DATABASE SYSTEMS

GROUP 3

COURSEWORK 1
DESIGN

Contents

Section	Page Number
Application Chosen	3
List of all Assumptions	3
Conceptual Schema (ER diagram)	4
Translation of Conceptual into Relational Schema	5
Final Relational Schema	6
Diagram of Relational Schema	6
Normalisation	7
Explanation of 4NF	9

Application chosen: Hotel System

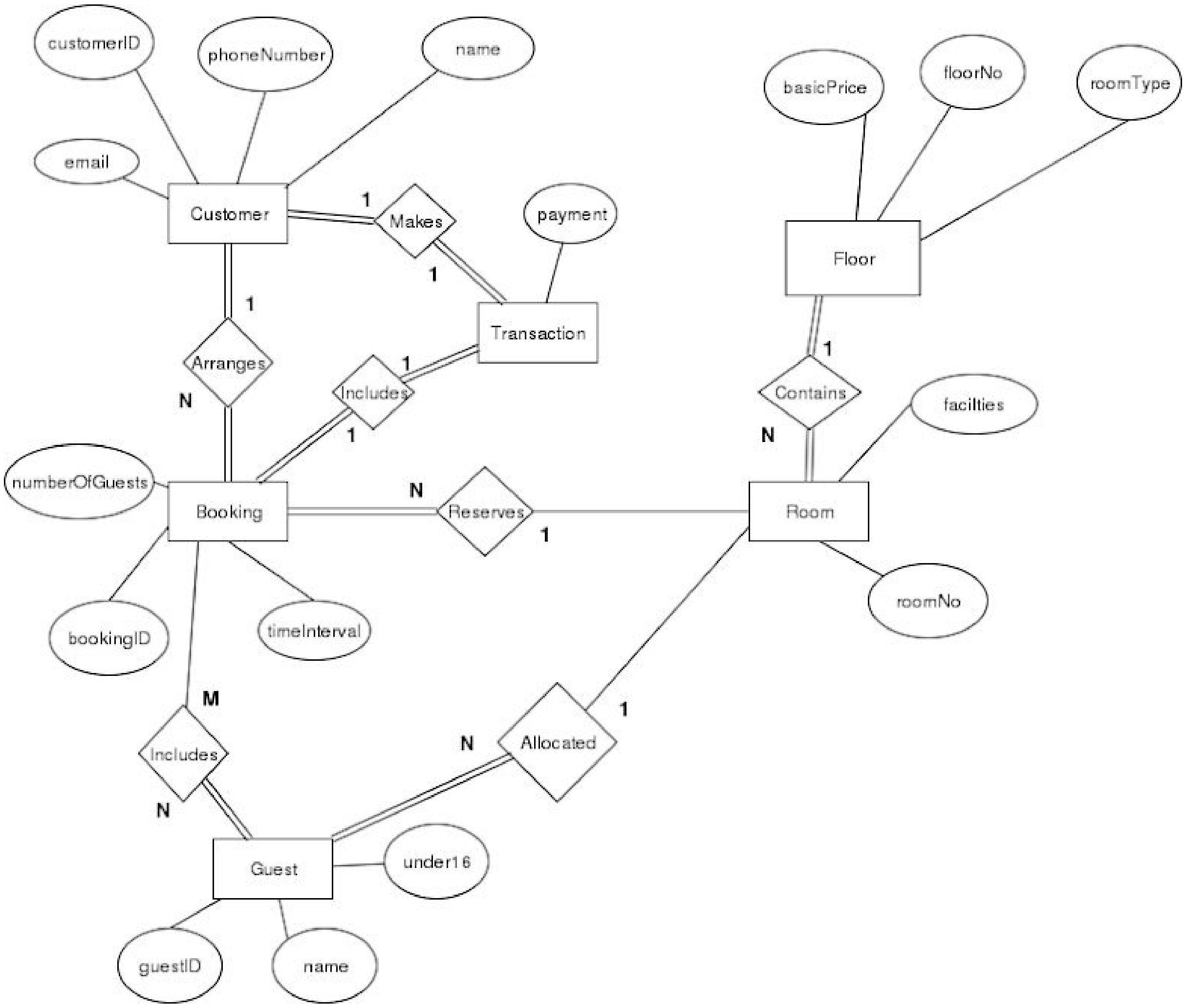
A hotel has rooms of three types: excellent, deluxe and magnificent. It has several rooms of each of these types, but all rooms of a particular type are priced the same and have the same facilities. Each room is situated on only one floor of the hotel, and a room is identified by the floor number followed by another two-digit number. Customers may book accommodation for a particular type of room for a single period or may make several bookings for several visits to the hotel. For each booking they would specify the names of the people who would be staying.

Thus a person may wish to make a booking for themselves alone, or with a friend or possibly with a friend and child. In any case, no more than three people may share a room. The hotel assigns these people to an appropriate available room. Each type of room has a basic price and the cost of the room with several guests is calculated from this basic price, the type of room, and the number of guests sharing the room. Your system must be able to support the storage of the above information, including being able to track the guests included on a particular booking, and the guests allocated to each hotel room.

List of all Assumptions

- There are only three types of rooms.
- Each type has several rooms that are available.
- All rooms for a specific type are the same price and have the same facilities available.
- There are 3 floors, each floor only contains rooms of a specific type.
- Each room is identified by the floor number followed by a two-digit number.
- Customers can make several bookings of any room type of their choice.
- Each booking will record the names of people that are staying (Customer and Guests).
- All rooms have a capacity of upto three people.
- Rooms can not be booked during their reserved period.
- All rooms will have a basic price and the booking cost is determined by the number of guests and the type of room.
- Adults and children are priced differently.
- Guests can be booked in several different bookings.

Conceptual Schema: Entity-Relationship Diagram



Translation of Conceptual Schema (ER Diagram) to Relational Schema

Formatting key:
PrimaryKey *ForeignKey* **CompositeKey** Changes made from previous NF *Relationship/Attribute being considered*

Binary 1-1 Relationships

Foreign key in one relation to map relationship

Customer(**customerID**, name, phoneNumber, email)

Transaction(**transactionID**, payment, *customerID*)
Booking(**bookingID**, timeInterval, numberOfGuests, *transactionID*)

Binary 1-N Relationships

Include a foreign key on the N side of the relationship

Booking(**bookingID**, timeInterval, numberOfGuests, *customerID, transactionID*)

Customer-Booking

Booking(**bookingID**, timeInterval, numberOfGuests, *customerID*, *roomNo, transactionID*)

Booking-Room

Floor(**floorNo**, roomType, basicPrice)

Room(**roomNo**, facilities, *floorNo*) *Room-Floor*

Guest(**guestID**, name, under16)

To meet specification

The room number is the floor number followed by the door number

Room(**roomNo**, ***floorNo***, facilities)

Booking(**bookingID**, timeInterval, numberOfGuests, *customerID*, *roomNo, floorNo, transactionID*)

Binary M-N Relationships

New relation that contained both the primary keys from each entity and any additional attributes.

GuestBooking(***bookingID***, ***guestID***) *Booking-Guest*

Final Relational Schema

Customer(**customerID**, name, phoneNumber, email)

Transaction(**transactionID**, payment, *customerID*)

Booking(**bookingID**, timeInterval, numberOfGuests, *customerID*, *roomNo, floorNo, transactionID*)

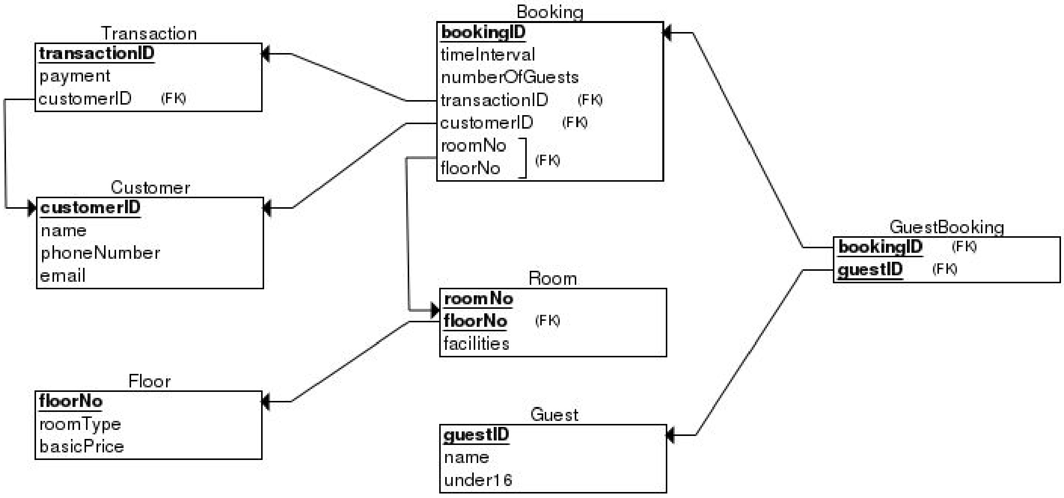
Floor(**floorNo**, roomType, basicPrice)

Room(**roomNo**, ***floorNo***, facilities)

Guest(**guestID**, name, under16)

GuestBooking(***bookingID***, ***guestID***)

Diagram of Relational Schema



Normalisation

1NF - Atomic values

Customer(**customerID**, firstname, surname, phoneNumber, email) *name*
Booking(**bookingID**, arrivalDate, departureDate, numberOfAdults, numberOfChilds, *customerID, roomNo, floorNo, transactionID*) *numberOfGuests*
Transaction(**transactionID**, amount, currency, date, *customerID*) *payment*
Room(**roomNo**, ***floorNo***, *facilityID*) *facilities*
Facility(**faciltyID**, numberOfBeds, bedType, miniFridge)
Floor(**floorNo**, roomType, basicPrice)
Guest(**guestID**, firstname, surname, under16) *name*
GuestBooking(***bookingID, guestID***)

Changes Made

The name columns in Customer and Guest can consist of '*firstname*' and '*surname*' which indicates multi-value and therefore requires changing. The '*timeInterval*' is also made up of the start and end dates of a booking and in order to calculate the prices for adults and children, '*numberOfGuests*' needed to be split up. The '*facilities*' attribute consisted of several values, it made sense to create a seperate relation called '*Facility*' and hold a foreign key in Room instead.

2NF - Non-key columns dependent on Primary Key

Customer(**customerID**, firstname, surname, phoneNumber, email)
Booking(**bookingID**, arrivalDate, departureDate, numberOfAdults, numberOfChilds, *customerID, roomNo, floorNo, transactionID*)
Transaction(**transactionID**, amount, currency, date, *customerID*)
Room(**roomNo**, ***floorNo***, *facilityID*)
Facility(**faciltyID**, numberOfBeds, bedType, miniFridge)
Floor(**floorNo**, roomType, basicPrice)
Guest(**guestID**, firstname, surname, under16)
GuestBooking(***bookingID, guestID***)

Changes Made

No changes were made as all the non-key columns were fully functionally dependent on the determinants (primary key).

3NF - Columns non-transitively dependent

Customer(**customerID**, firstname, surname, phoneNumber, email)

Booking(**bookingID**, arrivalDate, departureDate, numberOfAdults, numberOfChilds, *customerID, roomNo, floorNo, transactionID*)

Transaction(**transactionID**, amount, currency, date, *customerID*)

Room(**roomNo**, ***floorNo***, *facilityID*)

Facility(**faciltyID**, numberOfBeds, bedType, miniFridge)

Floor(**floorNo**, *roomType*) *basicPrice*

RoomPrice(**roomType**, basicPrice)

Guest(**guestID**, firstname, surname, under16)

GuestBooking(***bookingID, guestID***)

Changes Made

The '*basicPrice*' attribute was dependent on '*roomType*' which depended on '*floorID*', which meant it was transitively dependent. To solve this a new relation called '*RoomPrice*' was created that made '*roomType*' a primary key that '*basicPrice*' relied on and also a foreign key in '*Floor*'.

BCNF - Boyce - Codd Normal Form (Every determinant is a candidate key)

Customer(**customerID**, firstname, surname, phoneNumber, email)

Booking(**bookingID**, arrivalDate, departureDate, numberOfAdults, numberOfChilds, *customerID, roomNo, floorNo, transactionID*)

Transaction(**transactionID**, amount, currency, date, *customerID*)

Room(**roomNo**, ***floorNo***, *facilityID*)

Facility(**faciltyID**, *numberOfBeds*, miniFridge) *bedType*

NoB(**numberOfBeds**, bedType)

Floor(**floorNo**, *roomType*)

RoomPrice(**roomType**, basicPrice)

Guest(**guestID**, firstname, surname, under16)

GuestBooking(***bookingID, guestID***)

Changes Made

'*bedType*' was determined by the '*numberOfBeds*' attribute so a new relation called '*NoB*' that has '*numberOfBeds*' as primary key is created. Every room has the capacity to accommodate 3 people and therefore the '*NoB*' relation will only have 2 columns, where 2 beds mean that a double bed is included and 3 beds mean only single beds.

4NF - BCNF containing no non-trivial MVDs

(mvd non trivial - A and B union not entire relation or B not subset of A).

Customer(**customerID**, firstname, surname, phoneNumber, email)
Booking(**bookingID**, arrivalDate, departureDate, numberOfAdults, numberOfChilds, *customerID, roomNo, floorNo, transactionID*)
Transaction(**transactionID**, amount, currency, date, *customerID*)
Room(**roomNo**, ***floorNo***, *facilityID*)
Facility(**faciltyID**, *numberOfBeds*, miniFridge)
NoB(**numberOfBeds**, bedType)
Floor(**floorNo**, *roomType*)
RoomPrice(**roomType**, basicPrice)
Guest(**guestID**, firstname, surname, under16)
GuestBooking(***bookingID, guestID***)

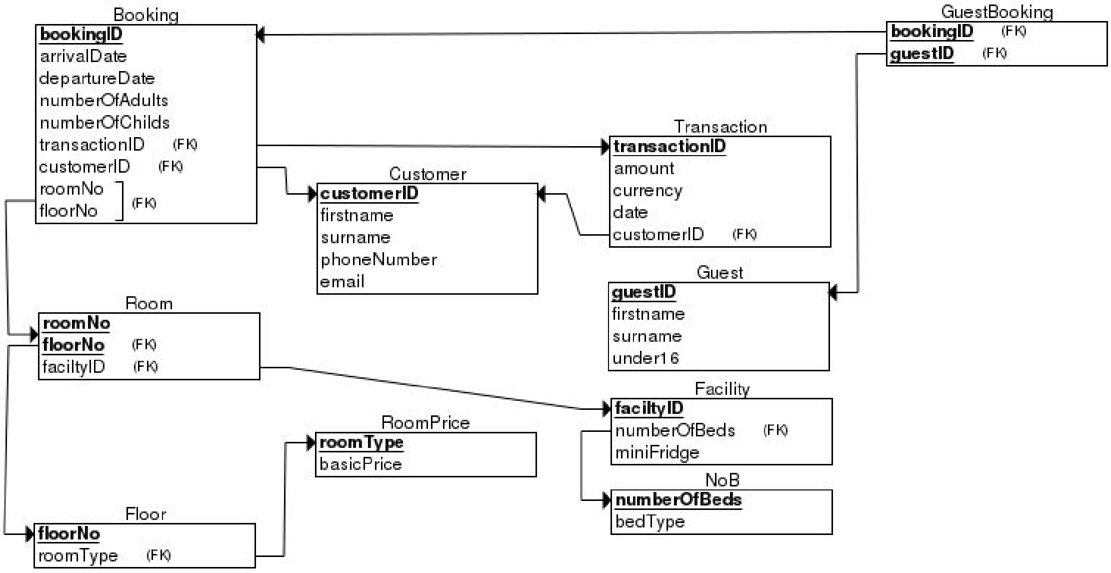
Changes Made

No changes were made from BCNF as the multi-valued dependencies were trivial.

Explanation for 4NF

The final, normalised schema is complete in 4th Normal Form as it has met the criteria of 3NF, which inherently means that 2NF and 1NF have been accomplished, and the schema no longer contains any trivial multivalued dependencies. So without any non-trivial MVDs left to remove the schema is in 4NF.

4NF Relational Schema Diagram


Booking
bookingID
arrivalDate
departureDate
numberOfAdults
numberOfChilds
transactionID (FK)
customerID (FK)
roomNo
floorNo
(FK)
GuestBooking
bookingID (FK)
guestID (FK)
Transaction
transactionID
amount
currency
date
customerID (FK)
Customer
customerID
firstname
surname
phoneNumber
email
Guest
guestID
firstname
surname
under16
Room
roomNo
floorNo (FK)
faciltyID (FK)
Facility
faciltyID
numberOfBeds (FK)
miniFridge
RoomPrice
roomType
basicPrice
NoB
numberOfBeds
bedType
Floor
floorNo
roomType (FK)