

Teknik Envanter 3 – Ertelenen İşler

- Aşağıda bir gerçek hayat problemi, basitleştirilerek çalışma için uyarlanmıştır.
- Projede iletilen bir talep için desteğiniz gerekmektedir

1- Sorumlu olduğumuz servisler, banka müşterileri ile alakalı temel işlemleri gerçekleştirmektedir. Bu servisler geliştirildikten sonra, projenin derlenmiş .dll dosyaları ilgili ekiple paylaşılır ve onlar tarafından projelerinde aşağıdaki şekildeki gibi kullanılır:

```
static void Main(string[] args)
{
    int müşteriNumarası = 15000000;

    ÇalıştırmaMotoru.KomutÇalıştır("MuhasebeModulu", "MaaşYatır", new object[] { müşteriNumarası });

    ÇalıştırmaMotoru.KomutÇalıştır("MuhasebeModulu", "YıllıkÜcretTahsilEt", new object[] { müşteriNumarası });
}
```

2- İletilen talep sonucu sistemimizde yaptığımız performans iyileştirmeleri kapsamında, bazı komutların çalıştırılmasının sistemin yoğun olmadığı bir zamana ertelenebilmesi gerekmektedir. Bu gereksinimlerden ötürü, var olan Muhasebe Modülü projesinde *BekleyenİşlemleriGerçekleştir* isimli yeni bir method oluşturulması ve *KomutÇalıştır* methodunun uygun bir şekilde düzenlenmesi gerekmektedir.

```
public static void BekleyenİşlemleriGerçekleştir()
{
    // ?
}
```

Sizden ekte senaryoya uygun oluşturulan Proje üzerinden devam ederek uygun bir çözüm geliştirmenizi talep ediyoruz



Teknik Envanter 3 – Örnek Program.cs

- <https://github.com/bt-servis-otomasyonu/default-repo/blob/master/Program.cs>

```
namespace DegerlendirmeSorulari
{
    using System;
    using System.Reflection;

    class Program
    {
        static void Main(string[] args)
        {
            int müşteriNumarası = 15000000;

            ÇalıştırmaMotoru.KomutÇalıştır("MuhasebeModulu", "MaaşYatır", new object[] { müşteriNumarası });

            ÇalıştırmaMotoru.KomutÇalıştır("MuhasebeModulu", "YıllıkÜcretTahsilEt", new object[] { müşteriNumarası });

            ÇalıştırmaMotoru.BekleyenİşlemleriGerçekleştir();
        }

        public class ÇalıştırmaMotoru
        {
            public static object[] KomutÇalıştır(string modülSınıfAdı, string methodAdı, object[] inputs)
            {
                // ?

                throw new NotImplementedException();
            }

            public static void BekleyenİşlemleriGerçekleştir()
            {
                // ?
            }
        }

        Console.WriteLine("{0} numaralı müşterinin otomatik ödemeleri gerçekleştirildi.", müşteriNumarası);
    }

    public class Veritabanıİşlemleri
    {
        // ?
    }
}
```

Teknik Envanter 3 – Faydalı Bilgiler

- Gerekli gördüğünüz herhangi bir işlemi veritabanı kullanarak gerçekleştirebilirsiniz
- Veritabanı erişimi gereksinimlerinizi *Veritabanıİşlemleri* sınıfı içinde oluşturduğunuz method'larla yalnızca açıklama yazarak yapabilirsiniz. (Basitlik açısından tablo ve sorgu tanımlarını oluşturma işlemlerini atlayınız.)
- Tüm komutların değil yalnızca belirli / tanımlanan komutların bu şekilde ertelenmesi gerektiğini göz önünde bulundurunuz.
- Modüldeki *MaaşYatır*, *YıllıkÜcretTahsilEt* gibi method'ların içi boş olmasına rağmen ilgili işlemlerin yapıldığını varsayınız. Bu method'lar için geliştirme yapmanız beklenmiyor.
- Bonus: Tüm method'lar yalnızca müşteri numarası parametresi almaktadır. Gereksinimleri gerçekleştirdikten sonra, servis method'larındaki parametre tipleri ve sayılarının öngörülemediği durumlar için de bir çözüm gerçekleştirebilirsiniz.

