:35  
Location: konum test 1 - ankara  
Organizer: test2
- test event 2avasvassavsa** (Upcoming)  
Category: test 1  
Start: 30 Dec 2025 04:51  
End: 01 Jan 2026 04:51  
Location: konum test 1 - ankara  
Organizer: test2
- test event 000** (Ongoing)  
Category: test 1  
Start: 29 Dec 2025 04:15  
End: 30 Dec 2025 04:15  
Location: konum test 1 - ankara  
Organizer: test2
- test event 2** (Upcoming)  
Category: Conference  
Start: 07 Dec 2025 15:35  
End: 21 Dec 2025 15:35
- test event 1** (Upcoming)  
Category: Conference  
Start: 24 Nov 2025 11:30  
End: 01 Dec 2025 01:30

bir yapıya sahip olmuştur.

Sistem üç ana katmandan oluşmaktadır:

## 2.1 Presentation Layer (MVC – View Katmanı)

Bu katman, kullanıcıların sistemle etkileşime geçtiği arayüzü temsil etmektedir. HTML, CSS ve Razor View yapısı kullanılarak aşağıdaki sayfalar geliştirilmiştir:

- Login / Sign-Up sayfaları
- Event List (Etkinlik Listesi)
- Event Detail (Etkinlik Detayı)
- Event Create (Etkinlik Oluşturma)
- Admin Dashboard

- ➡ Ana sayfa
- ➡ Etkinlik listesi ekranı

## 2.2 Controller (İş Mantığı Katmanı)

Controller katmanı, kullanıcıdan gelen istekleri karşılar ve gerekli işlemleri gerçekleştirir. Bu projede başlıca aşağıdaki controller'lar kullanılmıştır:

- AuthController
- EventsController
- TasksController
- ParticipantsController
- AdminController
- API Controller'lar

Her controller, **yetkilendirme kontrolleri** yaparak sadece ilgili rolün işlem yapmasına izin verir.

---

## 2.3 Data Layer (Veri Katmanı)

Veri katmanı olarak **Supabase PostgreSQL** kullanılmıştır. ASP.NET Core uygulaması, Supabase'e **Npgsql** bağlantı sağlayıcısı üzerinden bağlanmaktadır. Veri erişimi **Entity Framework Core** ile gerçekleştirilmiştir.

---

# 3. KULLANICI ROLLERİ VE YETKİLENDİRME YAPISI

Sistemde üç farklı kullanıcı rolü bulunmaktadır:

## 3.1 Admin

- Tüm etkinlikleri görüntüleyebilir
- Etkinlikleri düzenleyebilir veya silebilir
- Tüm kullanıcıları ve görevleri görebilir
- Sistem istatistiklerine erişebilir

## 3.2 Organizer

- Yeni etkinlik oluşturabilir
- Kendi etkinliklerine görev ekleyebilir
- Katılımcı davet edebilir
- Kendi etkinliklerini düzenleyebilir

## 3.3 Guest

- Davet edildiği etkinlikleri görüntüleyebilir
- Etkinlik detaylarını inceleyebilir

**Etkinlik Yönetim Sistemi**

**Giriş Yap**

Kullanıcı Adı

Şifre

**Giriş Yap**

Hesabın yok mu? [Kayıt Ol](#)

**Etkinlik Yönetim Sistemi**

**Hesap Oluştur**

Kullanıcı Adı \*

E-posta \*

Şifre \*

Rol Seçimi \*

**Kayıt Ol**

Zaten hesabınız var mı? [Giriş Yap](#)

Yetkilendirme işlemleri **Session tabanlı** olarak gerçekleştirilmiştir. Kullanıcının rol bilgisi oturumda tutulmakta ve her istekte kontrol edilmektedir.

## 4. GERÇEKLENEN SİSTEM MODÜLLERİ

### 4.1 Giriş ve Kayıt (Authentication) Modülü

Kullanıcılar sisteme e-posta ve şifre bilgileri ile giriş yapabilmektedir. Yeni kullanıcılar kayıt formu üzerinden sisteme eklenir.

- Kullanıcı bilgileri `Users` tablosunda tutulur
- Başarılı giriş sonrası kullanıcı rolüne göre yönlendirme yapılır
- Oturum bilgileri Session üzerinde saklanır

- ➡ Login ekranı  
➡ Sign-Up ekranı

### 4.2 Etkinlik Oluşturma Modülü

Etkinlik Adı \*

Örnek: Yıllık Teknoloji Konferansı 2024

Açıklama

Etkinliğinizinizi açıklayın...

Kategori

Kategori Seçin

Durum \*

Yaklaşan

Kategori listesi API üzerinden (categories tablosu) yüklenir.

Başlangıç Tarihi & Saati \*

dd.mm.yyyy, --::--

Bitiş Tarihi & Saati \*

dd.mm.yyyy, --::--

Konum

Konum Seçin

Konum İstesi API üzerinden (locations tablosu) yüklenir.

Ek Konum Notu (Opsiyonel)

Örnek: Kat 2, Salonun sol tarafı

Bütçe (\$)

Örnek: 5000.00

Opsiyonel: Bu etkinlik için bütçe girin

Etkinliği Oluştur

Organizer rolündeki kullanıcılar yeni etkinlik oluşturabilmektedir. Etkinlik oluşturma sırasında aşağıdaki bilgiler sisteme girilir:

- Etkinlik adı
- Açıklama
- Kategori
- Başlangıç ve bitiş tarihi
- Konum
- Etkinlik durumu

Bu bilgiler Events tablosuna kaydedilir ve etkinlik, oluşturan kullanıcı ile ilişkilendirilir.

➡ Etkinlik oluşturma formu

#### 4.3 Görev (Task) Yönetimi Modülü

Her etkinlik için yapılması gereken görevler tanımlanabilmektedir. Görevler:

- Görev adı
- Açıklama
- Bitiş tarihi
- Atanan kullanıcı
- Görev durumu

bilgilerini içermektedir.

Görevler Tasks tablosunda saklanmakta ve ilgili etkinlik ile ilişkilendirilmektedir.

---

#### **4.4 Katılımcı Yönetimi Modülü**

Etkinliğe kullanıcı veya misafir davet edilebilmektedir. Katılımcı bilgileri:

- Ad – Soyad
- E-posta
- Rol
- Davet durumu

şeklinde tutulmaktadır.

---

#### **4.5 Etkinlik Listeleme ve Detay Modülü**

Kullanıcılar:

- Kendi oluşturdukları
- Davet edildikleri

etkinlikleri liste halinde görebilmektedir. Detay sayfasında:

- Etkinlik bilgileri
- Görevler
- Katılımcılar

birlikte gösterilmektedir.

---

#### **4.6 Admin Paneli**

Admin paneli, sistemin genel durumunu izlemek için geliştirilmiştir. Panelde:

- Toplam etkinlik sayısı
- Aktif / tamamlanan etkinlikler
- Kullanıcı sayısı
- Görev istatistikleri

görüntülenmektedir.

---

## 5. API YAPISI VE SWAGGER ENTEGRASYONU

Sistemde **RESTful API** mimarisi kullanılmıştır.

Her ana varlık için ayrı API endpoint'leri oluşturulmuştur:

- /api/events
- /api/tasks
- /api/participants
- /api/users
- /api/categories
- /api/Invitations
- /api/Locations
- /api/Notifications

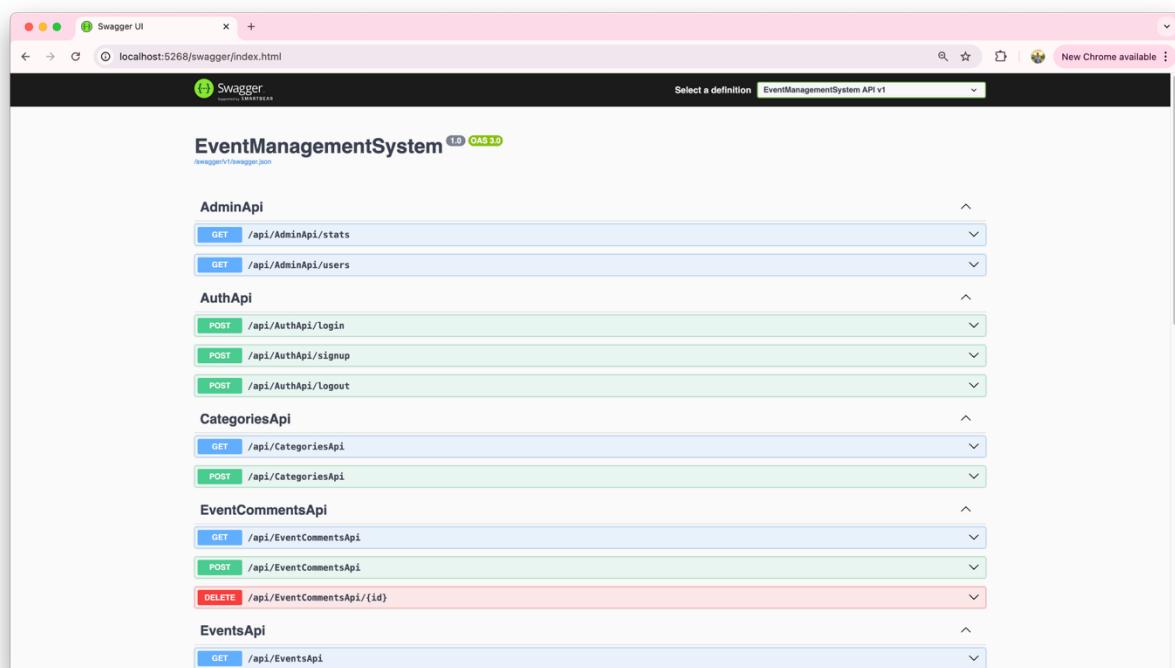
API'lerin test edilebilmesi ve dokümantasyonu için **Swagger UI** projeye entegre edilmiştir. Swagger sayesinde API endpoint'leri görsel olarak test edilebilmektedir.

- ➡ Swagger UI ana ekranı
  - ➡ API test örnekleri
- 

## 6. BİLDİRİM (NOTIFICATION) SİSTEMİ

Kullanıcı deneyimini artırmak amacıyla bildirim mantığı projeye eklenmiştir. Sistem, önemli olaylarda kullanıcıyı bilgilendirmektedir:

- Etkinliğe davet edilme



The screenshot shows the Swagger UI interface for the EventManagementSystem API v1. The main title is "EventManagementSystem 1.0 OAS 3.0". The interface is organized into sections for different APIs: AdminApi, AuthApi, CategoriesApi, EventCommentsApi, and EventsApi. Each section contains a list of endpoints with their HTTP methods and URLs. The "EventCommentsApi" section is expanded, showing three POST methods: "/api/EventCommentsApi", "/api/EventCommentsApi", and "/api/EventCommentsApi/{id}"; the last one is highlighted with a red background.

- Yeni görev atanması
- Etkinlik güncellemeleri

Bu yapı, sistemin gerçek hayatı kullanım senaryolarına daha uygun hâle gelmesini sağlamıştır.

---

## 7. VERİTABANI TASARIMI

Veritabanı olarak **Supabase PostgreSQL** kullanılmıştır. Kullanılan ana tablolar:

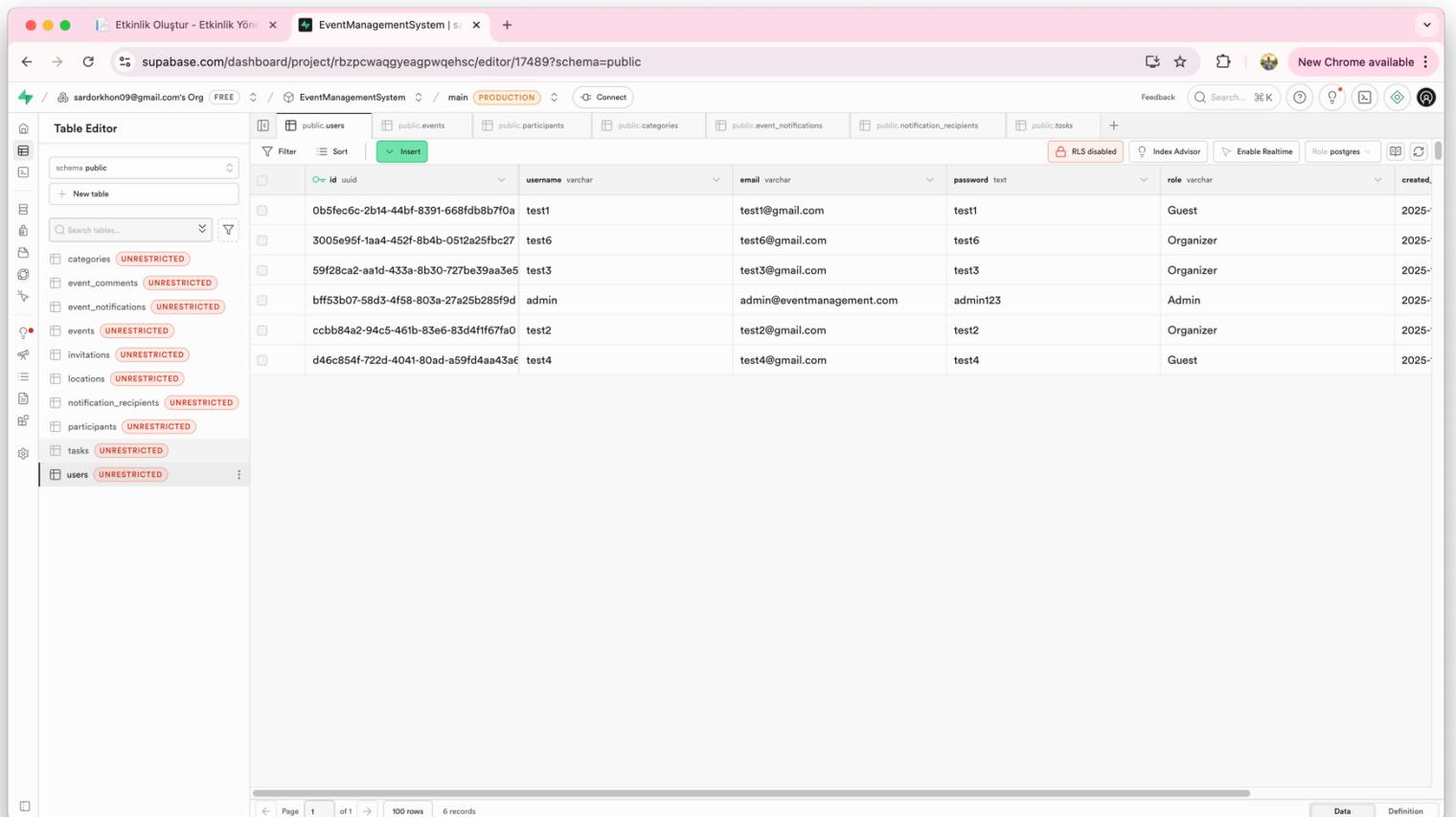
- Users
- Events
- Tasks
- Participants
- Categories
- Invitations
- Locations
- EventNotifications

Tablolar arası ilişkiler **foreign key** yapıları ile kurulmuştur.

➡ Supabase tablo yapısı

## 8. KULLANILAN TEKNOLOJİLER

- ASP.NET Core MVC
- Entity Framework Core
- Supabase PostgreSQL
- RESTful API
- Swagger UI
- HTML, CSS, JavaScript



The screenshot shows the Supabase Table Editor interface for the 'public.users' table. The table has the following columns: id (uuid), username (varchar), email (varchar), password (text), role (varchar), and created\_ (date). There are six rows of data:

	<code>id</code>	<code>username</code>	<code>email</code>	<code>password</code>	<code>role</code>	<code>created_</code>
1	0b5fec6c-2b14-44bf-8391-668fdb8b7f0a	test1	test1@gmail.com	test1	Guest	2025-
2	3005e95f-1aa4-452f-8b4b-0512a25fb27	test6	test6@gmail.com	test6	Organizer	2025-
3	59f28ca2-a1d-433a-8b30-727be39a3e5	test3	test3@gmail.com	test3	Organizer	2025-
4	bff53b07-58d3-4f58-803a-27a25b285f9d	admin	admin@eventmanagement.com	admin123	Admin	2025-
5	ccbb84a2-94c5-461b-83e6-83d4ff67fa0	test2	test2@gmail.com	test2	Organizer	2025-
6	d46c854f-722d-4041-80ad-a59fd4aa43a6	test4	test4@gmail.com	test4	Guest	2025-

The left sidebar shows other tables: public.events, public.participants, public.categories, public.event\_notifications, public.notification\_recipients, and public.tasks. The 'users' table is selected.

## 9. TEST VE DOĞRULAMA

Sistem manuel olarak test edilmiştir. CRUD işlemleri, yetkilendirme kontrolleri ve API çağrıları başarıyla çalışmaktadır. Swagger UI üzerinden API testleri yapılmıştır.

---

## 10. SONUÇ VE DEĞERLENDİRME

Bu projede, dönem başında planlanan tüm modüller eksiksiz şekilde tamamlanmıştır. Ek olarak Swagger UI ve bildirim sistemi entegre edilerek sistem daha profesyonel bir yapıya kavuşturulmuştur.

Geliştirilen Etkinlik Yönetim Sistemi, gerçek dünyada kullanılabilecek düzeyde, modüler ve genişletilebilir bir web uygulamasıdır.