**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** | **Tasarım (Design-2)**  **Hangi yazılım tasarımı kalıplarını kullanıcak?**  **1 adet Creational (Oluşturucu), 1 adet Structural(Yapısal) ve 1 adet Behavioral (davranışsal) tasarım kalıplarının özellikleri , hangi ihtiyaçları karşılamak için kullanıldığının açıklanması.** |
| **AÇIKLAMA** | |
| **Hangi yazılım tasarımı kalıplarını (software design patterns) kullanacaksınız?**  Projemde aşağıda belirttiğim yazılım tasarım kalıplarını kullanmak istiyorum.  Creational (Oluşturucu) tasarım kalıbı olarak : SINGLETON tasarım kalıbı,  Structural (Yapısal) tasarım kalıbı olarak : FACADE tasarım kalıbı,  Behavioral (Davranışsal)tasarım kalıbı olarak : COMMAND tasarım kalıbını kullanmak istiyorum.  **Projenizde kullanmak üzere seçtiğiniz 1 adet Creational (Oluşturucu), 1 adet Structural(Yapısal) ve 1 adet Behavioral (davranışsal) tasarım kalıplarının özelliklerini açıklayınız ve hangi ihtiyaçları karşılamak için projenizde kullandığınızı ifade ediniz.**   1. Creational (Oluşturu) tasarım kalıbı : SINGLETON     Bu tasarım kalıbının özelliklerini ve kullanmak isteme sebeplerimi şöyle izah edebilirim ;   * Uygulama geliştirirken en popüler ve bilindik oluşturucu tasarım kalıbı olması, * Popüler ve bilindik olmasından ötürü bir problemle karşılaştığımda daha fazla kaynağa erişme imkanımın olması, * Bir sınıftan birden fazla örneğin oluşturulmayıp, sadece bir örnekle kullanılabilmesi ve bunun sonucunda kod karmaşıklığının önüne geçilebilmesi, * Projemde veri tabanında Singleton tasarım desenini kullanmayı düşünmem çünkü veri tabanı sınıfının nesnesini çoğu sınıfta kullanmam lazım . Onun yerine tasarım kalıbı sayesinde sadece bir kez örnek oluşturabilme avantajımın olması , * Sadece veri tabanı sınıfında değil , örneğin entity sınıfları olan satıcı, müşteri, bilet gibi sınıfların sık sık nesnelerini oluşturmam nedeniyle singleton tasarım kalıbıyla projemin daha sade, anlaşılır, kod tekrarı olmadan ve düzenli bir şekilde tamamlanacağına inanmam sebebiyle singleton tasarım kalıbını kullanmak istiyorum.  1. Structural (Yapısal)tasarım kalıbı : FACADE   Bu tasarım kalıbının özelliklerini ve kullanmak isteme sebeplerimi şöyle izah edebilirim ;   * Karmaşık olan bir sistemin kullanıcıya daha basit hale getirmesi ve bunun kullanım açısından kolaylığının artması, * Facade tasarım kalıbıyla ilgili bulduğum dokümanların güvenilir olması ve bir problemle karşılaştığımda bu kaynaklardan yararlanma imkanımın daha fazla olması , * Projeme uygulama konusunda ekleme, silme, güncelleme ve arama işlemleri olacağından benzer metotları inteface yardımıyla daha kolay bir erişim sağlayabileme kolaylığımın olması, * Projemde aşağıdaki metotlar için tek bir arayüz yapıp kullanmayı düşündüm ;        * Ayrıca her arayüz için kullandığım aşağıdaki buton metotları içinde kullanmayı düşündüm ;      1. Behavioral (Davranışsal)tasarım kalıbı : COMMAND   Bu tasarım kalıbının özelliklerini ve kullanmak isteme sebeplerimi şöyle izah edebilirim ;   * Bir fonksiyon birden fazla işlev için kullanılabilir, * Güncel bir tasarım kalıbı olması , * Doküman açısından sıkıntısı olmaması , * Kodun daha sade , düzenli, anlaşılır ve metotlar açısından daha kolay erişim sağlaması , * Örnek olarak projemde hem ekleme hem silme işlemi var. Bir sınıfa bir fonksiyon yazıp crudIslem yap diyerek diğer sınıflarda bu ekle metodu içinde, sil metodu içinde bu metotu çağırıp, gerekli metodu çağırabileceğimi düşündüm. | |