

The King

William Hotel

*By:*

Mathew Migliore, Tyler Segovia, and Kevin Romero

*Presented to:*

Basil J. Cleese and The Development Team

*of:*

The King William Hotel

*Date:*

2020-11-26

Contents

[Introduction 2](#_Toc57219123)

[Mission Statement 3](#_Toc57219124)

[Objectives 4](#_Toc57219125)

[User View Preface 5](#_Toc57219126)

[Low access: 5](#_Toc57219127)

[High access: 5](#_Toc57219128)

[Full access: 5](#_Toc57219129)

[User View 6](#_Toc57219130)

[System Boundary Preface 7](#_Toc57219131)

[System Boundary 8](#_Toc57219132)

[3NF Preface 9](#_Toc57219133)

[3NF Diagram 9](#_Toc57219134)

[Final Word 10](#_Toc57219135)

# Introduction

The King William Hotel, a historic much-loved landmark built in 1923. It serves the community of Ontario and offers guests hospitality and services that makes the guest want to come back. Due to the hotel’s older age and transition to the modern world, the hotel has renovated it’s rooms to look part of an early 20th era, Although the rooms have changed, the hospitality and service remain the same. This transition to the modern world and its perpetual top of the line service has come at cost. Although the staff maintain a great system to book the hotel’s information, it is now time for their system to adopt a new modern take, not only to facilitate but to also improve their service. With this Report, our team will be doing our best to show case the many different approaches we have taken to meet the client’s requests with as much accuracy as possible and to present the application we have created. Without further delay, here is our solution to The King William case.

# MISSION STATEMENT

The purpose of the King Williams Database Application is to store and manage information to improve efficiency and control of hotel processes.

# OBJECTIVES

- To maintain (enter, update, delete) data on rooms

- To maintain (enter, update, delete) data on room reservations

- To maintain (enter, update, delete) data on customers

- To maintain (enter, update, delete) data on customer billing

- To maintain (enter, update, delete) data on chargeable items

- To maintain (enter, update, delete) data on transactions

- To maintain (enter, update, delete) data on employees

- To maintain (enter, update, delete) data on employment positions

- To perform searches on rooms

- To perform searches on room reservations

- To perform searches on customers

- To perform searches on customer billing

- To perform searches on chargeable items

- To perform searches on transactions

- To perform searches on employees

- To perform searches on employment positions

- To track the status of rooms

- To track the status of customer bills

- To track the status of customer requests

- To report on room reservations

- To report on customers

- To report on customer billing

- To report on chargeable items

- To report on transactions

- To report on employees

# User View Preface

For the user views we took into consideration the employees that would access the system and what each employee should or should not be able to see and/or do while logged in. We managed to break the system down into 6 categories ranging from low to full access of the system.

## Low access:

This is a split between Kitchen Staff, Room Cleaning, Accounting & Stock, and Front Desk. They are each able to view, update, and edit certain parts of the Users information, but are not able to do everything for the users. They are also unable to edit the information for themselves or any other higher-level user.

## High access:

Management has access over all the previous users, can edit, update, and the records, but do not have full access. They cannot delete records.

## Full access:

Admin has full access over everything. They can edit, update, and delete all records and all users.

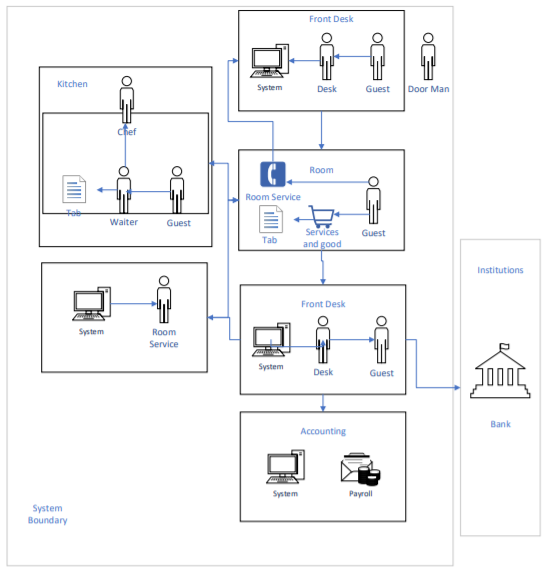
# User View



# System Boundary Preface

For the System Boundary we’ve took a lot of inspiration from the user view and mostly wanted to focus on the six major groups and their basic operations within the system that they would go through when creating, viewing, and modifying any information from the system. Once we had that we thought about other external groups that interacted with those functions.

# SYSTEM BOUNDARY



# 3NF Preface

For the data normalization our group took a different approach than the standard, instead of starting with the list, we wanted to create the graph so we could envision how the data would really work and how we could organize it in a way that would be programmatically easy to insert, create and update tables. Once the graph was completed, we translated that to the regular 3NF list, which helped us find more logical errors that we may have missed in the graph.

# 3NF Diagram

reservations(reservationID, roomNumber, customerID, billID, numberOfGuests, startDate, endDate, notes)

rooms(roomNumber, floorNumber, roomTypeID, statusID, notes)

roomStatus(statusID, statusDescription)

roomTypes(roomTypeID, roomType, pricePerNight)

customers(customerID, firstName, lastName, phoneNumber, addressID)

addresses(addressID, addressLine1, addressLine2, city, provinceID, country, postalCode)

provinces(provinceID, provinceCode, provinceName)

customerBilling(billID, billAmount, reservationID, paymentType, amountOwing)

chargeableItems(itemID, itemName, itemDescription, itemPrice)

transactions(transactionID, billID, itemID, amountOfItems, date)

staff(staffID, firstName, lastName, phoneNumber, addressID, positionID, salary, hiredDate, terminationDate, photo)

employmentPositions(positionID, positionTitle, positionDescription, startingSalary)

users(userID, username, password, roleID, staffID)

roles(roleID, roleTitle, roleDescription)

# Data Dictionary

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Table | Column | Data Type | References | Default | Not Null |
| <reservations> | reservationID | INT |  | Identity | Y |
| <reservations> | roomNumber | INT | <rooms>.roomNumber |  | Y |
| <reservations> | customerID | INT | <customers>.customerID |  | Y |
| <reservations> | billID | INT | <customerBilling>.billID |  | Y |
| <reservations> | numberOfGuests | INT |  |  | Y |
| <reservations> | startDate | DATETIME |  |  | Y |
| <reservations> | endDate | DATETIME |  |  | Y |
| <reservations> | notes | VARCHAR |  |  | N |
| <rooms> | roomNumber | INT |  | Identity | Y |
| <rooms> | floorNumber | INT |  |  | Y |
| <rooms> | roomTypeID | INT | <roomTypes>.roomTypeID |  | Y |
| <rooms> | statusID | INT | <roomStatus>.statusID |  | Y |
| <rooms> | notes | VARCHAR |  |  | N |
| <roomStatus> | statusID | INT |  | Identity | Y |
| <roomStatus> | statusDescription | VARCHAR |  |  | Y |
| <roomTypes> | roomTypeID | INT |  | Identity | Y |
| <roomTypes> | roomType | VARCHAR |  |  | Y |
| <roomTypes> | pricePerNight | MONEY |  |  | Y |
| <customers> | customerID | INT |  | Identity | Y |
| <customers> | firstName | VARCHAR |  |  | Y |
| <customers> | lastName | VARCHAR |  |  | Y |
| <customers> | phoneNumber | VARCHAR |  |  | Y |
| <customers> | addressID | INT | <addresses>.addressID |  | Y |
| <addresses> | addressID | INT |  | Identity | Y |
| <addresses> | addressLine1 | VARCHAR |  |  | Y |
| <addresses> | addressLine2 | VARCHAR |  |  | N |
| <addresses> | city | VARCHAR |  |  | Y |
| <addresses> | provinceID | INT | <provinces>.provinceID |  | Y |
| <addresses> | country | VARCHAR |  | ‘Canada’ | Y |
| <addresses> | postalCode | CHAR |  |  | Y |
| <provinces> | provinceID | INT |  | Identity |  |
| <provinces> | provinceCode | CHAR |  |  | Y |
| <provinces> | provinceName | VARCHAR |  |  | Y |
| <customerBilling> | billID | INT |  | Identity | Y |
| <customerBilling> | billAmount | MONEY |  |  | Y |
| <customerBilling> | reservationID | INT | <reservations>.reservationID |  | Y |
| <customerBilling> | paymentType | VARCHAR |  |  | Y |
| <customerBilling> | amountOwing | MONEY |  |  | Y |
| <chargeableItems> | itemID | INT |  | Identity | Y |
| <chargeableItems> | itemName | VARCHAR |  |  | Y |
| <chargeableItems> | itemDescription | VARCHAR |  |  | Y |
| <chargeableItems> | itemPrice | MONEY |  |  | Y |
| <transactions> | transactionID | INT |  | Identity | Y |
| <transactions> | billID | INT | <customerBilling>.billID |  | Y |
| <transactions> | itemID | INT | <chargeableItems>.itemID |  | Y |
| <transactions> | amountOfItems | INT |  |  | Y |
| <transactions> | date | DATETIME |  |  | Y |
| <staff> | staffID | INT |  | Identity | Y |
| <staff> | firstName | VARCHAR |  |  | Y |
| <staff> | lastName | VARCHAR |  |  | Y |
| <staff> | phoneNumber | VARCHAR |  |  | Y |
| <staff> | addressID | INT | <addresses>.addressID |  | Y |
| <staff> | positionID | INT | <employmentPositions>.positionID |  | Y |
| <staff> | salary | MONEY |  |  | Y |
| <staff> | hiredDate | DATE |  |  | Y |
| <staff> | terminationDate | DATE |  |  | N |
| <staff> | photo | VARCHAR |  |  | N |
| <employmentPositions> | positionID | INT |  | Identity | Y |
| <employmentPositions> | positionTitle | VARCHAR |  |  | Y |
| <employmentPositions> | positionDescription | VARCHAR |  |  | Y |
| <employmentPositions> | startingSalary | MONEY |  |  | Y |
| <users> | userID | INT |  | Identity | Y |
| <users> | username | VARCHAR |  |  | Y |
| <users> | password | VARCHAR |  |  | Y |
| <users> | roleID | INT | <roles>.roleID |  | Y |
| <users> | staffID | INT | <staff>.staffID |  | N |
| <roles> | roleID | INT |  | Identity | Y |
| <roles> | roleTitle | VARCHAR |  |  | Y |
| <roles> | roleDescription | VARCHAR |  |  | Y |

# FINAL WORD

We hope that the solutions that we have created above are exactly to your specifications and likings. We have taken into consideration every piece of information you have given us and concluded that our research and findings will provide you with an excellent application and database. If you do find anything you would like to change about our report or the application, please do not hesitate to contact us. Do not hesitate to provide any feedback on any of the information we have provided here today.

Thank you for choosing us, and we hope to hear back from you soon.