Gebze Technical University

Computer Engineering

CSE 222

2017 Spring

HOMEWORK 01 REPORT

MUSTAFA BİNGÜL

141044077

Course Assistant:

Nur Banu ALBAYRAK

**Q2-** C programlama dilinde encapsulation struct lar tarafından sağlanır bir (.h) dosyasında structlarımızı tanımlarız. Verilere ulaşım ve methodlar ise prototiplerimizi tanımladığımız bir (.h) dosyası olur. Bu (.h) dosyasındaki prototiplerin implementini (.c) dosyasında sağlarız. Prototipi olan fonksiyonlar public tir. Prototipi olmayan fonksiyonlarda implement edersek bunlarda private olur.

Bu durumun sağlanması başka bir teknik ise function pointerlar ile sağlanabilir.

**1.Requirements**

**1.1-Overall Description**

**Part-1**

Ödevde basit bir kütüphane otomasyonu istenmiştir.

Tek yönetici vardır. (User ID:0 Password:123)

Yönetici kitap veya kullanıcı eklediğinde id leri otomatik sistem tarafından atanmaktadır.

Kullanıcı veya kitap ekledikten sonra yönetici id leri görmek isterse menülere ulaşıp görebilir.

Yönetici kullanıcı ekleyip çıkarabilir aynı şekilde kitapta ekleyip çıkarabilir. Birden fazla kullanıcı vardır ve istedikleri kitabı ödünç alabilir ve geri bırakabilirler.

Son olarak kitap ve kullanıcı verileri (.csv) uzantılı dosyada tutulmaktadır.

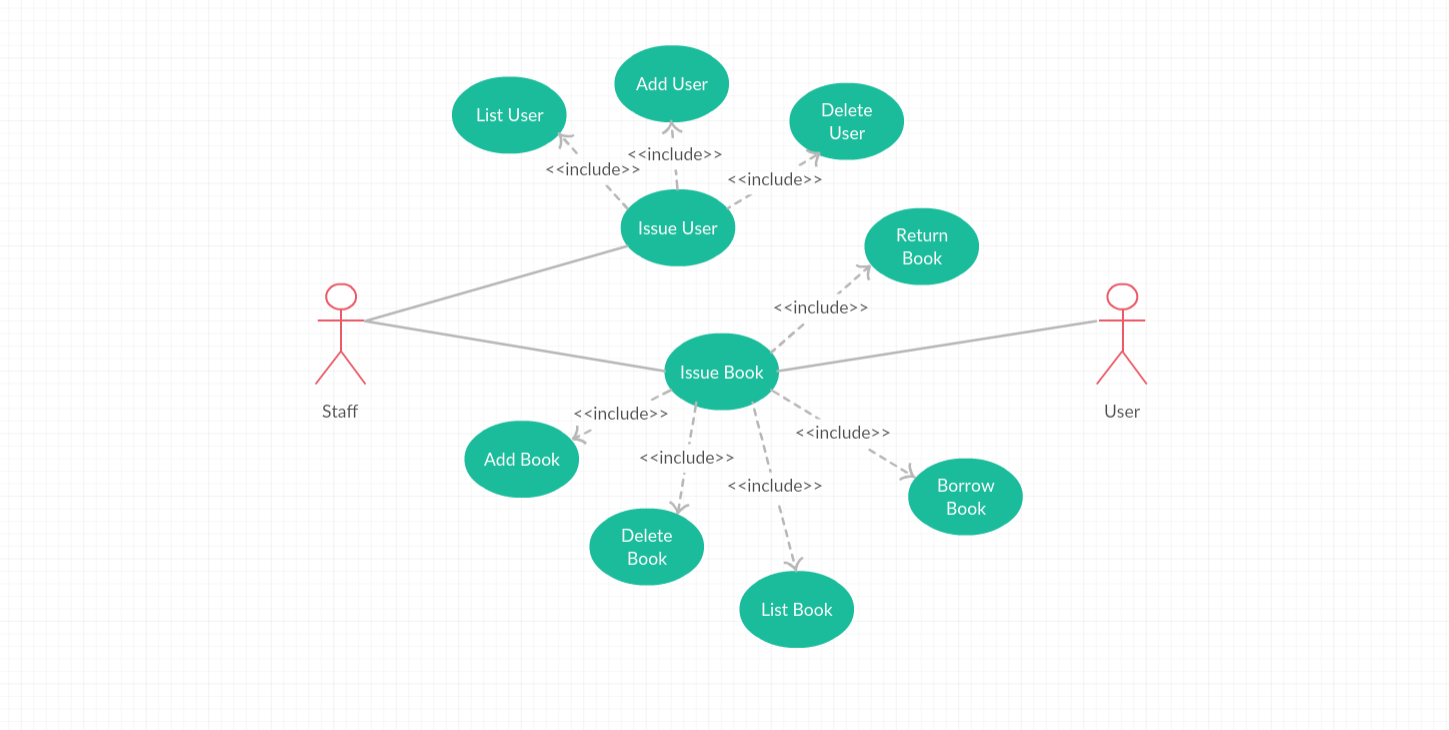
**Part-2**

C programlama dilinde encapsulation ve verilerin bir arada nasıl tutulduğu hakkında raporda bilgi vermemiz istenmiştir. Ve bu durum sağlanmıştır.

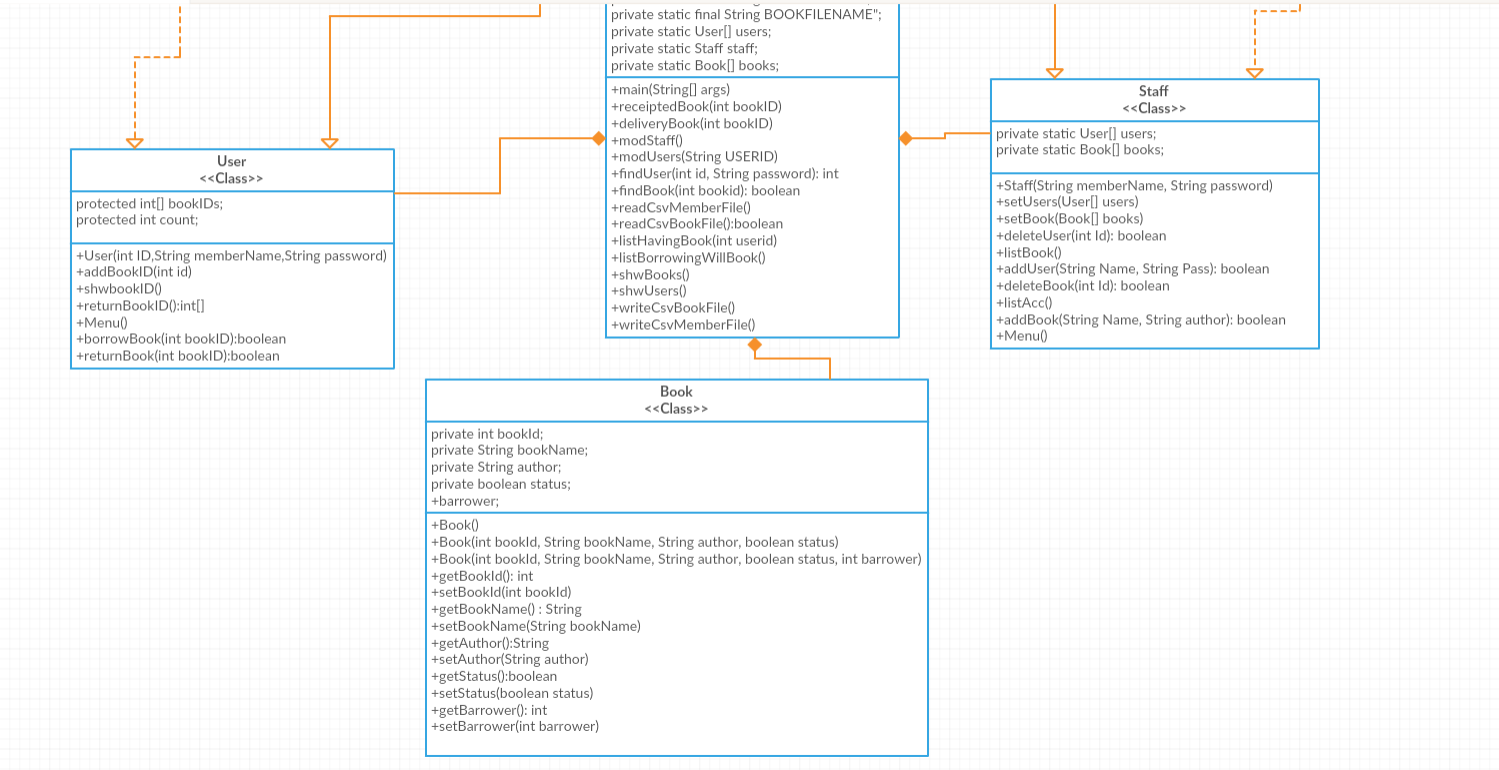
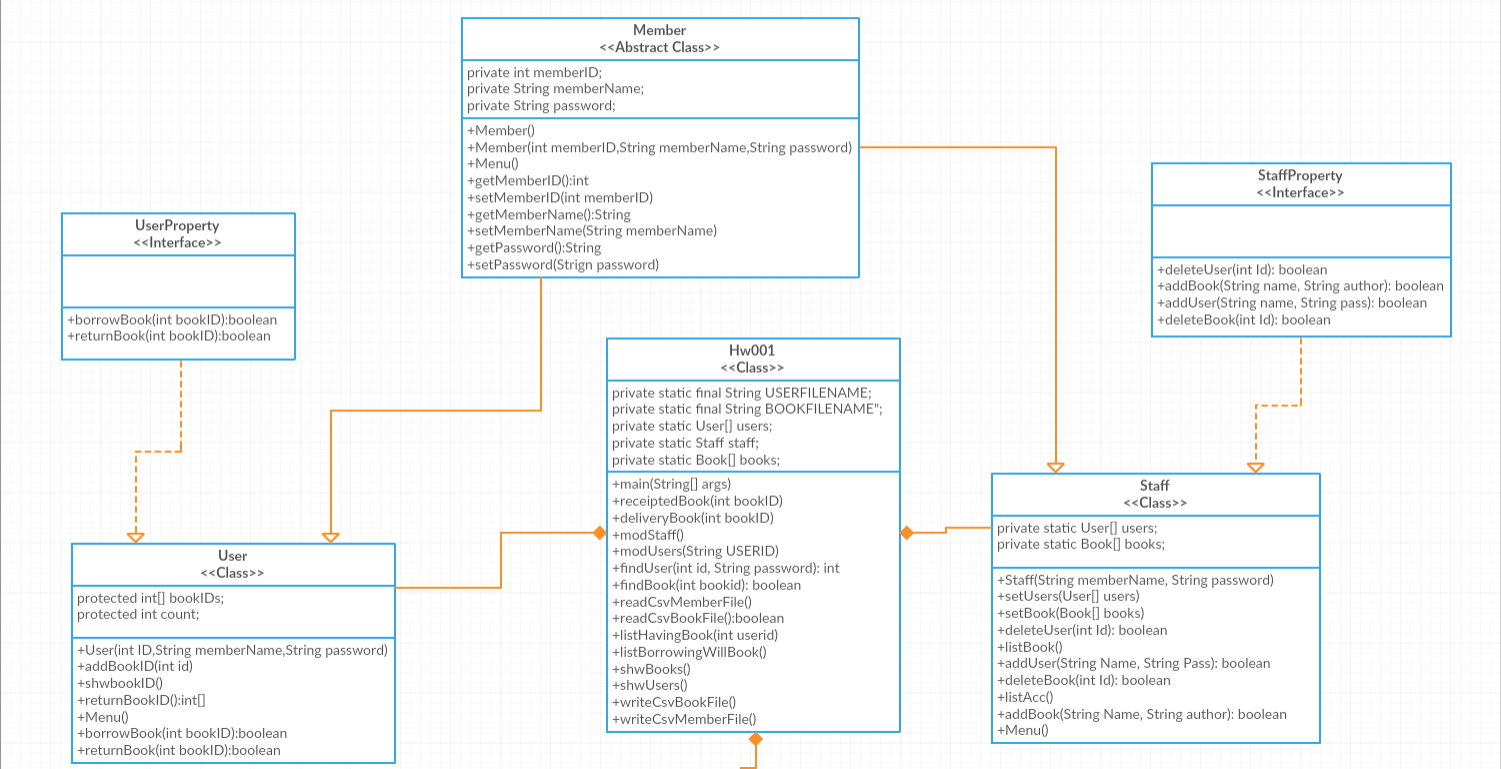
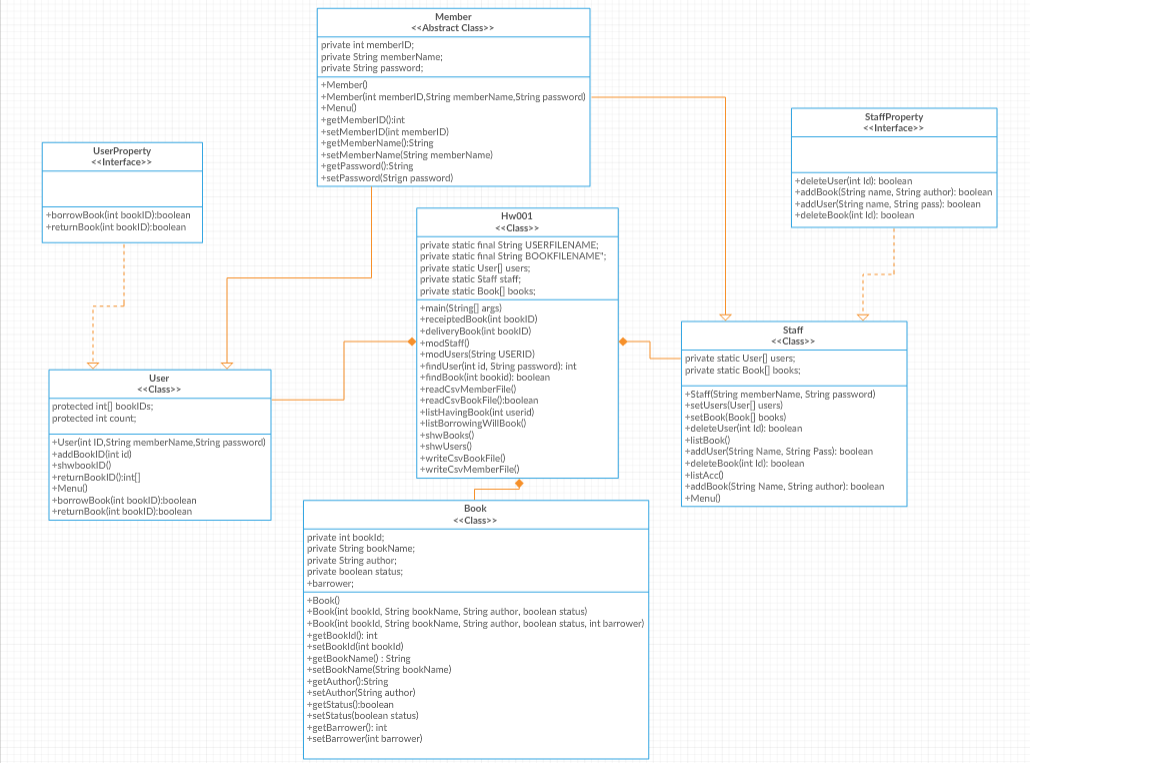
**1.2-System Requirements**

1.2.1 book.csv ve mem.csv dosyaları.

**2. Use Case Diagrams**



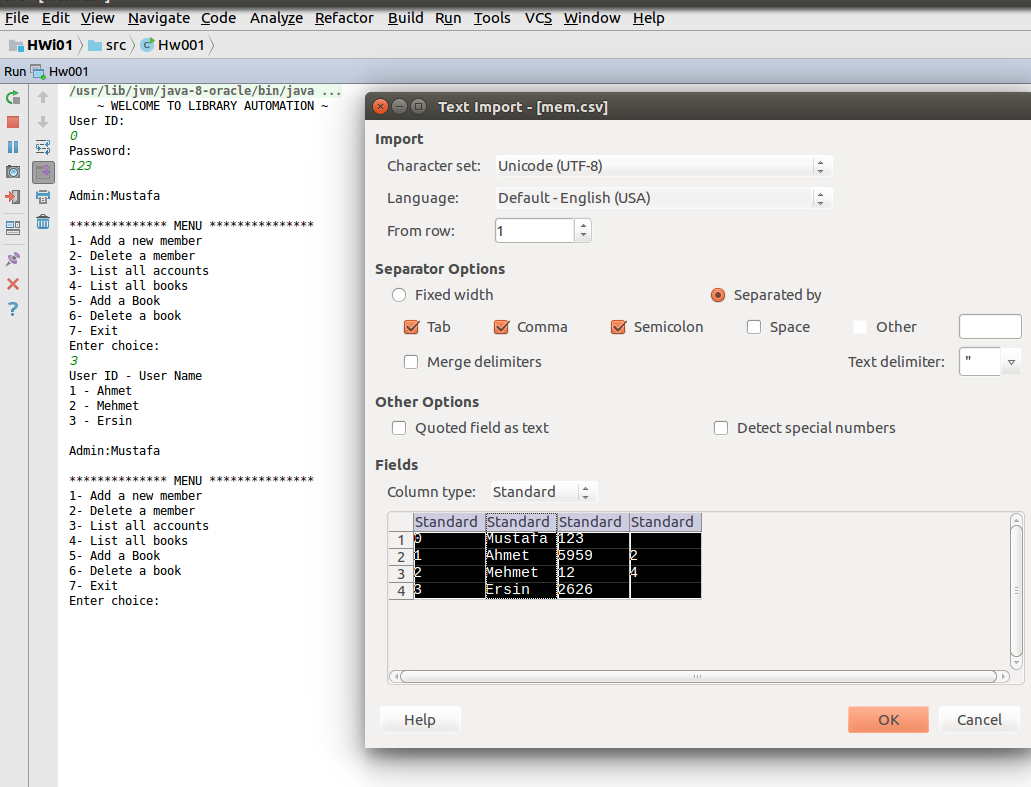
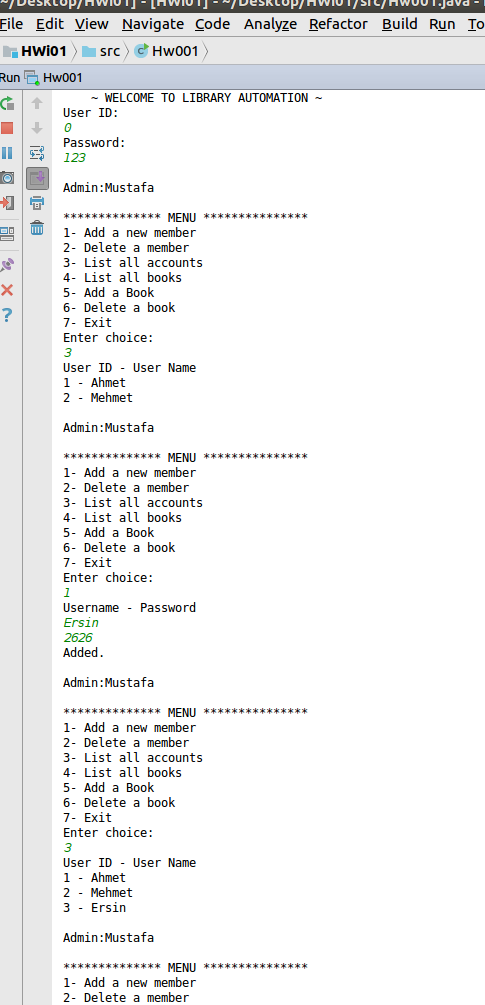
**3. Class Diagrams**



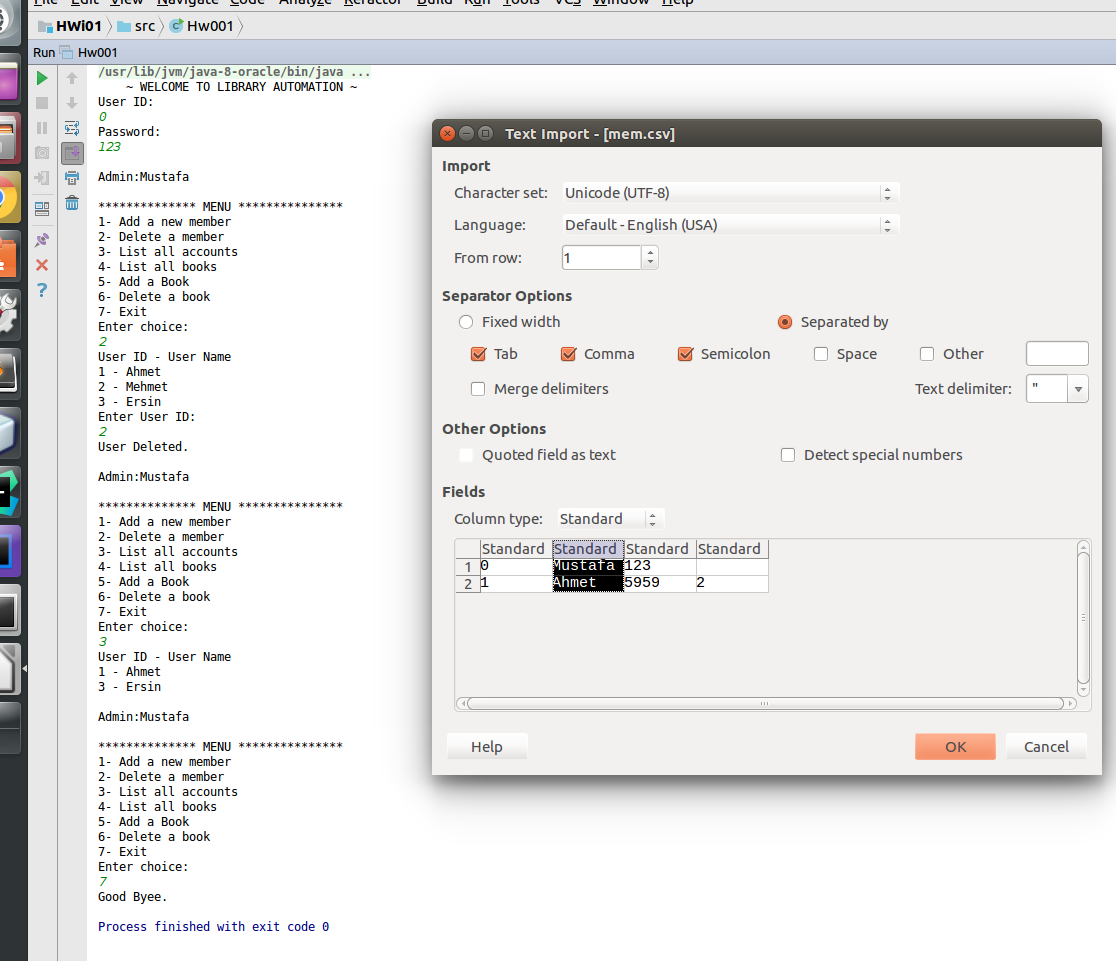
**5. Problem Solutions Approach**

Basit bir kütüphane otomasyonu yapmamız istendi. Bu durumu düşündüm ve kütüphane yöneticisi, kütüphane kullanıcılarının ve kitap nesnelerinin olması gerektiğine karar verdim. Book,Staff,User,Member sınıflarımı ve StaffProperty ve UsersProperty interface lerini oluşturdum. Staff sınıfımı abstract olan Member sınıfımdan extend edip StaffProperty interface imden implement ettim. Aynı şekilde User sınıfımıda Member sınıfımdan extend edip UserProperty interface inden implement ettim. Ve main imi yazdığım Hw001 sınıfımın içinde User ve Book arrayları oluşturup ve Staff objesi oluşturdum. Ve ödevde istenilen yönetici için ve kullanıcılar için gerekli methodları yazarak projeyi böyle gerçekleştirdim.

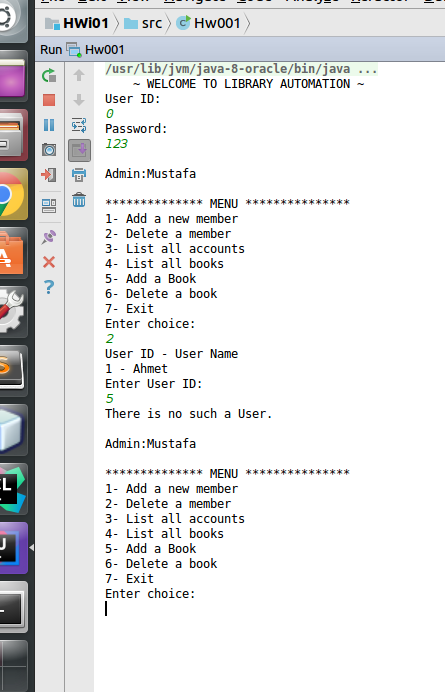
**6. Test Cases**



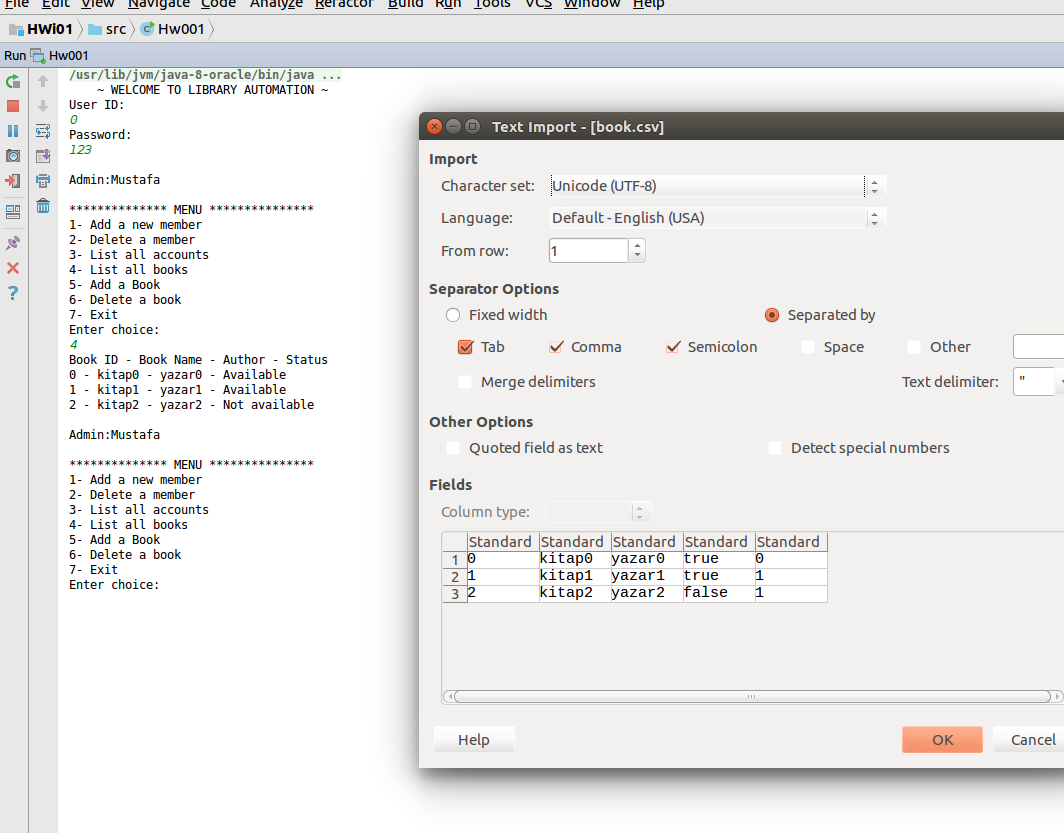
6.1-Kullanıcı ekleme başarılı.



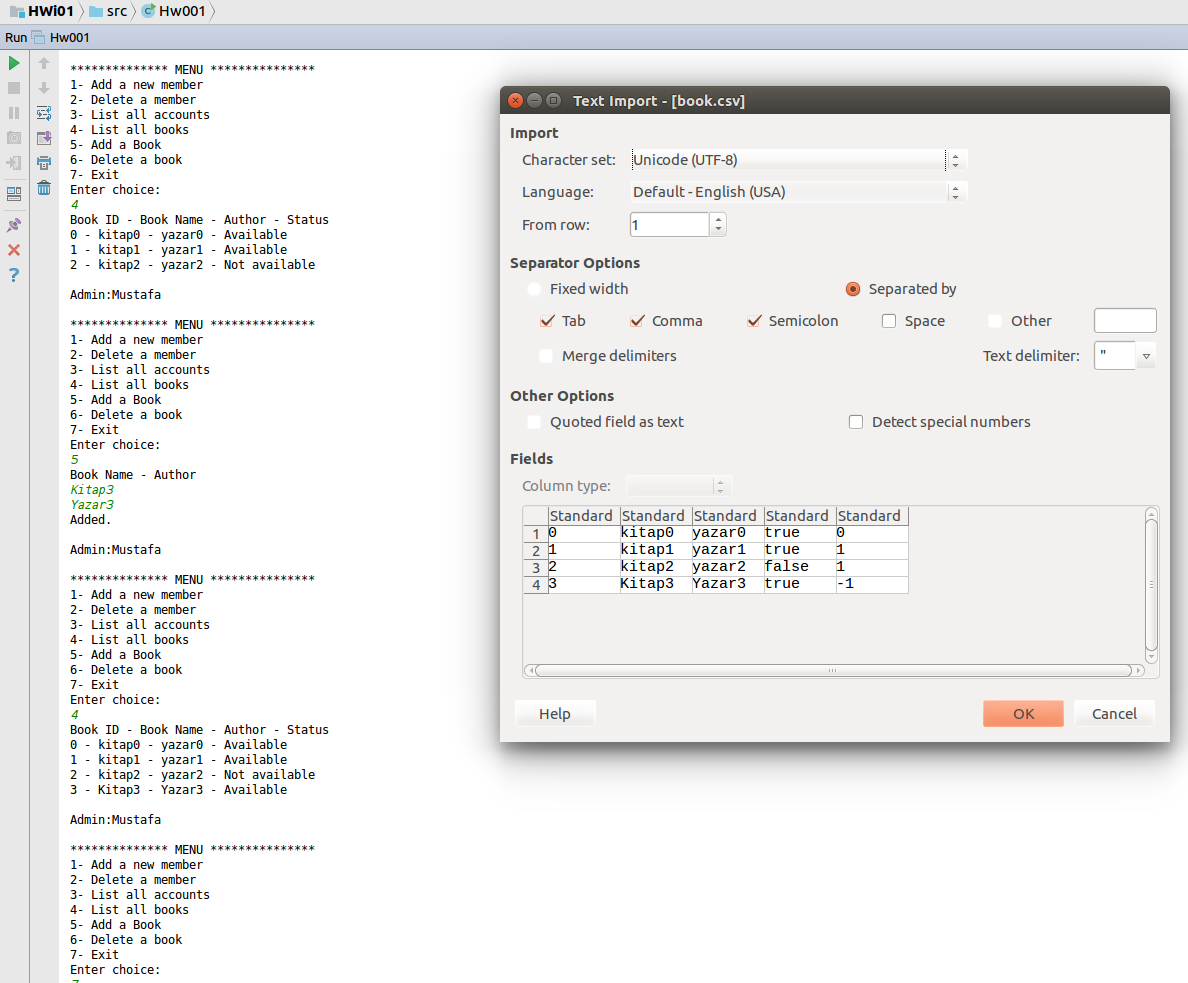
6.2-Kullanıcı silme başarılı.



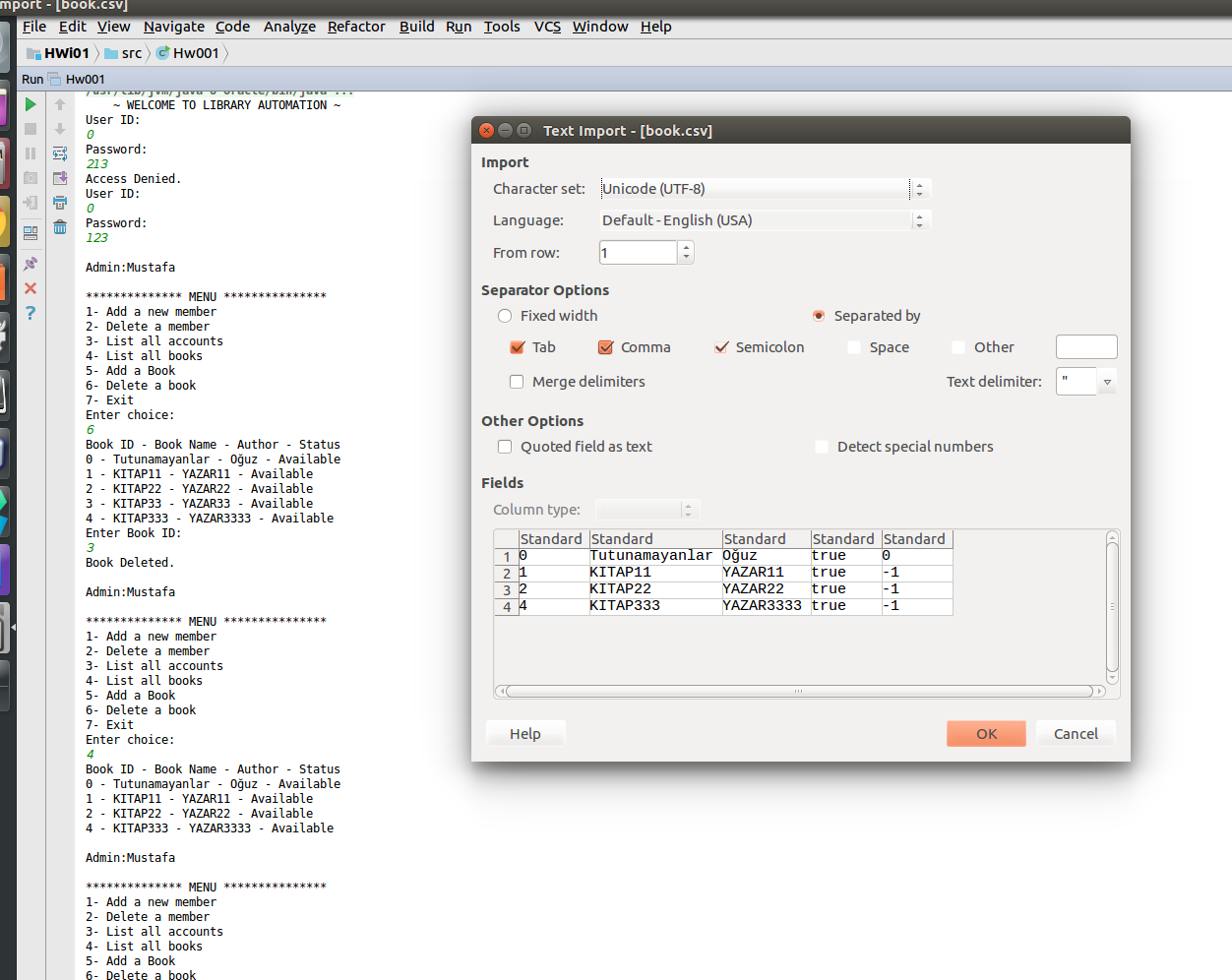
6.3-Olmayan kullanıcıyı silememe başarılı.



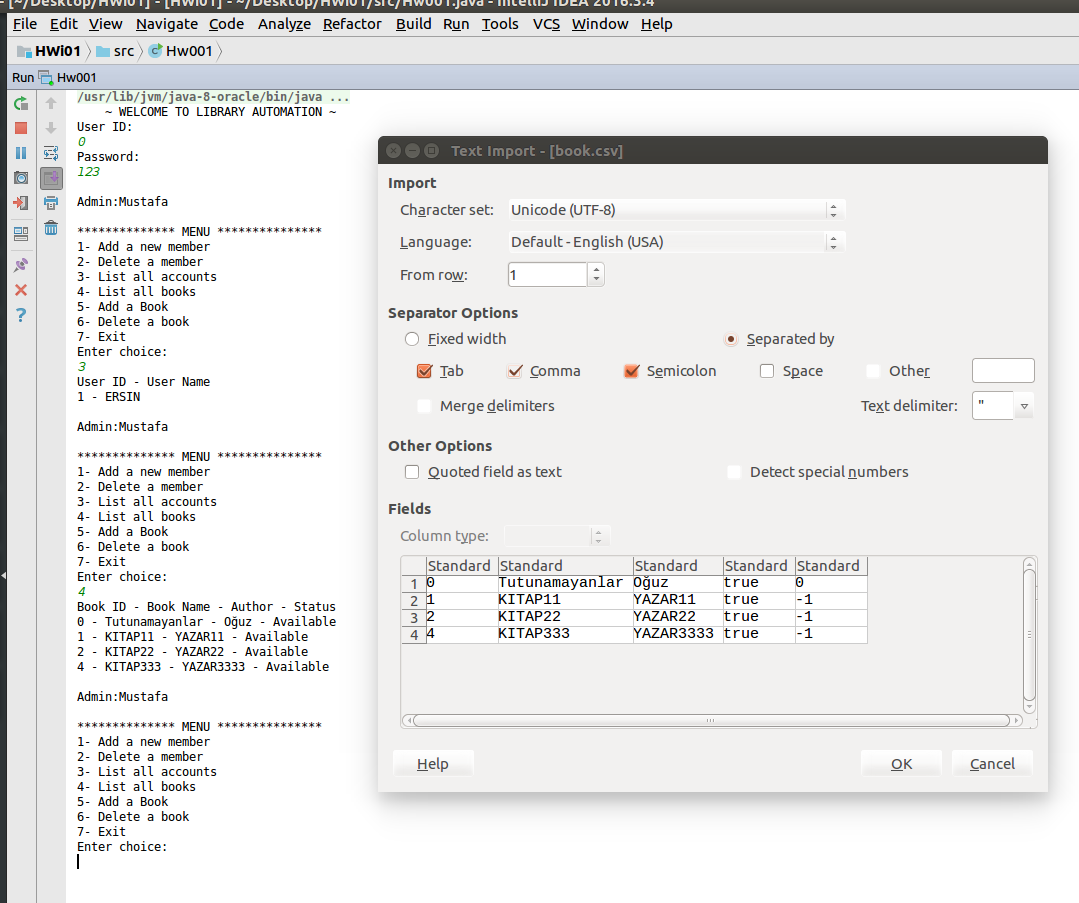
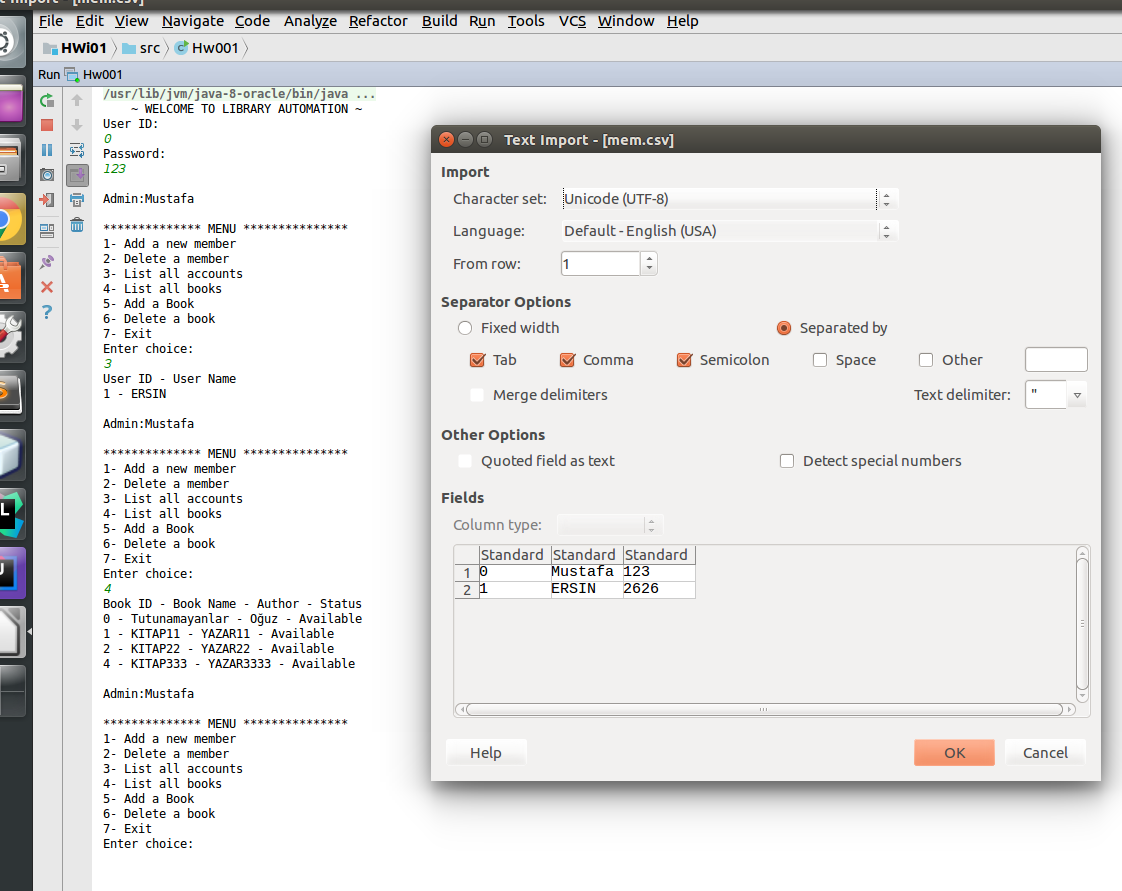
6.4-Sadece veritabanındaki kitaplar listelenmektedir.



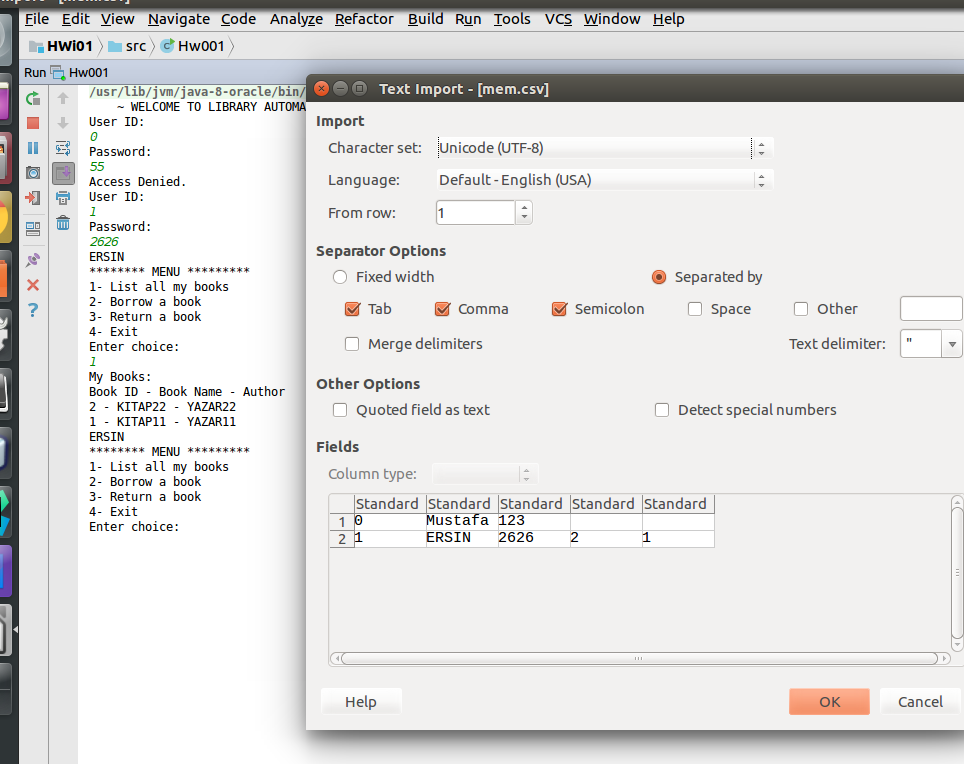
6.5-Kitap ekleme başarılı.



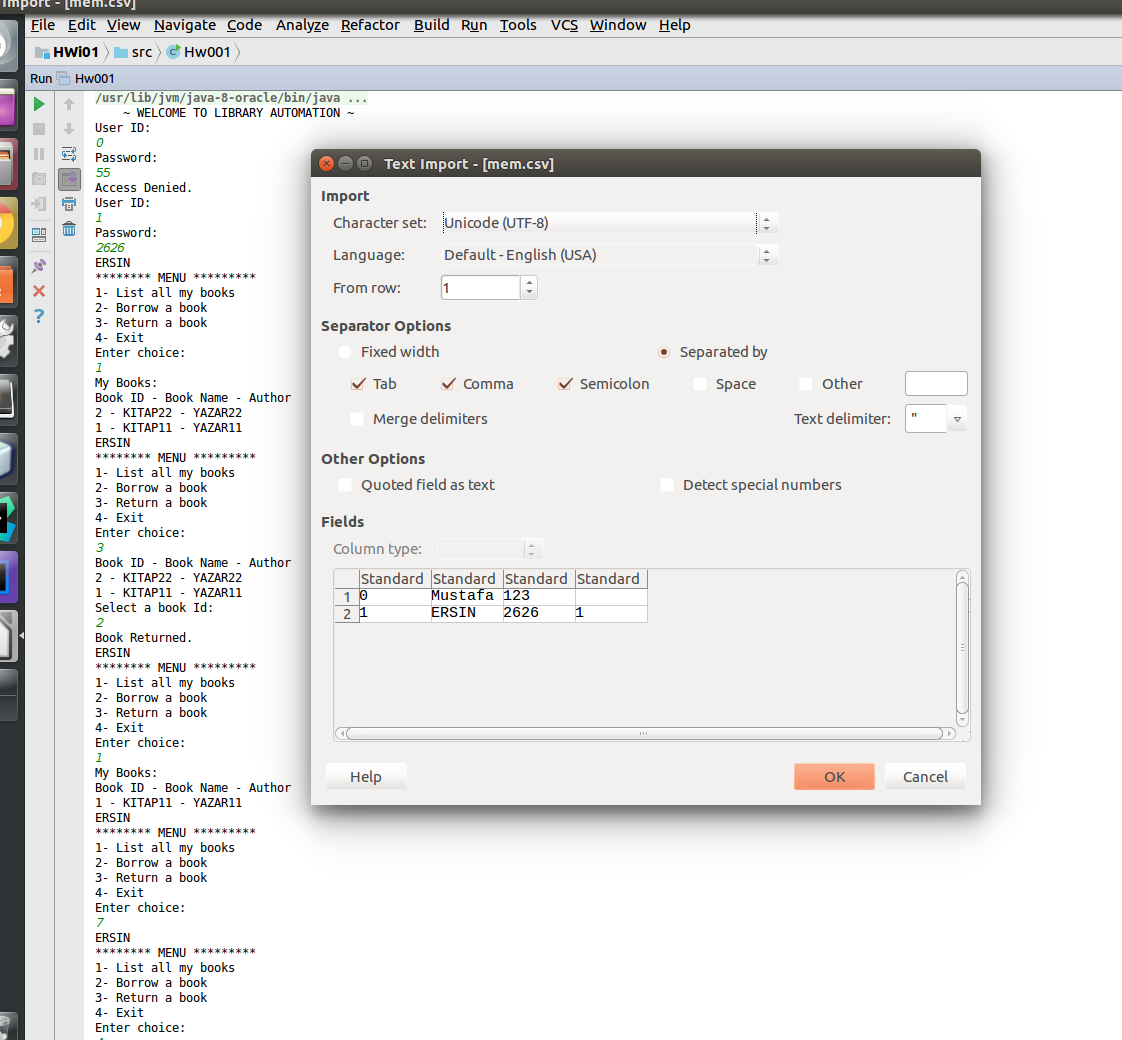
6.6-Kitap silme başarılı.



6.7-Tüm kitapları ve kullanıcıları listeleme başarılı.



6.8-Sadece kullanıcının sahip olduğu kitapları sergileme başarılı.



6.9-Ödünç alınan kitabı geri verme başarılı.

**7. Running and Results**

Bir yönetici vardır. (User ID:0 Password:123) Program çalıştırıldığından doğrulama gerçekleşir doğrulama ID ye göre gerçekleşir ve doğrulamadan sonra yönetici veya kullanıcı olarak devam eder. Ve program kullanıcılara id numaralarını otomatik atamaktadır. Eğer program yöneticiyi doğrularsa karşısına yönetici menüsü gelir. Ve yönetici; kitap ve kullanıcı ekleme, silme yetkilerine sahiptir aynı zamanda kütüphanedeki tüm kitapları ve tüm kullanıcıları görebilmektedir. Eğer program kullanıcıyı doğrularsa kullanıcı menüsü gelir. Ve kullanıcı; kitap ödünç alabilir ve geri verebilir aynı zamandan sadece ödünç aldığı tüm kitapları görebilir.

[https://github.com/mstfbngl/](https://github.com/mstfbngl/HW01_141044077)