Temasek Polytechnic

School of Informatics and IT

**Diploma in Information Technology (IT)**

Meeting Minutes

**Project Particulars**

|  |  |
| --- | --- |
| **Tutor** | Qi Yutao |
| **Class** | P01 |
| **Project Title** | Delonix Regia Hotel Management System |

**Project Team’s Particulars**

|  |  |
| --- | --- |
| **Matric Number** | **Student Name** |
| 1601553I | Yip Xiu Han |
| 1603625C | Ch’ng Wai Kit Wesley |
| 1601705E | Qamarul Fattah Bin Hamdan |

|  |  |  |
| --- | --- | --- |
| Date: | 13/05/2017 (Monday) |  |
|  |  |  |
| Venue: | Library |  |
|  |  |  |
| Present: | ALL |  |
|  |  | |
| Absent with apologies: | -NIL- |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **S/No** | **Item** | | **Action By** |
| **1.**  **2.**  **3**.  **4.**  **5.**  **6.**  **7.**  **8.**  **9.**  **10.**  **11.**  **12.**  **13.**  **14.**  **15.**  **16.** | Meeting started at **14:30HRS**  **Apologies for absence**No apologies were received.  **Stated Main Functions**  Mr Wang mentioned that the main functions of the hotel management system must include 3 modules, the room availability and booking module, a housekeeping and staff management module as well as a reporting module.  **Users of the System**  Mr Wang mentioned that there will be 3 different types of users. The first level of users are the   * **End Users** (Reception Staff), which can access parts of the reporting module as well as have full access to the Room Availability and Booking Module. * **Management Users** have access to all 3 modules (Room Availability and Booking, Housekeeping and Staff Management and Reporting module. * **Hotel Administrator** also has access to all 3 modules, in addition, the User Account and Login Creation Module.   **Room Availability and Booking Module Specifications**  Mr Wang requires the system to accept certain data inputs. Some of which are:   * **Booking Details** (Last Name, First Name , No. of adult guests and children) * **Personal Contact Details** (Phone number , E-mail, Home and Mailing address (E.g. Street address, block and house number, postal code and country.) * **Payment Details** Credit card payment (Credit Card Number, Cardholder Name, Date of Expiration) * **Check-In Details** Check-In Date & Time Desired check out date and time * **Additional Remarks** E.g. Whether guest requires king-size or queen-sized bed Smoking or non-smoking room * **Check-Out Details** Whether guest requires a late checkout   **Process Flow of Check-Out Function**Mr Wang elaborates on the process flow of the steps the guest/hotel staff take if the guest wishes to check out*.* Hotel guest checks out, typically should do so before 12pm. 12pm is stipulated check out time.  Guest will bring baggage to reception area, pass the room key over to the reception staff. Staff will ask whether guests have consumed any items in the mini-bar.  Despite any response, cleaning staff will check against item list to see if previous guests have taken anything from the mini-bar  Payment invoice will be generated which will calculate the no. of days they have stayed. Will also consider costs of mini-bar items. Payment invoice will contain guest details. E.g. (Check-In, Check-Out, no. of nights guest have stayed, Room Rates, Additional Costs)  Invoice will be given to the guest for them to double check and eventually payment will be made.  **Elaboration of Room Availability and Booking Module**  Mr Wang requests that this module will allow all three levels of users, to modify the guest records as and when they would like it. Some of the records that should be able to be changed would be the Room Type, the No. of Adults/Children staying in the room.  **Elaboration on Features of Staff Management Module**  Mr Wang requested for the implementation of 2 key features for the Staff Management Module.   * The first key feature is the ability for the system to keep records on the details of the staff, such as their name and phone number. * The second key feature is for the system to keep records on the roles that are assigned to the staff.   **Elaboration on Duties of Housekeeping Module**  Mr Wang listed the duties that the staff can be assigned to, which are General Maintenance, Room Maintenance, Estate Maintenance and Security.  **Elaboration on Types of Reports of the Reporting Module**  Mr Wang talked about the 5 different types of reports he would like in the Reporting Module   * The first type of report, mentioned as the Room Status report, will list all the rooms in the hotel as well as their respective room status (vacant / occupied / scheduled for cleaning). * The second type of report should be able to list all guests in a room see all occupants in room. * The third type of report should be able to list all the guests in all the rooms of the hotel (at any given point of time). * The fourth type of report, mentioned as the Room Occupancy report, should generate statistics on what the room occupancy was on a daily, weekly, monthly and yearly schedule. * The fifth and final report, mentioned as the Housekeeping report, will list the duties that staff has been allocated to.   **Requested additional feature for the Reporting Module**  Mr Wang briefly requested that he would like the ability to be able to preview the reports before sending them to the printer.  **Stated Budget**  Mr Wang mentioned that the budget discussed with Mr Lim has been finalized at 70,000 dollars.  **Operating Environment**  Mr Wang mentioned that the operating environment of the system will be on a single computer at the receptionist counter.  **System Availability**  Mr Wang mentioned that the system will be available 24/7, unless there is a major renovation. He also started that system backups may be performed between 3 to 4am.  **Interface To Other Systems**  Mr Wang said the system does not need to interface with other existing systems.  **Special Requirements**  Mr Wang briefly talked about exporting the 5 reports to a spreadsheet, specifically Microsoft Excel. | | Wesley  Wesley  Wesley  Wesley  Wesley  Xiu Han  Xiu Han  Xiu Han  Xiu Han  Xiu Han  Qamarul  Qamarul  Qamarul  Qamarul |
|  |  | |  |
|  |  | |  |
|  |  |  |  |
|  |  |  |  |

Meeting ended at **17:30 HRS**

Recorded by: ALL

Vetted by: ALL

Temasek Polytechnic

School of Informatics and IT

**Diploma in Information Technology (IT)**

Software Requirement Specifications (SRS)

**Project Particulars**

|  |  |
| --- | --- |
| **Tutor** | Qi Yutao |
| **Class** | P01 |
| **Project Title** | Delonix Regia Hotel Management System |

**Project Team’s Particulars**

|  |  |
| --- | --- |
| **Matric Number** | **Student Name** |
| 1601553I | Yip Xiu Han |
| 1603625C | Ch’ng Wai Kit Wesley |
| 1601705E | Qamarul Fattah Bin Hamdan |

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 13/11/2017 | 1.0 | Initial Draft | ALL |
| 21/11/2017 | 1.1 | Completed Document | ALL |
| 22/11/2017 | 1.2 | Vetted Document | Wesley |
| 22/11/2017 | 1.3 | Compiled Document | Wesley |

**Table of Contents**

[**DISTRIBUTION OF WORKLOAD**](#_6f47f3ttzd5w)9

[**OVERVIEW OF REQUIREMENTS**](#_jo4t9ruvh60t)9

[System Functions](#_2et92p0) 9

[User Characteristics](#_bzcyi3u90xr) 10

[General Constraints](#_2cowsyld9owk) 10

[Functional Requirements](#_lzkbl2zc4nda) 11

[Data Requirements](#_onh0h5dqao5u) 12

[User Interface Requirements](#_ga03ng8c5c0j) 21

[Interface with Other Systems](#_3xkskzr95hy3) 28

[Assumptions](#_ithj7qk9flk1) 28

[**OPERATIONAL AND QUALITY REQUIREMENTS**](#_d3vzigc79i60)28

[Operating Environment](#_cw72fjfn5jq5) 28

[Development Constraints](#_ei8iiw6x1mha) 29

[Performance](#_8wyq8w5a2q1z) 29

[Availability](#_l9bu9xgrhl1v) 30

[Security and Access Control Requirements](#_vr3oq9fenrjs) 30

[**SPECIAL REQUIREMENTS**](#_1qriw2nbcgmr)31

[**REFERENCES**](#_zf97wbk7tvpk)32

# DISTRIBUTION OF WORKLOAD

|  |  |
| --- | --- |
| **Requirement Gathering** | **Members** |
| Room Availability & Booking Module | Wesley |
| User Creation & Login Module | ALL |
| Housekeeping & Staff Management Module | Qamarul |
| Reporting Module | Xiu Han |
| User Characteristics | Wesley |
| General Constraints | Xiu Han, Wesley |
| Functional Requirements | ALL |
| Data Requirements | ALL |
| User Interface Requirements | Qamarul |
| Interface with other Systems | Xiu Han |
| Assumptions | ALL |
| Operating Requirements | Xiu Han, Wesley |
| Development Constraints | ALL |
| Performance | Xiu Han, Wesley |
| Availability | ALL |
| Security and Access Control Requirements | Wesley |
| Special Requirements | Xiu Han |

# 

# OVERVIEW OF REQUIREMENTS

## System Functions

**Room availability and Booking Module**

* This module should allow the staff at the front desk to take note of the guest details when guests are registering for a room in the hotel.
* Tracks the reservations made by customers. Also tracks the status of rooms, if the room is available at specific times.
* If a room is occupied, it will record the guest’s information as well as their check-in and checkout times.
* Allows administration or reception staff to edit existing guest records. Examples of information to be edited are room changes, guest names and the number of adults and children that are in the room.

**User Creation and Login Module**

* This module should allow the hotel system administrators to create, retrieve, update and delete user accounts for both the end users and management users.

**Housekeeping and Staff management Module**

* This module should allow the owner of the hotel to keep track of records of the staff including their details as well as respective roles.

**Reporting Module**

* This module should be able to help generate different types of reports for the hotel.

## User Characteristics

**End Users**: Helps customers to book, check in and checkout of hotel. Reception Staff work at the front counter: Only has access to parts of the reporting module but has full access to the Room Availability and Booking Module.

**Management Users**: Manages the hotel operations and also has access to all 3 modules (Room Availability and Booking Module, Housekeeping and Staff Management Module, Reporting Module).

**Hotel Administrators**: Have access to all 3 modules, including the User Account and Login module. Module is able to create accounts for new staff and new users.

## General Constraints

* **Limited Budget:** Final budget of S$70,000 allocated for developing the software.
* **Low-end Hardware:** Only one computer at the front desk will be running the software, computer is running windows xp support package tool, 1gb ram, 160gb hard disk.
* **Slow Connection:** Connection to the hotel module services might also be slow, seeing as it is connected to a shared broadband that is also simultaneously used to maintain the hotel’s blog website.
* **Limited Time:** A short period of time is given to code the software.

## Functional Requirements

**Room Availability and Booking Module**

* The Room Availability and Booking Module is accessible by all the three types of users. (End Users, Management Users & Administrators)
* This module allows the creation, retrieval, alteration and deletion of rooms as well as room reservations.
* This module also allows users to edit the guest records in real time. For example, after the guest has arrived and been processed, administrators and management should be able to change some of those details, for example, different room, changes in no.of adults and children staying the room due to unforeseen circumstances.
* All this information should be able to be modifiable in the system at all times.

Details of what the system should have noted down are as follows:

* **Booking Details** (Last Name, First Name , No. of adult guests and children)
* **Personal Contact Details** (Phone number , E-mail, Home and Mailing address (E.g. Street address, block and house number, postal code and country.)
* **Payment Details**  
  Credit card payment (Credit Card Number, Cardholder Name, Date of Expiration)
* **Check-In Details**  
  Check-In Date & Time  
  Desired check out date and time
* **Additional Remarks**  
  E.g. Whether guest requires king-size or queen-sized bed  
  Smoking or non-smoking room
* **Check-Out Details**  
  Whether guest requires a late checkout

**User Account Creation and Login Module**

* This module is only accessible by the system administrator.
* This module allows the account creation, retrieval, alteration and deletion of both End and Management user accounts.
* The system administrator must be present at the front desk in order for the user account creation to function.
* The administrator must login to this module with their personal username and password. After doing so, they will have the option to generate the different user accounts depending on what level the staff are registered on.
* The input required for these accounts are mentioned below in the Data Dictionary under the “staff” section.

**Housekeeping and Staff Management Module**

* This module is only accessible by the hotel owner.
* This module keeps records of staff details such as staff name, date of birth, bank account number, details of home address and phone number.
* The module also contains duty types of the staff, including general maintenance, room maintenance, estate maintenance and security.

**Reporting Module**

* This module is also only accessible to management and administrator users.
* This module generates five types of reports which are elaborated below.
* **Room Status Report**: List all the rooms in the hotel as well as their respective room status, such as
* a room is vacant or otherwise available
* a room is occupied
* a room is vacant and has been scheduled for cleaning
* **Room Guests Report**: List all the guests in one particular room (such as two adults and three children).
* **Hotel Guests Report**: List all the guests in all the rooms at any given point in time (such as all the guests in all the rooms on the 20th of January this year).
* **Room Occupancy Report**: Generate statistics indicating what was the room occupancy on a daily, weekly, monthly or yearly period.
* **Housekeeping Report**: List duties that the staff has been allocated to, generate housekeeping schedule based on a daily, weekly or monthly basis.

## Data Requirements

**Booking**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Data Format** | **Field Size** | **Description** | **Example** |
| BookingID | String | BNN | 4 | ID of Booking | B01 |
| CheckInDateTime | DateTime | DD/MM/YYYY HH:MM:SS | 10 | Date of Check-In of Guests | 29/02/2020  18:42:21 |
| CheckOutDateTime | Date | DD/MM/YYYY  HH:MM:SS | 10 | Date of Check-Out of Guests | 29/02/2020  19:42:10 |
| AdultsNum | Int | NN | 4 | Number of adult guests staying in room. | 4 |
| ChildNum | Int | NN | 8 | Number of child guests staying in room. | 8 |
| AdditionalRemarks | String | -NIL- | 500 | Extra details regarding Guests stay. | Desired Check out time and date, additional requests for pillows/toiletries/beds. Whether guest wishes to have a smoking/non-smoking room. Whether guest requires a late checkout. |
| GuestID | String | GNN | 4 | ID for Guests | G01 |
| RoomNo | Int | NN-NN | 5 | Room Number | 13-06 |
| PaymentModeID | String | PMNN | 5 | Payment Mode ID | PM01 |

**Payment**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Data Format** | **Field Size** | **Description** | **Example** |
| PaymentModeID | String | PMNN | 5 | Payment Mode ID | PM01 |
| PaymentModeName | String | -NIL- | 10 | Either Cash or Credit | Cash = PM01  Credit = PM02 |

**Guest**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Data Format** | **Field Size** | **Description** | **Example** |
| GuestID | String | GNN | 4 | ID for Guests | G01 |
| GuestFirstname | String | -NIL- | 20 | First Name for Guests | Xiu Han |
| GuestLastname | String | -NIL- | 20 | Last Name for Guests | Yip |
| GuestContactNo | Int | NNNN  NNNN | 9 | Contact Number of Guest | 6235 3535 |
| GuestCountryCallingCode | Int | NNN | 4 | Country Calling Code | 65 |
| GuestEmail | VarChar | -NIL- | 50 | E-mail Address of Active Guests staying at the hotel | wesleychng@hotmail.com |
| GuestHomeAddress | VarChar | <Street Address, Block and House Number, Postal Code, Country> | 100 | Home and Mailing address of Active Guests staying at the hotel. | 5 Simei Street 8, Blk #02-10, 529583, Singapore |
| GuestCreditCardNo | Int | NNNN NNNN NNNN NNNN | 20 | Credit Card Number of Guest | 2410 3029 4920 2042 |
| GuestCardholderName | String | -NIL- | 20 | Cardholder Name of Guest | Wesley Ch’ng |
| GuestCardExpDate | Date | MM/YYYY | 10 | Expiration Date of Guest’s Credit Card | 07/26 |

**Role**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Data Format** | **Field Size** | **Description** | **Example** |
| RoleID | String | RNN | 4 | The level of the user. (R01/R02/R03) | R01 |
| RoleName | String | -NIL- | 50 | The level of the user. (End User/Management/Administrator) | Administrator = R01  Management User = R02  End User = R03 |

**Duty**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Data Format** | **Field Size** | **Description** | **Example** |
| DutyID | String | DNN | 4 | The ID of the Duty Assigned | D01 |
| DutyName | String | -NIL- | 50 | Name of the Duty the Staff is supposed to do. | Front Desk Phonecall |
| DutyDescription | String | -NIL- | 100 | Description of the Duty\_Name | Answers the phones at the front desk. |

**Duty Schedule**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Data Format** | **Field Size** | **Description** | **Example** |
| DutyScheduleID | String | DSNN | 5 | ID of the Schedule the staff is supposed to work during | DS01 |
| StaffID | String | SNN | 4 | ID for Staff | S01 |
| DutyID | String | DNN | 4 | The ID of the Duty Assigned | D01 |
| StartDate | Date | DD/MM/YYYY | 10 | Start Date of the Schedule | 21/11/2017 |
| EndDate | Date | DD/MM/YYYY | 10 | End Date of the Schedule | 22/11/2017 |
| DutyScheduleRemarks | String | -NIL- | 100 | Additional Remarks for the Duty. | Package coming in on 21/11/2017 |

**Invoice**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Data Format** | **Field Size** | **Description** | **Example** |
| InvoiceID | String | INN | 4 | ID of Invoice | I01 |
| GuestID | String | GNN | 4 | ID for Guests | G01 |
| GuestFirstname | String | -NIL- | 20 | First Name for Guests | Xiu Han |
| GuestLastname | String | -NIL- | 20 | Last Name for Guests | Yip |
| GuestContactNo | Int | NNNN  NNNN | 9 | Contact Number of Guest | 6235 3535 |
| GuestCountryCallingCode | Int | NNN | 4 | Country Calling Code | 65 |
| GuestEmail | VarChar | -NIL- | 50 | E-mail Address of Active Guests staying at the hotel | wesleychng@hotmail.com |
| GuestHomeAddress | VarChar | <Street Address, Block and House Number, Postal Code, Country> | 100 | Home and Mailing address of Active Guests staying at the hotel. | 5 Simei Street 8, Blk #02-10, 529583, Singapore |
| AdultsNum | Int | NN | 4 | Number of adult guests staying in room. | 4 |
| ChildNum | Int | NN | 8 | Number of child guests staying in room. | 8 |
| RoomNo | Int | NN-NN | 5 | Room Number | 13-06 |
| RoomType | String | -NIL- | 15 | Room Type | Standard Room |
| RoomRate | String | -NIL- | 100 | Specific Hotel Room Rate | $35 per night |
| CheckInDateTime | DateTime | DD/MM/YYYY HH:MM:SS | 10 | Date of Check-In of Guests | 29/02/2020  18:42:21 |
| CheckOutDateTime | Date | DD/MM/YYYY  HH:MM:SS | 10 | Date of Check-Out of Guests | 29/02/2020  19:42:10 |
| NightsStayed | Int | NN | 3 | Number of Nights Stayed | 5 |
| Consumables | String | -NIL- | 100 | Minibar Foods Consumed | x1 Potato Chips  x2 Chocolate Bar |
| FinalRemarks | String | -NIL- | 500 | Extra details regarding Guests stay, Staff perspective. | Guest took something from hotel room? |
| NetPrice | Double | NNNN.NN | 10 | Final Cost of Hotel Stay | $1500.00 |
| PaymentModeName | String | -NIL- | 10 | Either Cash or Credit | Cash = PM01  Credit = PM02 |

**Staff**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Data Format** | **Field Size** | **Description** | **Example** |
| StaffID | String | SNN | 4 | ID for Staff | S01 |
| StaffUsername | String | -NIL- | 50 | Username for Staff Login | Staff123 |
| StaffPassword | String | -NIL- | 50 | Password for Staff Login | StaffPW123 |
| StaffFirstname | String | -NIL- | 20 | First Name for Staff | Wesley |
| StaffLastname | String | -NIL- | 20 | Last Name for Staff | Ch’ng |
| StaffDOB | Date | DD/MM/YYYY | 10 | Date of birth for Staff | 29/02/2020 |
| StaffBankAccountNumber | VarChar | NNN-NNNNN-N | 10 | Bank Account Number of Staff | 360-92953-2 |
| StaffHomeAddress | VarChar | -NIL- | 100 | Home Address of Staff | Simei St 5 #02-22 |
| StaffPhoneNumber | Int | NN | 10 | Contact Number of Staff | 6222 6555 |
| RoleID | String | -NIL- | 20 | Duty of Staff in the hotel | R01 |
| StaffSalary | Double | NNNN.NN | 8 | Monthly Salary of Staff | S$ 1234.50 |

**Report**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Data Format** | **Field Size** | **Description** | **Example** |
| ReportID | String | RNNN | 5 | Report ID | R124 |
| ReportType | String | -NIL- | 25 | Type of the report | Housekeeping Report |

**Room**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Data Format** | **Field Size** | **Description** | **Example** |
| RoomNo | Int | NN-NN | 5 | Room Number | 13-06 |
| RoomID | Int | NNN | 3 | Room ID | 619 |
| RoomSize | Int | NNN | 3 | Room Size (square feet) | 325 |
| RoomType | String | -NIL- | 15 | Room Type | Standard Room |
| RoomRate | String | -NIL- | 100 | Specific Hotel Room Rate | $35 per night |
| RoomBeds | String | -NIL- | 50 | The descriptions of the beds in the room. | 2x Single Bed  2x Queen Bed  2x King Bed |

**Room Availability**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Data Format** | **Field Size** | **Description** | **Example** |
| RoomID | Int | NNN | 3 | Room ID | 321 |
| Availability | Boolean | -NIL- | 5 | Check Room Availability | True |
| RoomState | Boolean | -NIL- | 5 | Display Room Clean | False |

**Inventory**

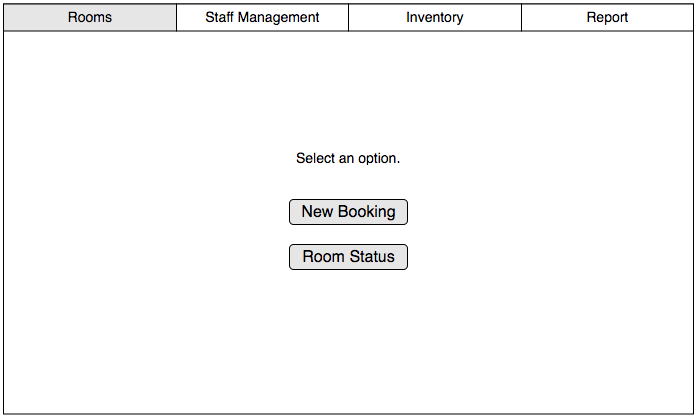
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Data Format** | **Field Size** | **Description** | **Example** |
| ItemID | Int | NNN | 3 | Item ID | 129 |
| ItemName | String | -NIL- | 10 | Item Name | Mop |

**Inventory Stock**

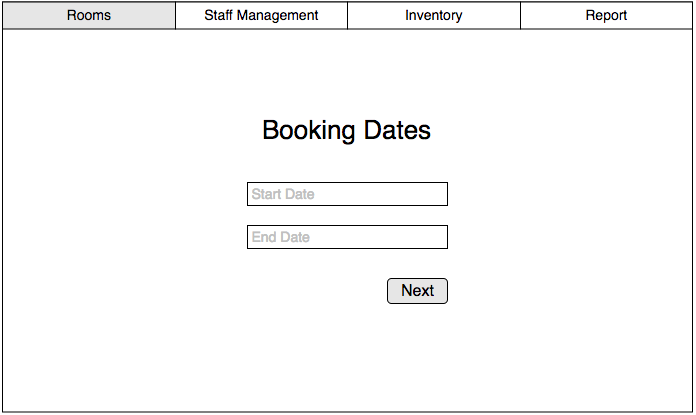
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Data Format** | **Field Size** | **Description** | **Example** |
| ItemID | Int | NNN | 3 | Item ID | 129 |
| Stock | Int | NNN | 3 | Stock Left | 17 |
| LowStockWarning | Boolean | -NIL- | 5 | Activate Low Stock Warning | True |

## User Interface Requirements

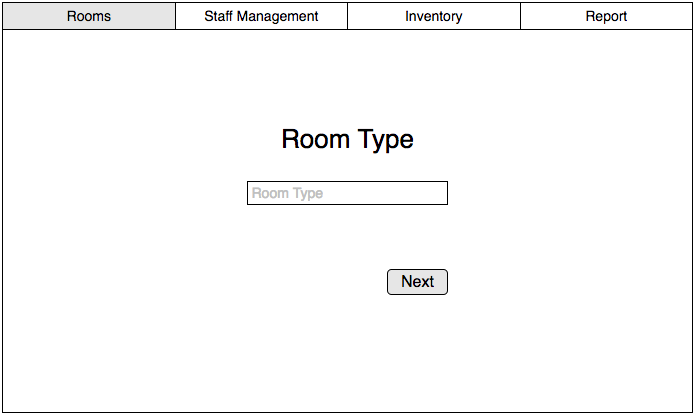
Room



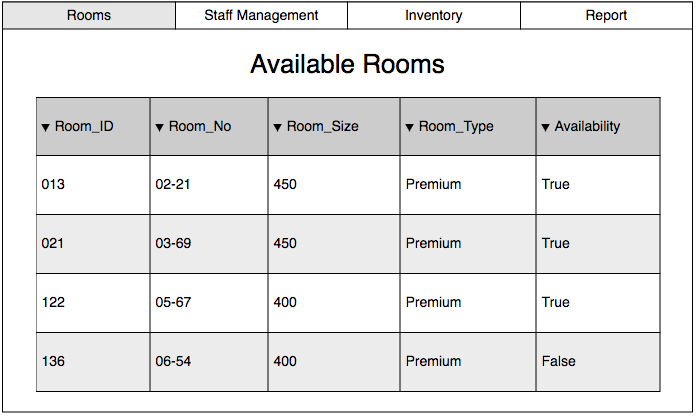
Room Booking



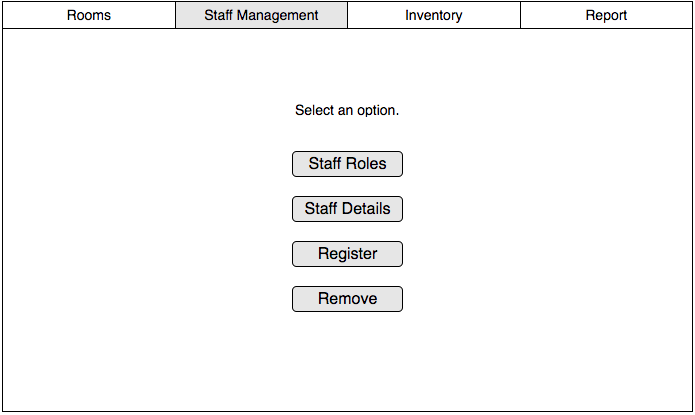
Room Type



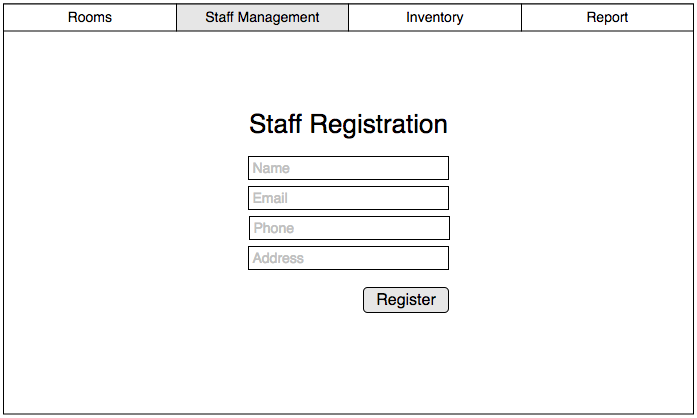
Room Availability



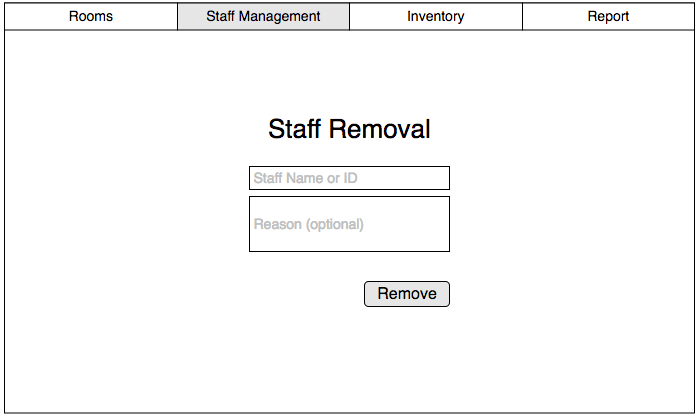
Staff Management



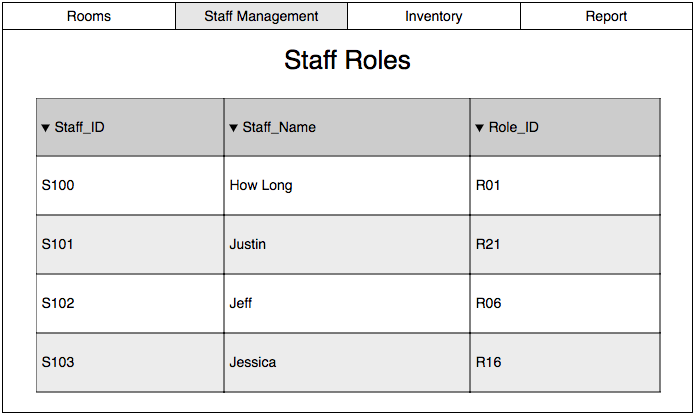
# Staff Registration



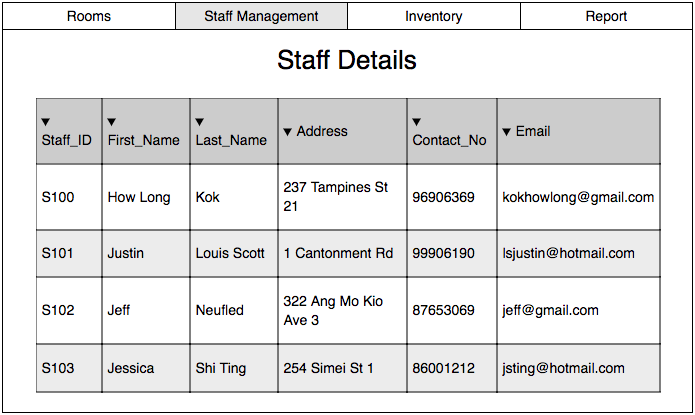
Staff Removal



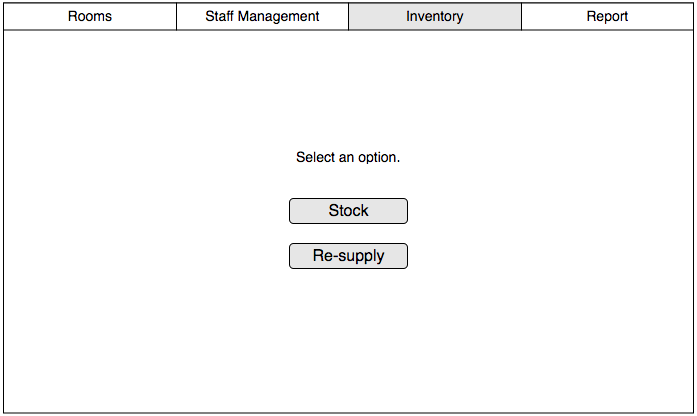
Staff Roles



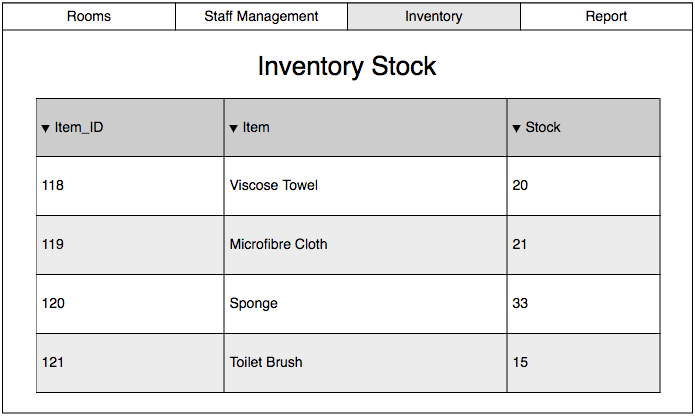
Staff Details



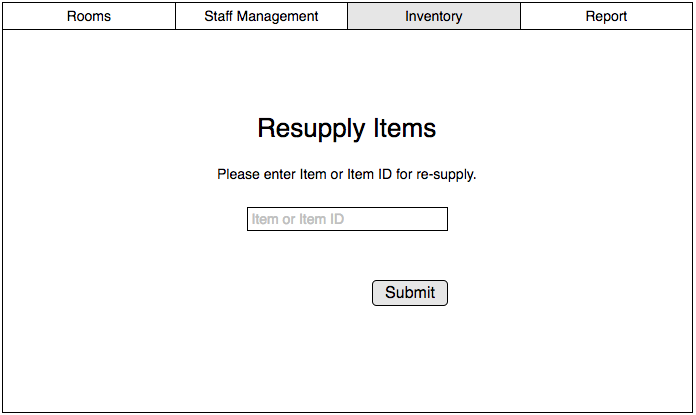
Inventory



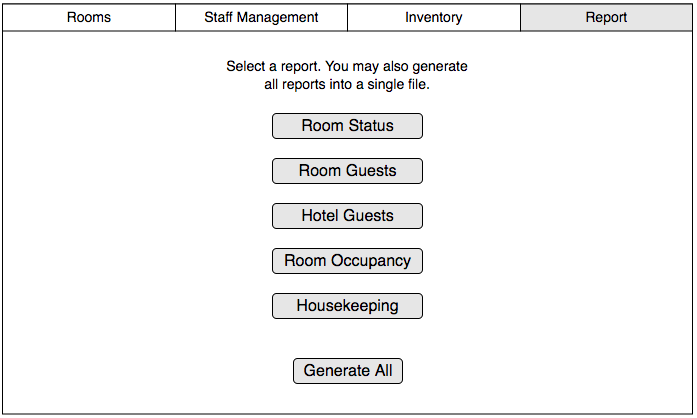
Inventory Stock



Re-supply Items



Report



## Interface with Other Systems

As the software is self dependent and will only be running on one computer on a local broadband and software, no additional systems will need to be integrated into the software.

## Assumptions

* Payment will be made at the reception counter when the guests checks out from the hotel.
* Invoice will include hotel bills and mini-bar purchases.
* Guests always check out before 12pm.
* Cleaning staff will always check the rooms if the guests have consumed food or drinks from the mini-bar.
* Only the reception staff will be using the software.
* Reception staff will enter guest’s details instead of the guests.
* The hotel has less than 1000 rooms.

# OPERATIONAL AND QUALITY REQUIREMENTS

## Operating Environment

**Hardware**

* Pentium 4 Computer 1GB RAM, monitor, keyboard, and mouse
* The computer only has a 1 gigabyte random-access-memory (RAM) and a 160 gigabyte hard disk storage.
* The computer is connected to the internet via a broadband connection.
* The computer is located at a reception counter where the staff of the hotel will be using the system.
* A laser printer will be needed to print reports and notes.
* An additional database server will be added to run the hotel management system

**Software**

* The system will only be deployed on an old computer running on Windows XP Service Pack II.
* Software is designed to run on any platform above Microsoft Windows Service Pack 2
* Microsoft .NET Frameworks 4.0 or above
* Microsoft SQL Server Management Studio Express 2010

## Development Constraints

**Group Constraints**

* Members have different schedules and therefore may not be able to meet as a group to update on each other’s progress.
* The project is to be finished in eight weeks.
* Due to the hotel not doing well, the budget for developing this project might be constrained.

**Module Constraints**

**Room availability and Booking Module**

* This module is constrained in a sense that records may only be altered manually. This module is not autonomous and records may differ.

**User Creation and Login Module**

* This module is constrained as only the administrator can perform user account and creation for end users and management users.

**Housekeeping and Staff management module**

* The records of the staff need to be manually filled out and also updated manually as and when necessary changes need to be made.

**Reporting Module**

* Report generation is based on data that is manually registered and thus may be prone to human error as a result.

## Performance (Optional, if we have time try to write response time for each button on the UI for the hotel system)

The performance of the system will not be as efficient as more popular services such as Google.com as the software will be developed using cheaper technologies and hardware, partly due to the limited budget. As such, the response time of the system can be slow. For example, searching for a room will require at least 3 seconds before the results are returned. Therefore, while the response time of the system is not as fast as Google.com, since Delonix Regia is a relatively small hotel, the longer response times should not be a major issue while running the software.

The acceptable response time for each page of the room booking process is 2-4 second on average and 2-3 seconds during peak hours.  
  
The acceptable response time for the checking of room availability function is 2-4 seconds on average and 2-3 seconds during peak hours.  
  
The acceptable response time for the housekeeping and staff management function is 2-5 seconds on average and 2-4 seconds during peak hours.  
  
The acceptable response time for the reporting function is 2-4 seconds on average and 2-3 seconds during peak hours.

## Availability

The system will be running 24/7 for 365 days, unless there are any major renovations to the hotel.

Regular backups can be done early morning from around 2am to 3am daily. Major backups can also be done on the first Sunday of each month. Backup records are to be kept and archived for 5 years.

## Security and Access Control Requirements

**Security Table**

|  |  |  |
| --- | --- | --- |
| **Name of Security** | **Description** | **Example** |
| 2FA (Two Factor Authentication) | An extra layer of security that is known as "multi factor authentication" that requires not only a password and username but also something that only, and only, that user has on them, i.e. a piece of information only they should know or have immediately to hand - such as a physical token. | E.g. Scenario when guests purchases room service. 2FA is used to confirm this purchase. |
| E-Mail Verification | E-Mail Verification will be used to verify the guest's account once their account has been created. | E.g. Scenario when guests complete their booking at the front desk, front desk staff will require users to check their e-mail using their phones in order to verify account creation. |
| E-Receipt | E-Receipts will be issued to charge guests on rooms services, culinary services, booking of facilities (suite rooms). This is done as when the guest checks out, the guest can use the E-Receipt as reference for payment purposes. | E.g. When guests checks out of hotel, they are able to compare the prices of the services they have used with the invoice issued to them at the end of their stay. |

**Access Control**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **End Users (Reception)** | **Management Users** | **Administrators** |
| Login/Logout | ✓ | ✓ | ✓ |
| Room Availability and Booking | ✓ | ✓ | ✓ |
| Housekeeping and Staff | ✓ | ✓ | ✓ |
| Report Generation |  | ✓ | ✓ |
| User Account and Creation |  |  | ✓ |

# SPECIAL REQUIREMENTS

One of the requested special requirement of the software is the ability for guests to be able to inquire about the room availability of the Delonix Regia hotel online.

Another additional special requirement of the software is the ability for the generated reports from the Reporting Module to be exported to Microsoft Excel and previewed before printing.

# REFERENCES

It All Starts with Your Stunning Website. (n.d.). Retrieved from

<https://www.wix.com/>

What is 2FA? (n.d.). Retrieved November 13, 2017, from

<https://www.securenvoy.com/two-factor-authentication/what-is-2fa.shtm>

**Temasek Polytechnic**

**School of Informatics and IT**

**Diploma in Information Technology (IT)**

Software Design Specifications (DS)

**Project Particulars**

|  |  |
| --- | --- |
| **Tutor** | Qi Yutao |
| **Class** | P01 |
| **Project Title** | Delonix Regia Hotel Management System |

**Project Team’s Particulars**

|  |  |
| --- | --- |
| **Matric Number** | **Student Name** |
| 1601553I | Yip Xiu Han |
| 1603625C | Ch’ng Wai Kit Wesley |
| 1601705E | Qamarul Fattah Bin Hamdan |

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 13/11/2017 | 1.0 | Initial Draft | ALL |
| 21/11/2017 | 1.1 | Completed Document | ALL |
| 22/11/2017 | 1.2 | Vetted Document | Wesley |
| 22/11/2017 | 1.3 | Compiled Document | Wesley |

**Table of Contents**

[**1.**](#_gjdgxs) **DISTRIBUTION OF WORKLOAD 36**

[**2.**](#_30j0zll) **ARCHITECTURE DESIGN 36**

[**3.**](#_1fob9te) **USER INTERFACE (UI) DESIGN 39**

[**4.**](#_3znysh7) **PROGRAM DESIGN 46**

[**5.**](#_2et92p0) **DATABASE DESIGN 54**

1. **DISTRIBUTION OF WORKLOAD**

|  |  |
| --- | --- |
| **Design** | **Members** |
| Architecture Design | Wesley |
| User Interface (UI) Design | Qamarul |
| Program Design | ALL |
| Database Design | ALL |

1. **ARCHITECTURE DESIGN**

**Multitier Architecture Background**

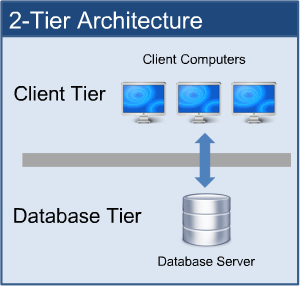
In software engineering, a multitier architecture (often referred to as n-tier architecture) or multilayered architecture is a client–server architecture in which presentation, application processing, and data management functions are physically separated.

The three most common layers in a multitier architecture are:

* Presentation Layer - The Presentation layer which is also known as the client layer which is the top most layer of an application. This layer is used to allow the users to view the application and to access the web pages. The main function of this layer is to communicate with the application layer. The presentation layer passes the information which is entered by the user to the application layer. For example, the login page in a hotel management system will require the user to enter values into the text fields of “Username” and “Password”, of which this information is passed onto the application layer.
* Application/Business Layer - The Application layer which is also known as the Business layer which contains the written business logic. For example, once the user keys in their login information into the provided text fields, the application layer interacts with the data layer which sends the login information to the presentation layer. The application layer provides users with the ability to carry out operations on the application. It also controls the application’s functionality by performing detailed processing. This layer acts as a medium between the presentation layer and the data layer.
* Data Access Layer - The Data Access layer consists of the application’s database. The application layer communicates with the data layer to request for the retrieval of data. It contains methods that connects the database and performs the required functions such as the create, retrieve, update and delete.

**Proposed System Architecture Design**

After careful analysis, our team has decided on implementing the 2-tier architecture design into this hotel management system.



The above is an example of a 2-tier architecture system. It is designed for a client-server communication. The application must be connected to a network, such a Azure SQL database, in order to function properly. The presentation layer and the application layer will be on the client side while the data layer will be on the server side. The “client” will be the program that is running on the hotel management device used by the hotel staffs to capture and upload guest information while the “server” will be the machine that the client gets content from. Many clients would be able to access the information on a single server thus, this architecture design is suitable for a hotel management system.

**Advantages of 2-tier architecture:**

Greater Performance - The database layer and application/business layer is physically close which offers a greater performance hence, allowing response time and communication to happen at a faster rate.

Navigation - User maneuverability (for both staff and developers) with a 2-tier system is easily achieved as applications can be easily developed, modified and maintained.

**Disadvantages of 2-tier architecture:**

Scalability - This architecture lacks scalability as it only supports a very limited number of users. This architecture will not be suitable if Hotel Delonix Regia continues to expand in size as a very limited number of user connections can be supported before application performance is degraded as clients require separate connections and CPU memory to proceed.

Circumstantial usage - Businesses with rapidly changing rules and regulations are not suitable since the database server needs to handle the business logic which slows down database performance.

Alteration of Database Server - As most applications which are used for interaction is dependent on the database structure, this creates an issue if the owners of Hotel Delonix Regia decide to redesign the UI. As the database structure would have to be re-designed as well since it corresponds with the UI.

**Summary**

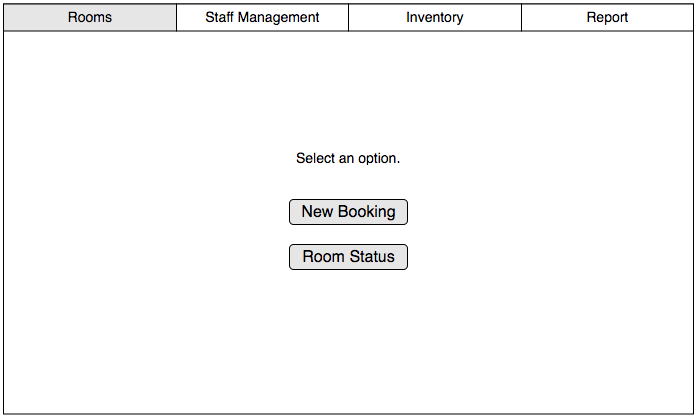
Hence, the 2-tier architecture design is recommended for the hotel management system as we plan on using a web form.

The presentation and business layer will be installed on the client’s machines that will access the Azure SQL Database so that there will be the least amount of downtime. The advantages of the 2-tier architecture have been discussed above and we have found out that this architecture design is useful for a hotel management system as multiple staff will be able to access and update the system at the same time.

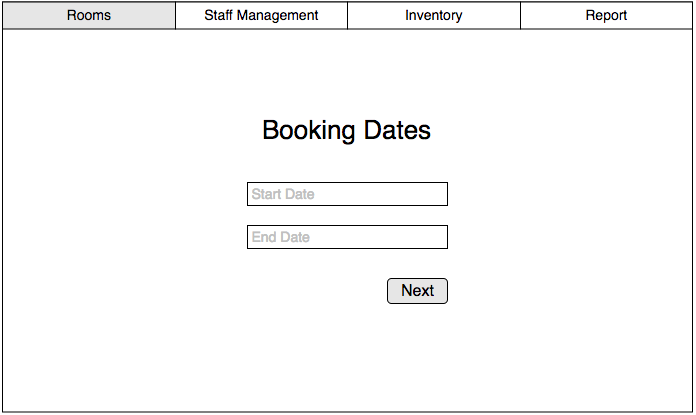
The 2-tier architecture would be able to meet Mr Lim’s requirements and it is less complex than a 3-tier architecture and thus, it can be easily implemented maintained.

1. **USER INTERFACE (UI) DESIGN**

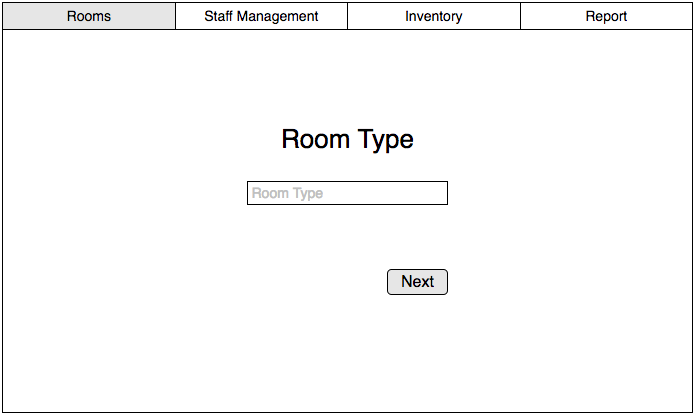
Room



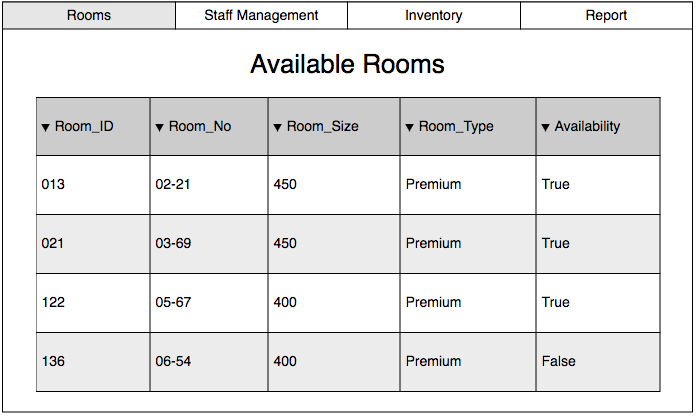
Room Booking



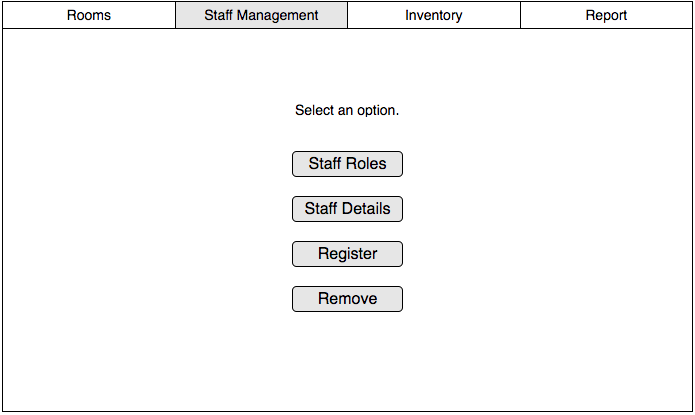
Room Type



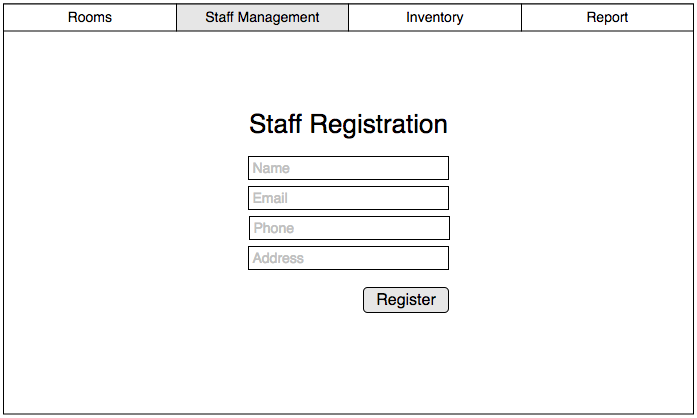
Room Availability



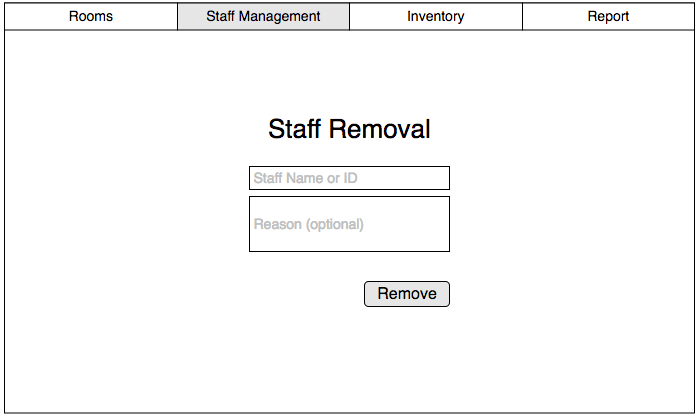
Staff Management



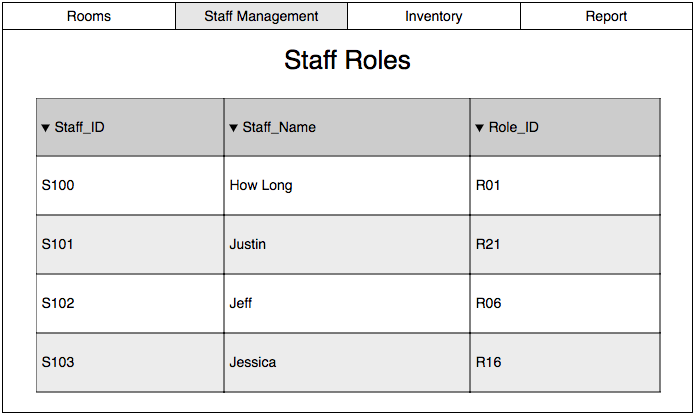
Staff Registration



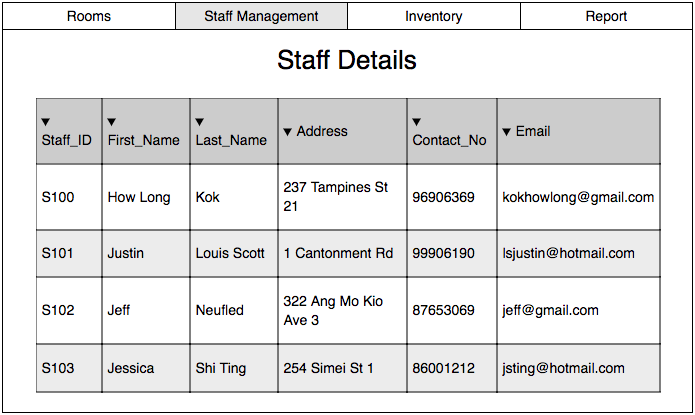
Staff Removal



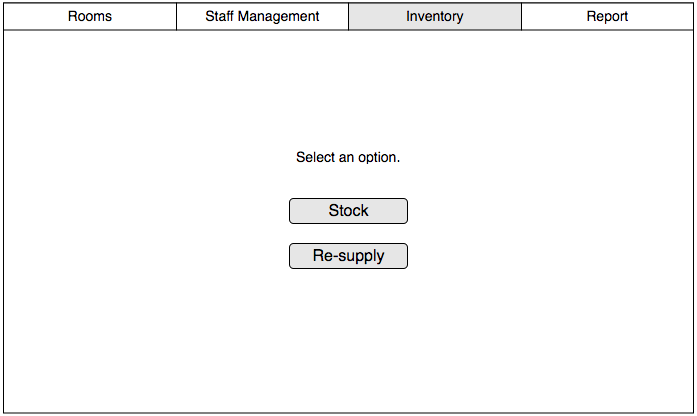
Staff Roles



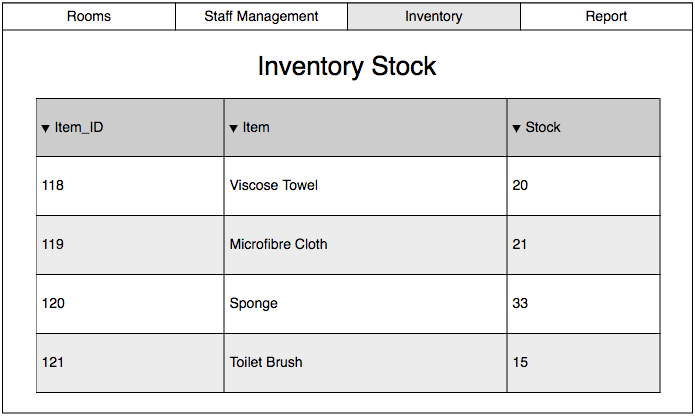
Staff Details



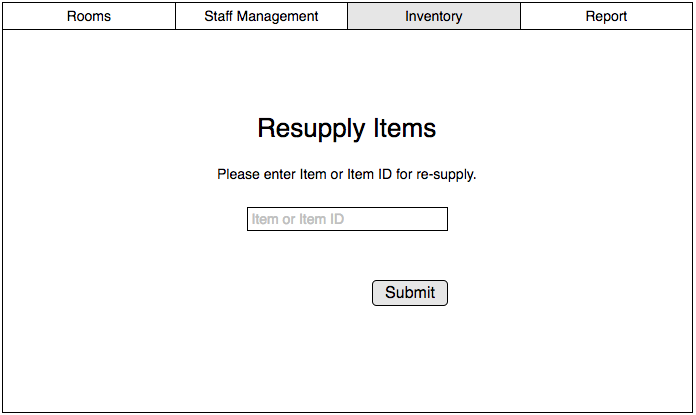
Inventory



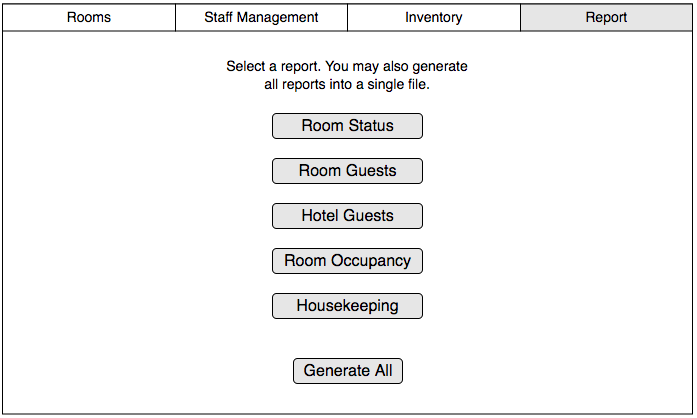
Inventory Stock



Resupply Items



Report



1. **PROGRAM DESIGN**

The development of the HMS will be split up into two groups, one will be programming the logic and the functions of the system, the other group will be designing the UI. Due to time constraint, these two development tasks will be carried out together, afterwards we will merge the interface with the codes at the end of the software development.  
  
We will be using object oriented way of programming to organise various variables/attributes, and use a database helper class to access the database.

**Module 1 - Room Availability and Booking Module (Wesley):**

**Use Case 1 : Check-In (List of Rooms)**

**Brief Description**

This use case occurs when the End Users / Management Users / Administrators wishes to check a guest into the hotel management system.

**Actors(s)**

End Users / Management Users / Administrators

**Main Flow**

1. This use case begins when a user selects the check in option on the hotel management system.
2. System displays the check in page.
3. System prompts the user to enter the guest’s room number.
4. User keys in the guest’s room number.
5. System displays the existing bookings for that specific room.
6. User asks for the guest’s name.
7. User checks for booking(s) under the guest’s name.
8. User selects the row that has the guest’s name.
9. User checks the other booking details with the guest (E.g. Name, No. of Guests, Room Details).
10. User presses the check in button.
11. System prompts the user for confirmation.
12. User selects confirm check-in button.
13. System saves the guest’s information and the room status is changed from “Booked” to “Occupied”. The use case ends here.

**Alternate Flow(s)**

Guest name recurs

7.1. User clarifies with guest on other booking details (E.g. Room Details) to get the correct information.

7.2. User continues from step 8.

Guest’s details are non-existent

7.1. User clarifies with guest on whether they have successfully booked a room (likelihood answer will be no).

7.2. User moves onto Use Case 2.

Guest’s details differ from database

7.1. User clarifies with guest on booking details.

7.2. User moves onto Use Case 5.

**Use Case 2: Check-In (Registration)**

**Brief Description**

This use case occurs when the End Users / Management Users / Administrators wishes to register a guest’s booking into the hotel management system.

**Actors(s)**

End Users / Management Users / Administrators

**Main Flow**

1. This use case begins when a user selects the new booking option in the check-in module of hotel management system.
2. System prompts the user to enter the start date and end date the guest wishes to book for.
3. User asks the guest for the booking dates.
4. User keys in the booking dates.
5. System prompts user to select the room type.
6. User asks the guest for their room preference.
7. User selects the guest’s room preference.
8. System displays rooms that are available on the booking dates that coincides with the guest’s room preference.
9. User selects a room number for the guest.
10. System prompts the user to enter the room, customer and booking details.
11. User asks the guest for their details.
12. User keys in the room, customer and booking details.
13. System prompts the user on whether they would require any additional requirements (E.g. Extra Pillow).
14. User asks the guest if they would require any additional requirements.
15. User keys in the special requirements.
16. System prompts the user for payment.
17. User asks the guest if they would like to pay by cash or card.
18. User collects the cash from the guest.
19. System prompts the user to confirm the booking.
20. User selects the confirm booking button.
21. System saves the information the database and changes the room status of the room from “Available” to “Booked”. Use case ends.

**Alternate Flow(s)**

Guest pays by card.

18.1. User prompts the guest to insert card into the payment terminal and enter pin.

18.2. User hands the card back to the guest and clarifies transactional information.

18.3. User continues from step 19.

Invalid information entered (can occur from any step from step 4 onwards)

4.1. User clarifies with the guest again on whatever information is required.

4.2. User re-enters the information and continues with the flow of the use case.

Booking already exists.

20.1. System displays booking information already exists.

20.2. User rectifies the problem by clarifying with guest on whether they have already made a booking and other booking details.

**Use Case 3: Check-Out**

**Brief Description**

This use case occurs when the End Users / Management Users / Administrators wishes to check out a guest from the hotel management system.

**Actors(s)**

End Users / Management Users / Administrators

**Main Flow**

1. This use case begins when a user selects check-out module of hotel management system.
2. System displays form interface and prompts the user to select the room number.
3. User selects the room number.
4. System takes the values of the room number and displays the details on the form interface.
5. User adds new row in items consumed.
6. System takes in the values and shows the row in the form.
7. System will calculate the total amount through the quantity of items consumed E.g. (Minibar) and price.
8. User clicks generate invoice button.
9. System prompts for confirmation.
10. User confirms the action.
11. User keys in the room, customer and booking details.
12. System generates and updates the invoice in the database).
13. System changes the room status from “Occupied” to “Available but not cleaned”. Use case ends here.

**Alternate Flow(s)**

Requested room check-out not in use.

3.1. User clarifies with customer if their desired checkout number of room is correct.

3.2. User repeats step 3.

**Use Case 4: Room (Current Room Status)**

**Brief Description**

This use case occurs when the End Users / Management Users / Administrators wishes view the current room status.

**Actors(s)**

End Users / Management Users / Administrators

**Main Flow**

1. This use case begins when a user selects room module of hotel management system.
2. System displays form interface and prompts user to enter room number to search.
3. User keys in the room number.
4. System takes in the value of the room number and displays the room details with status on the form interface. Use case ends here.

**Alternate Flow(s)**

Entered room number not found.

4.1. System displays “Room number not found” error message.

4.2. System redirects user to Step 2.

**Use Case 5: Room (Edit Details of Room)**

**Brief Description**

This use case occurs when the End Users / Management Users / Administrators wishes to edit the details of a room using the hotel management system.

**Actors(s)**

Management Users / Administrators

**Main Flow**

1. This use case begins when a management user/administrator selects selects the edit room details option the room module of hotel management system.
2. System prompts the user to enter the room number and room type he/she wishes to search for.
3. Manager/Administrator keys in room number and room type and clicks search.
4. System takes in the values of the inputs and displays the result in the form of a table.
5. Manager/Administrator clicks on the specific row of the table.
6. System displays the details of the room on the “details” tab.
7. Manager/Administrator updates the text fields of the room details and clicks the update button.
8. System prompts for confirmation.
9. Manager/Administrator selects confirm.
10. System saves the changes and updates the database. Use case ends.

**Alternate Flow(s)**

Entered room number not found.

4.1. System displays “Room number not found” error message.

4.2. System redirects user to Step 3.

**Module 2 - Housekeeping and Staff Management Module (Qamarul):**

**Main Flow:**

1. The use case starts when the user (hotel owner) wants to register or remove a staff
2. System prompts user ‘Register’ or ‘Remove’
3. If user selects ‘Register’, go to Register Alternate Flow
4. If user selects ‘Remove’, go to Remove Alternate Flow

**Register Alternate Flow:**

1. System displays a staff registration form
2. User enters staff details, and submits form
3. System updates Staff table

**Remove Alternate Flow:**

1. System displays a form, prompts user to enter Staff ID
2. User enters Staff ID
3. System prompts user to confirm staff removal
4. User confirms and submits form
5. System removes entered Staff ID from Staff table.

**Module 3 - Reporting Module (Xiu Han):**

**Main Flow:**

1. The use case starts when the user (staff) wishes to generate a type of report for the hotel.
2. If user selects the report generation for the status of the rooms in the hotel, go to Room Status Report alternate flow.
3. If user selects the report generation for listing the guests in a room, go to Room Guests alternate flow.
4. If user selects the report generation for listing all the guests in the hotel, go to Hotel Guests alternate flow.
5. If user selects the report generation for checking the statistics of room occupancy in the hotel, Room Occupancy alternate flow.
6. If user selects the report generation for Housekeeping report generation, go to Housekeeping alternate flow.

**Room Status Report Alternate Flow:**

1. User searches for an existing room in the hotel by its room number to modify its status.
2. User keys into the system whether the selected room is vacant, occupied, or scheduled for cleaning.
3. User repeats this process for as many times as needed.
4. User generates the report.

**Room Guests Report Alternate Flow:**

1. User keys into the system the room number of the hotel room.
2. User logs the name of the guests in the room along with other details.
3. User generates the report.

**Hotel Guests Report Alternate Flow:**

1. User selects the rooms by their room numbers to be compiled for the report (to account for the guests in those rooms).
2. User generates the report.

**Room Occupancy Report Alternate Flow:**

1. User selects an existing room in the hotel by its room number.
2. System returns the number of times the room was occupied during that current month.
3. User generates the report.

**Housekeeping Report Alternate Flow:**

1. User selects a staff in the hotel by their ID.
2. User keys in what duties the staff is assigned to.
3. User generates the report.
4. **DATABASE DESIGN**

**Database Schema**

**Booking** (**BookingID**, CheckInDateTime, CheckOutDateTime, AdultsNum, ChildNum, AdditionalRemarks, GuestID, RoomNo, PaymentModeID)

**Payment** (**PaymentModeID**, PaymentModeName)  
 **Guest** (**GuestID**, GuestFirstname, GuestLastname, GuestContactNo, GuestCountryCallingCode, GuestEmail, GuestHomeAddress, GuestCreditCardNo, GuestCardholderName, GuestCardExpDate)

**Role** (**RoleID**, RoleName)

**Duty** (**DutyID**, DutyName, DutyDescription)

**DutySchedule** (**DutyScheduleID**, StaffID, DutyID, StartDate, EndDate, DutyScheduleRemarks)

**Invoice** (**InvoiceID**, GuestID, GuestFirstname, GuestLastname, GuestContactNo, GuestCountryCallingCode, GuestEmail, GuestHomeAddress, AdultsNum, ChildNum, RoomNo, RoomType, RoomRate, CheckInDateTime, CheckOutDateTime, NightsStayed, Consumables, FinalRemarks, NetPrice, PaymentModeName)

**Staff** (**StaffID**, StaffFirstname, StaffLastname, StaffDOB, StaffBankAccountNumber, StaffHomeAddress, StaffPhoneNumber, RoleID, StaffSalary)

**Report** (**ReportID**, ReportType)

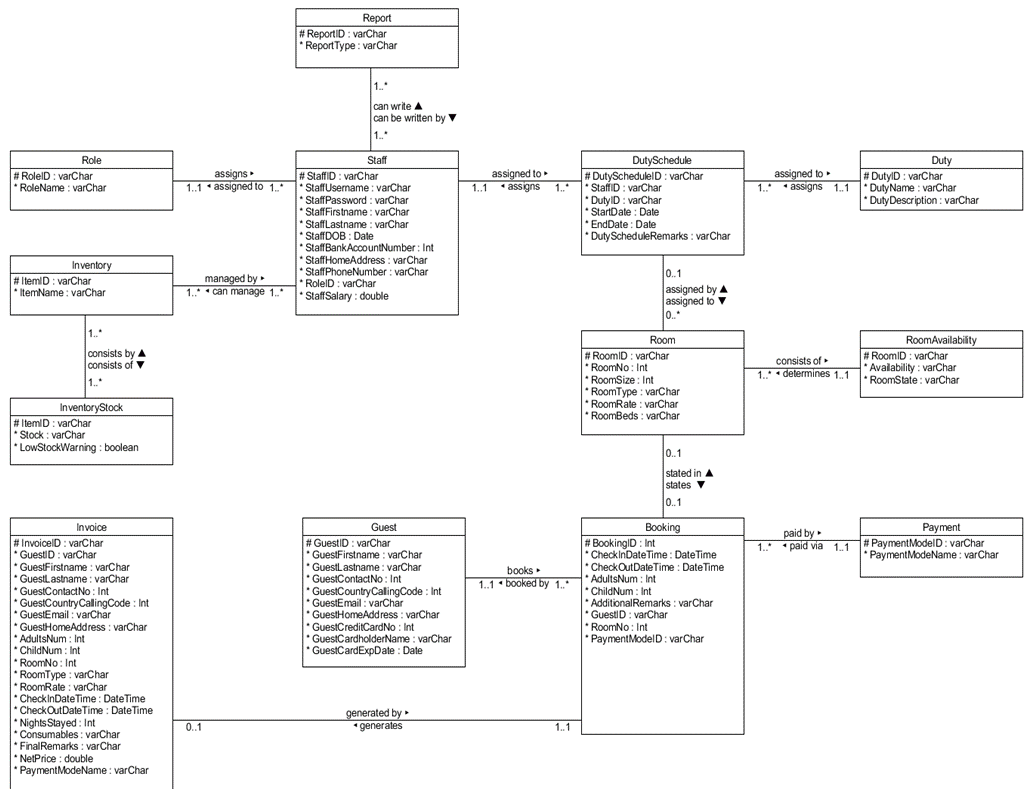
**Room** (**RoomID**, RoomNo, RoomSize, RoomType, RoomRate, RoomBeds)

**RoomAvailability** (**RoomID**, Availability, RoomState)

**Inventory** (**ItemID**, ItemName)

**InventoryStock** (**ItemID**, Stock, LowStockWarn)

**Database Table**



1. **References**

Multitier architecture. (2017, November 11). Retrieved November 21, 2017, from

<https://en.wikipedia.org/wiki/Multitier_architecture>

What is Difference Between Two-Tier and Three-Tier Architecture? (2013, January 06).

Retrieved November 21, 2017, from <http://www.softwaretestingclass.com/what-is-difference-between-two-tier-and-three-tier-architecture/>

What is Difference between Two-Tier and Three-Tier Architecture? (2016, August 01).

Retrieved November 21, 2017, from <https://www.tuturself.com/posts/view?menuId=90&postId=465>

Temasek Polytechnic

School of Informatics and IT

**Diploma in Information Technology (IT)**

Team/Peer Evaluation

|  |  |
| --- | --- |
| **Project Title:**  Delonix Regia Hotel Management System | |
| **Student No: 1603625C Reg 03** | **Student Name: Ch’ng Wai Kit Wesley** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rate the overall team performance against each criterion. Circle one number from  1 (inadequate) to 5 (superior) | | | | | |
| Team spirit | **1** | **2** | **3** | **4** | **5** |
| Overall effectiveness | **1** | **2** | **3** | **4** | **5** |
| Rewarding experience | **1** | **2** | **3** | **4** | **5** |
| Team productivity | **1** | **2** | **3** | **4** | **5** |
| Process quality | **1** | **2** | **3** | **4** | **5** |
| Product quality | **1** | **2** | **3** | **4** | **5** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rate the contribution of each team member (including yourself). Circle one number from  1 (inadequate) to 5 (superior) | | | | | |
| Myself | **1** | **2** | **3** | **4** | **5** |
| Qamarul Fattah Bin Hamdan | **1** | **2** | **3** | **4** | **5** |
| Yip Xiu Han | **1** | **2** | **3** | **4** | **5** |
| ~~<State name of Team Member 3>~~ | **1** | **2** | **3** | **4** | **5** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rate the quality of work (including timeliness) of each team member (including yourself). Circle one number from 1 (inadequate) to 5 (superior) | | | | | |
| Myself | **1** | **2** | **3** | **4** | **5** |
| Qamarul Fattah Bin Hamdan | **1** | **2** | **3** | **4** | **5** |
| Yip Xiu Han | **1** | **2** | **3** | **4** | **5** |
| ~~<State name of Team Member 3>~~ | **1** | **2** | **3** | **4** | **5** |
| Rate the help and support you have received from each team member. For yourself, rate the support and help you have given to other team members. Circle one number from  1 (inadequate) to 5 (superior) | | | | | |
| Myself | **1** | **2** | **3** | **4** | **5** |
| Qamarul Fattah Bin Hamdan | **1** | **2** | **3** | **4** | **5** |
| Ch’ng Wai Kit Wesley | **1** | **2** | **3** | **4** | **5** |
| ~~<State name of Team Member 3>~~ | **1** | **2** | **3** | **4** | **5** |

|  |
| --- |
| **Comments:** |
| **It was quite a daunting task to finish so much work in such a little timeframe.** |
| **However, I’m glad that my teammates were there to back me up when needed.** |
|  |
|  |
|  |
|  |
|  |

**Signature: Ch’ng Wai Kit Wesley Date: 21/11/2017**

Temasek Polytechnic

School of Informatics and IT

**Diploma in Information Technology (IT)**

Team/Peer Evaluation

|  |  |
| --- | --- |
| **Project Title:**  Delonix Regia Hotel Management System | |
| **Student No: 1601705E** | **Student Name: Qamarul Fattah Bin Hamdan** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rate the overall team performance against each criterion. Circle one number from  1 (inadequate) to 5 (superior) | | | | | |
| Team spirit | **1** | **2** | **3** | **4** | **5** |
| Overall effectiveness | **1** | **2** | **3** | **4** | **5** |
| Rewarding experience | **1** | **2** | **3** | **4** | **5** |
| Team productivity | **1** | **2** | **3** | **4** | **5** |
| Process quality | **1** | **2** | **3** | **4** | **5** |
| Product quality | **1** | **2** | **3** | **4** | **5** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rate the contribution of each team member (including yourself). Circle one number from  1 (inadequate) to 5 (superior) | | | | | |
| Myself | **1** | **2** | **3** | **4** | **5** |
| Wesley | **1** | **2** | **3** | **4** | **5** |
| Xiu Han | **1** | **2** | **3** | **4** | **5** |
|  | **1** | **2** | **3** | **4** | **5** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rate the quality of work (including timeliness) of each team member (including yourself). Circle one number from 1 (inadequate) to 5 (superior) | | | | | |
| Myself | **1** | **2** | **3** | **4** | **5** |
| Wesley | **1** | **2** | **3** | **4** | **5** |
| Xiu Han | **1** | **2** | **3** | **4** | **5** |
|  | **1** | **2** | **3** | **4** | **5** |
| Rate the help and support you have received from each team member. For yourself, rate the support and help you have given to other team members. Circle one number from  1 (inadequate) to 5 (superior) | | | | | |
| Myself | **1** | **2** | **3** | **4** | **5** |
| Wesley | **1** | **2** | **3** | **4** | **5** |
| Xiu Han | **1** | **2** | **3** | **4** | **5** |
|  | **1** | **2** | **3** | **4** | **5** |

|  |
| --- |
| **Comments:** |
|  |
|  |
|  |
|  |
|  |
|  |
|  |

**Signature: Qamarul Fattah**

**Date: 07 Nov 2017**

Temasek Polytechnic

School of Informatics and IT

**Diploma in Information Technology (IT)**

Team/Peer Evaluation

|  |  |
| --- | --- |
| **Project Title:**  Delonix Regia Hotel Management System | |
| **Student No: 23** | **Student Name: Yip Xiu Han** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rate the overall team performance against each criterion. Circle one number from  1 (inadequate) to 5 (superior) | | | | | |
| Team spirit | **1** | **2** | **3** | **4** | **5** |
| Overall effectiveness | **1** | **2** | **3** | **4** | **5** |
| Rewarding experience | **1** | **2** | **3** | **4** | **5** |
| Team productivity | **1** | **2** | **3** | **4** | **5** |
| Process quality | **1** | **2** | **3** | **4** | **5** |
| Product quality | **1** | **2** | **3** | **4** | **5** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rate the contribution of each team member (including yourself). Circle one number from  1 (inadequate) to 5 (superior) | | | | | |
| Myself | **1** | **2** | **3** | **4** | **5** |
| Qamarul Fattah Bin Hamdan | **1** | **2** | **3** | **4** | **5** |
| Ch’ng Wai Kit Wesley | **1** | **2** | **3** | **4** | **5** |
| ~~<State name of Team Member 3>~~ | **1** | **2** | **3** | **4** | **5** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rate the quality of work (including timeliness) of each team member (including yourself). Circle one number from 1 (inadequate) to 5 (superior) | | | | | |
| Myself | **1** | **2** | **3** | **4** | **5** |
| Qamarul Fattah Bin Hamdan | **1** | **2** | **3** | **4** | **5** |
| Ch’ng Wai Kit Wesley | **1** | **2** | **3** | **4** | **5** |
| ~~<State name of Team Member 3>~~ | **1** | **2** | **3** | **4** | **5** |
| Rate the help and support you have received from each team member. For yourself, rate the support and help you have given to other team members. Circle one number from  1 (inadequate) to 5 (superior) | | | | | |
| Myself | **1** | **2** | **3** | **4** | **5** |
| Qamarul Fattah Bin Hamdan | **1** | **2** | **3** | **4** | **5** |
| Ch’ng Wai Kit Wesley | **1** | **2** | **3** | **4** | **5** |
| ~~<State name of Team Member 3>~~ | **1** | **2** | **3** | **4** | **5** |

|  |
| --- |
| **Comments:** |
| **We were able to complete substantial portions of the project by meeting up and discussing and dividing our workloads.** |
|  |
|  |
|  |
|  |
|  |
|  |

**Signature: \_\_\_\_\_\_\_\_\_\_\_\_Xiu Han\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_21/11/2017\_\_\_\_\_\_**