

Gebze Technical University
Computer Engineering

CSE 222
2017 Spring

HOMEWORK 1 REPORT

OSMAN AKKUŞ
151044055

Course Assistant:
NurBanu Albayrak
Şeyma Yücer

1. System Requirements

Yazılacak olan Library Management System de 2 tane kullanıcı erişim sağlayabiliyor. Bunlardan biri Personel(Staff) , birisi de Kullanıcı(User) olarak giriş yapabilecek.

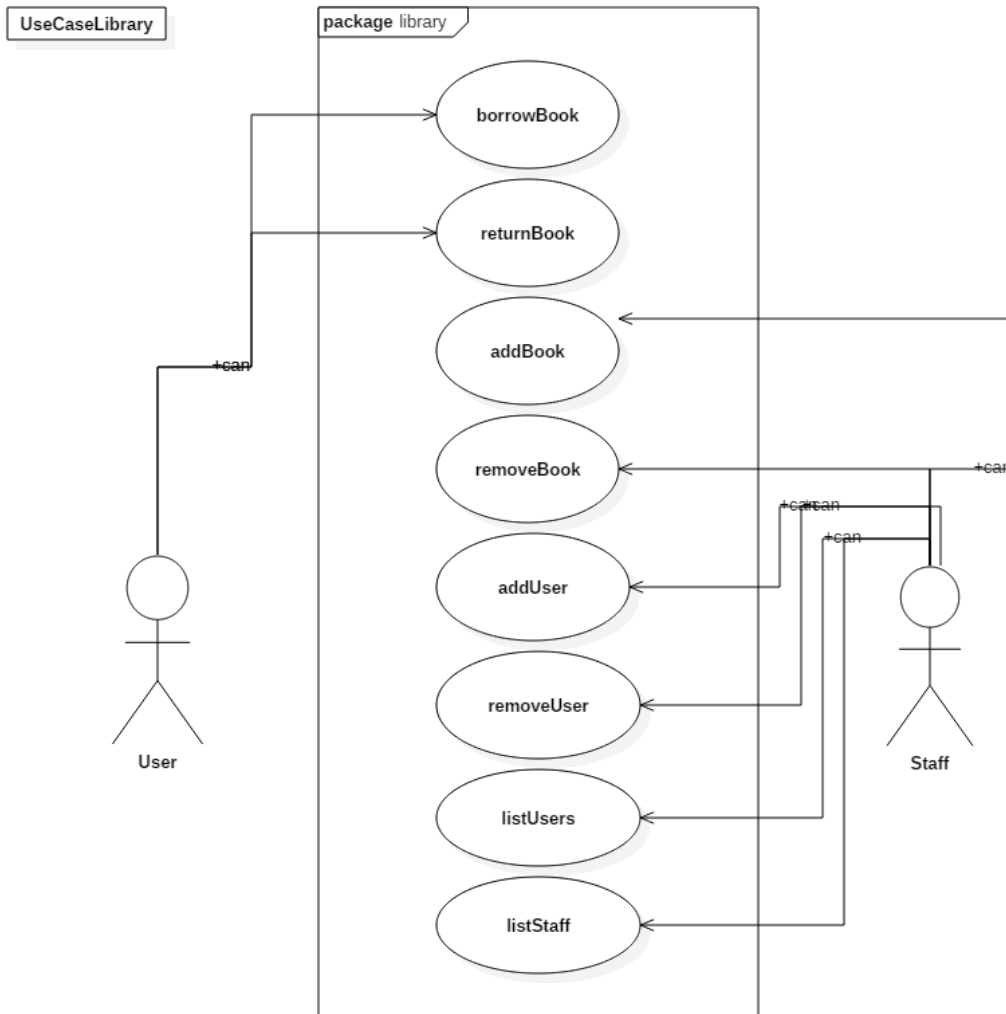
Kullanıcının yapabileceği işlemler sırasıyla:

- 1- Kitap almak
- 2- Aldığı kitabı iade etmek

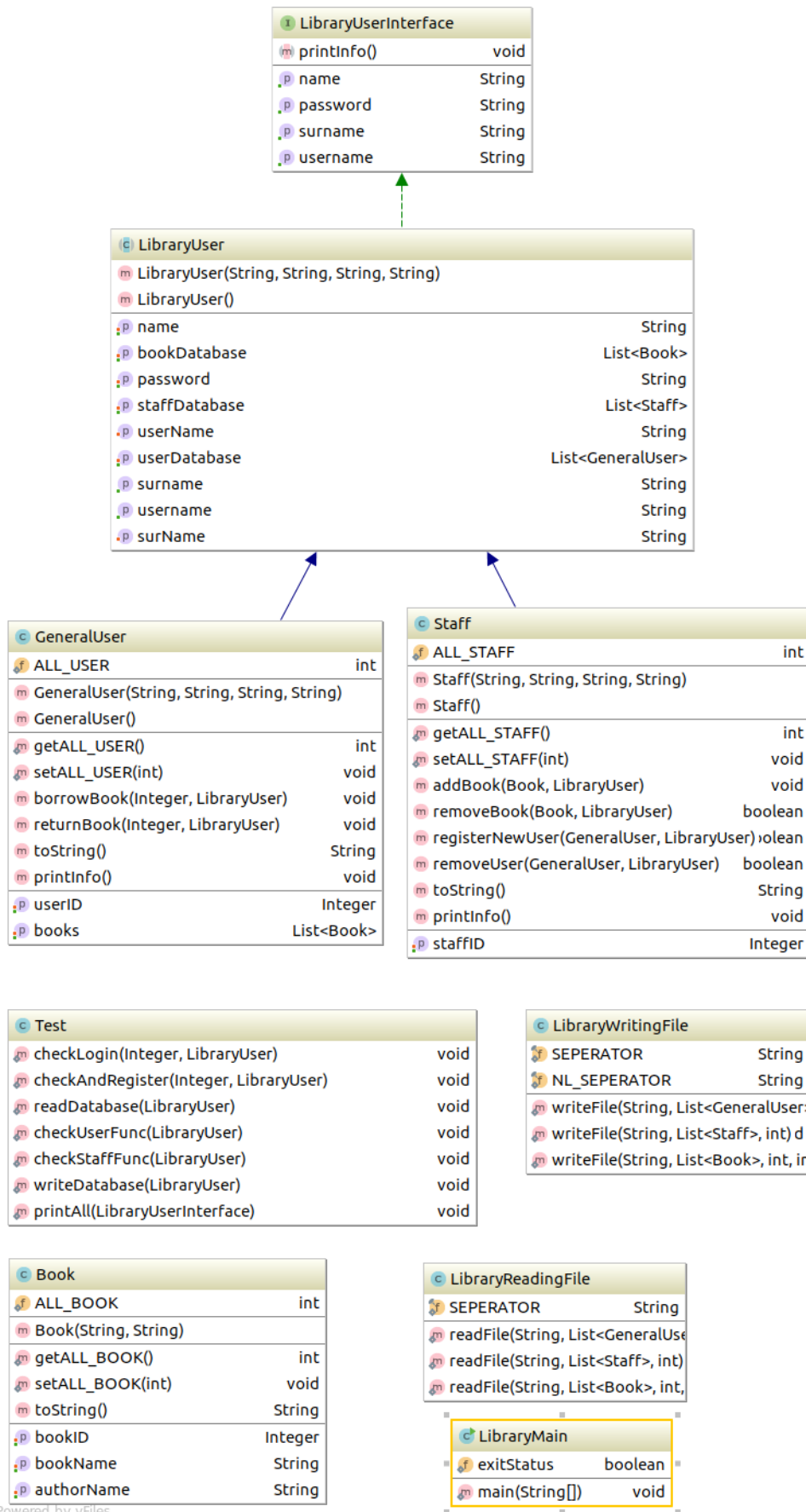
Personelin yapabileceği işlemler sırasıyla:

- 1- Kitap eklemek
- 2- Kitap çıkarmak
- 3- Yeni Kullanıcı Eklemek
- 4- Kullanıcı Silmek
- 5- Bütün Kullanıcıları Listelemek
- 6- Bütün Personeli Listelemek

2. Use Case Diagrams



3. Class Diagrams



4. Other Diagrams

Add other diagrams.

5. Problem Solutions Approach

Probleme uygun çözümleri düşünmeye başladığım ilk zamanlarda bunu daha verimli bir şekilde nasıl yazabilirim hedefiyle uğraşmaya başladım daha sonra aklımdan tasarladığım interface i ve devamında yazacağım class ları tek tek implement etmeye başladım.

1- Interface tasarımı

2- Interface i implement edecek classlar

Bu iki öncüle göre çalışmaya başladıktan sonra bir <interface>Library bir <abstract class> ve bunlardan türeyen Staff ve GeneralUser olmak üzere iki class yazdım. Daha sonra yardımcı olmaları bakımından reading ve writing işlemleri yapıcak iki class daha yazdım.

Ayrıca her kitabın bilgisini tutmak için Book class ını yazdım. Yazdığım abstract class da ise Staff , GeneralUser ve Book için toplamda 3 database oluşturdum.Bu databaselerde yapılan her güncellemeler .csv uzantılı dosyalara kaydedildi.

Bunların hepsini Test adlı java class ında static fonksiyonlarda döngüler içinde test ettim.

6. Test Cases

Daha ayrıntılı anlatmak için tek tek yazıcam

Oncelikle şöyle bir ekran geliyor.

Welcome to the Library Management System

[1]Login

[2]Register

[-1]Exit

Buradan kaydolmak için input olarak 1 girince :

Login System as:

[1] User

[2] Staff

[-1]Return Menu

ekranı çıkıyor buradan input olarak 1 girince User olarak giriş yaparsınız ve User a özel işlemler yapabilirsiniz.

Input olarak 2 girince Staff olarak giriş yapıp Staff a özel işlemler yapabilirsiniz.

Daha detaylı olarak Running and Results daki screenshotlardan öğrenebilirsiniz.

7. Running and Results

Bu kısımda eklediğim screenshotlarda programın nasıl çalıştığını ve ne sonuçlar çıkardığını daha rahat bulabilirsiniz.

otto@otto-X555LB: ~/cse222/HWs/HW1

Tr (54%) Pzt Şub 27 20:59 Osman

```
otto@otto-X555LB:~/cse222/HWs/HW1$ java LibraryMain
Welcome to the Library Managament System
```

```
[1]Login
[2]Register
[-1]Exit
2
Register System as:
[1] Staff
[-1]Return Menu
1
```

```
Enter the information below
Name: osman
Surname: akkus
Username: otto
Password: 1234
Welcome to the Library Managament System
```

```
[1]Login
[2]Register
[-1]Exit
1
Login System as:
[1] User
[2] Staff
[-1]Return Menu
2
```

```
Enter Your
UserName: otto
Password: 1234
```

Operations You Can Do

```
[1] Add A New Book
[2] Remove A Book
[3] Add A New User
[4] Remove A User
[5] List All User
[6] List All Staff
[-1] Return Home
```

1



otto@otto-X555LB: ~/cse222/HWs/HW1

Tr (53%) Pzt Şub 27 21:02 Osman

```
[2] Remove A Book
[3] Add A New User
[4] Remove A User
[5] List All User
[6] List All Staff
[-1] Return Home
1
```

```
Enter the information below
BookName: problem solving and design in C
Author: koffman
```

Operations You Can Do

```
[1] Add A New Book
[2] Remove A Book
[3] Add A New User
[4] Remove A User
[5] List All User
[6] List All Staff
[-1] Return Home
3
```

3

```
Enter the information below
Name: enes
Surname: gonultas
Username: nsgonultas
Password: 123456
```

Operations You Can Do

```
[1] Add A New Book
[2] Remove A Book
[3] Add A New User
[4] Remove A User
[5] List All User
[6] List All Staff
[-1] Return Home
3
```

3

```
Enter the information below
Name:
```



otto@otto-X555LB: ~/cse222/HWs/HW1

Tr (53%) Pzt Şub 27 21:03 Osman

```
[5] List All User
[6] List All Staff
[-1] Return Home
3
```

Enter the information below

Name: adem
Surname: kaya
Username: aKaya
Password: 1111

Operations You Can Do

```
[1] Add A New Book
[2] Remove A Book
[3] Add A New User
[4] Remove A User
[5] List All User
[6] List All Staff
[-1] Return Home
2
```

Choose the book by its ID to remove

1,unix system programming,Robins & Robins

1,problem solving and design in C,koffman

1

Operations You Can Do

```
[1] Add A New Book
[2] Remove A Book
[3] Add A New User
[4] Remove A User
[5] List All User
[6] List All Staff
[-1] Return Home
1
```

Enter the information below

BookName:

otto@otto-X555LB: ~/cse222/HWs/HW1

Tr (53%) Pzt Şub 27 21:04 Osman

```
[5] List All User
[6] List All Staff
[-1] Return Home
4
```

Choose the user by its ID to remove

1,enes,gonultas,nsgonultas,123456

1,adem,kaya,aKaya,1111

1

Operations You Can Do

```
[1] Add A New Book
[2] Remove A Book
[3] Add A New User
[4] Remove A User
[5] List All User
[6] List All Staff
[-1] Return Home
6
```

ID: 1

Name: osman
Surname: akkus

Operations You Can Do

```
[1] Add A New Book
[2] Remove A Book
[3] Add A New User
[4] Remove A User
[5] List All User
[6] List All Staff
[-1] Return Home
-1
```

Welcome to the Library Managment System

```
[1]Login
[2]Register
[-1]Exit
```

otto@otto-X555LB: ~/cse222/HWs/HW1

Tr (53%) Pzt Şub 27 21:04 Osman

Tr (53%) Pzt Şub 27 21:05 Osman

Operations You Can

[1] Borrow A Book



PART 2

Abstract Data Type Generating in C

Bildiğimiz gibi Object Oriented dillerden farklı olarak C daha çok functional bir dildir ve Objeye dayalı bir programla yapmak oldukça zor olsada imkansız değildir.

Ödevin bu kısmında bizden C de ADT lerin nasıl oluşturulabileceği hakkındaydı. Araştırmalarım göre bu C deki struct yapısı ile function pointer mantığı ile mümkün.

<http://inst.eecs.berkeley.edu/~selfpace/studyguide/9C.sg/Output/ADTs.in.C.html>

sitesinden aldığım bir örnekte bunu anlamak daha da mümkün.

stack.h

```
typedef struct StackStructType *StackType;

/* Return a pointer to an empty stack. */
extern StackType InitStack ( );
/* Push value onto the stack, returning success flag. */
extern boolean Push (int k);
/* Pop value from the stack, returning success flag. */
extern boolean Pop ( );
/* Print the elements of the stack. */
extern PrintStack (StackType stack);
```

stack.c

```
#include "stack.h"
#define STACKSIZE 5
struct StackStructType {
    int stackItems [STACKSIZE]; /* an array of items */
    int nItems; /* plus how many there are */
};
typedef struct StackStructType *StackType;

/*
**      Return a pointer to an empty stack.
*/
StackType InitStack ( ) {
    char *calloc( );
    StackType stack;
    stack = (StackType) calloc (1, sizeof (struct StackStructType));
    stack->nItems = 0;
    return (stack);
}
...
```