

**Gebze Technical University
Computer Engineering**

CSE 222 - 2018 Spring

HOMEWORK 1 REPORT

**BURAK DEMİRCİ
141044091**

1.1 Course Assistant: Fatma Nur Esirci

2 INTRODUCTION

2.1 Problem Definition

Temel Otel Yönetim Sistemi: Otel odalarının müşteri için rezervasyon , kayıt ve kayıt silme işlemlerinin yapılması bunlarla alakalı temel kontroller.

Otel sistemini iki farklı kullanıcı tarafından kullanılacak.

1. Kullanım Müşteri Kullanımı: Müşteri otele rezervasyon yapabilecek ve rezervasyonu iptal edebilecek
2. Kullanım Respyon Görevlisi: Rezervasyon, rezervasyonu iptali, kayıt ve kayıt silme yapabilecek.

Yapılan her işlem log(kayıt) olarak dosyalarda tutulacaktır.

2.2 System Requirements

Hotel Management sistemi için gerekli olan yapılar, veriler ve yöntemler aşağıdaki gibidir: Otel için gerekli veriler Otelin odalarının numaraları her bir odanın kapasitesi ve odaların durumları sistem kurulmadan önce girilmelidir. Kullanıcı için bir dosya oluşturulmalıdır ve Resepsiyonist için bir dosya oluşturulup içerisine resepsiyonistin adı, kimlik numarası (ID) ve şifresi baştan dosyanın içerisine konulmalıdır. Sistem için gerekli olan yapılar Oda(Room) yönetim sistemi Müşteri (Guest) yönetim sistemi, Resepsiyon (Receptionist) yönetim sistemi ve bütün bu sistemleri entegre halde yöneten Otel (Hotel) yönetim sisteminden oluşmalıdır. Sistem müşteri (guest) veya resepsiyon (receptionist) görevlisi olarak giriş yapılarak kullanılmalıdır. Müşteri rezervasyon (book room) ve rezervasyon iptali (cancel reservation) yapabilmelidir. Resepsiyon görevlisi ise çalışan olduğundan öncelikle sistemde daha çok yetkisi bulunmalıdır ve bu yetkilerden dolayı sisteme resepsiyon görevlisi olarak girilirken isim ve şifre sorgulaması ile girilmelidir. Resepsiyon görevlisi rezervasyon yapabilme, rezervasyon iptali, kayıt yapabilme ve kayıt silme işlevleri yanı sıra kendi şifresini değiştirebilme özelliklerine sahip olmalıdır.

Class Yapılarının Genel Özellikleri

Hotel(String roomFile, String receptionistFile,String guestFile)

Kullanılacak bütün dosyaları kurucu yönteminde (constructor) alan bu class room,guest ve receptionist objelerini yöneterek sistemin çalışmasını sağlar.

Receptionist(String name,int id,String pass)

Resepsiyonistin ismini, kimlik numarasını ve şifresini kurucu yönteminde (constructor) alan bu class resepsiyon görevlisinin kişisel özelliklerinin oluşmasını sağlar.

Guest(String name,int id,int cp,int roomN) ve Guest()

İki ayrı kurucu yönteme (constructor overloading) sahip olan kullanıcı class'ı bir müşterinin otel sistemini kullanabilmesi için gerekli bilgi ve özelliklerin tutulduğu objedir. İki ayrı kurucu yönteme sahip olmasının nedeni ise kayıt olmadan kullanıcı sistemi kullanabilmesini sağlamaktır. Sistemde değişikliğe sebep olacak kullanım sonucu (booking) kullanıcıdan bu bilgiler istenilir.

Room(int No ,int capacity, int st)

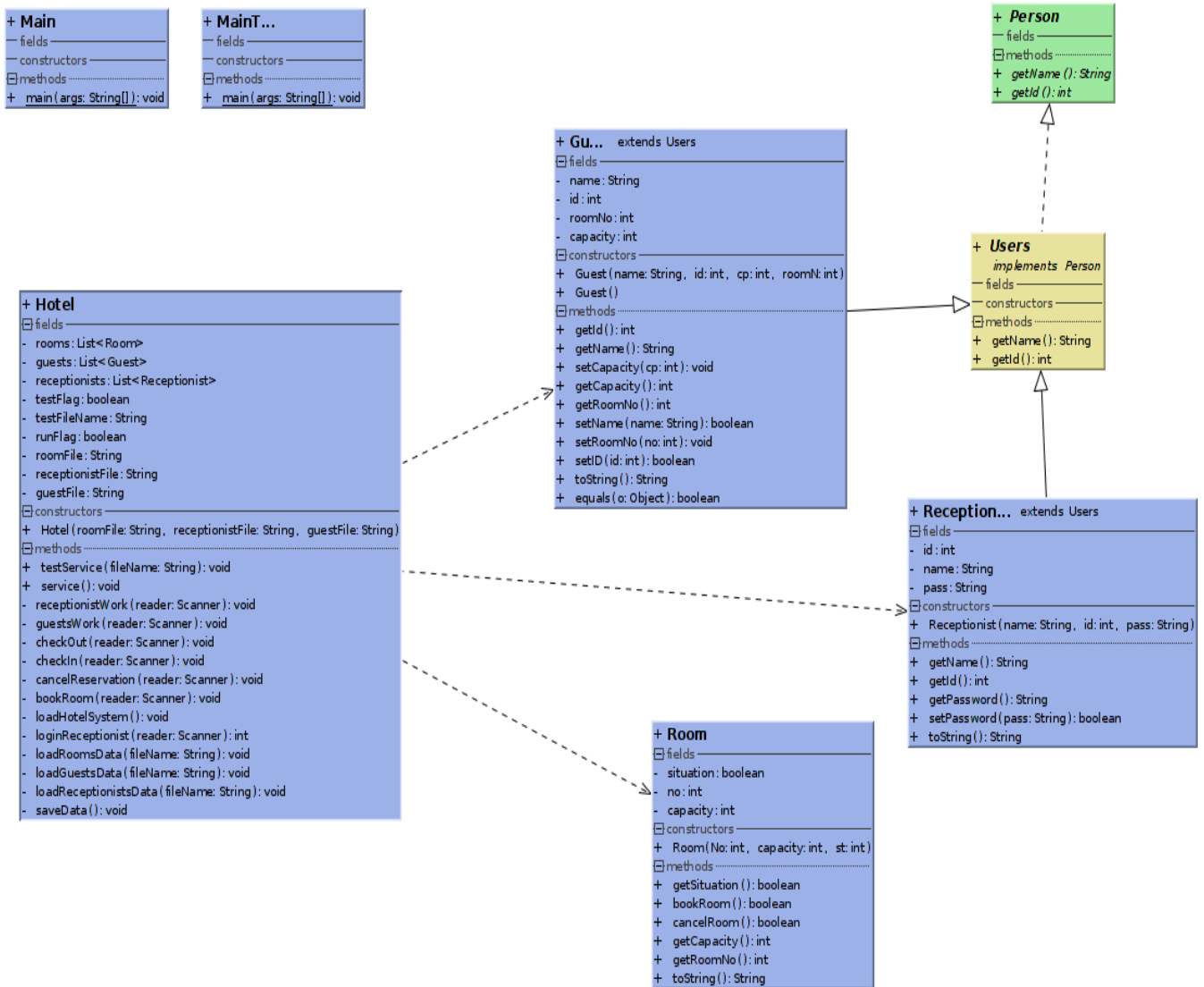
Room(Oda) kurucu yöntemi (constructor) oteldeki odaların özelliklerinin ve durumlarının (boş,dolu) bulunduğu ve bu özelliklerin kullanılmasını sağlayan class'tır

3 METHOD

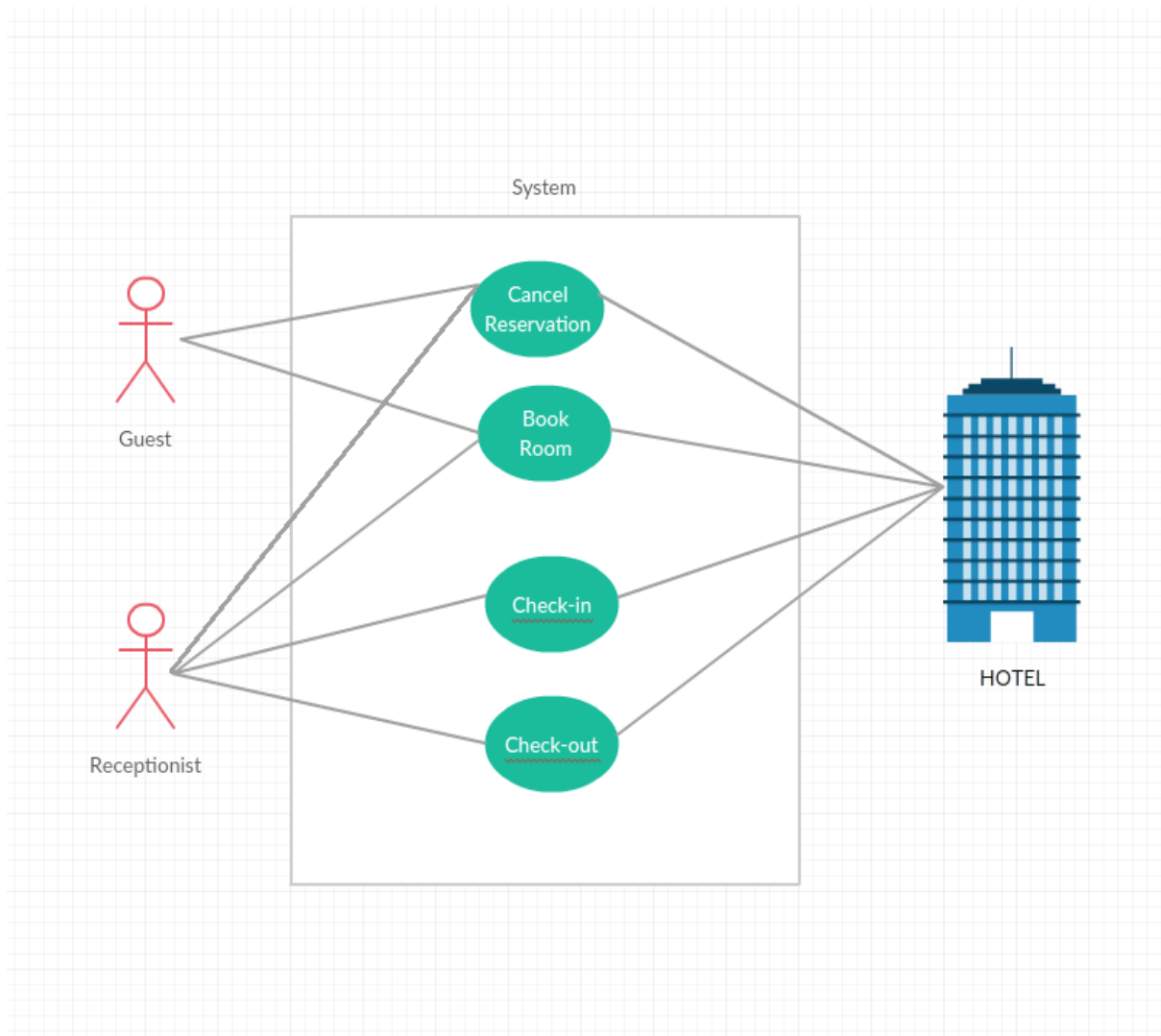
3.1 Class Diagrams

Guest ve Receptionist ikisinde insan olduğu için Person interfacesinden imlement edilmiş olan Otel için User abtract classını extends ederek kullanan yapılarıdır.

Hotel ise Guests, Receptionists ve Room classlarının objelerini ArrayList yapısında tutarak kullanan bir sistemdir.



3.2 Use Case Diagrams



3.3 Problem Solution Approach

Problemin çözümü için öncelikle kaç çeşit data dosyasına (*.csv) ihtiyaç olduğu belirlendi.

Room için Room.csv

Parametreler : RoomNumber, RoomCapacity , Room Situation (101,4,0)

RoomStuation : 0 => Boş , 1=> Dolu

Guest için Guest.csv

Parametreler : Name, ID , RoomCapacity , RoomNo (Burak,1234567,2,101)

RoomCapacity => 0 ise rezarvasyon yapılmamış , RoomNo => -1 ise kayıt yapılmamış

Receptionist için Receptionist.csv

Parametreler : Name, ID , Password (Burak,123654, burak123)

Class Yapısının belirlenmesi

Person interface sinden implement edilen Users abstract classı Guests ve Receptionists tarafından extends ediliyor.

Hotel class'ı ise room,guest ve receptionist objelerini array list olarak tutuyor.

Otel yönetim sistemin çalışma sisteminin belirlenmesi

Öncelikle Hotel objesi gerekli dataların bulunduğu dosya isimleri verilerek oluşturuyor.

Bütün datalar dosyalardan okunup belirlenen array listlere (okunan veriye göre) obje olarak ekleniyor. Hotel class'ının service() methodu çağrılarak sistem işlemeye başlıyor.

Öncelikle sistem kullanıcıdan Guests veya Receptionist olarak girişi seçmesi bekleniyor.

Guest ile Giriş yapıldığında : Book Room ve Cancel Reservation işlemleri için giriş yapması bekleniyor . Kullanıcı sadece 1 oda kiralayabilir veya rezervasyon yaptırabilir.

İşlemler için Kimlik No (ID) ve İsim (Name) alınır.

Receptionist ile Giriş Yapıldığında : BookRoom,Check-in,Check-out,SetPassword,Exit

İşlemleri için giriş yapmadan önce resepsiyon görevlisinden isim ve şifre istenilir bilgiler doğruysa yukarıdaki seçenekler çıkar ve görevliden bunlardan hangi işlemi yapacak ise girmesi beklenir. Book,Check-in ve Check-out işlemleri için müşteri Kimlik Numarası (ID) ve müşterinin talep ettiği veya rezervasyon (book room) yaptırdığı kişi sayısına göre oda belirlenip kayıt işlemi yapılır.SetPassword ise resepsiyon görevlisinin kendi giriş şifresini değiştirmesi. Exit ise sistemden çıkış yapılmasını sağlar.

Dosyaları Kaydetme (Log)

Her işlem yapıldıktan sonra değişen verileri korumak amaçlı bütün veriler kendilerine ait olan dosyalara kendilerine ait sistemde yazdırılır.

4 RESULT

4.1 Test Cases

Guest Test(JUnit) :

Guest Class'sının test edilebilecek metodları beklenen değere göre test edilmiştir.

Receptionist Test (JUnit) :

Receptionist Class'sının test edilebilecek metodları beklenen değere göre test edilmiştir.

Room Test (JUnit) :

Room Class'sının test edilebilecek metodları beklenen değere göre test edilmiştir.

Project: HW_1 ~ /Desktop/Veri

Files: MainTest.java, Room.java, Rooms.csv, Receptionists.csv, Guests.csv, GuestTest.java

```

1  import ...
2
3
4
5  class GuestTest {
6
7      Guest guest = new Guest( name: "burak", id: 12345, cp: 3, roomN: 101);
8
9      @Test
10     void getId() { assertEquals( expected: 12345, guest.getId()); }
11
12
13
14     @Test
15     void getName() { assertEquals( expected: "burak", guest.getName()); }
16
17
18
19     @Test
20     void getCapacity() { assertEquals( expected: 3, guest.getCapacity()); }
21
22
23
24     @Test
25     void getRoomNo() { assertEquals( expected: 101, guest.getRoomNo()); }
26
27
28
29     @Test
30     void setName() { assertEquals( expected: true, guest.setName("burakT")); }
31
32
33
34     @Test
35     void setRoomNo() {
36
37     }
38
39     @Test
40     void setID() { assertEquals( expected: true, guest.setID(1011)); }
41
42
43 }

```

Run: GuestTest

All 7 tests passed - 142ms

Test Results:

Test	Time
GuestTest	142ms
getName()	142ms
getId()	101ms
setID()	18ms
setRoomNo()	6ms
getRoomNo()	9ms
setName()	8ms
getCapacity()	8ms

Process finished with exit code 0

Project: HW_1 ~ /Desktop/Veri

Files: MainTest.java, Room.java, Rooms.csv, Receptionists.csv, Guests.csv, GuestTest.java, ReceptionistTest.java

```

1  import ...
2
3
4
5  class ReceptionistTest {
6
7      Receptionist receptionist = new Receptionist( name: "burak", id: 12345, pass: "burak123");
8
9
10     @Test
11     void getName() { assertEquals( expected: "burak", receptionist.getName()); }
12
13
14
15     @Test
16     void getId() { assertEquals( expected: 12345, receptionist.getId()); }
17
18
19
20     @Test
21     void getPassword() { assertEquals( expected: "burak123", receptionist.getPassword()); }
22
23
24
25     @Test
26     void setPassword() { assertEquals( expected: true, receptionist.setPassword("burak1354")); }
27
28 }

```

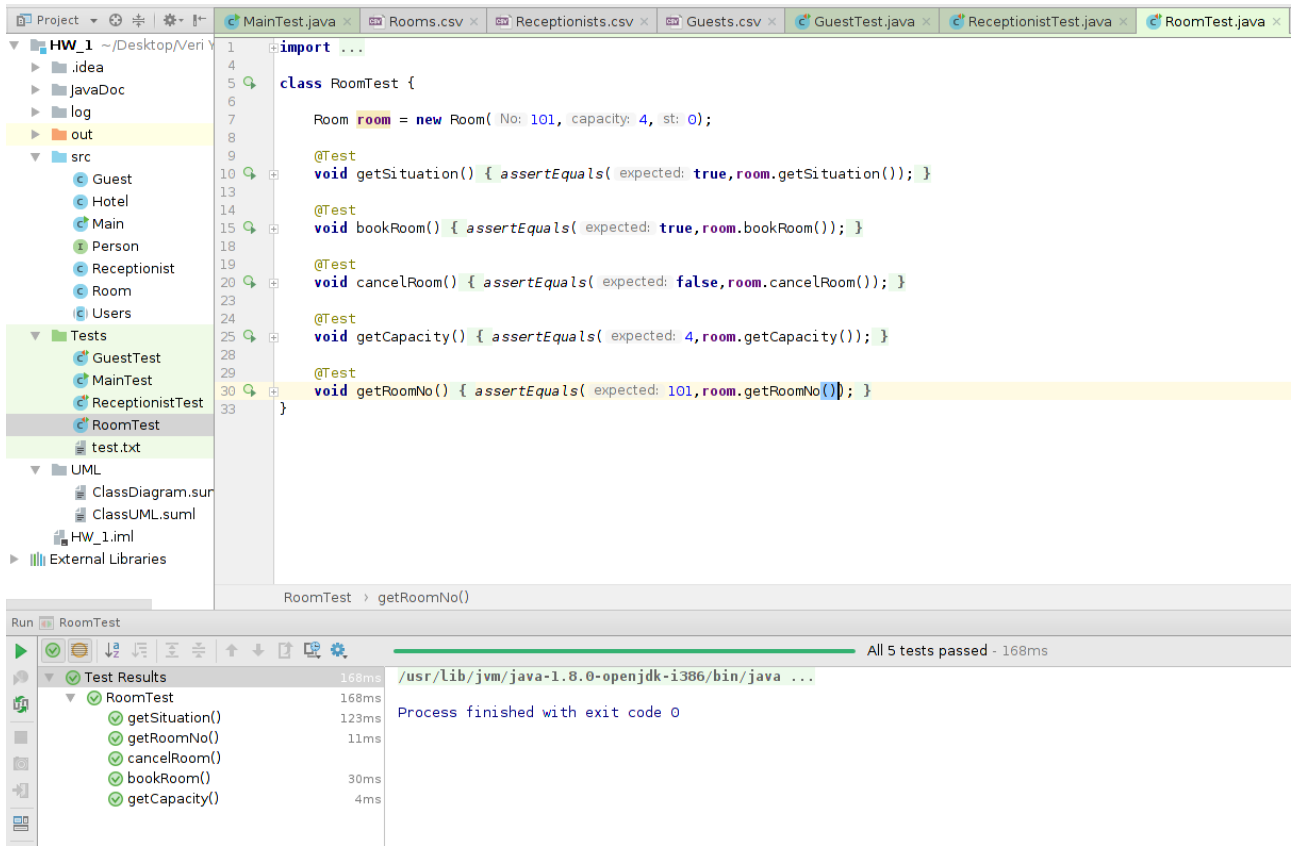
Run: ReceptionistTest

All 4 tests passed - 128ms

Test Results:

Test	Time
ReceptionistTest	128ms
setPassword()	128ms
getPassword()	89ms
getName()	14ms
getId()	6ms
getId()	19ms

Process finished with exit code 0



4.2 Running Results

Guests işlemleri :

Guests sistemi kullanırken sadece kimlik numarası (ID) ve isim (name) alınır. Herhangi bir şifre ile giriş guest için bulunmamaktadır.

Book Room : İlk işlemde aşağıdaki resimde de görüldüğü gibi Guest olarak giriş yapıldığında Book Room (Rezervasyon) yapma seçeneğini giriyor . Giriş yaptıktan sonra kullanıcıdan kimlik numarası (ID) ve kaç kişilik oda istenildiğinin girilmesi bekleniyor giriş yaptıktan sonra sitem eğer müşterinin isteğine uygun oda varsa booking(rezervasyon) yapıyor yok ise ana ekrana dönüş yapıyor. Booking (rezervasyon) yapıldığında sistem bütün verileri ilgili dosyalara kaydediyor.

Cancel Reservation : Guest olarak giriş yaptıktan sonra rezervasyon iptali (Cancel Reservation) işlemi seçildiğinde kullanıcıdan kimlik numarası isteniliyor. Kimlik numarası üzerinden kişiye ait olan bir rezervasyon var mı yok mu kontrol ediliyor eğer kişiye ait bir rezervasyon var ise rezervasyon iptal ediliyor , kişiye ait rezervasyon yok ise belirtilerek (Rezervasyon bulunamadı) bütün işlemlerde olduğu gibi veriler ilgili dosyalara yazılarak ana menüye dönülüyor.

```
Run Main
/usr/lib/jvm/java-1.8.0-openjdk-1386/bin/java ...
#####
# OTEL SARI #
#####

Giriş yapmak istediğiniz Yöntemin kodunu Giriniz
Kod Yöntem
1 Misafir
2 Resepsiyon

1

Misafir Girişi Yapıldı

Yapmak istediğiniz İşlemin Kodunu Giriniz
Kod İşlem
1 Rezervasyon
2 Rezervasyon İptali

1

Kimlik Numarası Giriniz :
12365
Sistemde kaydınız yok Kayıt işlemi için
İsminizi Giriniz :
murat
Kaç Kişilik Oda İstiyorsunuz Lütfen Giriniz:
5

Rezervasyon İşlemi Başarı ile Yapılmıştır.
Odaya kesin kayıt işlemi Otele geldiğinizde sağlanacaktır

#####
# OTEL SARI #
#####

Giriş yapmak istediğiniz Yöntemin kodunu Giriniz
Kod Yöntem
1 Misafir
2 Resepsiyon

1

Misafir Girişi Yapıldı

Yapmak istediğiniz İşlemin Kodunu Giriniz
Kod İşlem
1 Rezervasyon
2 Rezervasyon İptali

2

Kimlik Numaranızı Giriniz :
12365

Başarı ile iptal edildi

Compilation completed successfully in 972ms (2 minutes ago)
```

Receptionist işlemleri:

Receptionist sistemi kullanırken ilk önce isim ve şifre ile sisteme girmesi gerekmektedir.

Book Room : Giriş yapıldıktan sonra kullanıcıdan kimlik numarası (ID) ve kaç kişilik oda istenildiğinin girilmesi bekleniyor giriş yapıldıktan sonra sistem eğer müşterinin isteğine uygun oda varsa booking(rezervasyon) yapıyor yok ise ana ekrana dönüş yapılıyor. Booking (rezervasyon) yapıldığında sistem bütün verileri ilgili dosyalara kaydediyor.

Cancel Reservation : Kimlik numarası üzerinden kişiye ait olan bir rezervasyon var mı yok mu kontrol ediliyor eğer kişiye ait bir rezervasyon var ise rezervasyon iptal ediliyor .

Check-in : Kişinin kimlik numarası(ID) üzerinden rezervasyonu var ise direkt uygun olan oda kişiye verilir rezervasyonu yok ise book room işlemleri gibi isim ve istenilen oda kapasitesi sorularak boş oda var ise oda kişiye verilir.

Check-out : Kişinin kimlik numarası(ID) üzerinden kişiye ait bir oda var mı yok mu sorgusu yapıldıktan sonra eğer kişiye ait bir oda var ise çıkış işlemleri yapılır. Kişiye ait oda yok ise belirtilerek menüye dönüş yapılır.

Set Password : Receptionist sisteme giriş yaparken kullandığı şifreyi yeniden belirlemesini sağlayan sistemdir.

Çıkış : Resepsiyonist hesabından çıkış yapmasını sağlayarak ana menüye dönüş yapılır.


```
Run Main
/usr/lib/jvm/java-1.8.0-openjdk-1386/bin/java ...
#####
# OTEL SARI #
#####

Giriş yapmak istediğiniz yöntemin kodunu giriniz
Kod Yöntem
1 Misafir
2 Resepsiyon

2
İsminizi giriniz :
burak
Şifrenizi giriniz :
burak123

Giriş başarılı
Yapmak istediğiniz işlemin kodunu giriniz
Kod İşlem
1 Rezervasyon
2 Rezervasyon iptali
3 Kayıt oluştur
4 Kayıt sil
5 Şifre yenile
6 Çıkış

1
Kimlik numarası giriniz :
12356
Kaç kişilik oda istiyorsunuz lütfen giriniz:
5

Rezervasyon işlemi başarı ile yapılmıştır.
Odaya kesin kayıt işlemi otele geldiğinizde sağlanacaktır

Yapmak istediğiniz işlemin kodunu giriniz
Kod İşlem
1 Rezervasyon
2 Rezervasyon iptali
3 Kayıt oluştur
4 Kayıt sil
5 Şifre yenile
6 Çıkış

2
Kimlik numaranızı giriniz :
12356

Başarı ile iptal edildi

Run Main
Yapmak istediğiniz işlemin kodunu giriniz
Kod İşlem
1 Rezervasyon
2 Rezervasyon iptali
3 Kayıt oluştur
4 Kayıt sil
5 Şifre yenile
6 Çıkış

3
Kimlik numarası giriniz :
12356
Kaç kişilik oda istiyorsunuz lütfen giriş yapınız
5
Kayıt başarı ile oluşturuldu.
Kayıt bilgisi :
İsim :murat
Kimlik No :12356
Oda No :105
Yapmak istediğiniz işlemin kodunu giriniz
Kod İşlem
1 Rezervasyon
2 Rezervasyon iptali
3 Kayıt oluştur
4 Kayıt sil
5 Şifre yenile
6 Çıkış

4
Kimlik numarası giriniz : 12356
Kimlik No :12356 İsim :murat Oda No :105
Kayıt silme işlemi başarı ile gerçekleşti
Yapmak istediğiniz işlemin kodunu giriniz
Kod İşlem
1 Rezervasyon
2 Rezervasyon iptali
3 Kayıt oluştur
4 Kayıt sil
5 Şifre yenile
6 Çıkış

5
Yeni şifrenizi giriniz > burak123
Şifreniz başarı ile değiştirildi
Yapmak istediğiniz işlemin kodunu giriniz
Kod İşlem
1 Rezervasyon
2 Rezervasyon iptali
3 Kayıt oluştur
4 Kayıt sil
5 Şifre yenile
6 Çıkış

6
#####
```