****

**BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**NESNE YÖNELİMLİ ANALİZ VE TASARIM DERSİ**

**PROJE RAPORU**

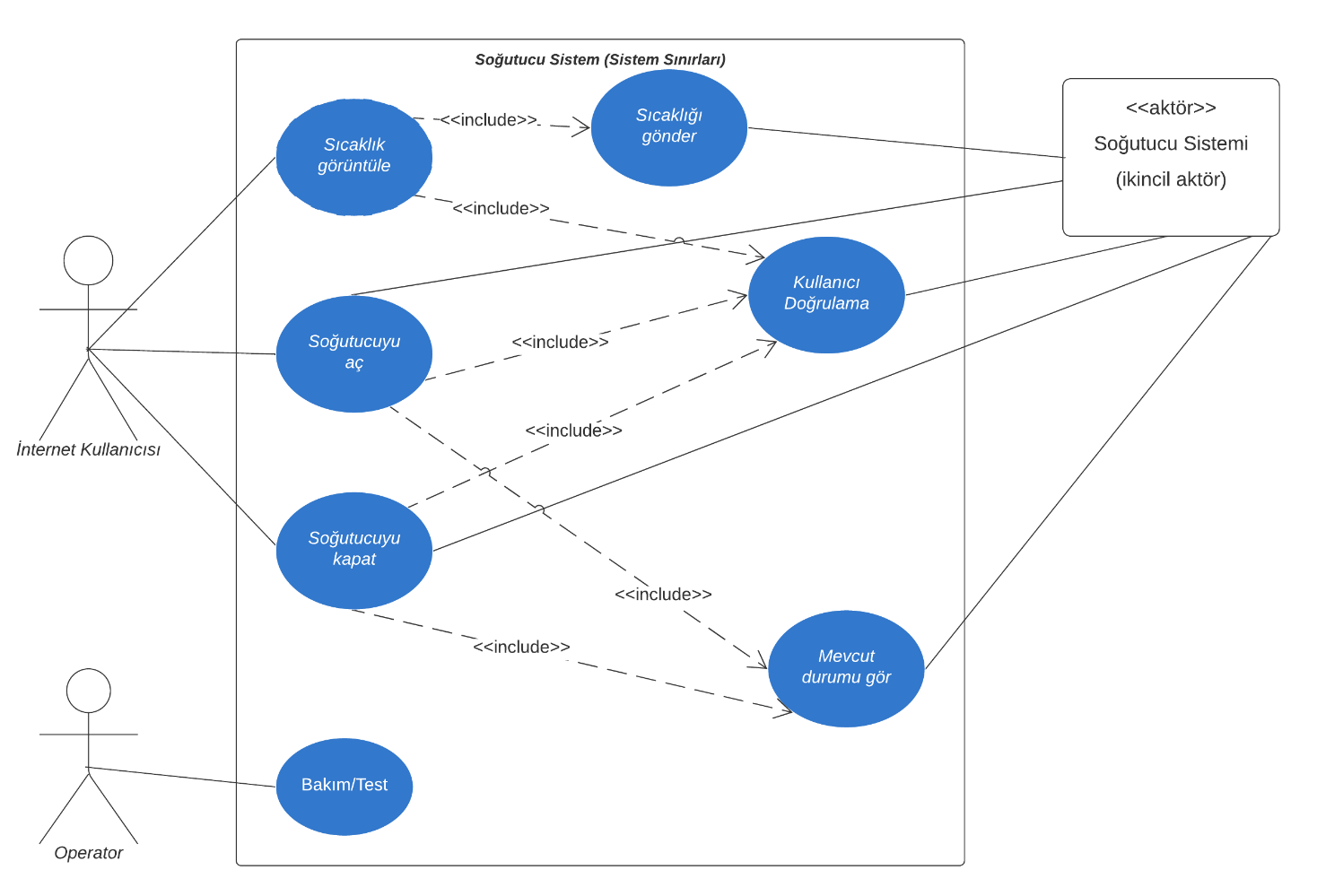
**ÖĞRENCİ:** Yunus Emre AKINCI

**ÖĞRENCİ NO:** B201210101

**ŞUBE:** 1B

**MAİL:** <yunus.akinci1@ogr.sakarya.edu.tr>

**a-)Kullanıcı Durumu (Use Case) Diyagramı**



**b.1-) Sıcaklık Görüntüleme Kullanım Durumu Ayrıntıları**

* Kullanım durumu tanımı:
  + Kullanım Adı: Sıcaklık Görüntüleme
  + Ağ ara yüzünü kullanarak sıcaklık görüntüleme işlemini tanımlar.
  + 15.04.2022, 16.04.2022 v1.0.0 emreaknci
* İlgili aktörler: İnternet kullanıcısı, Soğutucu Sistem(ikincil)
* Giriş koşulu: Soğutucu sistemi bulunan kullanıcı soğutucunun internet sitesine girer.
* Çıkış koşulu: Kullanıcı işlemini tamamlar.
* Özel gereksinimler: UI gereksinimleri, İşlem gecikmesi en fazla 1 dk. olmalı, 24 saat çalışmalı
* Olay akışı:

Ana Olay Akışı (Başarılı)

* 1. Kullanıcıdan adı istenir.

Not: Kırmızı renkli işlem basamakları kullanıcı Doğrulama (erişim denetimi) kullanım durumunu ifade eder.

* 1. Uygulama kullanıcı doğrulaması yapar.
  2. Kullanıcıdan şifresi istenir.
  3. Kullanıcı doğrulama ve yetki kontrolü için soğutucu sistemine istek gönderilir.
  4. Sistem erişim isteğini kabul eder.
  5. Uygulama kullanıcıya sıcaklık değerini gösterir.

Alternatif Olay Akışları

A1. Kullanıcı doğrulanamadı (2)

1. Uygulama sonlanır.

A2. Yanlış şifre (4)

1. 3’ten az kez yanlış ise yeniden gir.
2. 3 kez yanlış girilmiş ise kullanıcıya bilgi ver.
3. Uygulama sonlanır.

**b.2-) Soğutucuyu Açma Kullanım Durumu Ayrıntıları**

* Kullanım durumu tanımı:
  + Soğutucuyu Açma
  + Ağ ara yüzünü kullanarak soğutucuyu çalıştırma işlemini tanımlar.
  + 17.04.2022, 18.04.2022 v1.0.1 emreaknci
* İlgili aktörler: İnternet kullanıcısı, Soğutucu Sistem(ikincil)
* Giriş koşulu: Soğutucu sistemi bulunan kullanıcı soğutucunun internet sitesine girer.
* Çıkış koşulu: Kullanıcı işlemini tamamlar.
* Özel gereksinimler: UI gereksinimleri, İşlem gecikmesi en fazla 1 dk. olmalı, 24 saat çalışmalı
* Olay akışı:

Ana Olay Akışı (Başarılı)

* 1. Kullanıcıdan adı istenir.

Not: Kırmızı renkli işlem basamakları kullanıcı Doğrulama (erişim denetimi) kullanım durumunu ifade eder.

* 1. Uygulama kullanıcı doğrulaması yapar.
  2. Kullanıcıdan şifresi istenir.
  3. Kullanıcı doğrulama ve yetki kontrolü için soğutucu sistemine istek gönderilir.
  4. Sistem erişim isteğini kabul eder.
  5. Uygulama soğutucuyu çalıştırır.

Alternatif Olay Akışları

A1. Kullanıcı doğrulanamadı (2)

1. Uygulama sonlanır.

A2. Yanlış şifre (4)

1. 3’ten az kez yanlış ise yeniden gir.
2. 3 kez yanlış girilmiş ise kullanıcıya bilgi ver.
3. Uygulama sonlanır.

A3. Soğutucu çalıştırılamadı (6)

1. Soğutucu zaten çalışıyorsa işlem gerçekleşmez. Uygulama sonlanır.
2. Teknik bir sıkıntı varsa kullanıcıya bilgi verilir. Uygulama sonlanır.

**b.3-) Soğutucuyu Kapatma Kullanım Durumu Ayrıntıları**

* Kullanım durumu tanımı:
  + Soğutucuyu Kapatma
  + Ağ ara yüzünü kullanarak soğutucuyu kapatma işlemini tanımlar.
  + 19.04.2022, 20.04.2022 v1.0.2 emreaknci
* İlgili aktörler: İnternet kullanıcısı, Soğutucu Sistem(ikincil)
* Giriş koşulu: Soğutucu sistemi bulunan kullanıcı soğutucunun internet sitesine girer.
* Çıkış koşulu: Kullanıcı işlemini tamamlar.
* Özel gereksinimler: UI gereksinimleri, İşlem gecikmesi en fazla 1 dk. olmalı, 24 saat çalışmalı
* Olay akışı:

Ana Olay Akışı (Başarılı)

* 1. Kullanıcıdan adı istenir.

Not: Kırmızı renkli işlem basamakları kullanıcı Doğrulama (erişim denetimi) kullanım durumunu ifade eder.

* 1. Uygulama kullanıcı doğrulaması yapar.
  2. Kullanıcıdan şifresi istenir.
  3. Kullanıcı doğrulama ve yetki kontrolü için soğutucu sistemine istek gönderilir.
  4. Sistem erişim isteğini kabul eder.
  5. Uygulama soğutucuyu kapatır.

Alternatif Olay Akışları

A1. Kullanıcı doğrulanamadı (2)

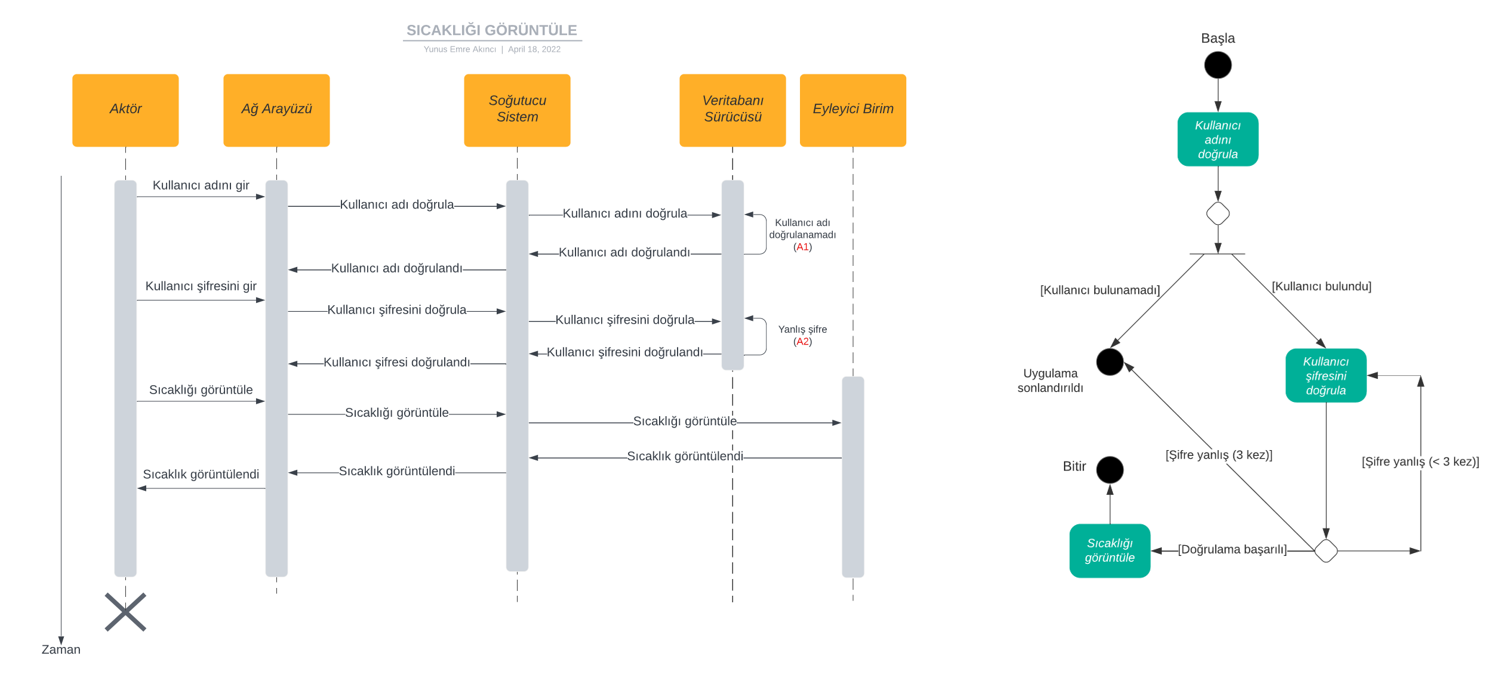
1. Uygulama sonlanır.

A2. Yanlış şifre (4)

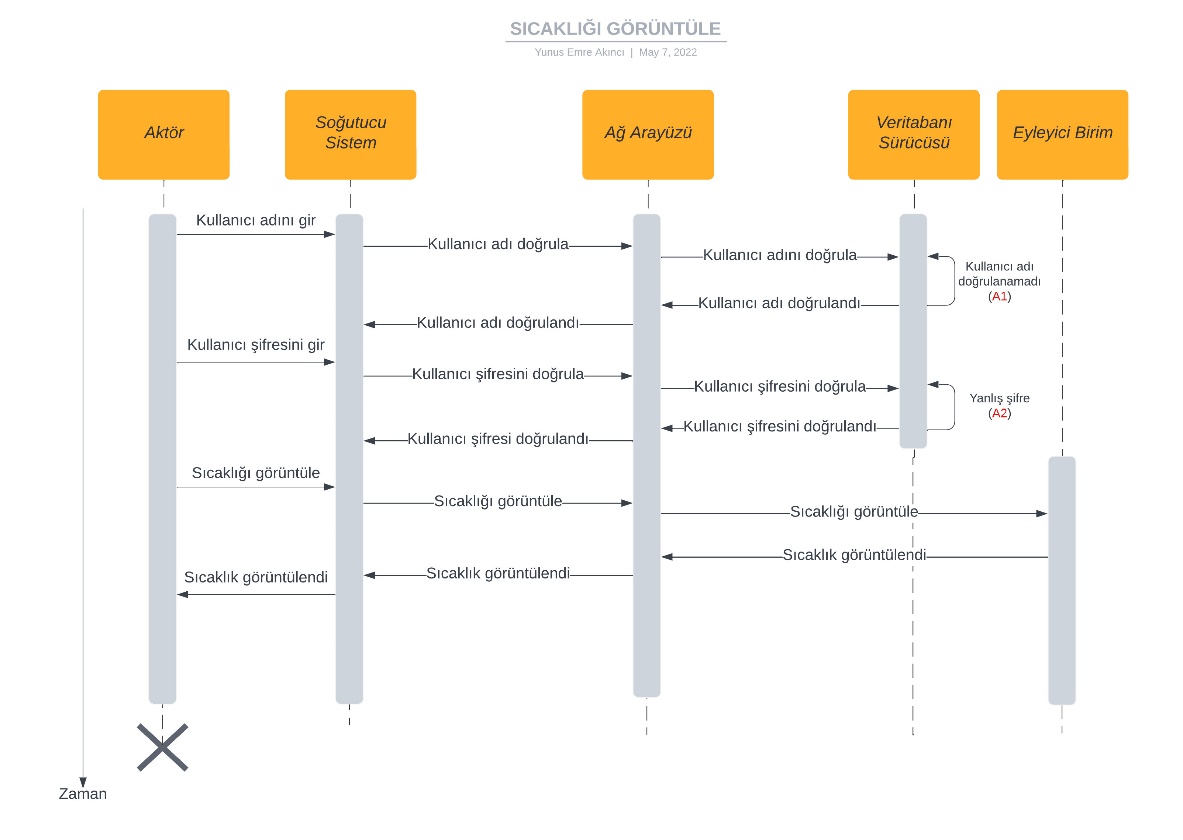
1. 3’ten az kez yanlış ise yeniden gir.
2. 3 kez yanlış girilmiş ise kullanıcıya bilgi ver.
3. Uygulama sonlanır.

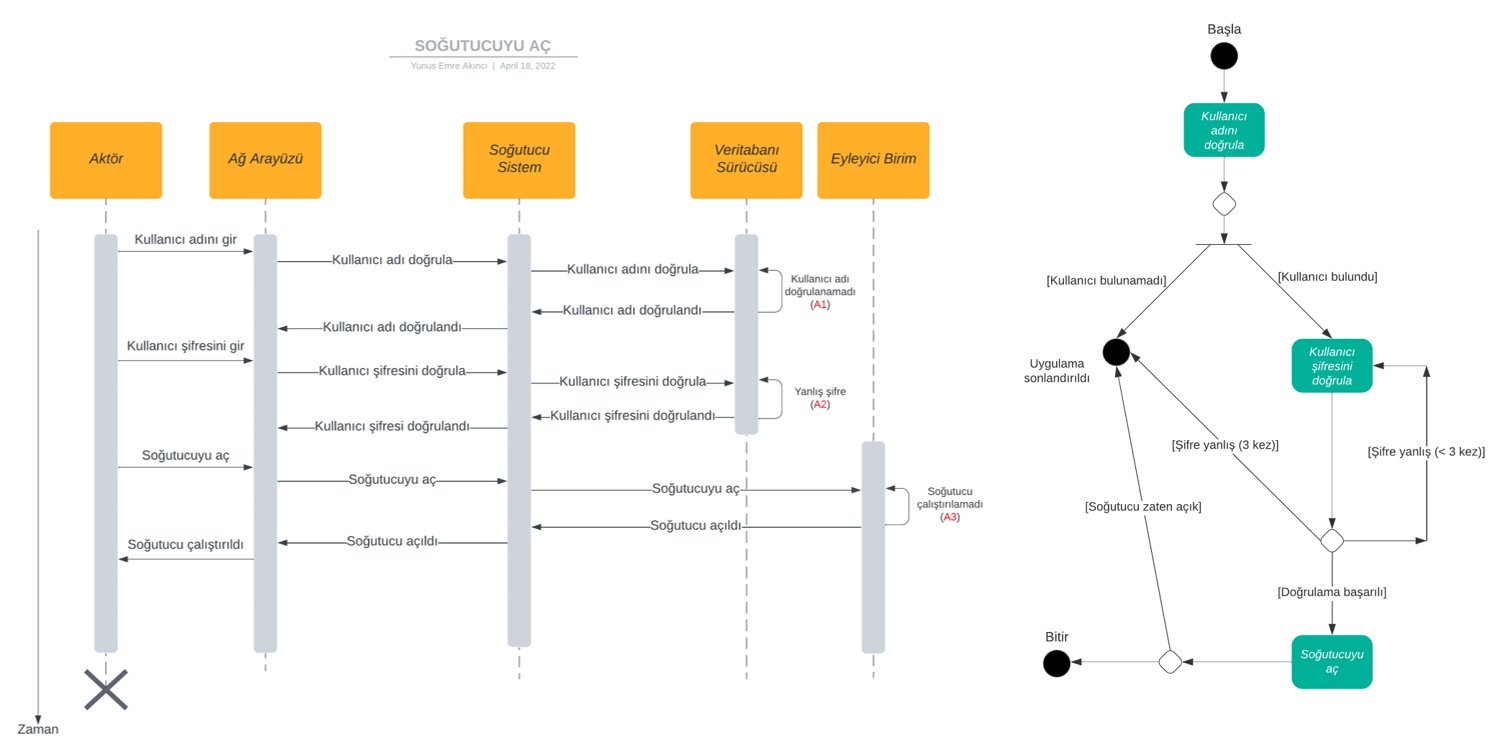
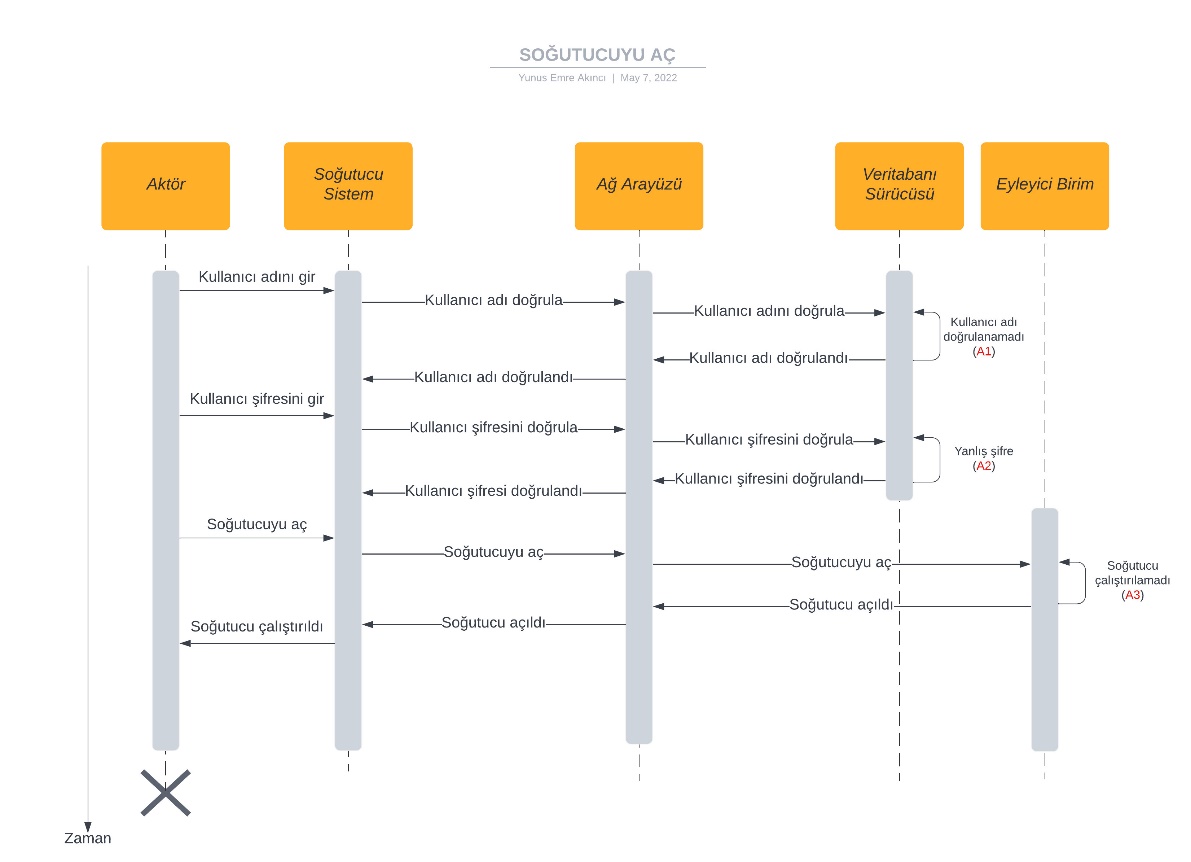
A.3 Soğutucu kapatılamadı (6)

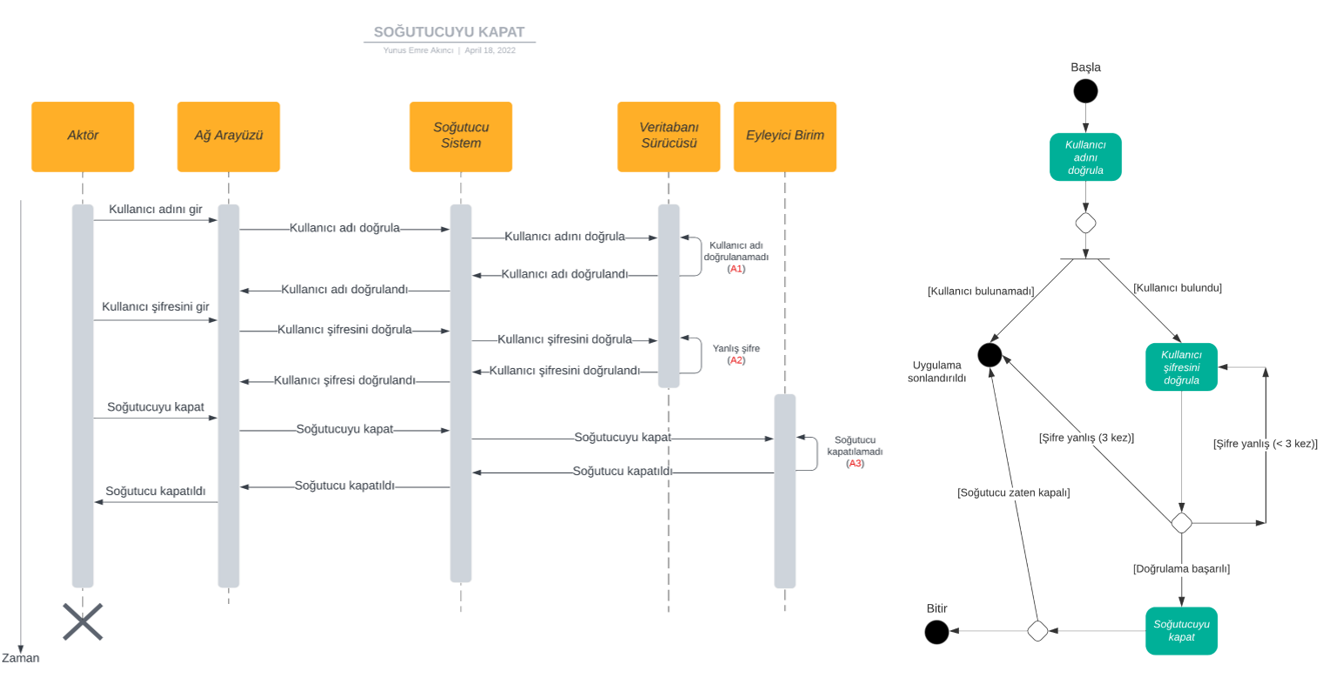
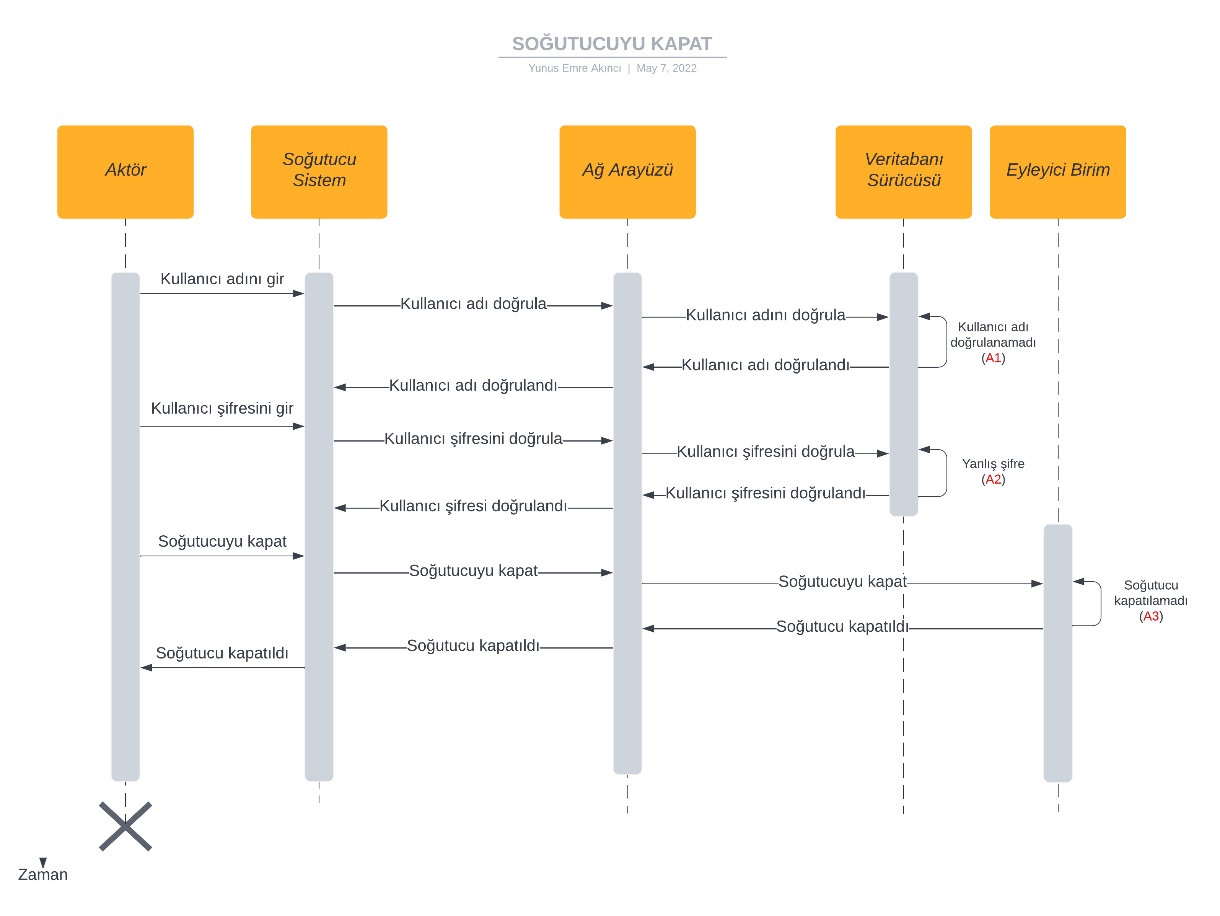
1. Soğutucu zaten kapalıysa işlem gerçekleşmez. Uygulama sonlanır.
2. Teknik bir sıkıntı varsa kullanıcıya bilgi verilir. Uygulama sonlanır.

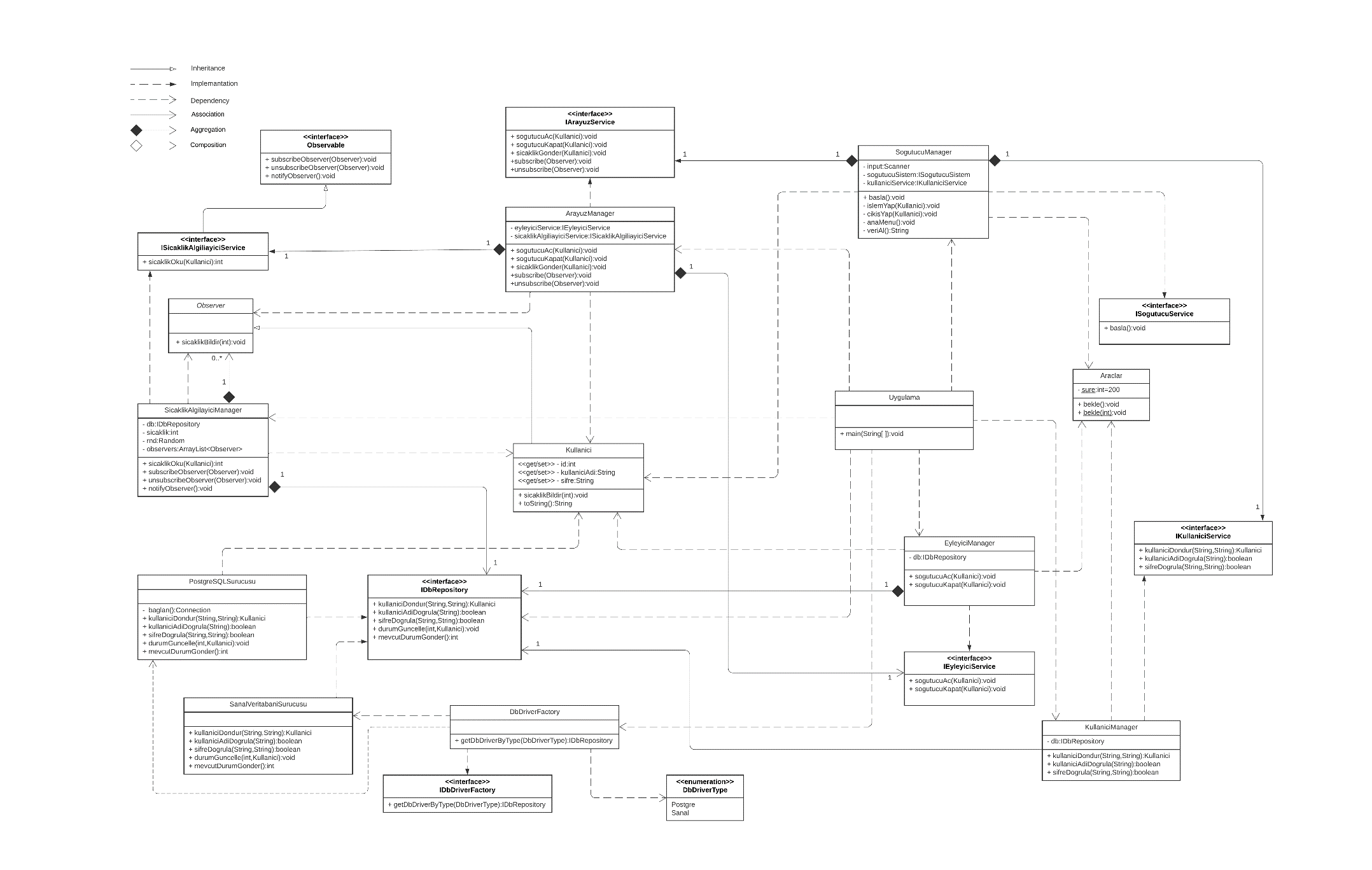
**c-) Sıralama Şemaları**

**d-) Etkinlik Şemaları**

****





**e-) Sınıf Şeması**



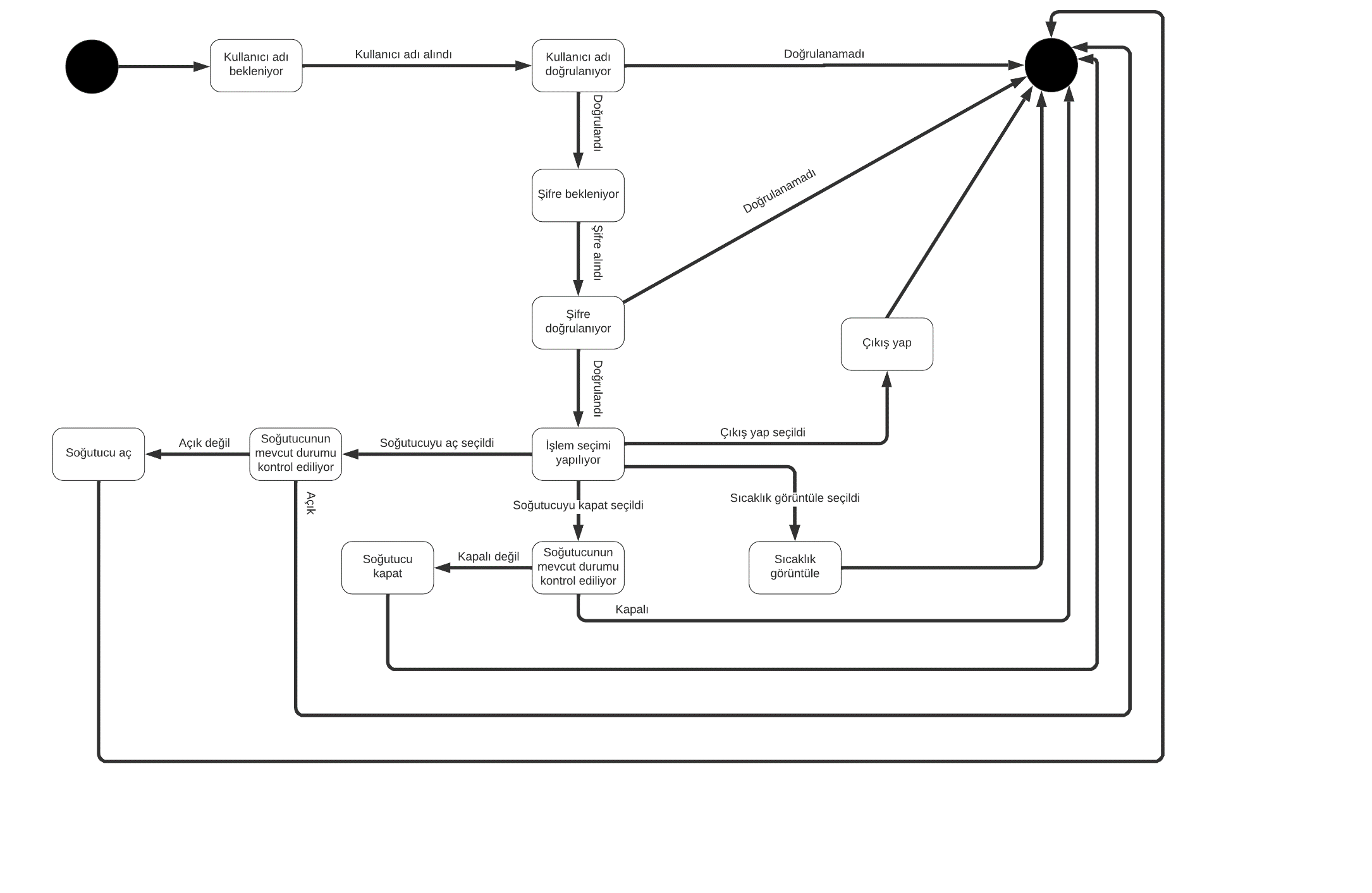






tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu**f-) CRC Kartları**

**g-) Durum Şeması**

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu**h-) Kullanıcı Doğrulama Ekranı**

Kullanıcıdan önce kullanıcı adı isteniyor ve veri tabanında aranıyor. Kullanıcı bulunamazsa uygulama tekrar kullanıcı adı istiyor.

Kullanıcı adı doğrulandığında kullanıcı şifresi isteniyor ve kullanıcının şifresi ile karşılaştırılıyor. Şifreler uyuşmuyorsa tekrardan şifre isteniyor. 3 kere yanlış şifre girilirse uygulama sonlanıyor (hesabın bloke olması gerçekleştirilmedi).

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu**i.1-)Soğutucu açma/kapatma ekranı**

Soğutucu aç seçeneği seçildiğinde soğutucunun mevcut durumu ara yüz katmanı ile veri tabanındaki kayıtlardan kontrol edilir. Bu kontrol sonucu soğutucu sisteme mevcut durum bildirilir. Soğutucu sistem zaten açıksa bu işlem gerçekleşmez. Ana menüye geri dönülür.

Aynı işlemler benzer mantıkla soğutucuyu kapatma seçeneği için de geçerlidir.

**i.2-)Sıcaklığı görüntüleme ekranı**

(Sıcaklık algılayıcısı çağrıldığında yapıcı metodunda 15-25 arasında rastgele bir değer üretir)

Sıcaklık, soğutucu açıkken her görüntülenişinde 5 azalır.

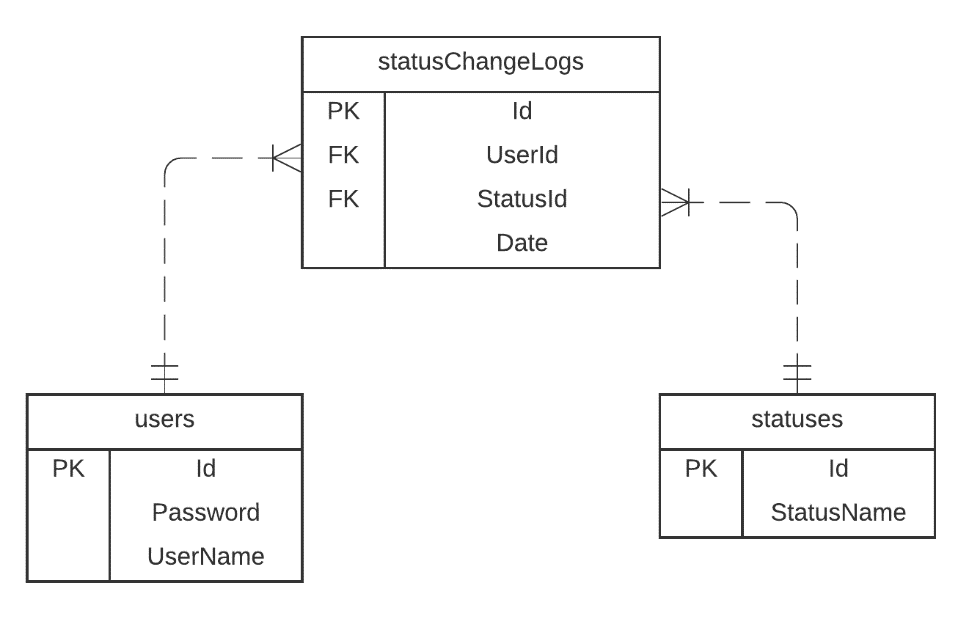
Sıcaklık, soğutucu kapalıyken her görüntülenişinde 5 artar.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldumetin içeren bir resim

Açıklama otomatik olarak oluşturuldu

**j-) Veri Tabanı Görüntüleri**

****

**Veri tabanı şeması**

**“users” tablosu**

metin içeren bir resim

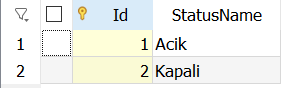
Açıklama otomatik olarak oluşturuldu

**“statusChangeLogs” tablosu**

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

**“statuses” tablosu**



**k-) Dependency Inversion**

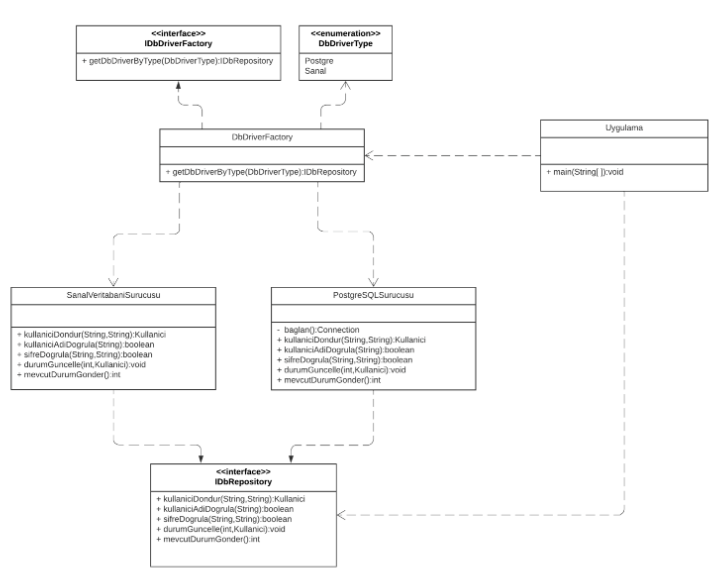
**Dependency Inversion ilkesi şöyle der:**

* Yüksek seviyeli modüller, düşük seviyeli modüllerden hiçbir şey almamalıdır. Her ikisi de soyutlamalara bağlı olmalıdır (örneğin, arayüzler).
* Soyutlamalar ayrıntılara bağlı olmamalıdır. Ayrıntılar (somut uygulamalar) soyutlamalara bağlı olmalıdır.

metin içeren bir resim

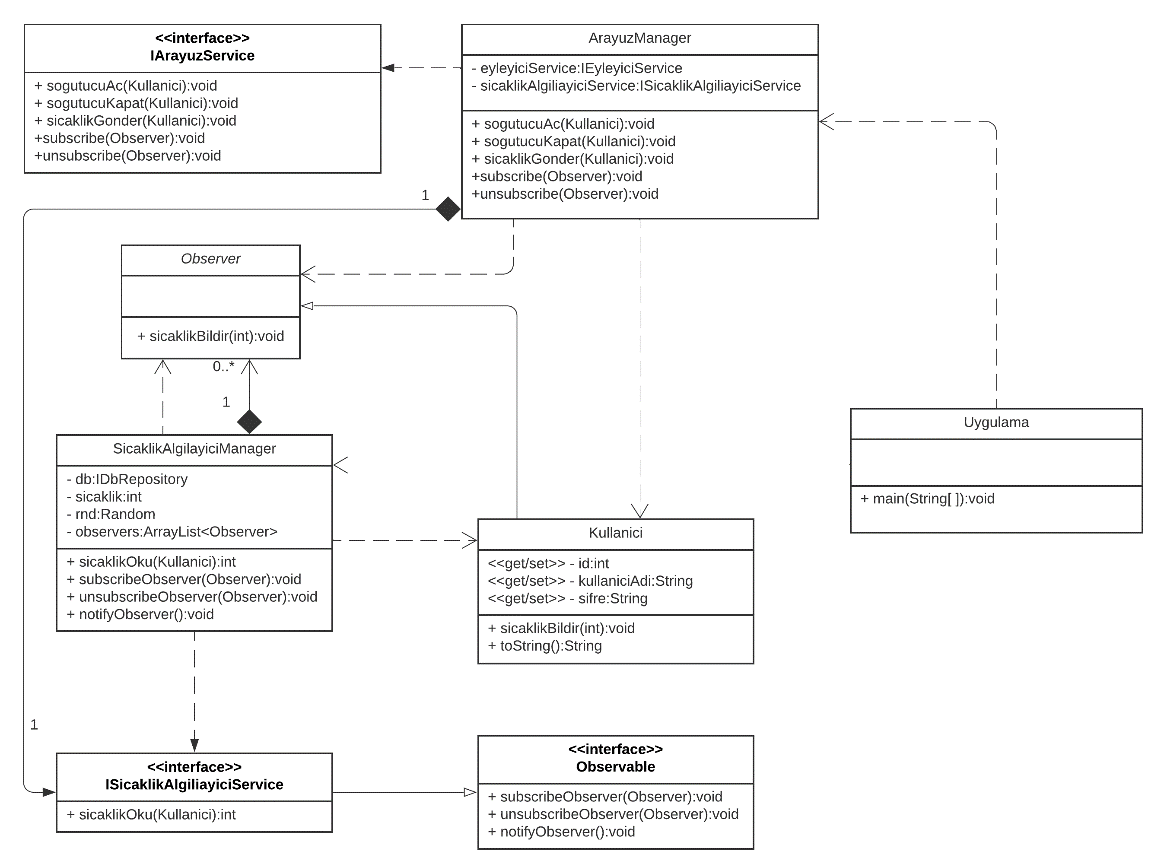
Açıklama otomatik olarak oluşturuldu

Uygulamadan örnek vermek gerekirse EyleyiciManager sınıfında yer alan fonksiyonlar bir veri tabanı bağlantısına ihtiyacı duyar. Ben bu ödevde PostgreSQL(PostgreSQLSurucusu sınıfı ile temsil ettim) kullandım ama yarın veri tabanımı değiştirmek isteyebilirim. Yani benim PostgreSQLSurucusu sınıfına bağlı olmamam lazım. Bunu sağlamak için alt sınıftaki fonksiyonların imzasını içeren bir IDbRepository ara yüzü kullandım. EyleyiciManager sınıfını dışarıdan çağırdığımda parametre olarak IDbRepository referansı tutan bir nesne vermem yeterli olacaktır.

**l.1-) Factory Method Tasarım Deseni**

Factory Method, üst sınıfta nesneler oluşturmak için bir arabirim sağlayan, ancak alt sınıfların oluşturulacak bu nesne türünü değiştirmesine izin veren bir yaratımsal (creational) desen türüdür.

Ödevimde bu deseni IDbRepository’den oluşturulabilecek nesneler için kullandım.

**l.2-) Observer Tasarım Deseni**

Observer tasarım deseni, birden fazla nesneyi takip ettikleri başka bir nesnede gerçekleşen olaylarla ilgili bilgilendirmeyi sağlayan bir abonelik mekanizması oluşturmayı amaçlar.

Ödevimde bu deseni sıcaklık değeri her okunduğunda kullanıcıya haber vermek için kullandım.

**m-) Uygulama kaynak kodları**

GitHub: <https://github.com/emreaknci/cs-SakaryaUniversity/tree/main/BSM204-Nesne%20Y%C3%B6nelimli%20Analiz%20ve%20Tasar%C4%B1m>