

# Mediator Design Pattern

## ALPER ÇELİK 160202023

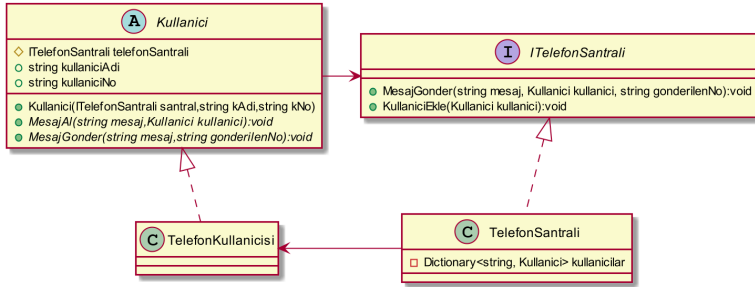
### 1. Mediator Design Pattern ALPER ÇELİK 160202023

### 2. Neden kullanılır?

Mediator Design Pattern nesnelerin aralarındaki iletişimin tek bir noktadan sağlanması ve koordine edilmesi gerektiği durumlarda kullanılır.

Nesneler birbirleri ile doğrudan konuşmak yerine merkezi bir yapı aracılığı ile haberleşirler bu sayede nesneler arasında bağımlılık azalır. Nesneler birbirlerinin kim olduklarını bilmeden merkez aracılığı ile haberleşebilirler.

### 3. UML Diagramı



### 4. Kod Blokları

```
class TelefonSantrali : ITelefonSantrali
{
    private Dictionary<string, Kullanici> kullanicilar; ❶

    public TelefonSantrali()
    {
        kullanicilar =new Dictionary<string, Kullanici>();
    }

    public void kullaniciekle(Kullanici kullanci) ❷
```

```
{
    if (!kullanici.ContainsValue(kullanici))
    {
        Console.WriteLine("Eklenmek istenen {0} kullanicisi
        listede yok. Listeye eklendi.",kullanici.kullaniciAdi);
        kullanici[kullanici.kullaniciNo] = kullanici;
    }
    else
    {
        Console.WriteLine("{0} kullanicisi zaten
        kayıtlı.",kullanici.kullaniciAdi);
    }
}

public void MesajGonder(string mesaj, kullanici kullanici,string
gonderilenNo)
{
    kullanici gonderilecekKullanici =
    kullanici[kullanici.gonderilenNo];❸
    gonderilecekKullanici.MesajAl(mesaj, kullanici);
}
}
```

- ❶ Telefon Santralinin kullanıcıları tutması için gerekli Dictionary tipinde nesne
- ❷ Dictionary'e eleman ekleme
- ❸ Santrale arabuluculuk için gelen mesajın alınıp, daha sonra iletilmesi

```
class TelefonKullanicisi : kullanici
{
    public TelefonKullanicisi(ITelefonSantrali santral, string kAdi,
    string kNo):base(santral,kAdi,kNo)
    {

    }

    public override void MesajAl(string mesaj, kullanici
    kullanici)❶
    {
        Console.WriteLine("\n( {0} kişinin telefonu )",
        this.kullaniciAdi);
        Console.WriteLine("{0} numara ile eşleşen {1} adlı
        kişiden mesaj aldınız. \nMesaj içeriği: {2}", kullanici.kullaniciNo,
        kullanici.kullaniciAdi, mesaj);
    }
}
```

```
        public override void MesajGonder(string mesaj, string
gonderilenNo)②
        {
            telefonsantrali.MesajGonder(mesaj, this, gonderilenNo);
        }
    }
```

- ① Santralden gelen mesajların ilgili kullanıcıya iletilmesi
- ② Kullanıcının, santral yardımıyla mesajını göndermesi

```
abstract class Kullanici①
{
    protected ITelefonSantrali telefonSantrali { get; set; }
    public string kullaniciAdi { get; set; }
    public string kullaniciNo { get; set; }

    public Kullanici(ITelefonSantrali santral, string kAdi, string
kNo)
    {
        this.telefonSantrali = santral;
        this.kullaniciAdi = kAdi;
        this.kullaniciNo = kNo;
    }
    public abstract void MesajAl(string mesaj, Kullanici kullanici);
    public abstract void MesajGonder(string mesaj, string
gonderilenNo);
}
```

- ① Kullanici class'ına ait gerekli fonksiyon ve değişkenlerin tanımlanması.

```
class Program
{
    static void Main(string[] args)
    {
        TelefonSantrali telefonSantrali = new TelefonSantrali();
        Kullanici A = new TelefonKullanicisi(telefonSantrali, "A",
"0");
        Kullanici B = new TelefonKullanicisi(telefonSantrali, "B",
"1");
        Kullanici C = new TelefonKullanicisi(telefonSantrali, "C",
"2");

        telefonSantrali.kullaniciEkle(A); ①
        telefonSantrali.kullaniciEkle(B);
    }
}
```

```
        telefonsantrali.kullaniciEkle(C);  
        telefonsantrali.kullaniciEkle(A);  
        Console.WriteLine("\n");  
        A.MesajGonder("Merhaba C, nasilsin?", "2"); ❷  
        C.MesajGonder("İyiym A, sen nasilsin?", "0");  
        Console.ReadLine();  
    }  
}
```

- ❶ TelefonSantraline kullanıcıların eklenmesi.
- ❷ TelefonSantrali aracılığı ile mesajlaşmanın sağlanması.

```
interface ITelefonSantrali //Mediator Interface ❶  
{  
    void MesajGonder(string mesaj, Kullanici kullanici, string  
gonderilenNo);  
    void KullaniciEkle(Kullanici kullanici);  
}
```

- ❶ Mediator interface'in tanımlanması ve gerekli fonksiyonların yazılması

## 5. Ekran Çıktısı

```
Eklenmek istenen A kullanicisi listede yok. Listeye eklendi.  
Eklenmek istenen B kullanicisi listede yok. Listeye eklendi.  
Eklenmek istenen C kullanicisi listede yok. Listeye eklendi.  
A kullanicisi zaten kayıtlı.  
  
( C kişinin telefonu )  
0 numara ile eşleşen A adlı kişiden mesaj aldınız.  
Mesaj içeriği: Merhaba C, nasilsin?  
  
( A kişinin telefonu )  
2 numara ile eşleşen C adlı kişiden mesaj aldınız.  
Mesaj içeriği: İyiym A, sen nasilsin?
```

Kullanıcılar arabulucu sınıfta tanımlanmış olan dictionary tipindeki yapıya eklenmiştir. Ardından ekran çıktısında görüldüğü üzere, telefon santrali arabulucuğu ile mesajlar bu arabulucu sınıf üzerinden iletilmiştir.