

Aditya Kumar & Aditya Jadhav

CSC 460

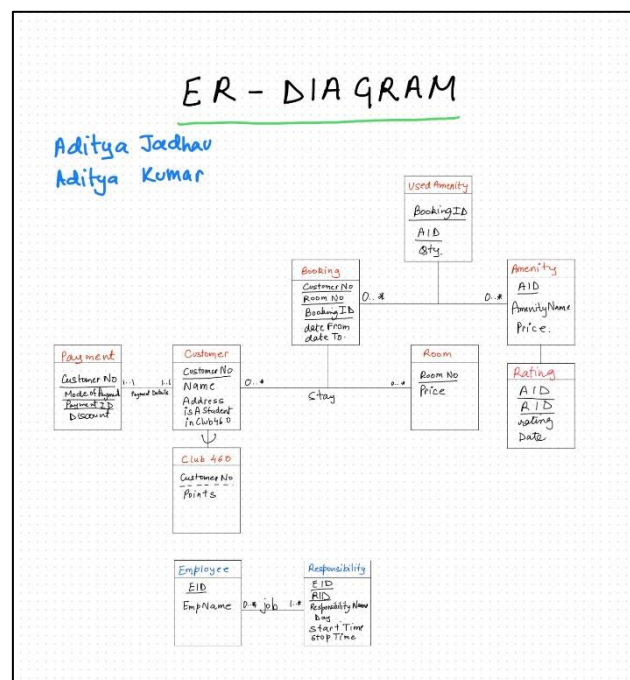
May 03, 2020

Dr. Lester McCann

## FINAL PROJECT: MOTEL 460

### REQUIRED EXTERNAL DOCUMENTATION

#### 1. Conceptual Database Design



Here are some important details that couldn't be depicted in the E-E-R diagram:

- The *ModesOfPayment* field in the Payment relation can have the following values: 'CC (Credit Card)', 'Cash', 'DD(Demand Draft)' and 'Cheque.'
- *IsAStudent* and *InClub460* contain Boolean values in Customer.
- *dateFrom* can't be > greater than *dateTo* in Booking.
- Double-Booking is not permitted.
- *startTime* must be < *stopTime* in Responsibility.
- *BookingID* and *ResponsibilityID* are permitted to have duplicate values, though it is unlikely. (BookingID can duplicate for members of same family)
- Unlike other IDs, *PaymentID* is just the check number, DD or credit card number, and can be null if the payment is made using cash.

## 2. Logical Database Design

customer (customerno, name, address, student, club460)  
club460 (customerno, points)  
payment (customerno, modeofpayment, paymentid, discount)  
room (roomno, price)  
booking (customerno, roomno, bookingid, datefrom, dateto)  
amenity (amenityid, name, price)  
usedamenity (bookingid, aid, qty)  
rating (amenityid, ratingid, rating, date)  
employee (employeeid, name, dob)  
responsibility (employeeid, respid, title, starttime, stoptime, day)

## 3. Normalization Analysis

Since none of the attributes in any of the relations are set-valued, all the tables are in **First Normal Form**.

### **List of functional dependencies for each table:**

1. *Customer*: customerno  $\rightarrow$  name, customerno  $\rightarrow$  address, customerno  $\rightarrow$  student, customerno  $\rightarrow$  club460
2. *Club460*: customerno  $\rightarrow$  points
3. *Payment*: {customerno, modeofpayment, paymentid}  $\rightarrow$  discount
4. *Room*: roomno  $\rightarrow$  price
5. *Booking*: {customerno, roomno, bookingid}  $\rightarrow$  dateFrom, {customerno, roomno, bookingid}  $\rightarrow$  dateTo
6. *Amenity*: amenityid  $\rightarrow$  name, amenityid  $\rightarrow$  price
7. *UsedAmenity*: {bookingid, aid}  $\rightarrow$  qty
8. *Rating*: {amenityid, ratingid}  $\rightarrow$  rating, {amenityid, ratingid}  $\rightarrow$  date
9. *Employee*: employeeid  $\rightarrow$  name, employeeid  $\rightarrow$  dob
10. *Responsibility*: {employeeid, respid}  $\rightarrow$  title, {employeeid, respid}  $\rightarrow$  startitme, {employeeid, respid}  $\rightarrow$  stoptime, {employeeid, respid}  $\rightarrow$  day

Other than the primary keys chosen, there are no candidate keys in any of the relations.

Since there are no extra candidate keys, and all the non-prime attributes in every table are fully functionally dependent on the primary keys, all the relations are in **Second Normal Form**.

Since for all the dependencies  $X \rightarrow Y$  of every table,  $X$  is a super key, the relations are also in **Third Normal Form**.

#### 4. Query Description

Our custom query returns the employees having a responsibility with the responsibility name/title supplied by the user.

This would help the user to figure out which employees have a particular responsibility such as cleaning, cooking etc., and such information can influence future employment decisions.