Temasek Polytechnic

School of Informatics and IT

**Diploma in Information Technology (IT)**

Assignment 2

**Project Particulars**

|  |  |
| --- | --- |
| **Tutor** | Mr Mel Goh |
| **Class** | P03 |
| **Project Title** | Delonix Regia Hotel Management System |

**Project Team’s Particulars**

|  |  |
| --- | --- |
| **Matric Number** | **Student Name** |
| **1405505G** | **Chia Yue Zhu Ameline** |
| **1400790J** | **Douglas Tan Jin Jun** |
| **1401279C** | **Muhammad Azrul B Mohd Sanif** |

**Table of Contents**

[**1. DISTRIBUTION OF WORKLOAD 4**](#_Toc451933823)

[**2. OVERVIEW OF REQUIREMENTS 5**](#_Toc451933824)

[**2.1 System Functions 5**](#_Toc451933827)

[**2.2 User Characteristics 6**](#_Toc451933828)

[**2.2.1 Physical User 6**](#_Toc451933829)

[**2.2.2 System User 9**](#_Toc451933830)

[**2.3 General Constraints 10**](#_Toc451933831)

[**2.4 Functional Requirements 10**](#_Toc451933832)

[**2.4.1 FR01: Login 12**](#_Toc451933833)

[**2.4.2 FR02: Booking and Reservation 13**](#_Toc451933834)

[**2.4.3 FR03: Search 15**](#_Toc451933835)

[**2.4.4 FR04: Payment 16**](#_Toc451933836)

[**2.4.5 FR05: Reports 17**](#_Toc451933837)

[**2.4.6 FR06: Room availability and Booking module 18**](#_Toc451933838)

[**2.4.7 FR07: User account and Login Creation module 19**](#_Toc451933839)

[**2.4.8 FR08: Preview 20**](#_Toc451933840)

[**2.4.9 FR09: Backup 21**](#_Toc451933841)

[**2.5 Data Requirements 22**](#_Toc451933842)

[**2.6 User Interface Requirements 24**](#_Toc451933843)

[**2.7 Interface with Other Systems 24**](#_Toc451933844)

[**2.8 Assumptions 25**](#_Toc451933845)

[**3 OPERATIONAL AND QUALITY REQUIREMENTS 26**](#_Toc451933846)

[**3.1 Operating Environment 26**](#_Toc451933847)

[**3.2 Development Constraints 26**](#_Toc451933848)

[**3.3 Performance 27**](#_Toc451933849)

[**3.4 Availability 28**](#_Toc451933850)

[**3.5 Security and Access Control Requirements 28**](#_Toc451933851)

[**4 SPECIAL REQUIREMENTS 29**](#_Toc451933852)

[**5 ARCHITECTURE DESIGN 29**](#_Toc451933853)

[**6 USER INTERFACE (UI) DESIGN 32**](#_Toc451933854)

[**6.1 Login 32**](#_Toc451933855)

[**6.2 Main page 32**](#_Toc451933856)

[**6.3 Room availability and Booking module 33**](#_Toc451933857)

[**6.4 Housekeeping and Staff management module 34**](#_Toc451933858)

[**6.5 Search 34**](#_Toc451933859)

[**7 PROGRAM DESIGN 34**](#_Toc451933860)

[**7.1 Description 34**](#_Toc451933861)

[**7.1.1 Use Case FR01: Login 34**](#_Toc451933862)

[**7.1.2 Use Case FR02 - Maintain Booking and Reservations 36**](#_Toc451933863)

[**7.1.3 Use Case FR03 - Search Guest 39**](#_Toc451933864)

[**7.1.4 Use Case FR04 - Receive Payments 42**](#_Toc451933865)

[**7.1.5 Use Case FR05 - Generate Reports 46**](#_Toc451933866)

[**7.1.6 Use Case FR06 - Maintain Room Availability and Booking 50**](#_Toc451933867)

[**7.1.7 Use Case FR07 - Maintain User Account and Login Creation 53**](#_Toc451933868)

[**7.1.8 Use Case FR08 - Print Preview of Report 58**](#_Toc451933869)

[**7.1.9 Use Case FR09 - Create Backup 60**](#_Toc451933870)

[**8 DATABASE DESIGN 62**](#_Toc451933871)

[**8.1 Database Schema 62**](#_Toc451933872)

[**8.2 ERD Diagram 63**](#_Toc451933873)

[**8.3 Data Dictionary 63**](#_Toc451933874)

[**8.3.1 Guest Table 63**](#_Toc451933875)

[**8.3.2 Staff Table 64**](#_Toc451933876)

[**8.3.3 Room Table 65**](#_Toc451933877)

[**8.3.4 Payment Table 65**](#_Toc451933878)

[**8.3.5 Login Table 65**](#_Toc451933879)

[**9 REFERENCES 66**](#_Toc451933880)

# DISTRIBUTION OF WORKLOAD

Software Specification Requirement: SSR

Software Design Specification: SDS

|  |  |
| --- | --- |
| **Requirement Gathering** | **Members** |
| 1) Distribution of workload (SSR and SDS) | Ameline |
| 2.1) System Functions (SSR) | Ameline |
| 2.2) User Characteristics (SSR) | Ameline |
| 2.3) General Constraints (SSR) | Azrul |
| 2.4) Functional Requirements (SSR) | Ameline |
| 2.5) Data Requirements (SSR) | Ameline |
| 2.6) User Interface Requirements (SSR) | Douglas |
| 2.7) Interface with Other Systems (SSR) | Azrul |
| 2.8) Assumptions (SSR) | Group |
| 3.1) Operating Environment (SSR) | Azrul |
| 3.2) Development Constraints (SSR) | Douglas |
| 3.3) Performance (SSR) | Douglas |
| 3.4) Availability (SSR) | Douglas |
| 3.5) Security and Access Control Requirements (SSR) | Azrul |
| 4) Special Requirements (SSR) | Douglas |
| 5) Architecture Design (SDS) | Ameline |
| 6) User Interface (UI) Design (SDS) | Ameline |
| 7) Program Design (SDS) | Azrul |
| 8) Database Design (SDS) | Douglas |
| 9) References (SSR and SDS) | Group |

# OVERVIEW OF REQUIREMENTS



## System Functions

The hotel management software system shall aid in the management and facilitation of the hotel in order to improve efficiency during check-in and check-out and provide a convenient platform for all the users to accomplish their tasks. Before users are able to fully utilize the software system, the system should allow all users to be able to log into their account which are registered in the user account and login creation module.

To meet all the user’s needs and ensure properly management in the hotel, the system shall allow users to check-in any guests when they first arrive at the hotel. Based on the information provided by the guest, the system shall capture the information and provide a clean vacant room for the guest to stay. In the information provided by the guest, it will not only state the guest’s personal details, but also their desired data and time to check out. So the system shall capture the check-out date and time of the guest, and allow users to check a guest out before 12pm. Furthermore, the system shall provide search functions for users to search for guest’s details and room information during check-out.

During check-out, the user will search for the guest record and proceed with payments. The system shall calculate the total charges the guest has accumulated over his/her stay in the hotel. The system shall generate a payment invoice that states the specific charges and its cost that was incur during the stay.

After the guest made his/her payments, the system shall also use the information provided to generate reports. The system should be able to generate four different modules according to their requirements. In addition, each of the module are also accessible to different types of users. The five reports are:

1. Room availability and booking module:

This report is accessible to all users. Under this report, the system shall allow the management users and administrators to edit the guest information as and when the guest arrive at the hotel.

1. Housekeeping and staff management module:

This report is accessible to the management users and administrators. Under this report, the system shall generate housekeeping schedule based on daily, weekly, and monthly basis.

1. Reporting module:

This report is accessible to the management users and administrators. Under this report, the system shall allow the users to input information pertaining to the room and the guest

1. User account and login creation module:

This report is accessible only to the administrators. Under this report, the system shall allow the administrator to create a new account for new staff and new users.

This report is accessible to all users. Under this report, the system shall generate statistics on the room occupancy based on a day-to-day, weekly, and monthly and yearly basis.

The system shall allow all reports to be previewed before it is printed out. This shall allow users to edit and update any information of the report. The system shall then backup all the data in the reports. In addition, all five reports must be exported to excel. Hence the system should allow all the reports to be export as a spreadsheet to excel.

In conclusion, all these functions can be used to make sure that all users are meeting all the hotel’s requirements and are providing good quality serves for the hotel.

## User Characteristics

### **Physical User**

There are three types of users that interact and use the software system. The users are: end-users, management users, and administrator users. Each of these three types of users has different use and privileges of the system, so each of them has their own requirements.

Firstly, one of the user that will be using the software system is the end-users. The end-users consist of employees who works at the receptions front desk. As the name implies, these employees are responsible for the reception area of the hotel, which is the first place guests go to when they arrive at the hotel. Duties of a receptionist include:

* Verifying of a guest’s reservations using the hotel management system
* Checking-in the guest to their respective rooms:
  + Getting credit card information from the guest
  + Handing the room pass card to guests
  + Answering any questions the guest have
* Dealing with booking by phone and face-to-face
* Preparing bills and taking payments
* Completing hotel procedures when guests arrive and leave using the hotel management system

Furthermore, end-users have access to only certain parts of the reporting module and full access to the room availability and booking module. This means that the end-users have to be able to properly enter the guest’s information and booking details so that the information can be correctly stored in those reports, for the management user and administrators to view. Moreover, the room availability and booking module will enable end-users to view each guest’s personal and hotel booking information. If there is any changes needed to be made to the guest or room information, the end-users will be able to apply the changes immediately.

The second type of users that will be using the software system is the management user. The management users consist of employees who are usually supervising other employees on their duties. Some examples of management users are:

* Front desk supervisor: The front desk supervisor manages the front desk/reception area of the hotel. As part of his job scope, he will be in-charge of scheduling his staff’s working hours, so that the front desk is always managed with the right number of receptionist at each shift. Hence, he will have to constantly update the staff’s schedule and duty type under housekeeping and staff management module.
* Housekeeping supervisor: The duty of a housekeeping supervisor ensures that the housekeepers of the hotel is properly trained. Similarly, as part of his job scope, he will be in-charge of scheduling his housekeeper’s working hours, so that the rooms are properly cleaned and maintained with the right number of housekeepers at each shift. Hence, he will have to constantly update the staff’s schedule and duty type under housekeeping and staff management module.

In addition, management users will have the authority to access the full module of room availability and booking, housekeeping and staff management, and reporting. All these reports will provide information on the guests, staffs who are working in the hotel, and room occupancy.

Lastly, the last type of users who will be using the software system is the administrator. One example of an administrator is Mr Wang. He will manage the overall system to ensure consistent and accurate data information. Using his authority to have access to the room availability and booking module, housekeeping and staff management module, reporting module, and user account and login creation module, he will be able to make necessary implementations and changes to the business operations.

Below is a table that illustrate key responsibilities of each individual users who are going to interact with the software system.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Responsibilities and Positions** | | |
| **No.** | **End-Users** | **Management Users** | **Administrator** |
| **1.** | Manage inflow of guests when they arrive to the hotel | Manage and supervise employees who are working under them | Manage the overall business operations of the hotel |
| **2.** | * Have access to certain parts of the reporting module * Have full access to the room availability and booking module * Have full access to the room occupancy report | * Have full access to the room availability and booking module, housekeeping and staff management module, and reporting module * Have full access to the room occupancy report | * Have full access to the room availability and booking module, housekeeping and staff management module, and reporting module * Have full access to the room occupancy report |
| **3.** | Given the authority to create a new guest’s under the room availability and booking module | * Given the authority to edit, update and delete information under the room availability and booking module. * Given the authority to assign duty types to housekeepers. * Given the authority to edit, update and delete information under the housekeeping and staff management module. * Given the authority to view different reports under the reporting module | * Given the authority to edit, update and delete information under the room availability and booking module. * Given the authority to edit, update and delete information under the housekeeping and staff management module. * Given the authority to view different reports under the reporting module. * Given the authority to create a new user, and register he/her password under the user account and login creation module |

### **System User**

There are three types of systems that aids in the construction of the hotel management system. The systems are: the client, the server, and the database. Each of these three types of systems has different functions and manages the overall hotel management system in different areas.

Firstly, the client. This system must be connected to the server at all times. Since the hotel is operating 24/7, the client will have to keep connection with the server while the physical actors interact with the system. Furthermore, Mr Wang wants to print the different reports under the reporting module. Hence, the client must be able to print the reports and offer other options as well.

The second system is, the server. The server is the system that accepts multiple connections from the client and saves the information into the database. Furthermore, the server have to accept the client’s connections and log the whole process.

Lastly, the last type of system is the database. According to an additional feature that Mr Wang requests the room availability and booking module to have, is for the system to allow the management users and administrators to edited and updated whenever they need. Hence, besides handling all the data inputs of the system; the database must also ensure secure updates to be done to the system.

In conclusion, the primary actor is the client that connects the server and receive reports. The end-users is to be used as a physical actor to describe the physical users of the client. The same term is also used for the management users and the administrators who are the physical user of the database and the server.

## General Constraints

Incompatible database

There is a need to ensure that the version of Microsoft SQL Server can be installed and run on the PC running on Windows XP SP2. When the database is being developed on another computer running a much higher version of SQL Server, the database file may not able to be imported into the hotel’s computer which can run the lower versions.

Limited IT infrastructure

It is not possible for every users to share a single computer located at the receptionist counter as there will be cases that the receptionist are busy using it to attend to guests. So there is a need to have another device for the administrator and management users.

Obsolete PC

The computer located at the reception counter is using Windows XP SP2 which is released in 2005. Microsoft had ceased support for Windows XP since April 8, 2014. It have only 1GB of RAM which may lead to lagging when the computer runs a lot of programs. It is equipped with only 160GB of hard drive space which might not be enough to store large amount of data.

## Functional Requirements

This section of the report contains the listing of all the functional requirements for the hotel management system. The list contains unique numbers and names with a short description of each requirement. The following section describe these requirements in full detail.

**FR01:** Login - The system should allow users to be able to log into their account which are registered in the user account and login creation module.

**FR02:** Booking and reservation - The system shall capture the information provided by the guest and assign a clean vacant room for the guests

**FR03:** Search - The system shall provide search functions for users to search for guest’s details and room information during check-out.

**FR04:** Payment - The system shall calculate the total charges the guest has accumulated over his/her stay in the hotel and generate a payment invoice

**FR05:** Reports - The system should be able to generate five different reports according to their requirements

**FR06:** Room availability and Booking module - The system shall allow the management users and administrators to edit the guest information as and when the guest arrive at the hotel

**FR07:** User account and Login creation module - The system shall allow the administrator to create a new account for new staff and new users

**FR08:** Preview - The system shall allow all reports to be previewed before it is printed out

**FR09:** Backup - The system shall backup the data in the reports and be kept in the system for 5 years

Beside listing and describing the different functions of the system, this section of the report will provide ranking and prioritization for each function. It will be done based on how important it is and the impact it has towards the system. Below is a table that shows the terms that i will be using along with its explanation.

|  |  |  |
| --- | --- | --- |
| **No.** |  | **Meanings** |
| 1. | 5  Severe | The software will not be complete unless the requirement is satisfied. It is an important and critical requirement required for the overall system to work |
| 2.. | 4  Significant | The software might be complete unless the requirement is satisfied. It is significantly important for the overall system to work |
| 3. | 3  Moderate | The software may or may not be complete unless the requirement is satisfied. It is moderately important for the overall system to work |
| 4. | 2  Minor | The software may be able to work. It supports other necessary system operations required for the overall system to work |
| 5. | 1  Minimal | The software will be able to work. It enhance the overall design and infrastructure of the system |

### **FR01: Login**

Description

The system should allow all users to be able to log into their account which are registered in the user account and login creation module

Users

All users: End users, Management users, and Administrators

Ranking

5

Input

Users is able to input his/her registered staff ID and password in the respective tab labeled “Username” and “Password” in the system’s user interface. The inputted information will be validated against the software system’s database for authenticity and authentication.

In the event that the user is unable successfully log into his/her account after multiple tries, he/she will be ask by the system to consult the administrative team. If the user faces difficulties logging into the system, it may be due to incorrect username and password being entered.

Display

The two input tab labels should be positioned in the center of the “Login” screen. The “Login” button shall will be placed directly below the labels.

System processing

Once the user clicks the “Login” button, the software system shall determine if the user has input the correct type of information. If the user have not enter the correct “Username” and “Password”, the system shall alert the user of the error. In addition, if the user did not make any inputs into the system, the system shall alert the user of leaving the field empty. Once the user make the necessary changes to the information and it is accepted by the system, the information is encrypted using symmetric encryption and sent across the IP network to the server. The server then compare the inputted information with those in the database. If the inputted information is correct, the system shall then proceed to the next part of the hotel management process: allow the user to select the different module he/she is able to access to. The different modules will be described in the subsequent requirement description.

System Output

The system shall display an error message if either the username or password entered by the user is incorrect. In the “Username” and “Password” tab where users input their staff ID and password, the system shall alert the user with a pop up message that states, “Your Username or Password is incorrect!” The text color of the type error message should be in red.

Once the user has clicked the “Login” button and the information entered is correct, the system shall advance to the next interface to allow the user to check a guest into a vacant room or to view a report. The checking-in process and interface is described in *FR02*: *Check-in*. Furthermore, being able to advance to the next interface also let the user know that the system have rectify and accepted his/her account details.

### **FR02: Booking and Reservation**

Description

The system shall capture the information provided by the guest and assign a clean vacant room for the guest

Users

End users and Management users

Ranking

5

Input

The user shall input the necessary information in the text field labeled according to the respective details. The system shall have four types of data in which the user must input to make room reservations.

The first text field is designated the “Name of Guest”. There shall be separate text fields for the first\_name and last\_name of the guest. A string of character shall be the type of data that is input in these fields. These fields shall be a required field.

The second text field is designated the “Contact” of the guest. There shall be separate text fields for the phone\_number, email, and home\_address/mailing\_address. The phone\_number must be in integer and in the format +##-###-####, where #is a number from 0 to 9, and shall be input in the text field. A string character shall be the type of data that is inputted into email and home\_address/mailing\_address. All these information shall be required fields.

The third text field is designated the “Payment” of the guest. There shall be separate text fields for the credit\_card\_number, credit\_card\_holder\_name, and expiry\_date. The credit\_card\_number must be in integer and shall be input in the text field. The credit\_card\_holder shall be a string of character. The expiry\_date shall be in the date data type and in the format DD/MM/YYYY, where D,M and Y are numbers from 0 to 9. All these information are required fields.

The “Room Reservation? text field is where the users will input details regarding reservations for the rooms. There shall be separate text fields for check\_in\_date, check\_in\_time, check\_out\_date, check\_out\_time, additional\_remarks,and comments. Both the check\_in\_date and check\_out\_date shall be a date data type, in the format DD/MM/YYYY, where D,M,Y are numbers from 0 to 9. Both the check\_in\_time and check\_out\_time shall be a timestamp data type, in the format ##:##:##, where # is numbers from 0 to 9. The time required format will be in 24hrs format. A string of characters shall be the type of data that is inputted into the additional\_remarks and comments. Additional remarks shall be used to indicate special requirements the guest wants. On the other hand, comments shall be used to indicate if a guest is going to check-out late.

A “Save” button for processing all the information shall be located near the bottom of the screen. The user shall click the “Save” button once she/he has entered the necessary and required information.

Display

The four text fields shall be positioned in the center of the “Booking and Reservation’s user interface. Each type of field will be arranged in a vertical structure. The “Save” button shall be located near the bottom of the screen, under all of the input fields.

System processing

Once the user click the “Save” button, the system determines if the user has input the correct type of information. If the user had not enter the correct information as defined by the input section of this requirement describe, the system shall alert the user with a pop up message. The pop up message will state which information is incorrect, and the user shall be able to correct the input data. Once the information is accepted by the system, the information is encrypted using symmetric encryption and sent over the IP network to the server. The server will place the data into its proper table in the database. The system shall then proceed to the next part of check-in process: assigning a clean vacant room for the guest, etc.

System Output

The system shall display an error message if any of the information that the user has entered is incorrect as defined by the input section of this requirement description. In the “Booking and Reservation” user interface where the user inputs the information, the system shall alert the user of incorrect information be displaying a pop up message stating the incorrect detail. The text color of the pop up message should be in red.

Once the user click the “Save” button and the information entered is correct, the system shall advance to the next interface to allow the user to continue with the check-in process. The advancement to the next interface will also let the user know that the system have rectify and accepted the input information. All the information that the user inputs shall be stored in the database on the server.

### **FR03: Search**

Description

The system shall provide search functions for users to search for guest’s details and room information during check-out

Users

All users: End users, Management users, and Administrators

Ranking

2

Input

Users are able to search the system for guest’s records by inputting the guest’s phone\_number into the system

Display

Users shall see a text bar and a “Search” button on the upper left side of the system’s user interface. The text bar shall have a description of the type of filter to the left of the text field. The description should state “Guest’s Phone Number”. The guest’s phone number that the user request will be displayed in at the left-hand corner of the ext field.

System processing

Once the user click the “Search” button, the computer queries the database using the text from the fields to narrow down the result. If text field is empty, the system shall return all the phone number which have been logged into the database. The guest’s phone number is returned to the computer which process the data and display the information back.

System Output

The guest’s phone number is directed to the system’s user interface and is displayed inside the text bar called “Guest’s Phone Number”. If no information is found, the system shall generate an error message telling the user what to do.

### **FR04: Payment**

Description

The system shall calculate the total charges the guest has accumulated over his/her stay in the hotel and generate a payment invoice

Users

End users

Ranking

4

Input

The system shall allow the user to input the guest’s phone number to search for his record in the database. Besides that, the user will have to input food products the guest has consumed from the minibar during his stay in the room. This information is then stored into a designated table in the database. The user will have to select the payment mode- cash or credit card, on behalf of the guest. This will be executed using radio buttons where the user is able to clearly select an option. There will be a “Confirm” button that the user shall select when he/she is ready to check the guest out, and a “Back” button for the user to exit the payment page.

Display

At the top of the interface, there will be a search function. For more information about the search function, refer to FR03. The system shall display the guest’s personal information after retrieving the record from the database. All these information will be positioned at the top of the interface, one beside the other. Following that, there will be two radio buttons right below the guest’s information. The radio buttons will be placed horizontally across the interface. The left radio button will be the “Credit Card” button, whereas, the right radio button will be the “Cash” button.

If the guest decides to pay by credit card, his payment details will be retrieved from the database and placed directly below the radio buttons. To check if the user did enter the guest’s credit card information during his/her check-in, the user is able to select the “Check” button that will be placed beside the guest’s credit card information. All the information will be using text field. The text field will be empty before retrieval. For more information about the data types of the guest, the data types are described in more detail in FR02.

On the other hand, if the guest decide to pay by cash, the guest’s credit card information will not be displayed. Instead, the system shall display the breakdown of the charges and the total charges incur by the guest during this stay at the hotel.

At the bottom of the interface, the “Confirm” button will be positioned at the bottom left of the interface, and the “Back” button will be positioned at the bottom right.

System processing

Once the user selects the “Check” button, the computer queries the database using the text phone number to narrow down the results. If the text field is empty, the system shall return all the phone number which have been logged in the database. The relevant information is returned to the requesting computer which process the data and display all the guest personal and credit card details. When the user enters the “Confirm” button, the system shall place the information in a proper table located at the database.

If the user decides not to check-out the guest, the user can select the “Back” button that will redirect the user to the previous interface.

System Output

When the user selects the “Confirm” button, the system will encrypt the data using symmetric encryption and transfer to the server, where it will place the data in a proper table stored in eh database. The interface will then direct the user to the main page

### **FR05: Reports**

Description

The system should be able to generate five different reports according to their requirements

Users

All users: End users, Management users, and Administrators

Ranking

3

Input

The users can view the five different reports in the system. The five reports are Room occupancy report, Room status report, List all the guest in one room report, Housekeeping report, and List all the guests in all the room report. Users are able to view the statistics as well for a particular date by selecting a daily, weekly, monthly or yearly view. This information is selected from a dropdown box. Once a view is selected, the user must click the “View Report? Button to see the information.

Display

The user interface shall have a dropdown box and button on the top left hand side of the display. Once the user clicks the button, the information requested is populated in the center of the screen. If no information is returned from the database, the following message shall be displayed: “No information found.”

System processing

Once the desired view is selected from the dropdown box and the “View Report” button is pressed, the user’s computer shall query the database for the desired information. If the database returns data, this data is processed and displayed in a readable form, or else an appropriate error message is generated informing the user why the data is not available,

System Output

All output is send to the system’s user interface. The reports are accessible via the tab marked “Report”, which can be found at the main user interface page.

### **FR06: Room availability and Booking module**

Description

The system shall allow the management users and administrators to edit the guest information as and when the guest arrive at the hotel

Users

Management users and Administrators

Ranking

3

Input

There are four data types that users have the options to edit. The four data types have been described in detail in FR02.

Display

The display of this interface will be similar to the interface display describe in FR02.

System processing

When the “Edit” button is clicked, the system shall preview the modified data before saving it into the database.

System Output

The output of the guest information is the changes that the user made to the four data types. The four data types are described in more details in FR02.

### **FR07: User account and Login Creation module**

Description

The system shall allow the administrator to create a new account for new staff and new users

Users

Administrators

Ranking

5

Input

The user shall input the necessary information in the “User account and Login Creation” in the system’s user interface. The system shall have five data types of data in which the user must input to create a new staff or new users for the hotel.

The first fields are designated the “Name of staff”. There shall be separate text fields for the first\_name, and last\_name of the staff. A string of characters shall be the type of data that is inputted in these fields. This shall be a required field.

The “Contact” of the staff shall be the second type of information that the user must input. There shall be separate text fields for the home\_address, and phone\_number of the staff. A string of characters shall be the type of data that is inputted home\_address field. The phone\_number must be in the format +##-###-####, where #is a number from 0 to9, and shall be inputted in a text field. This shall be a required field for the creation of a new staff.

The “date\_of\_birth” of the staff shall be the third type of information that the user input into the system. The date\_of\_birth must be in DD/MM/YYYY format, where D,M, and Y are number from 0 to 9. This shall not be a required field.

The fourth fields is designated the “Bank”. The bank\_accoount\_number shall be in a string of characters that the user have to input into the system.

The last field is designated the “Duty Types”. There are four duty types that can be assigned to staffs. The four duty types are general maintenance, room maintenance, security, and estate maintenance. The system shall allow users to input either one of the duty types for each housekeepers.

Display

The five types of user input fields shall be positioned in the center of the “User account and Login creation” user interface. Each type of field will be placed vertically upwards. The “Create” button shall be located near the bottom of the screen, under all the input fields.

System processing

Once the user clicks on the “Create” button, the system determines if the user has input the correct type of information. If the user has not input the correct information as defined in by the display section of this requirement description, the system shall notify the user. The system shall display a pop up messaging stating the incorrect information, and the user shall be able to correct the input data. Once the information is accepted by the system, the information is encrypted using symmetric encryption and set over the IP network to the server. This server place the data into its proper table in the database.

System Output

The system shall display an error message if any of the information that the users has entered is incorrect as defined by the input section of this requirement description. In the “User account and Login creation” interface where the user inputs the information, the system shall alert the user of incorrect information by displaying a pop up message. The pop up message will display the incorrect information and alert the user to change. The text color of the pop up message will be in red.

Once the user has clicked the “Create” button and the information is correct, the system shall allow the user to preview the information before creating it. Following that, the system shall advance to the next interface to allow the user to continue using the system. The advancement to the next interface will also let the user know that the system accepts the inputted information. All the information that the user input shall be stored in the database server.

### **FR08: Preview**

Description

The system shall allow all reports to be previewed before it is printed out

Users

All users: End user, Management user, and Administrators

Ranking

2

Input

No input required for this requirement

Display

The system shall display all the reports. For more information about the reports, refer to FR05 for a detailed description of the various reports.

System Output

Each report’s details and information

### **FR09: Backup**

Description

The system shall backup the data in the reports and be kept in the system for 5 years

Users

All users: End users, Management users, and Administrators

Ranking

3

Input

The system shall allow the user to select the “Backup” button to manually request the computer to back up the reports

Display

Under each report, there will be a “Backup” button located at the bottom center of the interface.

System processing

Once the user click the “Backup” button after logging and saving all the relevant information, the information will be duplicated and encrypted using symmetric encryption, and sent over the IP network to a server. This server places the data into its proper table in the database. The user can also request certain information from the database, and the database shall process the request and return the relevant information to the user’s computer. The user’s computer then processes this information and displays the selected reports in the user’s interface.

System Output

The database shall record all the systems events to log file on the server. System events include errors, data, and backup times. This output shall be a duplication of the actual report and be accessible to the system administrators so they can determine if the database is functioning correctly.

## Data Requirements

There are mainly six different types of data categories for the software system. Each categories will store different data of the system and allow different groups of users to access into it. The six different types of data categories are:

1. Room availability and booking module:

This category consist of data related to the guest. This set of data are accessible to the end-users, the management users and the administrative users. The types of data that will be stored in this category is:

* guest\_ID (var integer 30) - Primary Key
* staff\_ID (var integer 15) - Foreign Key
* first\_name (varchar  30)
* last\_name (varchar 30)
* room\_number (var integer 4) - Foreign Key
* number\_of\_adult\_per\_room (var integer 2)
* number\_of\_children\_per\_room (var integer 2)
* total\_number\_of\_guest\_per\_room (var integer 2)
* phone\_number (var integer 25)
* email\_address (varstring 30)
* home\_address / mailing\_address (varstring 60)
* credit\_card\_number (var integer 25)
* credit\_card\_holder’s\_name (varchar 60)
* expiry\_date (Date)
* additional\_remarks (varstring 100)
* check\_in\_time (Time stamp)
* check\_in\_date (Date)
* comment (varstring 100)
* check\_out\_date (Date)
* check\_out\_time (Time stamp)
* additional\_room\_charges (varstring 99)
* length\_of\_stay (var Integer 2)

1. Housekeeping and staff management module:

This category consist of data related to staffs working at the hotel. This set of data are accessible to the management users and administrative users. The types of data that will be stored in this category is:

* staff\_ID (var integer 15) - Primary Key
* first\_name (varchar  30)
* last\_name (varchar 30)
* date\_of\_birth (Date)
* bank\_aacount\_number (varstring 20)
* email\_address (varstring 30)
* home\_address / mailing\_address (varstring 60)
* phone\_number (var integer 25)
* duty\_type (varstring 30)

1. User account and login creation module:

This category consist of data related to detail of users. This set of data are accessible to the management users and administrative users. The types of data that will be stored in this category is:

* staff\_ID (var integer 15) - Composite Key
* password (varstring 30)

1. Payment:

This category consist of data related to the payment once the guest check-out of the room. This set of data are accessible to the end-users. The types of data that will be stored in this category is:

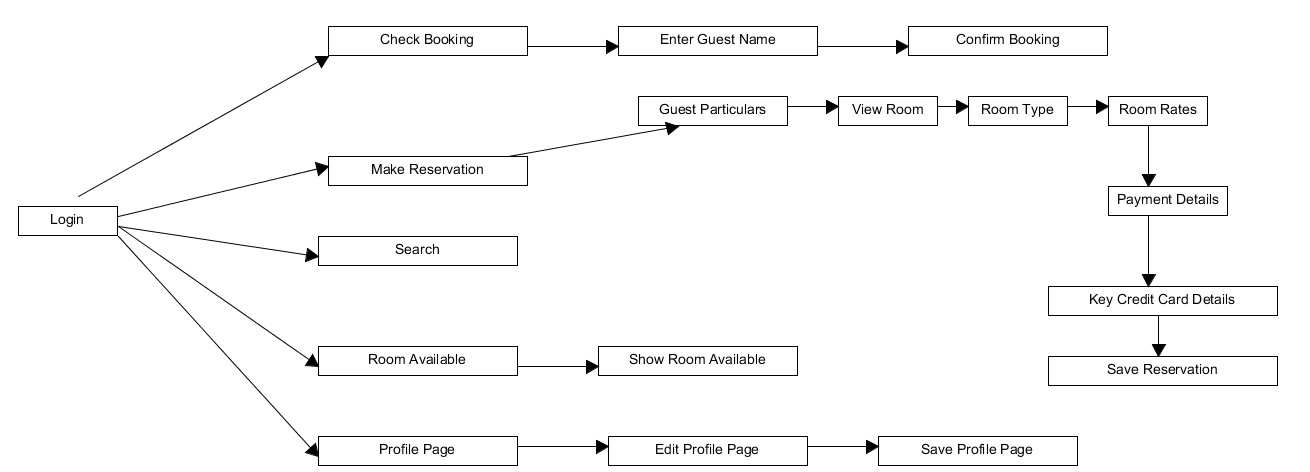
* guest\_ID (var integer 30) - Composite Key
* payment\_mode (var string 20)

1. Room

This category consist of data related to the room. This set of data are accessible to the end-users, and management users. The types of data that will be stored in this category is:

* room\_number (var integer 4) - Primary Key
* room\_rates (var integer 3)

## User Interface Requirements



*Figure 1 – User Interface Requirements*

This is what I imagined after the staff login to the system, it will show five features which is check booking, make reservation, search, room available and profile page. For the check booking, it is to check whether the guests has make a reservation in the hotel therefore the staff can just key in the guest name and the system would confirm whether the user make a booking. For the second features, which is the make reservation page, the user would have to key in the guests particulars then the system would bring the user to the view room page where the user would select the room they wanted and choose the room type that they want and proceed to the room rates page where the system would show the prices of the different rooms. After the user have choose they will then be proceed to the payment page where the user would have to key in credit card details and then the system would process the transaction and if everything is fine it will save reservation and confirm the reservation is made. While for the search page, it is to search for anything when in doubt and it will link the closest page that the web can find. While for the room available page, it show which room is currently available. While lastly the profile page, it is for the users to edit their profile page for example a change in their contact number they would have to proceed to the edit profile page and update their profile page to the latest one.

## Interface with Other Systems

For security purposes, the web-based client must use the HTTPS protocol that allows for secure communication over the Internet.

## Assumptions

* Ameline:

1. I assume that the administrator have the authority to delete modules, since Mr Wang is part of the administrative team.
2. I assume that the management users have the authority to delete staff records in the house keeping and staff management module.
3. I assume that redundant data and information of the hotel will be deleted to reduce memory storage space in the database server.
4. I assumed that each room key number will have 4 digits. The first 2 digits will be the floor that the room is located at, and the following two digits will be the room number. Example of room number: 1022, 905, etc.
5. I assumed that all the guests staying at the hotel are walk-in guests, since there is no other platform for guests to make their bookings and reservations
6. I assumed that the management users will be the one assigning the duty types to each housekeepers in the hotel
7. I assumed that all the users have the authority to back up the reports and records
8. I assumed that the management user and administrators will be able to create, delete, update and edit data information; since they have the full access to the different modules.

* Azrul

1. I assumed that the receptionist will be the only users to have to use a single computer located at the reception counter.
2. I assumed that the management users and administrator would use different computers like laptops or other devices so that they can access the client.
3. I assumed that the manager will tell their staff about their assigned jobs.
4. I assumed that the hotel’s broadband have no wireless capabilities.

* Douglas

1. I assumed the response timing that the system will take during peak non peak hours
2. I assumed that the maintenance date and time will always be the same
3. I assumed that the data is correct
4. I assumed that these are the features that will appear when the users has login in the navigation
5. I assumed that we would be able to finish the system on time
6. I assumed that  the budget is not enough for the system

# OPERATIONAL AND QUALITY REQUIREMENTS

## Operating Environment

The hotel management system will be developed as a web application using Microsoft Visual Studio. It will be deployed live using Microsoft Azure subscriptions. The web-based client can be accessed via web browsers. So the hotel’s computer would only need to have web browsers with internet connections using broadband.

Client

Mr Wang intended to install the completed software in a single computer located at the receptionist counter. The client computer specification is a Pentium 4 CPU running the Windows XP Service Pack 2. It have a size of 1 GB ram and 160GB hard disk drive. As we decided to adopt the 3-tier architecture, the computer will have to use the web browsers. The client can also be accessed from other devices. Furthermore, management users and administrators can access the hotel management system via web browsers on other computer devices.

Server

The web application will use the web server so that the client are able to retrieve or store data into the database. It will be deployed live using Microsoft Azure Services. The application programming interface (API) will be developed using Visual Studio. It will cost around $300 per month for Azure web services, and single SQL database. The database cannot be installed on the the reception computer due to its small memory that may leads to system lagging. So it must be installed on a server device and through virtualization, any computer can access the server.

Database

All the information entered into the system will be stored into the Microsoft SQL Server. It can be managed by Microsoft SQL Server Management Studio that have authentication.

## Development Constraints

In software architecture design, constraints come in two basic flavors - technical and business.  On most projects there are only a handful of constraints, but these constraints are a highly influential architectural driver.  Constraints, as the dictionary definition above indicates, are a limiting factor and severely restrict options for making design decisions.  They are also fixed design decisions that cannot be changed and must be satisfied. One of it would be software such as Visual Studios 2013 and Microsoft SQL Server. They are the key thing that is used to create the system. Without the software, it is impossible to create the system. Therefore it is essential to have all the software before the development of the system. Another factor would be time factor, we the developers had to plan our time properly and ensure that our progress is on track and not fall behind time as it could delay the starting date of the system resulting in loss and trust of Mr. Wang who entrusted us with the developing of the company. Lastly budget may also be a factor, as a lack of budget may cause the developers not to have all the essential resources to develop the software and this can cause the development timing to be delayed. In worst case scenario, it can cause the development not to be fully developed.

## Performance

Different timing the response and waiting time may vary. For example room availability and booking module part, the response time would be 10mins on average and up to 1 hour during peak hours. This is because the system need time to process the reservation, check for the room availability and the credit card details. Therefore it will take a long time if there is a reservation during the peak hours. Most of the time the peak hours occur during the 11-2pm. This is because this timing is when the guests has checked out from the room and it may cause some confusion in the system.

While for housekeeping and staff management, the response time for housekeeping would be up to 2 hour during peak hours and 20 mins. This is because during the peak hours, the housekeeper would need to clean up the room which the guest has checked out to welcome new guests that will be checking in later on. The housekeeper would also need time to clean up each room therefore the guests has to wait 20 mins for the housekeeper to clean up during non-peak hours. For the staff management, the response time for staff management would be 20mins during peak hours and 5mins during non-peak hours. This is because the staff may be busier to update their particulars when they are working, thus the waiting would be longer. The system would also need time to process the edited profile of the staff therefore 5mins is required.

The reporting module response time would be 2 hours during peak hours and 1 hour during non-peak hour. This is because the system need time to list all the rooms in the hotel and find out the room status whether is available or occupied or vacant and scheduled for cleaning. The system would need time to calculate how many guest is living in a room and calculate all the guests in all the room. They would also need to generate statistics on the room for the room occupancy. Lastly the housekeeping report, the system would have to list out all the duties. So that to ensure that the staff have been allocated to generate housekeeping schedule based on daily, weekly and monthly schedule. Another thing is available to the management for administrator uses.

The last function would be user account and login creation module, the response time for user account and login creation module would be 20mins during peak hours and 5mins for non-peak hours. This is because the user would have to create a new account to log in and make a reservation for the hotel. New staff would also require to create an account so that to access to the hotel features.

## Availability

According to what Mr. Wang mentioned in the interview above, the system is supposed to run 24/7 every day. There should not be any failure nor any shut down in the system. The appropriate timing for maintenance should be in the wee hours in the morning like 2-3am when the guests is sleeping instead of peak hours like 11-12pm when the guests are checking out. Therefore the maintenance should be done on the 1st Sunday of every month for 4 hours from 1-5am. While the data backup should be done every week for 2 hours    and the records should be kept for 5 years before they could be safely discarded.

## Security and Access Control Requirements

The end-users will have access to the room availability and booking module and selected reports in the reporting module that is important for them to carry-out their day-to-day operations. The end-users are composed of the receptionist staff only and they are given the full control permission for Room Availability and Booking module only. They are also given read-only permission for certain parts of the reporting module.

The hotel supervisors will be the management users and they can access to every modules. This grants them with full control permission to Room Availability and Booking, Housekeeping and Staff Management and Reporting module.

The administrator which is Mr Wang himself will have full control over all the 3 modules and he will also be the only staff to have access to maintain staff user accounts so that he personally authorizes the staff to use the system. Only selected staff such as receptionist, supervisors can be granted user account privileges. Furthermore, the administrator can also have access to the database and they can use the SQL Server Management Studio to enter query into the database.

The client system will have authentication to prevent unauthorized users from accessing the system. So the client only allows the staff of Regia Hotel to access it in order to carry-out booking for the guests or other tasks.

Database will need to be encrypted so that third-party will not able to read them.  So it would not affect the hotel’s credibility in protecting sensitive information that their guests provided. So the SQL Server database will undergo data encryption.

# SPECIAL REQUIREMENTS

One of the special requirements that is not included is the five reports that are reported in the reporting module to be able to export to excel so that the Mr. Wang would have numeric figures to be able to his spreadsheet. Another one would be a function so that the guest would be able to enquire the online availability online as this can save Mr. Wang trouble as he has to manual check the system and reply to the email manually. However due to time and budget constraints, the two features will have to wait for next time to be invented.

# ARCHITECTURE DESIGN

To achieve the system robustness, flexibility and resistance to potential change, we have selected the 3-tier architecture design to be implemented to the hotel. The architecture is composed of three layers: the user interface layer, the application logic layer and the database layer. The purpose of implementing a 3-tier system architecture to the hotel is to solve complicated problems that the staff may face in future while utilizing the software. Hence, to make the software development work more easily and efficiently, this architecture design segment the entire system into three main components. This is a benefit to the hotel because detecting of problems and error will be made easier if problems do occur.

The 3-tier architecture design consist of 3 main components. They are the interface layer, logic layer and data modelling layer. Firstly, the interface layer in the three-tier architecture offers the user a friendly and convenient entry to communicate with the system while the software application logic layer performs the controlling functionalities and manipulating the underlying logic connection of information flow. And lastly, the data modelling job is conducted by the database layer, where it can store, index, manage and model information needed for the software application. The following sections of the report will further elaborate the different layers and their contributions towards the system

* User Interface Layer

The first tier is the user interface tier. This tier manages the input and output of data and their display. With the intention of offering greater convenience to the user, the system is prototyped on the internet. The users are allowed to access into the system by using existing web browser software. The user interface tier contains HTML component needed to collect incoming information and to display information received from the application logic tiers. The guest communicates with the web server via the application protocol, sending request and receiving replies.

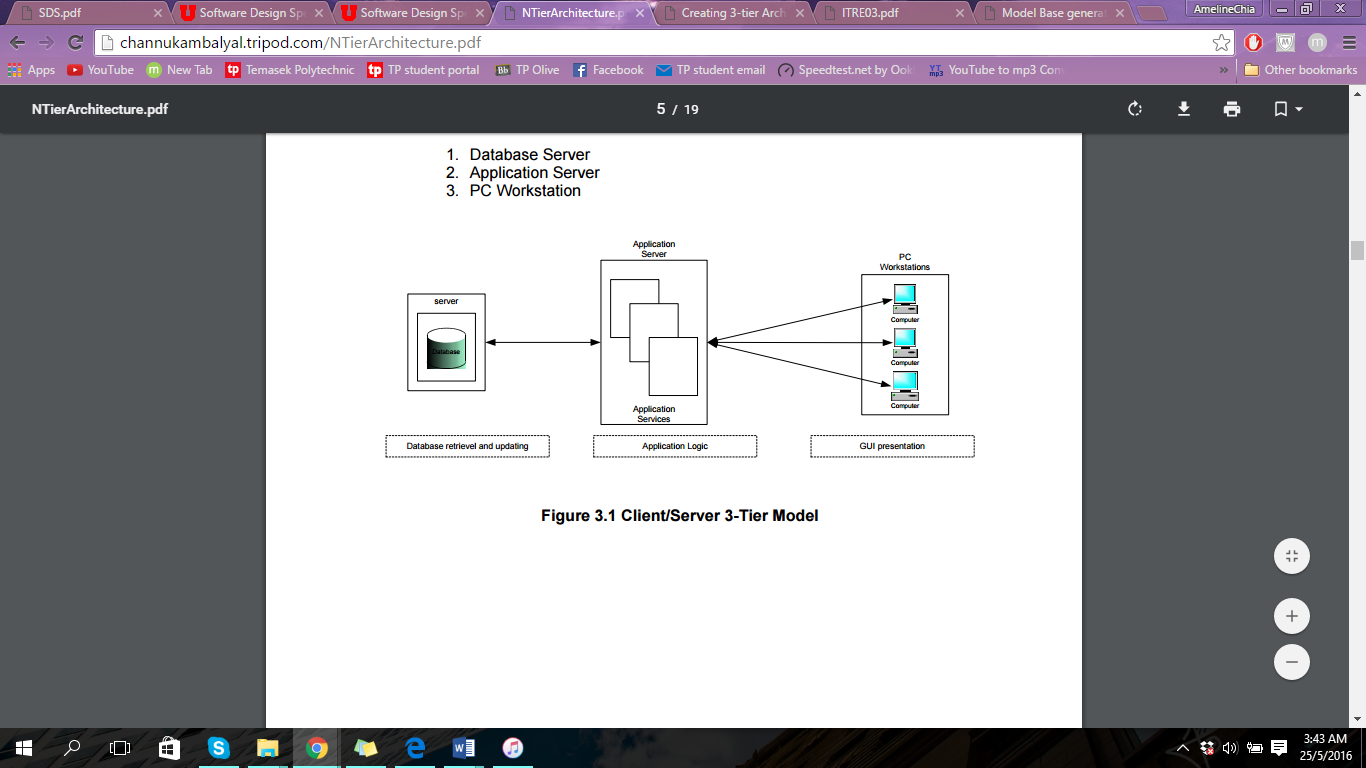
* Application Logic Tier

The application logic tier is the middle tier, which bridges the gap between the user interface and the underlying database, hiding technical details from the users. Components in this tier receive requests coming from the interface tier and interprets the request into actions. It acts according to the user’s demand towards the software and database.

* Database Tier

The database tier is responsible for modelling and storing information needed for the system and for optimizing the data access. Data needed by the application logic layer are retrieved from the database, then the computation results produced by the application logic layer are stored back in the database. Since data are one of the most complex aspects of many existing information systems, it is essential in structuring the system. Both the facts and rules captured during data modelling and process are important to ensure the data integrity.

By combining all the three different layers together, it create an organized structure for the software to run on. Figure 2 shows the overall infrastructure of the architecture design.



*Figure 2- Three-tier architecture design*

With reference to the problems provided by Mr Wang, there are plenty of functions required for all three tiers to collaboratively work together as one. For example, the hotel management software that we will be creating will the first layer of the architecture design - The User Interface Layer. When a user wants to search for a guest, they will enter the guest’s phone number into the provided software interface. He/she will enter the guest’s phone number and enters the “Search” button. By selecting the “Search” button, the requested phone number will be sent to the Application Logic Tier. At this stage, symmetric encryption will be executed to the phone number to protect the data from threats.

After which, the protected data will be sent to the database where all the data are stored. Using the phone number entered by the users narrow down the search for a specific guest. Programs will run to sift out the matching data. Once the matching record have been detected and retrieved from the database, it will pass through the application logic tier once again. The data will be encrypted with symmetric encryption and sent back to the user interface tier, where it will display the matching guest record to the user.

There are many benefits of implementing 3-tier architecture design to the hotel system. The benefits are:

* Consistent data used across all the system

The database tier provides database management functionality that ensures that the data is consistent throughout the distributed environment through the use of feature, such as data locking, consistency, and replication.

* Improves performance, flexibility, maintainability, reusability, and scalability

A features that improves performance, flexibility, maintainability, reusability and scalability of the system is called the centralizing process logic. Centralized process logic makes administrations and modifications of data easier by localizing the system’s functionality so that changes will not be duplicated, causing maintainability and reusability issues in future.

* Reliable, scalable and flexible

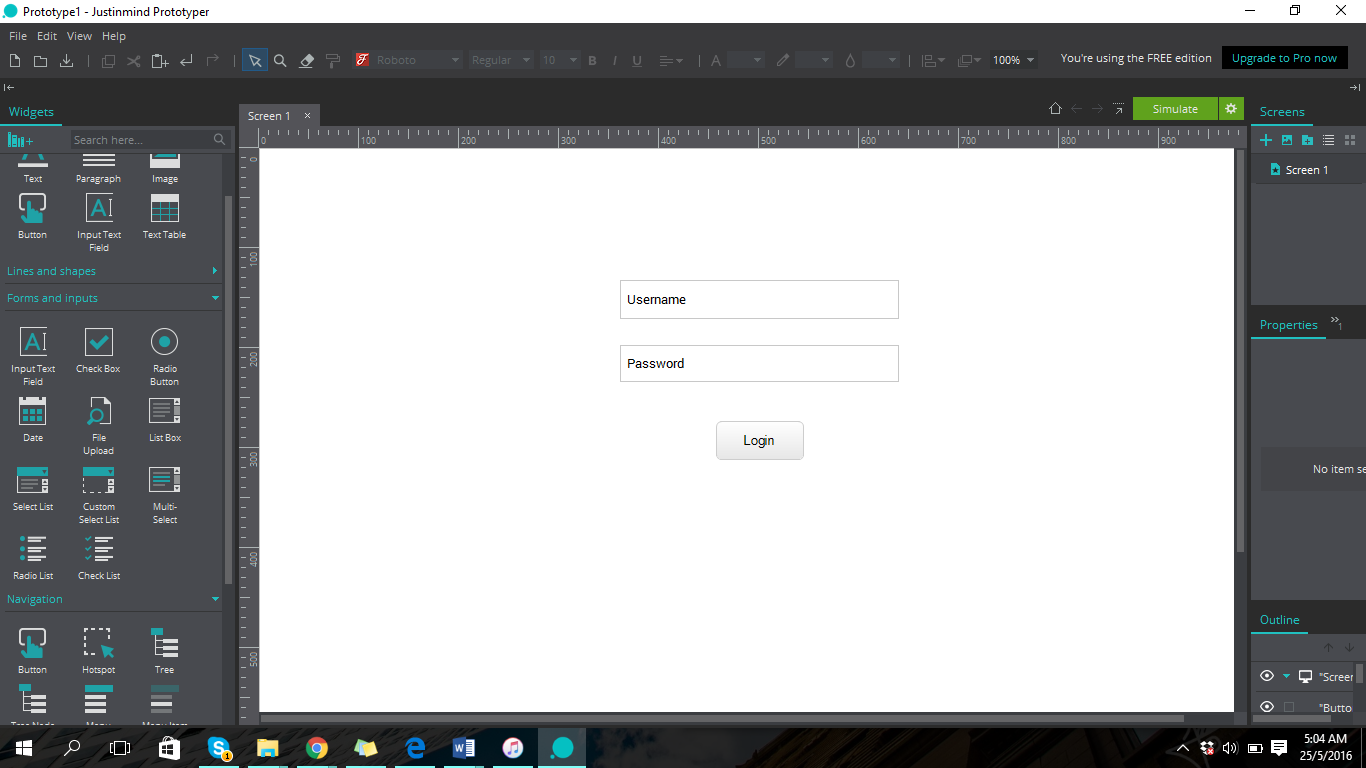
The application logic tier controls transactions and asynchronous queuing to ensure reliable completion of transactions.

In conclusion, three-tier architecture design is a good implementation to use for the hotel because each component of the design have different functions and deals with different problems. Since the hotel management system keep track of countless of data, it is crucial to the hotel that data are properly stored in an organized manner. Creation, retrieval and updating of staff and guest’s data will be so much easier to execute. Not only will it save a lot time during data extraction, but it will improve the overall efficiency of the system.  Furthermore, the structure of three-tier allow security measures to be easily implemented into the system.  Hence, we chose this architecture design compared to the other designs.

# USER INTERFACE (UI) DESIGN

Below are some simple user interface design that I have come out with. It is a simple designs that show the software system’s basic functionality.

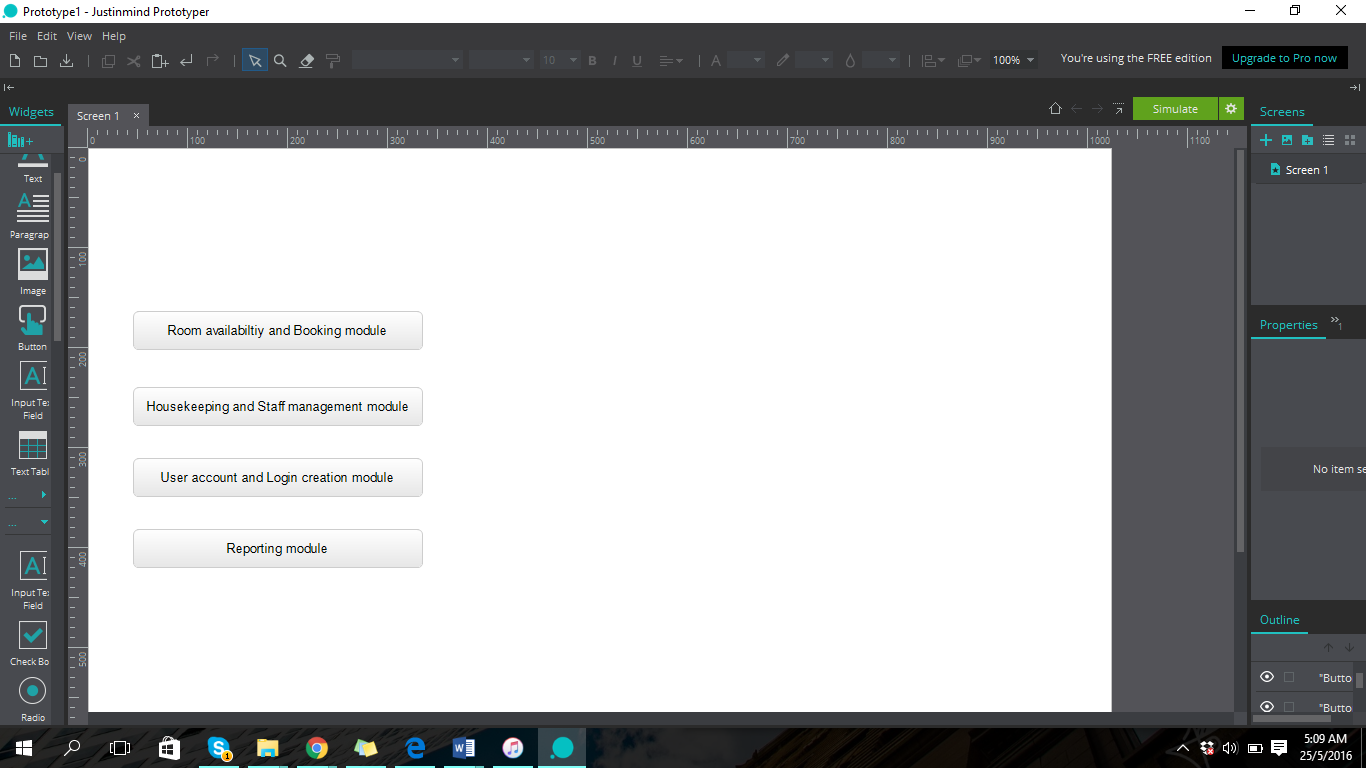
## Login



*Figure 3 – Login interface*

Figure 3 show the login page of the software. Users are able to input their staff ID and password into the respective text fields. The reason for arranging the different elements vertically because it is more organized and systematic. Users will be able to Login into the software unambiguously.

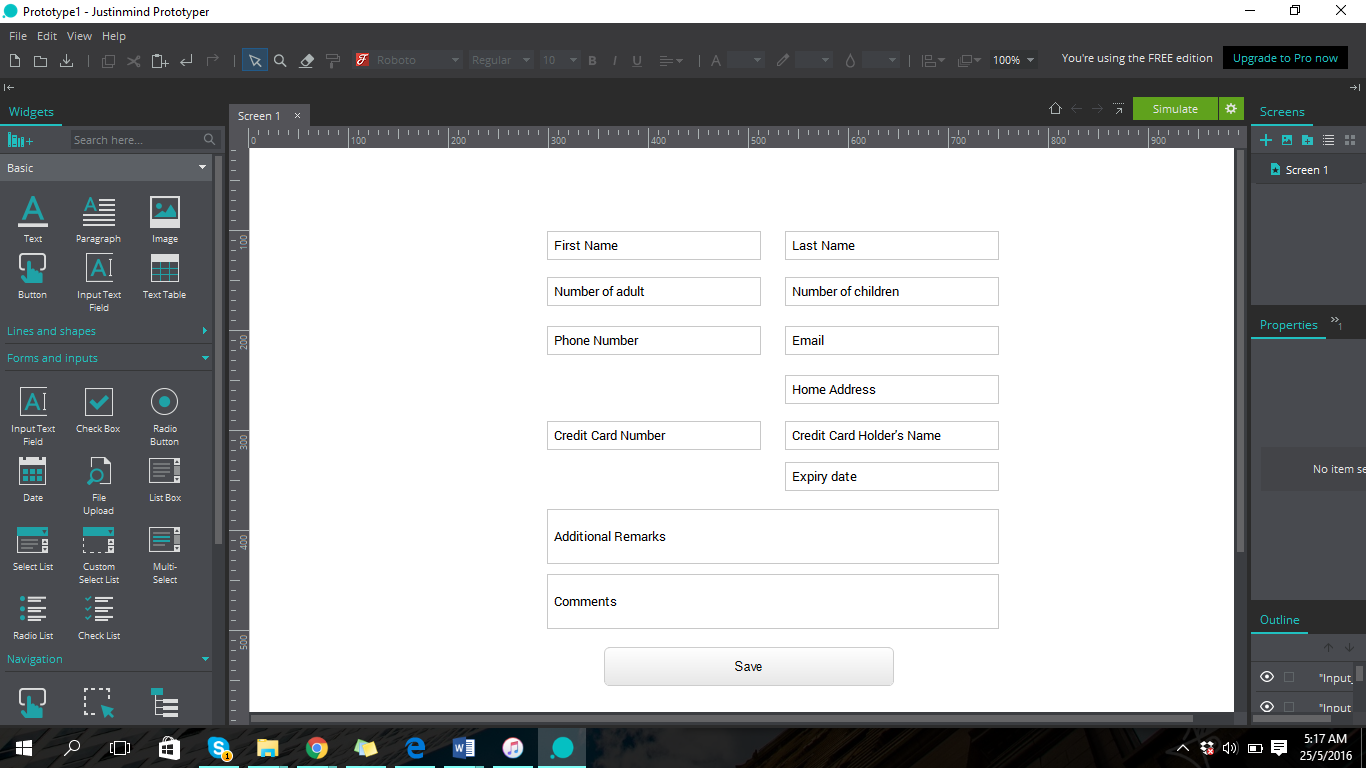
## Main page



*Figure 4 – Main page interface*

Figure 4 show the main page of the software where users able to choose between the four options – Room availability and Booking module, Housekeeping and Staff management module, User account and Login creation module, and Reporting module. This page will direct users to the different modules available in the software. The reason for arranging the different elements vertically because it is more presentable and organized. Users are able to clearly see the different options available for them to select.

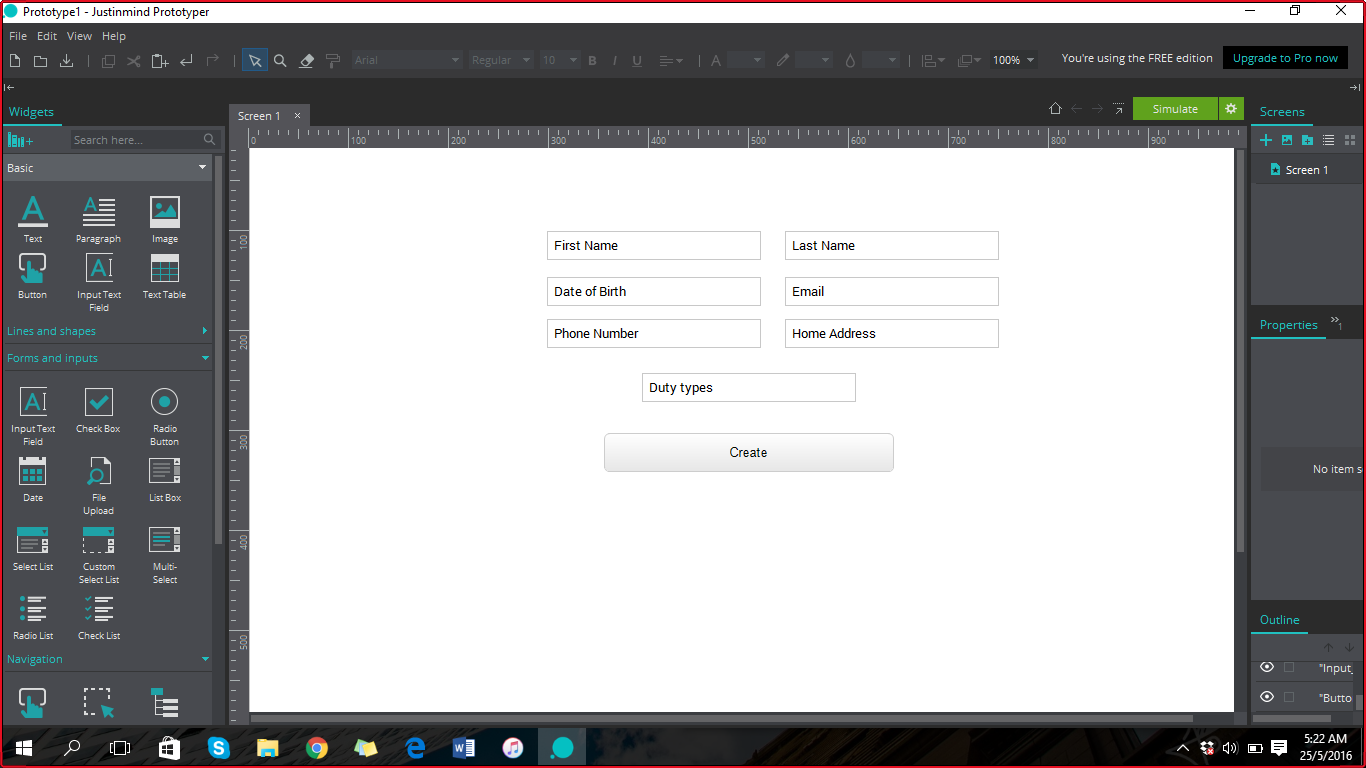
## Room availability and Booking module



*Figure 5 – Room availability and Booking module interface*

Figure 5 is the room availability and booking module where users are able to enter guest’s information into the system and store it in the database. Since there are more details required of this module, I decided to arrange the elements in columns and rows. This will ensure that all elements are visible to the user, and the user is able to easily key in the details without have to think twice.

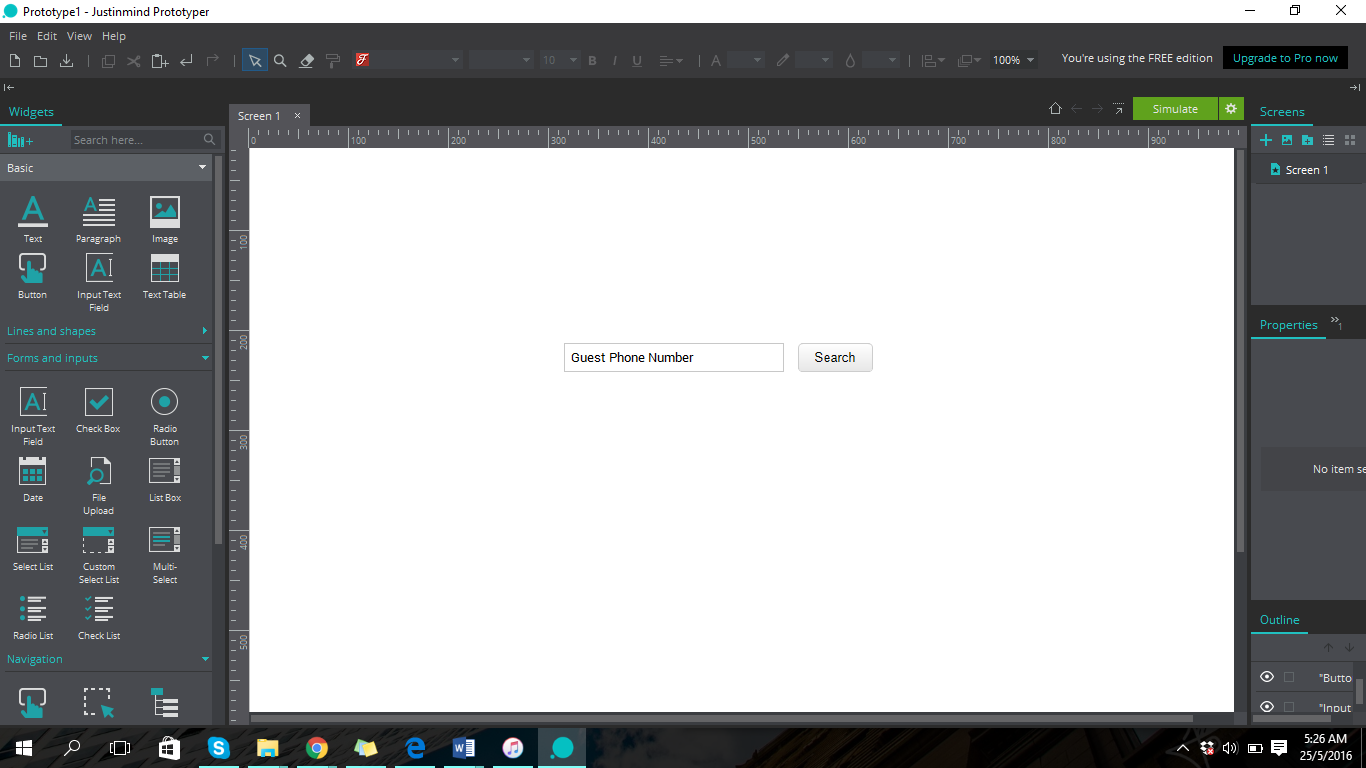
## Housekeeping and Staff management module



*Figure 6 –Housekeeping and staff management module interface*

Figure 6 is a platform that allow users to create and edit the staff’s details. Duty types is meant to be a dropdown bar, however due to presentation reasons, I have placed switch to a text box. For the real making of the systems, the duty types will all be stored in a dropdown bar. By using a dropdown bar instead of a text bar limit the user’s options.

## Search



*Figure 7 – Search guest information interface*

This figure shows the search function of the software. This search function only allow users to search for guest’s record during check-out. The “Search” button is placed beside the text field so that users are able to easily maneuver the page.

# PROGRAM DESIGN

## Description

Program design is done by planning out the use case which is the flow of events in completing the task of much implemented functions. Then, the sequence diagram is drawn to illustrate the use cases and it will help to create the methods required for the VOPC.

### **Use Case FR01: Login**

Brief Description

This use case allows all the users to login into the system via the web-based client.

Actors

End-users

Management users

Administrators

Flow of Events

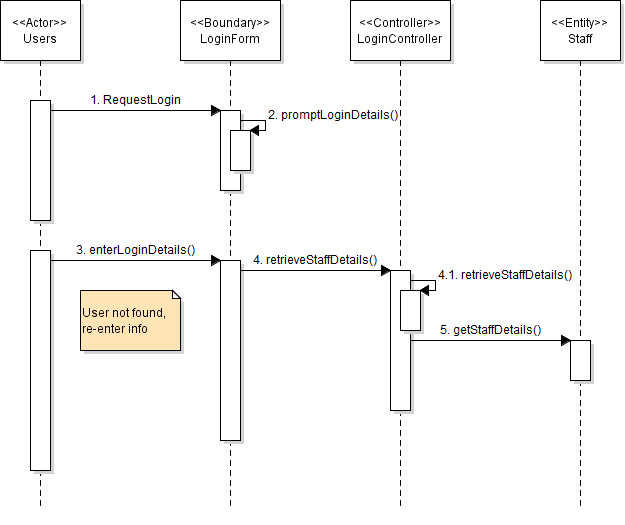
*Basic Flow*

|  |  |  |  |
| --- | --- | --- | --- |
| **Sequence No.** | **Actor Action** | **Sequence No.** | **System Response** |
| 1. | This use case starts when the actors wishes to login into the system | 2. | The system prompts for staff’s id and password |
| 3. | The user enters their staff’s id and password | 4. | The system checks if it exists in the database |
|  |  | 5. | The system redirects the user into the main page |
|  |  | 6. | The use case ends |

*User not found*

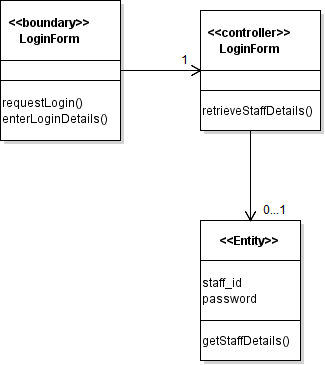
If the user was not found, the user might entered the wrong information. System will prompt if the user wanted to re-enter the information.

Sequence Diagram



*Figure 8 – Login Sequence Diagram*

VOPC

****

*Figure 9 – Login VOPC*

### **Use Case FR02 - Maintain Booking and Reservations**

Brief Description

This use case allows the receptionist to save the information that their guest provided into the system and assign a room for them.

Actors

End-users - Receptionist

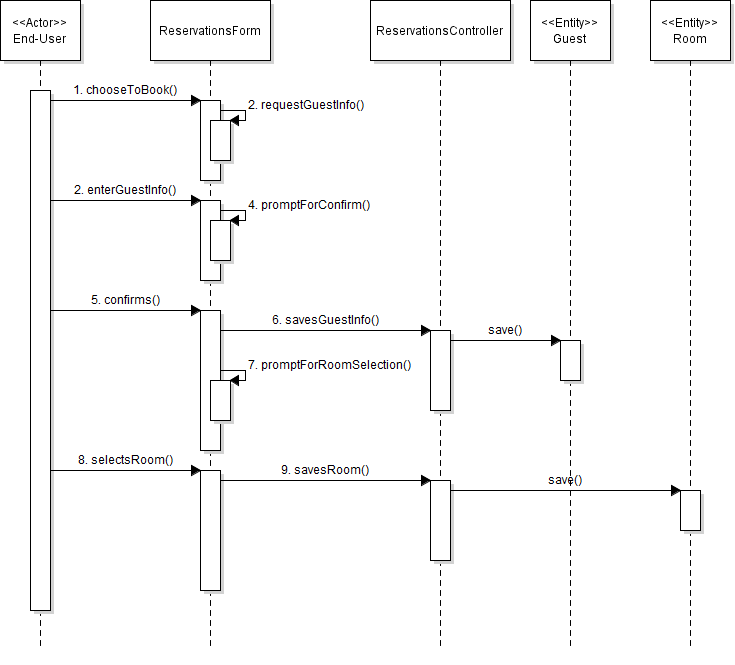
Management users

Flow of Events

*Basic Flow*

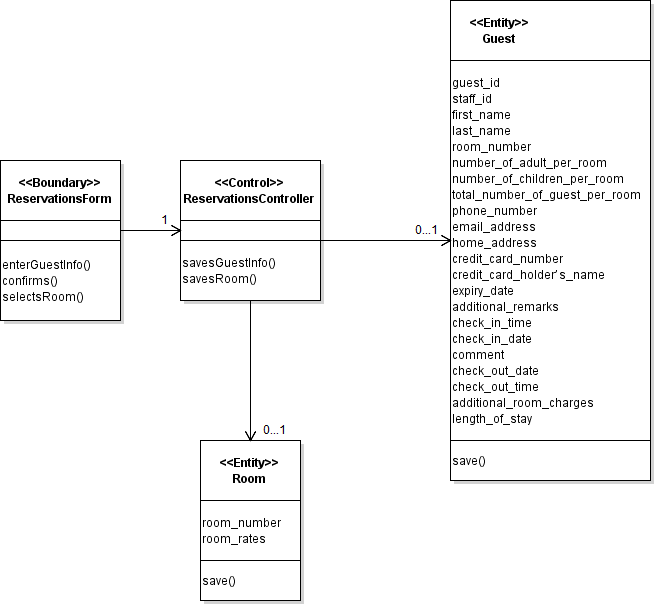
|  |  |  |  |
| --- | --- | --- | --- |
| **Sequence No.** | **Actor Action** | **Sequence No.** | **System Response** |
| 1. | This use case starts when the user wanted to enter guest information for check-in guest | 2. | The system prompts for guest information |
| 3. | The user enters the guest information | 4. | The system prompts for confirmation |
| 5. | The user confirms the information entered | 6. | The system saves it and redirects the user to the room allocation page |
|  |  | 7. | The system prompts for selection |
| 8. | The user selects the room | 9. | The system saves it into the database |
|  |  | 10. | The use case ends |

Sequence Diagram



*Figure 10 – Maintain Booking and Reservation diagram*

VOPC



*Figure 11 – Main Booking and Reservation VOPC*

### **Use Case FR03 - Search Guest**

Brief Description

This use case allows the receptionist to search for guest and room information during check-out.

Actors

End-users - Receptionist

Management users

Administrator

Flow of Events

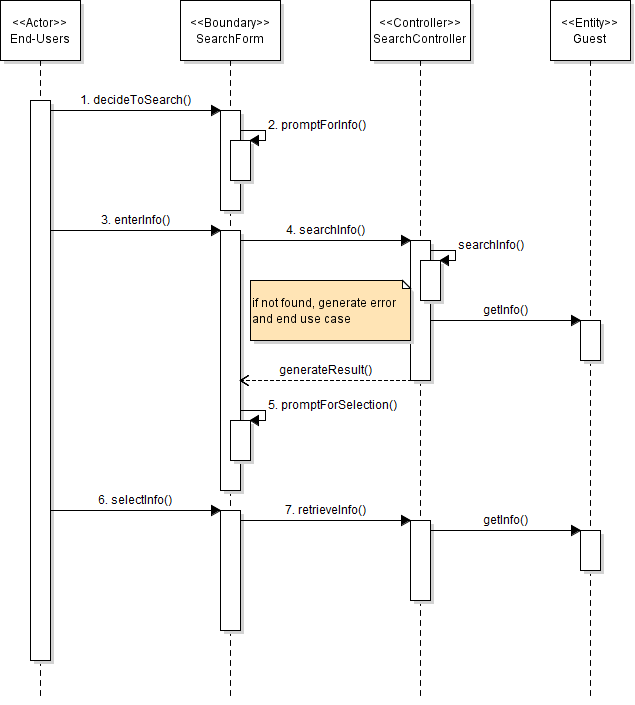
*Basic Flow*

|  |  |  |  |
| --- | --- | --- | --- |
| **Sequence No.** | **Actor Action** | **Sequence No.** | **System Response** |
| 1. | This use case starts when the receptionist wanted to search for the guest and room information for check-out | 2. | The system prompts for guest and room information |
| 3. | The user enters the guest and room information | 4. | The system searches for the information |
|  |  | 5. | The system generates the search result and prompts for selection |
| 6. | The receptionist selects the guest information | 7. | The system redirects user to the check-out page with the guest and room information |
|  |  | 8. | The system prompts for check-out details |
| 9. | The receptionist enters the checkout details | 10. | The system saves and the use case ends |

*Guest information Not Found*

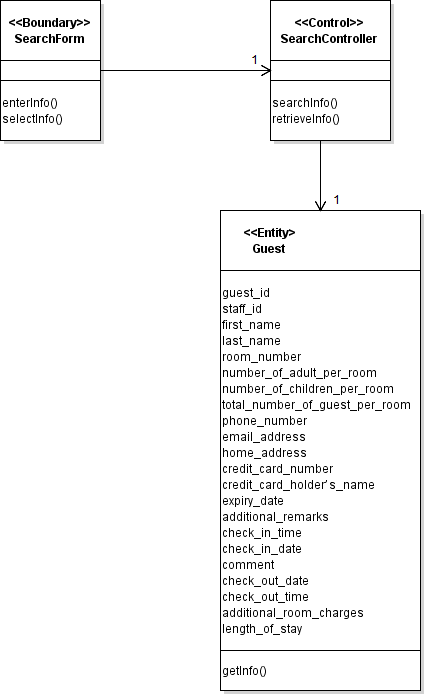
If the search results generates not found message, it may be caused of incorrect entry or the guest details did not exist in the system.

Sequence Diagram



*Figure 12 – Search guest sequence diagram*

VOPC



*Figure 13 – Search guest VOPC*

### **Use Case FR04 - Receive Payments**

Brief Description

This use case allows the receptionist to receive payments from the guests after calculating the total charge during their stay and generate invoices.

Actors

End-users - Receptionist

Flow of Events

*Basic Flow*

|  |  |  |  |
| --- | --- | --- | --- |
| **Sequence No.** | **Actor Action** | **Sequence No.** | **System Response** |
| 1. | This use case starts when the receptionist were to receive payments from guest | 2. | The system prompts for the phone number of guest |
| 3. | The receptionist enters the phone number for search | 4. | The system search and display all the guest information |
|  |  | 5. | The system prompts for any minibar consumption |
| 6. | The receptionist enters the details | 7. | The system calculates the total charge |
|  |  | 8. | The system request for payment modes - cash or card |
| 9. | The receptionist selects the guest’s choice | 10. | If the selection is cash, go to cash alternative flow |
|  |  | 11. | Otherwise, go to card alternative flow |
|  |  | 12. | The system prints invoices |
| 13. | The receptionist collects the invoice for the guest | 14. | The system saves and the use case ends |

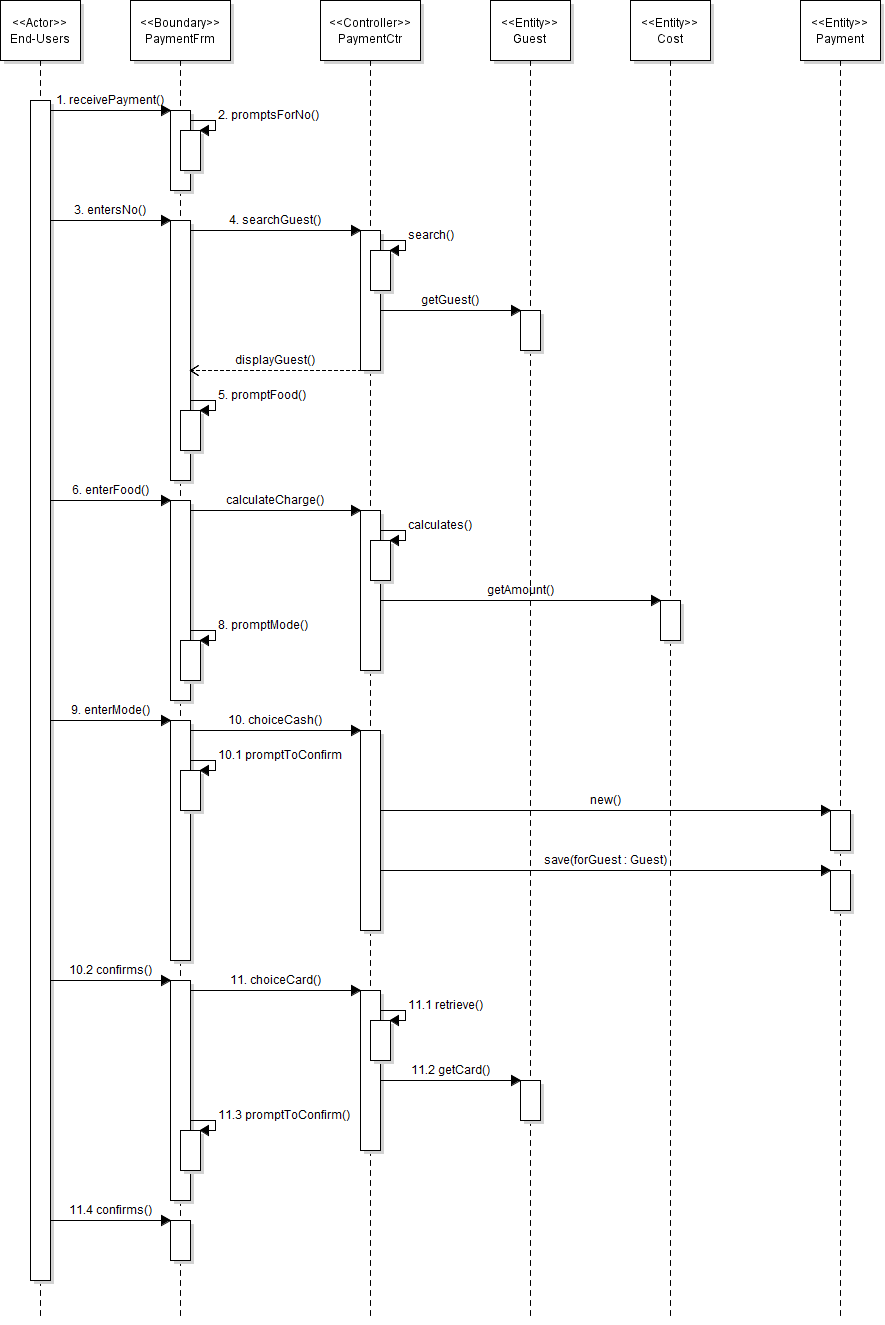
*Cash Alternative Flow*

|  |  |  |  |
| --- | --- | --- | --- |
| **Sequence No.** | **Actor Action** | **Sequence No.** | **System Response** |
|  |  | 10.1 | The system request confirmation on payment |
| 10.2 | The receptionist confirms payment had been made | 10.3 | The system saves the details |

*Card Alternative Flow*

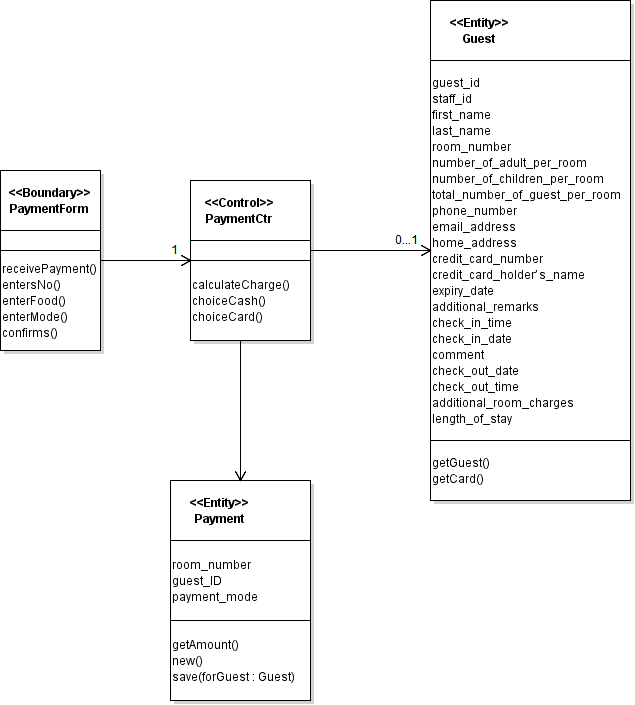
|  |  |  |  |
| --- | --- | --- | --- |
| **Sequence No.** | **Actor Action** | **Sequence No.** | **System Response** |
|  |  | 11.1 | The system retrieves credit card information from the database for the guest |
|  |  | 11.2 | The system request for confirmation |
| 11.3 | The receptionist confirms payment had been made | 11.4 | The system saves the details |

Sequence Diagram

****

*Figure 14 – Receive payments sequence diagram*

VOPC

****

*Figure 15 – Receive payments VOPC*

### **Use Case FR05 - Generate Reports**

Brief Description

This use case allows every authorized users to choose the reports that they wanted to generate

Actors

End-users - Receptionist

Management users

Administrator - Mr Wang

Flow of Events

*Basic Flow*

|  |  |  |  |
| --- | --- | --- | --- |
| **Sequence No.** | **Actor Action** | **Sequence No.** | **System Response** |
| 1. | This use case starts when the user wanted to generate reports | 2. | The system prompts for selection of 5 different reports - room occupancy, room status, list of guests in one room, housekeeping, list of guest in all room |
| 3. | The user makes their selection | 4. | If the choice is room occupancy, go to occupancy alternative flow |
|  |  | 5. | If the choice is room status, go to status alternative flow |
|  |  | 6. | If the choice is list of guests in one room, go to one room alternative flow |
|  |  | 7. | If the choice is housekeeping, go to housekeeping alternative flow |
|  |  | 8. | If the choice is list of guest in all room, go to all room alternative flow |
|  |  | 9. | The system prompts for selection - day-to-day, weekly, monthly and annually |
| 10. | The user makes their selection | 11. | The system saves and the use case ends |

*Occupancy Alternative Flow*

|  |  |  |  |
| --- | --- | --- | --- |
| **Sequence No.** | **Actor Action** | **Sequence No.** | **System Response** |
|  |  | 4.1 | They system generates the report |
|  |  | 4.2 | The system request saving or discard the report |
| 4.3 | The user makes their selection | 4.4 | The system saves or discard the report |

*Status alternative flow*

|  |  |  |  |
| --- | --- | --- | --- |
| **Sequence No.** | **Actor Action** | **Sequence No.** | **System Response** |
|  |  | 5.1 | The system retrieves and generates the room status report |

*One Room Alternative Flow*

|  |  |  |  |
| --- | --- | --- | --- |
| **Sequence No.** | **Actor Action** | **Sequence No.** | **System Response** |
|  |  | 6.1 | The system retrieves and generates the report |

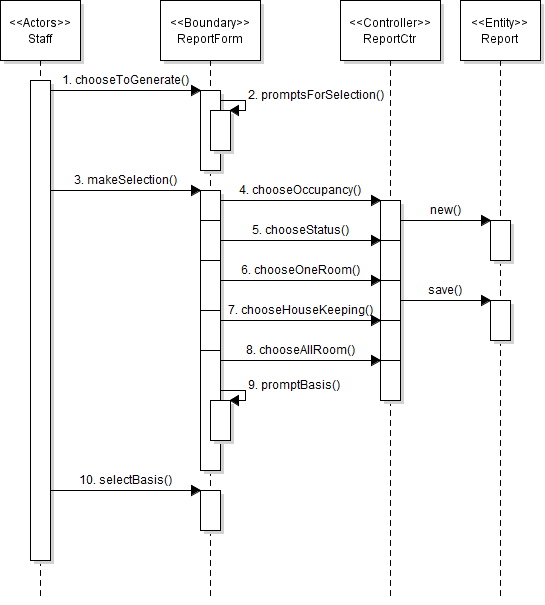
*Housekeeping alternative flow*

|  |  |  |  |
| --- | --- | --- | --- |
| **Sequence No.** | **Actor Action** | **Sequence No.** | **System Response** |
|  |  | 7.1 | The system retrieves and generates the schedule report |
|  |  | 7.2 | The system prompts for duty types selection for each and every staff |
| 7.3 | The user make their selection accordingly | 7.4 | The system updates and saves the report |

*All Room Alternative Flow*

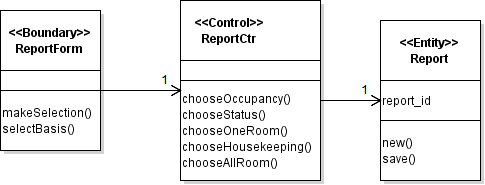
|  |  |  |  |
| --- | --- | --- | --- |
| **Sequence No.** | **Actor Action** | **Sequence No.** | **System Response** |
|  |  | 8.1 | The system retrieves and generates the report |

Sequence Diagram

****

*Figure 16 – Generate reports sequence diagram*

VOPC

****

*Figure 17 – Generate reports sequence diagram VOPC*

### **Use Case FR06 - Maintain Room Availability and Booking**

Brief Description

This use case allows the management users and administrators to maintain the room availability and booking. This may include modifying and deleting guest information.

Actors

Management users

Administrators

Flow of Events

*Basic Flow*

|  |  |  |  |
| --- | --- | --- | --- |
| **Sequence No.** | **Actor Action** | **Sequence No.** | **System Response** |
| 1. | This use case starts when the user wanted to make changes to the guest information | 2. | The system prompts for guest information |
| 3. | The user enters the guest information | 4. | The system searches for the information |
|  |  | 5. | The system generates the full report of the guest |
|  |  | 6. | The system prompts for 2 choices - update or delete guest information |
| 7. | The user make the selection | 8. | If the choice is update, go to update guest alternative flow |
|  |  | 9. | If the choice is delete, go to delete guest alternative flow |
|  |  | 10. | The system saves and the use case ends |

*Update Alternative Flow*

|  |  |  |  |
| --- | --- | --- | --- |
| **Sequence No.** | **Actor Action** | **Sequence No.** | **System Response** |
|  |  | 7.1 | The system redirects the user to the update page |
| 7.2 | The user can make changes to it if necessary and click save | 7.3 | The system prompts for confirmation |
| 7.4 | The user confirms the information | 7.5 | The system updates the details |

*Delete Alternative Flow*

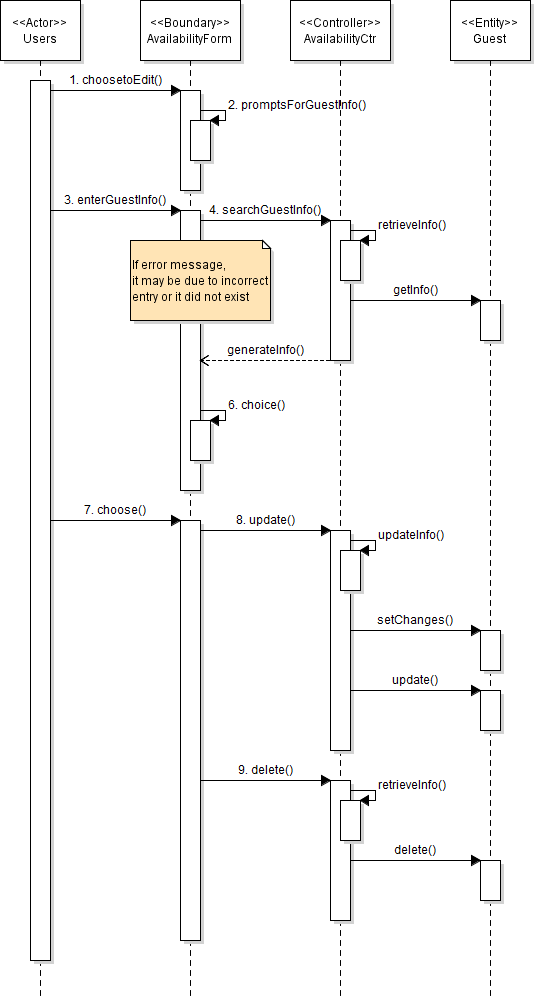
|  |  |  |  |
| --- | --- | --- | --- |
| **Sequence No.** | **Actor Action** | **Sequence No.** | **System Response** |
|  |  | 8.1 | The system prompts for confirmation to delete |
| 8.2 | The user clicks confirm to delete | 8.3 | The system deletes the guest from the database |

*Guest information Not Found*

Intercepts basic flow, step 4

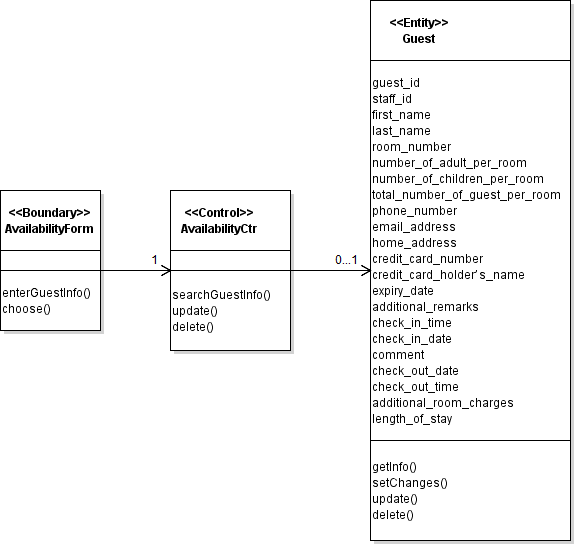
If the search results generates not found message, it may be caused of incorrect entry or the guest details did not exist in the system. The system asks for re-entry of information or else the use case ends.

Sequence Diagram



*Figure 18 – Maintain room availability and booking*

VOPC



*Figure 19 – Maintain room availability and booking VOPC*

### **Use Case FR07 - Maintain User Account and Login Creation**

Brief Description

This use case allows the administrators to maintain the user accounts in the login creation module. This may involves creating, modifying and deleting the user accounts

Actors

Administrators

Flow of Events

*Basic Flow*

|  |  |  |  |
| --- | --- | --- | --- |
| **Sequence No.** | **Actor Action** | **Sequence No.** | **System Response** |
| 1. | This use case starts when the administrator decided to create, modify and/or delete user accounts from the system | 2. | The system prompts for selection - create new user account, modify user accounts or delete existing user accounts |
| 3. | The administrator makes their selection | 4. | If the choice is create new user account, go to create alternative flow |
|  |  | 5. | If the choice is modify user account, go to modify alternative flow |
|  |  | 6. | If the choice is delete user account, go to delete alternative flow |
|  |  | 9. | The system saves and the use case ends |

*Create Alternative Flow*

|  |  |  |  |
| --- | --- | --- | --- |
| **Sequence No.** | **Actor Action** | **Sequence No.** | **System Response** |
|  |  | 4.1 | The system redirects the user to the create new user account page |
|  |  | 4.2 | The system prompts the administrator to fill in the required details |
| 4.3 | The admin fills in the required details and clicks save | 4.4 | The system prompts for confirmation |
| 4.5 | The admin confirms the details | 4.6 | The system creates the user account and saves |

*Update Alternative Flow*

|  |  |  |  |
| --- | --- | --- | --- |
| **Sequence No.** | **Actor Action** | **Sequence No.** | **System Response** |
|  |  | 5.1 | The system prompts for user account details |
| 5.2 | The user enters the details | 5.3 | The system looks up for the details in the user accounts |
|  |  | 5.4 | The system displays the search result and requests of selection |
| 5.5 | The admin makes the selection | 5.6 | The system retrieve and displays the selected user account information |
| 5.7 | The admin makes the required changes to the user account and saves | 5.8 | The system prompts for confirmation |
| 5.9 | The admin confirms the changes | 5.10 | The system updates and saves |

*Delete Alternative Flow*

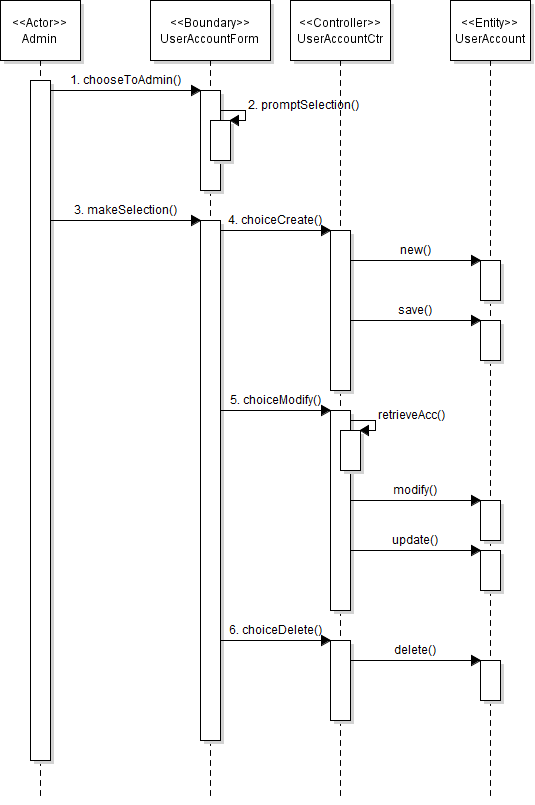
|  |  |  |  |
| --- | --- | --- | --- |
| **Sequence No.** | **Actor Action** | **Sequence No.** | **System Response** |
|  |  | 6.1 | The system displays all the user accounts and requests for keywords to search |
| 6.2 | The user enters the keywords to search | 6.3 | The system displays possible matches to the keywords in the login creation module |
| 6.4 | The user selects the chosen user accounts to be deleted | 6.5 | The system displays the user account information |
|  |  | 6.6 | The system requests for confirmation to delete |
| 6.7 | The admin confirms to delete the user account | 6.8 | The system deletes the account and updates |

*User Account Not Found*

Intercepts alternative flow, step 6.3 and step 5.4

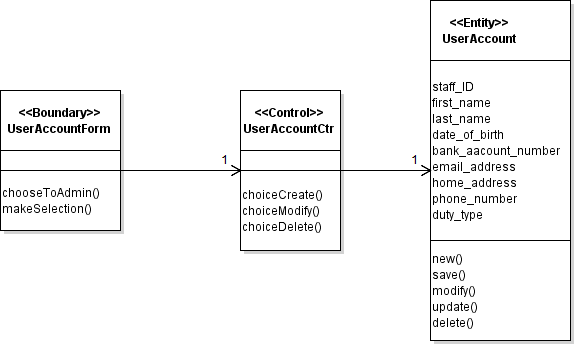
If the search results generates not found message, it may be caused of incorrect entry or the user account did not exist in the system. The system asks for re-entry of information or else the use case ends.

Sequence Diagram



*Figure 20 – Maintain User account and Login creation sequence diagram*

VOPC



*Figure 21 – Maintain User account and Login creation VOPC*

### **Use Case FR08 - Print Preview of Report**

Brief Description

This use case allows the management users and administrators to look into the report that had been generated in the system. They can choose to print it out.

Actors

Management users

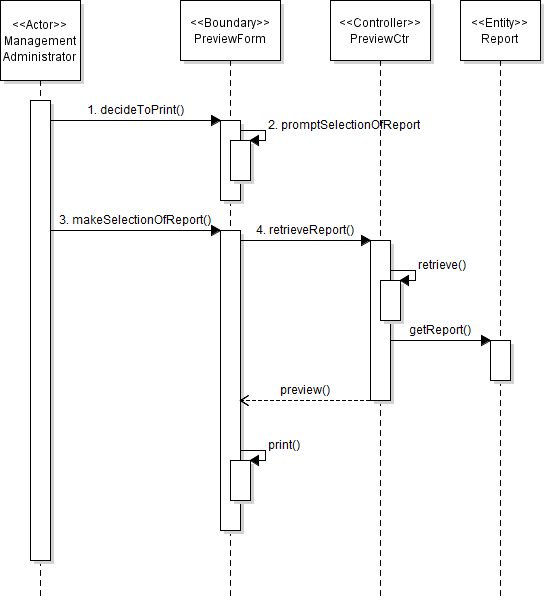
Administrators

Flow of Events

*Basic Flow*

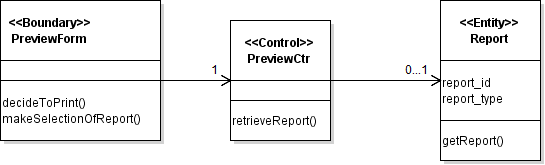
|  |  |  |  |
| --- | --- | --- | --- |
| **Sequence No.** | **Actor Action** | **Sequence No.** | **System Response** |
| 1. | This use case starts when the user decided to preview and print the report | 2. | The system prompts for selection of saved report |
| 3. | The user makes their selection | 4. | The system retrieves the full details of the report and preview it |
|  |  | 5. | The system gives an option to print |
| 6. | The user selects print | 7. | The system prints the report and saves |
|  |  | 8. | The use case ends |

Sequence Diagram



*Figure 22 – Print preview of report sequence diagram*

VOPC

**

*Figure 23 – Print preview of report VOPC*

### **Use Case FR09 - Create Backup**

Brief Description

This use case allows the management users and administrators to create backup for the data stored in the reports for the duration of 5 years in the system.

Actors

End-users

Management users

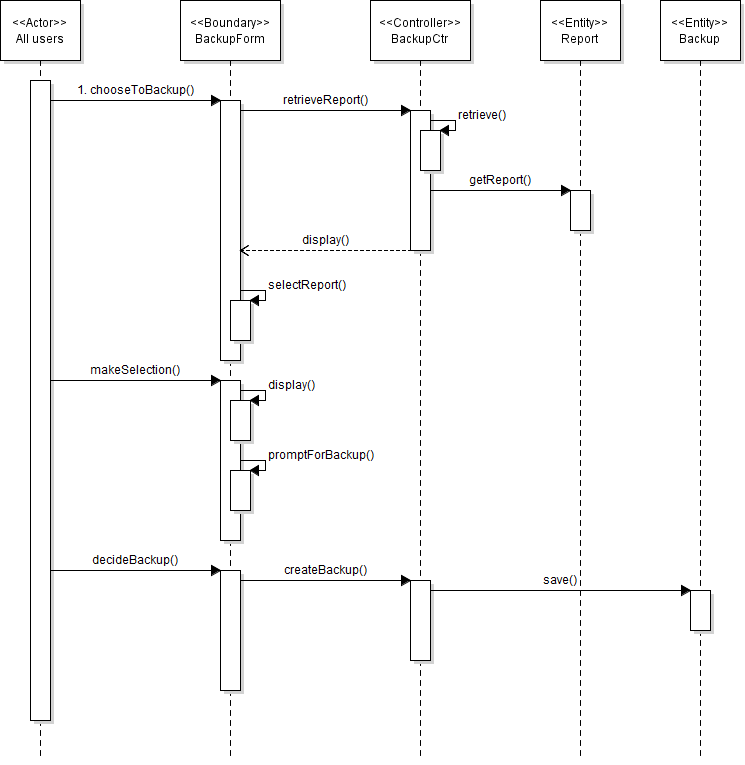
Administrators

Flow of Events

*Basic Flow*

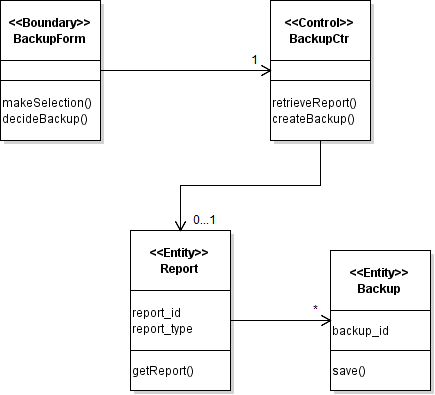
|  |  |  |  |
| --- | --- | --- | --- |
| **Sequence No.** | **Actor Action** | **Sequence No.** | **System Response** |
| 1. | This use case starts when the user wanted to create backup of reports | 2. | The system displays all the available reports |
|  |  | 3. | The system prompts the user to make a selection |
| 4. | The user makes their selection | 5. | The system displays all the selected report |
|  |  | 6. | The system prompts for backup |
| 7. | The user clicks on backup | 8. | The system saves the data in the reports as backup and the use case ends |

Sequence Diagram



*Figure 24 – Create backup sequence diagram*

VOPC



*Figure 25 – Create backup VOPC*

# DATABASE DESIGN

## Database Schema

Guest (guestID, firstName, lastName, roomNo, phoneNo, emailAddress, homeAddress,creditCardNo, creditCardHolderName, expiryDate, additionalRemarks, checkInTime, checkInDate, checkOutTime, checkOutDate, additionalChargers, lengthOfStay, comments)

Staff (staffID, firstName, lastName, dateOfBirth, bankAccNo, phoneNo, emailAddress, home Address,   dutyType)

Room (roomNo, roomRates, roomType)

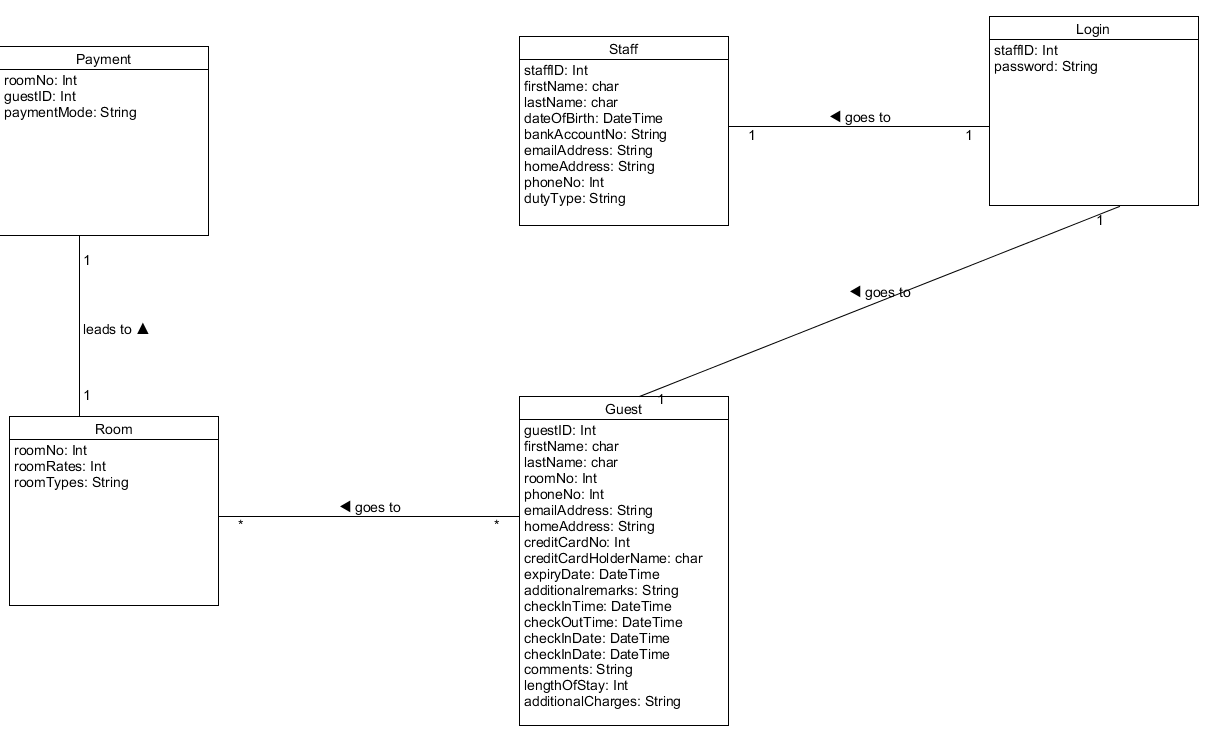
Payment (guestID, paymentMode)

Login (staffID, password)

Legend:

Green arrow = foreign key

## ERD Diagram



*Figure 26 – ERD Diagram*

## Data Dictionary

### **Guest Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Table column name | Table column datatype | Table column NULL option | Table Column is PK | Table Column is FK |
| Guest\_ID | var  Integer (30) | NULL | Yes | No |
| First\_Name | varChar (30) | NULL | No | No |
| Last\_Name | varChar (30) | NULL | No | No |
| Room\_No | var  Integer (4) | NULL | No | Yes |
| PhoneNo | var  Integer (25) | NULL | No | No |
| Email\_Address | varString (30) | NULL | No | No |
| Home\_Address | varString (60) | NULL | No | No |
| Credit\_Card\_No | var  Integer (25) | NULL | No | No |
| Credit\_Card\_Holder\_Name | varChar (60)) | NULL | No | No |
| Expiry\_Date | Date | NULL | No | No |
| Additional\_remarks | varString (100) | NULL | No | No |
| Check\_In\_Time | Time | NULL | No | No |
| Check\_Out\_Time | Time | NULL | No | No |
| Check\_In\_Date | Date | NULL | No | No |
| Check\_In\_Date | Date | NULL | No | No |
| Comments | varString (100) | NULL | No | No |
| Length\_ Of \_Stay | var  Integer (2) | NULL | No | No |
| Additional\_Charges | varString (99) | NULL | No | No |

### **Staff Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Table column name | Table column datatype | Table column NULL option | Table Column is PK | Table Column is FK |
| Staff\_ID | var  Integer (15) | NULL | Yes | No |
| First\_Name | varChar (30) | NULL | No | No |
| Last\_Name | varChar (30) | NULL | No | No |
| Date\_Of\_Birth | Date | NULL | No | No |
| Bank\_Account\_No | varString (20) | NULL | No | No |
| Email\_Address | varString (30) | NULL | No | No |
| Home\_Address | varString (60) | NULL | No | No |
| Phone\_No | var  Integer (25) | NULL | No | No |
| Duty\_Type | varString (30) | NULL | No | No |

### **Room Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Table column name | Table column datatype | Table column NULL option | Table Column is PK | Table Column is FK |
| Room\_No | var  Integer (4) | NULL | Yes | No |
| Room\_Rates | var  Integer (3) | NULL | NO | NO |
| Room\_Types | varChar (30) | NULL | NO | NO |

### **Payment Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Table column name | Table column datatype | Table column NULL option | Table Column is PK | Table Column is FK |
| Room\_No | var  Integer (4) | NULL | No | No |
| Guest\_ID | var  Integer (30) | NULL | No | No |
| Payment\_Mode | Var String (20) | NULL | No | No |

### **Login Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Table column name | Table column datatype | Table column NULL option | Table Column is PK | Table Column is FK |
| Staff\_ID | var  Integer (15) | NULL | Yes | No |
| Password | varString(30) | NULL | No | No |

Realistic is one of the key things that I consider when I was designing the database schema, this is because to ensure that these database are as close as to Mr. Wang ideal requirements. I have researched on many other hotel and these data are the most common kind that other hotel will provides too. Simplicity is also another factor that I have considered, this is because I want the software to be used by the staff and guests easily.

It gives a clearer picture on what should be put into the data storage requirements, as it is shown in a diagram way and a table form so that the users will understand it better and therefore can use it to plan for the software system.

There could be more attributes such as location or room description in the database schema

# REFERENCES

Technical difference between a standalone app vs web app. (n.d.). Retrieved May 24, 2016, from <http://stackoverflow.com/questions/7845451/technical-difference-between-a-stand-alone-app-vs-web-app>

Taking advantage of BIRT standalone runtime. (n.d.). Retrieved May 24, 2016, from <http://stackoverflow.com/questions/19051144/taking-advantage-of-birt-standalone-runtime>

SQL Server Encryption Options. (n.d.). Retrieved May 24, 2016, from <http://sqlmag.com/database-security/sql-server-encryption-options>

Risk Impact Assessment and Prioritization. (n.d.). Retrieved May 24, 2016, from <https://www.mitre.org/publications/systems-engineering-guide/acquisition-systems-engineering/risk-management/risk-impact-assessment-and-prioritization>

What is data modeling? - Definition from WhatIs.com. (n.d.). Retrieved May 24, 2016, from <http://searchdatamanagement.techtarget.com/definition/data-modeling>

Functional Requirements vs Non Functional Requirements . (2012). Retrieved May 24, 2016, from <http://reqtest.com/requirements-blog/functional-vs-non-functional-requirements/>

What is enterprise server? - Definition from WhatIs.com. (n.d.). Retrieved May 24, 2016, from <http://whatis.techtarget.com/definition/enterprise-server>

Define User Characteristics. (n.d.). Retrieved May 24, 2016, from <http://www.uregina.ca/external/communications/web-project/characteristics.html>

Support for Windows XP ended April 8th, 2014. (n.d.). Retrieved May 24, 2016, from <https://www.microsoft.com/en-us/WindowsForBusiness/end-of-xp-support>

Data Encryption Standard. (n.d.). Retrieved May 24, 2016, from <https://en.wikipedia.org/wiki/Data_Encryption_Standard>

HTTPS. (n.d.). Retrieved May 24, 2016, from <https://en.wikipedia.org/wiki/HTTPS>