

**Gebze Technical University
Computer Engineering**

CSE 222 - 2018 Spring

HOMEWORK 1 REPORT

**CELAL CAN KAYA
161044014**

Course Assistant: Fatma Nur Esirci

1 INTRODUCTION

1.1 Problem Definition

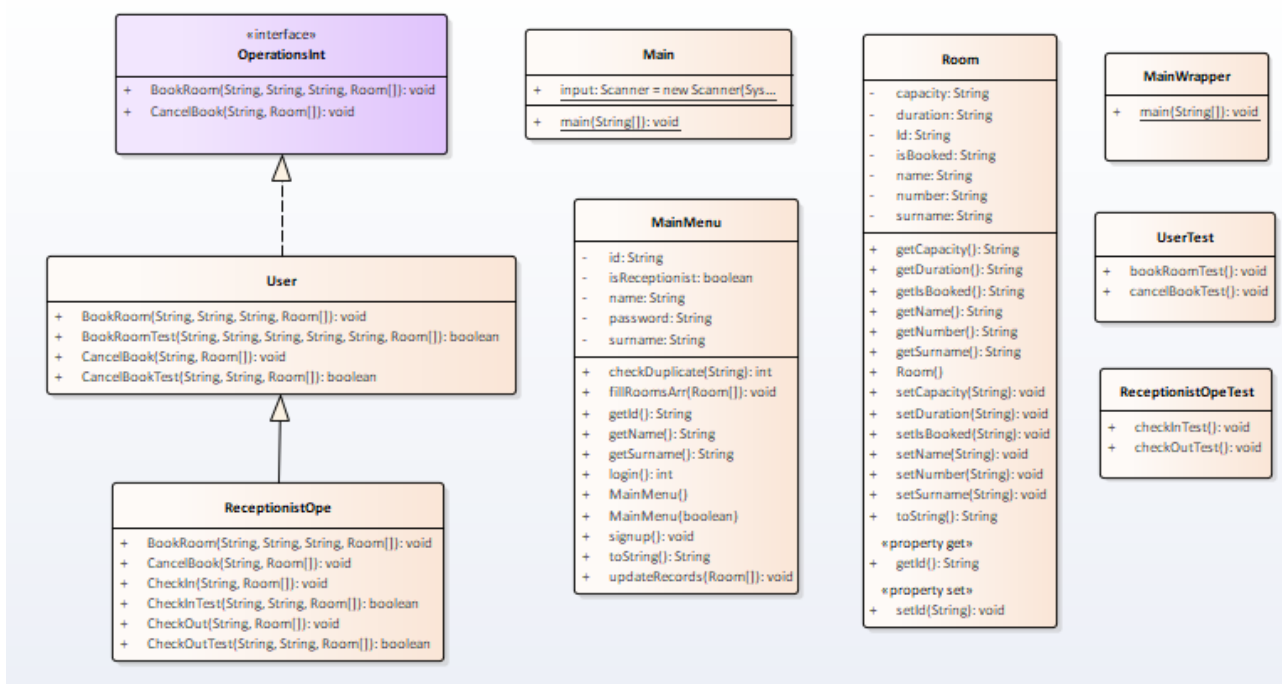
Günümüzde her şeyin dijitalleşmesi ve veri sayısının sürekli artması nedeniyle bir işi manuel olarak yapmak hem zor hem de hata yapmaya gayet müsait olacaktır. Kullanıcılar için zahmetsiz bir şekilde otel odası rezerve edebileceği ve rezerve ettiği odayı iptal edebileceği bir sistem gerekmektedir. Aynı zamanda resepsiyonistler için kullanıcının yaptığı işlere ek olarak kullanıcıları check-in ve check-out edebilecekleri bir sistem gerekmektedir. Bu nedenle kullanıcıların ve resepsiyonistlerin ortak kullanabileceği basit bir otel yönetim sistemi gerekmektedir.

1.2 System Requirements

Projemin çalışması için oda kayıtlarını tuttuğum “rooms.csv” dosyası, kullanıcıların ve resepsiyonistlerin giriş bilgilerini tuttuğum “users.csv” ve “receptionist.csv” dosyası gerekmektedir. Kullanıcıdan aldığımız girdileri otomatik olarak dosyadan kullanması için 3 tane farklı senaryo dosyasını (“1.txt”, “2.txt”, “3.txt”) proje dosyamın içine ekledim. Bu senaryoları denemek için MainWrapper class’ında dosyayı istediğiniz dosya ile değiştirmek yeterlidir. Ayrıca csv dosyalarında manuel olarak oynama yapılmamasını tavsiye ediyorum. Unit Test yaparken csv dosyasına kayıtlı olarak yolladığım kayıtlarla karşılaştırıldığı için unit testlerde bozukluğa sebep verebilir. Eğer MainWrapper yerine Main class’ını çalıştırıp inputları elle girerseniz program sonunda csv dosyasını ilk halinde bırakmanız (rezervasyon yaptıysanız iptal etmek, checkin yapılsa check out etmek gibi) unit testler ve senaryo dosyalarım için yararlı olacaktır.

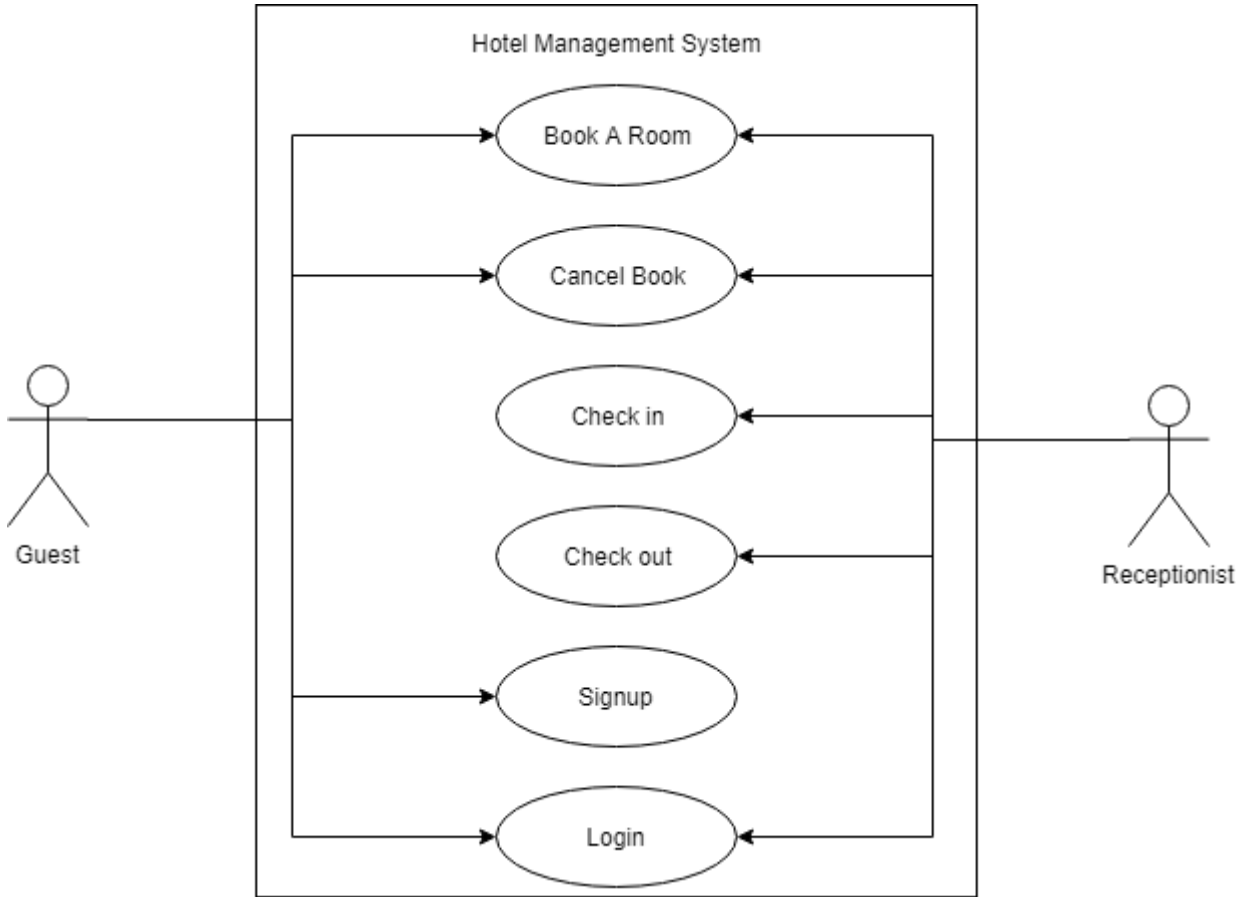
2 METHOD

2.1 Class Diagrams



OperationsInt adlı class'ım interfacedir.User ve resepsiyonistte BookRoom ve CancelRoom operasyonlarını yapacağı için User class'ında OperationsInt'i implement ettim.ReceptionistOpe class'ında ise User class'ımdan extend ettim ve BookRoom, CancelBook metodlarını override ettim. Main class'ımda Room tipinde bir array oluşturarak csv dosyasından çektiğim bilgileri Room class arrayine aktardım.MainMenu class'ımdaki login ve signup metodlarından aldığım verilerle MainMenu classındaki kullanıcıya ait verilere atadım.

2.2 Use Case Diagrams



Sistem, misafir ve resepsiyonist olmak üzere 2 kullanıcı tipi tarafından kullanılmaktadır.Misafir kullanıcımız sisteme kayıt olma, sisteme giriş yapma, oda rezervasyonu yapma ve yaptığı rezervasyonu iptal etme işlemlerini kullanabilir.Check-in ve check out işlemlerini kullanamaz.Çünkü odaya giriş ve ücreti ödeyip odadan çıkış yapma işlemleri otel yönetimi gözetiminde olmalıdır.Resepsiyonist ise sadece kayıt olma işlemini kullanamaz.Çünkü sistem yöneticileri(resepsiyonistler) zaten sisteme kayıtlıdır.Sistemin güvenliği açısından sistem üzerinden “receptionist.csv” dosyasına yeni kullanıcı kaydı oluşturma kapalıdır.Manuel olarak sistem kullanıcıya sunulmadan önce resepsiyonist kayıtları yapılmıştır.

2.3 Other Diagrams (optional)

2.4 Problem Solution Approach

Sistemde login istendiği için ekstra olarak kullanıcıların kayıt olması için kayıt olma sistemi ekledim. Kayıt sisteminde alınan veriler “users.csv”ye kaydediliyor. Giriş yapma sisteminde ise kullanıcının ya da resepsiyonistin girdiği veriler “users.csv” ya da “receptionist.csv” dosyasındakilerle uyuşuyor ise sisteme giriş yapılıyor. Bu sayede giriş yaptığımız kullanıcıya göre yapabileceğimiz işlemler karşımıza çıkacaktır. Kullanıcı olarak giriş yaptıysak rezervasyon yapma ve rezervasyon iptali menüsü karşımıza çıkıyor. Eğer rezervasyon yapmak istersek “rooms.csv” dosyası kontrol edilerek kullanıcının istediği cinsten bir oda varsa rezervasyon işlemi tamamlanıyor. Rezervasyon iptalinde ise kullanıcının id’sine kayıtlı olan oda var ise onlar karşısına çıkıyor ve istediği odanın rezervasyonunu iptal ediyor. Resepsiyonist olarak giriş yaptığımızda misafir kullanıcının yaptığı işlemlerin aynısı vardır fakat o metodları override ederek kimin adına ve idsine rezervasyon/rezervasyon iptali yapılmak isteniyorsa “users.csv”den kayıtlı olup olmadığı kontrol ediliyor. Eğer “users.csv” de gerçekten böyle bir kullanıcı var ise extend ettiğim classtaki metodlar çağırılıyor. Check-in yapma sisteminde ise aynı şekilde id kontrol ediliyor ve o kişinin rezerve ettiği odalar listeleniyor. Kullanıcıyı istediği oda için check in yapıyor. Check out sisteminde ise check in sisteminden farklı olarak check in yapılmış odalar listeleniyor. İstenilen oda check out yapılıyor ve kaç lira ücret tahsil edildiği kullanıcıya bildiriliyor. Check out yaptığımız odanın bilgileri boşaltılıyor ve tekrardan rezerve edilmeye hazır oluyor. Bu şekilde bizden istenen problemdeki gereklilikler yerine getiriliyor.

3 RESULT

3.1 Test Cases

```
@Test
public void bookRoomTest() throws IOException {
    MainMenu m = new MainMenu();
    User user = new User();
    Room[] rooms = new Room[15];
    for(int i=0; i<15; i++){
        rooms[i] = new Room();
    }
    m.fillRoomsArr(rooms);
    m.updateRecords(rooms);
    Assert.assertEquals(user.BookRoomTest( name: "Emre", surname: "Kaya", id: "880", kisi: "1", gun: "4", rooms), actual: true);
    Assert.assertEquals(user.BookRoomTest( name: "TariK", surname: "KiliC", id: "1010", kisi: "2", gun: "5", rooms), actual: true);
    Assert.assertEquals(user.BookRoomTest( name: "Fatih", surname: "Sevim", id: "98765", kisi: "4", gun: "5", rooms), actual: true);
    Assert.assertEquals(user.BookRoomTest( name: "Fatih", surname: "Sevim", id: "98765", kisi: "5", gun: "2", rooms), actual: false);
}

1 test passed - 20ms
```

▼ UserTest (com.company) 20ms
 ✓ bookRoomTest 20ms

```
"C:\Program Files\Java\jdk1.8.0_151\bin\java" ...
Kaç Kişilik Oda Rezerve Etmek İstiyorsunuz?(1-2-3-4)
Kaç Günlük Rezerve Etmek İstiyorsunuz? 3 Numaralı Oda Rezerve Edildi.
Kaç Kişilik Oda Rezerve Etmek İstiyorsunuz?(1-2-3-4)
Kaç Günlük Rezerve Etmek İstiyorsunuz? 8 Numaralı Oda Rezerve Edildi.
Kaç Kişilik Oda Rezerve Etmek İstiyorsunuz?(1-2-3-4)
Kaç Günlük Rezerve Etmek İstiyorsunuz? 14 Numaralı Oda Rezerve Edildi.
Kaç Kişilik Oda Rezerve Etmek İstiyorsunuz?(1-2-3-4) Lütfen geçerli bir deger giriniz!
Process finished with exit code 0
```

“Rooms.csv” dosyasına rezervasyon yaptırmaya çalışıyorum. İlk 3 testten geçiyor çünkü 1, 2 ve 4 kişilik odalar mevcut olduğu için true return ediyor. Fakat son testte 5 kişilik oda mevcut olmadığı için false return ediyor. Böylece 4 testte de başarılı bir şekilde geçiyor.

```
@Test
public void cancelBookTest() throws IOException {
    MainMenu m = new MainMenu();
    User user = new User();
    Room[] rooms = new Room[15];
    for(int i=0; i<15; i++){
        rooms[i] = new Room();
    }
    m.fillRoomsArr(rooms);
    m.updateRecords(rooms);
    Assert.assertEquals(user.CancelBookTest( id: "880", oda: "1", rooms), actual: true);
    Assert.assertEquals(user.CancelBookTest( id: "200", oda: "2", rooms), actual: false);
}

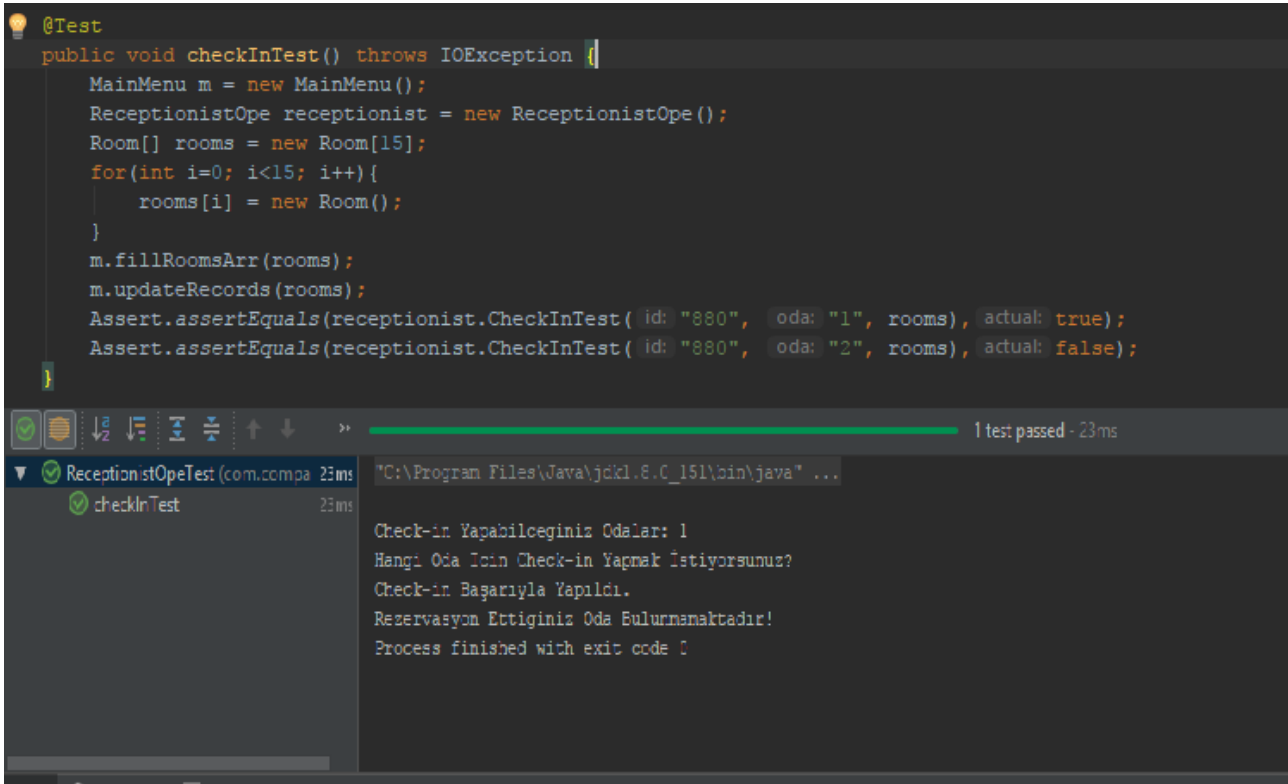
1 test passed - 25ms
```

▼ UserTest (com.company) 25ms
 ✓ cancelBookTest 25ms

```
"C:\Program Files\Java\jdk1.8.0_151\bin\java" ...
Rezervasyon İptali Yapabilceğiniz Odalar: 1 2
Hangi Odanın Rezervasyonunu İptal Etmek İstiyorsunuz?
Rezervasyon Başarıyla İptal Edildi.
Rezervasyon Ettiğiniz Oda Bulunmamaktadır!
Process finished with exit code 0
```

“Rooms.csv” dosyasından rezervasyon iptali yaptırmaya çalışıyorum. 1. Odada 880 numaralı kişi

kayıtlı olduğu için rezervasyon iptali başarılı olur ve true return ediyor.Fakat 2. Test için 2 numaralı odada 880 numaralı kişi checkin yaptırmış olduğu için rezervasyon iptal edilemiyor ve false return ediyor.Böylece 2 testte geçiyor.



```
@Test
public void checkInTest() throws IOException {
    MainMenu m = new MainMenu();
    ReceptionistOpe receptionist = new ReceptionistOpe();
    Room[] rooms = new Room[15];
    for(int i=0; i<15; i++){
        rooms[i] = new Room();
    }
    m.fillRoomsArr(rooms);
    m.updateRecords(rooms);
    Assert.assertEquals(receptionist.CheckInTest( id: "880", oda: "1", rooms), actual: true);
    Assert.assertEquals(receptionist.CheckInTest( id: "880", oda: "2", rooms), actual: false);
}
```

1 test passed - 23ms

ReceptionistOpeTest (com.compa 23ms

checkInTest 23ms

Check-in Yapabilceğiniz Odalar: 1
Hangi Oda İçin Check-in Yapmak İstiyorsunuz?
Check-in Başarıyla Yapıldı.
Rezervasyon Ettiğiniz Oda Bulunmamaktadır!
Process finished with exit code 0

“Rooms.csv” dosyasından checkin yaptırmaya çalışıyorum. 1. Odada 880 numaralı kişi kayıtlı olduğu ve oda rezerve edildiği için checkin başarılı olur ve true return ediyor.Fakat 2. Test için 2 numaralı odada 880 numaralı kişi checkin yaptırdığı için tekrardan check in yapılamıyor ve false return ediyor.Böylece 2 testte geçiyor.

```
@Test
public void checkOutTest() throws IOException {
    MainMenu m = new MainMenu();
    ReceptionistOpe receptionist = new ReceptionistOpe();
    Room[] rooms = new Room[15];
    for(int i=0; i<15; i++){
        rooms[i] = new Room();
    }
    m.fillRoomsArr(rooms);
    m.updateRecords(rooms);
    Assert.assertEquals(receptionist.CheckOutTest( id: "880", oda: "1", rooms), actual: false);
    Assert.assertEquals(receptionist.CheckOutTest( id: "880", oda: "2", rooms), actual: true);
}
```

1 test passed - 25ms

ReceptionistOpeTest (com.compa 25ms)

checkOutTest 25ms

Check-out Yapabilceğiniz Odalar: 2
Hangi Oda İçin Check-out Yapmak İstiyorsunuz?
Lütfen Geçerli Bir Oda Giriniz.
Check-out Yapabilceğiniz Odalar: 2
Hangi Oda İçin Check-out Yapmak İstiyorsunuz?
Check-out Başarıyla Yapıldı.
200 TL Tahsil Edildi.Yine Bekleriz.

Process finished with exit code 0

“Rooms.csv” dosyasından checkout yaptırmaya çalışıyorum. 1. Odada 880 numaralı kişi rezerve ettiği için checkout yapamıyor ve false return ediyor.Fakat 2. Test için 2 numaralı odada 880 numaralı kişi checkin yaptırdığı için checkout yapıyor ve true return ediyor.Böylece 2 testende geçiyor.

3.2 Running Results

Proje dosyamın içindeki “3.txt” isimli dosyadan aldığım inputlardan alınan output:

```
GTU Otele Hosgeldiniz!

1 - Resepsiyonist Olarak Giris Yap
2 - Kullanici Olarak Giris Yap
3 - Sisteme Kayit Ol
0 - Otomasyondan Cık

Islem Seciniz:

Lutfen ID'nizi giriniz:
Lutfen Parolanizi giriniz:
Giris Basarili!
1 - Rezervasyon Yap
2 - Rezervasyonunu iptal Et
0 - Ana Menuye Don

Islem Seciniz:

Kaç Kigilik Oda Rezerve Etmek Istiyorsunuz?(1-2-3-4)
Kaç Günlük Rezerve Etmek Istiyorsunuz? 14 Numaralı Oda Rezerve Edildi.
1 - Rezervasyon Yap
2 - Rezervasyonunu iptal Et
0 - Ana Menuye Don

Islem Seciniz:

1 - Rezervasyon Yap
2 - Rezervasyonunu iptal Et
0 - Ana Menuye Don

Islem Seciniz:

Rezervasyon İptali Yapabileceginiz Odalar: 1 2 14
```

Öncelikle kullanıcı girişi yapıp 4 kişilik oda rezerve ediyorum.Daha sonra ise rezervasyon iptali yapıp 14. Odadaki kaydımı siliyorum.

```
Rezervasyon İptali Yapabileceginiz Odalar: 1 2 14
Hangi Odanın Rezervasyonunu İptal Etmek Istiyorsunuz?
Rezervasyon Başarıyla İptal Edildi.
1 - Rezervasyon Yap
2 - Rezervasyonunu iptal Et
0 - Ana Menuye Don

Islem Seciniz:

1 - Resepsiyonist Olarak Giris Yap
2 - Kullanici Olarak Giris Yap
3 - Sisteme Kayit Ol
0 - Otomasyondan Cık

Islem Seciniz:

Lutfen ID'nizi giriniz:
Lutfen Parolanizi giriniz:
Giris Basarili!
1 - Rezervasyon Yap
2 - Rezervasyonunu iptal Et
3 - Check-in Yap
4 - Check out Yap
0 - Ana Menuye Don

Islem Seciniz:

İsminiz:
Soyisminiz:
ID'niz:
Kaç Kigilik Oda Rezerve Etmek Istiyorsunuz?(1-2-3-4)
Kaç Günlük Rezerve Etmek Istiyorsunuz? 14 Numaralı Oda Rezerve Edildi.
1 - Rezervasyon Yap
2 - Rezervasyonunu iptal Et
3 - Check-in Yap
4 - Check out Yap
0 - Ana Menuye Don
```

Ana menüye dönüp resepsiyonist olarak giriş yapıyorum ve aynı kişi için tekrar 4 kişilik oda rezerve ediyorum.


```
1 - Rezervasyon Yap
2 - Rezervasyonunu Iptal Et
3 - Check-in Yap
4 - Check out Yap
0 - Ana Menüye Dön

İslem Seciniz:

1 - Rezervasyon Yap
2 - Rezervasyonunu Iptal Et
3 - Check-in Yap
4 - Check out Yap
0 - Ana Menüye Dön

İslem Seciniz:

ID'niz:
Check-in Yapabilceğiniz Odalar: 1 14
Hangi Oda İçin Check-in Yapmak İstiyorsunuz?
Check-in Başarıyla Yapıldı.
1 - Rezervasyon Yap
2 - Rezervasyonunu Iptal Et
3 - Check-in Yap
4 - Check out Yap
0 - Ana Menüye Dön

İslem Seciniz:

ID'niz:
Check-out Yapabilceğiniz Odalar: 2 14
Hangi Oda İçin Check-out Yapmak İstiyorsunuz?
Check-out Başarıyla Yapıldı.
800 TL Tahsil Edildi.Yine Bekleriz.
```

14. Oda boş olduğu için 14. Odayı tekrardan rezerve etmiş oluyoruz.Daha sonra rezerve ettiğimiz bu odayı check-in yapıyorum.Check-in yaptıktan sonra ise check-out yapıyorum.Kaldığı gün sayısını, oda kapasitesiyle çarpıyorum.Daha sonra ise sonucu 50 TL ile çarpıyorum ve ne kadar ücret tahsil edildiğini ekrana yazdırıyorum.Daha sonra ise ana menüye dönüp otomasyondan çıkış yapıyorum.

Denemeniz için ekstra olarak “1.txt” ve “2.txt” de farklı senaryolar yazıp ekledim.Dilerseniz, MainWrapper class’ında “3.txt” yi yukardaki dosyalardan biriyle değiştirip o senaryolarıda test edebilirsiniz.

“1.txt” dosyamda kullanıcı girişi yapıp oda rezerve ediyorum.Daha sonra ise aynı oda üzerinde rezervasyon iptali yapıp otomasyondan çıkıyorum.

“2.txt” dosyamda ise resepsiyonist girişi yapıp kullanıcı için rezervasyon yapıyorum.Daha sonra rezerve ettiğim odada check-in yapıyorum.Check-in yaptıktan sonra ise check-out yaparak ne kadar ücret tahsil edildiğini gösterip programı sonlandırıyorum.

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