Booking\_hotel\_room.com

Sylwia Kmiec

***This report, together with the SQL script shows the steps taken to design, write and implement database of real web-application.***

***For this project I have chosen application: Booking\_hotel\_room.com.***

# Table of Contents

**TABLE OF CONTENTS 1**

**I.** **SUBMITTED FILES 1**

**II.** **REQUIREMENTS ANALYSIS 1**

A. Subsection 1

**III.** **ERD 2**

A. Explanation of Relationships 2

**IV.** **RELATIONAL MODEL 2**

**V.** **DESCRIPTION OF THE IMPLEMENTATION 2**

**VI.** **REFLECTION OF APPROACH 2**

# Submitted Files

**Table 1. Submitted Files**

|  |  |
| --- | --- |
| ***Name*** | ***Description*** |
| *Project\_Report* | *Report of the project* |
| *Project SQL script* | *All performed steps in MySQL* |
| *ERD\_UML* | *Excel with ER diagram and UML* |

# Requirements Analysis

The application “Booking\_hotel\_room.com” has been created to allow customers to search, view and book rooms at hotels all around the globe. Once the booking of the rooms is completed customers making payments via that application to the Hotel that has been reserved.

In my project I have modelled only the part that is related to the booking/reservation process. I have omitted the part related to placing payments and receiving confirmation of the booking.

*Subsection*

Short summary of the process:

Customer is making reservation of the room by placing booking/order. The reservation indicates which room has been reserved and for how long. In my model in one reservation, we are not allow to book multiple rooms. Each room booking needs to have separate reservation. Rooms belongs to the Hotel that owns them, rooms also have Photos attached to them. The hotels have reviews sections, reviews are made by customers.

# ERD

# 

## Explanation of Relationships

**Customer <-> Reservation is many to many** – one customer can have many reservations, reservation can be done by many customers

**Customer <-> Address, one to one** – customer has one address, address has one customer

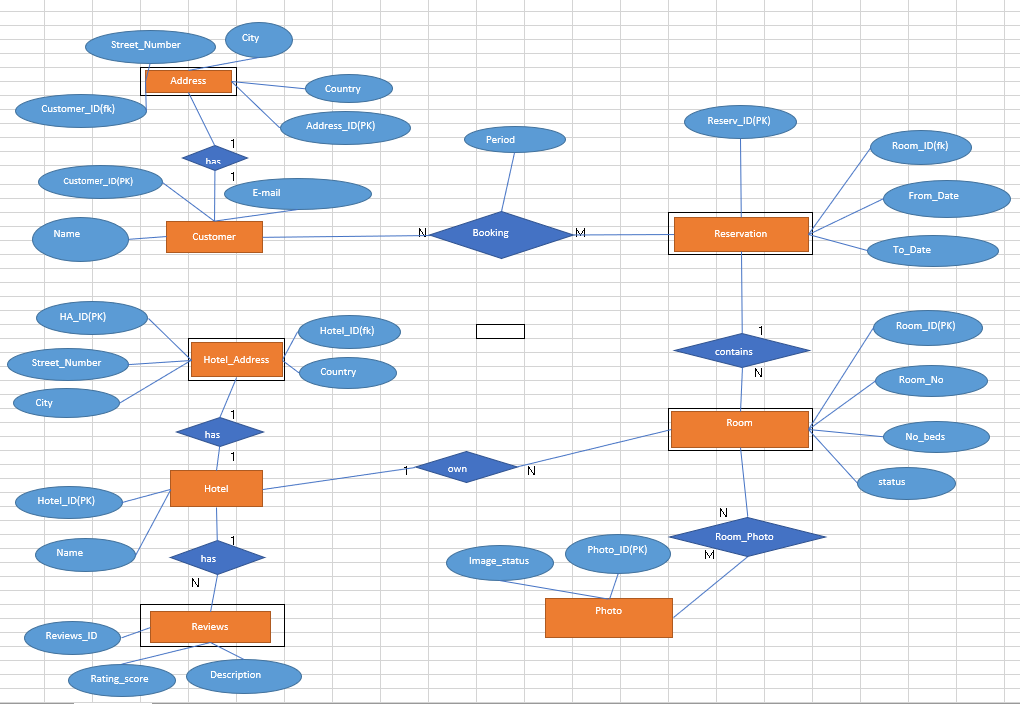
**Reservation <-> Room is one to many** – reservation is done for one room only, room may be booked by many reservations, (this could potentially be many to many relationship as in practice one customer by placing one reservation is able to reserve multiple number of rooms but for this project I wanted to avoid this and limited that option)

**Room <-> Photo, many to many –** room may have many photos, photos are done for many rooms

**Hotel <-> Room, one to many** – hotel may have many rooms but rooms maybe in one hotel

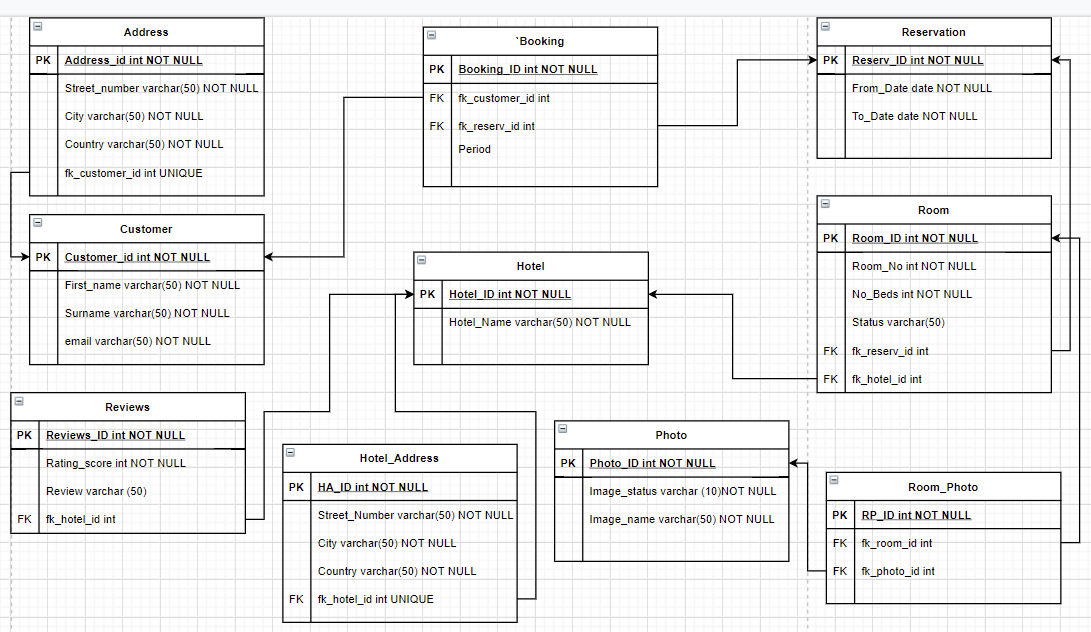
**Hotel <-> Reviews one to many** – hotel may have many reviews, reviews are for one hotel

**Hotel <-> Hotel\_Address – one to one** – hotel may have one address, address may be for one hotel



# Relational Model

I found this UML class diagram very informative, by looking at it I was able to properly create tables with correct attribute and keys.



# Description of the Implementation

After creating ER diagram, and then based on the UML I have Created all the tables in the Database with the needed attributes and assigned primary and foreign keys.

I have also inserted sample of data into those tables.

Once all of that performed, I have created several queries of the values from these tables.

**Joins:**

First join has been created to list customers with country of Russia in their address.

Second join has been created to list Name of hotels which consist ‘Hotel’ in the name with their location country and ordered in descending order.

Third join is listing Photo name of the room, room number and Hotel and country of location of that hotel.

Fourth join is listing Hotels who got rating score of 4 and the review description.

**Views:**

To be able to see room status and pictures linked to the room I have created View roomstatus.

Second view is for Hotel name and the review of that hotel.

Another view is showing all information about customer and the reservation he has done such: number of days he has booked and the dates of that reservation.

**Procedures:**

The first procedure called ‘daysreserved’ is listing Reservation ID altogether with Number of days that are booked by that reservation.

The second procedure is is listing Hotel names limited to those that have ‘hotel’ in the name and their addresses from the address table.

**Functions:**

This function is supposed to show number of reservations done by particular customer, the parameters for the function are Customer First name and Surname, the output is INT – number of reservations done.

**Triggers:**

When there will be new entry into Reservation table with new dates, the table booking will get updated with new period calculation which is based on the formula DATEDIFF().

The challenge here for me was that it the trigger was supposed to also populate the foreign keys from customer table and reservation table. I have managed to modify the trigger the way that it is populating foreign key from Reservation table. But was not able to link it to the third table Customer. Customer ID would need to get manually updated each time new reservation entered.