**MARMARA ÜNİVERSİTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ**

**VERİ YAPILARI PROJESİ**

**-Kütüphane Otomasyonu-**

Muhammed Fatih YILMAZ

mfth78@hotmail.com

171420501

Ömer Alptuğ YENİPINAR

omertoyenipinar@gmail.com

170420022

KULLANDIĞIMIZ VERİ YAPILARI

* Stack
* Linked List
* Queue
* Quick Sort
* Linear Search

**Proje Tanıtımı**

Projemizi sıfırdan tasarladık. Projemiz kütüphanede yapılan işlemleri sağlayan bir otomasyondur. Öncelikle kullanıcı verilerini saklamak için veri yapılarını kullanmak istedik ve bu sebeple kullanıcı kayıtları yaparken Linked List kullandık. Linked List üzerinde gezinerek kullanıcı verilerini topladık. Bununla birlikte kullanıcı işlemleri için Linked List’in yanı sıra Stack yapısına da yüzeysel olarak değindik. Kullanıcıları bastırırken Stack yapısını kullanarak kullanıcı göster fonksiyonuyla son eklenen kullanıcıları ekrana bastırdık. Kullanıcı kitap alırken kitap kontrolünü sağlayıp eğer o kitap başkası tarafından alınmış ise Queue yapısını kullanarak o anda almak isteyen kullanıcıyı sıraya soktuk. Bu sayede kitaba sahip olan kullanıcı kitabı geri teslim ettiğinde sıradaki kullanıcının hesabına kitabı yazdık. Hem kullanıcılar hem de kitaplar için ayrı ayrı txt dosyası oluşturduk. Database yerine davranarak txt dosyalarını kullandık. Interface encapsulation ve İnheritance kavramlarını projemizde aktif olarak kullandık. Sort algoritmasıyla kitapları sayfa sayılarına göre sıraladık. Bu aşamada Quick sort kullandık. Kitap ararken ise linear search ile ascii kodlarından kitap isimlerini arattık ve buldurduk. Burada bilgilerini ekrana bastırdık. Kullanıcı ve admin için yeni arayüz oluşturduk.

Programımızda bulunan arayüzler 11 adettir bunlar şu şekilde sıralanmaktadır:

1. Admin Kontrol

Bu bölümde 5 tane arayüz tasarladık

* Kullanıcı Ekleme
* Kullanıcı Silme
* Kitap Ekleme
* Kitap Silme
* Kullanıcıları Listele

1. Kullanıcı Menü

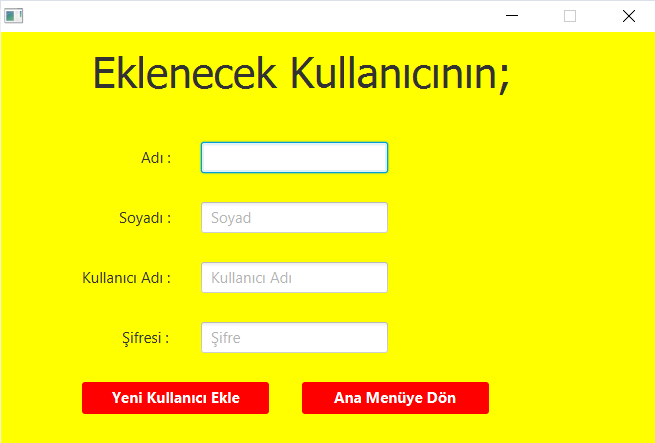
* Kitap Al
* Kitap Bırak
* Kütüphaneyi Göster
* Ana Menü

1. **Admin Kontrol**

Admin direkt olarak sisteme giriş yapabiliyor. Ardından karşısına 6 adet buton çıkıyor. Bu butonlar yukarıda belirttiğim arayüzlere geçiş yapmayı sağlıyor. Ekleme silme işlemleri bu arayüzden gerçekleştiriliyor.

**Kullanıcı Ekleme**

Kullanıcılar admin sisteminden Linked Liste ekleniyor. Eklenen kullanıcılar kayıt olması için database yerine daha hızlı çözüm sürerek Users.txt’ye kaydediliyor.

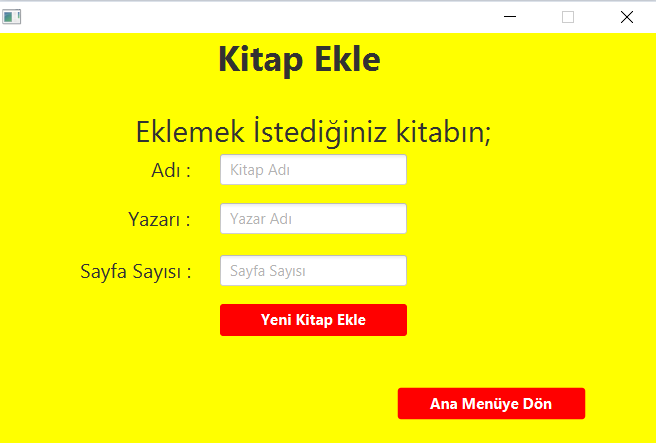


**Kullanıcı Silme**

Burada username’i verilen kullanıcı önce Linked Listten ardından refreshBookList methoduyla kullanıcı listesinden siliniyor bu method Book classının içinde bulunup her kullanıcı removeUser işlemi gerçekleştiğinde methodun içinde gerçekleşiyor.

**Kitap Ekleme**

Yukarıdaki ekleme işlemleri gibi burada da kitaplar ekleniyor tek fark burada ArrayList kullandık.



**Kitap Silme**

Aynı şekilde ismi verilen kitap eğer ArrayListte bulunursa burada siliniyor

**Kullanıcıları Listeleme**

En zorlandığımız kısımlardan bir tanesi de bu kısımdı normalde Traverse işlemi yapıp Linked List üzerinde gezdiğimizden Kullanıcıları return edemiyorduk. Yoksa döngü sonlanıyordu. En sonunda Liste üzerinde teker teker gezip istediğimiz indexteki kişiye gelince döndüren metodu yaptık gezinmek içinse kullanmak istediğimiz yerlerde for ile tek tek Linked Listin size’ı kadar Liste üzerinde index verip kullanıcıları çektik. Fakat kullanıcılar ters basılıyordu. Bunuda engellemek için Stack kullanıp Last in First out mantığıyla listeleyeceğimiz kullanıcıları ters çevirip bilgilerini bastırdık.



1. **Kullanıcı Menü**

Aşağıdaki menüde kullanıcılar istedikleri işlemleri seçerek o pencereye yönlendiriliyorlar.



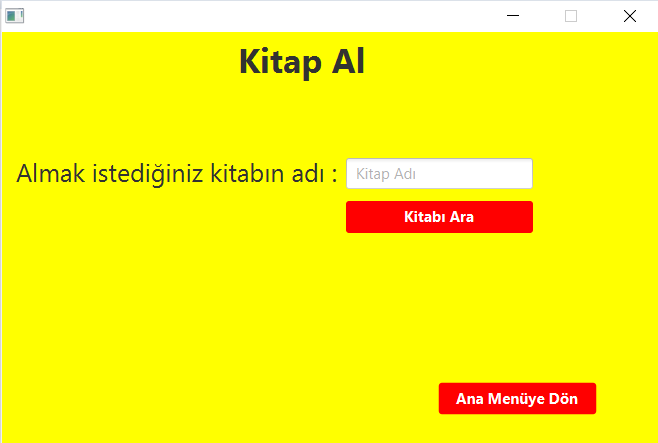
**Kitapları Göster**

Bu method sayesinde kullanıcılar kitapları Quick Sort algoritmasını kullanarak sayfa sayısına göre sıralayabiliyor. Kitap ara methodu ise yazdığınız kitap ismi eğer kitaplar arasında varsa linear search arama algoritmasını kullanarak kitapları önünüze getiriyor.



**Kitap Al**

Burada kullanıcılar almak istedikleri kitapları yazıyor. Yazdığı kitap kimse tarafından alınmamışsa Book.txt ‘de verdiğimiz kullanıcı alanına kullanıcı adı yazılıyor. Eğer hiçkimse almamışsa default olarak null geliyor. Eğer herhangi bir kullanıcı kitabı aldıysa başka almak isteyen kişi Queue yapısı kullanılarak sıraya ekleniyor eğer alan kullanıcı kitabı bırakır ise hemen beklemeden sıradaki kullanıcıya kitabı devrediyor.



**Kitap Bırak**

Kitap bırak fonksiyonu ile yazılan kitap eğer kullanıcıda ise kitabı bırakıyor sırada kullanıcı var ise ona geçiyor yoksa null olarak kitabın kullanıcısı ayarlanıyor.

**Kaynaklar**

<https://www.geeksforgeeks.org/java-program-to-sort-names-in-an-alphabetical-order/>

<https://www.programiz.com/dsa/quick-sort>

<https://stackoverflow.com/questions/10275587/finding-loop-in-a-singly-linked-list>