İlaç Takip ve Uyarı Sistemi - Proje Raporu

Öğrenci: Ferat Kılın

Ders: Nesneye Yönelik Programlama

Danışman: Ahmet Nusret Özalp

Tarih: 08/06/25

# İçindekiler

1. Giriş

2. Sistem Tasarımı

2.1 Uygulama Mimarisi

2.2 Kullanılan Teknolojiler

3. Veritabanı Tasarımı

3.1 ER Diyagramı

3.2 Veritabanı Tabloları

4. Sınıflar ve Fonksiyonlar

4.1 GirisForm

4.2 KayitForm

4.3 Form1

4.4 VeriTabaniBağlanti sınıfı

5. Test ve Sonuçlar

5.1 Test Ortamı

5.2 Test Senaryoları ve Sonuçları

6. Ekler

6.1 Tasarım Görüntüleri

6.2 Örnek Kodlar

.

# 1. Giriş

Bu proje, kullanıcıların ilaçlarını zamanında ve güvenli bir şekilde kullanmalarını sağlamak amacıyla geliştirilmiştir. Ayrıca, riskli ilaç kombinasyonları tespit edilerek kullanıcı bilgilendirilmektedir.

# 2. Sistem Tasarımı

## 2.1 Uygulama Mimarisi

Uygulama, kullanıcı arayüzü, iş mantığı ve veri erişim katmanlarından oluşmaktadır.

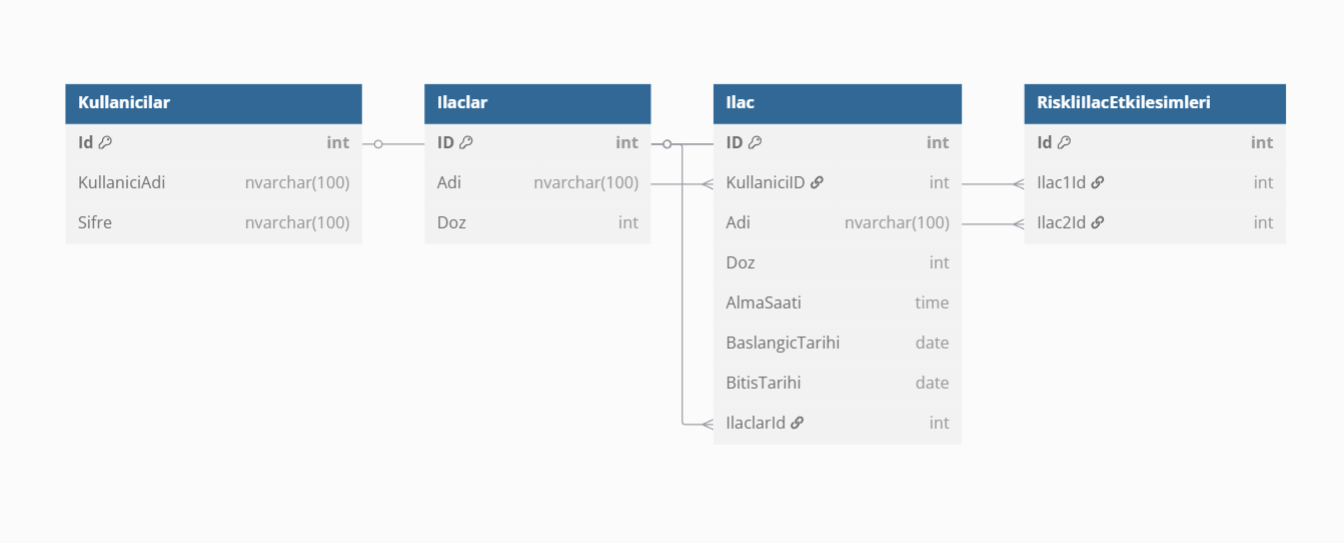
## 2.2 Kullanılan Teknolojiler

- C# (.NET Windows Forms)  
- Guna UI  
- MSSQL  
- C# Timer sınıfı

# 3. Veritabanı Tasarımı

Bu projede ilaç takip sistemi için dört temel tablo tasarlanmıştır. Kullanicilar tablosu kullanıcı bilgilerini, Ilaclar tablosu genel ilaç bilgilerini tutar. Ilac tablosu ise kullanıcıların kişisel ilaç kullanım kayıtlarını içerir ve Kullanicilar ile Ilaclar tablolarına bağlanır. RiskliIlacEtkilesimleri tablosu ise olası ilaç etkileşimlerini tutar ve iki farklı ilacı birbirine referans gösterir. Bu yapıyla kullanıcılar güvenli ve düzenli şekilde ilaçlarını takip edebilir, riskli kombinasyonlar sistem tarafından uyarılır.

## 3.1 ER Diyagramı



## 3.2 Veritabanı Tabloları

**Kullanicilar**

* Id (PK): Kullanıcının benzersiz kimlik numarası.
* KullaniciAdi: Kullanıcının sistemde giriş yapmak için kullandığı kullanıcı adı
* Sifre: Kullanıcının şifresi (güvenlik açısından genellikle şifrelenmiş/hassas veri olarak saklanır).

**Ilac**

* ID (PK): İlacın benzersiz kayıt numarası.
* KullaniciID (FK): İlacı kullanan kullanıcıyı belirten yabancı anahtar (Kullanicilar tablosuna referans).
* Adi: İlacın adı.
* Doz: İlacın doz miktarı.
* AlmaSaati: İlacın kullanılması gereken saat.
* BaslangicTarihi: İlacın kullanımının başladığı tarih.
* BitisTarihi: İlacın kullanımının sona erdiği tarih.

**RiskliIlacEtkilesimleri** (İsteğe bağlı)

* Id (PK): Etkileşim kaydının benzersiz kimlik numarası.
* Ilac1Id (FK): Etkileşime giren birinci ilacın kimlik numarası (Ilaclar tablosuna referans).
* Ilac2Id (FK): Etkileşime giren ikinci ilacın kimlik numarası (Ilaclar tablosuna referans).

# 4. Sınıflar ve Fonksiyonlar

## 4.1 GirisForm

GirisForm, kullanıcıların sisteme giriş yaptığı ilk ekrandır. Arayüz, Guna UI bileşenleri ile modernleştirilmiştir.

**Ana Bileşenler ve İşlevler:**

* txtKullaniciAdi ve txtSifre: Kullanıcıdan giriş bilgileri alınır.
* btnGiris\_Click: Giriş butonuna tıklanınca çalışır. Girilen bilgiler VeriTabaniBaglanti.GirisYap fonksiyonuyla doğrulanır. Başarılı giriş yapılırsa Form1 (ana ekran) açılır.
* btnKayit\_Click: Kayıt ekranını (KayitForm) açar.

Bu form, kullanıcı oturum yönetiminin başladığı noktadır.

## 4.2 KayitForm

KayitForm, kullanıcıların sisteme yeni bir hesapla kayıt olmalarını sağlar.

**Önemli İşlevler:**

* txtKullanici ve txtSifre: Yeni kullanıcı bilgileri girilir.
* btnKayit\_Click: Girilen bilgiler kontrol edilir. Eğer kullanıcı adı veritabanında mevcut değilse, yeni kayıt yapılır (VeriTabaniBaglanti.KayitOl() fonksiyonu).
* KullaniciVarMi(): Veritabanında aynı kullanıcı adı daha önce alınmış mı kontrol eder.

Form kullanımı basit olup yalnızca yeni kullanıcı eklemeye odaklanmıştır.

## 4.3 Form1

Form1, giriş sonrası kullanıcıya gösterilen ana formdur. Kullanıcının ilaçlarını görüntülemesine, eklemesine, silmesine ve riskli ilaç etkileşimlerini kontrol etmesine imkân tanır.

**Öne Çıkan Özellikler:**

* **İlaç Listesi:** Kullanıcıya ait ilaçlar DataGridView içinde listelenir.
* **İlaç Ekle/Sil:** Kullanıcı, combobox’tan ilaç seçerek saat ve tarih bilgisiyle birlikte sisteme ilaç ekleyebilir veya mevcut bir ilacı silebilir.
* **Riskli Etkileşim Kontrolü:** Aynı anda alınan ilaçlar arasında veritabanında tanımlı riskli kombinasyon varsa kullanıcı uyarılır.
* **İlaç Alma Hatırlatması:** Timer nesnesi her dakika kontrol yapar. Saat geldiğinde kullanıcıya sesli ve görsel hatırlatma yapılır.

**Önemli Fonksiyonlar:**

* Form1\_Load: Form yüklendiğinde kullanıcıya ait ilaçları veritabanından çeker ve arayüze yerleştirir. Timer başlatılır.
* ilacTimer\_Tick: Timer her dakika tetiklenir. Kullanıcının ilaç alma saatine yaklaşıldığında bildirim verilir.
* RiskliEtkilesimKontrolEt(): Kullanıcının aldığı ilaçlar içinde riskli kombinasyonlar var mı kontrol edilir.
* btnIlacEkle\_Click: Yeni ilaç sisteme eklenir ve tablo güncellenir.
* btnIlacSil\_Click: Seçilen ilaç veritabanından silinir ve arayüz yenilenir.

Bu form, projenin en kapsamlı ekranıdır ve temel işlevselliği burada sağlanır.

## 4.4 VeriTabaniBağlanti Sınıfı

Bu sınıf, uygulamanın tüm veritabanı işlemlerinden sorumludur. SQL Server veritabanına bağlantı sağlayarak kullanıcı giriş/kayıt işlemleri, ilaç ekleme/silme, ilaç listeleme ve riskli etkileşim sorgulama gibi işlemleri gerçekleştirir. Static olarak tanımlandığı için doğrudan erişim sağlanabilir.

#### Öne Çıkan Metotlar:

* **GirisYap(string kullaniciAdi, string sifre, out int kullaniciId)**  
  Kullanıcı adı ve şifre ile giriş işlemi yapar. Doğrulama başarılıysa ilgili kullanıcının ID bilgisini döner.
* **KayitOl(string kullaniciAdi, string sifre)**  
  Yeni kullanıcı veritabanına kaydedilir.
* **IlacEkle(Ilac ilac, int kullaniciId)**  
  İlgili kullanıcıya ilaç ekler ve bu ilacın ID’sini döner.
* **IlaclariGetir(int kullaniciId)**  
  Belirli bir kullanıcıya ait tüm ilaçları döner.
* **GenelIlaclariGetir()**  
  Sistem genelinde tanımlı tüm ilaçları listeler (comboBox gibi seçim listelerinde kullanılır).
* **IlacSil(int ilacId, int kullaniciId)**  
  İlgili kullanıcıya ait bir ilacı veritabanından siler.
* **EtkilesimVarMi(string ilacAdi1, string ilacAdi2)**  
  İki ilaç arasında riskli bir etkileşim olup olmadığını kontrol eder.
* **RiskliEtkilesimleriGetir()**  
  RiskliIlacEtkilesimleri tablosundaki tüm eşleşmeleri döner.

# 5. Test ve Sonuçlar

Bu bölümde, geliştirilen ilaç takip ve uyarı sisteminin temel işlevlerinin testleri gerçekleştirilmiş ve elde edilen sonuçlar değerlendirilmiştir.

## 5.1 Test Ortamı

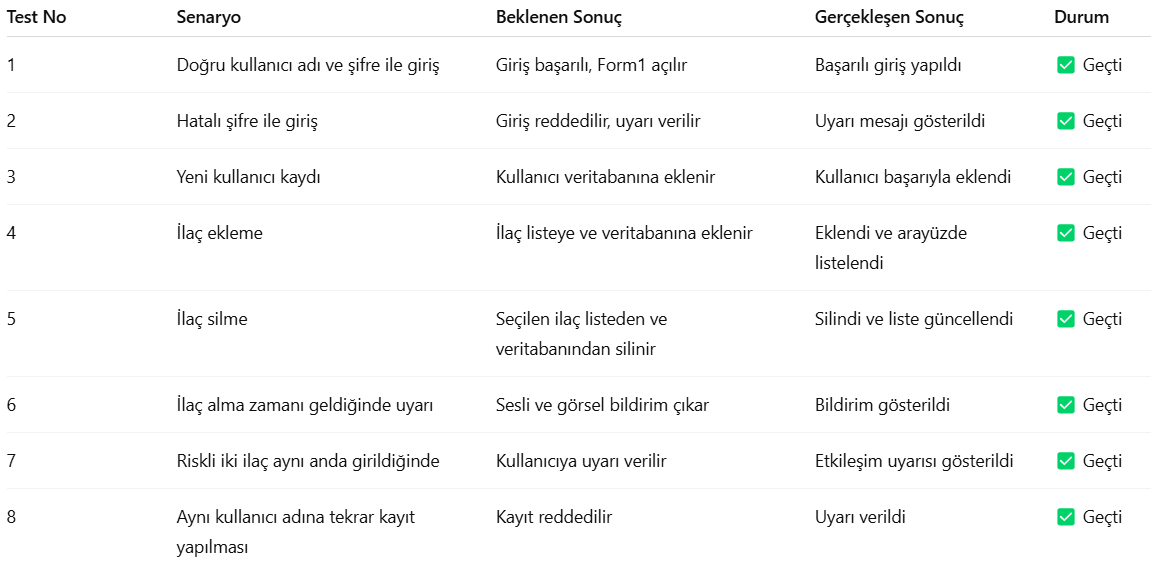
 **Geliştirme Aracı:** Visual Studio

 **Veritabanı:** MSSQL LocalDB

 **Arayüz:** Windows Forms (Guna UI ile zenginleştirilmiş)

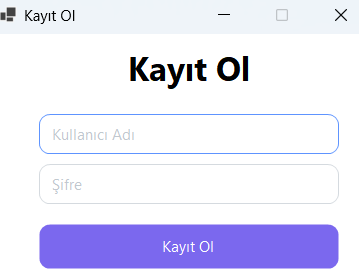
 **Test Kullanıcısı:** Test kullanıcıları ile sahte veriler girilerek senaryolar oluşturulmuştur.

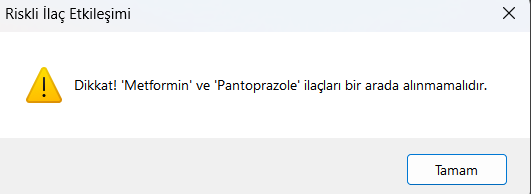
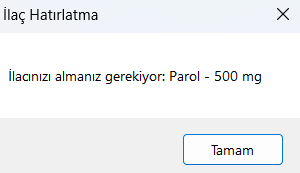
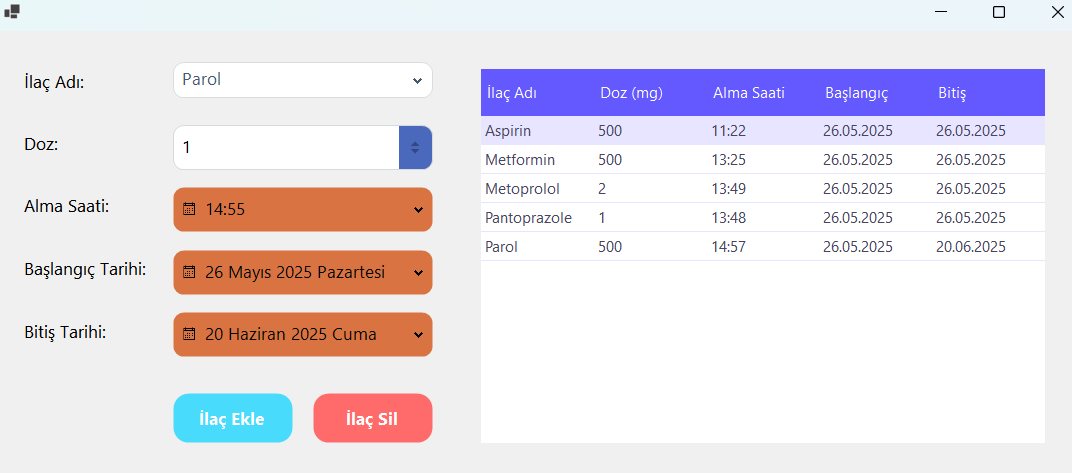
## 5.2 Test Senaryoları ve Sonuçları



# 6. Ekler

## 6.1 Tasarım Görüntüleri





## 6.2 Örnek Kodlar

