Randevu Takip Sistemi - Rapor

Randevu takip sisteminde kullanacağım nesnelerin sınıfların adı;

Hasta bilgisini tutan sınıf, (Patient)

Doktor bilgisini tutan sınıf, (Doctor)

Randevu bilgisini tutan sınıf (Appointment) ve

Hasta ve Doktor ortak özellik içerdiğinden ‘PeopleData’ adında bir soyut sınıf kullandım.

Hasta ve Doktor bilgilerinin ortak özelliklerini bahsedilen soyut sınıftan miras aldım.

Oluşturduğum bu sınıfların hepsinde bilgiyi ayrı ayrı tutması için her sınıfa özel ArrayList ekledim. ‘Controller’ içinden aldığım veriyi kullanabilmek için ArrayList’lerimi ‘static’ olarak tanımladım.

Ayrıca bazı noktalarda ‘static’ tanımlamak daha zorlu olacağından Constructor’lardan birine ‘overload’ işlemi uygulayıp boş bir yapı elde ettim.

Bu yapı sayesinde istediğim sınıfta bu yapının bir referansını oluşturarak sınıflardaki metotları çağırmayı hedefledim.

‘Test’ sınıfında yaptığım işlemlerin çalışıp çalışmadığını denedim.

Daha sonra GUI kısmında verileri başka sahneden alıp ‘TableView’ öğesine aktardım ve böylece ana sayfada tüm randevuların gözükmesini sağladım.

Seçilen randevunun silinmesi ve şikayet bilgisinin gösterilmesi için 2 ayrı buton daha ekledim.

Kullanıcı; veri girerken hata yaptığında, boş veri girip butona tıkladığında, herhangi bir randevu seçmeden ‘sil’ butonuna bastığında, randevu seçmeyip şikayetini göstermek istediğinde ‘try-catch’ kullanarak kullanıcıya hata mesajı vermeyi ve programın doğru çalışmasını devam ettirmeyi amaçladım.

Oluşturulan randevuların ‘.txt’ dosyasına bastırmak istenildiğinde çıkış sahnesinde kullanılabilecek ‘Export and Exit’ adında bir buton kullandım. Böylece veriler dışa aktarılmış oldu.

Ayrıca yaptığım her ekleme ve silme işlemlerini terminalde göstermek için ‘interface’ de metotlar tanımladım.

ID’nin 6 haneli olabilmesi için ‘setter’ fonksiyonlarda kontrol ettirdim.

Mehmet Ali Gülyurdu / 170421013