

**Gebze Technical University  
Computer Engineering**

**CSE 222 - 2018 Spring**

**HOMEWORK 1 REPORT**

**TARIK KILIÇ  
151044010**

Course Assistant: Fatmanur Esirci

# 1 INTRODUCTION

## 1.1 Problem Definition

Günümüzde artık her şeyin dijitalleşmesinden dolayı çoğu kurum işleri online olarak hizmet vermeye başlamıştır. Otellerde de rezervasyon, kayıt vs. işleri yapmak ve bunların hepsini kağıt olarak saklamak işleri zorlaştırır. Otel sahibi açısından kayıtları bulmak ve bunlar üzerinden değişiklik yapmak hem zaman kaybı hem de uğraş gerektirici bir iştir. Müşteri açısından baktığımız zaman rezervasyonu kendisi hızlı bir şekilde yapmak varken otel ile iletişim kurmak zorundadır. Bu sistem ile hem müşteri kimseyle iletişim kurmadan rezervini yaptırabilir veya iptal edebilir. Otel ise müşterilerin kayıt işlemlerini bilgisayarda tutarak hızlı bir şekilde ulaşabilir ve orada kayıtlı tutabilir.

## 1.2 System Requirements

### HOTEL SINIFI:

Bu classın constructorunda guest ve Receptionist objeleri oluşturulur. Değiştirmek veya eklenmek istenirse constructurdan eklenebilir. AddUser metodu kullanılarak eklemek gerekmektedir.

Filename static final olarak tutulmaktadır(main sınıfında).

Test modunda açmak istiyorsanız sağ üst run butonunun yanındaki listeneden testi seçiniz, main modunda açmak istiyorsanız maini seçiniz.

`boolean bookRoom(String filename,Room room,Guest guest)` : `bookRoom` metodunun bir dosya adına,bir Room objesine ve bir guest objesine ihtiyacı vardır. Girilecek oda boş olmalı ve guest in başka oda için işlem yapmamış olması gerekmektedir.

`boolean cancelRezerv(String filename,Guest guest)`: Bu metodun bir dosya adına ve bir guest objesine ihtiyacı vardır. İşlemin gerçekleşebilmesi için önceden oda rezerv i yapması gerekmektedir.

`boolean checkIn(String filename,Room room,Guest guest)`: Bu metodun bir dosya adına,bir Room objesine ve bir guest objesine ihtiyacı vardır.Girilecek oda boş olmalı veya guest in önceden girilecek oda için rezerv yapması beklenmektedir.

`boolean checkOut(String filename,Guest guest)`: Bu metodun bir dosya adına ve bir guest objesine ihtiyacı vardır. İşlemin gerçekleşebilmesi için önceden check-in olması gerekmektedir.

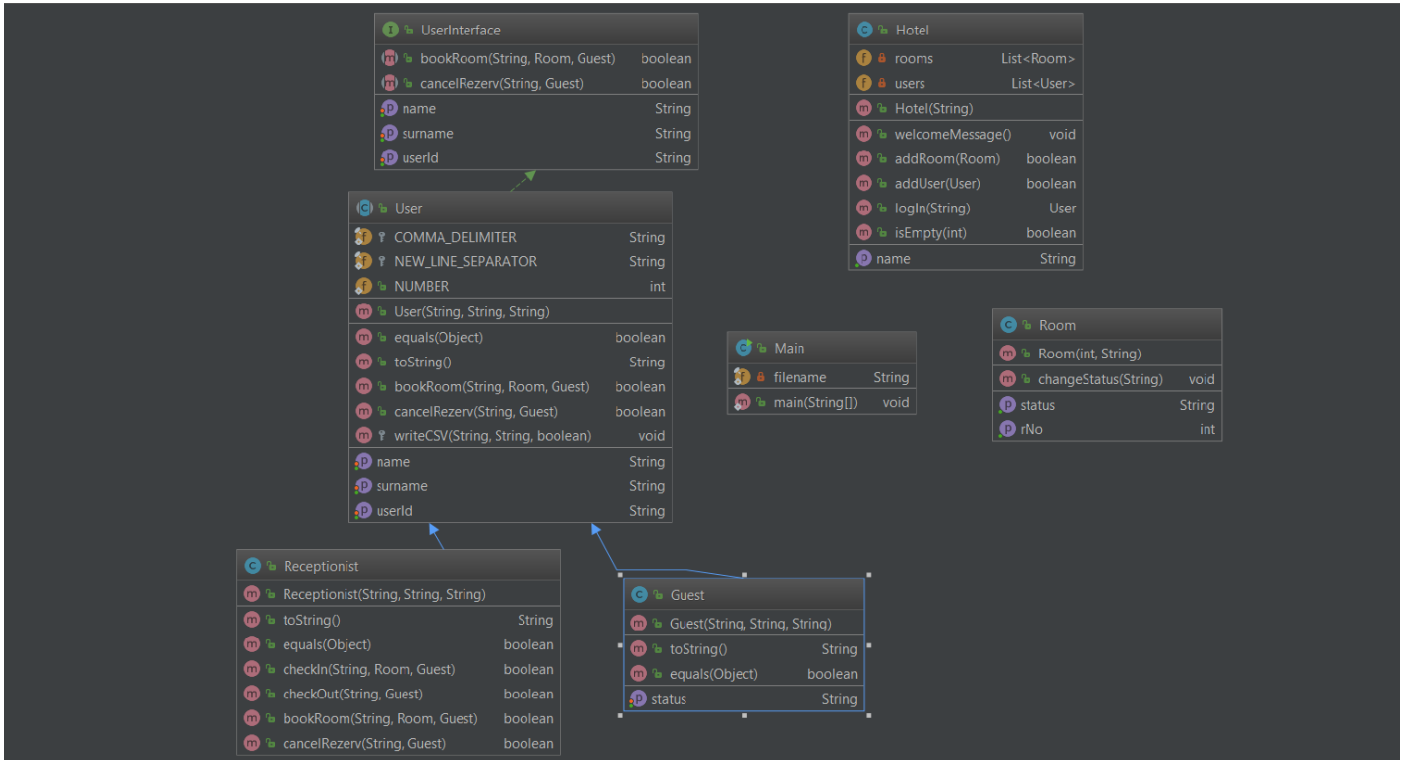
*User login(String id):* Kullanıcın giriř yapabilmesi için önceden sistemle kayıtlı olması gerekmektedir(Hotel sınıfında açıklanmıştır) ve bu metod kayıtlı kullanıcının userId ihtiyaç duyar. Sonucunda bir user objesi return eder. Receptionist veya guest referansına atamak gerekir.

*boolean isEmpty(int no):* Bu metod oda numarasına(no) ihtiyaç duyar.Sonucunda ekrana boş olup olmadığını bastırır.

*Void writeCSV(String line,String filename,boolean append):* Bu metod dosyaya yazılacak kullanıcı bilgilerini içeren bir string dizisine ve bunları kaydedeceği dosya adına ihtiyaç duyar. Ayrıca duruma göre overwrite edip etmeyeceğini boolean append olarak belirtmek gerekir.

## 2 METHOD

### 2.1 Class Diagrams



User(Abstract Class) classı UserInterface i implement ederek oluşturulur. UserInterface de bulunan bookRoom ve cancelRezerv metodları bu class da implement edilir .Dosyaya yazmada yardımcı olsun diye writeCSV fonksiyonu yazılmıştır.Guest classı bu fonksiyonları aynen kullanmıştır.Receptionist classı ise bu fonksiyonları override ederek yetkiler çoğaltılır(başka bir

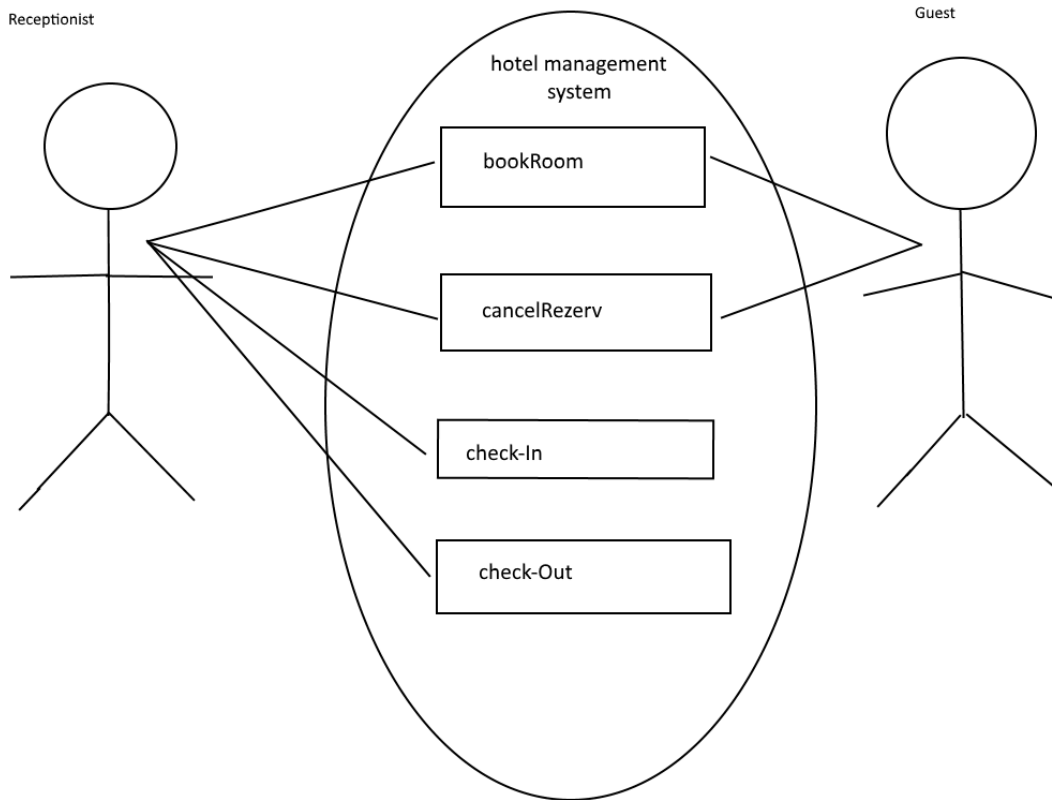
kişiy e rezerv yapmak). Bu classlar arasında is-a relation vardır.

Hotel ile room classlarında arasında has-a relation vardır. Hotel içinde room bilgilerinin arraylistini tutar ve ekleme yapar.

Programın mantığı bir hotel objesi oluşturulur. Oluşturulduğu anda 4 tane otomatik kullanıcı kayıt edilir. Ardından main de room objeleri oluşturularak addRoom metodu ile hotel classın içindeki listeye atılır.

login metodu kullanarak id parametresi girilir ve metod user arraylist in içinde kullanıcıyı bulup return eder. Return edilen User mainde oluşturduğum user referansına atanır. Eğer kullanıcı Guest ise guest cast, Receptionist ise receptionist cast yapıp referanslar initilaze edilir.

## 2.2 Use Case Diagrams



Receptionist rezerv, cancelRezerv, kayıt ve çıkış yapabilir. Guest ise rezervasyon ve cancelRezerv yapabilir.

## 2.3 Other Diagrams (optional)

Add other diagrams if required.

## 2.4 Problem Solution Approach

Müşteri ve resepsiyonist bazı ortak özelliklere sahip olduğundan(bookRoom(),cancelRezerv(),name,surname,id) bunların ortak bir yapıdan gelirse daha yeniden yazılabilir bir kod olacağından dolayı bu iki yapı UserInterface ve ondan türeyen User(abstract class)'dan türemiştir. bookRoom metodu abstract class da implement edilmiş olup Receptionist classında override edilmiştir.

Bu kullanıcıların login yapabilmeleri için önceden kayıt olmaları gerekmektedir. Bu programda Hotel classının constructorunda otomatik olarak 4 tane guest arrayList de tutulmaktadır. Login metodunda arrayListe bakar giriş yaptığı doğrulanır.

Odalar(Room) ise main classında oluşturulmuştur isteğe göre sayısı değiştirilebilir.Hotel classının içinde Room objesi tutan arrayList vardır.

Dört ana metod ile(bookRoom,cancelRoom,checkIn,checkOut) dosyaya yazılıp dosyadan okunarak işlemler yapılır.

NOT!:: Programın geçerliliği bitmesiyle sonra erer, tekrardan başlamasıyla sıfırdan dosyaya yazılır.

OOP design kullanılmıştır.Metodlar override edilmiştir.Inheritance kullanılmıştır.Interface ve abstract class oluşturulmuştur.Polymorphism vardır(bookRoom,cancelRezerv).Javadoc dosyanın içindedir.Error Handling yapılmıştır.

Checkout ve CancelRezerv metodlarını yazarken(Dosyaya yazma işlemleri) Stackoverflowdan faydalandım.

## 3 RESULT

### 3.1 Test Cases

#### **Main Test:**

- Hotel objesi oluşturuldu
- Guest ve Receptionist objeleri oluşturuldu.
- Login yapıldı.
- Room objeleri oluşturuldu.
- Oda durumu sorulup bookRoom,cancelRezerv,checkin,checkout metodları denendi.

#### **JUnit Test:**

- JUnit5 testleri yapıldı
- Tüm testlerden geçti.
- Yapılan metodlar: Checkin,Checkout,bookRoom,CancelRezerv,login.
- Farklı şekillerde test edildi.True ya da false olma durumları.

## 3.2 Running Results

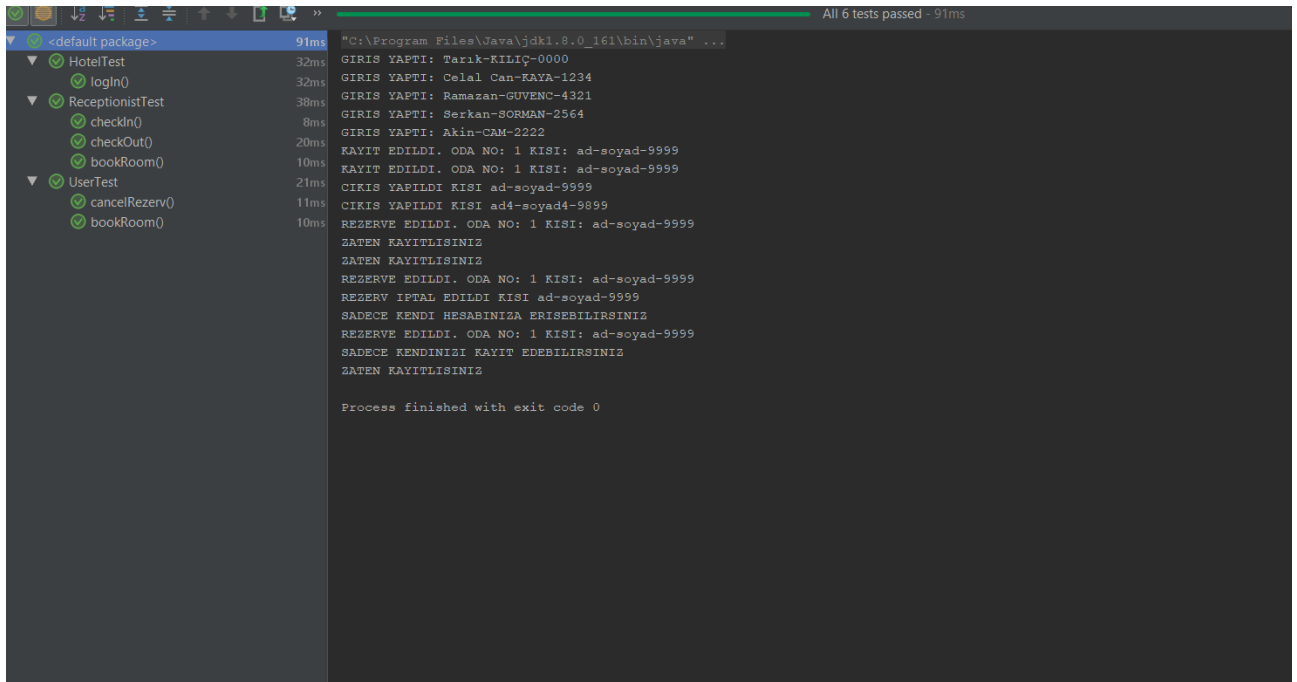
mainResult:

```
"C:\Program Files\Java\jdk1.8.0_161\bin\java" -xxx
-----OTEL OLUSTURULDU VE ISIM VERILDI-----
-----HOSGELDIN MESAJI-----
Welcome to GTU HOTEL

-----KULLANACAGIM REFERANSLAR OLUSTURULDU-----
-----LOGIN ISLEMLERI YAPILDI-----
GIRIS YAPTI: Tarik-KILIÇ-0000
RECEPTIONIST LOGED
GIRIS YAPTI: Celal Can-KAYA-1234
GUEST LOGED
GIRIS YAPTI: Ramazan-GUVENC-4321
GUEST LOGED
GIRIS YAPTI: Serkan-SORMAN-2564
GUEST LOGED
GIRIS YAPTI: Akin-CAM-2222
GUEST LOGED
-----ODALAR OLUSTURULDU-----
-----ODA DURUMU SORULDU-----
----- 1. MUSTERI REZERV YAPTIRDI(KENDISI)-----
REZERVE EDILDI. ODA NO: 5 KISI: Celal Can-KAYA-1234
-----ODA DURUMU SORULDU-----
Room 1 is EMPTY
----- 2. MUSTERI KAYIT YAPTIRDI-----
KAYIT EDILDI. ODA NO: 1 KISI: Ramazan-GUVENC-4321
-----ODA DURUMU SORULDU-----
----- 3. MUSTERI REZERV YAPTIRDI(RECEP TARAFINDAN)-----
REZERVE EDILDI. ODA NO: 4 KISI: Serkan-SORMAN-2564
-----ODA DURUMU SORULDU-----
-----ODA DURUMU TEKRAR SORULDU-----
----- 4. MUSTERI REZERV YAPTIRDI(KENDISI)-----
REZERVE EDILDI. ODA NO: 7 KISI: Akin-CAM-2222
----- 1. MUSTERI REZERVLI YERINE KAYIT YAPTIRDI-----
KAYIT EDILDI. ODA NO: 5 KISI: Celal Can-KAYA-1234
----- 2. MUSTERI CIKIS YAPTI-----
CIKIS YAPILDI KISI Ramazan-GUVENC-4321
----- 3. MUSTERI REZERV IPTAL ETTIRDI(KENDISI TARAFINDAN)-----
REZERV IPTAL EDILDI KISI Serkan-SORMAN-2564
----- 4. MUSTERI REZERV IPTAL ETTIRDI(RECEP TARAFINDAN)-----
REZERV IPTAL EDILDI KISI Akin-CAM-2222
```

- Main test hatasız şekilde çalıştı.
- Test Cases kısmında nasıl çalıştığı açıklandı.

testResult:



- JUnit test hatasız şekilde çalıştı.
- Test Cases kısmında nasıl çalıştığı açıklandı.

- Main titles -> 16pt , 2 line break
- Subtitles -> 14pt, 1.5 line break
- Paragraph -> 12pt, 1.5 line break