**Hotel Management System**

Semester Project Proposal

for

CSC 263 Database Systems

October 14, 2020

**Team Moody**

**Names of the team members:**

Steven Moody

Submitted to —

Dr. Bo Hatfield

**Contents**

Executive Summary 3

Problem Descriptions 4

Identifying Needs of Customers

Identifying Target Specifications

Potential Database Queries

Conceptual Design 5

Identify entities

Identify relationships

Identify constraints

Entity-Relationship Diagram

Logical Design 7

Database Schema

List of constraints that match with the schema

Attachment: Scripts that create the database and establish the database constraints 8

**Note:** “x” here indicates the appropriate page number for each section or subsection.

**Executive Summary**

The main reason for creating a Hotel Management System is simple, this is because it demonstrates a logical need that most clients would likely have when needing a database created which is some sort of management device. I feel that a hotel management system is simple enough to accomplish in one semester but complex enough to offer a variety of query options and database creation potential.

As for my conceptual design I chose to create 5 major entities each with a series of attributes that communicate well so as to be able to relate the tables accordingly. These entities are: Hotel, Employee, Customer, Rooms, and billing. Most of the constraints are relatively standard with most of my domain constraints ensuring proper entry types as well as various entity constraints as well as referential constraints as demonstrated in the logical design portion.

**Problem Descriptions**

Although my customer is imaginary, or in this case you the professor I wanted to provide a project that would be something I could see myself developing in the future in the real world. I chose to build a hotel management system for that very reason. Many companies use databases to track various amounts of information but in my opinion, likely few NEED them in comparison to a hotel chain.

A hotel chain relies almost solely on accurate databases creation and the ability to query and manage anything from customer needs through to billing adjustments and ensuring they are not double booking rooms.

In this section, you need to concentrate on accomplishing all of the following tasks for your project. You need to base your discussions on so called “user stories”.

* My customers will be myself and you(Bo Hatfield).
* Any good hotel management system should be able to do the following
  + Book rooms
  + Track customers by arrival and departure as well as the rooms occupied by them
  + Track employees
  + Track reservations and their associated costs
* The client should be able to perform the following queries
  + Search for available rooms
  + Search for a customer and identify what rooms the customer has booked
  + Show customer arrival/departure dates
  + Search for available rooms in other hotels
  + Search basic employee information

**Conceptual Design**

My entities are as follows:

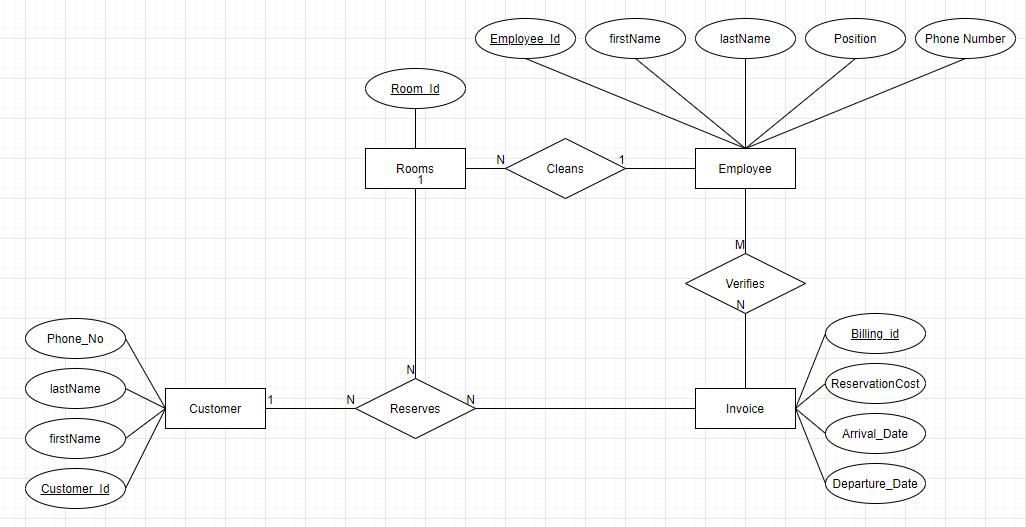
* Hotel- this is fairly obvious and stands for each hotel location
* Employee- Every hotel has to have employees staffed to ensure rooms are good to go and should be tracked within a table
* Customer- No hotel would remain in business without customers, and these customers need to be kept organized within a database
* Rooms- Every hotel has a large number of rooms and should be able to be tracked
* Billing-Billing is vital to a hotel functioning and should include information on customers, their arrival/departure as well as the rooms they booked and the rooms available

My relationships between my entities are as follows

* Hotels have employees- as explained above a hotel cannot function without employees being associated to it but employees can only work at one hotel
* Hotels have rooms-every hotel has a number of rooms but those rooms are specific to the hotel
* Customer Stays in hotels- Every hotel can have many customers but a customer can only book at one hotel
* Customer Stays in rooms- rooms can only be associated with one customer but a customer can book many rooms
* Rooms are reserved through billing-billing tracks all rooms and customers at a given hotel

The constraints I have so far are

* Employee
  + Employee\_Id (Primary Key, Unique Not Null)
  + firstName (Not Null)
  + lastName (Not Null)
  + Position (Not Null)
  + Phone Number(Not Null)
* Invoice
  + Billing\_Id (Primary Key, Unique Not Null)
  + Reservationcost (Not Null)
  + Arrival\_Date (Not Null)
  + Departure\_Date (Not Null)
* Verifies
  + Employee\_Id (Primary Key, Foreign Key, Unique Not Null)
  + Billing\_Id (Primary Key, Foreign Key, Unique Not Null)
* Rooms
  + Room\_Id (Primary Key, Unique, Not Null)
* Cleans
  + Room\_Id (Primary Key, Foreign Key, Not Null)
  + Employee\_Id (Primary Key, Foreign Key, Not Null)
* Customer
  + Customer\_Id(Primary Key, Unique, Not Null)
  + firstName (Not Null)
  + lastName (Not Null)
  + Phone Number (Not Null)
* Reserves
  + Customer\_Id(Primary Key, Foreign Key, Unique, Not Null)
  + Room\_Id (Primary Key, Foreign Key, Unique, Not Null)
  + Billing\_Id (Primary Key, Foreign Key, Unique Not Null)



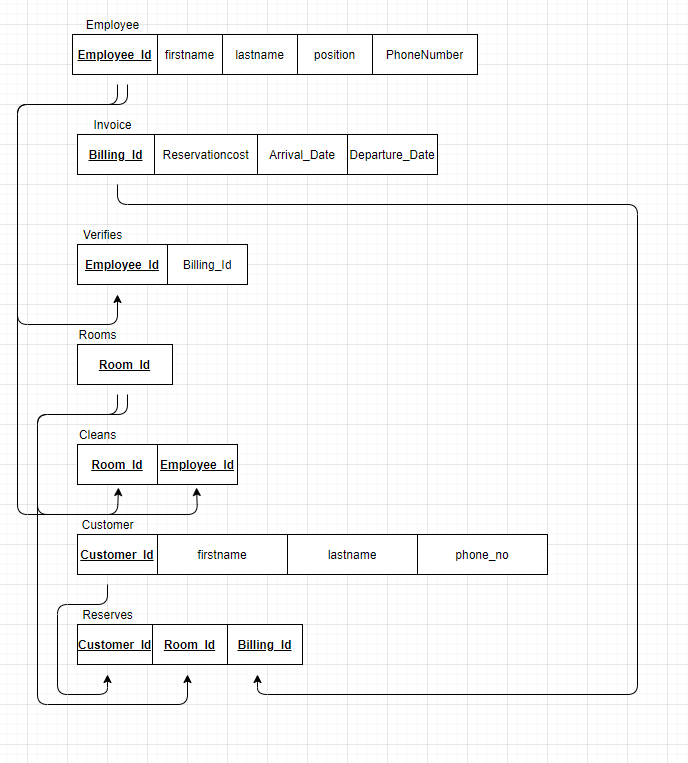
**Logical Design**

In this section you report the results of the steps in the Logical Design phase of the database design that the team performed. In detail, report the following tasks you performed.

* Design of the database schema.
* The final list of constraints that closely match the database schema.

To show the database schema for your database, you can use the representations we discussed in class. Or, you can show the schema in the form of a Data Dictionary. Please use Table 3.6 on page 89 of the text as an example of such a data dictionary. To read about the definition of data dictionary, please refer the text on page 88 in the text.

Note that this step of the database design directly affects the creation of your database in the next step. Therefore, you must define ALL necessary pieces of your database, i.e., entities, attributes, relationships, constraints (all types of constraints needed).



**Database Definition (by producing a script file)**

In this section you report the generation of the script that defines the database. Your script must be generated by using the result of the previous Logical Design. The script generated for your database can be run to generate all necessary structures for your database in MySQL DBMS. However, the script only contains the needed DDL to create the database structure. The following is a check list of what are required to be included in the script.

* Create all tables for the database, including all attributes for each table.
* Define and create all constraints necessary for the database.
* Define constraints needed by business rules.
* DO NOT populate each table.
* You must show the contents of the script here.

drop table if exists `hotel\_Employee`;

drop table if exists `invoice`;

drop table if exists `verifies`;

drop table if exists `rooms`;

drop table if exists `cleans`;

drop table if exists `customer`;

drop table if exists `reserves`;

create table `hotel\_Employee`(

`Employee\_Id` int(10) not null,

`firstname` varchar(30) not null,

`lastname` varchar(30) not null,

`position` varchar(30) not null,

`PhoneNumber` bigint(10) not null,

PRIMARY KEY (`Employee\_Id`)

);

create table `hotel\_Invoice`(

`Billing\_Id` int(4) not null,

`Reservationcost` varchar(30) not null,

`Arrival\_Date` varchar(30) not null,

`Departure\_Date` varchar(30) not null,

PRIMARY KEY (`Billing\_Id`)

);

create table `hotel\_Verifies`(

`Employee\_Id` int(10) not null,

`Billing\_Id` int(4) not null,

PRIMARY KEY (`Billing\_Id`,`Employee\_Id`)

);

create table `hotel\_Rooms`(

`Room\_Id` int(3) not null,

PRIMARY KEY(`Room\_id`)

);

create table `hotel\_Cleans`(

`Room\_Id` int(3) not null,

`Employee\_Id` int(10),

PRIMARY KEY(`Room\_Id`, `Employee\_Id`)

);

create table `hotel\_Customer`(

`Customer\_Id` int(4) not null,

`firstname` varchar(50)not null,

`lastname` varchar(50)not null,

`phone\_no` bigint(10)not null,

PRIMARY KEY(`Customer\_Id`)

);

create table `hotel\_Reserves`(

`Customer\_Id` int(4) not null,

`Room\_Id` int(3) not null,

`Billing\_Id` int(4) not null,

PRIMARY KEY(`Customer\_Id`, `Room\_Id`, `Billing\_Id`)

);

ALTER TABLE `hotel\_Cleans`

ADD CONSTRAINT foreign key (`Room\_Id`) references `hotel\_Rooms`(`Room\_Id`);

ALTER TABLE `hotel\_Cleans`

ADD CONSTRAINT foreign key (`Employee\_Id`) references `hotel\_Employee`(`Employee\_Id`);

ALTER TABLE `hotel\_Reserves` ADD CONSTRAINT

foreign key (`Customer\_Id`) references `hotel\_Customer`(`Customer\_Id`);

ALTER TABLE `hotel\_Reserves` ADD CONSTRAINT

foreign key (`Room\_Id`) references `hotel\_Rooms`(`Room\_Id`);

ALTER TABLE `hotel\_Reserves` ADD CONSTRAINT

foreign key (`Billing\_Id`) references `hotel\_Invoice`(`Billing\_Id`);