



# SAKARYA ÜNİVERSİTESİ

SAKARYA ÜNİVERSİTESİ

BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

NESNE YÖNELİMLİ ANALİZ VE TASARIM PROJESİ

Adı ve Soyadı: Yavuzhan ALBAYRAK, Berke ÖZKAN

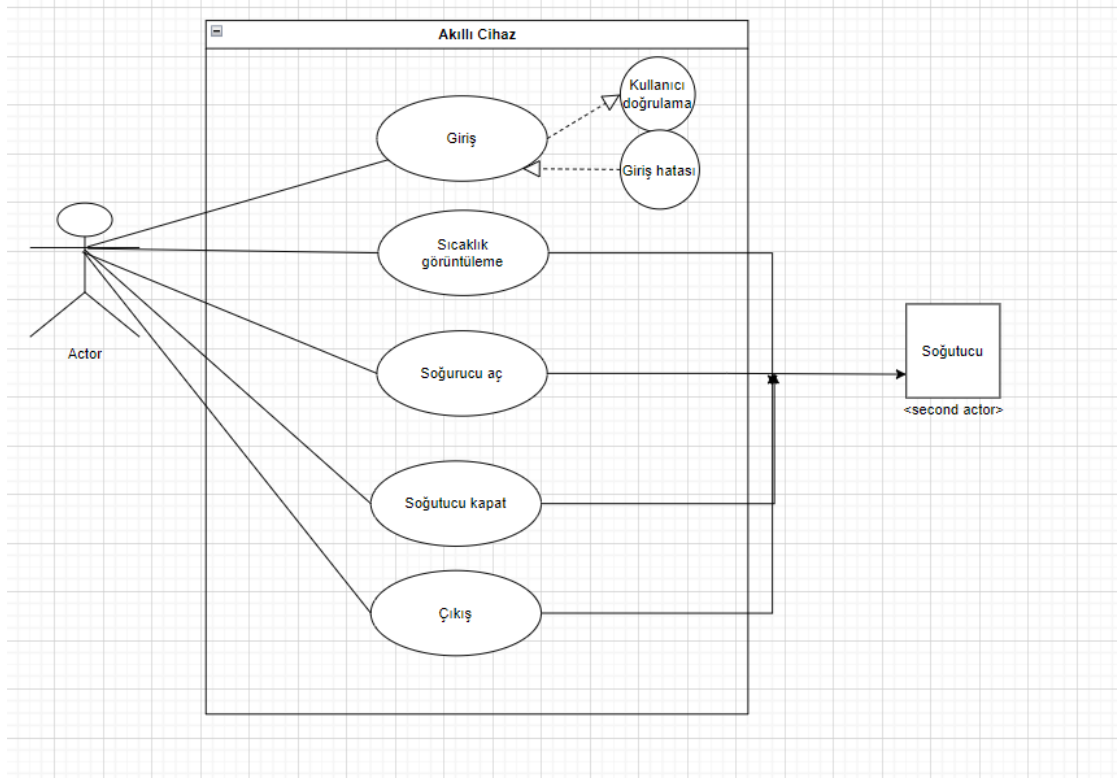
Öğrenci numarası: G201210008, G201210017

Grup: 2. Öğretim B Grubu

Dersi veren öğretim elemanı: Prof.Dr. Celal Çeken

## KULLANIM DURUMLARI

### 2. "Sıcaklık Görüntüle" Kullanım Durumu



## 2.KULLANIM DURUMLARI

### 1. "Sıcaklık Görüntüle" Kullanım Durumu

Hazırlayan: Yavuzhan ALBAYRAK, Berke ÖZKAN

>>Sıcaklık Görüntüle:

-Sıcaklık görüntüleme işlemini tanımlar.

-7.05.2022

11.05.2022

v1.1.0 yavuz(kullanici)

>>İlgili Aktörler: İnternet Kullanıcısı, Arayüz, Sıcaklık algılayıcı

>>Giriş Koşulu: Kullanıcı internet üzerinden sisteme girer. Arayüzden Soğutucuyu seçer. Kullanıcının internete bağlı bir soğutucusu vardır.

>>Çıkış Koşulu: Kullanıcı işlemini tamamlar.

>>Olay Akışı:

- Kullanıcı sisteme girer.
- Sistem kullanıcıdan giriş yapmasını ister.
- Kullanıcı bilgilerini girer.
- Veri tabanı kullanıcıyı doğrular.
- Arayüzü ekrana 4 seçenek listeler.
- Internet kullanıcısı sıcaklık görüntülemeyi seçer.
- Sistem talebi iletir.
- Talep sıcaklık algılayıcı üzerinden programı çalıştırır.
- Eyleyici birimi isteği alır. Arayüzüne gönderir.
- Arayüzü isteği alır.
- Arayüzü işlemi gerçekleştirir.
- Sıcaklık görüntülenir.

Alternatif Olay Akışı:

Alternatif1: Kullanıcı adı doğrulanamadı.

- Sistem hata mesajı verir.
- Tekrar giriş yapılmasını ister.
- İşlem sonlandırılır.

Alternatif2: Şifre doğrulanamadı.

- Sistem hata mesajı verir.
- Tekrar giriş yapılmasını ister.
- İşlem sonlandırılır.

Özel Gereksinimler:

- Sistem 7/24 Çalışır olmalıdır.
- Sistem hızlı olmalıdır.
- Sistem güvenilir olmalıdır.
- Soğutucu internete bağlı olmalıdır.

## 2. "Soğutucuyu Çalıştırılması" Kullanım Durumu

Hazırlayan: Yavuzhan ALBAYRAK, Berke ÖZKAN

>> Soğutucuyu Kapat

-Soğutucuyu kapatma işlemi tanımlar.

-07.05.2022, 11.05.2022

v1.1.0 berke(kullanici)

>>İlgili Aktörler: İnternet Kullanıcısı, Eyleyici.

>>Giriş Koşulu: Kullanıcı internet üzerinden sisteme girer. Arayüzden Soğutucuyu seçer. Kullanıcının internete bağlı bir soğutucusu vardır.

>>Çıkı Koşulu: kullanıcı işlemi tamamlar.

>>Olay Akışı:

- Kullanıcı sisteme girer.
- Sistem kullanıcıdan giriş yapmasını ister.
- Kullanıcı bilgilerini girer.
- Sistem kullanıcıyı doğrular.
- Ağ ara yüzü ekrana 4 seçenek listeler.
- Internet kullanıcısı soğutucuyu açma/kapatma seçer.
- Sistem talebi iletir.
- Talep Arayüz üzerinden Eyleyici birimine iletilir.
- Eyleyici işlem birimi talebi alır.
- Eyleyici işlemi gerçekleştirir.
- Arayüzü kullanılarak ekrana soğutucu açıldı/ kapatıldı yazdırılır.
- Kullanıcı ekrandan bilgiyi okur.
- Soğutucu kapatılır.

>>Alternatif Olay Akışı:

Alternatif1: Kullanıcı adı doğrulanamadı.

-Sistem hata mesajı verir.

-Tekrar giriş yapılmasını ister.

-İşlem sonlandırılır.

Alternatif2: Şifre doğrulanamadı.

-Sistem hata mesajı verir.

-Tekrar giriş yapılmasını ister.

-İşlem sonlandırılır.

Özel Gereksinimler:

-Sistem 7/24 Çalışır olmalıdır.

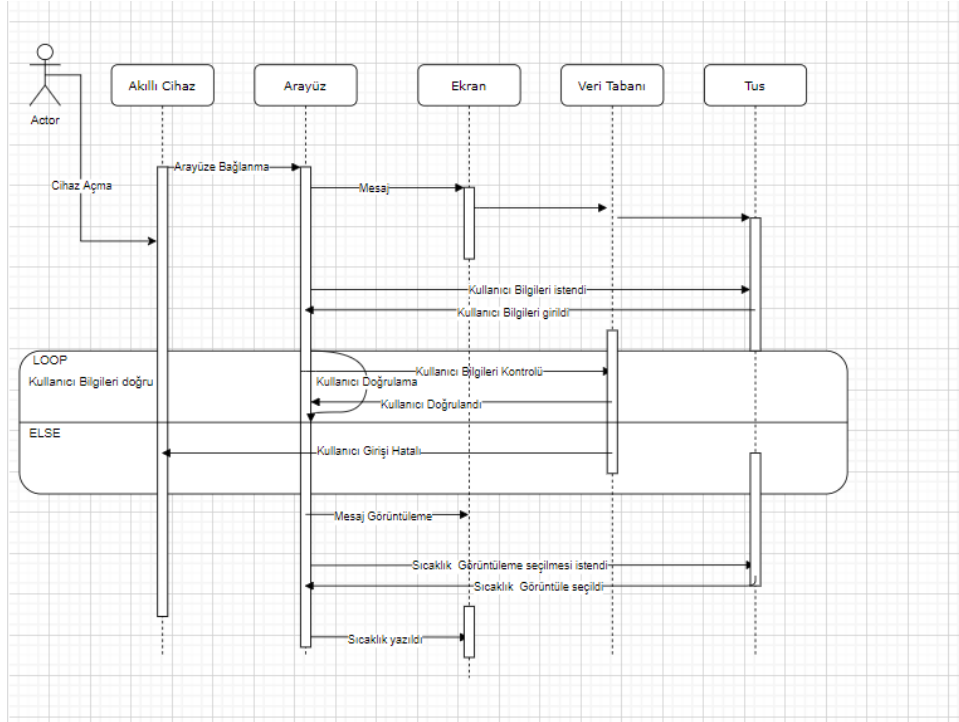
-Sistem hızlı olmalıdır.

-Sistem güvenilir olmalıdır.

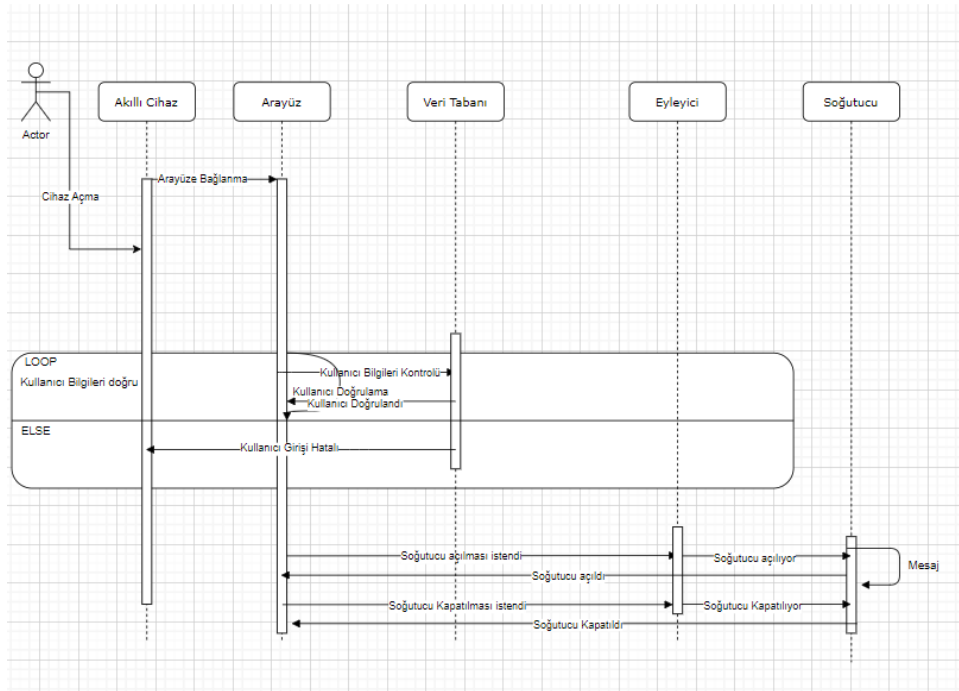
-Soğutucu internete bağlı olmalıdır.

### 3. Sıralama Şeması (sequence diagram):

#### A) Sıcaklığın Görüntülenmesi

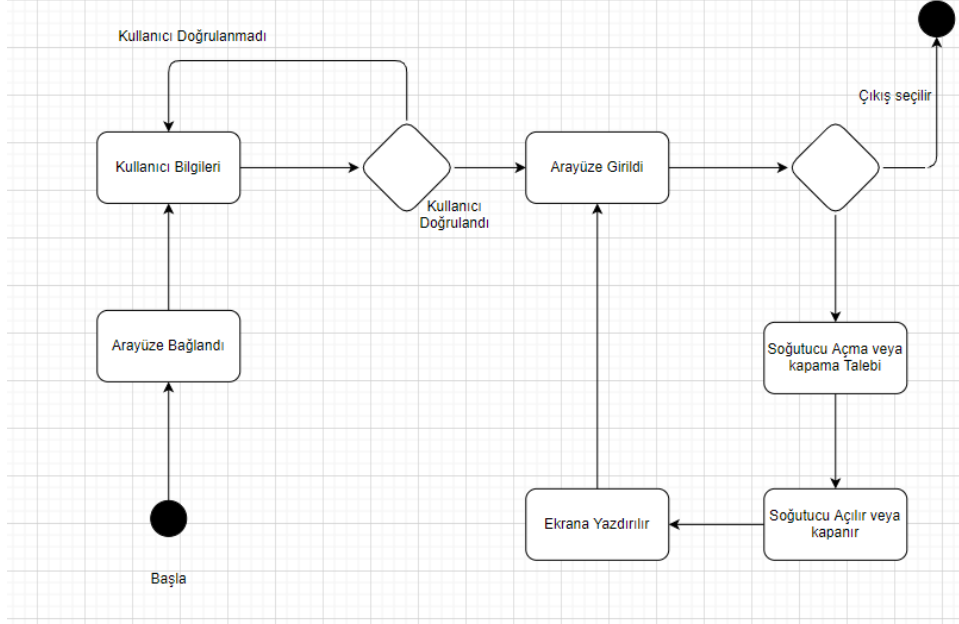


#### B) Soğutucunun Çalışması



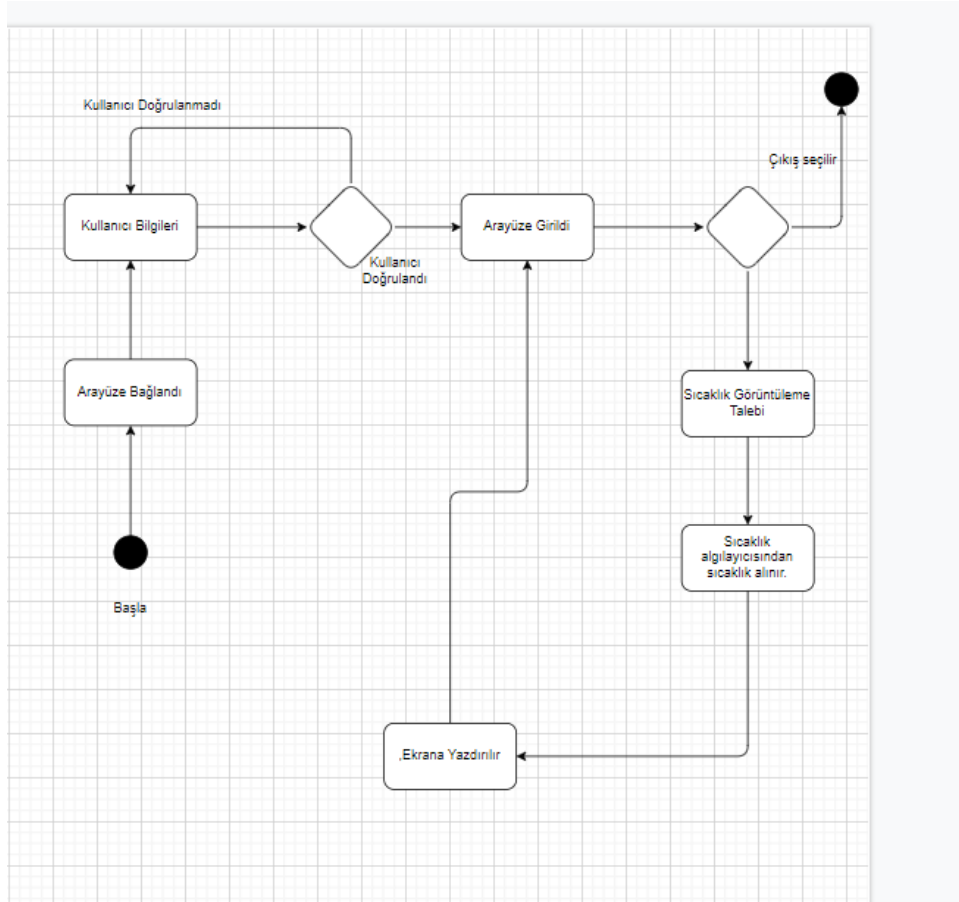
#### 4. Etkinlik Şeması (activity diagram):

##### a) Soğutucunun Çalıştırılması Etkinlik Şeması

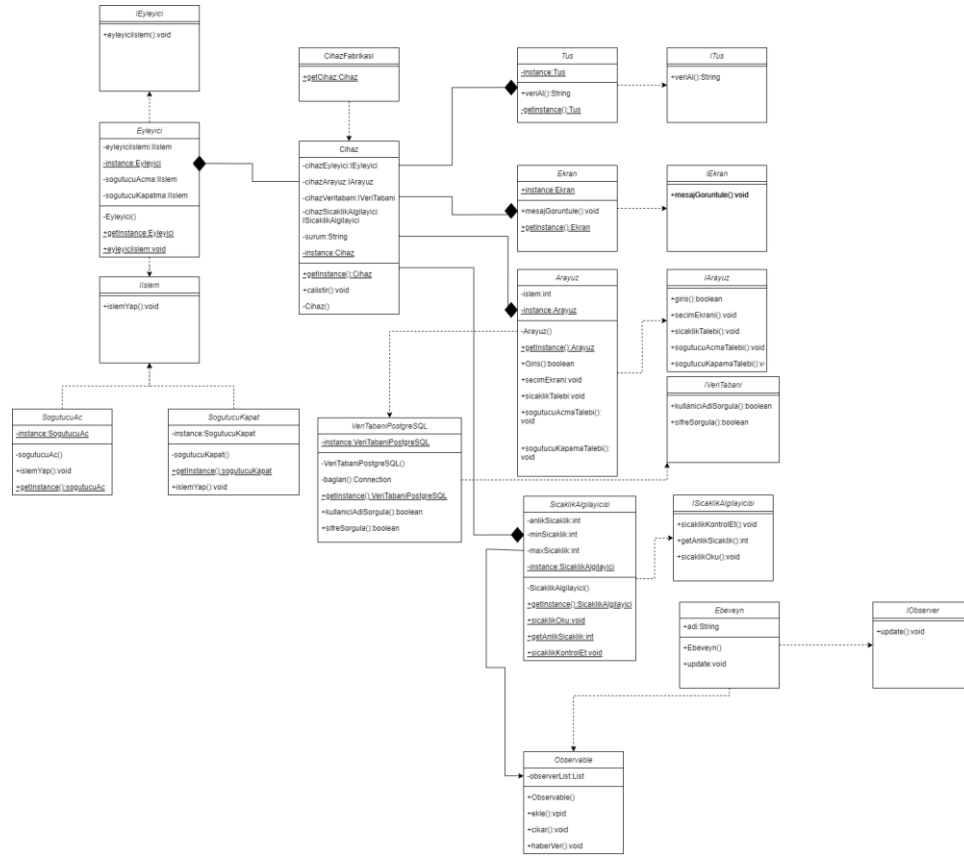




## B) Sıcaklığın Görüntülenme



## 5. Sınıf Şeması:

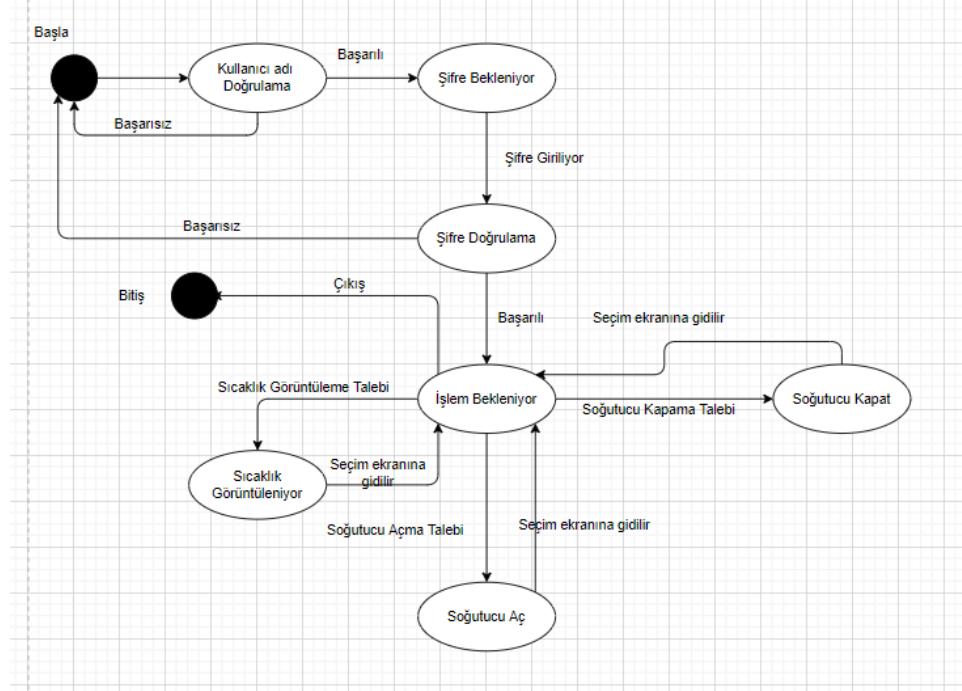


## 6. CRC kartları

Cihaz	
Sorumluluk	İş Birliği Yapılan Sınıf
Cihazın Başlatılması	Kendisi
Giriş Yap	Arayuz
İşlem Yap	Eyleyici
Sıcaklık Ölç	SicaklikAlgilayicisi
Ana ,Menü ve İşlem Seçimi	Arayuz

Eyleyici	
Sorumluluk	İş Birliği Yapılan Sınıf
Soğutucu Aç	SogutucuAc
Soğutucu Kapat	SogutucuKapat

## 7. Durum Makinesi Diyagramı



## 8. Kullanıcı Doğrulama :

```
Cihaz Açılıyor...
Cihaz: ACİK

Kullanıcı Adı: yavuz
Şifre: 1921
Kullanıcı Girişi Başarılı...!
```

## 9. Sıcaklığın Görüntülenmesi

```
1-Sıcaklık Görüntüle
2-Sogutucu Ac
3-Sogutucu Kapat
4-Cikis

İşlem Giriniz: 1

Sıcaklık: 44

Buse: Oda 44 Derece Olmus!!!
```

## 10. Soğutucunun açılıp/kapatılması

```
1-Sicaklik Goruntule
2-Sogutucu Ac
3-Sogutucu Kapat
4-Cikis

Islem Giriniz: 2

Sogutucu Aciliyor...
Sogutucu: ACIK
```

```
1-Sicaklik Goruntule
2-Sogutucu Ac
3-Sogutucu Kapat
4-Cikis

Islem Giriniz: 3

Sogutucu Kapatiliyor...
Sogutucu: KAPALI
```

## 11. Veri tabanı görüntüsü

	kullanici_id [PK] integer	kullanici_adi character varying (40)	sifre character varying (30)
1	2	berke	1923
2	1	yavuz	1921

Factory Method – Nesne oluşturmayla ilgili desenlerden biridir. Nesne oluşturmak istediğimizde “factory method” kullanarak bu işi istemci koddan ayırmış oluruz. Alt sınıflar

oluşturulacak bu nesne türünü kendileri belirleyebilir. Biz projemizde cihazFabrikasi tanımlayarak oluşturulmak istenen herhangi farklı bir cihazın oluşumuna zemin hazırlamış oluruz.

cihazFabrikasi’ndan Cihaz’ı oluşturarak projemize dahil ettik. Bu projede eklenebilecek yeni Cihazlar için o Cihazların sürüm modeli girilmelidir.

```
public class CihazFabrikasi {
    public static Cihaz getCihaz(String surum) {
        Cihaz cihaz;
        if("1.0.0".equalsIgnoreCase(surum)) {
            cihaz=Cihaz.getInstance(VeriTabaniPostgreSQL.getInstance(),Arayuz.getInstance(),Eyleyici.getInstance(),
                SicaklikAlgilayici.getInstance(15,35),Ekran.getInstance(),TusTakimi.getInstance());
        }else {
            System.out.println("Gecerli Bir Surum Giriniz");
            cihaz=null;
        }
        return cihaz;
    }
}
```

Observer - Çok sayıda nesneye, gözlemledikleri nesnede meydana gelen olayı bildirmek gerektiğinde kullanılabilir. Davranışsal desenlerden biridir

.Bu projede sıcaklık istenilen aralıkta değilse observer kullanılmıştır.

```
@Override
public void update(Observable observable, IEkran cihazEkran) {
    SıcaklikAlgilayici algilayici=(SıcaklikAlgilayici) observable;
    cihazEkran.mesajGoruntule(adi+":" + " Oda " + algilayici.getAnlikSıcaklik( cihazEkran ) + " Derece Olmus!!!\n\n");
}
```

Dependency Inversion – Nesnelerin arasındaki bağlantıda, yüksek seviyeli modül ile düşük seviyeli modül soyutlamaya bağlı olmalıdır. Her ikisinin birbirine doğrudan bağlı olması yerine arayüzler üzerinden bağlantı yapılmalıdır. Bu sayede modüller üzerinde yapılacak herhangi bir değişiklikten diğer modüller etkilenmeyecektir. Böylelikle yapılacak olan değişiklikler kolaylaşır. Aynı zamanda modüllerin birbirlerine olan bağımlılığını zayıflatmış oluruz.

Bu projemizde bağımlılığı azaltmak için Eyleyici sınıfının işlemlerini bir interface olan IIsleme bağlayarak varolan kodu değiştirmeden Eyleyiciye yeni işlemler eklemeyi sağladık.

```
public interface IIslem {

    void islemYap(IEkran cihazEkran);
}

private IIslem sogutucuAcma;
private IIslem sogutucuKapatma;

private Eyleyici() {
    sogutucuAcma=SogutucuAc.getInstance();
    sogutucuKapatma=SogutucuKapat.getInstance();
}

public static Eyleyici getInstance(){

public class Ekran implements IEkran {
    private static Ekran instance;

    private Ekran() {

    }

    public static Ekran getInstance() {
        if(instance==null){
            instance=new Ekran();
        }
        return instance;
    }
}
```

Kaynak Kodlar

Github: <https://github.com/yavuzhanalbayrak/AkilliCihaz/tree/main/AkilliCihaz>