**HASTANE RANDEVU SİSTEMİ DÖKÜMANTASYONU**

**Ahmet Hakan TULGAÇ**

## **ÖNSÖZ**

Bu dökümanda hasta, doktor, sekreter ve eczacı aktörlerinin yapacağı işlemleri daha kolay yapabilmesini, tüm aktörlerin her süreçte bilgi alabilmesini ve interaktif bir şekilde süreçlere dahil olabilmesini amaçladığımız hastane randevu sisteminin gereksinim analizi, kapsamı, amacı, sistem hazırlanırken kullanılan teknolojiler ve elde edilen sonuçlarından bahsedilmektedir.

## **İÇİNDEKİLER**

1. **GİRİŞ**
   1. Sistem Gereksinimleri
   2. Amaç
2. **UYGULAMANIN KULANIM ALANI**
   1. Kapsam
   2. Tanımlamalar
3. **YAZILIM GELİŞTİRME SÜRECİ**
   1. Yazılım Geliştirmede Genel Durum
   2. Kullanılan Teknolojiler
   3. Kullanılan Tasarım Desenleri
4. **DİYAGRAMLAR**
   1. Use Case Diyagramı
   2. Class Diyagramı
   3. Sequence Diyagramı
5. **SONUÇ**

5.1. Elde Edilenler

1. **KAYNAKLAR**

## **1-GİRİŞ**

**1.1-Sistem Gereksinimleri**

Sistem Gereksinimleri Hastane otomasyonu tasarlanırken aşağıdaki gereksinimler dikkate alınacaktır.

⦁ Hasta, hastaneye randevu talebinde bulunabilmeli ve süreci takip edebilmelidir.

⦁ Hasta, geçmişte aldığı randevuları ve kullandığı ilaçları inceleyebilmelidir.

⦁ Hasta güncel reçetesine erişebilmelidir.

⦁ Hasta reçetesinde bulunan ürünlerin tutarını öğrenebilmelidir.

⦁ Sekreter, tüm randevuları görüntüleyebilmeli, onaylayabilmeli veya iptal edebilmelidir.

⦁ Sekreter, diğer kullanıcılara bilgilendirmelerde bulunabilmelidir.

⦁ Doktor hastanın bilgilerine erişebilmeli, güncelleyebilmeli ve hastaya ilaç yazabilmelidir.

⦁ Eczane hastanın reçete bilgilerine erişebilmelidir.

⦁ Sistem, farklı kullanıcı profillerinin kayıt olabilmesini, giriş 1 yapabilmesini ve kullanıcı yetkilendirmelerini desteklemelidir.

⦁ Sistem her kullanıcı tipine ayrı arayüz sunmalıdır.

⦁ Tüm aktörler sisteme kayıtlı olmalıdır.

## **1.2-Amaç**

Hastane randevu sisteminin amacı genel olarak;

* Hastanın randevu alabilmesi ve bu süreci takip edebilmesi
* Hastanın randevu alırken doktor tercihinde bulunabilmesi
* Hastanın sistemden güncel ve eski randevularını görüntüleyebilmesi
* Hastanın reçetelerine erişebilmesi
* Doktorun randevu takip edebilmesi, iptal edebilmesi
* Sekreterin randevuları onaylaması veya iptal edebilmesi
* Eczanelerin hastaların reçetelerine erişebilmesi
* Randevu iptali durumunda hasta ve doktorun bilgilendirilmesi

İşlemlerini tek bir çatı altında toplayarak aktörlere kolaylık sağlamaktır.

**2-UYGULAMANIN KULLANIM ALANI**

## **2.1-Kapsam**

Bu proje aşağıdakileri kapsar:

### HASTA

* Hasta uygulama üzerinden randevu alabilir.
* Hasta uygulama üzerinden reçete bilgilerine erişebilir.
* Hasta randevusunun iptal olması durumundan haberdar olur.

### DOKTOR

* Doktor uygulama üzerinden randevu bilgilerine erişebilir.
* Doktor uygulama üzerinden randevu iptali yapabilir.

SEKRETER

* Sekreter uygulama üzerinden hastaların randevu taleplerini onaylayabilir veya iptal edebilir.

ECZANE

* Eczane, sistem aracılığıyla hastanın reçete bilgilerine erişebilir.

## **2.2-Tanımlamalar**

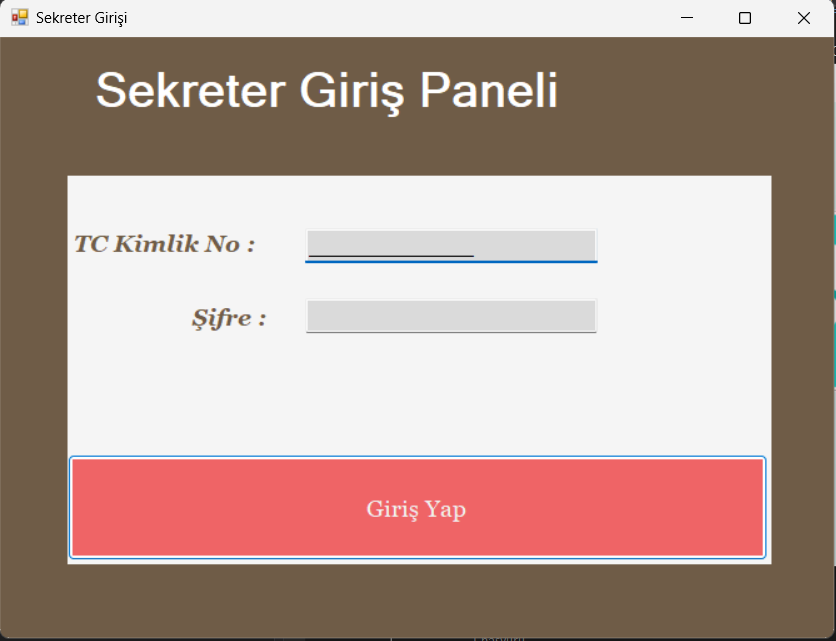
### TANIMLAMALAR

Hastane randevu sistemi kullanıcıların tüm ihtiyaçlarını kolaylıkla, istenilene en yakın şekilde karşılayabileceği bir sistem olarak düşünülmüştür ve tasarlanmıştır. Bu bağlamda hastane randevu sistemi erişilebilirliği ve işlemler hakkında bilgi edinmeyi kolaylaştırmıştır.

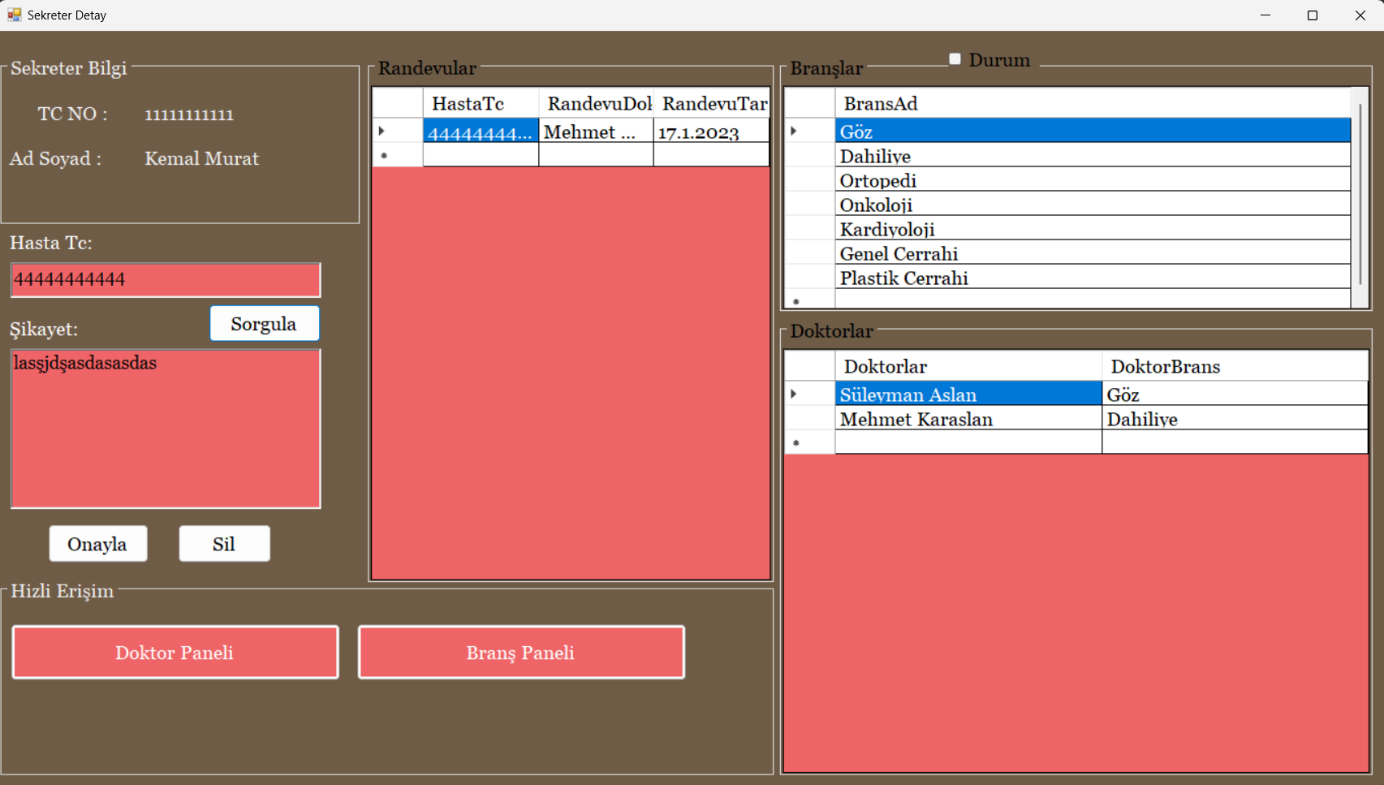
Bu süreçler sayesinde hiçbir aktörün zor durumda kalmayacağı, işlemleri daha kolay hale getiren bir sistem amaçlanır. Hastanelerdeki kalabalığın oluşturabileceği sıkıntıların önlenmesi de uygulamanın getirilerinden biridir.

**3-YAZILIM GELİŞTİRME SÜRECİ**

**3.1-Yazılım Geliştirmede Genel Durum**



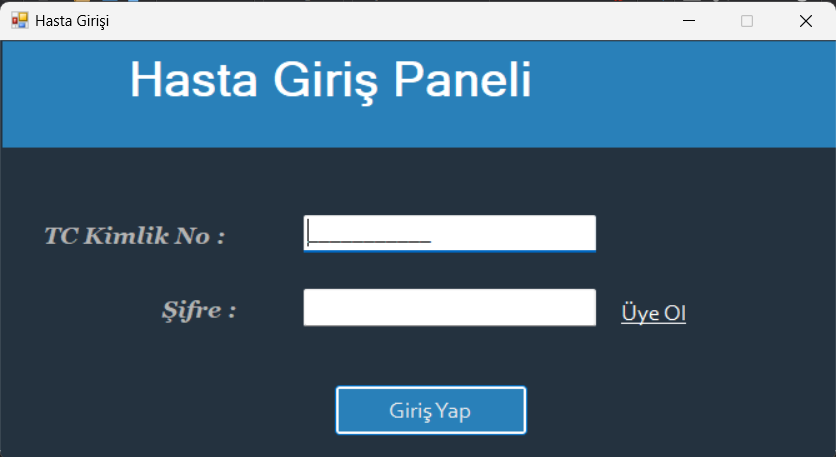
**Şekil 3.1.1:** Sekreter giriş ekranı



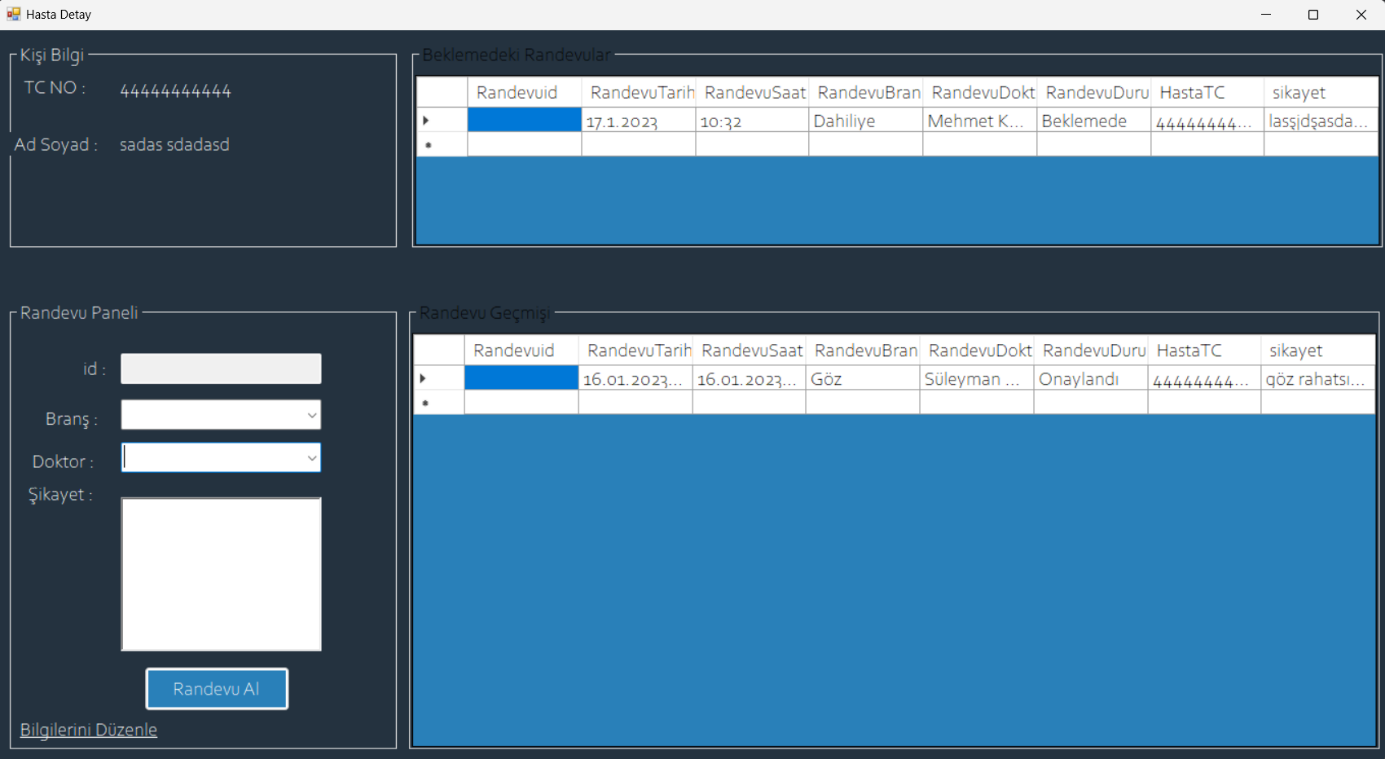
**Şekil 3.1.2 :** Sekreterin randevu taleplerini görüp onayladığı veya iptal ettiği bölüm



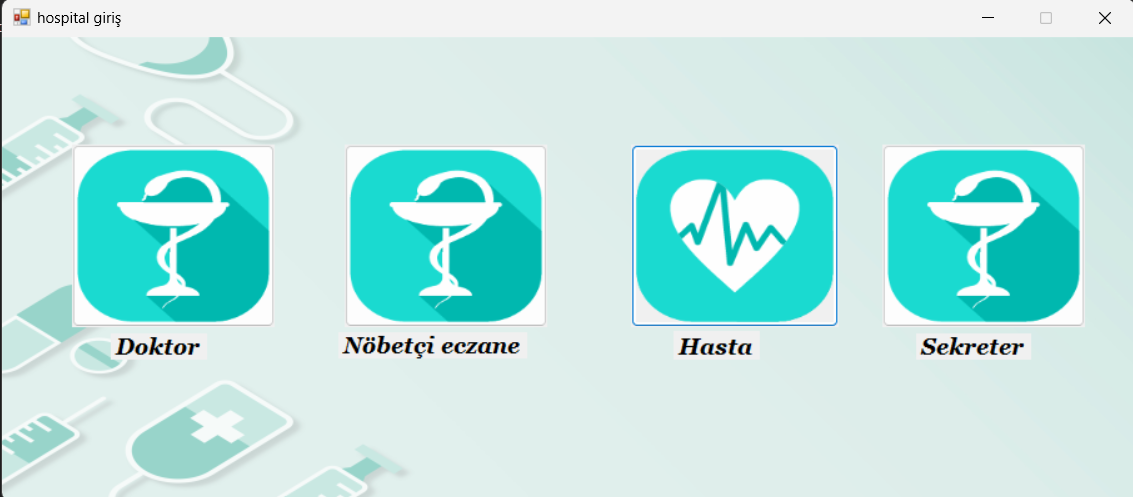
**Şekil 3.1.3:** Hasta kayıt ekranı



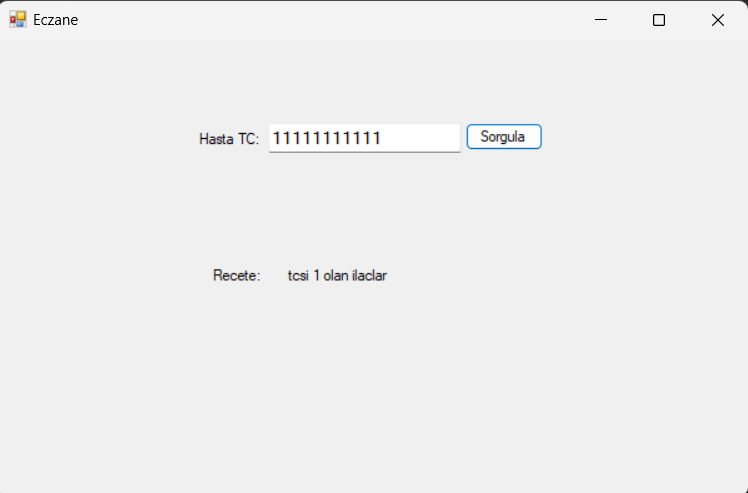
**Şekil 3.1.4:** Hasta giriş ekranı



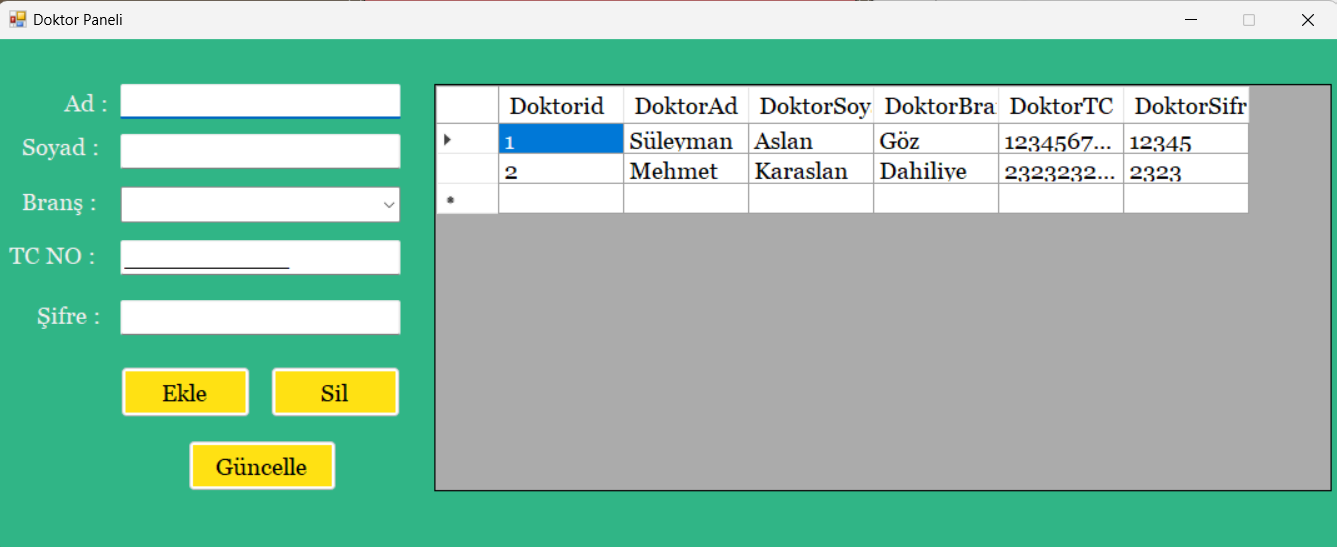
**Şekil 3.1.5:** Hastanın randevu talebi oluşturabildiği bölüm



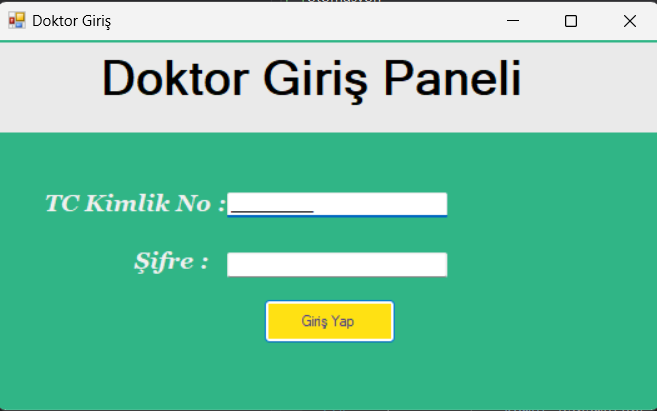
**Şekil 3.1.6:** Genel giriş ekranı



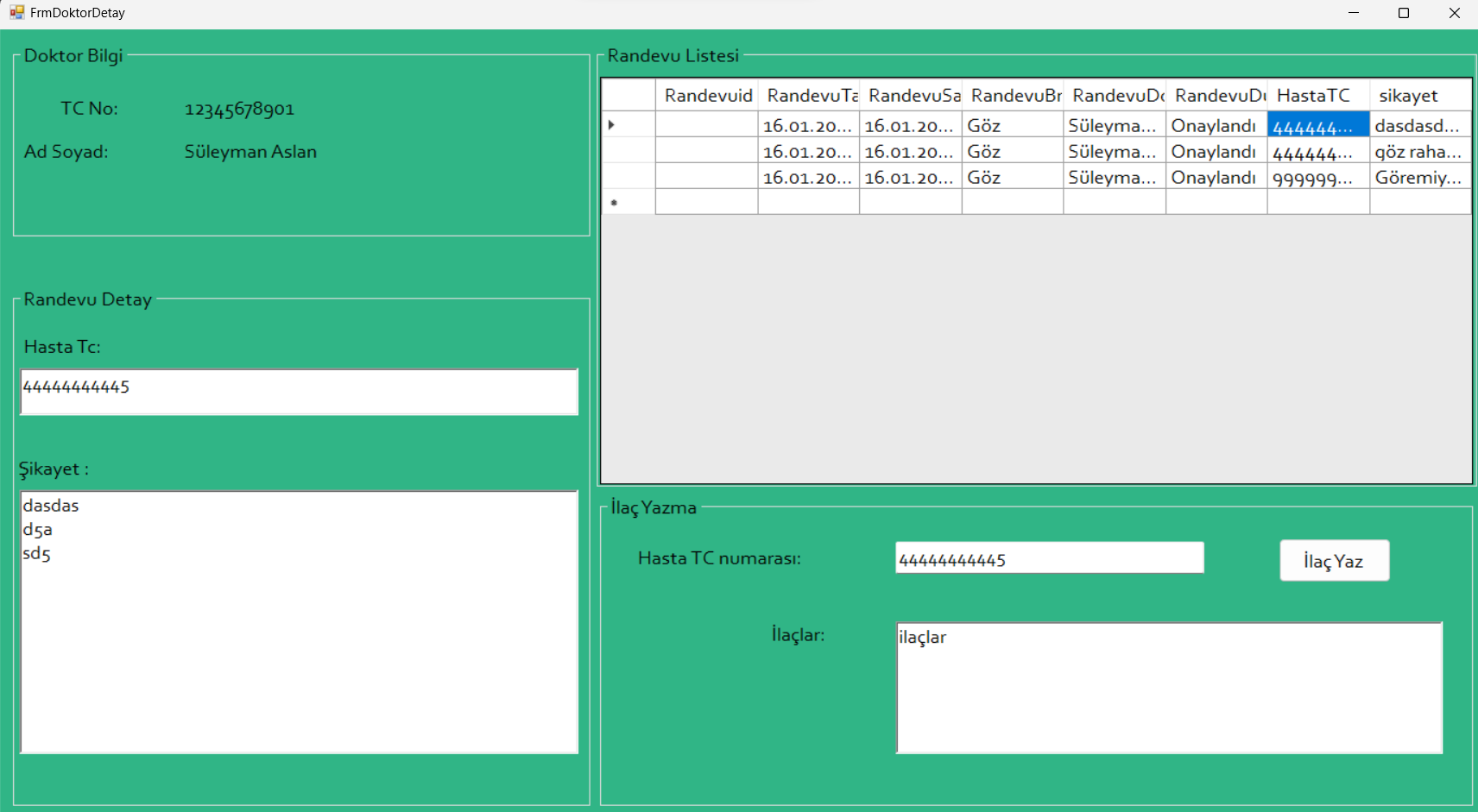
**Şekil 3.1.7:** Eczanelerin, hastaların reçete bilgilerini sorgulayabildiği bölüm



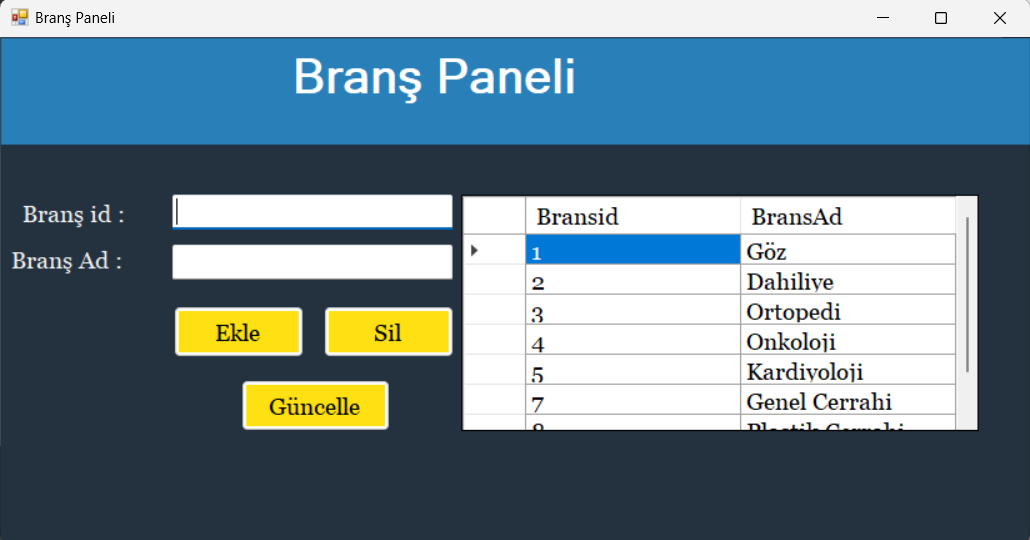
**Şekil 3.1.8**: Sekreterin, sistem aracılığıyla hastaneye yeni doktor kaydedebildiği veya mevcut doktorların sistemden silinmesi işlemlerini yapabildiği bölüm



**Şekil 3.1.9:** Doktor giriş ekranı:



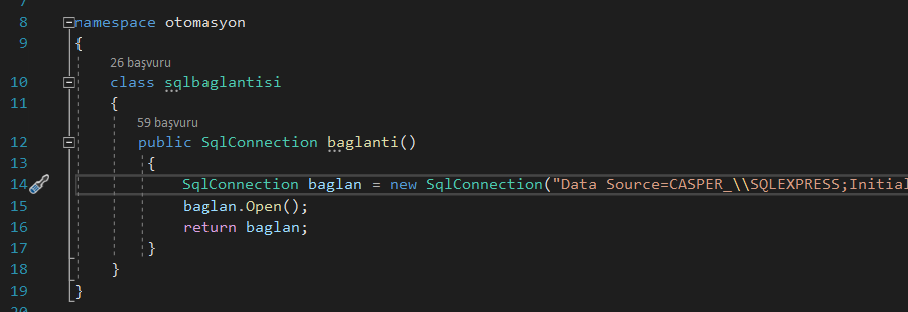
**Şekil 3.1.10:** Doktorun, sekreter tarafından onaylanan randevuları görebildiği ve hastalara ilaç yazabildiği bölüm

 **Şekil 3.1.11:** Sekreterin, hastalar hastanede hangi branşların olduğunu öğrenebilmesi ve bu branşlardan randevu alabilmesi sağlamak amacıyla hastaneye kayıtlı doktorlara göre branş ekleyebildiği veya silebildiği bölüm

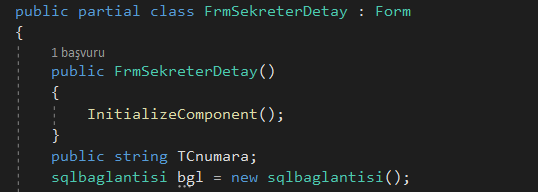
**3.2-Kullanılan Teknolojiler**

* Microsoft Visual Studio 2019
* Microsoft SQL Server 2019
* Draw.io

**3.3-Kullanılan Tasarım Desenleri**

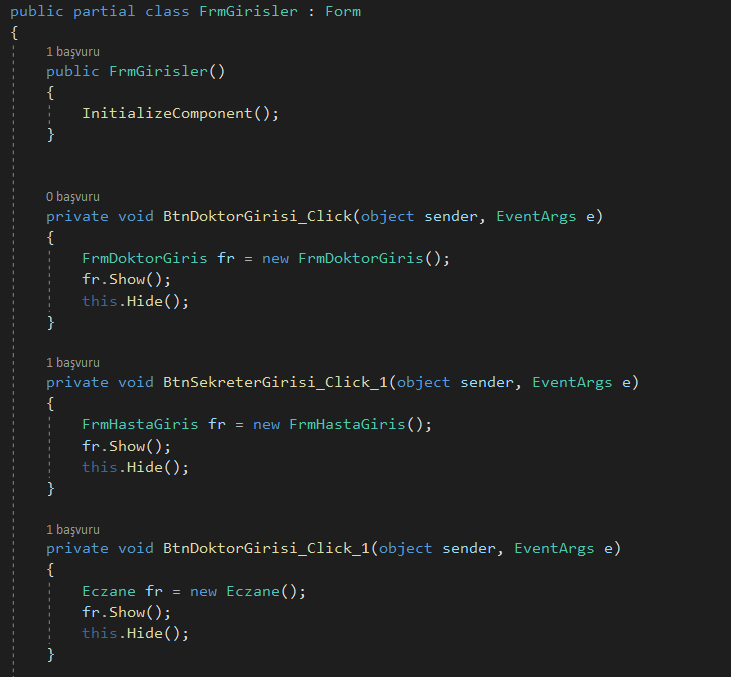


**Şekil 3.3.1**

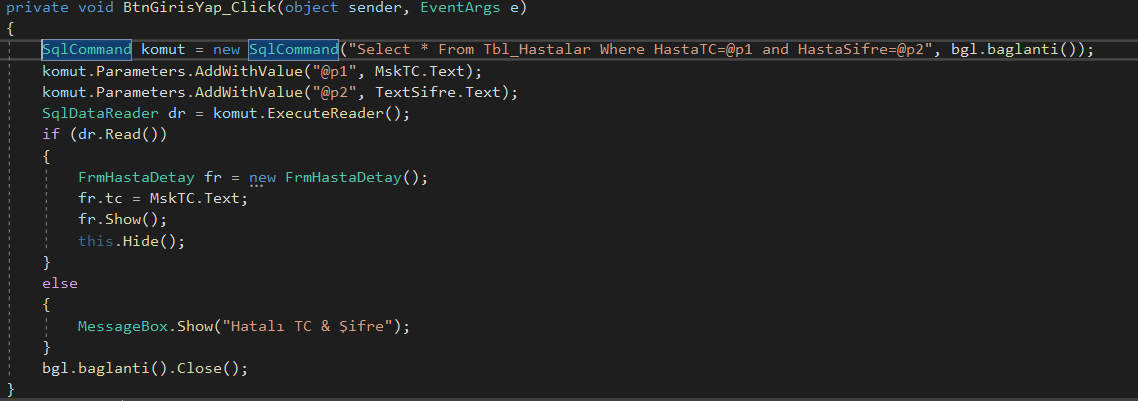


**Şekil3.3.2**

Sqlbaglantisi adındaki Class’ta veri tabanı ile bağlantı kuruldu. Daha sonra veritabanı ile işlem gerektiren classlarda nesne oluşturulup, bu classlara veritabanı arasında bağlantı sağlandı. (Şekil 3.3.1 ve 3.3.2)

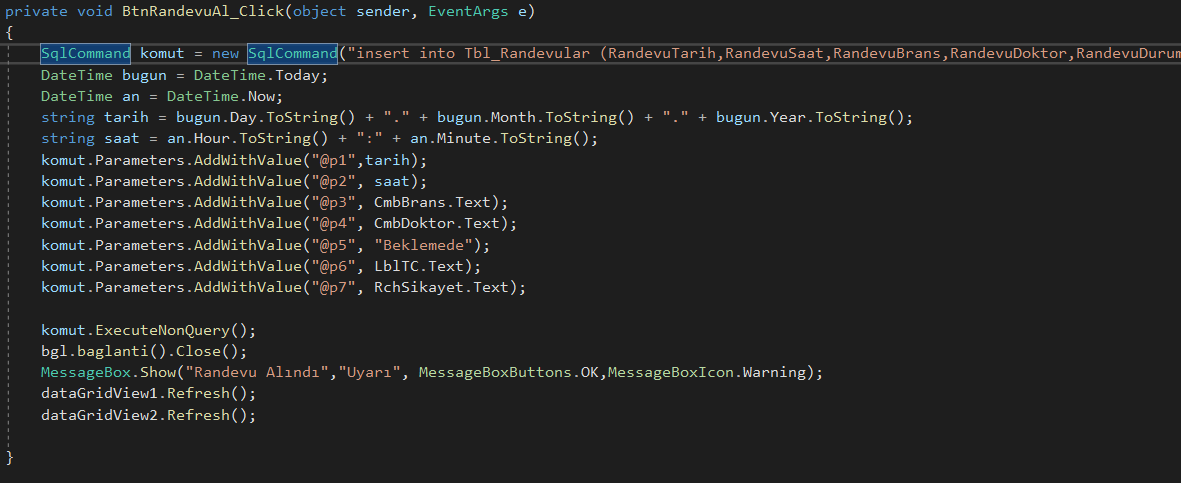


**Şekil 3.3.3**

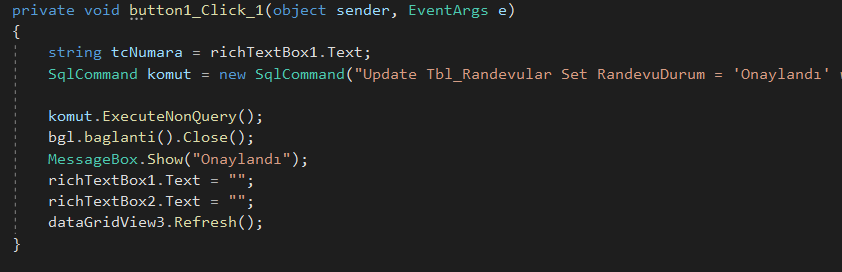


**Şekil 3.3.4**

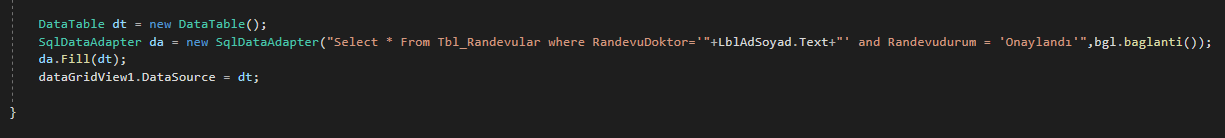
Farklı Kullanıcı tipleri sisteme, Girisler adındaki bir classı kullanarak giriş yaptı. Daha sonraki açılan arayüzler sadece ilgili kullanıcıya sunuldu.(Şekil 3.3.3 ve 3.3.4)



**Şekil 3.3.5:** Burada hasta randevu isteği oluşturur.



**Şekil 3.3.6:** Sekreter randevu isteklerini görüntüler herhangi birini onaylar ya da siler.

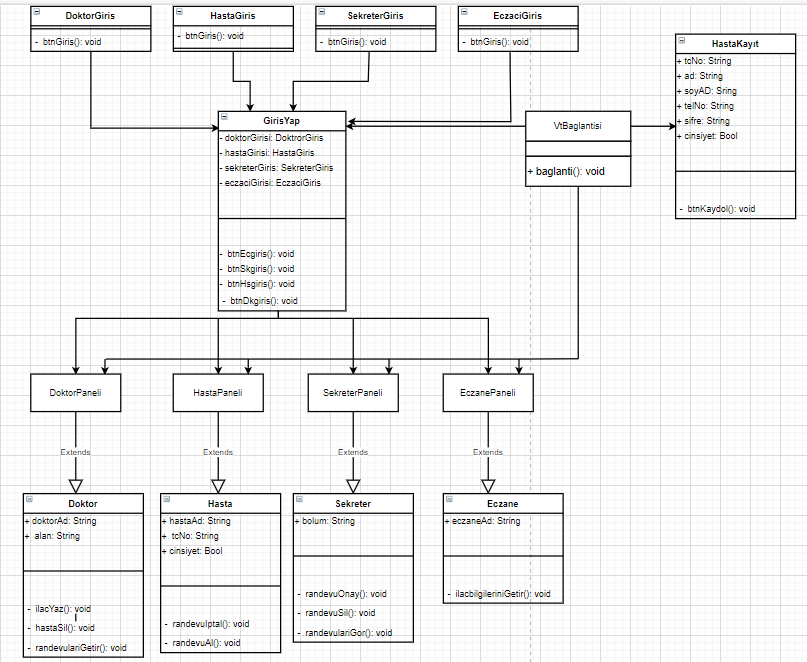


**Şekil 3.3.7:** Sekreter tarafından onaylanan randevular doktor tarafından görüntülenir.

**4-DİYAGRAMLAR**

**4.1-Class diyagramı**

Hazırlanan class diyagramı UML’in en sık kullanılan diyagramlardan biri olup bu projede nesne yönelimli analiz, tasarım ve programlamadaki sınıfları net ve anlaşılabilir şekilde ortaya çıkarmak için yapılmıştır.



**Şekil 4.1.1:** Hastane randevu sisteminin class diyagramı

**4.2-Use case diyagramı**

Use Case diyagramları, genellikle geliştirmenin erken aşamasında, iç ve dış

etkiler dahil olmak üzere sistem gereksinimlerini toplamak için kullanılır. Bu

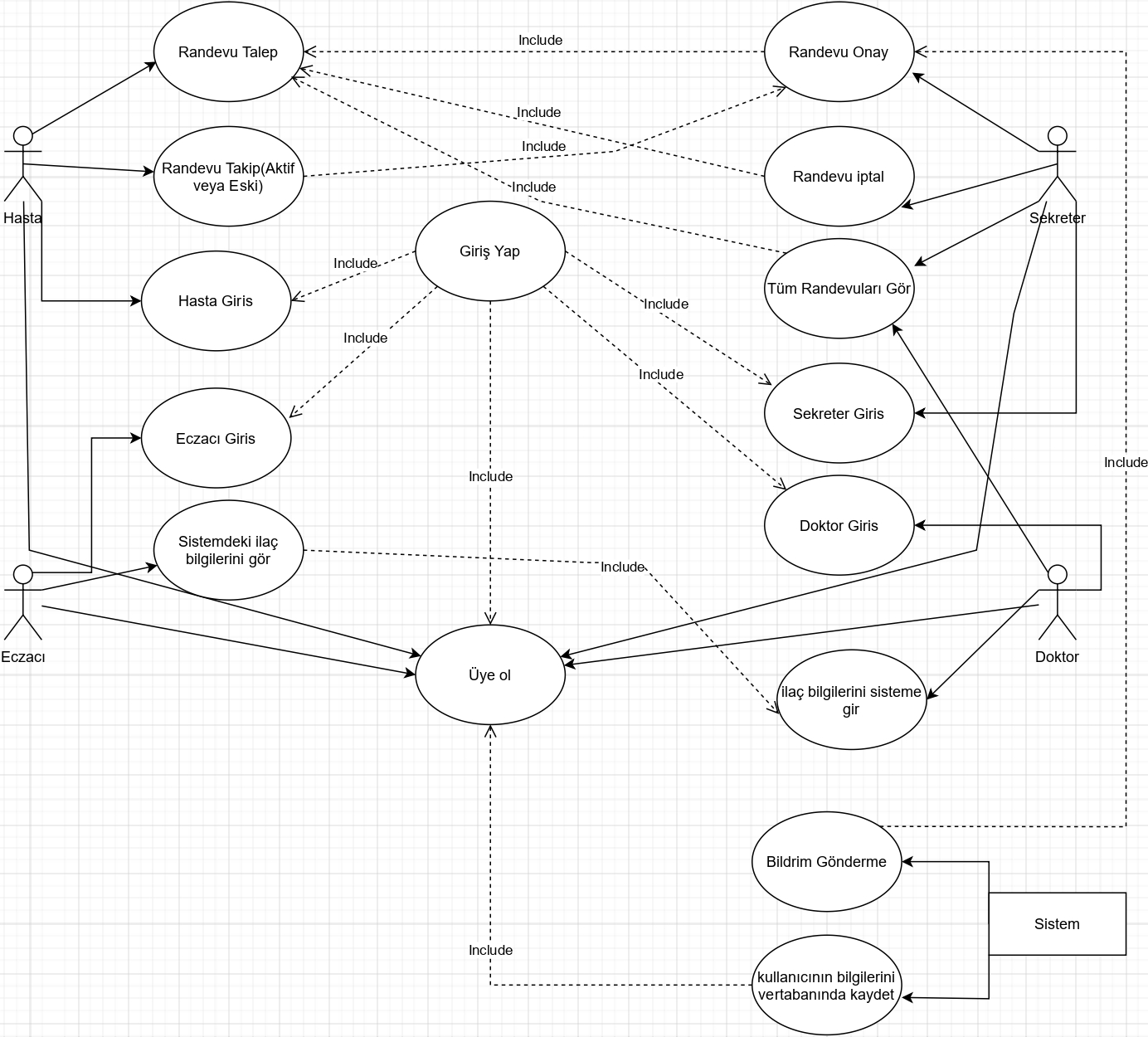
gereksinimler çoğunlukla tasarım gereksinimleridir. Bu projede kullanılan Use

Case diyagramı, iş süreçlerinin yönetilmesi aşamasında ihtiyaç duyulan tüm

fonksiyonları, bu fonksiyonları tetikleyecek aktörleri, fonksiyonlardan

etkilenecek aktörleri ve fonksiyonlar arasındaki ilişkiyi göstermek amaçlı

kullanılmaktadır.



**Şekil 4.2.1:** Hastane randevu sisteminin use-case diyagramı

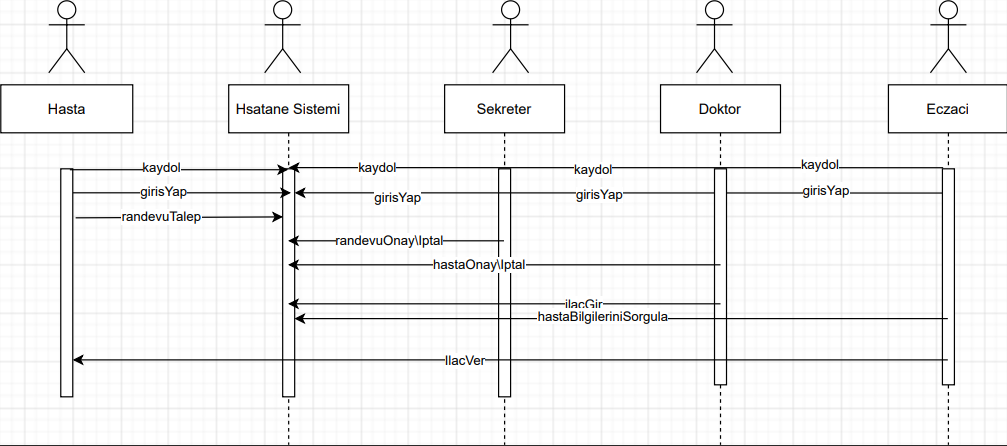
**4.3-Sequence diyagramı**

Sequence Diyagramı, belli bir kullanım senaryosunda (use case) rol alan

nesneleri ve bu nesneler arasında geçilen mesajları gösteren bir diyagramdır.

Bu projede belli bir zaman sırasına göre düzenlenmiş olan nesne etkileşimi fazla

olduğu için kullanılma gereği duyulmuştur.



**Şekil 4.3.1:** Hastane randevu sisteminin sequence diyagramı

**5-Sonuç**

**5.1-Elde Edilenler**

Proje sonucunda hastalar randevularını kolayca hastane randevu sistemi üzerinden alabilir. Bu randevu saatlerine dikkat edilmesi durumunda hastanelerde hastalar uzun süre beklemek zorunda kalmayacak ve kalabalık yüzünden acil durumu olan hastaların işlemlerinin aksaması önlenecektir.

Tüm aktörlerin, tüm işlemlerini tek bir çatı altında toplayan hastane randevu sistemi bu bağlamda bir bütünlük sağlayacaktır.

Hastaların doktor tercih edebilmesini sağlayan hastane randevu sistemi mahremiyete önem veren hastalar için büyük bir avantaj sağladığı gibi sağlık çalışanlarının, mahremiyeti sağlık çalışanlarına karşı uygulanan şiddet içerikli olayların sebebi olarak gösteren dış aktörlerden görebileceği her türlü şiddeti azaltmayı hedefler.

**6-KAYNAKLAR**

* sourcemaking.com/design\_patterns