



SAKARYA
ÜNİVERSİTESİ

BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ
NESNE YÖNELİMLİ ANALİZ VE TASARIM DERSİ
PROJE RAPORU

AD: Ayşegül

SOYAD: GENÇEL

NUMARA: B201210053

ŞUBE: 1.Öğretim B Grubu

MAİL: aysegul.gencel@ogr.sakarya.edu.tr

AD: Şeyma

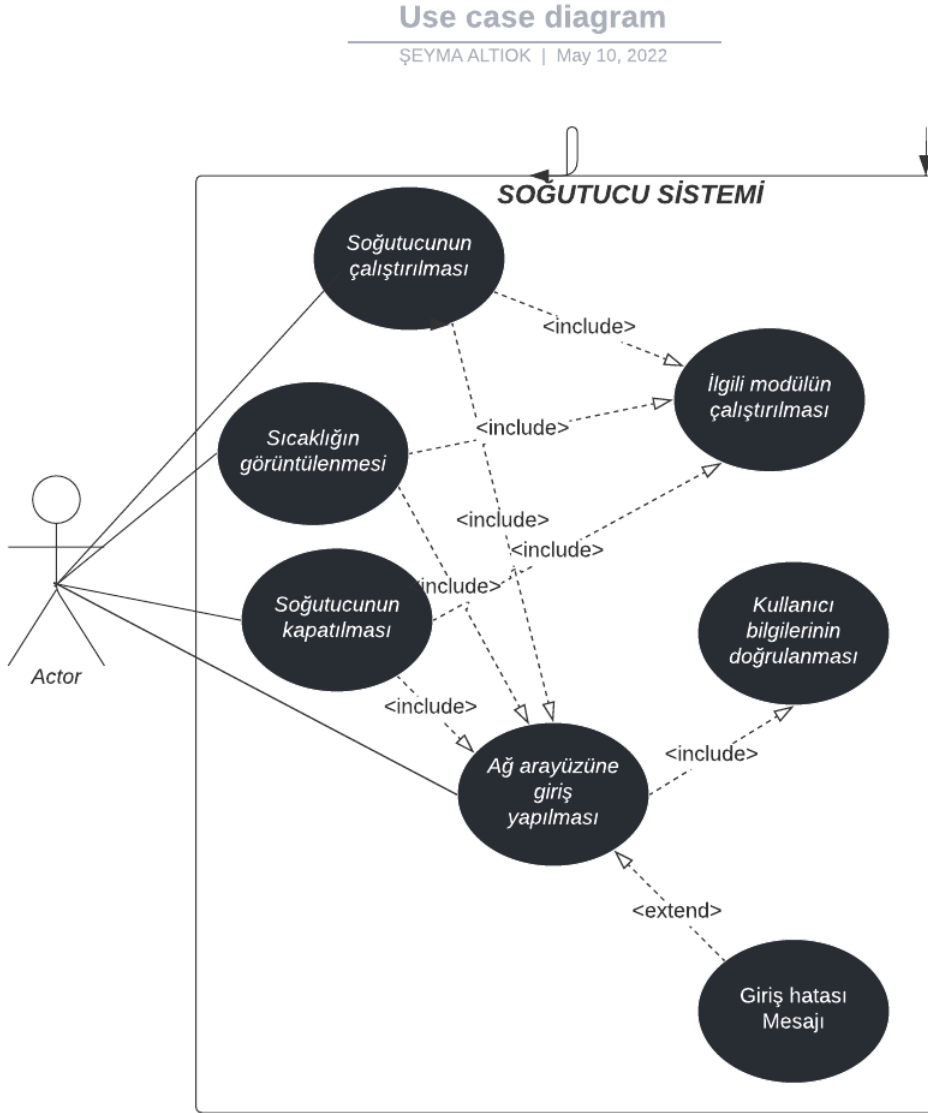
SOYAD: ALTIOK

NUMARA: B201210026

ŞUBE: 1.Öğretim A Grubu

MAİL: seyma.altiook@ogr.sakarya.edu.tr

A) USE CASE DİYAGRAMI



B) KULLANIM DURUMLARI METİNSEL TANIMLAR

a) Sıcaklık Görüntüleme

Kullanım Durumu Adı: Sıcaklık Görüntüle

Hazırlayanlar: Şeyma ALTIOK, Ayşegül GENÇEL

Sürüm: 1.0.0

Tarih: 11.04.2022

İlgili Aktörler: Kullanıcı

Giriş Koşulu: Kullanıcı ağ ara yüzüne giriş yapar, ağ ara yüzü üzerinden sıcaklık görüntülemek ister. Sıcaklık algılayıcıdan sıcaklık bilgisi alınır.

Çıkış Koşulu: Ağ ara yüzü kapatılır ve sıcaklık görüntüleme işlemi tamamlanır.

Ana Olay Koşulu:

1. Kullanıcı ağ ara yüzüne bağlanır.
2. Ağ ara yüzü ekranına “kullanıcı adı ve şifrenizi giriniz” mesajı gösterilir.
3. Kullanıcı, ağ ara yüzüne kullanıcı adı ve şifresini girer.

4. Kullanıcı doğrulanır.
5. Ağ ara yüzünden sıcaklık görüntüle fonksiyonu seçilir.
6. Sıcaklık algılayıcı çalışır ve ortamın sıcaklık derecesini algılar.
7. Sıcaklık algılayıcısından alınan sıcaklık değeri, merkezi işlem biriminden geçerek ağ ara yüzüne aktarılır.
8. Ağ ara yüzünde sıcaklık değeri ekrana yazdırılır.
9. Kullanıcı ağ ara yüzünde sıcaklığı görüntüler.
10. İşlem tamamlanır.

Alternatif Olay Akışı:

A1.Ağ ara yüzüne bağlanılmadı.

- 1.Kullanıcı ekranında internet bağlantısını kontrol etmesi için mesaj gösterilir.

A4.Kullanıcı Doğrulanamadı.

- 1.Kullanıcı adı ve şifre doğru girilene kadar giriş yapılması istenir.

A6.Sıcaklık ölçülemedi.

1. Ağ ara yüzünden tekrar deneyiniz mesajı verilir.
2. Tekrar ölçülemezse internet bağlantısının kontrol edilmesi istenir.
3. İnternet bağlantısında sıkıntı yoksa sensor problemi mesajı verilir ve işlem sonlanır.

Özel Gereksinimler:

1. UI gereksinimi
2. İşlemlerin gerçek zamanlı gerçekleştirilmesi
3. Sürekli ve hızlı internet bağlantısı
4. Gerekli ve dayanıklı sensorler

b) Soğutucu Çalıştırma

Kullanım Durumu adı: Soğutucu Açma

Hazırlayanlar: Şeyma ALTIOK, Ayşegül GENÇEL

Sürüm: 1.0.0

Tarih: 11.04.2022

İlgili Aktörler: Kullanıcı, Eyleyici

Giriş Koşulu: Kullanıcı ağ ara yüzüne giriş yapar, ağ ara yüzü üzerinden soğutucu açmak ister. Bu istek, merkezi işlem biriminden geçtikten sonra eyleyici ile soğutucu aç fonksiyonu kullanılarak gerçekleşir.

Çıkış Koşulu: Soğutucu açılır ve işlem tamamlanır.

Ana Olay Koşulu:

1. Kullanıcı ağ ara yüzüne bağlanır.
2. Ağ ara yüzü ekranına “kullanıcı adı ve şifrenizi giriniz” mesajı gösterilir.
3. Kullanıcı, ağ ara yüzüne kullanıcı adı ve şifresini girer.
4. Kullanıcı doğrulanır.
5. Ağ ara yüzünden kullanıcı sıcaklık durumuna göre soğutucuyu açma fonksiyonunu çalıştırılır.
6. Ağ ara yüzündeki soğutucu aç fonksiyonu merkezi işlem biriminden geçerek eyleyiciye soğutucuyu açması için istekte bulunur.
7. Eyleyici gelen isteği alır ve soğutucu açılır.
8. Ağ ara yüzüne bağlı olan tüm kullanıcılara soğutucu açıldı mesajı gösterilir.
9. İşlem tamamlanır.

Alternatif Olay Akışı:

A1.Ağ ara yüzüne bağlanılmadı.

- 1.Kullanıcı ekranında internet bağlantısını kontrol etmesi için mesaj gösterilir.

A4.Kullanıcı Doğrulanamadı.

- 1.Kullanıcı adı ve şifre doğru girilene kadar giriş yapılması istenir.

A6.Soğutucu ile merkezi işlem birimi haberleşemiyor.

- 1.Soğutucunun ve merkezi işlem birimi kontrol edilir.
- 2.Tekrar denenmesi istenir.
3. 3 denemeden sonra sistem mimarına başvurulması istenir.

A7.Soğutucu açılmadı.

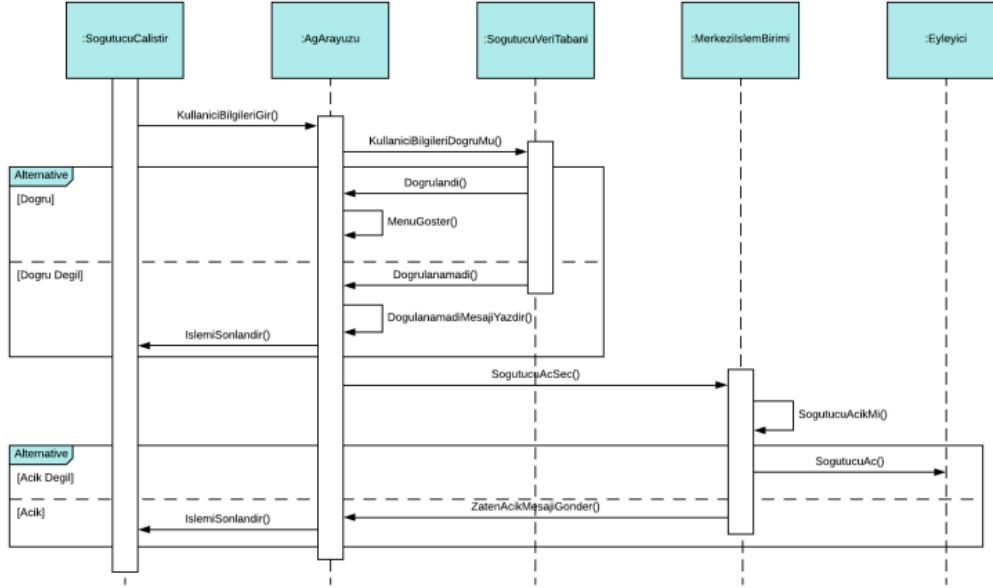
- 1.Ağ ara yüzünde tekrar deneyiniz mesajı verilir.
2. Tekrar açılmazsa internet bağlantısının kontrol edilmesi istenir.
3. İnternet bağlantısında sıkıntı yoksa soğutucunun kontrol edilmesi istenir.

Özel Gereksinimler:

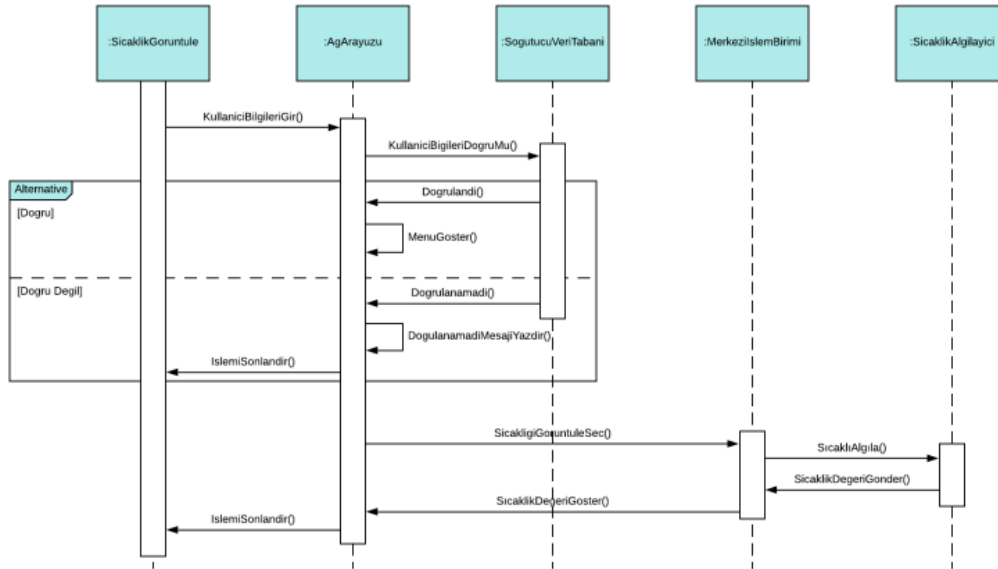
1. UI gereksinimi
2. İşlemlerin gerçek zamanlı gerçekleştirilmesi
3. Sürekli ve hızlı internet bağlantısı

C) SIRALAMA ŞEMALARI (SEQUENCE DIAGRAM)

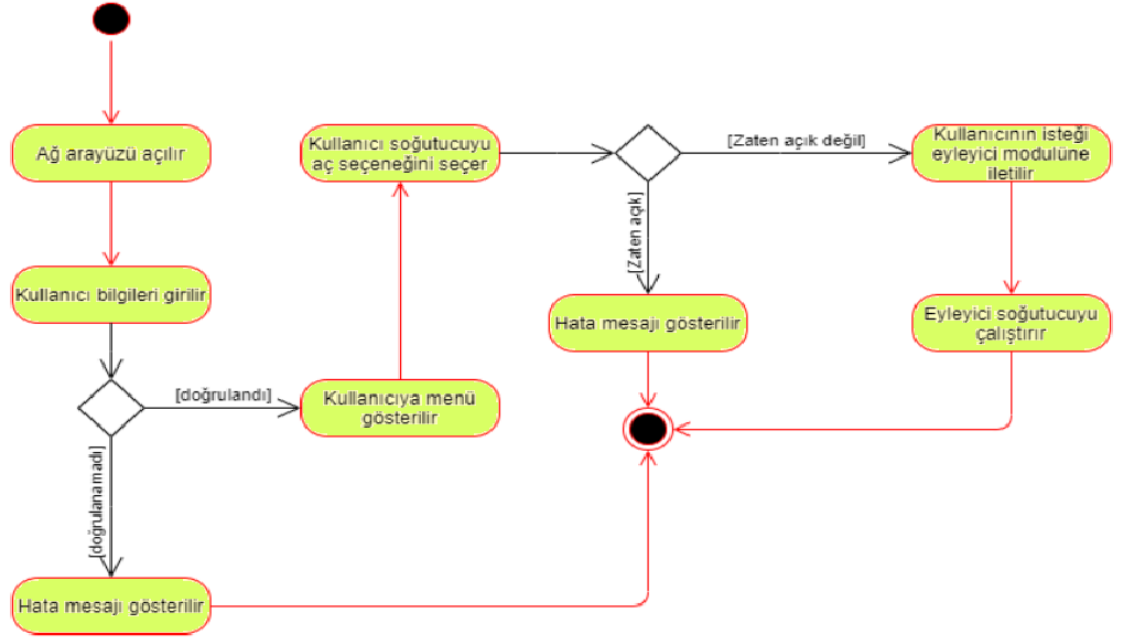
SOĞUTUCU ÇALIŞTIRILMASI SIRALAMA ŞEMASI



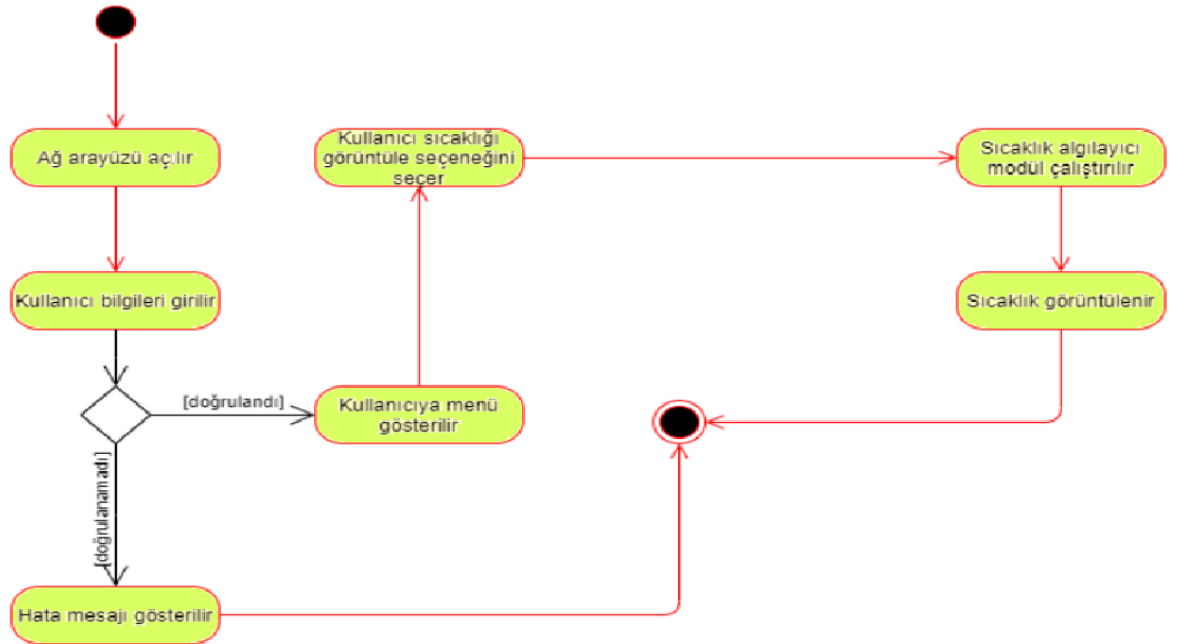
SICAKLIĞIN GÖRÜNTÜLENMESİ SIRALAMA ŞEMASI



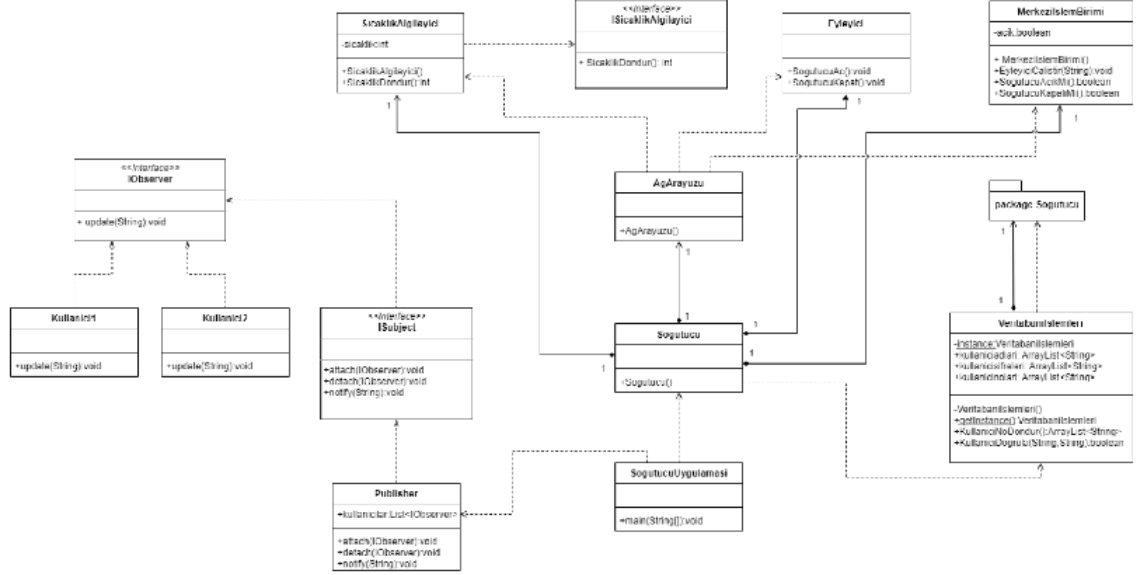
D) ETKİNLİK ŞEMALARI (ACTİVİTY DİAGRAM)
SOĞUTUCUNUN ÇALIŞTIRILMASI - ETKİNLİK ŞEMASI:



SICAKLIĞIN GÖRÜNTÜLENMESİ - ETKİNLİK ŞEMASI



E) SINIF ŞEMASI



F) CRC KARTLARI

EYLEYİCİ	
sorumluluklar	işbirlikleri
Soğutucu açma	IEyleyici
Soğutucu kapatma	

EYLEYİCİ	
sorumluluklar	işbirlikleri
Soğutucu açma	IEyleyici
Soğutucu kapatma	

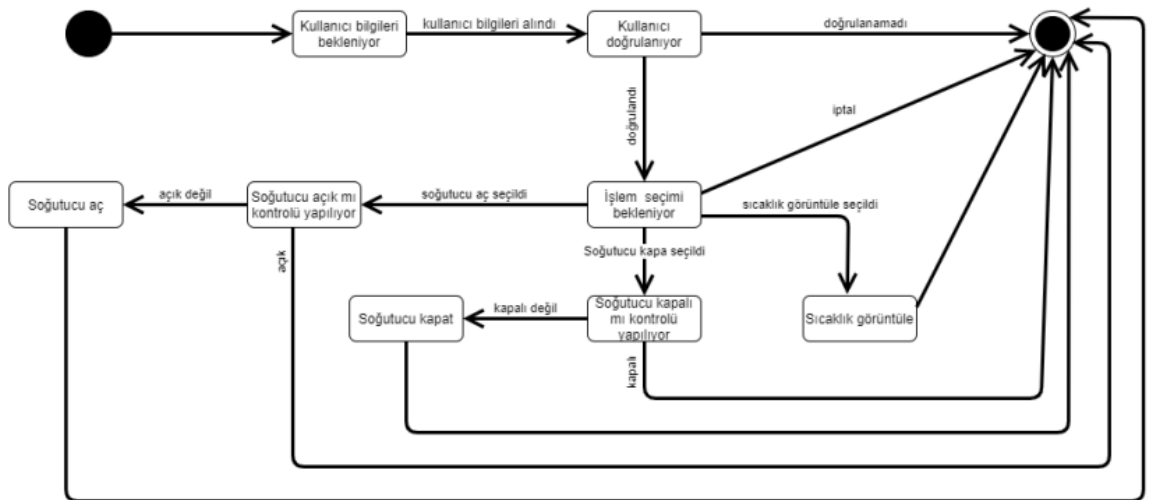
Kritik soğutma	
sorumluluklar	işbirlikleri
kritik durum	IObserver

LogDosya	
sorumluluklar	işbirlikleri
dosya	ILogger

LogEkran	
sorumluluklar	işbirlikleri
ekran	ILogger

MerkeziSistemBirimi	
sorumluluklar	işbirlikleri
kritik sıcaklık kontrol	IMerkeziSistemBirimi
sıcaklık okuma	ISıcaklıkAlgılayıcı
Soğutucu açma	IEyleyici
Soğutucu kapatma	ISubject

G) DURUM MAKİNA DİYAGRAMI



The screenshot displays the IntelliJ IDEA interface with the following components:

- Package Explorer (Left):** Shows the project structure under 'NesnelFonemliAnalizTasarim-master'. It includes 'JRE System Library [jdk-17.0.2]', 'Proje/src' containing 'soğutucuuygulamasi' (with files like AgArrayuzu.java, Eyleycici.java, IObservor.java, SicaklikAlgilyicisi.java, Subject.java, Kullanicil1.java, Kullanicil2.java, MerkezilemBirimi.java, Publisher.java, SicaklikAlgilyicisi.java, Soğutucu.java, Soğutucuygulamasi.java, Veritabanisistemleri.java) and 'Referenced Libraries'.
- Main Editor:** Displays the code for 'Veritabanisistemleri.java':

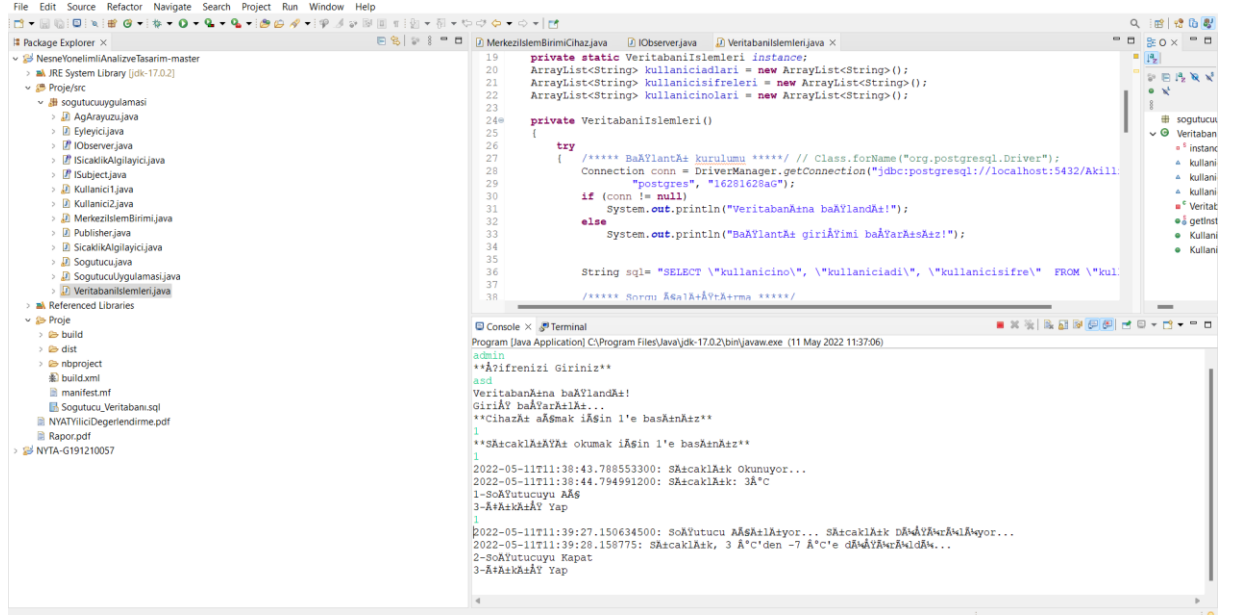

```

19 private static Veritabanisistemleri instance;
20 ArrayList<String> kullaniciladlari = new ArrayList<String>();
21 ArrayList<String> kullancisisifreleri = new ArrayList<String>();
22 ArrayList<String> kullanicinolari = new ArrayList<String>();
23
24 private Veritabanisistemleri()
25 {
26     try
27     {
28         /***/ BaKıYantAş kurulumu ****// Class.forName("org.postgresql.Driver");
29         Connection conn = DriverManager.getConnection("jdbc:postgresql://localhost:5432/Akıl1
30             "postgres", "16281628ag");
31         if (conn != null)
32             System.out.println("Veritabanısına baKıYlandık!");
33         else
34             System.out.println("BaKıYlantAş giriÅİyimi baKıYarAntaşAşz!");
35
36         String sql= "SELECT \'\\kullanico\'\\', \'\\kullanciadı\\', \'\\kullancisisfre\' FROM \'\\kul
37
38         /***** Sorm İŞenİAŞıYrAŞma *****/
      
```
- Console/Terminal (Bottom):** Shows the execution output:

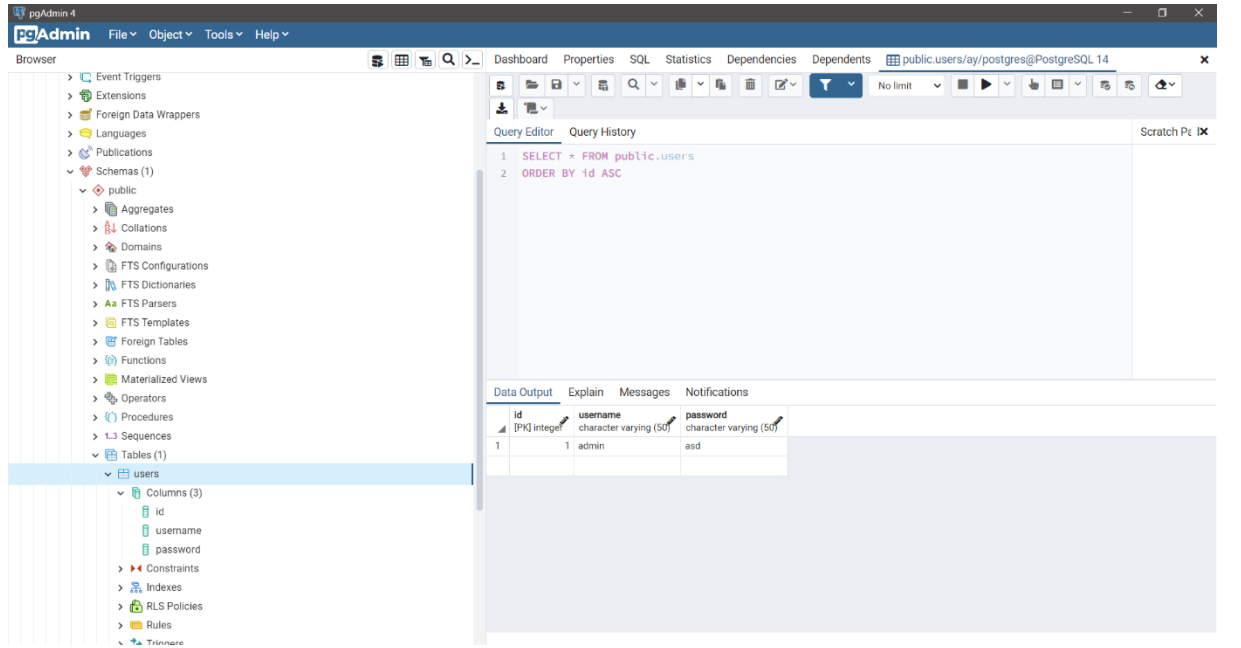

```

Program [java Application] C:\Program Files\Java\jdk-17.0.2\bin\javaw.exe (11 May 2022 11:37:06)
**Kullanıcılara AdAtınAşızAş Giriniz**
admin
**Åİfrenizi Giriniz**
asd
Veritabanısına baKıYlandık!
GiriÅİY baKıYarAntaş!
**Çihazda aşmak İÅİn l'e basAntaşz**
1
**SıcaklıkAŞAYArAŞ okumak İÅİn l'e basAntaşz**
1
2022-05-11T11:38:43.788553300: Sıcaklık Okunuyor...
2022-05-11T11:38:44.794991200: Sıcaklık: 3ÅC
1-soÅıtutucuYu AÅş
3-AŞatıkAŞıY Yap
      
```

-Soğukluk Görüntüleme



J) VERİTABANI EKRAN GÖRÜNTÜSÜ (KULLANICI VERİLERİNİN SAKLANDIĞI TABLONUN, VERİLERİNİ İÇEREN EKRAN GÖRÜNTÜSÜ)



K) DEPENDENCY INVERSION İLKESİ NEDİR UYGULAMADA NASIL GERÇEKLEŞTİRİLDİ

Dependency Inversion prensibe göre;

Bağları zayıflatmak için

. Yüksek seviyeli modüller, düşük seviyeli modüllere bağlı olmamalı; ikisinin de soyutlanması,

. Soyutlamanın detaylara değil; detayların soyutlamaya bağlı olması beklenmektedir.

Maddeleri basitleştirecek olursak; sınıflar arası bağımlılıkların minimal seviyeye indirilmesi ve bağımlılıkların sınıflar ile değil ara yüzler (interface) ile kurulması gerektiğine dayanır. Sistemimizi bu şekilde tasarlamazsak yüksek seviyeli bileşenler, düşük seviyeli bileşenlere bağımlı kalacak ve düşük seviyeli bir bileşen içerisinde yapılacak olan değişikliğin zincirleme olarak bağımlı olan tüm yüksek seviye bileşenleri de değişikliğe zorlayacaktır. Dependency Inversion tam olarak da bu bağımlılığın tersine çevrilmesini amaçlamaktadır.

L) FACTORY METHOD VE OBSERVER DESENLERİ VE UYGULAMADA NASIL GERÇEKLEŞTİRİLDİ

FACTORY METHOD, oluşturulacak nesnenin tam sınıfını belirtmek zorunda kalmadan **nesne oluşturma** sorunuyla başa çıkmak için fabrika yöntemlerini kullanan bir oluşturma desendir.

-LogDosya

-LogEkran

Observer: Çok sayıda nesneye, gözlemlenen nesnede gerçekleşen bir olayı **bildirmek için kullanılan desendir**. Projemde bu deseni IObserver, Publisher sınıfı yardımı ile gerçekleştirdim. IObserver interface'i kullanıcılara bağladım böylelikle gözlemlenen nesnede bir olay geldiğinde bağlı olan kullanıcılara Publisher sınıfından nesne oluşturarak ve bu kullanıcı nesnelerini publisher'a ekledim. Publisher'ın notify metodunu çağırarak kullanıcılara mesaj iletilmesini sağladım. Ödevde kritik sıcaklığı bildiriyor.

-IObserver.java

-Publisher.java