



softITo 4. DÖNEM 1. GRUP
BACKEND DEVELOPER
BİTİRME PROJESİ RAPORU

SmartMunicipality (Akıllı Belediye Yönetim Sistemi)

Melis Yanık

Temmuz, 2026

softITo Yazılım Bilişim Akademisi, İstanbul

İÇİNDEKİLER

1. GİRİŞ

- 1.1. Projenin Amacı
- 1.2. Projenin Kapsamı ve Kısıtlar

2. MATERYAL VE YÖNTEM

- 2.1. Kullanılan Teknolojiler
- 2.2. Sistem Mimarisi ve Katmanlı Yapı
- 2.3. Veritabanı Şeması ve İlişkisel Yapı (ER Model)
- 2.4. Raporlama ve Gösterge Paneli Mimarisi
- 2.5. Yapay Zeka Destekli Asistan Servisi
- 2.6. Sistem İzleme ve Loglama (Serilog)
- 2.7. Önbellekleme (Caching) Altyapısı

3. KURULUM VE YAPILANDIRMA

- 3.1. Dosya Hiyerarşisi ve Çözüm Yapısı
- 3.2. Kurulum Adımları
- 3.3. API Görünümleri (Swagger UI) ve Konsol Takibi

4. SİSTEM KULLANIMI VE EKRAN GÖRÜNTÜLERİ

- 4.1. Aşama 1: Karşılama ve Kimlik Doğrulama
- 4.2. Aşama 2: Vatandaş Portal Akışı
- 4.3. Aşama 3: Operatör ve Yönetici Paneli
- 4.4. Aşama 4: Yüksek Performanslı Analiz

5. SONUÇ

6. KAYNAKÇA

1. GİRİŞ

1.1. Projenin Amacı

SmartMunicipality, modern belediyeçilik süreçlerini dijitalleştirerek vatandaşlar ile yerel yönetim arasındaki iletişimi ve hizmet akışını optimize etmek amacıyla geliştirilmiş kurumsal düzeyde bir Akıllı Belediye Yönetim Platformudur.

Proje; katmanlı mimari, bağımsız API servisleri, rol bazlı yetkilendirme, hibrit veri erişim katmanı (EF Core & Dapper) ve yapay zeka entegrasyonu gibi modern yazılım desenleri kullanılarak inşa edilmiştir. Temel amaç, vatandaşların şikayet, fatura ve abonelik gibi işlemlerini tek bir merkezden hızlı ve güvenilir bir biçimde çözmelerini sağlamaktır.

1.2. Projenin Kapsamı ve Kısıtlar

Bu platform, vatandaşlar ve belediye personeli (operatör/yönetici) için iki ana portaldan oluşmaktadır. Sisteme giriş yapan kullanıcıların rolleri dinamik olarak e-posta uzantılarına göre belirlenmektedir:

- Yönetici (Admin): '@belediye.bel.tr' uzantılı hesaplar. Tüm sistemi yönetme, analiz raporlarını inceleme ve sistem ayarlarını yönetme yetkilerine sahiptir.
- Operatör (Operator): '@operator.bel.tr' uzantılı hesaplar. Abonelik kayıtlarını oluşturma, dönemsel fatura kesme, vatandaşlardan gelen şikayetleri değerlendirme ve yanıtlama işlemlerini yürütür.
- Vatandaş (Citizen): Standart e-posta uzantıları. Kendi aboneliklerini yönetir, sanal POS simülasyonu ile fatura öder, konum destekli şikayet bildirimi yapar ve yapay zeka asistanı ile etkileşime girer.

Yetki Sınırları ve Kısıtlar (Yetkilendirme Matrisi):

İlgili Modül	Vatandaş (Citizen)	Operatör / Yönetici
Fatura Görüntüleme & Güvenli Ödeme	✓	✓
Şikayet Bildirimi Oluşturma & Süreç Takibi	✓	✓
Yapay Zeka Asistanı Kullanımı	✓	✓
Yeni Abonelik Başvurusu	✓	✓
Şikayet Durumu Güncelleme & Resmi Cevap Yazma	✗	✓
Abonelik Onaylama, Sayaç Okuma & Fatura Üretme	✗	✓
Analitik Raporlar & İstatistik Gösterge Paneli	✗	✓

2. MATERYAL VE YÖNTEM

2.1. Kullanılan Teknolojiler

Projenin geliştirilme sürecinde kurumsal web uygulamaları standartlarına uygun olarak en güncel teknolojiler kullanılmıştır:

- Backend Framework: ASP.NET Core 8.0 (C#)
- Veritabanı Yönetimi: Microsoft SQL Server
- ORM (Object-Relational Mapping):
 - Entity Framework Core: Veritabanı şema yönetimi (Code-First) ve standart transactional CRUD işlemleri için.
 - Dapper: Yüksek performans gerektiren veri okuma, istatistik ve analitik raporlamalar için.
- Güvenlik ve Kimlik Doğrulama: ASP.NET Core Identity, JWT (JSON Web Token), Cookie Tabanlı Güvenlik Altyapısı.
- Loglama (Günlükleme): Serilog (Yapılandırılmış, dosya tabanlı hata izleme).
- Raporlama ve Dosya Dışa Aktarımı:
 - QuestPDF: Açık kaynak kodlu ve oldukça yetenekli bir kütüphane olan QuestPDF, fatura ödemeleri sonrasında vatandaşa profesyonel görünümlü, piksel-mükemmel (pixel-perfect) PDF makbuz ve fatura dökümleri sunmak için entegre edilmiştir. Dinamik tablo ve sayfa yapılarını kod tabanlı tasarlayabilmeyi sağlar.
 - EPPlus: Faturaların, şikayetlerin veya tablo formatındaki diğer kritik verilerin Excel (.xlsx) dosyası olarak dışa aktarımını sağlamak amacıyla kullanılmıştır. Bu teknoloji ile doğrudan hafıza (MemoryStream) üzerinden hücre biçimlendirmeli Excel raporları saniyeler içinde oluşturulur.
- Yapay Zeka Servisleri: Google Gemini AI API (`gemini-1.5-flash`) ve OpenAI API (`gpt-3.5-turbo`).

Proje Bağımlılıkları (Kritik NuGet Paketleri):

Sistemin farklı katmanlarında kullanılan temel üçüncü taraf kütüphaneler şunlardır:

- `Dapper` (v2.1.79) - Yüksek performanslı veri okuma (DATA katmanı).

- `Microsoft.EntityFrameworkCore.SqlServer` & `Tools` (v8.0.25) - Ana ORM aracı ve Migration işlemleri.
- `QuestPDF` (v2026.7.0) - MVC katmanında PDF fatura oluşturma.
- `EPPlus` (v8.6.1) - MVC katmanında Excel çıktıları alma.
- `Serilog` & `Serilog.AspNetCore` (v10.0) - Kapsamlı ve yapılandırılmış loglama mekanizması.
- `Swashbuckle.AspNetCore` (v6.6.2) - EF Core API ve Dapper API uç noktalarının Swagger UI ile dokümantasyonu.
- `Microsoft.AspNetCore.Authentication.JwtBearer` - Güvenli rol tabanlı JWT kimlik doğrulama.

2.2. Sistem Mimarisi ve Katmanlı Yapı

Sistem, gevşek bağlı (loosely coupled) ve REST API'ler aracılığıyla haberleşen servis tabanlı (Service-Oriented) bir mimariye sahiptir. N-Tier (Çok Katmanlı) yapısı sayesinde sunum, iş mantığı ve veri erişim sorumlulukları birbirinden kesin olarak ayrılmıştır.

* SmartMunicipality MVC (Sunum Katmanı): Kullanıcıların ve operatörlerin etkileşime girdiği web arayüzüdür. Veritabanı ile doğrudan iletişim kurmaz; tüm istekleri API endpoint'lerine yönlendirir.

* SmartMunicipality.EFCoreApi: Temel veri yönetimi (CRUD), Identity entegrasyonu, yetkilendirme (JWT) ve iş kurallarının işletildiği ana API katmanıdır.

* SmartMunicipality.DapperApi: Analitik raporların üretildiği, istatistiklerin çekildiği ve yüksek hız gerektiren okuma (Read) sorgularının bulunduğu bağımsız bir mikro API servisi.

2.3. Veritabanı Şeması ve İlişkisel Yapı (ER Model)

Projede veri bütünlüğünü ve güvenliğini sağlamak için SQL Server kullanılarak güçlü ilişkisel bağlar (Foreign Keys) kurulmuştur. Kimlik doğrulama işlemleri için `AspNetUsers` ve `AspNetRoles` tabloları temel alınmış olup, vatandaşların abonelikleri, ödedikleri faturalar ve oluşturdukları şikayet kayıtları bu tablolarla ilişkilendirilmiştir.

Aşağıdaki tablo, sistemin Entity-Relationship (ER) mimarisindeki ana varlıkları ve birbirleriyle olan bire-çok (1:N) ve bire-bir (1:1) ilişkilerini göstermektedir:

Ana Tablo	İlişki (Foreign Key)	Hedef Tablo
AspNetUsers	1 : N (Sahiptir)	Subscriptions
AspNetUsers	1 : N (Oluşturur)	Complaints
AspNetUsers	1 : N (Öder)	Bills
Subscriptions	1 : N (Üretir)	Bills
Subscriptions	1 : N (Kaydeder)	ConsumptionRecords
Complaints	N : 1 (Kategorisi)	ComplaintCategories
Complaints	N : 1 (Durumu)	ComplaintStatuses
Complaints	1 : 1 (Yanıtı)	ComplaintResponses
Bills	1 : 1 (Ödemesi)	Payments

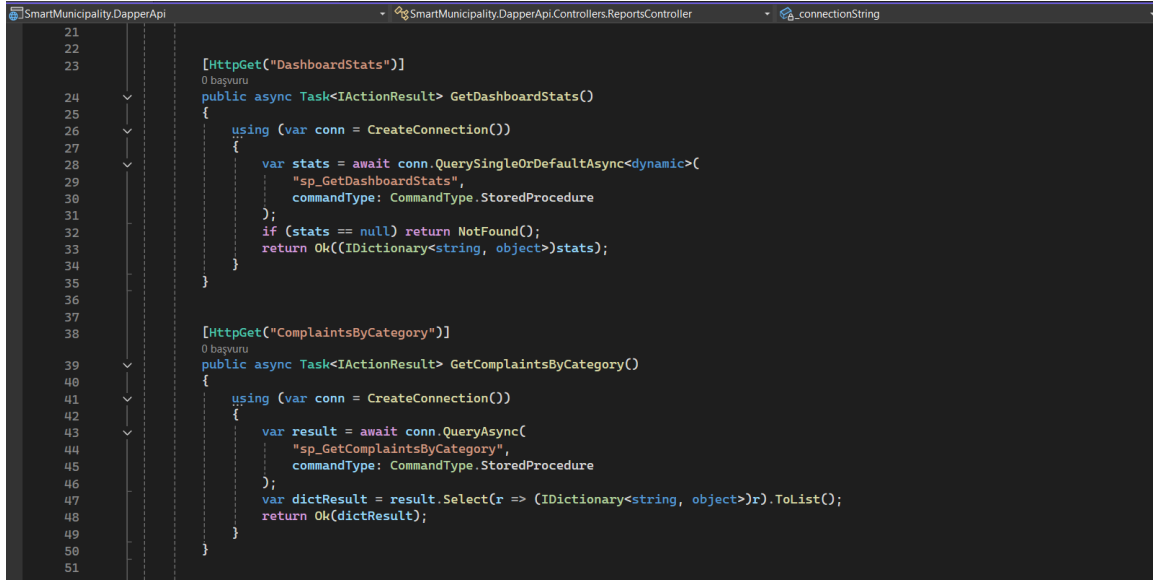
Sistemdeki Tablolar: `\_\_EFMigrationsHistory`, `AIAAlerts`, `Announcements`, `AspNetRoles`, `AspNetUsers`, `Bills`, `Complaints`, `ComplaintCategories`, `ComplaintResponses`, `ComplaintStatuses`, `ConsumptionRecords`, `Notifications`, `Payments`, `ServiceTypes`, `Subscriptions`.

2.4. Raporlama ve Gösterge Paneli Mimarisi

Sistemdeki karmaşık ve büyük hacimli veri raporlamalarında performans kaybını önlemek için Dapper ve SQL Stored Procedure (Saklı Yordam) yapıları tercih edilmiştir. Geliştirilen başlıca yordamlar:

- `sp\_GetDashboardStats`: Yönetici paneline ait özet sayısal istatistikleri sunar.
- `sp\_GetComplaintsByCategory` & `sp\_GetComplaintsByDistrict`: Kategori ve ilçe (bölge) bazlı şikayet dağılımlarını hesaplar.

- `sp\_GetMonthlyRevenue` & `sp\_GetYearlyRevenue`: Aylık ve yıllık tahsilat verilerini analiz grafiklerine uygun formatta getirir.
- `sp\_GetHeatMapData`: Harita üzerindeki (CBS) şikayet yoğunluğunu (HeatMap) hesaplamak için kullanılır.
- `sp\_GetComplaintsByStatus`: Sistemdeki açık/kapalı şikayet oranlarını analiz eder.



Şekil 2.1: Dapper Stored Procedure Entegrasyonu

Örnek Kod Kesiti (Dapper & Stored Procedure Entegrasyonu):

Aşağıdaki `ReportsController` kesiti, mikro ORM olan Dapper kullanılarak `sp\_GetDashboardStats` yordamının asenkron (async) ve yüksek performanslı bir biçimde nasıl çağrıldığını göstermektedir. Sistem, bu hibrit yapı sayesinde karmaşık SQL birleştirmelerini ve hesaplamalarını mikro saniyeler içinde çözümler:

```

[HttpGet("DashboardStats")]

public async Task<IActionResult> GetDashboardStats()

{

    using (var conn = CreateConnection())

    {

        var stats = await conn.QuerySingleOrDefaultAsync<dynamic>(

            "sp_GetDashboardStats",


```

```

        commandType: CommandType.StoredProcedure

    };

    if (stats == null) return NotFound();

    return Ok((IDictionary<string, object>)stats);

}

}

```

2.5. Yapay Zeka Destekli Asistan Servisi

Platform bünyesinde entegre olarak çalışan `AIService`, vatandaşların karşılaştığı problemleri anında çözmek için tasarlanmış bir akıllı asistan sunar. Su/doğalgaz abonelikleri, borç sorgulamaları veya şikayet süreçleri hakkındaki sorular doğal dil işleme (NLP) yetenekleriyle yanıtlanır.

Sistem, Gemini API ve OpenAI API kullanılarak dış zeka sistemleriyle entegredir. Ayrıca, dış servislere erişimin kısıtlı olduğu durumlar için önceden eğitilmiş, kural tabanlı bir yerel Fallback mekanizması devreye girerek kesintisiz hizmet sunar.

2.6. Sistem İzleme ve Loglama (Serilog)

Güvenilir ve izlenebilir bir backend altyapısı kurmak amacıyla sisteme Serilog entegre edilmiştir. Uygulamada meydana gelen olaylar, API istekleri ve veritabanı sorguları (Executed DbCommand) tarih bazlı günlük metin (txt) dosyalarına (Örn. `log20260706.txt`) aktarılır.

```

1  2026-07-18 02:17:41.859 +03:00 [INF] User profile is available. Using 'C:\Users\CASPER\AppData\Local\ASP.NET\
2  2026-07-18 02:18:01.711 +03:00 [INF] Executed DbCommand (35ms) [Parameters=[], CommandType='Text', Comm
3  SELECT 1
4  2026-07-18 02:18:01.807 +03:00 [INF] Executed DbCommand (27ms) [Parameters=[], CommandType='Text', Comm
5  SELECT OBJECT_ID(N'[_EFMigrationsHistory]');
6  2026-07-18 02:18:01.814 +03:00 [INF] Executed DbCommand (2ms) [Parameters=[], CommandType='Text', Comm
7  SELECT 1
8  2026-07-18 02:18:01.820 +03:00 [INF] Executed DbCommand (1ms) [Parameters=[], CommandType='Text', Comm
9  SELECT OBJECT_ID(N'[_EFMigrationsHistory]');
10 2026-07-18 02:18:01.896 +03:00 [INF] Executed DbCommand (12ms) [Parameters=[], CommandType='Text', Comm
11 SELECT [MigrationId], [ProductVersion]
12 FROM [_EFMigrationsHistory]
13 ORDER BY [MigrationId];
14 2026-07-18 02:18:01.955 +03:00 [INF] No migrations were applied. The database is already up to date.
15 2026-07-18 02:18:01.995 +03:00 [INF] Executed DbCommand (19ms) [Parameters=[], CommandType='Text', Comm
16
17 IF OBJECT_ID('sp_GetDashboardStats', 'P') IS NOT NULL DROP PROCEDURE sp_GetDashboardStats;
18 2026-07-18 02:18:02.021 +03:00 [INF] Executed DbCommand (13ms) [Parameters=[], CommandType='Text', Comm
19
20 CREATE PROCEDURE sp_GetDashboardStats
21 AS
22 BEGIN
23     SELECT
24         (SELECT COUNT(*) FROMAspNetUsers) AS TotalUsers,
25         (SELECT COUNT(*) FROM Complaints) AS TotalComplaints,
26         (SELECT COUNT(*) FROM Complaints WHERE StatusId = 1 OR StatusId = 2) AS PendingComplaints,
27         (SELECT COUNT(*) FROM Complaints WHERE StatusId = 3) AS ResolvedComplaints,
28         (SELECT COUNT(*) FROM Subscriptions WHERE IsActive = 1) AS ActiveSubscriptions,
29         (SELECT COUNT(*) FROM Subscriptions WHERE IsActive = 0) AS PassiveSubscriptions
30     END
31 2026-07-18 02:18:02.026 +03:00 [INF] Executed DbCommand (3ms) [Parameters=[], CommandType='Text', Comm
32
33 IF OBJECT_ID('sp_GetComplaintsByCategory', 'P') IS NOT NULL DROP PROCEDURE sp_GetComplaintsByCategory;
34 2026-07-18 02:18:02.039 +03:00 [INF] Executed DbCommand (10ms) [Parameters=[], CommandType='Text', Comm

```

Şekil 2.2: Serilog Detaylı Çıktısı

Aşağıdaki çıktıda görüldüğü üzere, Stored Procedure çağrılarının milisaniye cinsinden ne kadar sürede yürütüldüğü kayıt altına alınmaktadır:

```
2026-07-18 02:18:01.995 +03:00 [INF] Executed DbCommand (19ms)
[Parameters=[], CommandType='Text', ...]
```

```
2026-07-18 02:18:02.021 +03:00 [INF] Executed DbCommand (13ms)
[Parameters=[], CommandType='Text', ...]
```

```
CREATE PROCEDURE sp_GetDashboardStats AS BEGIN ...
```

```
2026-07-18 02:18:02.039 +03:00 [INF] Executed DbCommand (10ms)
[Parameters=[], CommandType='Text', ...]
```

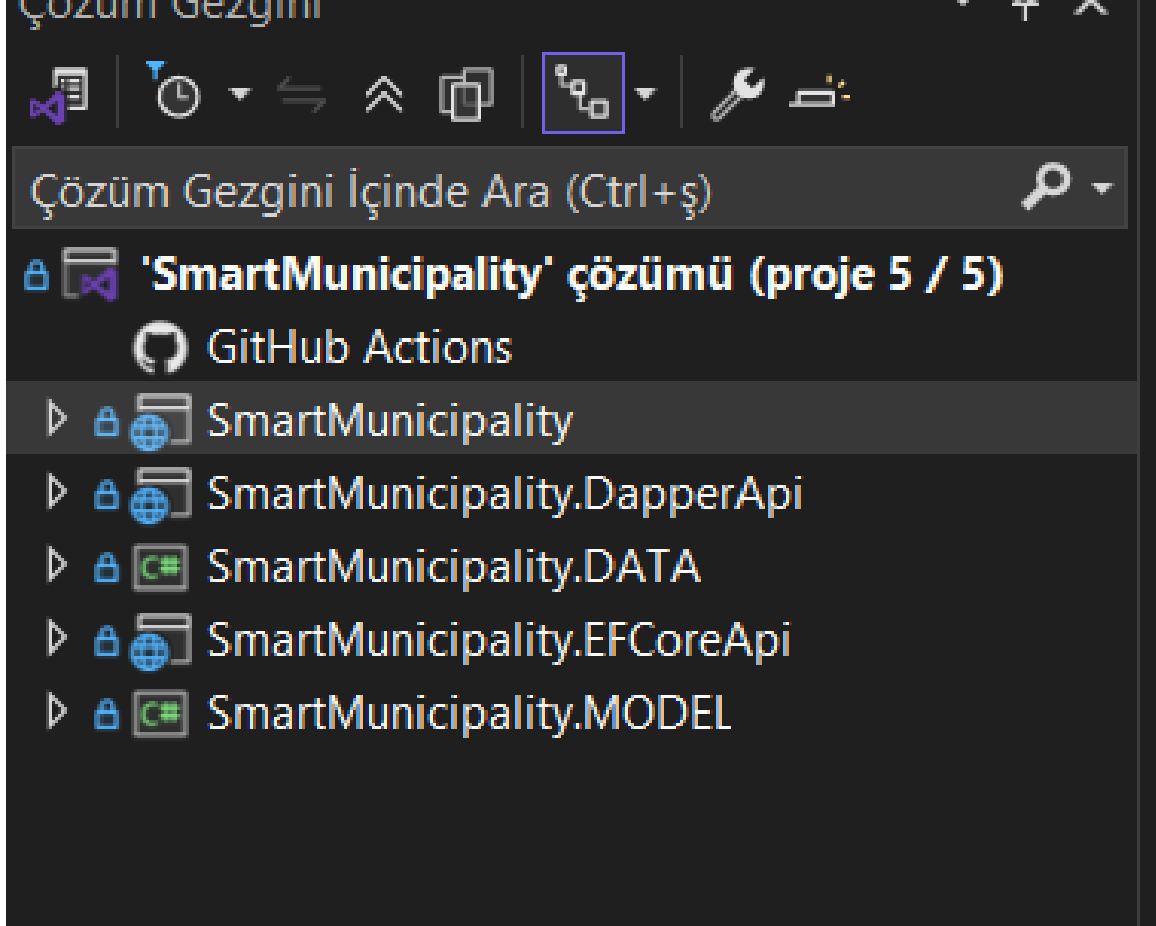
2.7. Önbellekleme (Caching) Altyapısı

Sistem genelinde performans artışı sağlamak, veritabanı üzerindeki yükü azaltmak ve sık erişilen verilerin hızlı getirilmesini garanti etmek amacıyla Önbellekleme (Caching) mekanizmaları kullanılmıştır. Özellikle vatandaşların sıkça görüntülediği genel istatistikler, il/ilçe verileri ve sabit şikayet kategorileri gibi görece durağan veriler MemoryCache veya dağıtık önbellek (Redis) mimarisi ile RAM üzerinde tutulmaktadır. Bu sayede veritabanına yapılan I/O (Girdi/Çıktı) maliyetleri minimize edilmiş ve milisaniyeler seviyesinde API yanıt süreleri elde edilmiştir.

3. KURULUM VE YAPILANDIRMA

3.1. Dosya Hiyerarşisi ve Çözüm Yapısı

Proje, modülerliği artırmak amacıyla Visual Studio ortamında birbirine referans veren 5 farklı projeden oluşan bir Solution (Çözüm) yapısında tasarlanmıştır.



Şekil 3.1: Ana Çözüm Gezgininde Hiyerarşisi

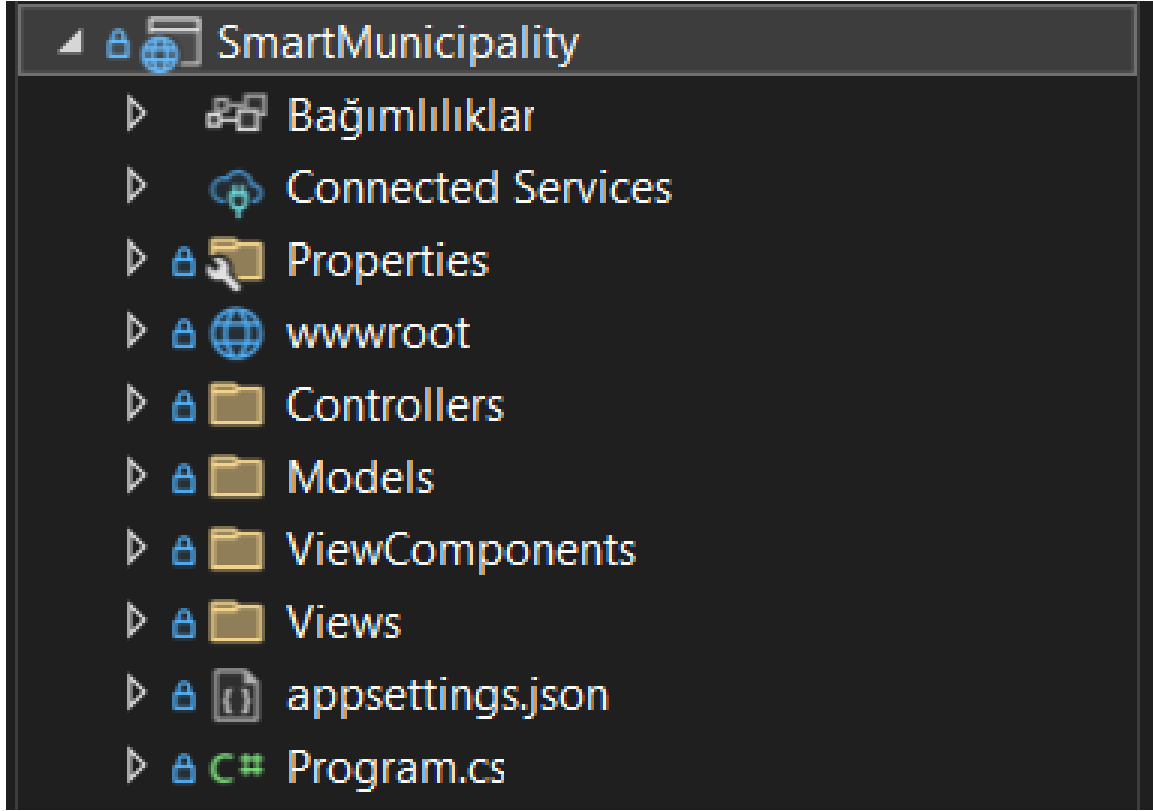
Hiyerarşide yer alan katmanlar ve barındırdıkları dizinlerin yapısal görevleri şöyledir:

1. SmartMunicipality (MVC):

Kullanıcıların ve operatörlerin etkileşime girdiği Frontend (Sunum) katmanıdır.

- 'Controllers': Web sayfalarının yönlendirilmesini sağlayan rotalar.

- `Models` / `ViewComponents`: Razor görünümüleriyle veri alışverişini ve modüler arayüz parçalarını sağlayan yapılar.
- `Views`: HTML/Razor sayfaları.
- `wwwroot`: Statik dosyalar (CSS, JS, ikonlar).



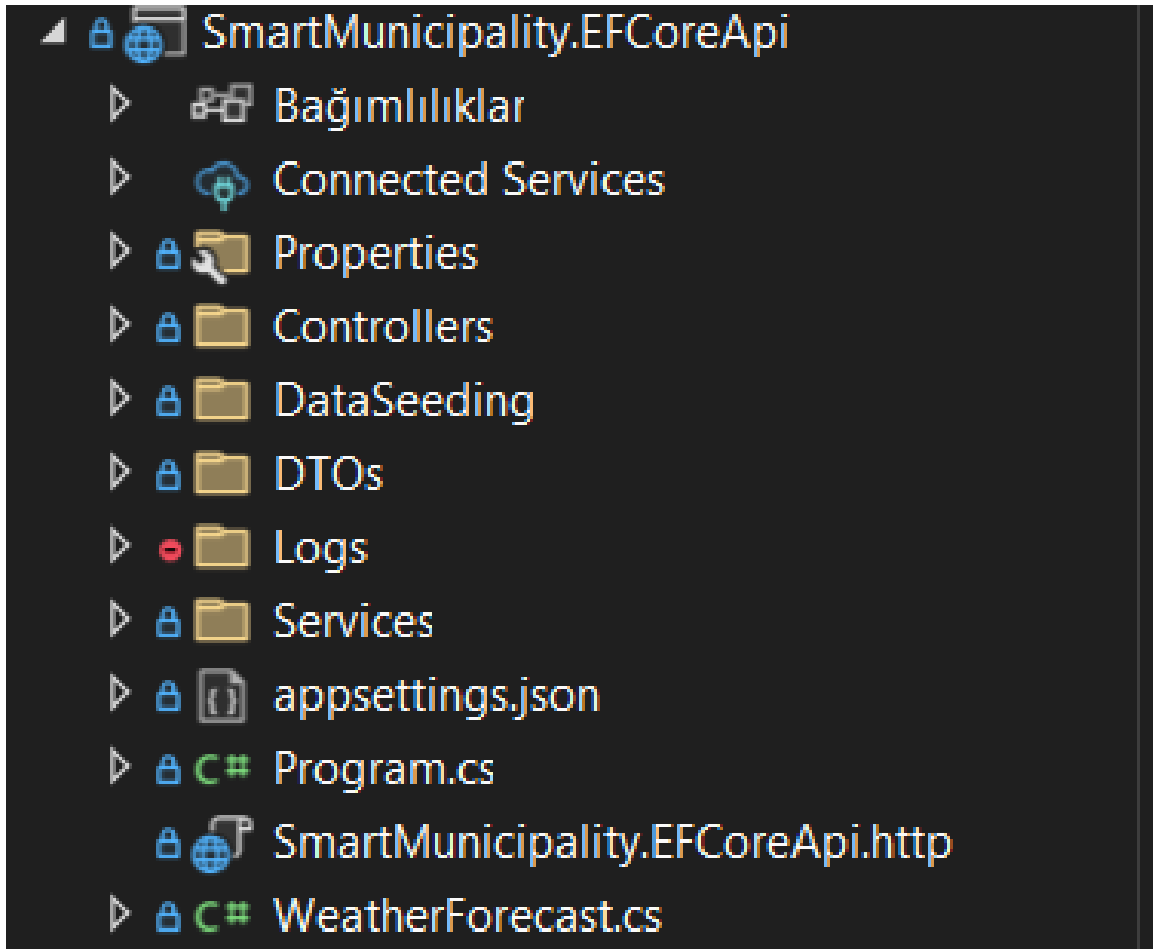
Şekil 3.2: SmartMunicipality (MVC) Katmanı

2. SmartMunicipality.EFCoreApi:

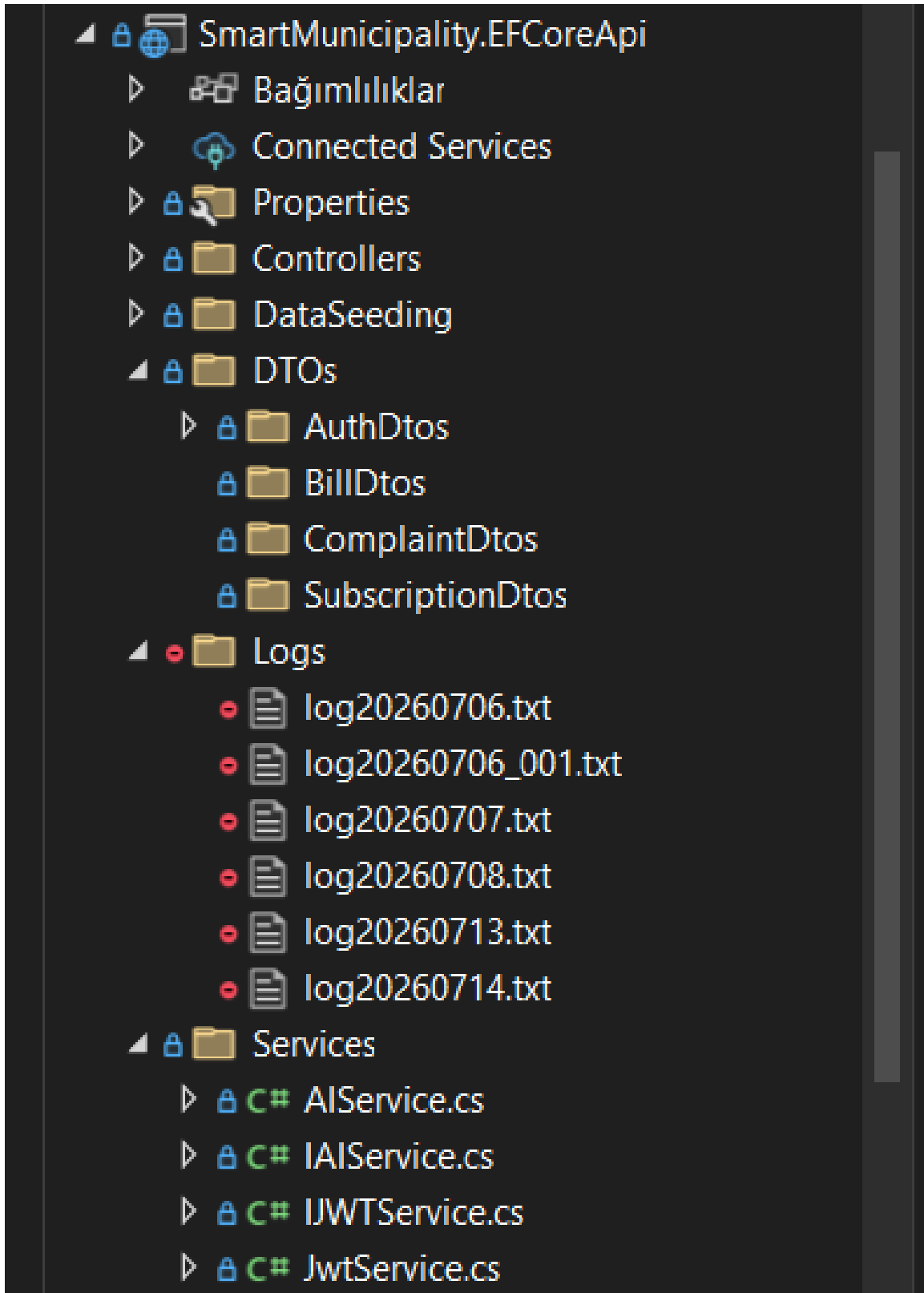
Ana CRUD işlemlerinin ve JWT güvenliğinin sağlandığı API servisi.

- `Controllers`: `AIChatbotController`, `AnnouncementsController`, `AuthController`, `BillController`, `ComplaintsController`, `ConsumptionController`, `DashboardController`, `NotificationsController`, `PaymentController`, `ProfileController`, `SubscriptionsController` ve `UserController` gibi ana uç noktaları (endpoints) barındırır.
- `DataSeeding`: Veritabanı ilk kez kurulduğunda varsayılan yönetici ve kategorileri ekleyen başlangıç tohumları.

- 'DTOs': İstemciyle API arası gereksiz veri taşımalarını engelleyen transfer nesneleri. Modül bazında sınıflandırılmıştır ('AuthDtos', 'BillDtos', 'ComplaintDtos', 'SubscriptionDtos').
- 'Logs': Serilog tarafından otomatik olarak tarih bazlı oluşturulan günlük hata ve izleme dosyalarının saklandığı dizin (ör. 'log20260706.txt', 'log20260713.txt' vb.).
- 'Services': Veri işleme mantıklarının ve dış entegrasyonların bulunduğu iş kuralları. Yapay Zeka servisi ('AIService', 'IAIService') ve kimlik doğrulama servisleri ('JwtService', 'IJWTService') bu katmandadır.



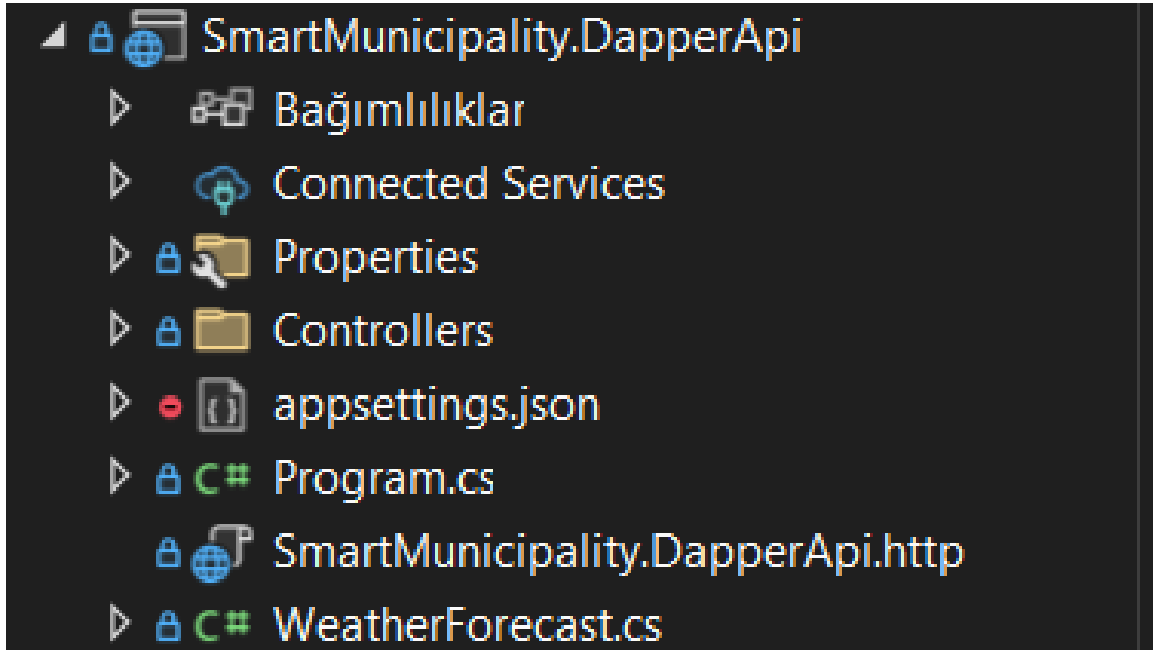
Şekil 3.3: EFCoreApi Katmanı



Şekil 3.4: EFCoreApi DTOs ve Logs Detayı

3. SmartMunicipality.DapperApi:

İstatistik ve raporlama işlemlerinden sorumlu yüksek performanslı bağımsız REST API servisidir. Ağırlıklı olarak Controller nesneleri üzerinden SQL Stored Procedure tetiklemeleri barındırır.

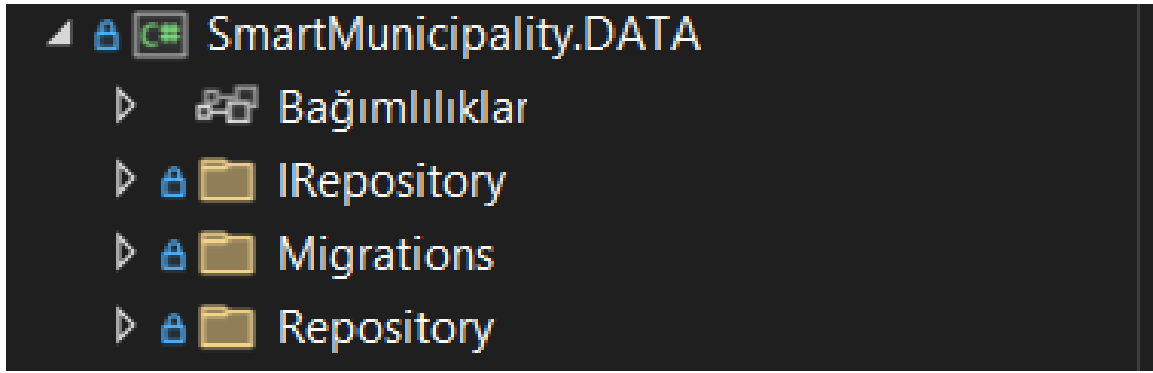


Şekil 3.5: DapperApi Katmanı

4. SmartMunicipality.DATA:

Veritabanı işlemlerini yürüten katmandır.

- 'IRepository' ve 'Repository': Veri erişim soyutlaması için kullanılan Repository Pattern sınıfları ve arayüzleri.
- 'Migrations': Entity Framework'ün veritabanı sürümleme (migration) ve şema dosyaları.

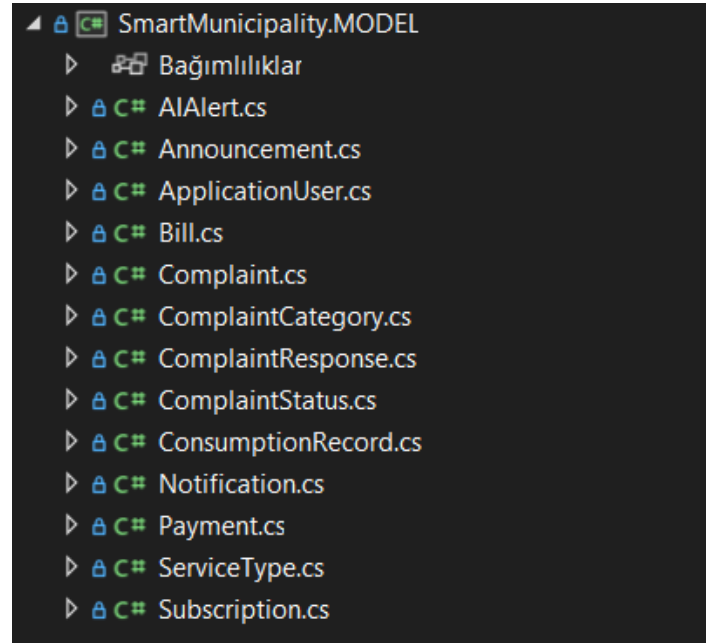


Şekil 3.6: DATA Katmanı

5. SmartMunicipality.MODEL:

Tüm bu projeler arasında dolaşan veri türlerini, Entity sınıflarını ve şemaları barındıran merkezi katmandır. Sistemdeki ana veritabanı tablolarını temsil eden sınıflar doğrudan burada yer alır:

- `ApplicationUser`, `Bill`, `Complaint`, `ComplaintCategory`, `ComplaintResponse`, `ComplaintStatus`, `ConsumptionRecord`, `Notification`, `Payment`, `Subscription`, `AIAAlert`, `Announcement`, `ServiceType` vb.



Şekil 3.7: MODEL Katmanı

3.2. Kurulum Adımları

Projeyi yerel geliştirme (development) ortamında çalıştırmak için aşağıdaki teknik adımlar izlenmelidir:

1. Projeyi Bilgisayara Klonlama:

```
``bash

git clone <repository-url>

cd SmartMunicipality

``
```

2. Veritabanı Konfigürasyonu: Entity Framework Core migrasyonları kullanılarak SQL Server üzerinde veritabanı şemalarının otomatik oluşturulması:

```
``bash

dotnet ef database update --project SmartMunicipality.DATA --startup-project SmartMunicipality.EFCoreApi

``
```

3. Servislerin Başlatılması (Multiple Startup): API servisleri ile web arayüzünün aynı anda haberleşebilmesi için, Visual Studio'da çözüm özelliklerinden (Solution Properties) "Multiple Startup Projects" ayarı seçilmelidir. Çalıştırılacak projeler:

- `SmartMunicipality.EFCoreApi`
- `SmartMunicipality.DapperApi`
- `SmartMunicipality` (Web Uygulaması)

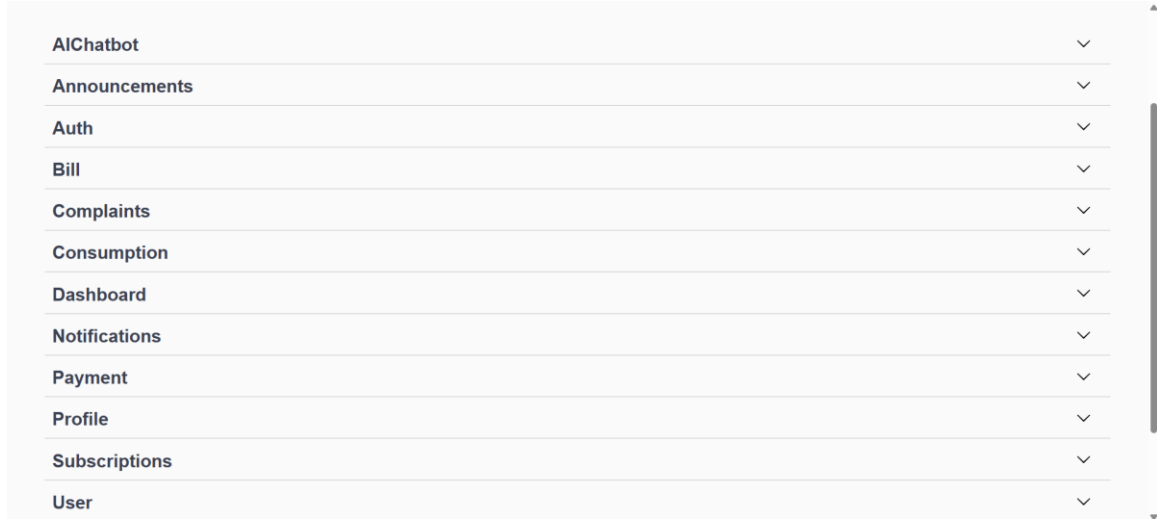
3.3. API Görünümleri (Swagger UI) ve Konsol Takibi

Projede geliştirilen API endpoint'lerinin dokümantasyonu ve arayüz üzerinden test edilebilmesi için Swagger UI entegre edilmiştir. Katmanlı yapı gereği EFCoreApi ve DapperApi projelerinin her biri kendi özel Swagger paneline sahiptir.

1. SmartMunicipality.EFCoreApi Swagger Arayüzü

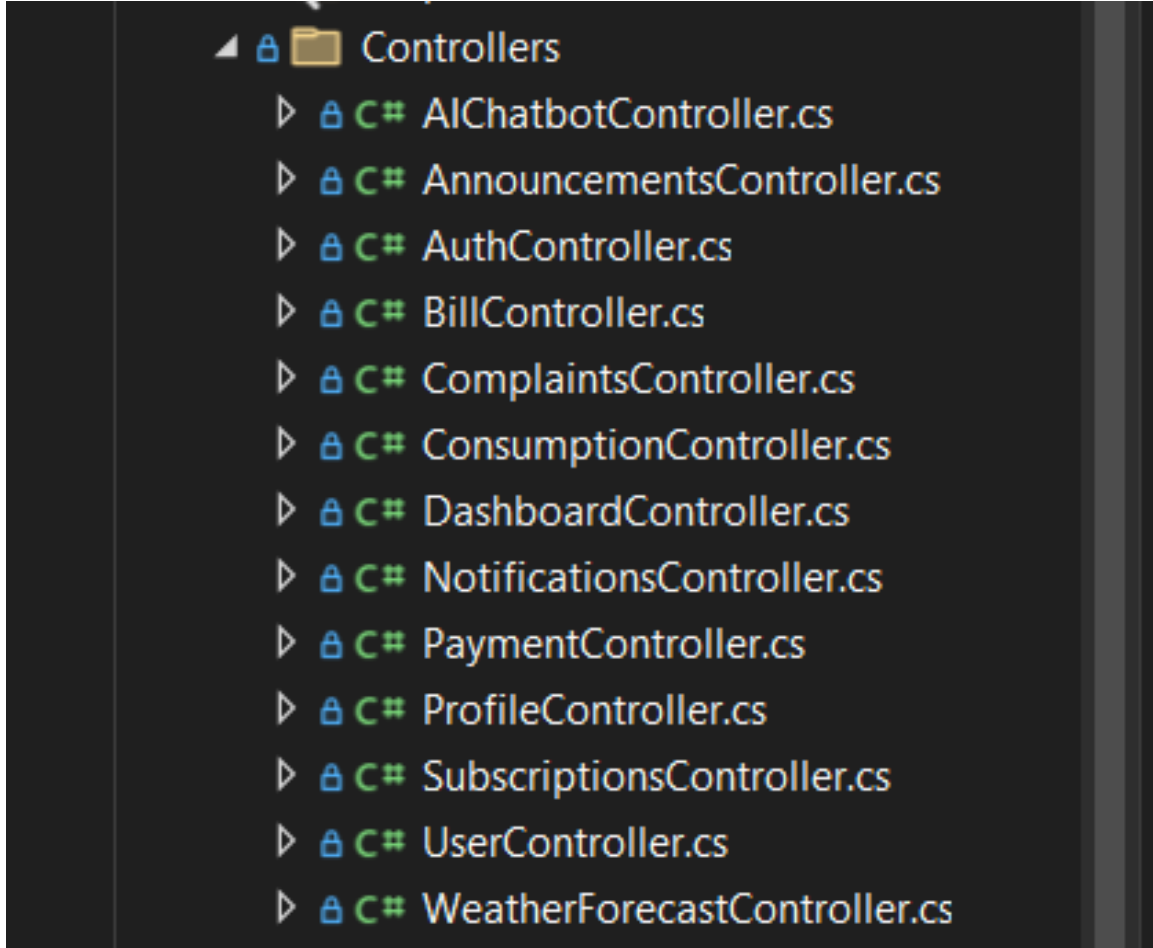
Temel iş mantığını ve veri tabanı kayıt işlemlerini yürüten EF Core API servisinde aşağıdaki ana modüller ve uç noktalar (endpoints) yer almaktadır:

- `AIChatbot`: Yapay zeka asistanı etkileşim uç noktaları.
- `Announcements`: Belediye duyuru ve haber yönetimi.
- `Auth`: Kullanıcı kayıt, giriş ve JWT token üretim işlemleri.
- `Bill` / `Consumption` / `Payment`: Fatura oluşturma, sayaç okuma ve sanal POS destekli güvenli ödeme işlemleri.
- `Complaints`: Şikayet bildirim sistemi ve durum güncellemeleri.
- `Dashboard` / `Notifications` / `Profile`: Yönetim paneli verileri, anlık bildirimler ve profil ayarları.
- `Subscriptions` / `User`: Yeni abonelik talepleri ve sisteme kayıtlı kullanıcı/personel listeleme.



AIChatbot	▼
Announcements	▼
Auth	▼
Bill	▼
Complaints	▼
Consumption	▼
Dashboard	▼
Notifications	▼
Payment	▼
Profile	▼
Subscriptions	▼
User	▼

Şekil 3.8: EF Core Api Görüntüsü



Şekil 3.9: EF Core API Controller Yapısı

2. SmartMunicipality.DapperApi Servisi (Konsol Çıktısı)

Yüksek performanslı raporlama ve analitik veri listeleme işlemlerini barındıran Dapper API servisi, arka planda güvenli portlar ('localhost:7201' ve 'localhost:5298') üzerinden asenkron dinleme yaparak MVC arayüzüne anlık istatistik akışı sağlar.

```
C:\Users\CASPER\source\repos > x + v - □ x
info: Microsoft.Hosting.Lifetime[14]
      Now listening on: https://localhost:7201
info: Microsoft.Hosting.Lifetime[14]
      Now listening on: http://localhost:5298
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
      Content root path: C:\Users\CASPER\source\repos\SmartMunicipality\Bitirme Projesi\SmartMunicipality.DapperApi
```

Şekil 3.10: Dapper API Konsol Çıktısı

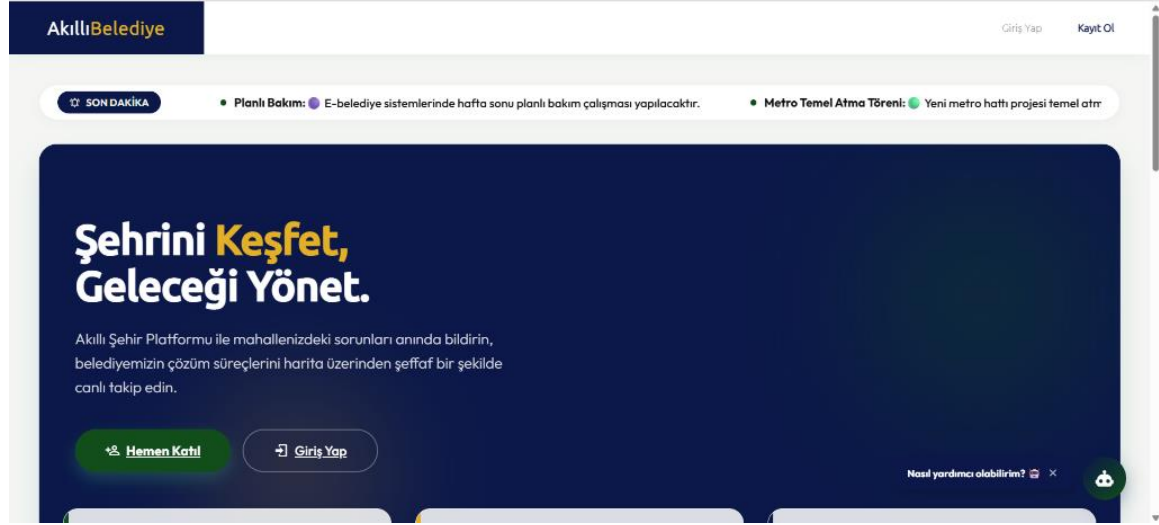
4. SİSTEM KULLANIMI VE EKRAN GÖRÜNTÜLERİ

> *Bu bölümde, platformu kullanan vatandaş ve operatörlerin izlediği adım adım süreçler (User Journey) ve arayüz ekran görüntüleri sunulmaktadır.*

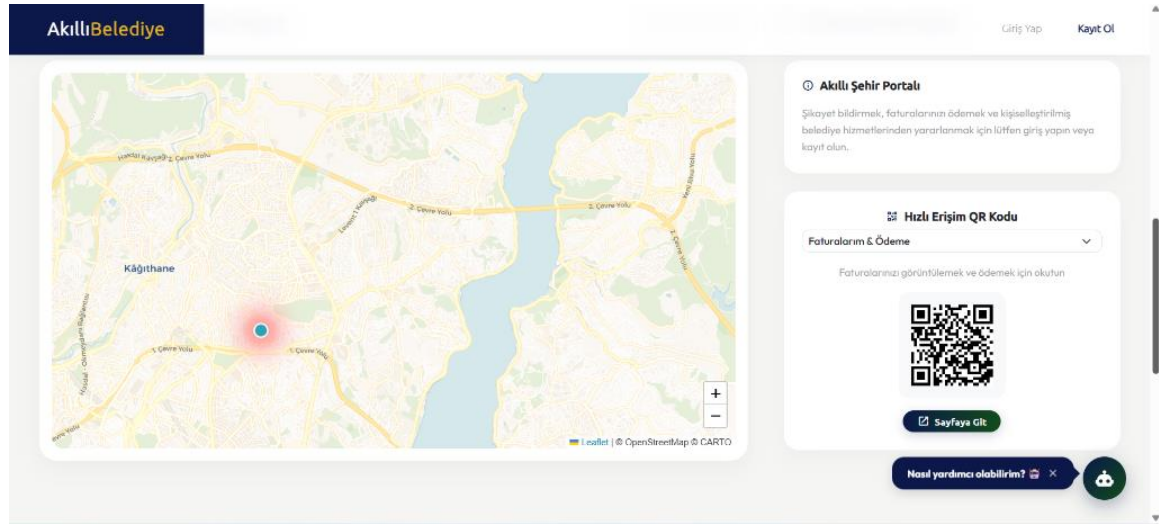
4.1. Aşama 1: Karşılama ve Kimlik Doğrulama

1. Belediye Portalı Ana Sayfası

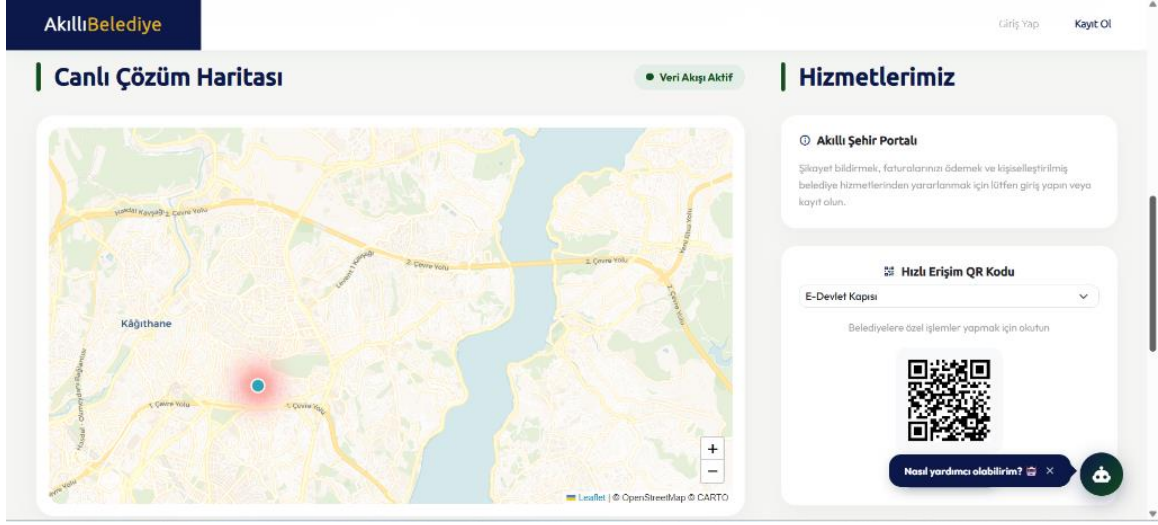
Sisteme erişildiğinde kullanıcıları karşılayan; belediyenin duyurularını, güncel haberlerini ve sisteme giriş bağlantılarını barındıran halka açık sayfadır.



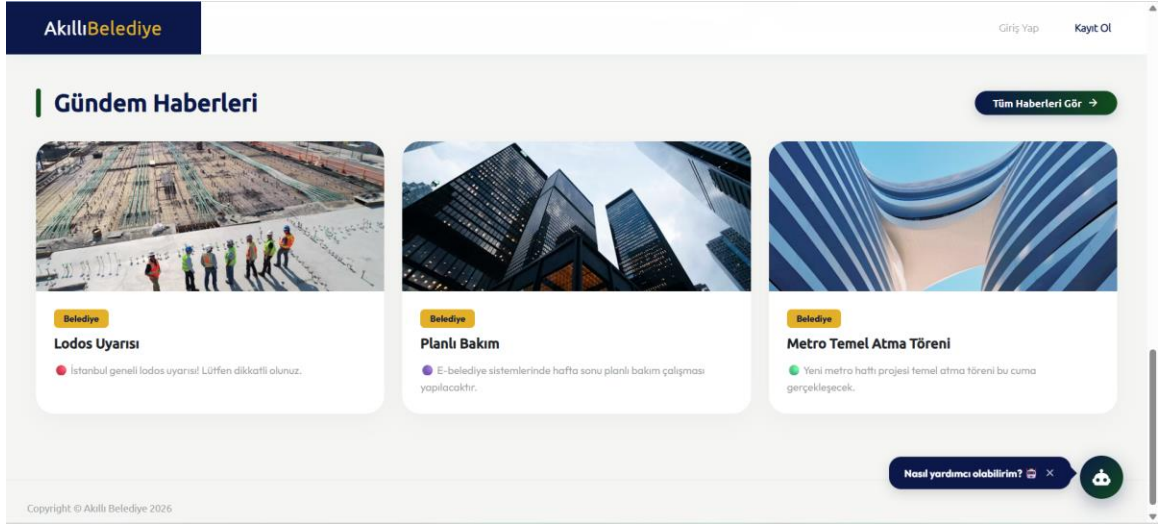
Şekil 4.1: Ana Sayfa İlk Kısım



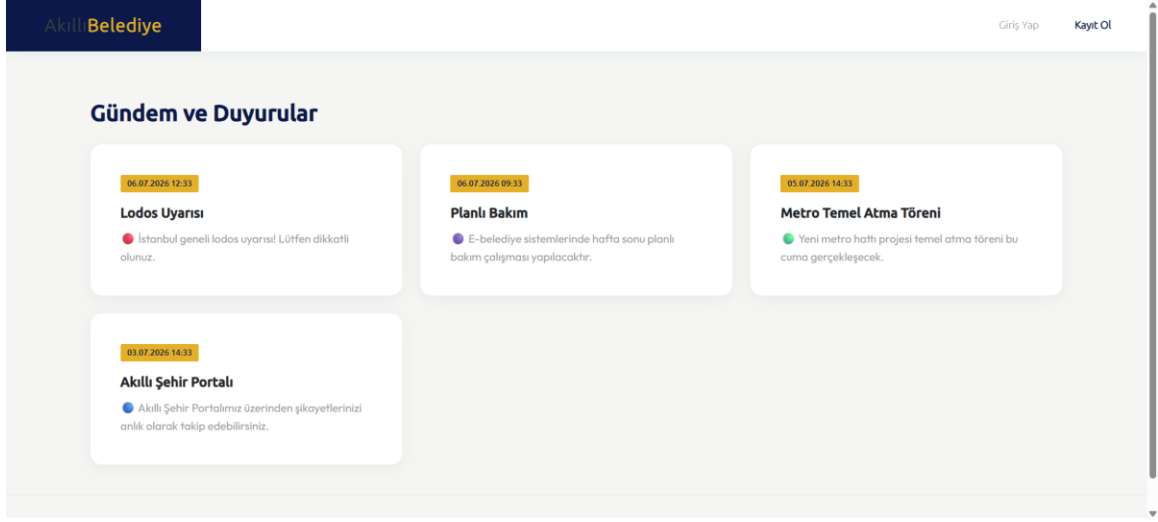
Şekil 4.2: Ana Sayfa Harita ve QR Kod Segmenti



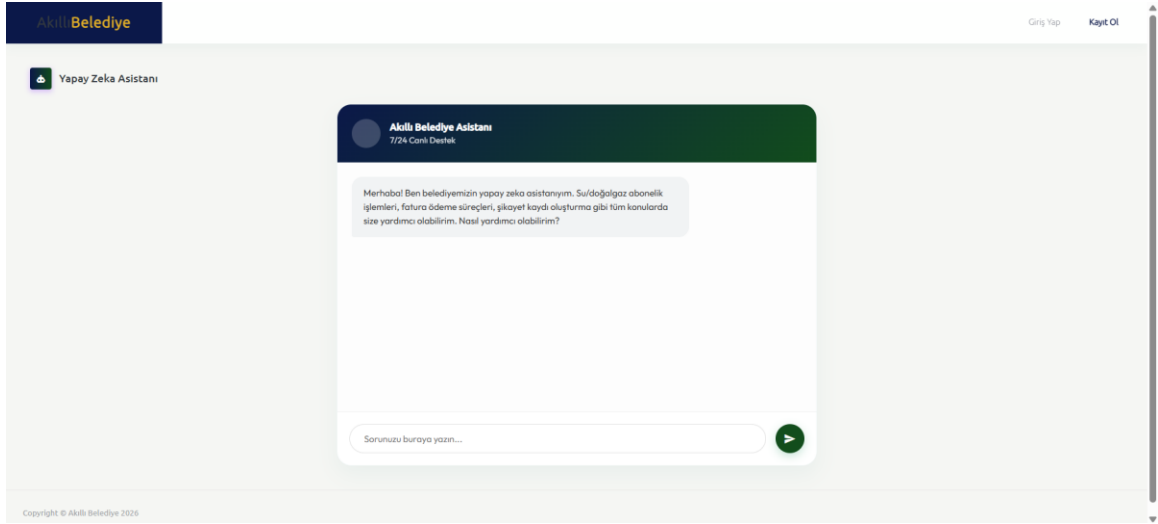
Şekil 4.3: Ana Sayfa Harita ve QR Kod Segmenti



Şekil 4.4: Ana Sayfa Haberler Kısmı



Şekil 4.5: Ana Sayfa Haberler Detay Kısmı



Şekil 4.6: Anasayfa Chatbot

2. Giriş Yap ve Kayıt Ol Ekranları

Güvenli, rol bazlı kimlik doğrulama altyapısına sahip kayıt ve giriş pencereleridir.

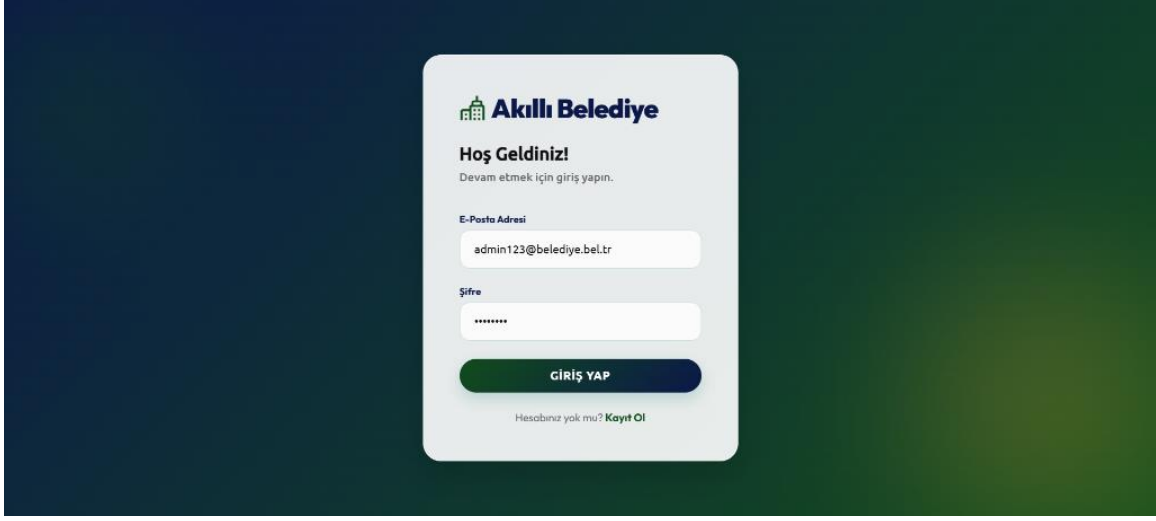
Vatandaş Hesap Oluşturma Ekranı

Şekil 4.7: Vatandaş Hesap Oluşturma

Vatandaş Giriş Paneli

Şekil 4.8: Vatandaş Giriş Paneli

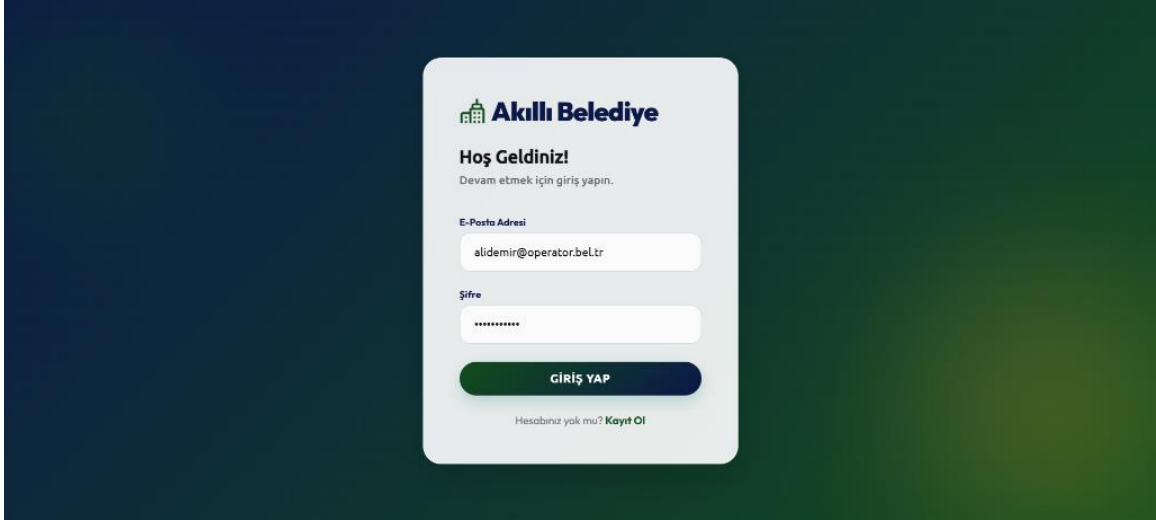
Yönetici Giriş Paneli (@belediye.bel.tr)



The screenshot shows a login page for the 'Akıllı Belediye' (Smart Municipality) system. The page has a dark blue background with a green gradient on the right. A white login card is centered. The card contains the 'Akıllı Belediye' logo, a welcome message 'Hoş Geldiniz!', and a prompt 'Devam etmek için giriş yapın.' (Log in to continue). Below this are two input fields: 'E-Posta Adresi' (Email Address) with the value 'admin123@belediye.bel.tr' and 'Şifre' (Password) with masked characters. A green 'GİRİŞ YAP' (Log In) button is below the password field. At the bottom of the card, there is a link 'Hesabınız yok mu? [Kayıt Ol](#)' (Don't have an account? [Sign Up](#)).

Şekil 4.9: Yönetici Giriş

Operatör Giriş Paneli (@operator.bel.tr)



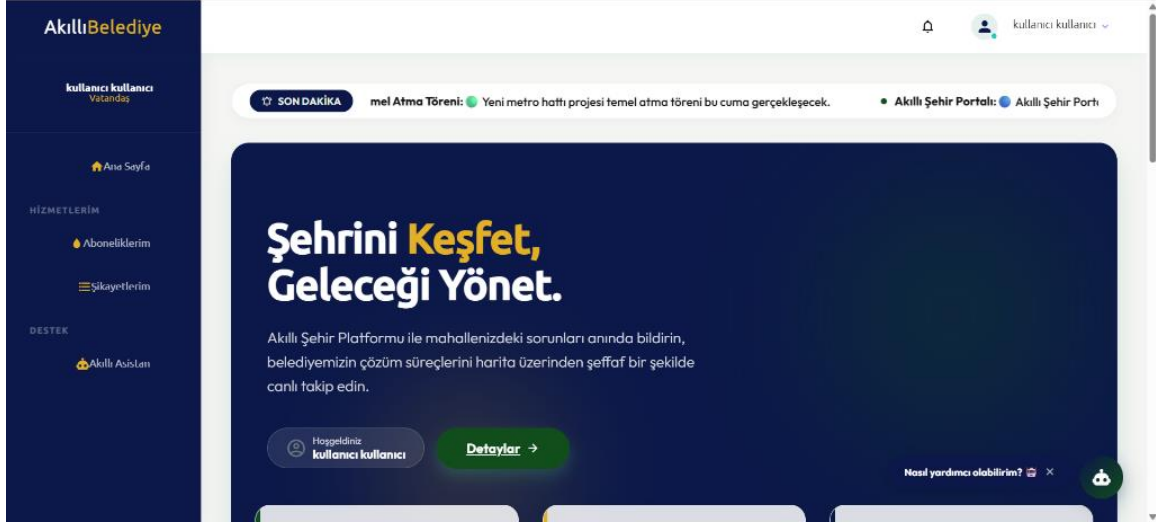
The screenshot shows a login page for the 'Akıllı Belediye' (Smart Municipality) system, specifically for the operator role. The page has a dark blue background with a green gradient on the right. A white login card is centered. The card contains the 'Akıllı Belediye' logo, a welcome message 'Hoş Geldiniz!', and a prompt 'Devam etmek için giriş yapın.' (Log in to continue). Below this are two input fields: 'E-Posta Adresi' (Email Address) with the value 'alidemir@operator.bel.tr' and 'Şifre' (Password) with masked characters. A green 'GİRİŞ YAP' (Log In) button is below the password field. At the bottom of the card, there is a link 'Hesabınız yok mu? [Kayıt Ol](#)' (Don't have an account? [Sign Up](#)).

Şekil 4.10: Operatör Giriş

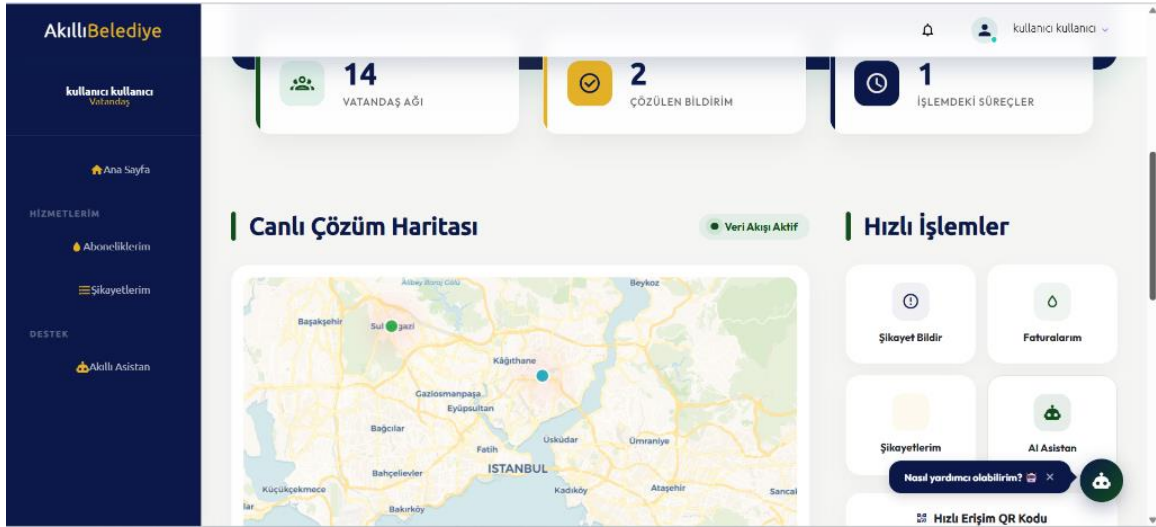
4.2. Aşama 2: Vatandaş Portal Akışı

1. Vatandaş Yönetim Paneli (Dashboard)

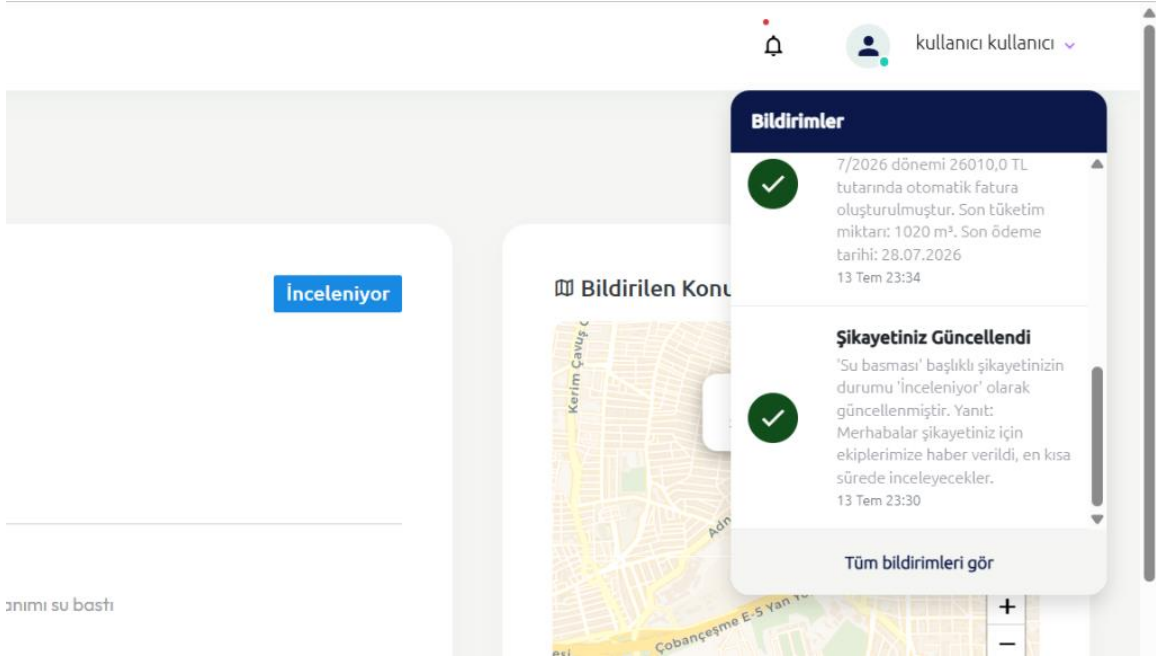
Vatandaşın başarılı giriş sonrası eriştiği; kendi aktif abonelik özetlerini, borç durumunu ve güncel bildirimleri (toast mesajları) takip edebildiği ana kontrol merkezidir.



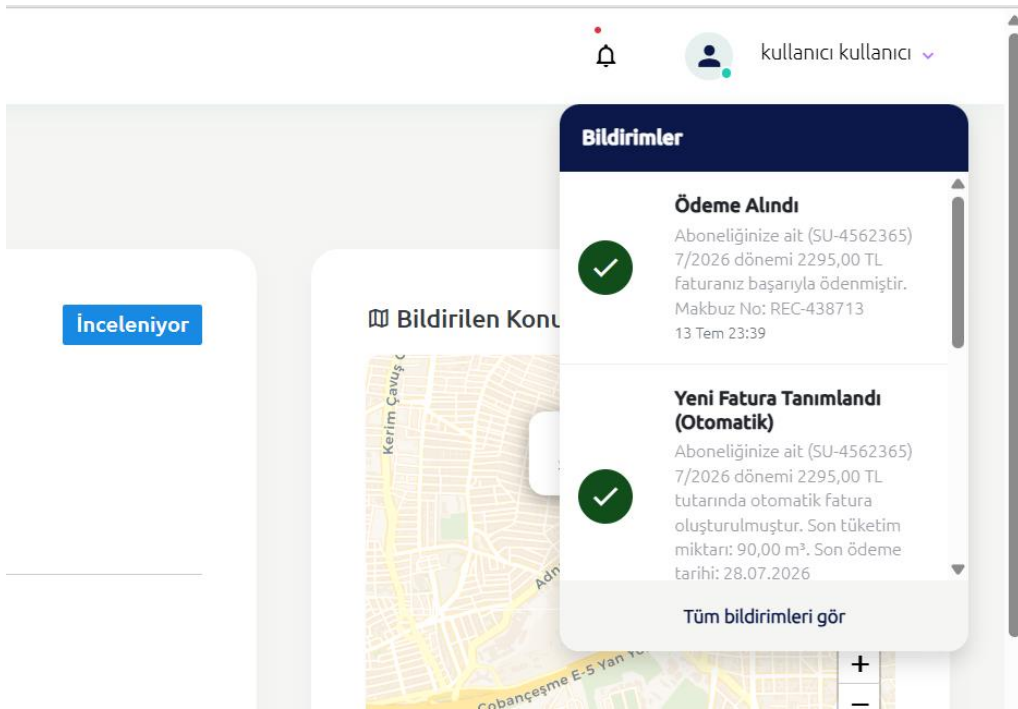
Şekil 4.11: Vatandaş Karşılama



Şekil 4.12: Hızlı İşlemler



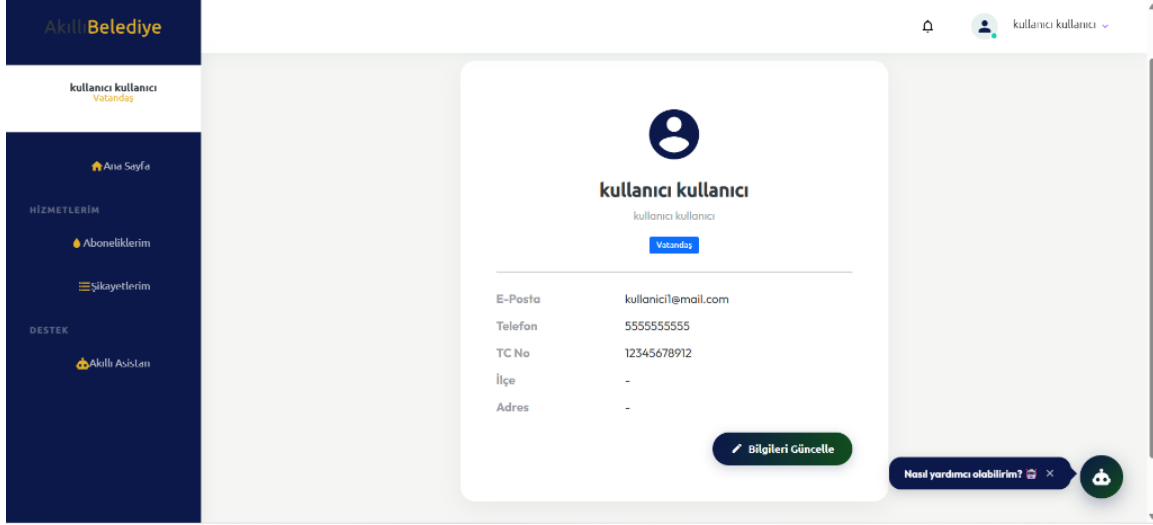
Şekil 4.13: Bildirim Paneli



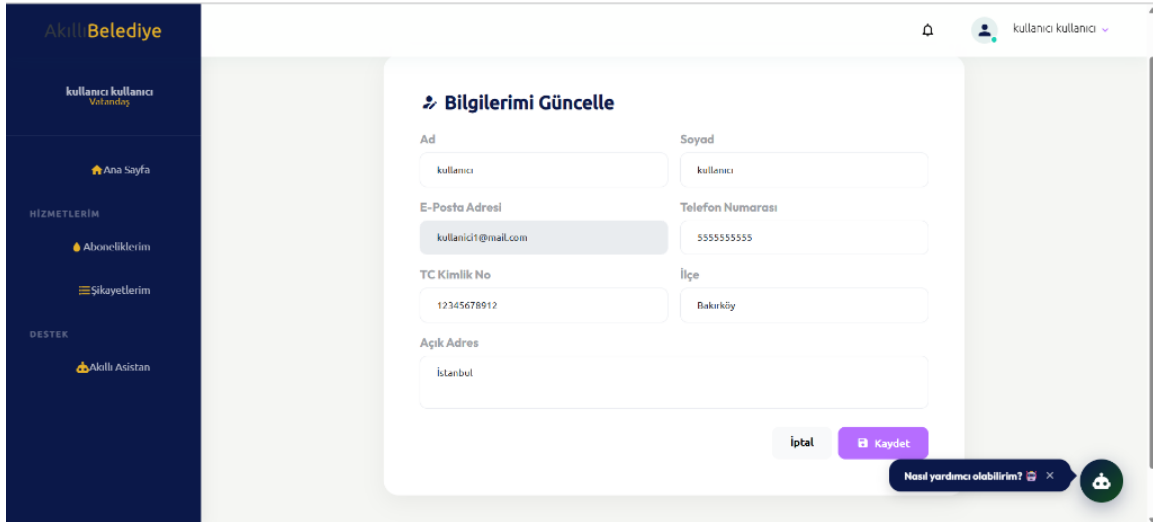
Şekil 4.14: Fatura Bildirimleri

2. Kişisel Profil Yönetimi

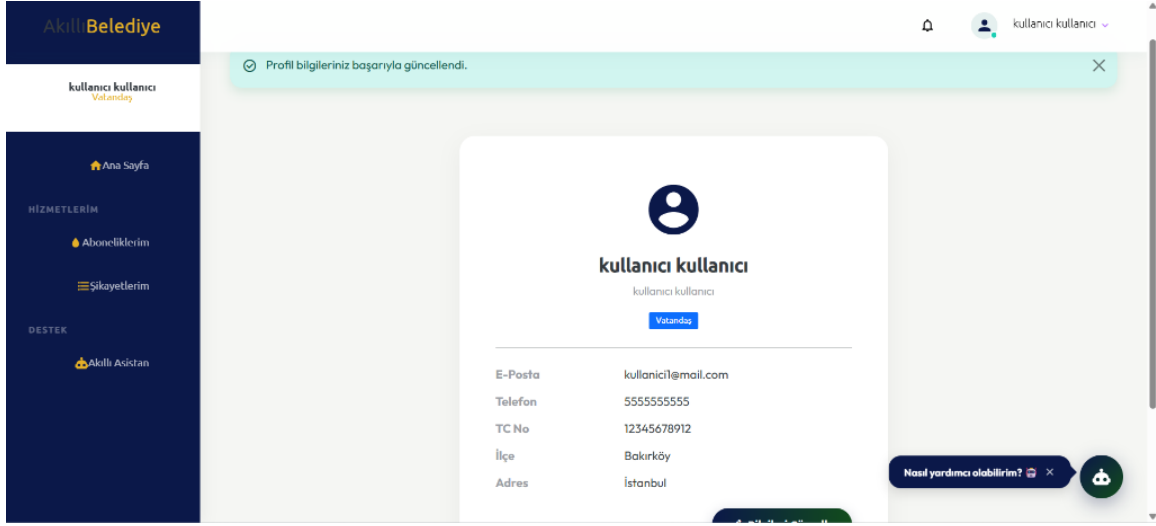
Vatandaşın sistemdeki kayıtlı bilgilerini (TC Kimlik No, İletişim Bilgileri, Adres) güncel tutabildiği profil alanıdır.



Şekil 4.15: Profil Görüntüleme



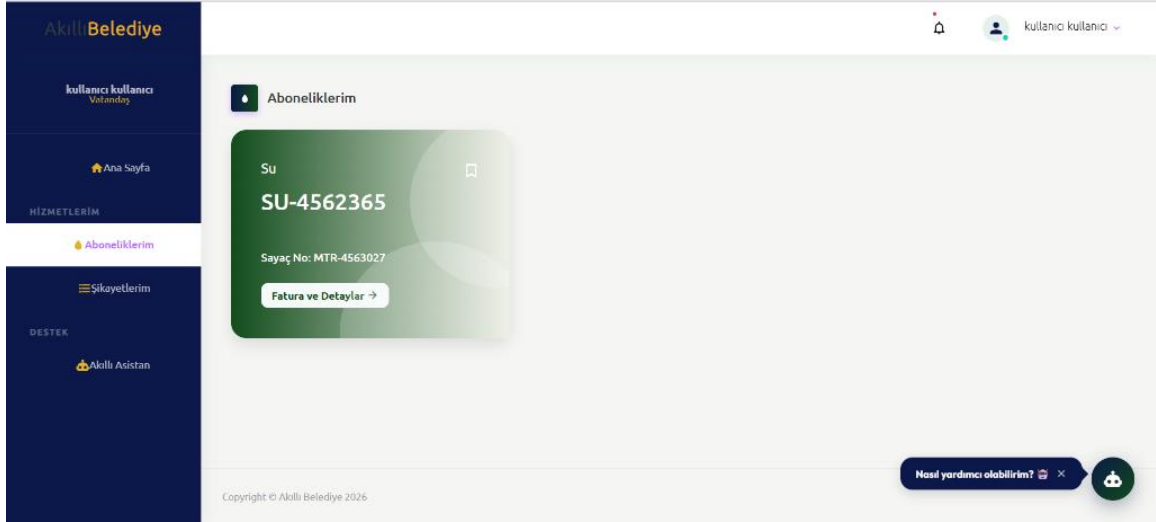
Şekil 4.16: Profil Güncelleme



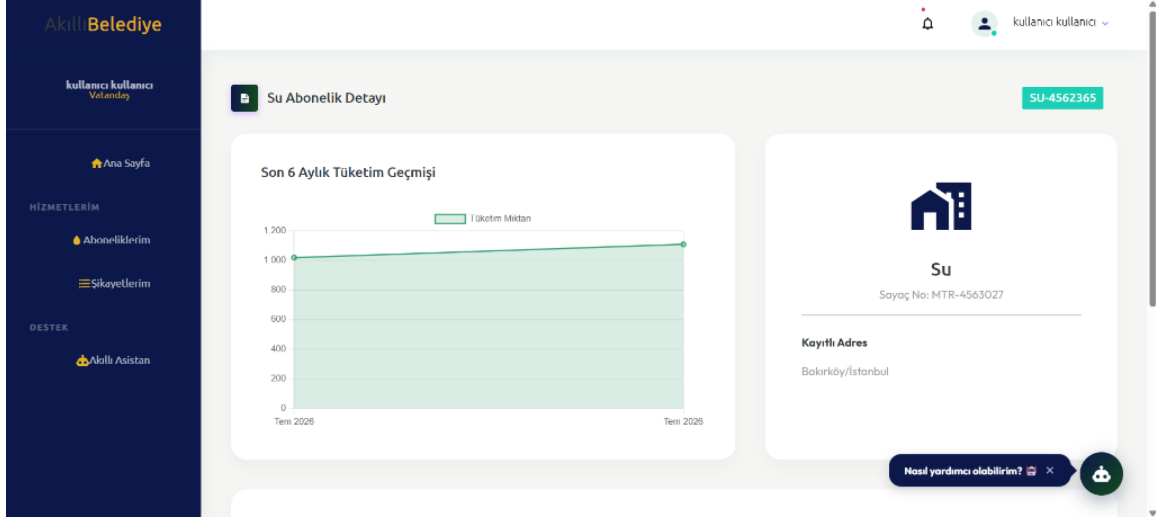
Şekil 4.17: Profil Başarı

3. Abonelik Başvurusu ve Yönetimi

Vatandaşların yeni bir hizmet (Su, Doğalgaz vb.) talebinde bulunduğu ve mevcut aboneliklerine ait geriye dönük kullanım trendlerini grafiksel olarak izlediği modüldür.



Şekil 4.18: Aktif Abonelikler



Şekil 4.19: Tüketim Grafiği

4. Fatura Sorgulama ve Güvenli Ödeme

Tüm faturaların şeffaf biçimde listelendiği, sanal POS arayüzü ile 3D Secure benzeri bir deneyimle ödeme yapılabildiği ve ödeme sonrası PDF makbuz ya da Excel listesi alınabilen kritik hizmet sayfasıdır.

Faturalarım

Arama kelimesini yazın...

Tüm Alanlar

Dönem	Fatura No	Son Ödeme	Tutar	Durum	İşlemler
2026 / 7	FAT-202607-AUTO-3655	28.07.2026	2295,00 TL	Ödenmedi	Öde PDF Excel
2026 / 7	FAT-202607-AUTO-9375	28.07.2026	26010,00 TL	Ödenmedi	Öde PDF Excel

Nasıl yardımcı olabiliriz?

Copyright © Akıllı Belediye 2026

Şekil 4.20: Fatura Listesi

AkıllıBelediye

kullanıcı kullanıcı
Vatandaş

Ana Sayfa

HİZMETLERİM

Aboneliklerim

Şikayetlerim

DESTEK

Akıllı Asistan

Fatura Ödeme

Kart Bilgileri

Kart Üzerindeki İsim
Kullanıcı: Kullanıcı

Kart Numarası
1111 2222 3333 4444

Son Kullanma Tarihi
22/11

CVV
123

Ödeme Yöntemi
Kredi Kartı

Güvenli Ödeme Yap İptal

Fatura Özeti

Hizmet Türü: Su

Abone No: SU-4562365

Fatura No: FAT-202607-AUTO-3655

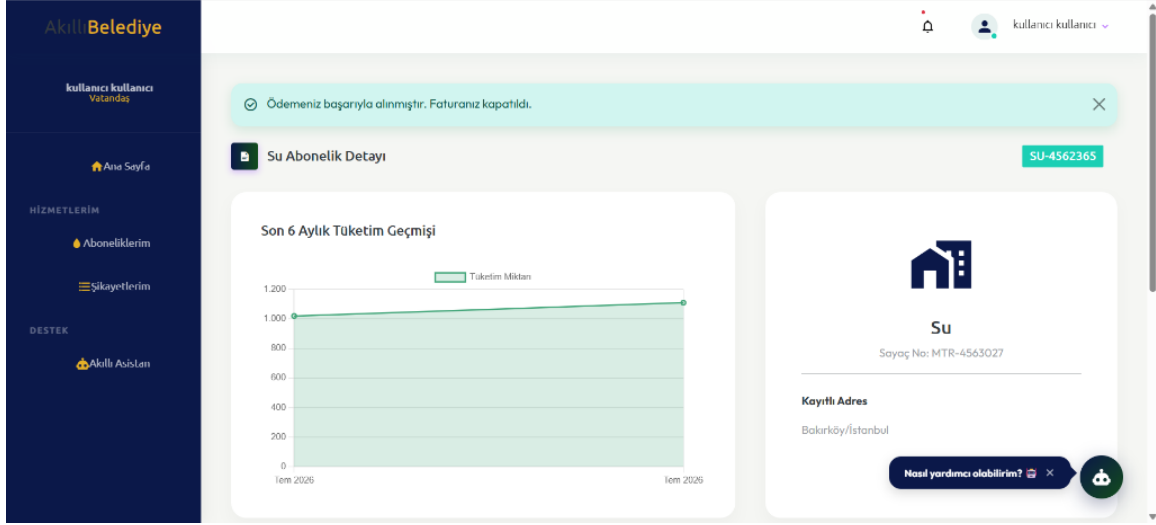
Dönem: 7 / 2026

Son Ödeme: 28.07.2026

Ödenecek Tutar: **2295,00 TL**

Nasıl yardımcı olabiliriz?

Şekil 4.21: Ödeme Formu



Şekil 4.22: Ödeme Başarılı

Faturalarım

Arama kelimesini yazın...

Tüm Alanlar

Dönem	Fatura No	Son Ödeme	Tutar	Durum	İşlemler
2026 / 7	FAT-202607-AUTO-3655	28.07.2026	2295,00 TL	Ödenmiş	PDF Excel
2026 / 7	FAT-202607-AUTO-9375	28.07.2026	26010,00 TL	Ödenmedi	Öde PDF Excel

Şekil 4.23: Güncel Fatura Listesi

SMART MUNICIPALITY FATURA

Sayın kullanıcı kullanıcı

Adres: Bakırköy/İstanbul

Hizmet Türü: Su
Abone No: SU-4562365
Fatura No: FAT-202607-AUTO-3655
Dönem: 7/2026
Son Ödeme: 28.07.2026
Durum: Ödendi
Tutar: 2295,00 TL

Şekil 4.24: PDF Çıktı

	A	B	C	D	E	F	G
1	SMART MUNICIPALITY - FATURA BİLGİLERİ						
2							
3	Abone No:	SU-4562365					
4	Fatura No:	FAT-202607-AUTO-3655					
5	Hizmet Türü:	Su					
6	Dönem:	7/2026					
7	Son Ödeme:	28.07.2026					
8	Tutar:	2295,00 TL					
9	Durum:	Ödendi					
10							

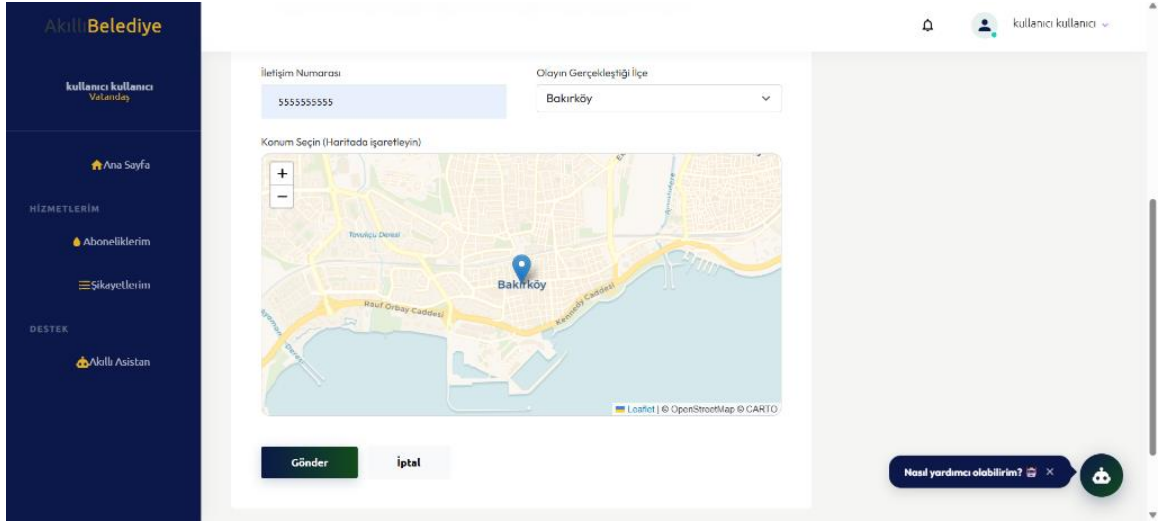
Şekil 4.25: Excel Çıktı

5. Konum Destekli Şikayet Bildirim Süreci

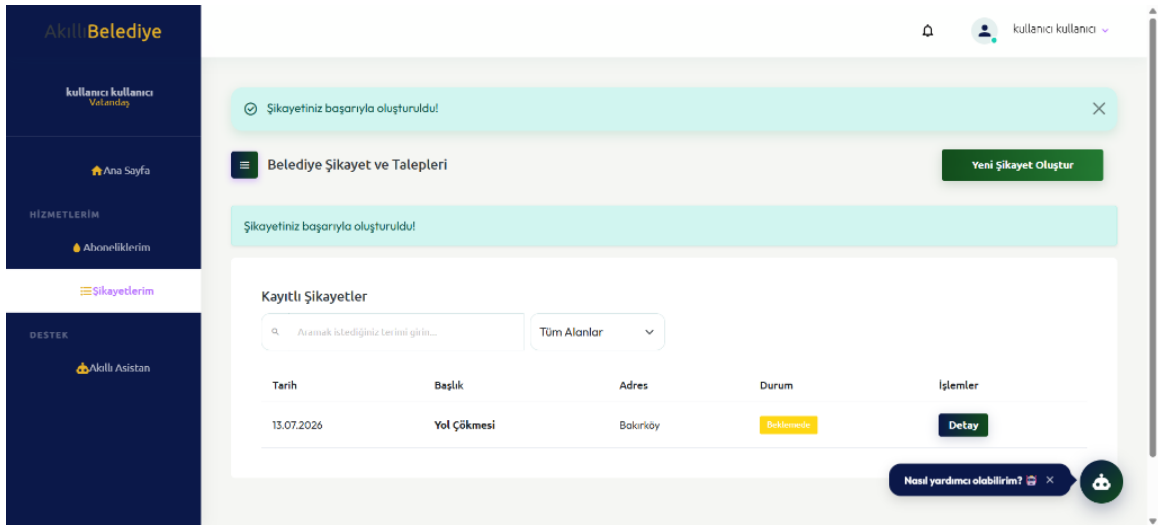
Vatandaşın çevresindeki bir sorunu harita üzerinden nokta atışı seçerek, resim ve açıklama ile belediyeye ilettiği; çözüm sürecini canlı olarak izlediği modüldür.

Şekil 4.26: Şikayet Listesi

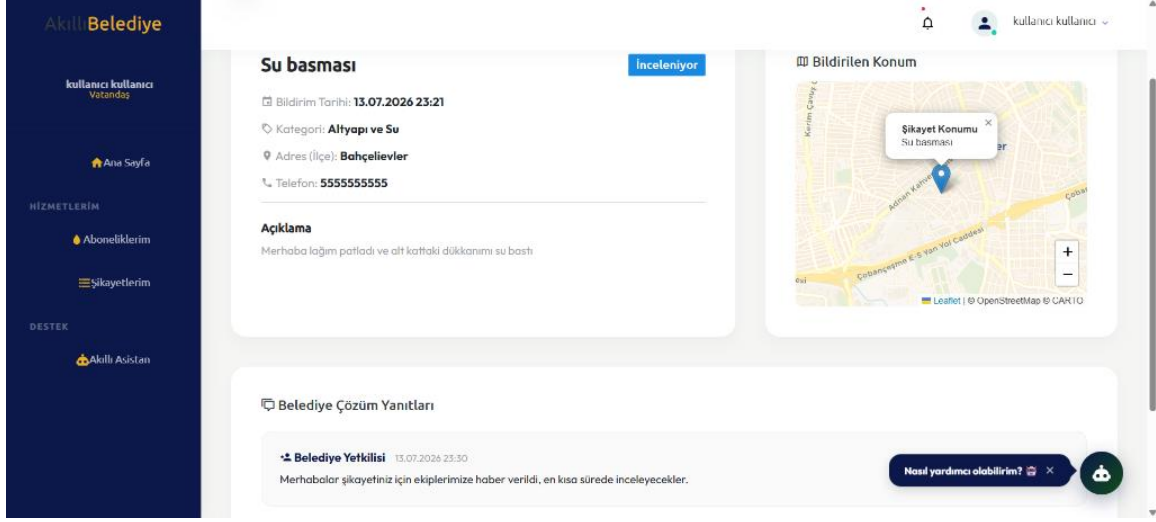
Şekil 4.27: Şikayet Formu



Şekil 4.28: Harita Seçimi



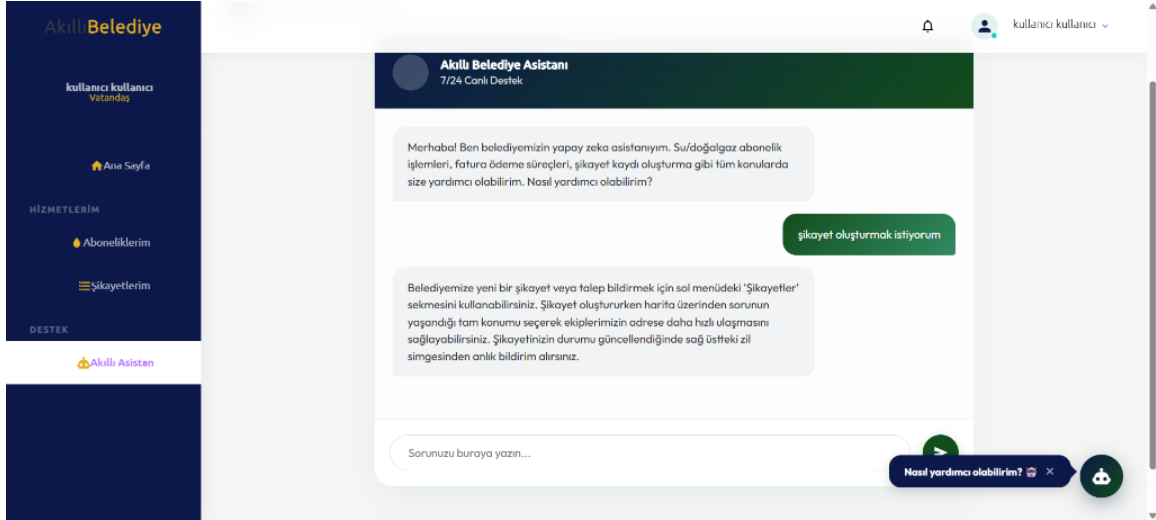
Şekil 4.29: Başarılı Kayıt



Şekil 4.30: Detay ve Yanıt

6. Yapay Zeka Destekli Belediye Asistanı (Chatbot)

Tüm sayfalardan erişilebilen yapay zeka asistanı ekranı.



Şekil 4.31: Chatbot

4.3. Aşama 3: Operatör ve Yönetici Paneli

1. Abonelik Onaylama ve Fatura Oluşturma

Belediye personelinin vatandaşlardan gelen talepleri incelediği, yeni birimler kaydettiği ve ay sonlarında sayaç değerlerini girerek otomatik borç tahakkuk ettirdiği operatör alanı.

Abone	Hizmet	Abone No	Sayaç No	Adres	Durum	İşlemler
Admin User	Doğalgaz	DG-544889	MTR-968012		Aktif	Sayaç Oku, İptal, Sil
Melis Yanık	Doğalgaz	DG-632844	MTR-751620		Aktif	Sayaç Oku, İptal, Sil
Melis Yanık	Doğalgaz	DG-274404	MTR-749383	Bekirköy, İstanbul	Aktif	Sayaç Oku, İptal, Sil
melis yanık	Doğalgaz	DG-990638	MTR-840718		Aktif	Sayaç Oku, İptal, Sil
Can Kaya	Doğalgaz	DG-636879	MTR-394871	Kadıköy, İstanbul	Aktif	Sayaç Oku, İptal, Sil

Şekil 4.32: Operatör Abonelikler

Abonelik Bilgileri

Kullanıcı (Vatandaş):

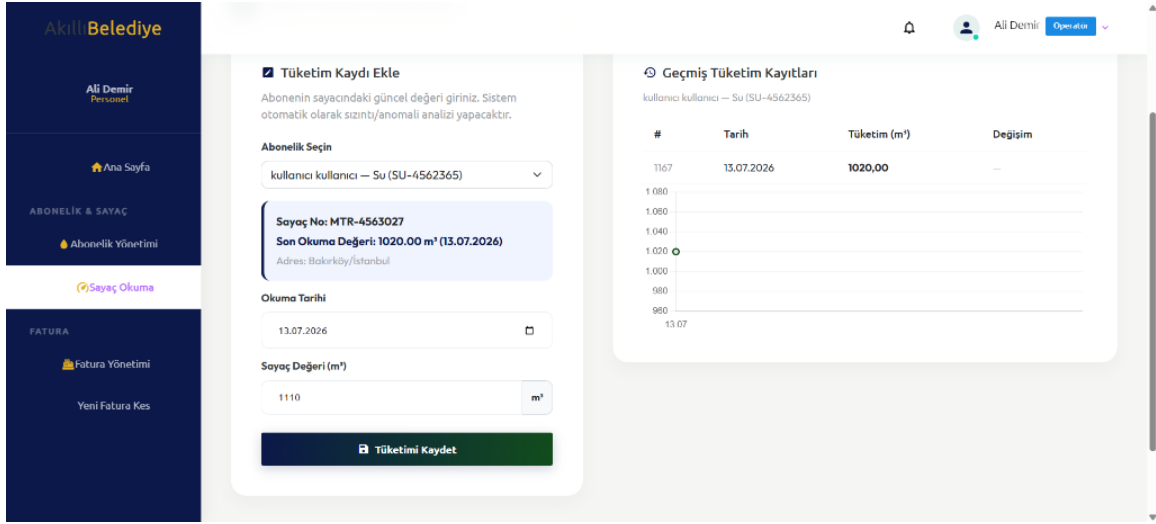
Hizmet Türü:

Abone Numarası:

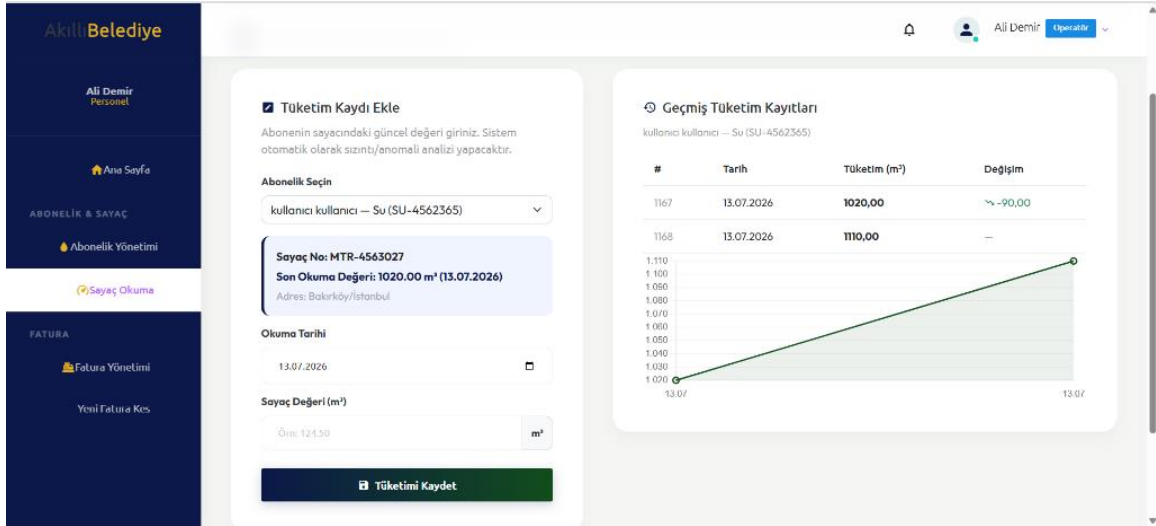
Sayaç Seri Numarası:

Hizmet Adresi:

Şekil 4.33: Abonelik Formu



Şekil 4.34: Sıyaç Okuma



Şekil 4.35: Sıyaç Başarılı

AkıllıBelediye

Ali Demir
Personel

Ana Sayfa

ABONELİK & SAYAÇ

Abonelik Yönetimi

Saygı Okuma

FATURA

Fatura Yönetimi

Yeni Fatura Kes

Fatura Yönetimi

Yeni Fatura Kes

Kesilen Faturalar

Aramak istediğiniz terimi girin...

Tüm Alanlar

Hizmet	Abone No	Fatura No	Dönem	Tutar	Son Ödeme	Durum
Su	SU-4562365	FAT-202607-AUTO-3655	7 / 2026	2295,00 TL	28.07.2026	Ödenmedi
Su	SU-4562365	FAT-202607-AUTO-9375	7 / 2026	26010,00 TL	28.07.2026	Ödenmedi
Emlak Vergisi	EV-234184	FAT-202607-EV-9647	7 / 2026	1675,00 TL	28.07.2026	Ödenmedi
Doğalgaz	DG-636879	FAT-202607-DG-7224	7 / 2026	909,91 TL	28.07.2026	Ödenmedi
Su	SU-619103	FAT-202607-SU-7419	7 / 2026	591,06 TL	28.07.2026	Ödenmedi
Emlak Vergisi	EV-885691	FAT-202607-3894	7 / 2026	1500,00 TL	23.07.2026	Ödenmedi

Şekil 4.36: Kesilen Faturalar

AkıllıBelediye

Ali Demir
Personel

Ana Sayfa

ABONELİK & SAYAÇ

Abonelik Yönetimi

Saygı Okuma

FATURA

Fatura Yönetimi

Yeni Fatura Kes

Fatura Yönetimi

Yeni Fatura Kes

Kesilen Faturalar

Aramak istediğiniz terimi girin...

Tüm Alanlar

Tüm Alanlar

Hizmet

Abone No

Fatura No

Dönem

Tutar

Son Ödeme

Durum

Su	SU-4562365	FAT-202607-AUTO-3655	7 / 2026	2295,00 TL	28.07.2026	Ödenmedi
Su	SU-4562365	FAT-202607-AUTO-9375	7 / 2026	26010,00 TL	28.07.2026	Ödenmedi
Emlak Vergisi	EV-234184	FAT-202607-EV-9647	7 / 2026	1675,00 TL	28.07.2026	Ödenmedi
Doğalgaz	DG-636879	FAT-202607-DG-7224	7 / 2026	909,91 TL	28.07.2026	Ödenmedi
Su	SU-619103	FAT-202607-SU-7419	7 / 2026	591,06 TL	28.07.2026	Ödenmedi
Emlak Vergisi	EV-885691	FAT-202607-3894	7 / 2026	1500,00 TL	23.07.2026	Ödenmedi

Şekil 4.37: Fatura Filtreleme Alanları

Fatura Yönetimi

Kesilen Faturalar

Ödendi Durum

Hizmet	Abone No	Fatura No	Dönem	Tutar	Son Ödeme	Durum
Su	SU-369587	FAT-202607-AUTO-4334	7 / 2026	2550,00 TL	23.07.2026	Ödendi
Doğalgaz	DG-636879	FAT-202606-DG-3224	6 / 2026	839,77 TL	28.06.2026	Ödendi
Su	SU-619103	FAT-202606-SU-5023	6 / 2026	511,11 TL	28.06.2026	Ödendi
Doğalgaz	DG-990638	FAT-202606-DG-8627	6 / 2026	741,92 TL	23.06.2026	Ödendi
Doğalgaz	DG-274404	FAT-202606-DG-2692	6 / 2026	890,31 TL	23.06.2026	Ödendi
Su	SU-764015	FAT-202606-SU-8780	6 / 2026	474,64 TL	23.06.2026	Ödendi

Şekil 4.38: Filtrelenmiş Faturalar

2. Şikayet Değerlendirme ve Yanıtlama

Bildirilen şikayetlerin durumunun (Beklemede, İnceleniyor, Çözüldü) güncellendiği ve vatandaşa resmi sonucun yazıldığı süreç yönetim paneli. Aynı zamanda yeni şikayet kategorileri de eklenebilmektedir.

Şikayet Yönetimi (Admin)

2 Beklemede 1 İnceleniyor 2 Çözüldü

Gelen Şikayetler

Şikayet ara... Tüm Durumlar

Tarih	Vatandaş Tel	Başlık	Adres	Durum	İşlem
13.07.2026 23:21	5555555555	Su basması	Bahçelievler	Beklemede	Yanıtla
13.07.2026 23:19	5555555555	Yol Çökmesi	Bakırköy	Beklemede	Yanıtla
08.07.2026 21:55	5624123696	sinirliyim Yanıt: merhaba sakinleştiriz	Sultangazi	Çözüldü	Yanıtla

Şekil 4.39: Şikayet Yönetimi

AkıllıBelediye

admin admin2
Yönetici

Ana Sayfa

YÖNETİM

- Admin Paneli
- Kullanıcı Yönetimi
- Duyuru Yönetimi
- Şikayet Kategorileri
- Raporlar & Analitik
- Yoğunluk Haritası

OPERASYONLAR

13.07.2026 23:21

Bahçelievler

5555555555

Açıklama

Merhaba lağım patladı ve alt kattaki dükkanımı su bastı.

İşlem Yap / Yanıtla

Durum Güncelle

İnceleniyor

Belediye Yanıt Metni

Merhabalar şikayetiniz için ekiplerimize haber verildi, en kısa sürede inceleyeceğiz.

Cevabı Gönder & Kaydet İptal

Olay Yeri

Bahçelievler

Leafler | © OpenStreetMap © CARTO

Şekil 4.40: Yanıt Formu

AkıllıBelediye

Kullanıcı Yönetimi

Duyuru Yönetimi

Şikayet Kategorileri

Raporlar & Analitik

Yoğunluk Haritası

OPERASYONLAR

Şikayet Yönetimi

Ahşonelik Yönetimi

Soyuş Okuma

Fatura Yönetimi

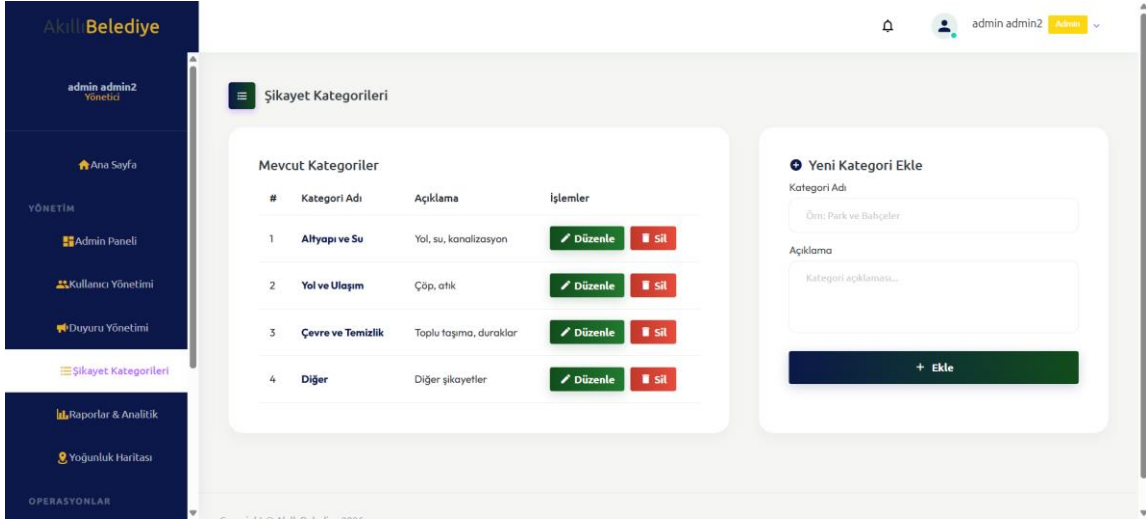
Gelen Şikayetler

Şikayet ara...

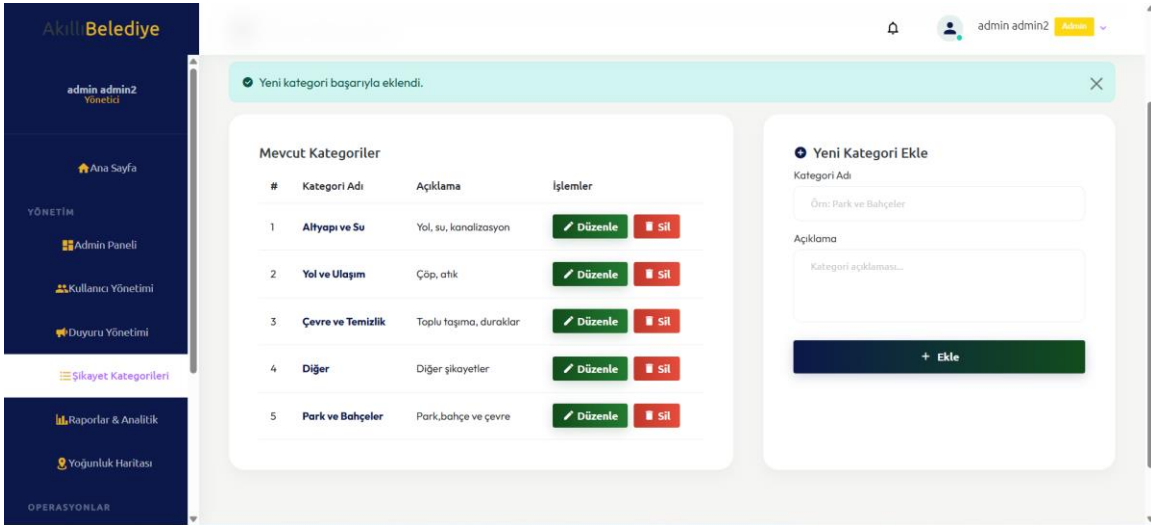
Tüm Durumlar

Tarih	Vatandaş Tel	Başlık	Adres	Durum	İşlem
13.07.2026 23:21	5555555555	Su basması Yanıt: Merhabalar şikayetiniz için ekiplerimize...	Bahçelievler	İnceleniyor	Yanıtla İptal
13.07.2026 23:19	5555555555	Yol Çökmesi	Bakırköy	Bekleniyor	Yanıtla İptal
08.07.2026 21:55	5624123696	sinirliyim Yanıt: merhaba sakinleştiniz	Sultangazi	Çözüldü	Yanıtla İptal
08.07.2026 18:16	5624123696	Su basması Yanıt: merhaba ekiplerimiz en kısa zamanda ...	Şişli	İnceleniyor	Yanıtla İptal
07.07.2026 15:44	5555555555	Sokak kirliliği Yanıt: oooo	Şile	Çözüldü	Yanıtla İptal

Şekil 4.41: Yanıt Başarılı



Şekil 4.42: Şikayet Kategorisi Ekleme



Şekil 4.43: Şikayet Kategorisi Ekleme Başarılı Mesajı

3. Sistem Ayarları: Kullanıcı ve Duyuru Yönetimi

Yöneticilerin sistemdeki kayıtlı kişileri gelişmiş filtrelerle (ilçe, rol) denetlediği ve ana sayfadaki kayan duyuruları (Marquee) anlık güncellediği bölüm.

The screenshot displays the 'Kullanıcı Yönetimi (Admin)' page. On the left is a dark blue sidebar with the 'AkıllıBelediye' logo and navigation links: 'admin admin2 Yönetici', 'Ana Sayfa', 'YÖNETİM' (containing 'Admin Paneli', 'Kullanıcı Yönetimi', 'Duyuru Yönetimi', 'Şikayet Kategorileri', 'Raporlar & Analitik', 'Yoğunluk Haritası'), and 'OPERASYONLAR'. The main content area has a header 'Kullanıcı Yönetimi (Admin)' and a section 'Kayıtlı Kullanıcılar'. Below this is a search bar with the placeholder 'Aramak istediğiniz terimi girin...' and a dropdown menu 'Tüm Alanlar'. The table below lists users with columns: Ad Soyad, E-Posta, Telefon, TC No, İlçe, Rol, and İşlemler. The table contains five rows of user data.

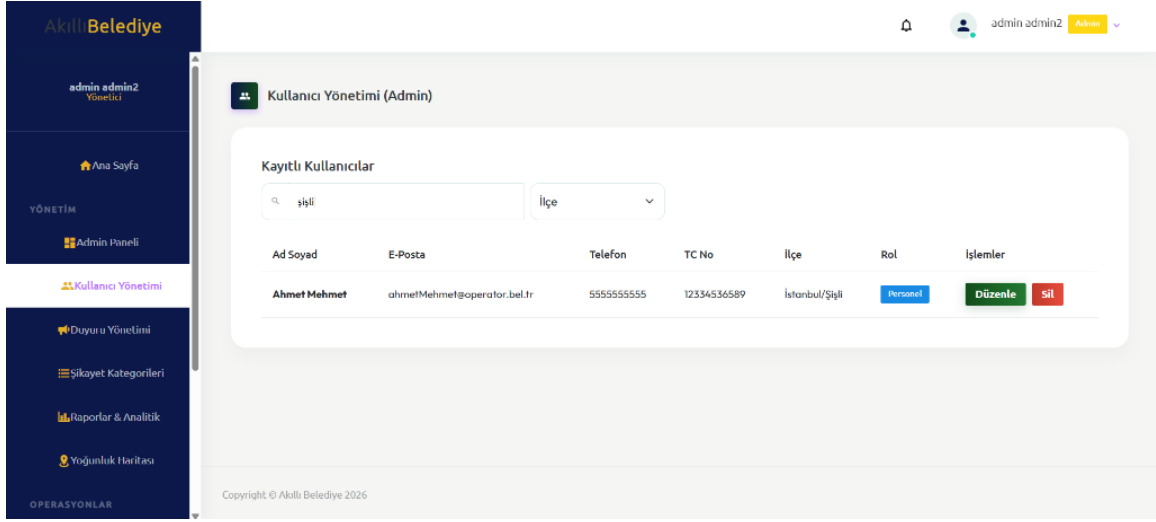
Ad Soyad	E-Posta	Telefon	TC No	İlçe	Rol	İşlemler
Ali Demir	alidemir@operator.bel.tr	5556321247	30156241289		Personel	Düzenle Sil
kullanıcı kullanıcı	kullanici@mail.com	5555555555	12345678912	Bakırköy	Yatandaş	Düzenle Sil
admin1 admin1	admin1@belediye.bel.tr	5555555555	12345678912		Admin	Düzenle Sil
admin admin2	admin123@belediye.bel.tr	5555555555	12396365412		Admin	Düzenle Sil
Test Admin	test@belediye.bel.tr	5551234567	12345678901		Admin	Düzenle Sil

Şekil 4.44: Kullanıcı Yönetimi

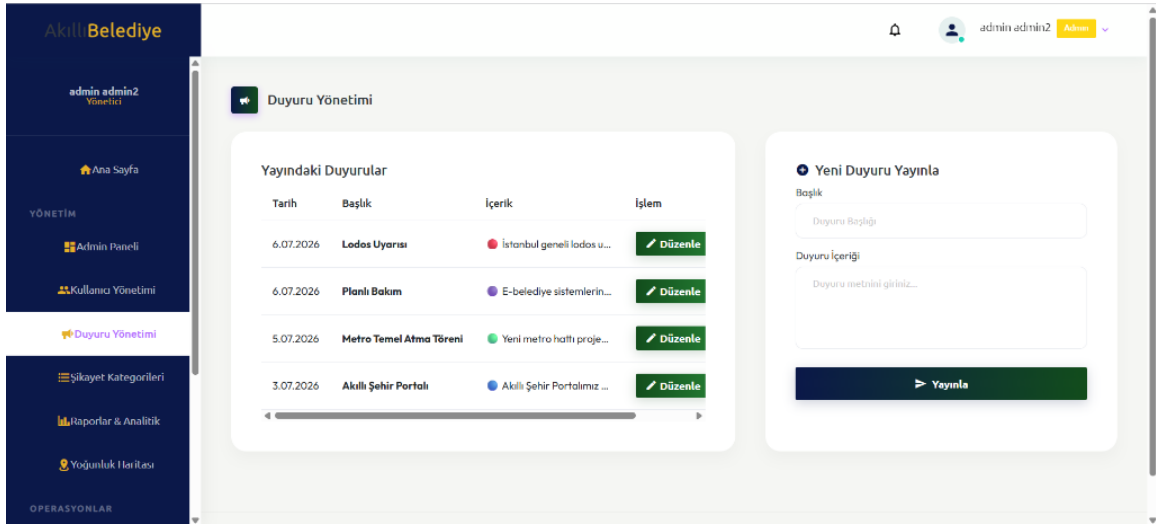
This screenshot is similar to the previous one but shows the 'Tüm Alanlar' dropdown menu expanded. The menu lists the following search criteria: 'Tüm Alanlar', 'Ad Soyad', 'E-Posta', 'TC No', 'İlçe', and 'Rol'. The table below remains the same as in the previous screenshot.

Ad Soyad	E-Posta	Telefon	TC No	İlçe	Rol	İşlemler
Ali Demir	alidemir@operator.bel.tr	5556321247	30156241289		Personel	Düzenle Sil
kullanıcı kullanıcı	kullanici@mail.com	5555555555	12345678912	Bakırköy	Yatandaş	Düzenle Sil
admin1 admin1	admin1@belediye.bel.tr	5555555555	12345678912		Admin	Düzenle Sil
admin admin2	admin123@belediye.bel.tr	5555555555	12396365412		Admin	Düzenle Sil
Test Admin	test@belediye.bel.tr	5551234567	12345678901		Admin	Düzenle Sil

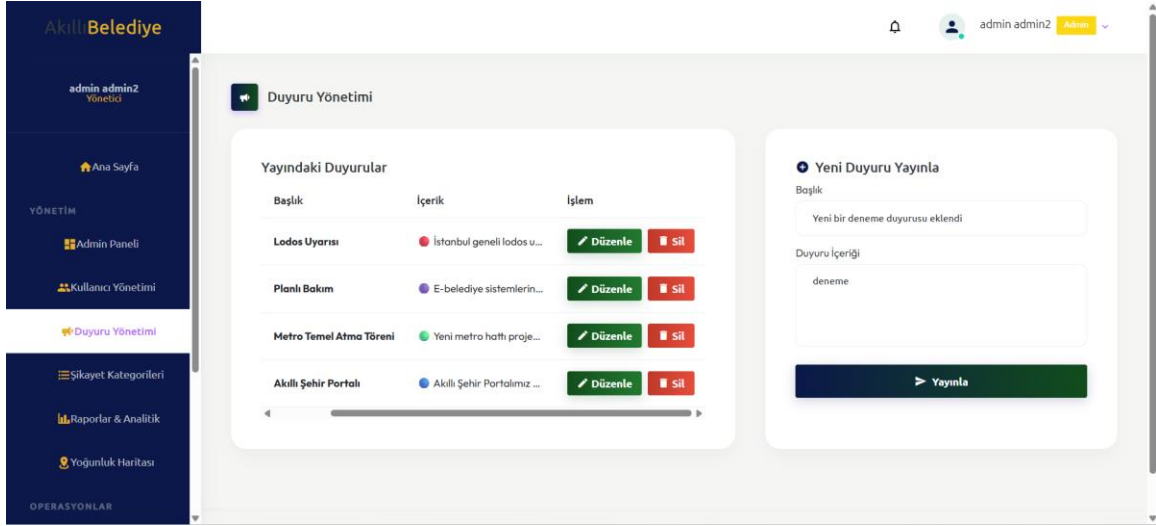
Şekil 4.45: Kullanıcı Arama Kriterleri



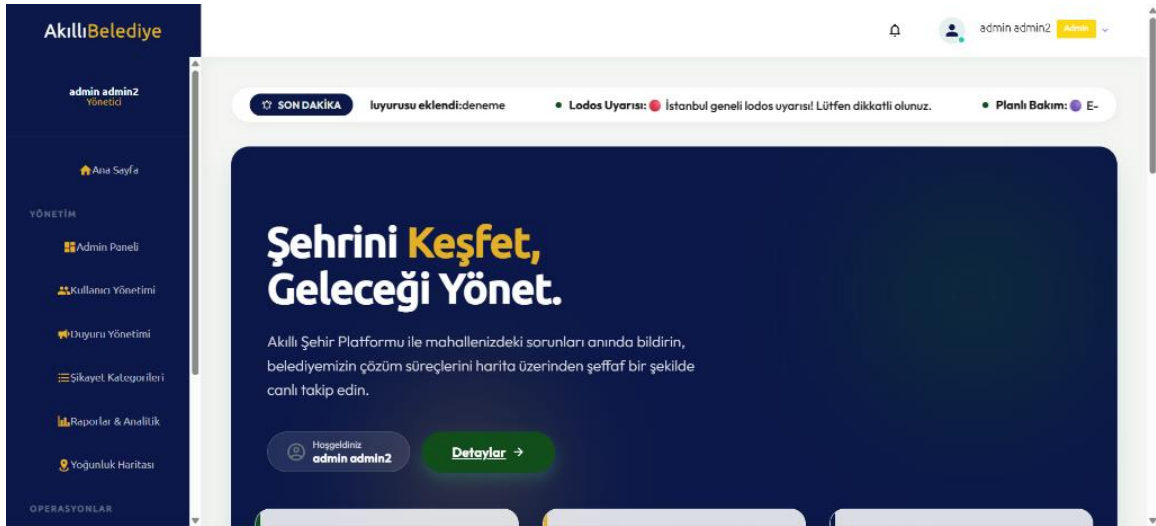
Şekil 4.46: Filtrelenmiş Kullanıcılar



Şekil 4.47: Duyuru Yönetimi



Şekil 4.48: Duyuru Ekleme Süreci

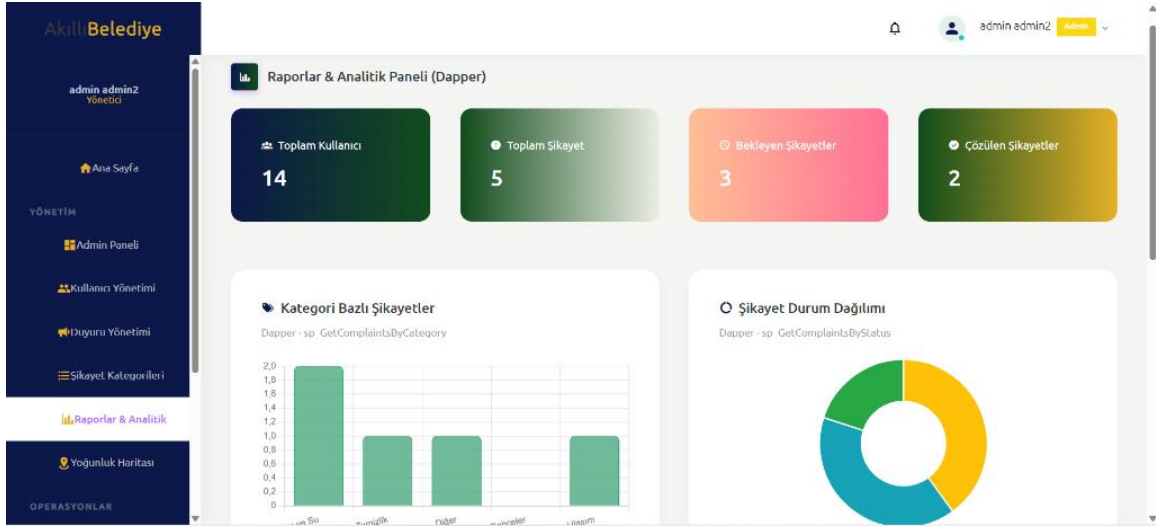


Şekil 4.49: Kayan Yazı

4.4. Aşama 4: Yüksek Performanslı Analiz

1. Dapper Destekli İstatistik Rapor Ekranı

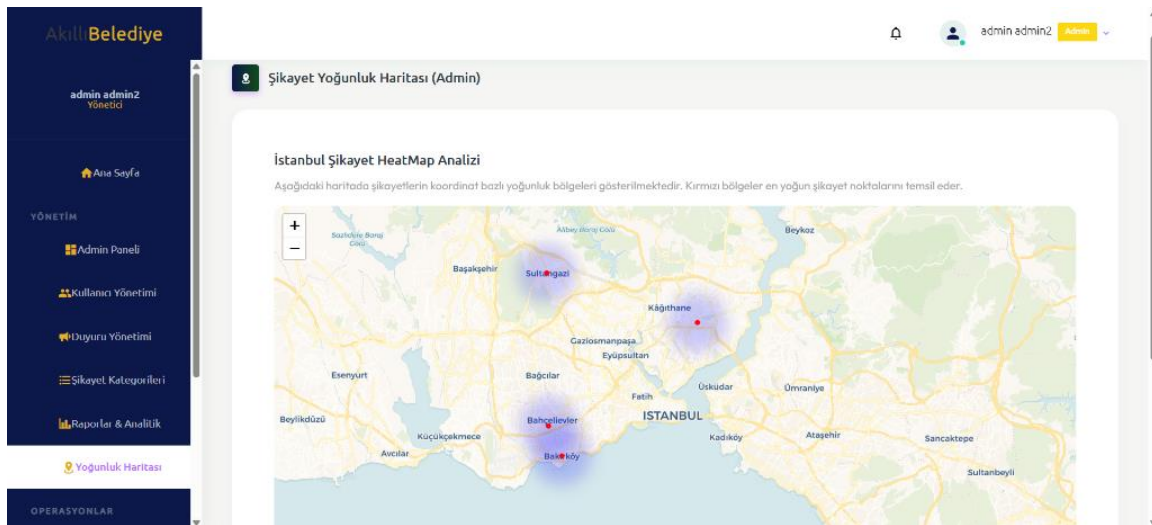
Yüksek verinin bulunduğu ortamlarda sistemin hız kaybetmemesi için Dapper mimarisi kullanılarak hazırlanan; ilçe şikayet dağılımları, aylık gelir analizleri ve coğrafi yoğunluk haritasının (HeatMap) sunulduğu ileri düzey yönetim gösterge panelidir.



Şekil 4.50: Analitik 1



Şekil 4.51: Analitik 2



Şekil 4.52: HeatMap

5. SONUÇ

Bu projede, modern yazılım mimarileri ve güncel teknolojiler (.NET 8.0, Entity Framework Core, Dapper Hibrit Yapısı) kullanılarak kurumsal düzeyde yüksek performanslı ve güvenli bir Akıllı Belediye Yönetim Platformu geliştirilmiştir.

Geleneksel, evrak tabanlı ve yavaş ilerleyen belediye süreçleri tamamen dijital ortama taşınmış; çevrimiçi abonelik yönetimi, güvenli fatura ödeme, harita/konum tabanlı interaktif şikayet bildirim modülü ve ileri seviye yapay zeka asistanı gibi yenilikler vatandaşların kullanımına sunulmuştur. Sistem, N-Tier (Çok Katmanlı) yapısı, API tabanlı ayrık servisleri ve sağlam kimlik doğrulama mimarisi sayesinde gelecekteki geliştirmelere ve yüksek trafikli kullanımlara karşı kolayca ölçeklenebilir bir formda başarıyla tamamlanmıştır.

6. KAYNAKÇA

[1] Microsoft ASP.NET Core Resmi Dokümantasyonu: <https://learn.microsoft.com/en-us/aspnet/core/>

[2] Entity Framework Core Rehberi: <https://learn.microsoft.com/en-us/ef/core/>

[3] Dapper (Micro ORM) Açık Kaynak Kütüphanesi: <https://github.com/DapperLib/Dapper>

[4] QuestPDF Kütüphanesi: <https://www.questpdf.com/>

[5] Google Gemini API Platformu: <https://ai.google.dev/>