**İnönü Üniversitesi**

**Bilgisayar Mühendisliği Bölümü**

**2019 – 2020 Nesneye Yönelik Yazılım Mühendisliği Dersi**

**Proje Analiz Rapor**

|  |  |
| --- | --- |
| **Ders Sorumlusu** | **Dr. Öğr. Üyesi Ahmet Arif AYDIN** |
| **Öğrenci(ler)** | **02160001016 / Ayşe ÖZAVCİ** |
| **Proje İsmi** | **Otobüs Bileti Rezervasyon Sistemi** |
| **Aşama Tanımı** | **Testing 1 + İmplementation-4**  **Uygulaması ve kodlanması gerçekleştirilen 1. , 2. ve 3. ihtiyaçların çözümünü sağlayan kod kısmının test edilmesi için gerekli olan test kodunun yazılması.**  **Seçilen creational (oluşturucu) tasarım kalıbının kodunun yazılması** |
| **AÇIKLAMA** | |
| ***Uygulaması ve kodlanması gerçekleştirilen ilk üç aşama ;***   * Java ile veri tabanı bağlantısının yapılması ve müşteri ve satıcı için login işlemi , * Satıcı’ nın bilet eklemesi , müşterinin bilet alıp kendi sistemine eklemesi ve bu işlemlerin arayüzde gösterilmesi , * Satıcı’ nın kendi sisteminden bilet silmesi, müşterinin daha önceden aldığı bileti iptal etmesi ve bu işlemlerin arayüzde gösterilmesi ,   ***İlk üç aşama için test kodu ve çalışması ;***    *AŞAMA 1*   * *Veri tabanı bağlantı testi için kod ;*     *Derleme sonucu ;*     * *Satıcı için login işlemi için test kodu ;*   *Satıcı telefonunu ve şifresini yazdıktan sonra doğru olup olmadığı kontrol edilir eğer doğruysa sisteme girdiğinde adını, soyadını ve adresini konsola yazdırdım;*     * Müşteri için login test kodu ;     Müşteri şifresini ve telefon numarasını yazdıktan sonra sisteme giriş yapar.Müşteri telefonunu ve şifresini yazdıktan sonra doğru olup olmadığı kontrol edilir eğer doğruysa sisteme girdiğinde adını, soyadını ve adresini konsola yazdırdım.  Konsol sonucu ;    *AŞAMA 2*   * Satıcı’ nın bilet eklemesi test kodu ;   Ekleme metodunda eklemeyi yaptıktan sonra veri tabanından eklenmiş mi diye kontrol ettim,bunun için aşağıdaki metodu yazdım ;  public void eklenmisMi(String kalkisTarihi,String varisTarihi,Integer saticiId) .Bu metodu biletEkle metodundan çağırdım ve veri tabanından eklenen bileti çekmeye çalıştım.  Test için aşağıdaki metodu yazdım ;  Örnek olarak aşağıda bileti veri tabanına eklemeye çalıştım ;    Ekledikten sonra veri tabanından çekmeye çalıştı, eklenmiş mi? diye ;  Ve eklenen bileti veri tabanından çekerek ekran çıktısı olarak verdi ;     * Müşterinin bilet alması ve test kodu ;   Ekleme yaptıktan sonra kontrol etmek için eklediğim bileti ekrana yazdırdım ,test metodunu aşağıdaki gibi çağırdım;    Müşteri için test metodum aşağıdaki gibidir ;  Örnek olarak id’si 30 olan bileti almaya çalıştım ;  Test çıktısı aşağıdaki gibidir ;      *AŞAMA 3*   * Satıcının kendi sisteminden bilet silmesi testi ;   Öncelikle bileti silmeye çalıştım daha sonra veri tabanından çekmeye çalıştım, eğer hiç sonuç dönmezse while(rs.next()) e hiç girmez ve bilet silinmiş demektir.Silme testi için silme fonksiyonundan test metodunu aşağıdaki gibi çağırdım ;  Test metodum ise aşağıdaki gibidir ;        Örnek olarak 29 numaralı bileti sildim ve veri tabanından test işlemini yaptım ;  Ekran çıktısı aşağıdaki gibi oldu ;    Veri tabanından kontrol edersem gerçekten silinmiş mi diye aşağıdaki gibi bilet bulunamaz ;   * Müşterinin kendi sisteminden bilet iptal etmesi test kodu;   Öncelikle bileti silmeye çalıştım daha sonra veri tabanından çekmeye çalıştım, eğer hiç sonuç dönmezse while(rs.next()) e hiç girmez ve bilet silinmiş demektir.Silme testi için silme fonksiyonundan test metodunu aşağıdaki gibi çağırdım ;  Test kodu ise aşağıdaki gibidir ;    Örnek olarak müşterinin daha önceden almış olduğu 42 no’lu bileti sildim ;    Sildikten sonra veri tabanında musteri\_bilet (manytomany) tablomu test işlemine tabi tutunca çıktı aşağıdaki gibi karşıma çıktı ;  Veri tabanından da gerçekten silinmiş mi diye kontrol edersem, karşıma hiçbir bilet sonucu çıkmaz ;    ***2-SINGLETON TASARIM KALIBININ KODU***  Singloten tasarım kalıbını veri tabanına bağlanırken birden fazla nesne oluşturmamak için kullandım. Veri tabanına bağlantı birçok yerde lazım oluyor, bu nedenle her yerden aynı nesneyi oluşturunca hem düzensiz bir kodlama hem de daha karmaşık, kod kalabalığı olan bir tasarım şekli oluşmuştu.İşte singleton burada bana bir avantaj sağladı.  Constructor’ın private olması gerekiyor.Ama sadece constructor’ı private yaptığımda classtan nesne oluşturduğumda sorun oluşur.O yüzden nesneyi başka bir yerde oluşturmadan kendi içinde yani constructor da oluşturdum.Ayrıca nesnenin kendisine dönen bir metot yazmam gerekir.Bunun içinde DBConnection sınıfımda private,static ve global bir değişken tanımladım;    Bu nesneyi dışarıya açmak için getter metodunu yazdım ve bu metodu static yaptım.Bu durumda diğer sınıflarda nesne oluşturmak için getDBConnection() metodunu çağırmam yeterli.Test için 2 husus var birincisi nesneyi getter metodunda new’le oluşturdum.Bir de sayı diye bir değişkeni bu metot her çağrıldıkça 1 arttırıp , ekrana yazdırdım.Bu sayede nesne nin kaç kez çağrıldığının ve kaç kez oluştuğunu test ettim, nesne her oluştuğunda ise ‘Ben oluştum’ diye ekrana yazdırdım.  Test sonucunda ise ekran çıktısı aşağıdaki gibi olur ;  Görüldüğü gibi nesne oluşturulması için 12 kez çağırılırken sadece 1 kez oluştu ve ‘Ben oluştum’ yazarak test sonucunda doğru çalıştığını test etmiş oldum.  Diğer class’larda nesne’yi kullanmak için getter metodunu çağırarak aşağıdaki gibi kullanılmıştır;    ***Yukarıdaki kodu yazarken bazı sorunlarla karşılaştım , sorunlar ve çözümler şu şekildedir;***   1. Eğer ben dbConnection nesnesini hiç kullanmayacak bile olsam bu nesne mutlaka 1 kez oluşuyordu, bu istenen bir şey değil.İstenen şey ihtiyaç duyulduğu anda 1 kez oluşsun daha sonra bu nesne oluşturulmasın , ilk oluşturulan nesne kullanılsın.Yani **Lazy Loading** yapmış oldum.Buna göre de nesneyi tanımladığım anda new ile yeni nesne oluşturmak yerine bunu getter metodunun içerisinde yaptım.Ayrıca sadece null olduğu durumda yeni nesne oluşturulmasını sağladım. 2. Diğer bir sorun ise aynı anda iki yerden çağrıldıysa o zaman nesne null iken 2 tane oluşturulma isteği gelir.Bu sorunu çözmek için metodun başına ‘synchronized’ anahtar sözcüğünü kullandım.Ayrıca sadece nesne null ise bu anahtar sözcüğünü kullanmasını sağladım.Bu kontrolü yaparakta **Double Checked Locking** yapmış oldum. | |