

**Project Manager:**

Marianna Hollanda Campos Pedroso

**Date:**

02-28-2022

SleepEasy Hotel

Registry System

**APPROVAL PAGE**

Joey Kitson, Client

SleepEasy Hotel Registry System

SHRS

This document is to receive the approval of the client, *Joey Kitson*, and the project manager, *Marianna Hollanda Campos Pedroso*, for the proposed project. By signing this document, both parties acknowledge they have read the document in its entirety and agree to the plans in this document. If the proposed plan is accepted, date and sign below to move forward with the proposed object outlined in this document.

|  |  |  |  |
| --- | --- | --- | --- |
| Approver Title | Approver Name | Signature | Date |
| Client | Joey Kitson |  |  |
| Project Manager | Marianna H C Pedroso |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version | Document Name | Date | Author | Comments |
| 1.0 | CIS2261\_ProjectAnalysisDesign\_Azam\_Badal\_Karina\_Marianna | 01-06-2022 | Marianna Pedroso | First Draft |
| 2.0 | CIS2261\_ProjectAnalysisDesign\_Azam\_Badal\_Karina\_Marianna | 02-14-2022 | Marianna Pedroso | Second Draft |
| 3.0 | CIS2261\_ProjectAnalysisDesign\_Azam\_Badal\_Karina\_Marianna | 02-28-2022 | Marianna Pedroso | Final Document |

**DOCUMENT TRACKING**

Table of Contents

**DOCUMENT TRACKING3**

**MANAGEMENT SUMMARY6**

**SYSTEM OVERVIEW7**

System Statement7

System Deliverables/Assumptions/Constraints7

Deliverables 7

Non-Deliverables8

Assumptions8

Constraints8

System Key Requirements9

System Modeling11

Event Table11

Domain Class Diagram13

Information Management – Create new Service14

Use Case Diagram14

Use Case 15

System Sequence Diagram16

Information Management – Logging into the system16

Use Case Diagram16

Use Case 17

System Sequence Diagram19

Information Management – Add or Remove staff19

Use Case Diagram20

Use Case 21

System Sequence Diagram21

**SYSTEM** **COMPONENTS DETAILS**22

**Program Design**22

Design Class Diagram22

Output Design22

Report (Output) – Room List22

Mock Up23

Report (Output) – Guest List24

Mock up25

Report (Output) – Staff25

Mock up26

Input Design27

Guest Registration 27

Add New Room Service Request28

Edit Staff Member 29

Database Design30

ERD30

Database Design Language (DBDL)30

Data Dictionary32

**SUPPORT PROCESSING DESIGN 34**

**ENVIRONMENTAL REQUIREMENTS35**

**IMPLEMENTATION REQUIREMENTS36**

**APPENDIX 38**

Appendix A: Team Contract/Charter 38

Appendix B: Change Request/Notification 43

Appendix C: Progress Status Report 44

**MANAGEMENT SUMMARY**

This document contains the overall system design for the SleepEasy Hotel Registration (SEHR). This system will allow manipulation of data depending on the access level (admin/regular staff). This document will detail the technical requirements to SEHR and will also serve as a development guide for the system and its parts with the use of analysis and design diagrams. The system includes management of:

* + Customer Registration
  + Room Service – a record of all open room services requests with names of staff members who are assigned to each request
  + Service data – will show all available services in the hotel
  + Staff data – will show each staff member with service they recorded in
  + Room inventory

As with any system that handles personally identifiable information, security will be critical. Securely implementing and staff training will be paramount to the successful deployment and operation of the SHRS system.

**SYSTEM OVERVIEW**

## System Statement:

The SleepEasy Hotel Registry System will provide full operational flexibility to users within each interface screen (add, update, delete).

The reports will include:

**Full room listing** - showing availability, type, rate, number, and description of each room. The description will be used to hold the information of any specifications or fixes that must be done before the next guest will be moved in.

**Guest checkout bill**

**Guest listing** – the alphabetical list of all current guests with their room numbers

**Services** - available by type of service listed alphabetically

**Staff report** - showing each staff member, employee number, and title.

System Deliverables/Assumptions/Constraints:

System Deliverables

|  |  |  |
| --- | --- | --- |
|  | Deliverable | Description |
| **1** | **Code and Requirements** |  |
|  | Database scripts | SQL scripts for the database to be used for the system. |
|  | Source Code and Executable | Application and source code for the application needed to run the system. |
|  | Customer Interaction System | Source code for customer interaction system to be used. |
|  | Test Plan | Documentation to ensure all system components meet the requirements. |
| **2** | **Documentation** |  |
|  | Installation Guide | Instruction manual for setup of the system in the office. |
|  | User Manual | Manual for staff members to use the system. |
|  | Training Modules | Training Modules for Staffs |

Non-Deliverables:

**Hardware** – Company will provide its own hardware required for the system to be installed and run.

**Hosting/Domain (**if web-based**) –** Company will provide its own hosting and domain plans for the web-based system.

Assumptions:

* Hardware and network infrastructure must meet the minimum requirements for the system to run without any problems. Higher system requirements are recommended but not required.
* All the equipment for testing will be available as required.
* The team will write the solution in C#
* Company is responsible for backing up data.
* Training will be conducted internally with no additional training costs incurred

Constraints:

**Time**: The SleepEasy Hotel Registry System must be completed, deployed and staff needs to be trained by April 22nd. deliverable due date.

**Cost**: The client and project manager must authorize any changes to scope related to timeline and budget.

**Scope**: Whole scope of work has been shared between team members. Also, team members are trained to back up each other on any task.

System Key Requirements:

* Inputs
  + Room Inventory
  + Customer Registration
  + Room Service
  + Service
  + Staff
* Outputs
  + Reports
    - Full room listing - showing availability
    - Guest checkout bill
    - Guest listing registered in what room alphabetical
    - Services available by type of service listed alphabetical
    - Staff showing each staff and what they recorded in services
* Data
  + Booking data
  + BookingStatus data
  + Guest data
  + Rooms data
  + Beds data
  + Staff data
  + Position data
  + Payments data
  + PaymentStatus data
  + Login data

• Process

* Multi-leveled access (Administrator & Members)
* Administrative position will have option to manage (add/update/delete/view) next data sets:

Room Inventory

Customer Registration

Room Service

Service

Staff

* Regular staff accounts will only have access to add/update Customer Registration and Room Service requests.
* Both accounts will have the option to see the next reports:

Full room listing - showing availability

Guest checkout bill

Guest listing registered in what room alphabetical

Services available by type of service are listed alphabetically

Staff showing each staff and what they recorded in services

• Security

* + Identity management for all systems   
    UserID/password
  + Restrict access rights to regular staff account

System Modeling:

Event Table:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **EVENT** | **TRIGGER** | **SOURCE** | **USE CASE** | **RESPONSE** | **DESTINATION** |
| **Administration** | | | | | |
| **Log in** | User Log in | Admin/ Regular Staff | Logging into the system | User Log in Confirmation | Admin/ Regular Staff |
| **Staff Account Creation** | New staff | Admin | Create staff account | Confirm creation | Admin |
| **Update Staff Account** | Update Staff Account | Admin | Update staff account | Confirm update | Admin |
| **Delete Staff Account** | Delete Staff Account | Admin | Delete Staff Account | Confirm delete | Admin |
| **View Staff Account** | View staff information | Admin | View staff record | Staff record | Admin |
| **Create New Service** | New Service | Admin | Create a new Service | Confirm created service | Admin |
| **Update Service Account** | Update Service Account | Admin | Update Service | Confirm updated service | Admin |
| **Delete Service Account** | Delete Staff Account | Admin | Delete Service | Confirm deleted service | Admin |
| **View Service** | View Service | Admin | View Service record | Service record | Admin |
| **Create Inventory** | New Inventory | Admin | Create a new Inventory | Confirm created inventory | Admin |
| **Update Inventory** | Update Inventory | Admin | Update Inventory | Confirm updated inventory | Admin |
| **Delete Inventory** | Delete Inventory | Admin | Delete Inventory | Confirm deleted inventory | Admin |
| **View Inventory** | View Inventory | Admin | View Inventory | Inventory record | Admin |
| **Regular Staff and Admin account** | | | | | |
| **Create Guest Registration** | New Guest Registration | Admin/Staff | Create New Registration | Confirm added registration | Staff |
| **Update Guest Registration** | Update Registration | Admin/Staff | Update Registration | Confirm updated registration | Staff |
| **Delete Guest Registration** | Delete Registration | Admin | Delete Registration | Confirm deleted registration | Admin/Staff |
| **View Guest Registration** | View  Registration | Admin/Staff | View  Registration | Registration record | Staff |
| **Create Room Service Request**  **(RSR)** | New RSR | Admin/Staff | Create New RSR | Confirm added RSR | Staff |
| **Update Room Service Request** | Update RSR | Admin/Staff | Update RSR | Confirm updated RSR | Staff |
| **Delete Room Service Request** | Delete RSR | Admin | Delete RSR | Confirm deleted RSR | Admin/Staff |
| **View Room Service Request** | View RSR | Admin/Staff | View RSR | Room Service Record | Staff |
| **Reports** | | | | | |
| **Rooms’ Listing** | View Rooms Status | Admin/Staff | View room availability | Room Detail Report | Admin/Staff |
| **Guest Checkout Bill** | View Check Out Bill | Admin/Staff | View Bill | Bill | Admin/Staff |
| **Guest In Hotel (ab)** | View Guests’ List | Admin/Staff | View Guests’ List | Guests’ Detail Report | Admin/Staff |
| **Services (ab)** | View All Services | Admin/Staff | View List of Services | Services Detail Report | Admin/Staff |
| **Room Service Staff** | View Room Service Staff | Admin/Staff | View RS Staff | RS Staff Detail Report | Admin/Staff |

System Domain Class Diagram:

Diagram

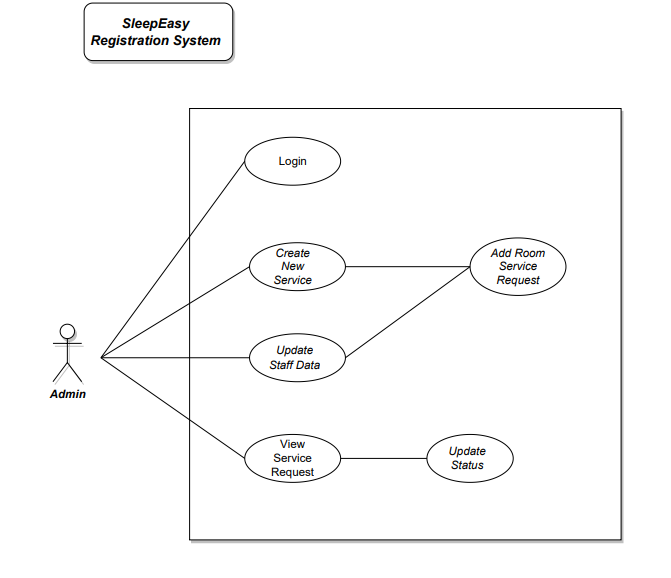
Description automatically generated

Information Management:

**Create new Service**

Use Case Diagram:

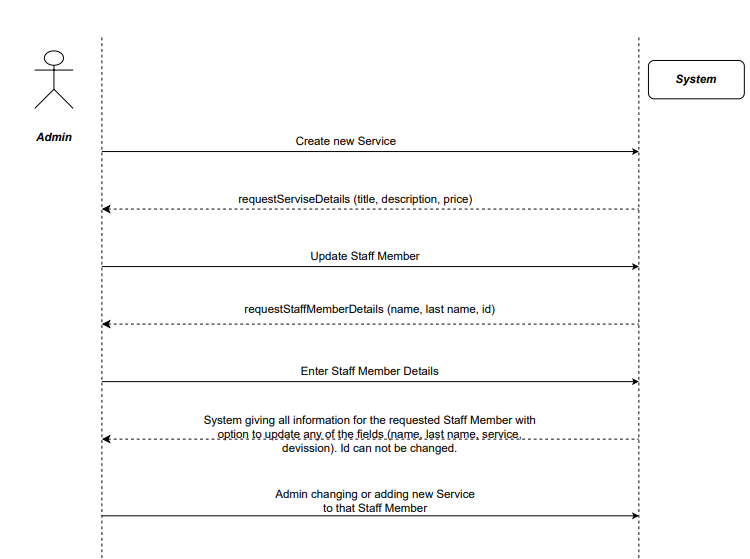
* Users will log in to the system with admin access.
* Admin will create new Service and Update Staff data by assigning new Services to Staff members.
* Then admin will add a new Room Service Request
* After that admin will have an option to View the Request and Update its Status.



###### *Use Case*

|  |  |  |
| --- | --- | --- |
| **Use case name:** | Create new Service | |
| **Scenario:** | Users with admin access will create new services Then update Staff data. After that, he will create a new room service request and have an option to view it and update its status. | |
| **Triggering event:** | New Services in the hotel becomes available | |
| **Brief description:** | Admin is creating new Service | |
| **Actors:** | Admin | |
| **Related use cases:** | Login | |
| Update Staff data | |
| Create Room Service request | |
| View Request and update its status | |
| **Preconditions:** | * Administrator successfully login with admin access. * Service does not exist in the system. | |
| **Postconditions:** | * Create Room service request with new Service * Update Room Service request | |
| **Flow of activities:** | **Actor** | **System** |
| 1. Admin clicks on ‘New Service.’      1. Admin enters needed information in the system.      1. Admin view Staff data   4. Admin Update Staff member | 1.1 System prompts for basic service information to create a new Service.     * 1. System checking if the Service does not exist.   2. System creates the Service.      * 1. System show Staff data.   2. Update chose staff member |
| **Exception conditions:** | Service already exists | |

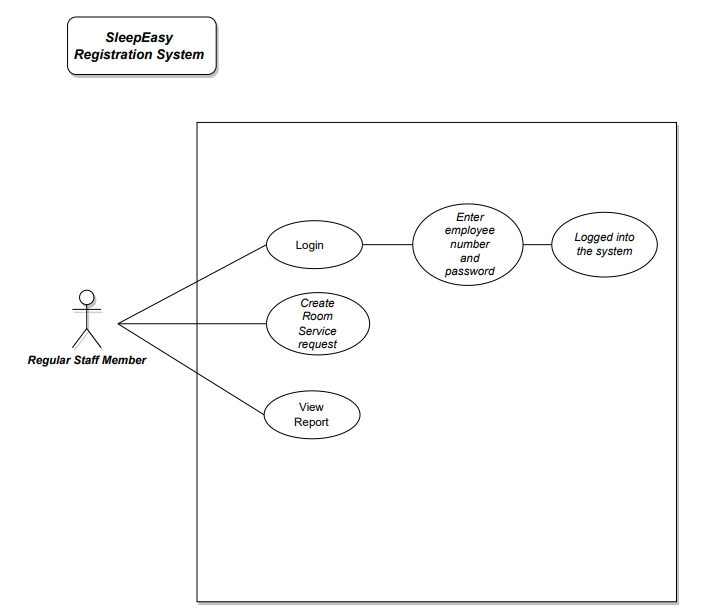
System Sequence Diagram:



**Logging into the system**

Use Case Diagram:

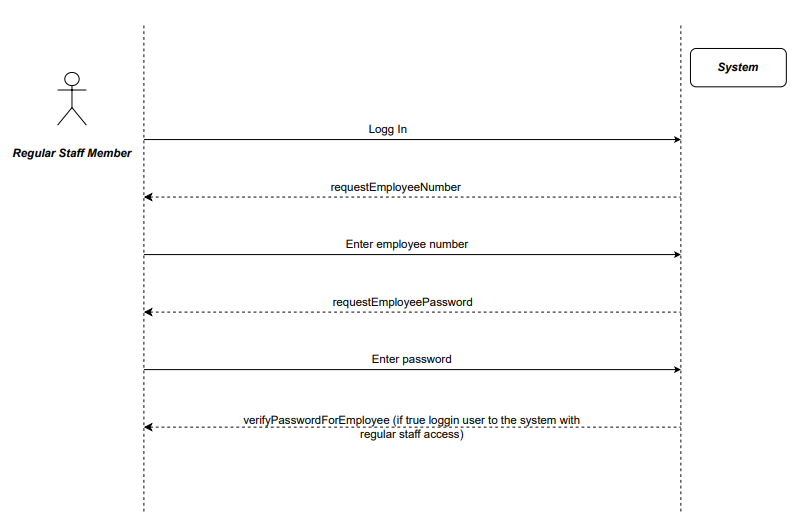
* A regular staff member will be logging into the system by providing employee number and password
* If provided employee number and password exist in data as a pair key–value, then the user will get access to the system as a regular staff member.



###### *Use Case*

|  |  |  |
| --- | --- | --- |
| **Use case name:** | Logging into the system | |
| **Scenario:** | A regular staff member will be trying to login into the system.  To do that he will be asked by the system to provide an employee number and password.  If the provided information is true, then the user will be logged into the system with a regular staff access layer.  If not, show the msg “Try again. Please provide existing employee number and password”. | |
| **Triggering event:** | User Log in | |
| **Brief description:** | Regular staff members trying to login into the system | |
| **Actors:** | Regular staff member | |
| **Related use cases:** | The user was provided with employee number and password | |
| Staff Account Creation | |
| **Preconditions:** | * Staff account successfully created | |
| **Postconditions:** | * Regular staff user logged into the system * View Report | |
| **Flow of activities:** | **Actor** | **System** |
| * User clicks on ‘Login’ * User provide employee number and password as a regular staff member * If true, the user will be logged into the system with regular user access. If not true, will be asked to try again. | * System asking for employee number and password * System verifies the pair employee number and password |
| **Exception conditions:** | * Employees do not exist in the system * A provided pair employee number and password do not match. | |

System Sequence Diagram:



**Add or Remove Staff**

Use Case Diagram:

* Admin will manipulate the tables of staff depending on their status.
* Admin will Add new staff create their Credentials to log into the system.
* Then admin will update a staff record after their termination and destroy their login information.

Diagram

Description automatically generated

###### *Use Case*

|  |  |  |
| --- | --- | --- |
| **Use case name:** | Add or Update Staff | |
| **Scenario:** | Admin must add a new hired staff or update a staff status after termination or change in position that also involves updating to creating the login information of the staff depending on their position. | |
| **Triggering event:** | New Staff and login become available, older staff records gets updated. | |
| **Brief description:** | Admin is creating or updating the staff | |
| **Actors:** | Admin | |
| **Related use cases:** | Login | |
| Update Staff | |
| Create Staff | |
| Create and update Login information | |
| **Preconditions:** | * Administrator successfully login with admin access. * Staff member do not exist for new creation if already exist only update information, Login username cannot be same. | |
| **Postconditions:** | * Create a new Staff if not available and create a new login. * Update an existing staff information with login access. | |
| **Flow of activities:** | **Actor** | **System** |
| 1. Admin clicks on ‘New Staff.’      1. Admin enters needed information in the system.      1. Admin Updates Staff data if existing already   4. Admin Update Login information | 1.1 System prompts for basic Staff information to create a new Staff.     * 1. System checking if the Staff does not exist.   2. System creates the Staff.      * 1. System show Staff information.   2. Update chose staff member to upgrade them to admin position to give access rights. |

System Sequence Diagram:

**Table

Description automatically generated with medium confidence**

**SYSTEM COMPONENTS DETAILS**

Program Design:

Design Class Diagram

Diagram

Description automatically generated

Output Design

###### *Report (Output) – Room List*

Report Analysis

**SYSTEM DOCUMENTATION**

NAME OF SYSTEM DATE

Room List 2022/02/25

ANALYST PURPOSE OF DOCUMENTATION

|  |  |  |
| --- | --- | --- |
| **FIELD** | **FIELD TIPE** | **FIELD LENGTH** |
| Roon Number | int | 5 |
| Description | String | 200 |
| Type | String | 10 |
| Rate | double | 5 |
| Status | String | 20 |

Generates and displays a list of Rooms with status and description.

SORT SEQUENCE

1. Table will be sorted by room number.

COMENTS

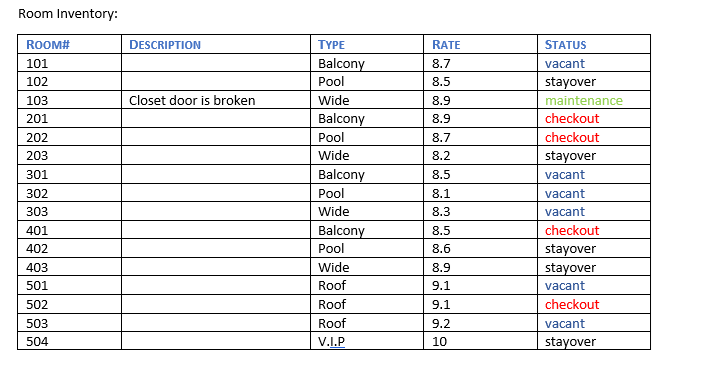
There will be 4 statuses

* Vacant – available room
* Maintenance – not available, something has to be fixed in the room
* Checkout – not available, previous guest have been left and room have to be cleaned
* Stayover – not available, room is booked or occupied

Mock Up

**Room List | SleepEasy Hotel**

**Date: 2022/02/24**



###### *Report (Output) – Guest List*

Report Analysis

**SYSTEM DOCUMENTATION**

NAME OF SYSTEM DATE

Guest List Report 2022/02/24

ANALYST PURPOSE OF DOCUMENTATION

|  |  |  |
| --- | --- | --- |
| **FIELD** | **FIELD TIPE** | **FIELD LENGTH** |
| Last Name | String | 25 |
| First Name | String | 25 |
| Room Number | int | 5 |
| Check-in Date | DateType | 10 |
| Checkout Date | DateType | 10 |

Generates and displays a list of Guests that are currently in the hotel.

SORT SEQUENCE

1. Table will be sorted by last names in alphabetical order.

Mock Up

**Guest List | SleepEasy Hotel**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Last Name** | **First Name** | **Room Number** | **Checkin** | **Checkout** |
| Berman | Margareth | 203 | 2022/02/20 | 2022/02/25 |
| McDonalds | Winston | 401 | 2022/02/22 | 2022/02/26 |
| Paterson | Bella | 115 | 2022/02/25 | 2022/02/26 |
| … |  |  |  |  |
| … |  |  |  |  |
| … |  |  |  |  |
| … |  |  |  |  |
| Date: 2022/02/24 | | | | |

*Report (Output) – Staff*

Report Analysis

**SYSTEM DOCUMENTATION**

NAME OF SYSTEM DATE

Staff 2022/02/25

ANALYST PURPOSE OF DOCUMENTATION

|  |  |  |
| --- | --- | --- |
| **FIELD** | **FIELD TIPE** | **FIELD LENGTH** |
| Name | String | 20 |
| Last Name | String | 20 |
| Employee Number | int | 10 |
| Title | String | 25 |

Generates and displays a list of Staff members with a title which represent the service that they assign to.

SORT SEQUENCE

The order will be alphabetical by the last name.

Mock Up

**Staff | SleepEasy Hotel**

|  |  |  |  |
| --- | --- | --- | --- |
| **Last Name** | **First Name** | **Employee Number** | **Title** |
| Armando | Nick | 111556 | Dry cleaning service |
| Markos | Maria | 125461 | Restaurant service |
| Vidal | Glen | 124556 | Room cleaning service |
| … |  |  |  |
| … |  |  |  |
| … |  |  |  |
| … |  |  |  |

Input Design

**Guest Registration**

This functionality will be open for both administration and regular staff.

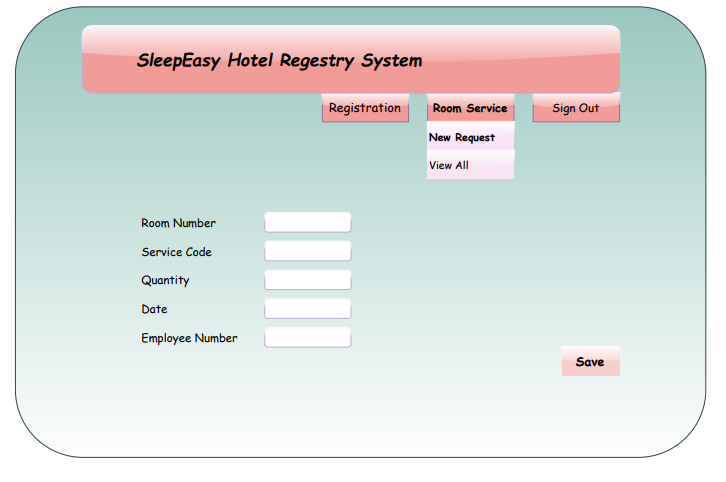
Address and Payment Method are buttons that are taking the user to another window to fill in all required information to register a new guest.



**Add new Room Service Request**

This functionality will be open for both administration and regular staff.

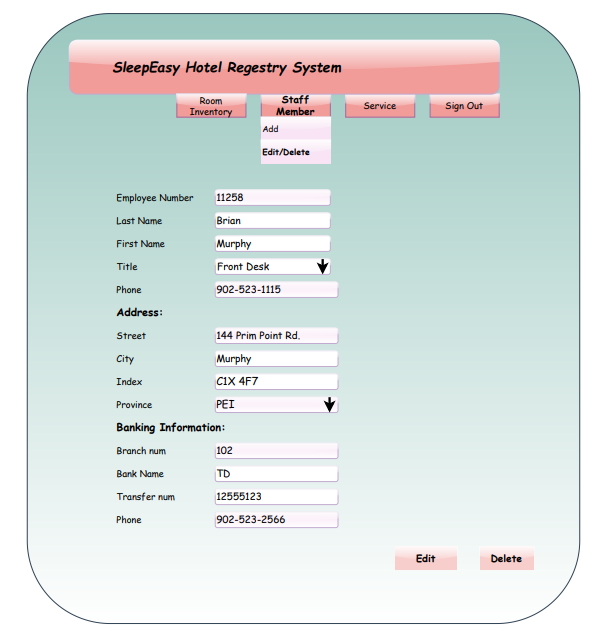
Employee Number field can stay empty. The purpose is to fill that field later on by updating Room Service request when particular employee will get the request.



**Edit staff member**

This functionality will be available only for administration.

Title and Province fields will be dropdown lists.



Database Design

ERD

Diagram, table

Description automatically generated

Database Design Language (DBDL)

Booking (BookingID, GuestID, CheckinDate, CheckoutDate, RoomID, StaffID, PaymentID, BookingStatusID)

FK: GuestID – Guest

FK: StaffID – Staff

FK: PaymentID – Payment

FK: BookingStatusID – Booking Status

FK: RoomID – Rooms

BookingStatus (BookingStatusID, BookingStatus)

Guest (GuestID, GuestFirstName, GuestLastName, GuestStreet, GuestState, GuestZipCode, GuestPhone, GuestEmailAddress)

Rooms (RoomID, BedType, Floor, RoomStatus, OccupiedDate)

FK: BedType – Beds

Beds (BedType, BedCode, WeekendRate)

Staff (sStaffID, StaffFirstName, StaffLastName, PositionID, StaffStreet, StaffCity, StaffZipCode, StaffPhone, StaffEmailAddress)

FK: sPositionID - Position

Position (PositionID, Position)

Login (LoginID, PositionId, Username, Password)

FK: PositionID - Position

Payment (PaymentID, RoomID, PaymentDate, Payment, PaymentType, PaymentStatusID)

FK: pRoomID – Room

FK: PaymentStatusID – PaymentStatus

PaymentStatus (PaymentStatusID, Status)

Data Dictionary

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Table | Column | Type | Reference | Description |
| Booking | BookingID GuestID CheckinDate CheckoutDate RoomID  StaffID BookingStatusID  PaymentID | Integer (10)  Integer (10)  Date  Date  Integer (10)  Integer (10)  Varchar (100)  Integer (10) | Guest  Staff  Room  Payment  BookingStatus | Booking ID  Guest ID  Check-in Date  Check-out Date  Room ID  Staff ID  Booking Status  Payment ID |
| Guest | GuestID GuestFirstName GuestLastName  GuestStreet  GuestState  GuestZipCode  gPhone  gEmailAddress | Integer (10)  Varchar (50)  Varchar (100)  Varchar (100)  Varchar (50)  Varchar (50)  Number (10)  Varchar (50) |  | Guest ID  Guest First Name  Guest Last Name  Guest Street  Guest State  Guest Zip Code  Guest Phone  Guest Email Address |
| BookingStatus | BookingStatusID  BookingStatus | Integer (10)  Varchar (100) |  | Booking Status ID  Booking Status |
| Room | RoomID  Floor  BedType  OccupiedDate  RoomStatus | Integer (10)  Number (2)  Varchar (100)  Date  Varchar(50) | Beds | Room ID  Room Floor  Bed Type  Occupied Date  Room Status |
| Beds | BedType  BedCode  WeekendRate | Varchar (50)  Integer (10)  Integer (4) | Rooms | Bed Type  Bed Code  Weekend Rate |
| Staff | StaffID  StaffFirstName  StaffLastName  PositionID  StaffStreet  StaffCity  sZipCode  sPhone sEmailAddress | Integer (10)  Varchar (50)  Varchar (100)  Integer (10)  Varchar (100)  Varchar (50)  Varchar (6)  Number (10)  Varchar (50) | Position | Staff ID  Staff First Name  Staff Last Name  Position ID  Staff Street  Staff City  Staff Zip Code  Staff Phone  Staff Email Address |
| Position | PositionID  Position | Integer (10)  Varchar (50) |  | Position ID  Position |
| Login | LoginID  PositionID  Username  Password | Integer (10)  Integer (10)  Varchar (50)  Varchar (50) | Position | Login ID  Position ID  Username  Password |
| Payment | PaymentID RoomID  PaymentDate  Payment  PaymentType  PaymentStatusID | Integer (10)  Integer (10)  Date  Currency  Varchar (50)  Integer (10) | Room  Payment Status | Payment ID  Room ID  Date of Payment  Payment Amount  Payment Type  Payment Status ID |
| PaymentStatus | PaymentStatusID PaymentStatus | Integer (10)  Varchar (100) |  | Payment Status ID  Payment Status |

**SUPPORT PROCESSING DESIGN**

Policy and Procedures

* To use SleepEasy Hotel Registry System administration staff and regular staff will need different levels of access. For that each one will be provided by private username and password. These username and password are not allowed to be shared or passed to someone.
* Any new guest registration or room service request cannot be deleted by regular staff. They must contact the Administrator and they will take care of that.
* Database backups to be performed regularly and cyclical and stored in the applicable data drive that is backed up offsite.
* Both admin and regular staff will have access to all reports. This condition is for acceptable workflow and shared responsibility.
* Only admin will have an access to create/update and delete from staff , service or inventory(room) data.

**ENVIRONMENTAL REQUIREMENTS**

System Environment Requirement

**Software**

* Windows 7 or later
* .Net Framework 4.5 or later
* Server PHP 7.0 and Apache for hosting the web application.

**Hardware**

* Server to meet the requirement of PHP and Maria DB
* Workstation computers with windows 7 or later with internet connection.

**Staff**

* No additional staff is required.

**IMPLEMENTATION REQUIREMENTS**

To implement the SleepEast Hotel Registry system, several items must occur:

Data Entry / Conversion

* The current spreadsheet tables will be parsed and migrated into the applicable database tables.
* Staff will have to verify each entry after the migration of data.

Security

* Security will be implemented in the SleepEast Hotel Registry system via a login; no information is available until a user is authenticated.
* The web application will not be publicly accessible, including the login screen.
* The deployment will be on-premises behind a firewall or in the cloud via the VPN.
  + Each staff will have their own username and password to access the system.
  + Staff will only have access applicable to their position (view only, edit rights, reports).
* All printed reports will have limited personally identifiable information (e.g., Booking details to be partially redacted on reports).
* Printers to be in secure locations and accessed by authorized individuals only.
* Accounts to be reviewed by Management.
* Training
  + All staff will need basic training on the SleepEast Hotel Registry system to log in, add and view room bookings, collect payments, make notes, and set up notifications.
  + Manager's training will include creating bookings, editing data, and maintaining the lookup tables.

Post-Implementation Review

* This review will be conducted at the end of the month the system is implemented, allowing the capture of initial feedback while fresh in everyone’s mind and allowing modifications or bug fixes to be addressed promptly.

**APPENDIX**

Appendix A: Team Contract/Charter

|  |  |  |
| --- | --- | --- |
| Name | Email | Phone Number |
| Azam Sheik | azam.sheikh721@gmail.com | 782-377-2550 |
| Badalkumar Patel | badalkumar9198@gmail.com | 902-314-0113 |
| Karina Akramov | kakramov@hollandcollege.com | 647-262-8211 |
| Marianna H C Pedroso | mhollandacampos@hollandcollege.com | 902-916-2480 |

Communication

* Microsoft Teams
* Group text messaging
* Video conferencing calls
* Audio conferencing calls
* Screen sharing
* File sharing

Goals

* To gain valuable experience that can be applied to future work
* Meet deliverable deadlines
* Stay within the project budget
* Establish milestones
* Learn from mistakes
* Successfully create and deliver analysis and design of the requested system that is:
* Functional
* Usable
* Efficient
* Simple
* What marks does the team as a whole want to achieve?
* The team would like to achieve a grade of 90% or higher.

Roles & Responsibilities

1. Marianna H C Pedroso – Project Manager
2. Azam Sheikh – System Analyst
3. Karina Akramov – System Analyst
4. Badalkumar Patel – Developer

Team Meetings

* Group text messaging will occur daily to discuss and set plans for the next 24 hours.
* Audio or video conference call meetings will occur twice each week to ensure team members are on the same page and to check that progress is being made to meet deliverable deadlines. Meetings will be held using Microsoft Teams collaboration tools.
* Meeting agendas will be developed by first identifying any goals for upcoming deliverables. Team members will also provide input prior to the meeting to ensure all needs are fulfilled. Time spent on each topic will be estimated in advance and a plan will be developed accordingly.
* The project manager will start and facilitate meetings, but each topic covered during a meeting will be led by a designated team member who is focusing on the subject.
* Microsoft Teams will be used to schedule meetings, and the built-in Microsoft Forms tool can be used to track decisions and activities through surveys, quizzes, and polls.
* Missing or being late for a meeting can lead to productivity loss, declined team morale, client dissatisfaction, and project management problems. To help avoid these issues meetings will be planned well in advance and all team members will be notified. Reminders and notifications will be set in Microsoft Teams.
* Some valid reasons for missing or being late to a meeting include poor planning or organization of a meeting. Scheduling conflicts are also a possibility. Since most communication and meetings will occur remotely through Microsoft Teams, technology failure is a risk.
* Repeated absences/lateness will be addressed by the project manager privately. An action plan will be developed to help the situation. This may include introducing time-management lessons or utilizing a more flexible schedule.

Decision-Making

Decisions on the development of this project will be made in consensus with the group conscience and will proceed as the group deems to be acceptable. However, decisions will be made in the following steps:

1. The problem or issue will first be identified.
2. Further information and details will then be gathered regarding this problem or issue, if necessary.
3. Solutions will be evaluated amongst team members and discussed.
4. Once the solutions have been discussed, the team will decide together which solution to implement to correct the problem or issue.
5. If there is not 100% agreement on a solution, the team will then discuss further until a decision can be made in which all members agree it is acceptable.

Note: Decisions will be made in 24 hours or less.

Decisions requiring assistance from the instructor will be those that may require further clarification or details to make a sound and efficient decision for the project.

Team Project Contribution & Member Accountability

1. How will the team proceed if someone’s work is deemed unacceptable?
   1. The issue will be addressed early and privately. This will help promote a more informal chat instead of a more serious discussion. It will also help prevent future occurrences of a similar issue.
   2. Thoughtful questions will be asked to determine if there is a solvable problem that is causing the declined quality of work.
   3. Relevant examples of more acceptable work will be provided.
2. If someone does not submit their work on time, what will be the course of action to ensure the project is completed and/or the overall team mark is not compromised?
   1. Questions will be asked to determine what happened, how the team member intends to recover, and how the team member intends to prevent it from happening again.
   2. Other team members will help complete the late task as quickly as possible to keep the project moving along and avoid compromising the team mark.
3. How will you reward individual members for outstanding contributions?
   1. Individual praise and showcase their work.
   2. Overachieving members may receive time off or a more flexible schedule without compromising future deadlines.
4. Under what circumstances will you remove a team member from your group?
   1. An excessive number of absences or missed deadlines.
   2. A team member leaves the CIS program. Access to project files and group collaboration tools will be revoked.
5. Grading assignment:
   1. The team would prefer to be graded as a group and receive the same marks.

Conflict

Steps towards handling conflicts within the team are as follows:

1. Acknowledge the conflict.
2. Discuss the conflict and any possible resolutions that may be made.
3. If necessary, request feedback from another team member to assist in the resolution.

Revisiting & Revising the Team Charter

The Team Charter will be re-evaluated using the decision-making protocols described in this Appendix. A review will occur at the mid-point of the project schedule, coinciding with Status Report Three on February 25, 2022.

All project members and the client will receive signed copies of the original and revised Team Charter.

Appendix B: Change Request/Notification

**SHRS | Change Request/Notification**

DATE: REQUEST #:

TO:

FROM:

SUBJECT:

|  |  |  |  |
| --- | --- | --- | --- |
| Key Dates |  | | |
| Anticipated Decision: |  | Expected Implementation: |  |

|  |
| --- |
| Brief Description |
|  |

|  |  |
| --- | --- |
| Impact (Time) | Days: |
|  |  |
|  |  |
| Impact (Budget) | Amount: |
|  | |

|  |
| --- |
| Recommendation/Resolution |
|  |

|  |  |  |
| --- | --- | --- |
| Approved ☐ | Disapproved ☐ | Reason: |

|  |  |  |  |
| --- | --- | --- | --- |
| Approver Title | Approver Name | Signature | Date |
| Client | Joey Kitson |  |  |
| Project Manager | Marianna H C Pedroso |  |  |

Appendix C: Progress Status Report

**SHRS| Progress Status Report**

DATE: REPORT #:

TO:

FROM:

SUBJECT:

|  |
| --- |
| Overview of Progress |
|  |

|  |  |  |  |
| --- | --- | --- | --- |
| Outstanding Issues/ Recommended Action | | Count: | 0 |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Change Requests Log | | Count: | 0 |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |

|  |
| --- |
| Plans for Next Period |
|  |

|  |
| --- |
| Overview of Budget (attached MS Project file) |
|  |