**1.Project description**

**2. Database**

**2.1. EER Modeling Diagram**

**2.2. ER-Model Mapping to Database Relational Schema:**

User(userId, password, userType)

Customer (userId\*, name, address, postalCode, country, email, phone, dob)

Staff(userId\*, name, address, postalCode, email, phone, position)

Reservation (reservationNr, startDate, endDate, userId, finalPrice, paymentDate\*)

Room (roomNr, type, price)

Service (serviceNr, type, price)

Reservation \_involves\_room (roomNr\*, reservationNr\*)

Reservation\_includes\_service (serviceNr\*, reservationNr\*)

*Notes: primary key , \* foreign key*

* 1. **Data Types (Domain) and Constraints:**

userId :TINYINT(6)

password**:** VARCHAR(20)

userType: VARCHAR(20)

name: VARCHAR(40)

postalCode: VARCHAR(8)

country: VARCHAR(20)

email: VARCHAR(20)

phone: TINYINT(14)

dob: DATE

position: VARCHAR(20)

reservationNr: TINYINT(6)

startDate: DATE

endDate: DATE

finalPrice: DECIMAL (4,2)

paymentDate: DATE

roomNr TINYINT(4)

Room.type: VARCHAR(20)

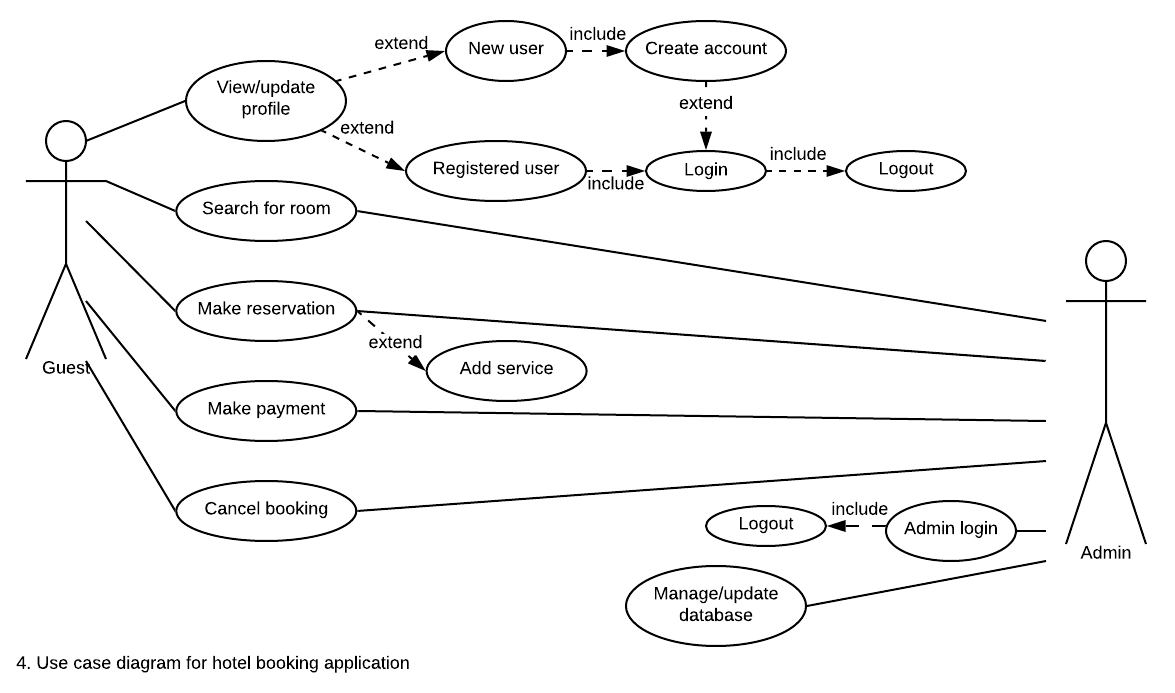
ServiceNr: VARCHAR(20)

Service.type CHAR(20)

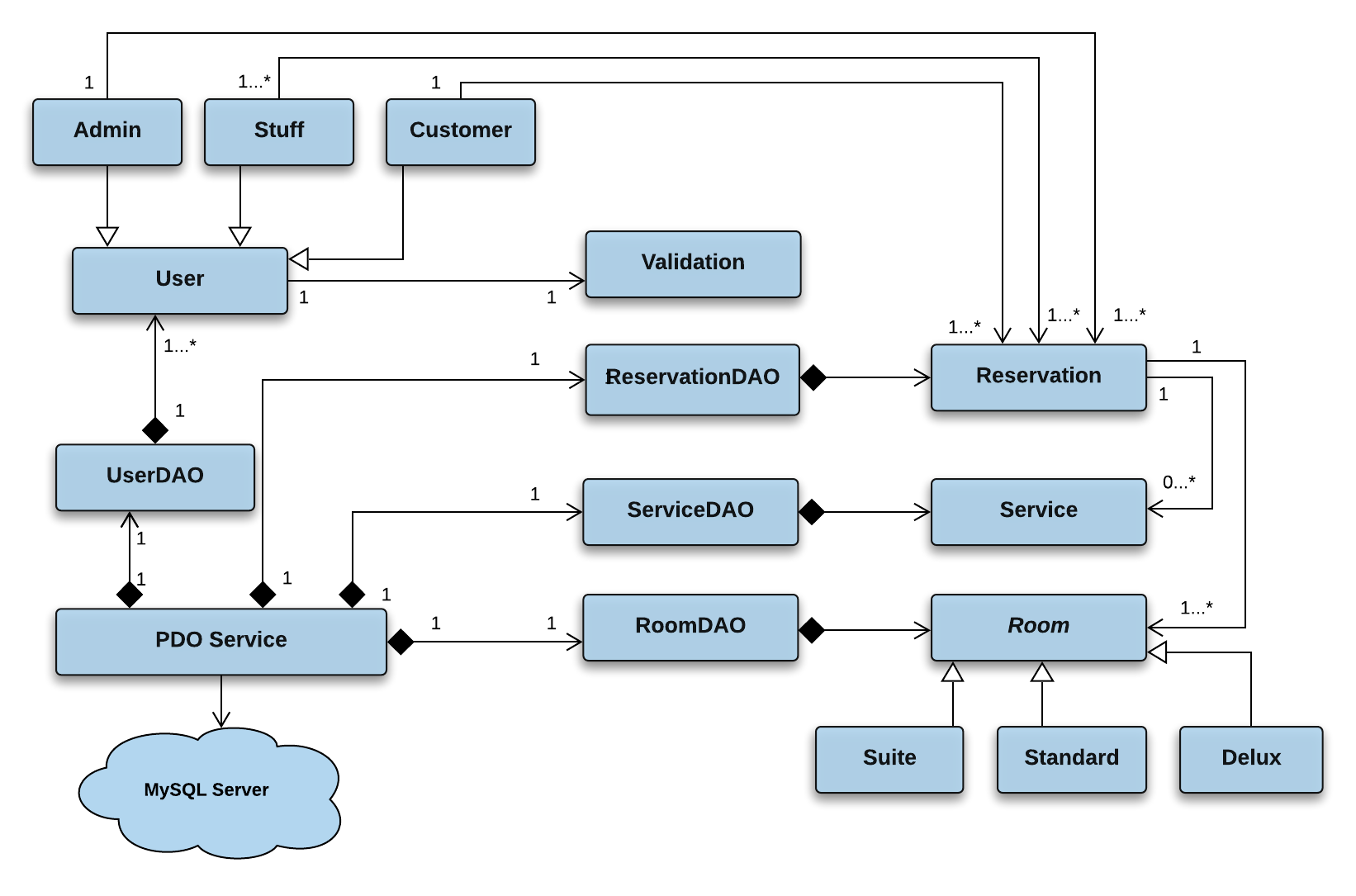
*Notes: All data constrained: NOT NUL*L

**3. UML diagrams for Hotel Booking Application**

**3.1. Use case diagram**

****

**3.2. Class diagram with navigability arrows**

****

*Notes:* For better readability *the diagram does not include Page class*

**3.3. Class diagrams with attributes and methods**

|  |
| --- |
| **User** |
| - userId: integer  - password: string  - userType(): string  - name: string  - address: string  - postalCode: string  - email: string  - phone: int |
| + verifyLogin(): bool  + login() |

|  |
| --- |
| **Page** |
| + title: string |
| + getHeader()  + displayMainArticle ()  + displayUserRegistrationForm()  + displayReservationForm()  + showNotification()  + getFooter()  ……………………….. |

|  |
| --- |
| **Validation** |
|  |
| + validateEntries()  ……………………….. |

|  |
| --- |
| **Customer** |
|  |
| + createAccount()  + updateAccount() |

|  |
| --- |
| **Staff** |
| - position: string |
|  |

|  |
| --- |
| **PDOService** |
|  |
|  |

|  |
| --- |
| **Admin** |
| - adminName: string  - email: sting |
| + updateDatabase() |

|  |
| --- |
| **ReservationDAO** |
|  |
|  |

|  |
| --- |
| **Reservation** |
| - reservationNr: integer  - startDate: date  - endDate: date  - finalPrice: decimal  - paymentDate: date |
| + addReservation()  + editReservation()  + deleteReservation() |
| + searchReservation()  + calculatePayment()  + printPayment()  + updatePayment() |

|  |
| --- |
| **RoomDAO** |
|  |
|  |

|  |
| --- |
| **ServiceDAO** |
|  |
|  |

|  |
| --- |
| ***Room*** |
| - roomNr: integer  - roomType: string |
| *+ getPricePerNight()*  + addRoom()  + deleteRoom()  + getAvailableDate(); |

|  |
| --- |
| **Service** |
| - serviceNr: integer  - type: string  - price: decimal |
| +getPrice()  +setPrice()  + addService()  + deleteService() |

|  |
| --- |
| **Standard** |
| - pricePerNight : double |
| + getPricePerNight()  + setPricePerNight() |

|  |
| --- |
| **Delux** |
| - pricePerNight : double |
| + getPricePerNight()  + setPricePerNight() |

|  |
| --- |
| **Suite** |
| + roomDescription: string  - pricePerNight: double |
| + getPricePerNight() |

**4. Meeting minutes**

**5. User manual**

**References:**