

**ONDOKUZ MAYIS ÜNİVERSİTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ**

**BIL407 Yazılım Mühendisliği**

**HASTANE RANDEVU YAZILIM SİSTEMİ**

**Katılımcılar:**

**16061806 Fatih UĞURLU**

**17060612 Ekrem TUNÇ**

**Ö19060012 Murat DEMİR**

**Proje Amacı:**

Projenin hedefi, herhangi bir hastanenin randevu gereksinimlerini karşılaması, işlemlerinin kolaylaştırması. Herhangi bir hastanede hastaların, doktorların ortak bir veri tabanı üzerinden işlemlerini takip edebilmelerini ve birimler arasında uyumlu çalışmalarını sağlamaktır. Bu yazılım hastanedeki bütün kaynakların (zaman, insan gücü vs.) en verimli şekilde kullanılmasına olanak verir. Elle takip edilen evrak ve yapılan işlemler bilgisayar tarafından takip edilip yönetilecek hale getirilir.

**Proje Özet**

**Hasta Kayıt Ekranı**

Hasta kayıt ekranında hasta bilgileri girilerek sitemde kaydı henüz hiç yok ise hasta kayıt işlemi yapılır.Kaydı yapılan hastalar hasta kayıt ekranında görüntülenir.Tc kimlik no , poliklinik ve doktora göre hasta arama işlemleri gerçekleştirilir."Hasta Özeti" butonu ile hangi hastanın hangi poliklinikte hangi doktora muayene olduğu görüntülenir."Hasta Listesi" butonu ile hasta görüntüleme ekranına geri döner."Ada Göre Sırala" ve "Yaşa Göre Sırala" butonları ile hasta bilgileri sıralanır."Hasta Sayısını Gör" butonu ile sistemde kayıtlı olan hasta sayısı görüntülenir."Doktor Listesi" butonu ile doktor bilgilerinin tutulduğu doktor listesi ekranına geçiş yapılır.

**Doktor Listesi Ekranı**

Doktor listesi ekranında doktor bilgileri görüntülenir.

" Doktor Ekle " butonu ile bilgileri girilen doktorun sisteme kaydı yapılır.

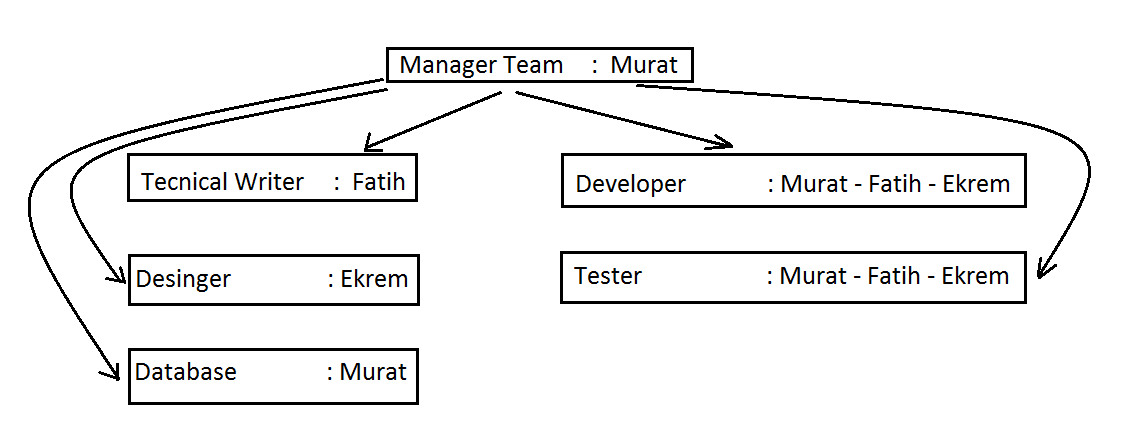
" Güncelle " butonu ile sistemde kayıtlı olan bir doktorun bilgileri güncellenir.

" Sil " butonu ile seçilen doktorun sistemden kaydı silinir.

**Randevu Ekranı**

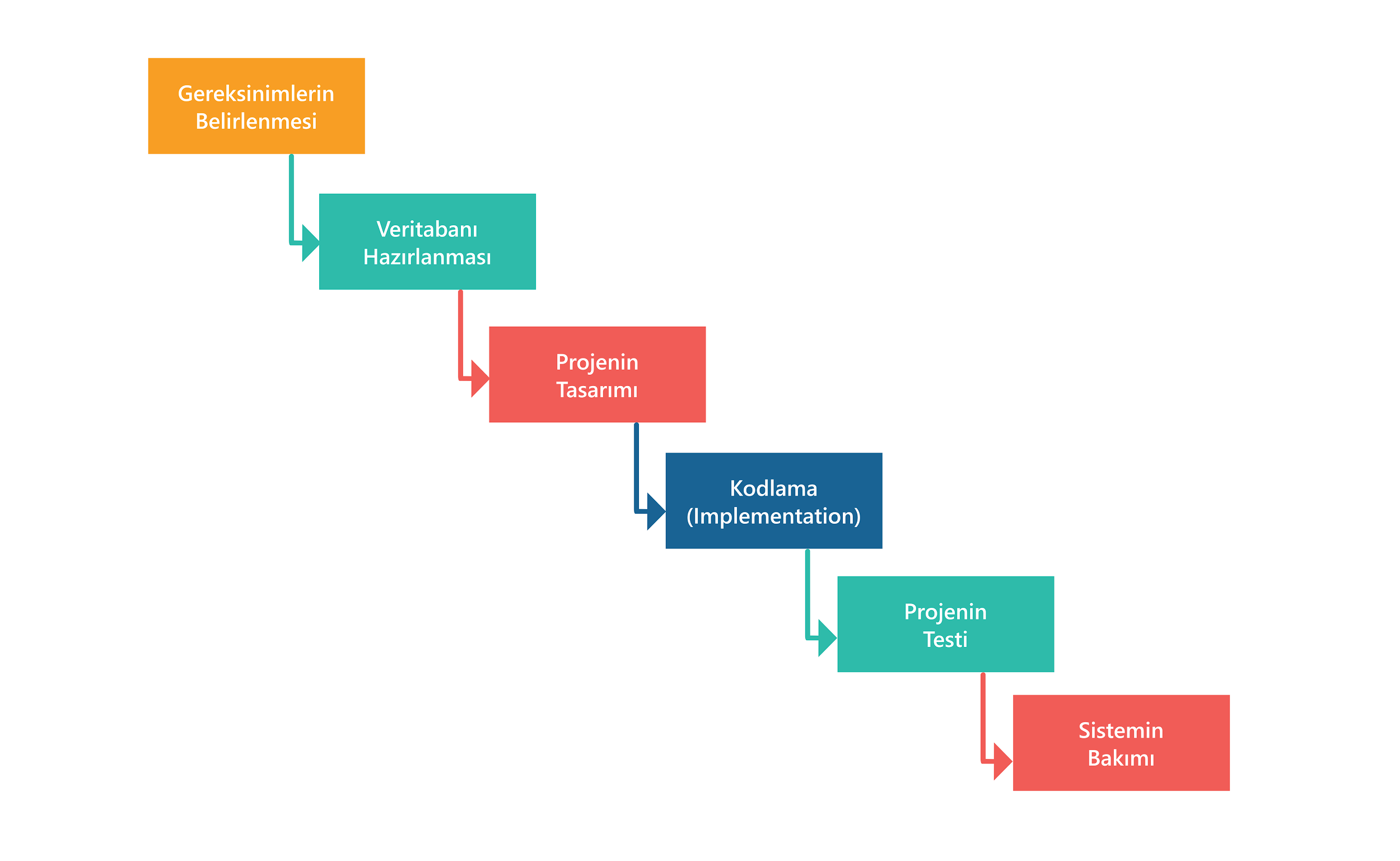
Hasta bilgisi girilerek sistemde kayıtlı hasta seçilir. Hastanın talebi doğrultusunda muayene olacağı bölüm ve doktor seçilerek randevu oluşturulur.





**SOFTWARE DEVELOPMENT LIFE CYCLE MODEL**

Waterfall modeli ile yazilim gelistirme yaşam döngümüzü gerçekledik.



**ORNEK USE-CASE DURUMLARI**

**1) Use-Case Adı :** Kullanıcı Giriş (Form2.cs)

**2) Katılımcı Aktörler :** Yönetici ve personel

**3) İş Akışı:**

* Kullanıcı kendi Bilgilerini(T.C kimlik ve şifresi) girer ve “Giriş Yap” butonuna tıklar.
* Kullanıcı bilgileri kontrol edilir.
* Bilgiler doğru ise siteme dahil edilir.

**4) Başlangıç Koşul/Koşulları:**

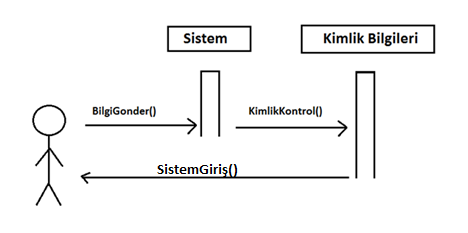
* Giriş butonuna basılması

**5) Bitiş Koşul/Koşulları:**

* Kullanıcı bilgileri girilmesi, kayıtlı olmadığının onaylanması.
* Başarılı şekilde sisteme giriş yapılabilmesi

**6) Kalite Koşul/Koşulları:**

* Bu işlemin 3- 5 saniye aralığında başarı ile tamamlanması

****

**1) Use-Case Adı:** Yönetici Kayıt

**2) Katılımcı Aktörler:** Yönetici ve Kullanıcı

**3) İş Akışı:**

* Yönetici ve kullanıcı bilgileri tanımlanır ve şifre belirlemesi yapılır

**4) Başlangıç Koşul/Koşulları:**

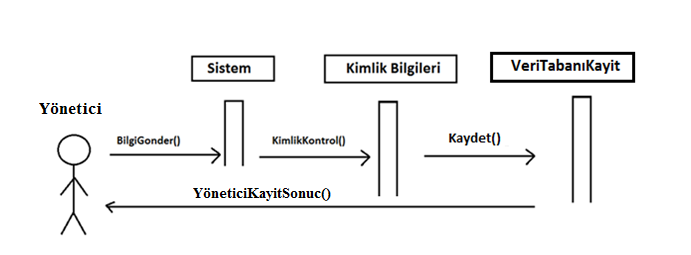
* Kayıt butonuna basılır

**5) Bitiş Koşul/Koşulları:**

* Kullanıcının sisteme kaydının tamamlanması.

**6) Kalite Koşul/Koşulları:**

* Bu işlemin 1 dakika aralığında başarı ile tamamlanması



**1) Use-Case Adı:** Merkez (Central.cs)

**2) Katılımcı Aktörler:** Kullanıcı - Hasta

**3) İş Akışı:**

* Burdan diğer UseCase ‘lere erişim mümkündür
* Burayı kontrol mekanizması olarak düşünebiliriz.

**4) Başlangıç Koşul/Koşulları:**

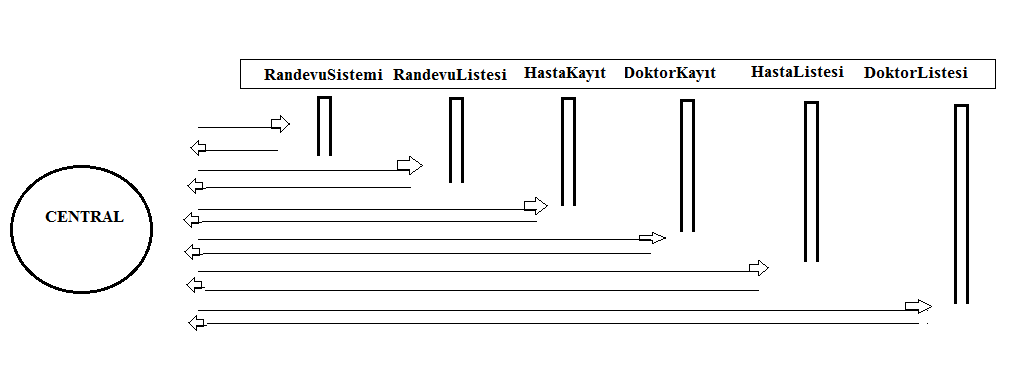
* İşlem yapmak istenilen UseCase seçilir sistem ona geçiş yapar.

**5) Bitiş Koşul/Koşulları:**

* Seçilen UseCase ‘e geçişin sağlanması
* RandevuSis – HastaList – DoktorList – RandevuList – HastaKayıt– DoktorKayıt

**6) Kalite Koşul/Koşulları:**

* Bu işlemin 3-5 saniye içerisinde tamamlanması

****

**1) Use-Case Adı:** Doktor Kayıt

**2) Katılımcı Aktörler:** Kullanıcı

**3) İş Akışı:**

* Doktora ait kimlik bilgileri ve doktorluk bilgileri sisteme girilir.

**4) Başlangıç Koşul/Koşulları**

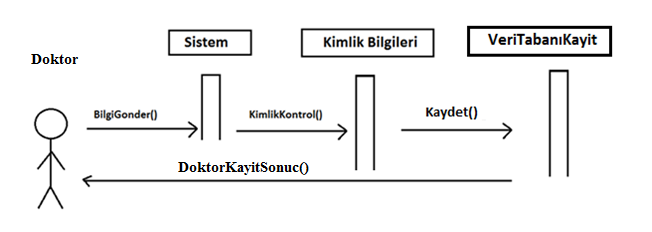
* Kayıt butonuna basılması

**5) Bitiş Koşul/Koşulları:**

* Doktorun sisteme başarılı şekilde kaydedilmesi

**6) Kalite Koşul/Koşulları:**

* Doktorun sisteme kaydının 30 sn içerisinde başarılı şekilde tamamlanması

****

**1) Use-Case Adı:** Hasta Kayıt

**2) Katılımcı Aktörler:** Kullanıcı , Hasta

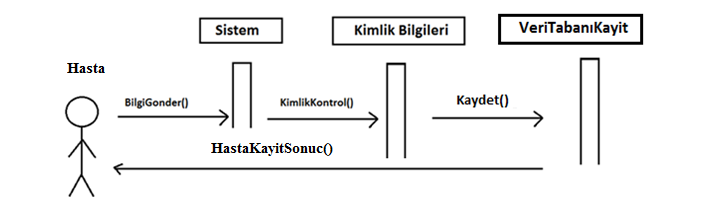
**3) İş Akışı:**

* Hasta bilgileri girilir.
* Hastanın sisteme kayıtlı olup olmadığı kontrol edilir.
* Eğer hasta sitemde kayıtlı değil ise kayıt işlemi yapılır.

**4) Başlangıç Koşul/Koşulları:** Hasta bilgileri girilmesi, kayıtlı olmadığının onaylanması.

**5) Bitiş Koşul/Koşulları:** Hastanın sisteme kaydının tamamlanması.

**6) Kalite Koşul/Koşulları:** Randevu alınmasının 2 dakikadan uzun sürmemesi.

****

**1) Use-Case Adı:** Randevu Al

**2) Katılımcı Aktörler:** Kullanıcı , Hasta

**3) İş Akışı:**

* Hasta TC ‘si girilir
* Doktor branşı ve Randevu saati seçilir

**4) Başlangıç Koşul/Koşulları**

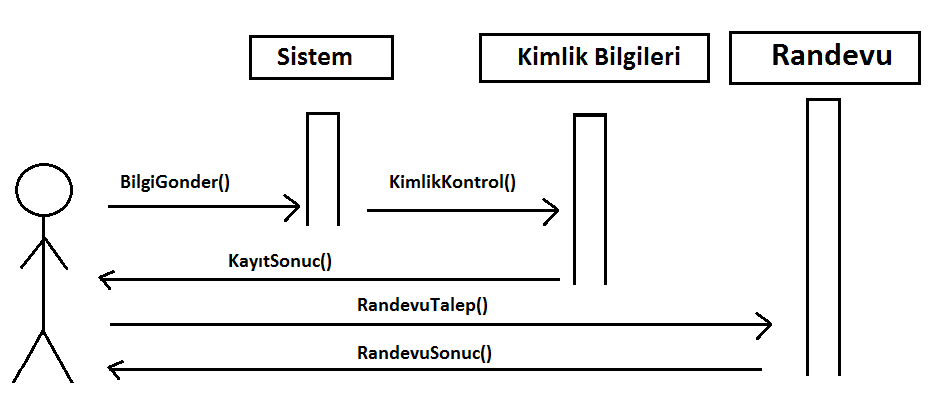
* İleri butonuna baslılarak RandevuEkle UseCase ‘i çağırılır
* Ardından RandevuAl butonuna tıklanılır.

**5) Bitiş Koşul/Koşulları:**

* Başarılı şekilde randevu alma işleminin tamamlanması

**6) Kalite Koşul/Koşulları:**

* Randevu alma işleminin 1 dakika içerisinde tamamlanması



**1) Use-Case Adı:** Doktor Listesi

**2) Katılımcı Aktörler:** Kullanıcı , Doktor

**3) İş Akışı:**

* Sistemde kayıtlı olan doktorlar listelenir
* Doktor silme/düzenleme işlemleri geçekleştirilir

**4) Başlangıç Koşul/Koşulları**

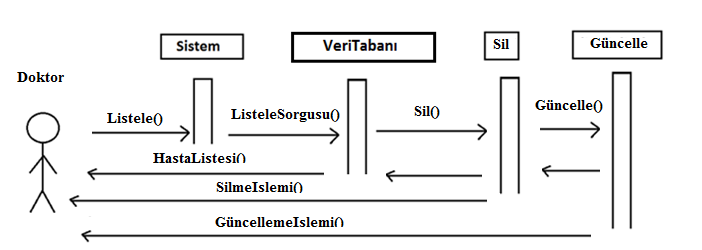
* Doktor için sil.düzenle butonuna basılması

**5) Bitiş Koşul/Koşulları:**

* Yapılan işlemin başarılı şekilde veri tabanında gerçekleştirilmesi

**6) Kalite Koşul/Koşulları:**

* Yapılan işlemin 5 sn içersinde tamamlanması

****

**1) Use-Case Adı:** Hasta Listesi

**2) Katılımcı Aktörler:** Kullanıcı , Hasta

**3) İş Akışı:**

* Sistemde kayıtlı olan doktorlar listelenir
* Hasta silme/düzenleme işlemleri geçekleştirilir

**4) Başlangıç Koşul/Koşulları**

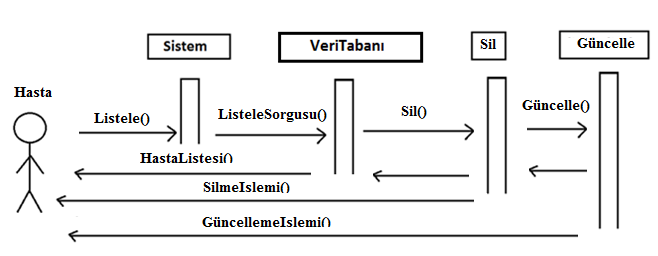
* Hasta için sil/düzenle butonuna basılması

**5) Bitiş Koşul/Koşulları:**

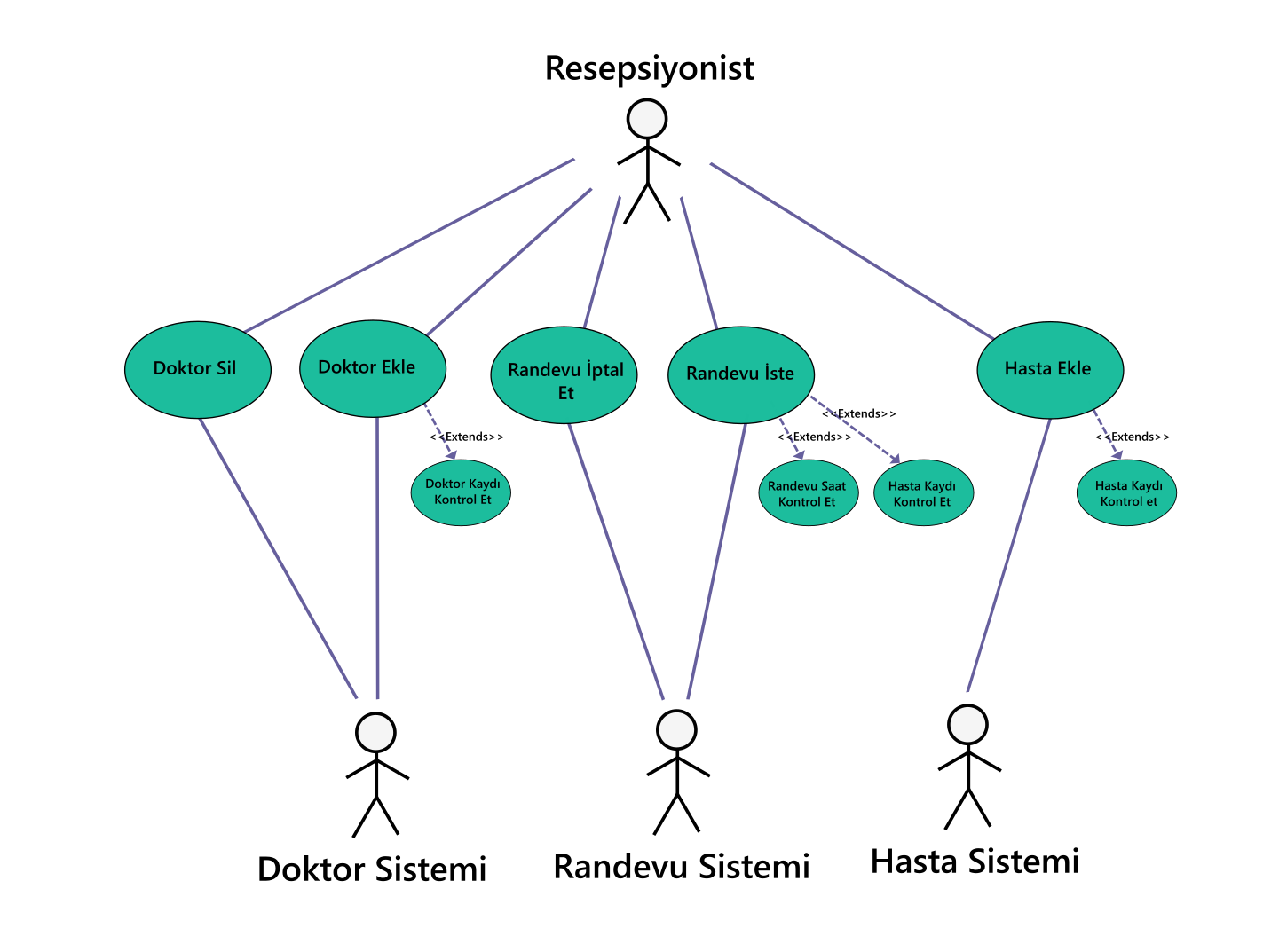
* Yapılan işlemin başarılı şekilde veri tabanında gerçekleştirilmesi

**6) Kalite Koşul/Koşulları:**

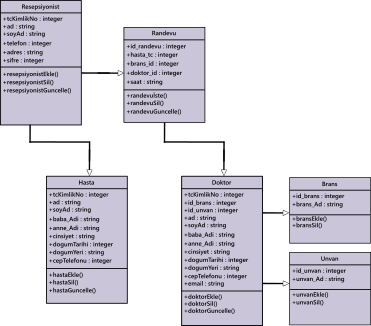
* Yapılan işlemin 5 sn içersinde tamamlanması

****

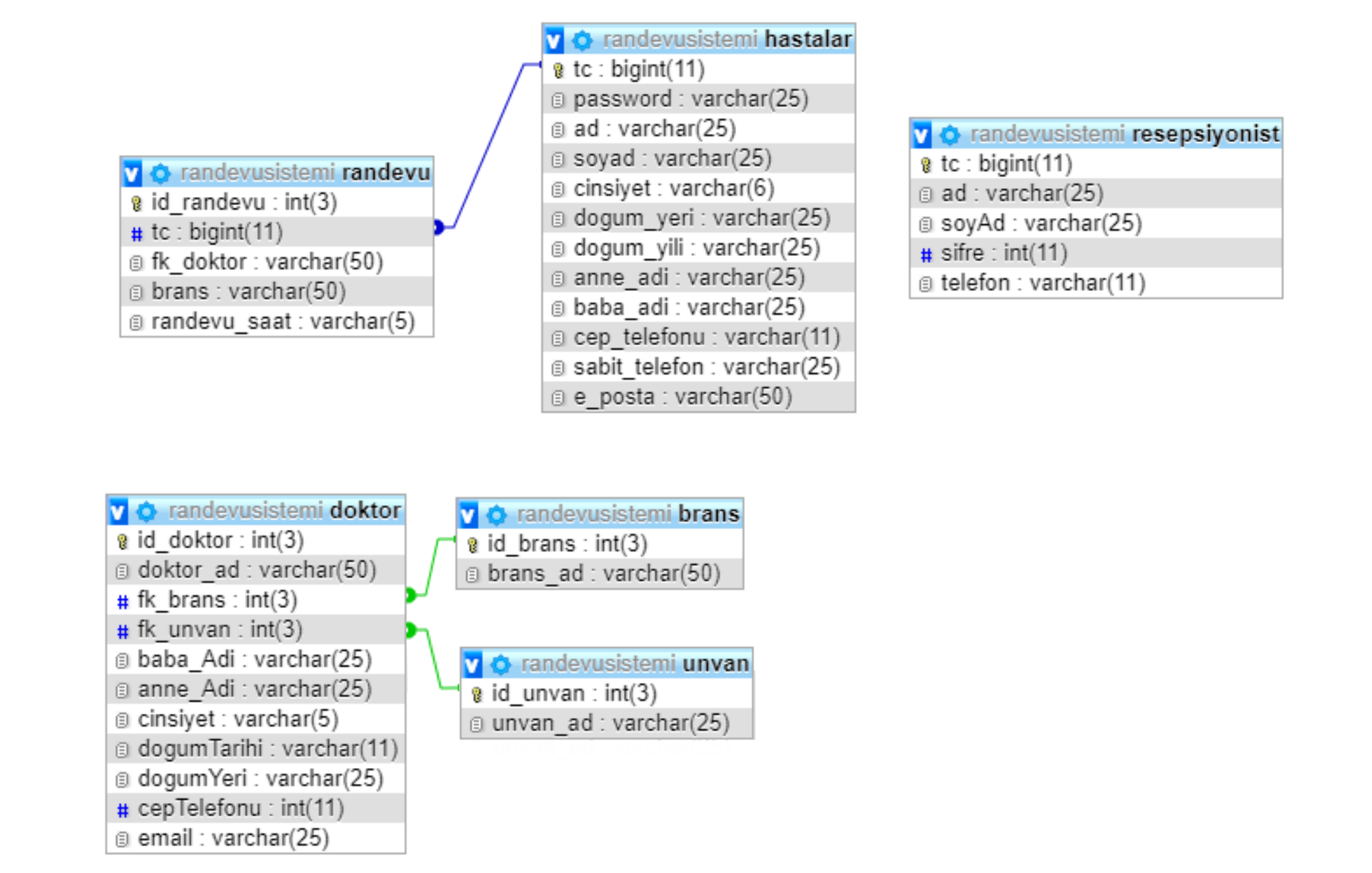
**Use-Case Grafik Tabanlı Diyagram**



**Class Diyagramı**

****

**Veri Tabani Relation Diyagramı**

****

**Pert Chart**

