

# CS6852 Assignment 2 - Hotel Ontology

## Group 7

cs20b006 Akila Tharini Sivakumar      cs20b070 Sasubilli Yuvan  
cs19b022 Katta Manasa      cs20b008 Araj Khandelwal  
cs20b007 Aniket Singh Patel

March 2023

### 1 Knowledge captured by the Ontology

1. A hotel has exactly one Hotel ID.
2. Every hotel has an address and a contact phone number.
3. Only a hotel may have a website and rating.
4. Every hotel has one or more departments.
5. A hotel may offer some services and/or amenities.
6. Amenities must be booked beforehand whereas services can be availed on demand. Hence, a service is not considered an amenity and vice versa.
7. A hotel has at least one floor.
8. Every room in a hotel is located in some floor.
9. Every department has a unique department ID.
10. Each department has one or more employees. It is headed by one of those employees.
11. Every employee is a person with a unique Employee ID.
12. Each employee belongs to some department.
13. Every employee has an address and a contact phone number.
14. Each employee works in a some specific role and anyone who works in some role in the hotel must be an employee.
15. An assistant is an employee who helps the head of some department.
16. Every employee has a unique login username.
17. There is a unique password for each login username.
18. Every room is represented by its room number, is of some type and costs a specific amount.
19. A room can be booked for a check-in to check-out duration only by making a reservation.
20. A customer is a person with a unique Customer ID.
21. Each customer has made at least one reservation at the hotel.
22. Each customer has an address and a contact phone number.
23. A reservation is identified by its Reservation ID and is made on some date.
24. Each service offered by the hotel has a service ID and an associated cost.
25. Booking for any amenity may be made by some customer for a particular slot time, date and number of guests.
26. Only amenities can have a maximum capacity.

## 2 DL Ontology TBox

$Hotel \equiv (= 1 \text{ hasID.HotelID})$	T.1
$\exists \text{hasRating}.\top \sqsubseteq Hotel$	T.2
$\exists \text{hasWebsite}.\top \sqsubseteq Hotel$	T.3
$Hotel \sqsubseteq \exists \text{hasAddress.Address}$	T.4
$Hotel \sqsubseteq \exists \text{hasContact.PhoneNo} \sqcap \exists \text{hasContact.Email}$	T.5
$Hotel \sqsubseteq \exists \text{hasDepartment.Department} \sqcap \forall \text{hasDepartment.Department}$	T.6
$\exists \text{offers}.(Service \sqcup Amenity) \sqsubseteq Hotel$	T.7
$Service \sqsubseteq \neg Amenity$	T.8
$Hotel \equiv \exists \text{hasFloor.Floor}$	T.9
$\exists \text{hasRoom.Room} \sqsubseteq Floor$	T.10
$Department \equiv (= 1 \text{ hasID.DeptID}) \sqcap (= 1 \text{ hasHead.Employee})$	T.11
$Department \equiv (\geq 1 \text{ belongsTo}^{-1}.Employee)$	T.12
$Employee \sqsubseteq Person \sqcap (= 1 \text{ hasID.EmpID})$	T.13
$DeptHead \equiv Employee \sqcap \exists \text{hasHead}^{-1}.Department$	T.14
$Assistant \equiv Employee \sqcap \exists \text{helps.DeptHead}$	T.15
$Employee \sqsubseteq \exists \text{belongsTo.Department}$	T.16
$Employee \equiv (= 1 \text{ hasLogin.Username})$	T.17
$Employee \equiv \exists \text{hasAddress.Address} \sqcap \exists \text{hasContact.PhoneNo}$	T.18
$Employee \equiv \exists \text{worksAs.Role}$	T.19
$Username \equiv (= 1 \text{ hasPwd.Password})$	T.20
$Room \equiv \exists \text{ofType.RoomType} \sqcap \exists \text{costs.RoomCost}$	T.21
$Customer \equiv Person \sqcap (= 1 \text{ hasID.CustID})$	T.22
$Customer \sqsubseteq \exists \text{hasAddress.Address} \sqcap \exists \text{hasContact.PhoneNo} \sqcap \exists \text{hasContact.Email}$	T.23
$Customer \equiv (\geq 1 \text{ hasReservation.Reservation})$	T.24
$Reservation \equiv (= 1 \text{ checkIn.Date}) \sqcap (= 1 \text{ checkOut.Date}) \sqcap (= 1 \text{ madeOn.Date})$	T.25
$Reservation \equiv (= 1 \text{ reserved.RoomNo})$	T.26
$Service \equiv (= 1 \text{ hasID.ServiceID}) \sqcap \exists \text{costs.ServiceCost}$	T.27
$Booking \equiv \exists \text{madeBooking}^{-1}.Customer \sqcap \exists \text{booked.Amenity}$	T.28
$Booking \equiv \exists \text{SlotStart.Time} \sqcap \exists \text{SlotDate.Date} \sqcap \exists \text{noOfPpl}.\top$	T.29
$\exists \text{maxCapacity}.\top \sqsubseteq Amenity$	T.30

## 3 Design Choices

The concepts used in the ontology are :

1. Hotel
2. Address
3. PhoneNo
4. Email
5. Service
6. Amenity
7. Department
8. Floor
9. Room
10. Person
11. Employee
12. Assistant
13. DeptHead

14. Username
15. Password
16. Role
17. RoomType
18. RoomCost
19. ServiceCost
20. Customer
21. Reservation
22. Date
23. Time
24. Booking
25. HotelID, DeptID, EmpID, CustID, ResID, SeviceID

The roles used in the ontology are:

1. hasId
2. hasRating
3. hasWebsite
4. hasAddress
5. hasContact
6. hasDepartment
7. offers
8. hasFloor
9. hasRoom
10. hasHead
11. helps
12. belongsTo
13. worksAs
14. hasLogin
15. hasPwd
16. ofType
17. hasReservation
18. checkIn, checkOut
19. madeOn
20. booked
21. costs
22. madeBooking
23. SlotTime
24. SlotDate
25. noOfPpl
26. maxCapacity

Of the above, the primitive concepts are Address, PhoneNo, Email, Floor, Person, Role, Password, RoomType, RoomCost, Date, Time, ServiceCost, HotelID, DeptID, EmpID, CustID, ServiceID.

### Explanation for design choices:

The hotel and department concepts will have their respective names as elements. Employee and customer concepts also consist of the names of the people. Whereas, floor and room concepts will consist of floor numbers and room numbers respectively. Reservation and booking concepts will consist of the respective IDs. This choice has been made since in most cases, names are unique and give a better idea about a hotel, person or department than an ID. But room, floor, booking etc are only identifiable by their IDs.

Since a hotel ID uniquely identifies a hotel, a hotel can have only one hotel ID and each ID can correspond to only one hotel. Hence, this has been expressed as in the equivalence axiom T.1. Only a hotel can have a rating and/or website. Thus, anything that has a rating must be a hotel. Similar reasoning applies for website. Hence, these have been expressed as GCIs (T.2, T.3) in the TBox.

Since Email and Website are only allowed for a hotel and have not been used anywhere else in the TBox, they have not been defined as concepts.

A hotel has an address and phone number. But so does every employee and customer. So, elements that have an address and contact detail need not be a hotel. Hence, this piece of information is modelled using GCI's (T.4, T.5, T.18, T.23) and not equivalence axioms.

The hotel must have atleast one department and it can have only members of the Department concept as its department. The first and second part of the conjunction in T.6 capture these two pieces of information respectively.

T.7 ensures that every service or amenity is offered only by a hotel. T.8 ensures that Services and Amenities are disjoint.

A hotel may have some floor without any rooms, that may be used for other purposes such as storage. Hence, T.10 has been written in this form instead of  $Floor \equiv \exists hasRoom.Room$

T.12 ensures that each department has one or more employees.

In T.14, an inverse role is used to find employees for whom there is at least one department headed by them. This DeptHead concept represents the heads of all the departments in the hotel.

Only a customer can make a reservation. Also a person has to have at least one reservation at the hotel to be considered a customer. Therefore, equivalence is appropriate in T.24.

In T.4, T.18 and T.23, the same Address concept is used. This is because an address, irrespective of whether it is of a hotel, employee or customer, has the same structure and set of possible values. Hence, there is no need to differentiate between them by creating separate concepts for Hotel Address, Employee Address etc.

Two separate concepts for RoomCost and ServiceCost have been used instead of a single Cost concept. This is because the possible values for RoomCost tend to be a small set with likely higher values than Service Cost. Hence, separating them makes the representation clearer.