



**T.C.**  
**SAKARYA ÜNİVERSİTESİ**

**BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**NESNE YÖNELİMLİ ANALİZ VE TASARIM**  
**PROJE RAPORU**

**Emirhan ŞENSOY**

**G191210040**

**2.Öğretim B Grubu**

**emirhan.sensoy@ogr.sakarya.edu.tr**

## Kullanıcı Doğrulama Ekranı

Kullanıcı doğrulama ekranı, girilen kullanıcı adı ve şifreyi alıp akıllı cihaz bilgi sisteminden PostgreSQL sunucuya erişerek veritabanında verilerin doğru girilip girilmediğini kontrol eder ve giriş işlemini gerçekleştirir.

```
Run - Analiz
Run: Main x
D:\Java\jdk-15.0.2\bin\java.exe "-javaagent:C:\Program Files\IntelliJ IDEA Community Editio
Kullanıcı Adı..:
emirhan.sensoy
Şifre..:
123
Kullanıcı Adı: emirhan.sensoy, Şifre: 123, Ad: Emirhan, Soyad: Şensoy
İşlem Kaydı -> 16-05-2021 15:48:44 -> Giriş yapıldı.
Ekran Kaydı -> 16-05-2021 15:48:44 -> Giriş yapıldı.
---Hoşgeldin Emirhan---
*****
Ana Menü
1-Sıcaklık Görüntüle
2-Soğutucuyu Aç
3-Soğutucuyu Kapat
4-Çıkış
*****
Seçiminiz..:
```

```
Run - Analiz
Run: Main x
D:\Java\jdk-15.0.2\bin\java.exe "-javaagent:C:\Program Files\IntelliJ IDEA Community Editio
Kullanıcı Adı..:
asdasd
Şifre..:
asdasd
Kullanıcı adı veya şifrenizi yanlış girdiniz.
Lütfen tekrar deneyiniz.
Kullanıcı Adı..:
|
```

## Sıcaklık Görüntüleme Ekranı

Sıcaklık görüntüleme ekranı, ağ arayüzü aracılığı ile algılayıcıdan aldığı sıcaklık değerini ekrana yazdırır.

```
Run - Analiz
Run: Main x
D:\Java\jdk-15.0.2\bin\java.exe "-javaagent:C:\Program Files\IntelliJ IDEA Community Editio
Kullanıcı Adı..:
emirhan.sensoy
Şifre..:
123
Kullanıcı Adı: emirhan.sensoy, Şifre: 123, Ad: Emirhan, Soyad: Şensoy
İşlem Kaydı -> 16-05-2021 16:06:53 -> Giriş yapıldı.
Ekran Kaydı -> 16-05-2021 16:06:53 -> Giriş yapıldı.
---Hoşgeldin Emirhan---
*****
Ana Menü
1-Sıcaklık Görüntüle
2-Soğutucuyu Aç
3-Soğutucuyu Kapat
4-Çıkış
*****
Seçiminiz..:
1
Sıcaklık Görüntüleniyor...
Sıcaklık = 36C
İşlem Kaydı -> 16-05-2021 16:06:57 -> Sıcaklık görüntülendi.
Ekran Kaydı -> 16-05-2021 16:06:57 -> Sıcaklık görüntülendi.
*****
Ana Menü
1-Sıcaklık Görüntüle
2-Soğutucuyu Aç
3-Soğutucuyu Kapat
4-Çıkış
*****
Seçiminiz..:
|
```

## Soğutucu Açma Ekranı

Soğutucu açma ekranı, ağ arayüzü aracılığı ile eyleyici üzerinden soğutucu durumunu kontrol edip soğutucu açma işlemini gerçekleştirir.

```
Run: Main x
D:\Java\jdk-15.0.2\bin\java.exe "-javaagent:C:\Program Files\IntelliJ IDEA Community Editi
Kullanıcı Adı...:
emirhan.sensoy
Şifre...:
123
Kullanıcı Adı: emirhan.sensoy, Şifre: 123, Ad: Emirhan, Soyad: Şensoy
İşlem Kaydı -> 16-05-2021 16:10:46 -> Giriş yapıldı.
Ekran Kaydı -> 16-05-2021 16:10:46 -> Giriş yapıldı.
---Hoşgeldin Emirhan---
*****
Ana Menü
1-Sıcaklık Görüntüle
2-Soğutucuyu Aç
3-Soğutucuyu Kapat
4-Çıkış
*****
Seçiminiz...:
2
Soğutucu açılıyor...
Soğutucu açıldı...
İşlem Kaydı -> 16-05-2021 16:10:50 -> Soğutucu aç çalıştırıldı.
Ekran Kaydı -> 16-05-2021 16:10:50 -> Soğutucu aç çalıştırıldı.
*****
Ana Menü
1-Sıcaklık Görüntüle
2-Soğutucuyu Aç
3-Soğutucuyu Kapat
4-Çıkış
*****
Seçiminiz...:
```

```
Run - Analiz
Run: Main x
D:\Java\jdk-15.0.2\bin\java.exe "-javaagent:C:\Program Files\IntelliJ IDEA Community Editi
Kullanıcı Adı...:
emirhan.sensoy
Şifre...:
123
Kullanıcı Adı: emirhan.sensoy, Şifre: 123, Ad: Emirhan, Soyad: Şensoy
İşlem Kaydı -> 16-05-2021 16:13:56 -> Giriş yapıldı.
Ekran Kaydı -> 16-05-2021 16:13:56 -> Giriş yapıldı.
---Hoşgeldin Emirhan---
*****
Ana Menü
1-Sıcaklık Görüntüle
2-Soğutucuyu Aç
3-Soğutucuyu Kapat
4-Çıkış
*****
Seçiminiz...:
2
Soğutucu açılıyor...
Soğutucu zaten açık...
İşlem Kaydı -> 16-05-2021 16:13:59 -> Soğutucu aç çalıştırıldı.
Ekran Kaydı -> 16-05-2021 16:13:59 -> Soğutucu aç çalıştırıldı.
*****
Ana Menü
1-Sıcaklık Görüntüle
2-Soğutucuyu Aç
3-Soğutucuyu Kapat
4-Çıkış
*****
Seçiminiz...:
|
```

## Soğutucu Kapatma Ekranı

Soğutucu kapatma ekranı, ağ arayüzü aracılığı ile eyleyici üzerinden soğutucu durumunu kontrol edip soğutucu kapatma işlemini gerçekleştirir.

```
Run - Analiz
Run: Main x
D:\Java\jdk-15.0.2\bin\java.exe "-javaagent:C:\Program Files\IntelliJ IDEA Community Editi
Kullanıcı Adı...:
emirhan.sensoy
Şifre...:
123
Kullanıcı Adı: emirhan.sensoy, Şifre: 123, Ad: Emirhan, Soyad: Şensoy
İşlem Kaydı -> 16-05-2021 16:14:41 -> Giriş yapıldı.
Ekran Kaydı -> 16-05-2021 16:14:41 -> Giriş yapıldı.
---Hoşgeldin Emirhan---
*****
Ana Menü
1-Sıcaklık Görüntüle
2-Soğutucuyu Aç
3-Soğutucuyu Kapat
4-Çıkış
*****
Seçiminiz..:
3
Soğutucu kapatılıyor...
Soğutucu zaten kapalı...
İşlem Kaydı -> 16-05-2021 16:14:45 -> Soğutucu kapat çalıştırıldı.
Ekran Kaydı -> 16-05-2021 16:14:45 -> Soğutucu kapat çalıştırıldı.
*****
Ana Menü
1-Sıcaklık Görüntüle
2-Soğutucuyu Aç
3-Soğutucuyu Kapat
4-Çıkış
*****
Seçiminiz..:
|
```

```
Run - Analiz
Run: Main x
D:\Java\jdk-15.0.2\bin\java.exe "-javaagent:C:\Program Files\IntelliJ IDEA Community Editi
Kullanıcı Adı...:
emirhan.sensoy
Şifre...:
123
Kullanıcı Adı: emirhan.sensoy, Şifre: 123, Ad: Emirhan, Soyad: Şensoy
İşlem Kaydı -> 16-05-2021 16:15:44 -> Giriş yapıldı.
Ekran Kaydı -> 16-05-2021 16:15:44 -> Giriş yapıldı.
---Hoşgeldin Emirhan---
*****
Ana Menü
1-Sıcaklık Görüntüle
2-Soğutucuyu Aç
3-Soğutucuyu Kapat
4-Çıkış
*****
Seçiminiz..:
3
Soğutucu kapatılıyor...
Soğutucu kapandı...
İşlem Kaydı -> 16-05-2021 16:15:52 -> Soğutucu kapat çalıştırıldı.
Ekran Kaydı -> 16-05-2021 16:15:52 -> Soğutucu kapat çalıştırıldı.
*****
Ana Menü
1-Sıcaklık Görüntüle
2-Soğutucuyu Aç
3-Soğutucuyu Kapat
4-Çıkış
*****
Seçiminiz..:
|
```

## Veritabanı Görüntüsü

Database:	AkıllıCihaz	Schema:	public	Table:	Kullanıcılar
	KullaniciAdi	Sifre	Ad	Soyad	
1	ahmet12	abc	Ahmet	Şahin	
2	ayseylmz	bbb	Ayşe	Yılmaz	
3	ceren_celik	111	Ceren	Çelik	
4	emirhan.sensoy	123	Emirhan	Şensoy	

## Dependency Inversion İlkesi

Dependency inversion ilkesi bağımlılığın ters çevrilmesi prensibi olarak da bilinir. Mantık olarak üst ve alt sınıfların birbirlerine bağımlı olmalarına karşı çıkan bir prensiptir. Bu bağımlılık arayüzler ile giderilir. Üst sınıfın, alt sınıfın implement ettiği arayüzü kullanması sağlanarak iki sınıf arasındaki bağımlılık azaltılır. Amaç projelerdeki zaman maliyetini düşürmek ve projeye sonradan eklenecek yapıların projeye girişini, okunabilirliğini ve yenilenebilirliğini artırmaktır.

```
IIslem.java
1 package mypackage;
2
3 interface IIslem
4 {
5     void islemYap();
6 }
7
```

```
SicaklikGoruntuleme.java
1 package mypackage;
2
3 public class SicaklikGoruntuleme implements IIslem
4 {
5     IEkran ekran;
6     IAlgilyici algilyici;
7     IAgArayuzu agArayuzu;
8
9     @private SicaklikGoruntuleme(SicaklikGoruntulemeBuilder builder)
10    {
11        this.ekran = builder.ekran;
12        this.algilyici = builder.algilyici;
13        this.agArayuzu = builder.agArayuzu;
14    }
15
16    @Override
17    public void islemYap()
18    {
19        Araclar.bekle( sure: 600);
20        ekran.mesajGoruntule(agArayuzu.sicaklikGonder(algilyici.sicaklikOku()));
21        Araclar.bekle( sure: 500);
22    }
23
24    public static class SicaklikGoruntulemeBuilder
```

```

70     private void islemSecimi()
71     {
72         int secim;
73         do
74         {
75             secim = anaMenuyuGoster();
76
77             switch (secim)
78             {
79                 case SICAKLIK_GORUNTULEME -> {
80                     IIslem sicaklikGoruntuleme = new SicaklikGoruntuleme.SicaklikGoruntulemeBuilder(ekran, algilayici, agArayuzu).build();
81                     sicaklikGoruntuleme.islemYap();
82                     publisher.notify( m: "Sıcaklık görüntülendi.");
83                 }
84                 case SOGUTUCU_AC -> {
85                     IIslem sogutucuAc = new SogutucuAc.SogutucuAcBuilder(ekran, eyleyici, agArayuzu).build();
86                     sogutucuAc.islemYap();
87                     publisher.notify( m: "Soğutucu aç çalıştırıldı.");
88                 }
89                 case SOGUTUCU_KAPAT -> {
90                     IIslem sogutucuKapat = new SogutucuKapat.SogutucuKapatBuilder(ekran, eyleyici, agArayuzu).build();
91                     sogutucuKapat.islemYap();
92                     publisher.notify( m: "Soğutucu kapat çalıştırıldı.");
93                 }
94                 case CIKIS -> {
95                     ekran.mesajGoruntule("Çıkış yapılıyor...");
96                     publisher.notify( m: "Çıkış yapıldı.");
97                 }
98                 default -> {
99                     Araclar.bekle( sure: 200);
100                    ekran.mesajGoruntule("1-4 arasında bir sayı girmelisiniz.");
101                    Araclar.bekle( sure: 600);
102                }
103            }
104        }
105        while(secim != 4);

```

## Builder Deseni

Builder tasarım deseni, uzun ve kompleks kodu kısaltıp kolaylaştırmaya yarayan bir tasarım desendir. Geliştirilen yazılımda farklı parametre kombinasyonlarına sahip birden çok kurucu fonksiyon veya çok fazla parametre alan bir obje varsa kullanılır.

```

1     package mypackage;
2
3     public class KullaniciHesabi
4     {
5         private String kullaniciAdi;
6         private InternetKullanicisi internetKullanicisi;
7
8         @
9         private KullaniciHesabi(KullaniciHesabiBuilder builder)
10        {
11            this.kullaniciAdi = builder.kullaniciAdi;
12            this.internetKullanicisi = builder.internetKullanicisi;
13        }
14
15        public String getAd() { return internetKullanicisi.getAd(); }
16        public String getSoyad() { return internetKullanicisi.getSoyad(); }
17
18        public static class KullaniciHesabiBuilder
19        {
20            private String kullaniciAdi;
21            private InternetKullanicisi internetKullanicisi;
22
23            public KullaniciHesabiBuilder(String kullaniciAdi, String ad, String soyad)
24            {
25                this.kullaniciAdi = kullaniciAdi;
26                this.internetKullanicisi = new InternetKullanicisi.InternetKullanicisiBuilder(ad, soyad).build();
27            }
28
29            public KullaniciHesabi build() {
30                return new KullaniciHesabi( builder, this);
31            }
32        }
33    }
34
35
36
37
38
39
40
41

```

```
kullaniciHesabi = new KullaniciHesabi.KullaniciHesabiBuilder(kullaniciAdi, ad, soyad).build();
```

## Observer Deseni

Observer adı üzerinde gözlemci, izleyici, gözcü yahut gözetmen diye nitelendirilen, anlamı gibi işlev gören bir tasarım desendir. Mevcut bir nesnenin durumunda herhangi bir değişiklik olduğunda, bu değişikliklerden diğer nesneleri haberdar etmeye yarar. Observer tasarım deseninde; takip edilecek nesneye subject, subject'i takip eden nesnelere concrete ve concrete nesnelerin birden fazla tip olmasını sağlayan arayüze de observer denir.

```
1 package mypackage;
2
3 import java.util.ArrayList;
4
5 public class Publisher implements ISubject
6 {
7     private ArrayList<IObserver> subscribers = new ArrayList<>();
8
9     @Override
10    public void attach(IObserver subscriber)
11    {
12        subscribers.add(subscriber);
13    }
14
15    @Override
16    public void detach(IObserver subscriber)
17    {
18        subscribers.remove(subscriber);
19    }
20
21    @Override
22    public void notify(String mesaj)
23    {
24        for(IObserver subscriber: subscribers)
25        {
26            subscriber.update(mesaj);
27        }
28    }
29 }
30
31
```

```
1 package mypackage;
2
3 public interface ISubject
4 {
5     void attach(IObserver o);
6     void detach(IObserver o);
7     void notify(String m);
8 }
9
```

```
1 package mypackage;
2
3 public interface IObserver
4 {
5     void update(String m);
6 }
7
```

```
1 package mypackage;
2
3 import java.text.SimpleDateFormat;
4 import java.util.Date;
5
6 public class EkranKaydi implements IObserver
7 {
8     Date date;
9     SimpleDateFormat formatter;
10
11    public EkranKaydi()
12    {
13        formatter = new SimpleDateFormat("dd-MM-yyyy HH:mm:ss");
14    }
15
16    @Override
17    public void update(String mesaj)
18    {
19        date = new Date();
20        System.out.println("Ekran Kaydı -> " + formatter.format(date) + " -> " + mesaj);
21    }
22 }
23
```



```

1 package mypackage;
2
3 import java.text.SimpleDateFormat;
4 import java.util.Date;
5
6 public class IslemKaydi implements IObserver
7 {
8     Date date;
9     SimpleDateFormat formatter;
10
11     public IslemKaydi()
12     {
13         formatter = new SimpleDateFormat( "dd-MM-yyyy HH:mm:ss");
14     }
15
16     @Override
17     public void update(String mesaj)
18     {
19         date = new Date();
20         System.out.println("İşlem Kaydı -> " + formatter.format(date) + " -> " + mesaj);
21     }
22 }
23

```

```

1 package mypackage;
2
3 public class Uygulama
4 {
5     public static void main(String[] args)
6     {
7         Publisher publisher = new Publisher();
8
9         IslemKaydi islemKaydi = new IslemKaydi();
10        EkranKaydi ekranKaydi = new EkranKaydi();
11
12        publisher.attach(islemKaydi);
13        publisher.attach(ekranKaydi);
14
15        AkilliCihaz cihaz = new AkilliCihaz(publisher);
16
17        cihaz.calistir();
18    }
19 }
20

```

```

Run - Analiz
Run: Main x
D:\Java\jdk-15.0.2\bin\java.exe "-javaagent:C:\Program Files\IntelliJ IDEA Community Editi
Kullanıcı Adı..:
emirhan.sensoy
Şifre..:
123
Kullanıcı Adı: emirhan.sensoy, Şifre: 123, Ad: Emirhan, Soyad: Şensoy
İşlem Kaydı -> 16-05-2021 16:06:53 -> Giriş yapıldı.
Ekran Kaydı -> 16-05-2021 16:06:53 -> Giriş yapıldı.
---Hoşgeldin Emirhan---
*****
Ana Menü
1-Sıcaklık Görüntüle
2-Soğutucuyu Aç
3-Soğutucuyu Kapat
4-Çıkış
*****
Seçiminiz..:
1
Sıcaklık Görüntüleniyor...
Sıcaklık = 36C
İşlem Kaydı -> 16-05-2021 16:06:57 -> Sıcaklık görüntülendi.
Ekran Kaydı -> 16-05-2021 16:06:57 -> Sıcaklık görüntülendi.
*****
Ana Menü
1-Sıcaklık Görüntüle
2-Soğutucuyu Aç
3-Soğutucuyu Kapat
4-Çıkış
*****
Seçiminiz..:
|

```

## **Proje Github Linki**

<https://github.com/emirhansensoy/ObjectOrientedAnalysisAndDesign-SmartDeviceProject>

## **Projenin Anlatıldığı Youtube Video**

<https://youtu.be/8ILPbymZOEg>