



Mehmet Akif Ersoy Üniversitesi

Bucak Teknoloji Fakültesi

Algoritma 2 Final Ödevi

Ödev konusu: Hastane randevu sistemi

Hazırlayanlar:

* Ahmet Nusret ASMA 2012101011
* Selim ÖNER 2012101009
* Denizhan DURSUN 2012101023

Öğretmen: Emrah HANÇER

2.) Problemin tanımı:

Hastaneye gitmek isteyen hastalarımızın hastanelerde yoğunluk oluşturmaması için ve zamanın çok kıymetli olduğu bu dönemde alınan sırayı beklemek zorunda kalmamaları için daha önce belirlenen bir saatte gelmelerini ve direk muayeneye alınmalarını sağlayacak, hastanelerdeki gereksiz yoğunluğu azaltacak bir hastane randevu sistemine hastanelerde ihtiyaç duyulmaktadır.

3.) Kullanılan teknolojiler ve tanıtımları:

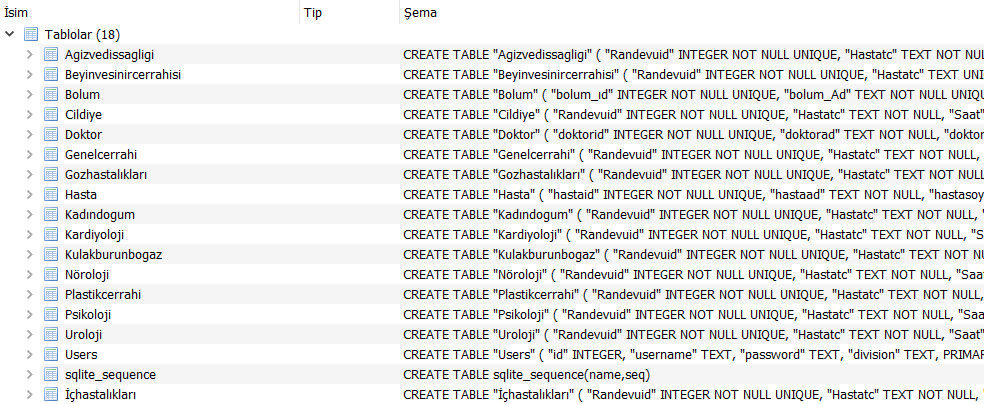
1-) NetBeans:

NetBeans java için bütünleşmiş bir geliştirme ortamıdır. NetBeans uygulamaların mödül denilen yazılım bileşenleri ile geliştirilmesine olanak tanır. Ayrıca PHP, C, C++, HTML, JavaScript gibi farklı diller için de kullanılabilir.

2-) SQLite:

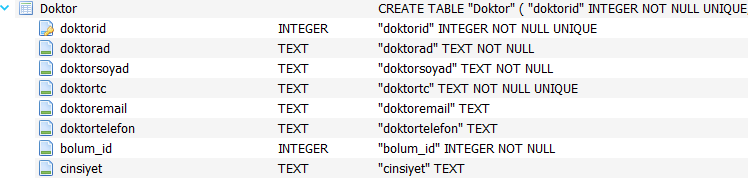
SQLite, C dili tabanlı bir veritabanı sistemidir. Birçok veritabanı sisteminin aksine SQLite server tabanlı değil, sistemin içine gömülü olarak kullanılan bir veritabanıdır. SQLite genellikle PostgreSQL syntax'lerini kullanır. SQLite, lokal kullanılan sistem uygulamaları için çok popüler bir veritabanıdır. En çok kullanılan veritabanı sistemi olup olmadığı tartışılabilir, ancak günümüzde birçok alanda kullanıldığı da göz ardı edilmemelidir.

4.) Veritabanı tasarımı ve açıklaması:



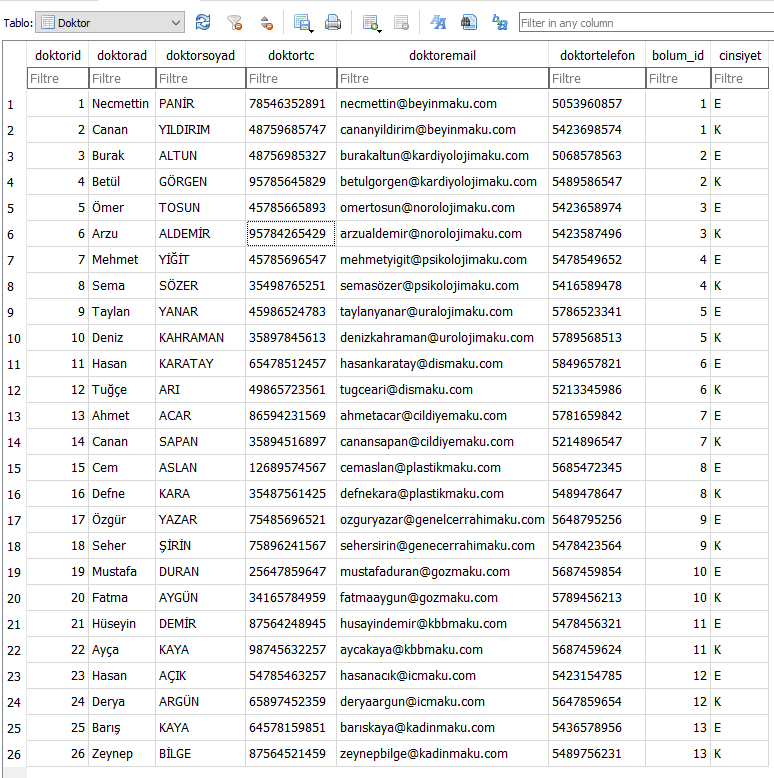
Tablo4.1

Tablo4.1 de gördüğümüz veritabanı yapısıdır. Burada tabloların isimlerini ve şema olarak nasıl yazıldıklarını görüyoruz. Veritabanına en geniş bakış açısı bu tablodur.



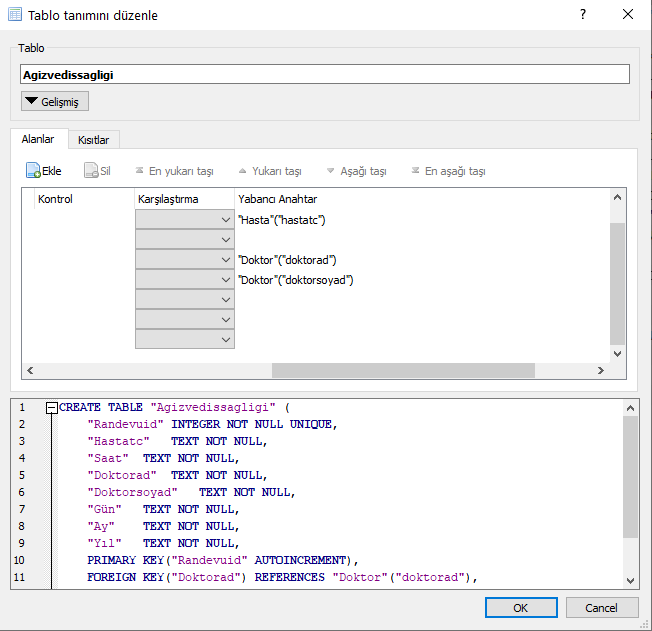
Tablo4.2

Tablo4.2 de gördüğümüz kısım doktor tabolomuzun içindeki titles (başlıklar), bu başlıkların elemanlarının hangi türlerde olduğu ve tablomuzun birincil anahtarını görüyoruz.



Tablo4.3

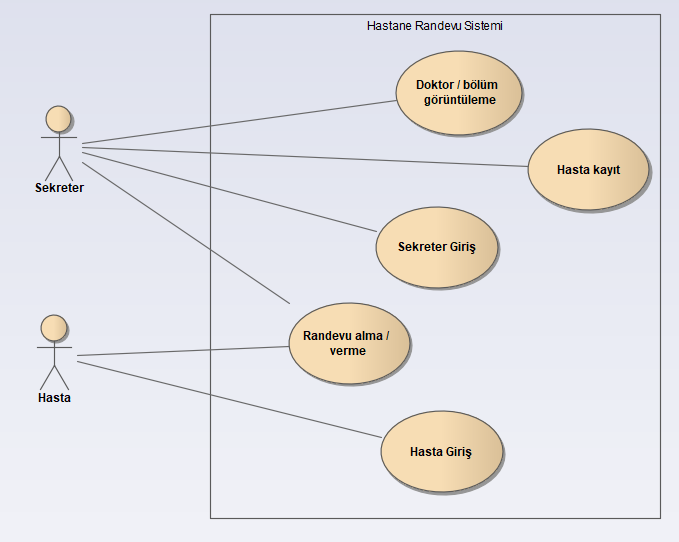
Tablo4.3 de ise doktor tablomuzun içerisindeki bilgileri görüyoruz.



Tablo4.4

Tablo4.4 de ise Ağız ve diş sağlığı bölümünün tablosu olan Agizvedissagligi tablosunu görüyoruz. Bu tabloda dikkat çekmesi gereken şey Foreign keys dir. Foreign key başka bir tablo ile bizim tablomuzu ilişkilendirerek elimizdeki veriye ait karşılıkları bulup çıkartmamızı sağlar. Burada “Hasta”(“hastatc”) kısmında Hasta tablosundaki hastatc kısmındaki veriyi ve o satırdaki verileri bu tabloda kullanma imkanı sağlamış oluyoruz. Tablo4.4 de aynı işlem “Doktor”(“doktorad”) ve “Doktor”(“doktorsoyad”) kısmında da yapılmıştır.

5.) Uygulamadaki modüllerin tanıtımı:



Tablo5.1

Tablo5.1 Hastane Randevu Sisteminin use-case diyagramıdır. Burada hasta ve sekreter kullanıcılarının olacağı ve bu kullanıcıların sistemde neler yapabileceği görünmektedir. Uygulama yapmaya başlamadan önce çıkartılması gereken, neyin nerede olacağını, nerelere kimlerin erişebileceğini gösteren temel tablodur.



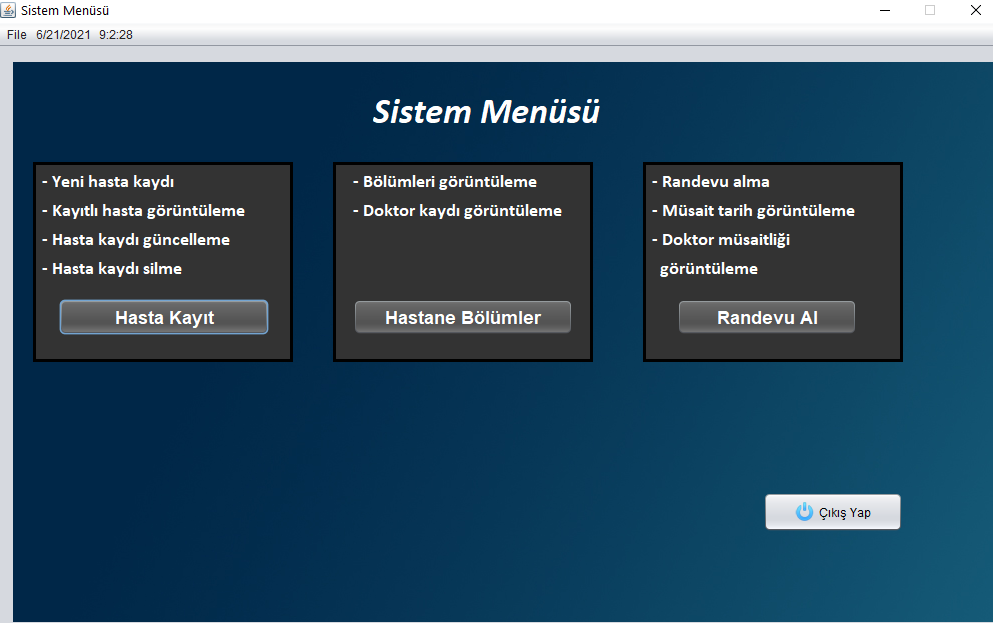
Tablo5.2

Tablo5.2 Uygulamamızın açılış sayfasıdır. Sekreter ve hasta girişi olmak üzere iki yola ayrılmaktadır.



Tablo5.3

Tablo5.3 sekreter giriş panelidir.



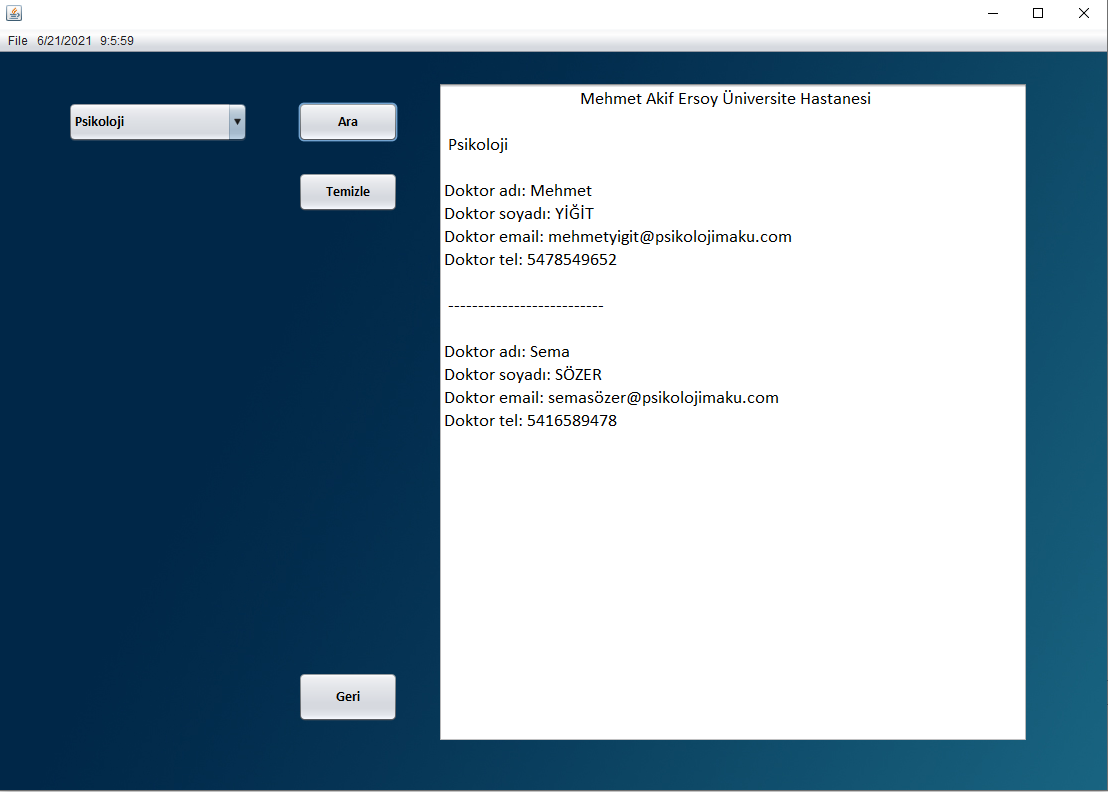
Tablo5.4

Tablo5.4 sekreter menü sayfasıdır. Bu sayfada sekreter yapmak istediği işlemi seçer.



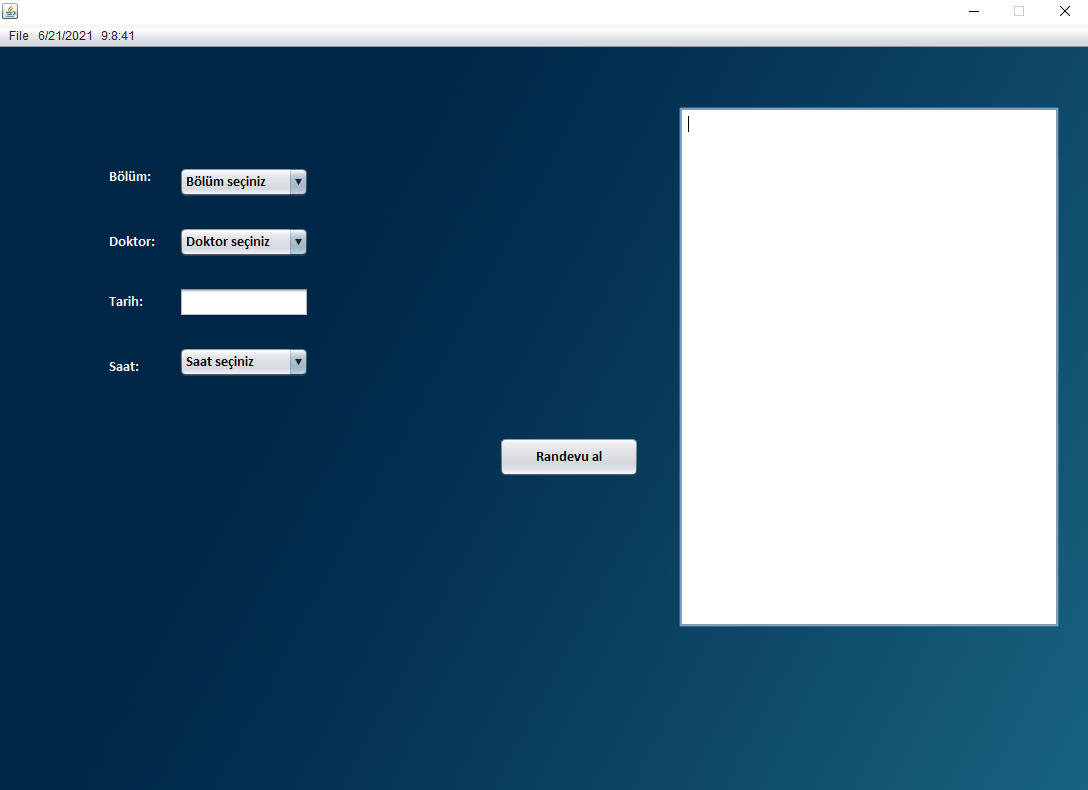
Tablo5.5

Tablo5.5 sekreterin hasta kaydı yapmasını sağlayan paneldir.



Tablo5.6

Tablo5.6 sekreterin doktor araması yapmasını sağlayan paneldir.



Tablo5.7

Tablo5.7 sekreterin hastalara randevu verebildiği ve hastalarında kendilerine randevu alabildiği paneldir.



Tablo5.8

Tablo5.8 Hasta girişini sağlayan paneldir.

6.) Sonuç:

Projemize başlarken mümkün olan en iyi sistemi yapmak hedefimizdi. Bu hedef doğrultusunda ilk toplantımızda Tablo5.1’deki use-case diyagramını çizdik ve o diyagram üzerinden ilerlemeye karar verdik. Diyagramımızdaki hedefleri başarılı bir şekilde gerçekleştirebildik. Sekreterin erişebildiği Doktor ve bölüm görüntüleme sayfası, hasta kayıt sayfası sekreter giriş sayfası ve randevu verme sayfamız var. Hastanın erişebildiği randevu alma ve hasta giriş sayfalarımız bulunmaktadır.

Eksik yanları projenin kısıtlı olmasından dolayı çok yoktur. Aklımıza gelen ileride yapılabilecek ve bizlerin eksik hissettiği yerler:

1) Hasta bilgilendirme

* 1. E-mail ve SMS bilgilendirmesi

1. Biraz daha estetik, göze hitap edecek çalışmaların yapılması
2. Tarih alma kısmı geliştirilebilir. JCalender kullanılabilir.

Proje başlangıcından itibaren düzenli olarak beraber çalışarak, yazılım dünyasındaki takım çalışması kavramını yakından öğrendik. Her gün kısa süreli toplanıp proje hakkında ki fikirlerimizi tartışarak Scrum kavramı hakkında tecrübe kazandık. Proje üzerinde çalışırken hem Java hem de SQL hakkında yeni bilgiler edindik ve kendimizi geliştirdik.