

*Subject:* **INTRODUCTION TO SOFTWARE ENGINEERING**

***SRS***

**HOTEL MANAGEMENT SOFTWARE**

***Instructor: Mas.Nguyen Thi Thanh Truc***

*Team* **27**

*Name of all member:*

Truong Van Khai – 21520274

Le Ngo Minh Đuc – 21520195

Le Yen Nhi - 21522427

Ngo Phuc Danh – 21521924

*Ho Chi Minh, 1st July 2023*

****

HO CHI MINH CITY NATIONAL UNIVERSITY

**UNIVERSITY OF INFORMATION TECHNOLOGY**

**SOFTWARE ENGINEERING DEPARTMENT**

## **CONTENTS**

[1. Introduction 3](#_Toc133007959)

[1.1. Purpose 3](#_Toc133007960)

[1.2. Document Conventions 3](#_Toc133007961)

[1.3. Project Scop 3](#_Toc133007962)

[1.4. References 3](#_Toc133007963)

[2. Overall Description 4](#_Toc133007964)

[2.1. Product Perspective 4](#_Toc133007965)

[2.2. User Classes and Characteristics 4](#_Toc133007966)

[2.3. Operating Environment 5](#_Toc133007967)

[2.4. Design and Implementation Constraints 5](#_Toc133007968)

[2.5. Assumptions and Dependencies (\*) 5](#_Toc133007969)

[3. System Features 6](#_Toc133007970)

[3.1. Create User Account 6](#_Toc133007971)

[3.1.1. Description 6](#_Toc133007972)

[3.1.2. Functional Requirements 6](#_Toc133007973)

[3.2. Sign-in 7](#_Toc133007974)

[3.2.1. Description 7](#_Toc133007975)

[3.2.2. Functional Requirements 7](#_Toc133007976)

[3.3. Create Team Profile 8](#_Toc133007977)

[3.3.1. Description 8](#_Toc133007978)

[3.3.2. Functional Requirements 8](#_Toc133007979)

[3.4. Scheduling Match 9](#_Toc133007980)

[3.4.1. Description 9](#_Toc133007981)

[3.4.2. Functional Requirements 9](#_Toc133007982)

[3.5. Record Match Result 10](#_Toc133007983)

[3.5.1. Description 10](#_Toc133007984)

[3.5.2. Functional Requirements 10](#_Toc133007985)

[3.6. Player Lookup 11](#_Toc133007986)

[3.6.1. Description 11](#_Toc133007987)

[3.6.2. Functional Requirements 11](#_Toc133007988)

[3.7. Create Tournament Report 12](#_Toc133007989)

[3.7.1. Description 12](#_Toc133007990)

[3.7.2. Functional Requirements 12](#_Toc133007991)

[3.8. Change or Add Regulations 13](#_Toc133007992)

[3.8.1. Description 13](#_Toc133007993)

[3.8.2. Functional Requirements 13](#_Toc133007994)

[4. Data Requirements 25](#_Toc133007995)

[4.1. Logical Data Model 25](#_Toc133007996)

[4.2. Data Dictionary 26](#_Toc133007997)

[4.3. Data Integrity, Retention, and Disposal 28](#_Toc133007998)

[5. External Interface Requirements 29](#_Toc133007999)

[5.1. User Interfaces 29](#_Toc133008000)

[5.2. Software Interfaces 29](#_Toc133008001)

[5.3. Hardware Interfaces 31](#_Toc133008002)

[5.4. Communications Interfaces 31](#_Toc133008003)

[6. Quality Attributes 32](#_Toc133008004)

[6.1. Usability Requirements 32](#_Toc133008005)

[6.2. Performance Requirements 32](#_Toc133008006)

[6.3. Reliability Requirements 32](#_Toc133008007)

[6.4. Security Requirements 32](#_Toc133008008)

[6.5. Availabilty Requirements 32](#_Toc133008009)

[6.6. Inverse Requirements 32](#_Toc133008010)

# **1. Introduction**

## 1.1. Purpose

This SRS describes the software functional and non-functional requirements for release 1.0 of the Hotel Managment. This document is intended to be used by the members of the project team who will implement and verify the correct functioning of the system. Unless otherwise stated, all requirements specified here are high priority and committed for release 1.0.

## 1.2. Document Conventions

No special typographical conventions are used in this SRS.

## 1.3. Project Scope

This system allows the guests to book rooms and contact the receptionist online, allows the hotel managers to keep track of their employee’s progress and the changes in their hotel. The system also allows for easy access to information and contact between all its users, hotel staff can use the system to organize and schedule their tasks and reach them easily, the system allows for handling of related financial matters.

## 1.4. References

* SRS template by Jacksonville State University
* SMS-SRS by team from previous course

# **2. Overall Description**

The product described in this document is a software for hotel management.

## 2.1. Product Perspective

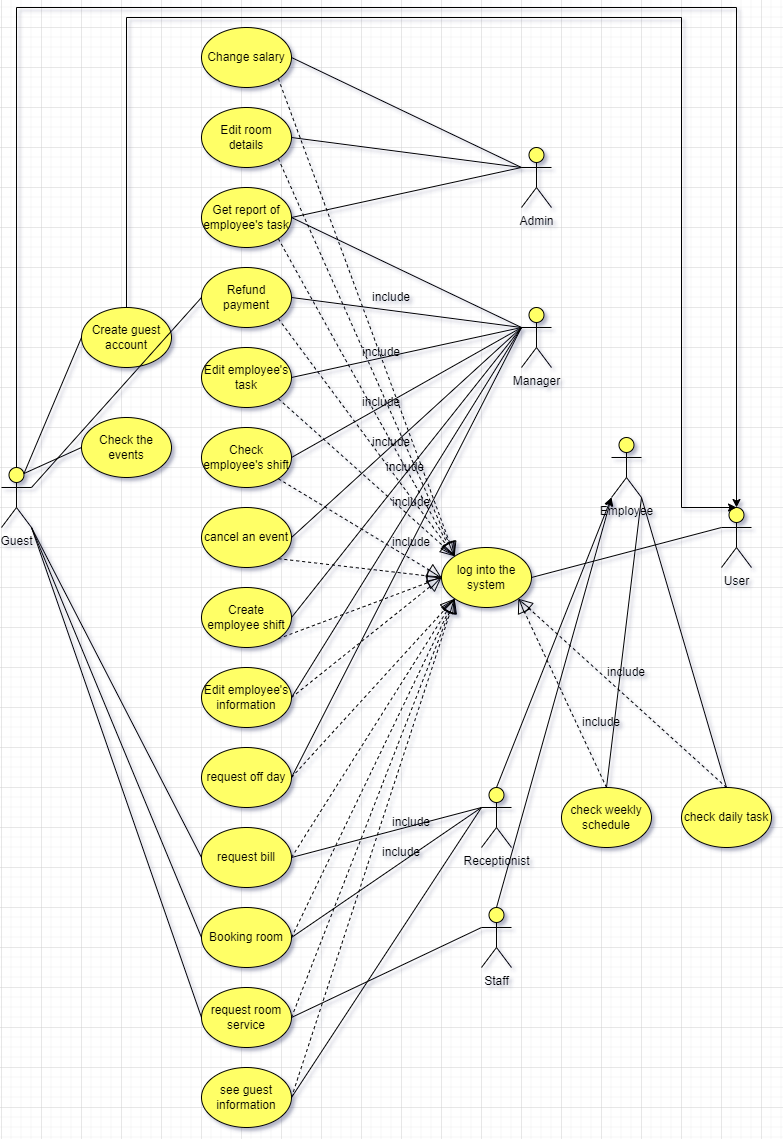


Figure 1.Use-Case Diagram

## 2.2. User Classes and Characteristics

|  |  |
| --- | --- |
| User | any person who uses the system. |
| Guest | any person who uses the system. |
| Employee | any person working at the hotel in any capacity. |
| Admin | administrator of the hotel also known as General manager. |
| Manager | manages staff and receptionists and deals with events hosted by the hotel, deals with guests in cases where he is needed. |
| Receptionist | deals with guest related matters and helps guests when necessary. |
| Staff | Can be cleaners, bellboys, waiters, cooks, and chefs. |

## 2.3. Operating Environment

|  |  |
| --- | --- |
| OE-1: | The Hotel Managment shall operate correctly with any operating system from Windows, MacOS, Linux that supported Python |
| OE-2: | The Hotel managment shall permit user access from the cooperate Intranet. |

## 2.4. Design and Implementation Constraints

|  |  |
| --- | --- |
| CO-1: | The system’s design, code, and maintenance documentation shall conform to the Hotel Managment SDD. |
| CO-2: | The system shall use the SQLite Library database management system, Python 3.10 |

## 2.5. Assumptions and Dependencies (\*)

### 3. System Features

## 3.1. Booking Room

### 3.1.1. Description

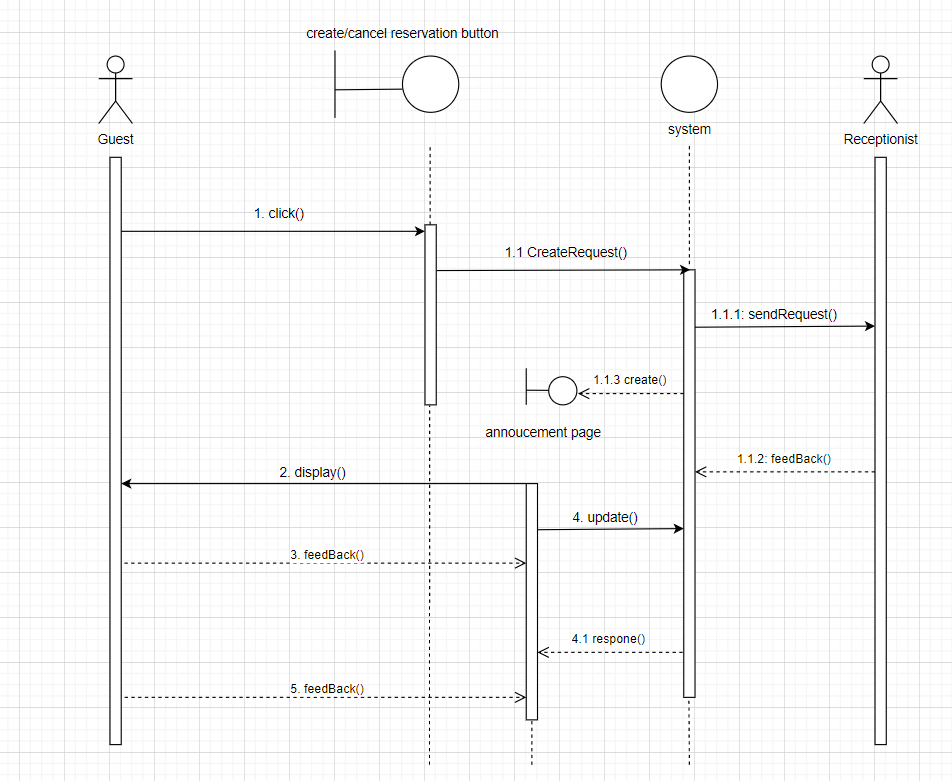


Figure 2. Booking Room Sequence Diagram

Receptionist, Guest can book a room.

### 3.1.2. Functional Requirements

*Request for booking room*

*Introduction*

• User requires the Hotel system to book a room.

*Inputs*

•User click to book a room.

*Processing*

•The Hotel system give a feedback and display room list for user and waiting for a user feedback.

## 3.2. See guest information

### 3.2.1. Description

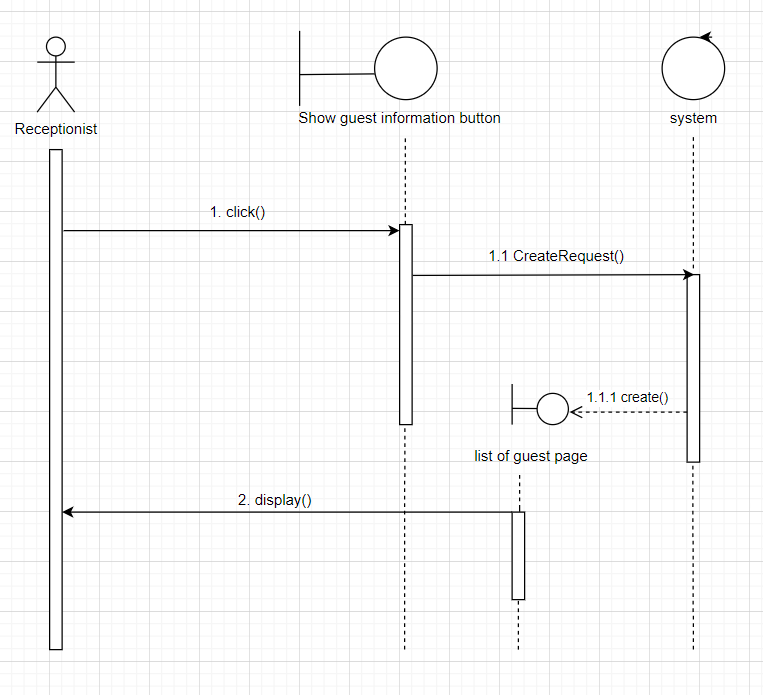


Figure 3. See guest Info Sequence Diagram

A receptionist can see the Guest information.

### 3.2.2. Functional Requirements

*Request for Seeing Guest Information*

*Introduction*

•A receptionist requires to see the Guest information.

*Inputs*

• Receptionist’s click.

*Processing*

•The Hotel system create a list of guests page and display to receptionist.

## 3.3. Check Daily Tasks

### 3.3.1. Description

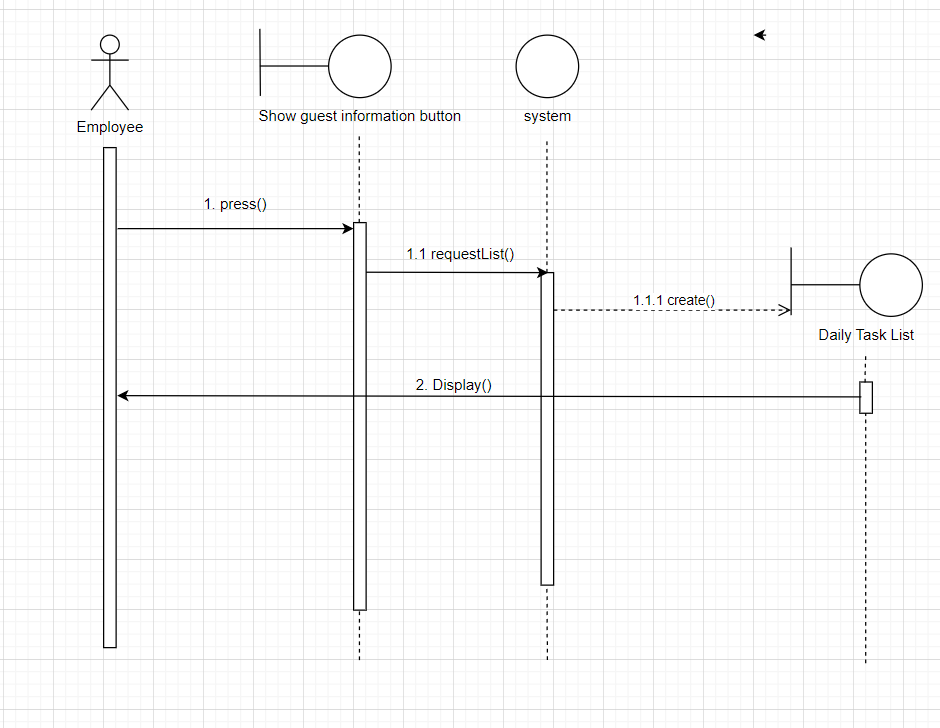


Figure 4. Check Daily Tasks Sequence Diagram

Employee can check the Daily tasks.

### 3.3.2. Functional Requirements

*Request for checking daily tasks*

*Introduction*

•The employee click the “Daily tasks button” to check Daily tasks.

*Inputs*

•Employee press the check daily task button

*Processing*

• The Hotel system displays the Daily tasks.

## 3.4. Check the events

### 3.4.1. Description

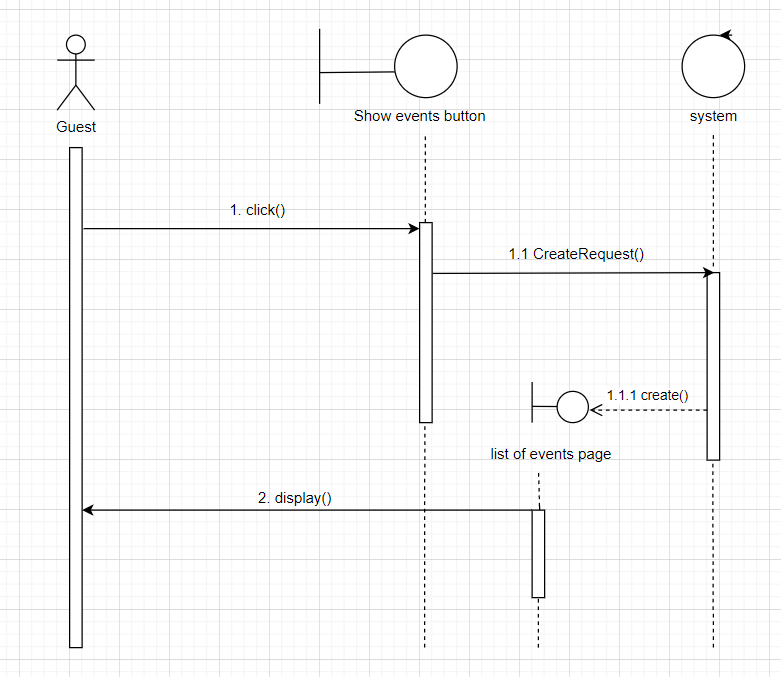


Figure 5. Check Daily Tasks Sequence Diagram

Any User and Guest can check the Events.

### 3.4.2. Functional Requirements

*Request for Check the Events*

*Introduction*

•Guest click to check the Events

*Inputs*

•Guest’s Click

*Processing*

•The Hotel system create a list of events page and display to Guest.

## 3.5. Change Salary

### 3.5.1. Description

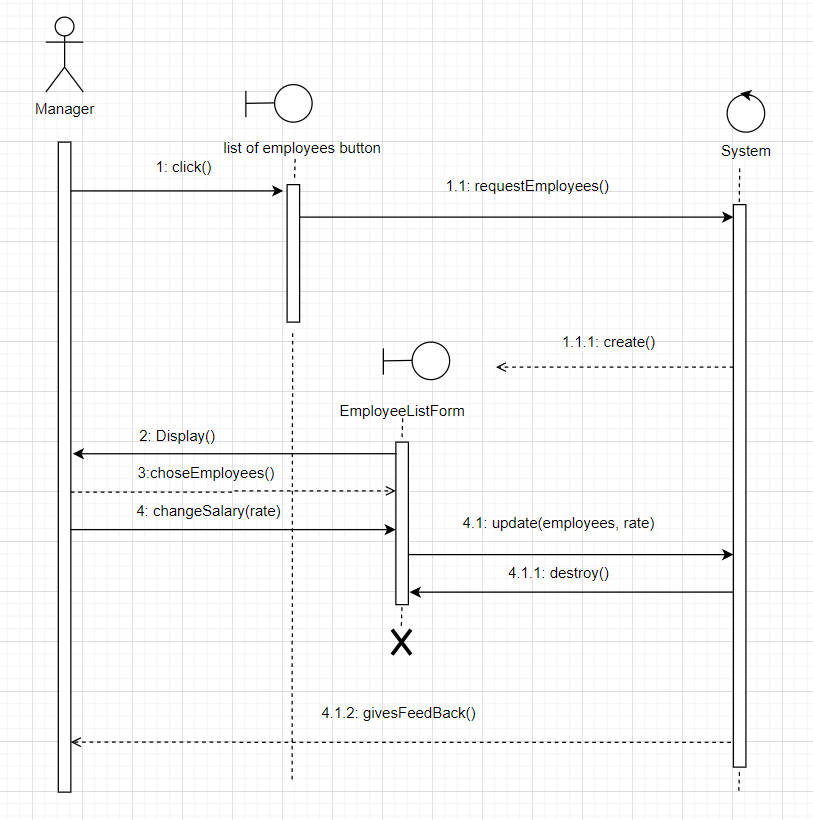


Figure 6. Change Salary Result Sequence Diagram

The manager can change the staff salary.

### 3.5.2. Functional Requirements

*Request for changing salary*

*Introduction*

• Manager click the button to change the emplyee salary.

*Inputs*

•Manager’s click.

*Processing*

•The Hotel system displays list of employees salary and wait for changing.

## 3.6. Edit room details

### 3.6.1. Description

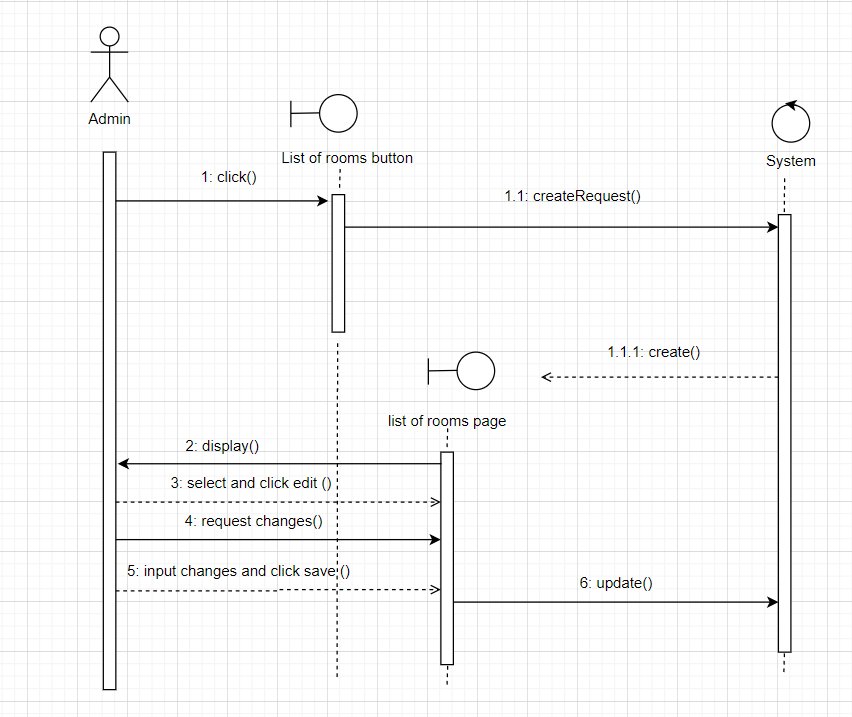


Figure 7. Player Lookup Sequence Diagram

Admin can change the room details.

### 3.6.2. Functional Requirements

*Request for changing rooms details*

*Introduction*

•The admin click to change the room details.

*Inputs*

•Admin’s click.

*Processing*

•The Hotel system displays and wait for request change.

## 3.7. Cancel an event

### 3.7.1. Description

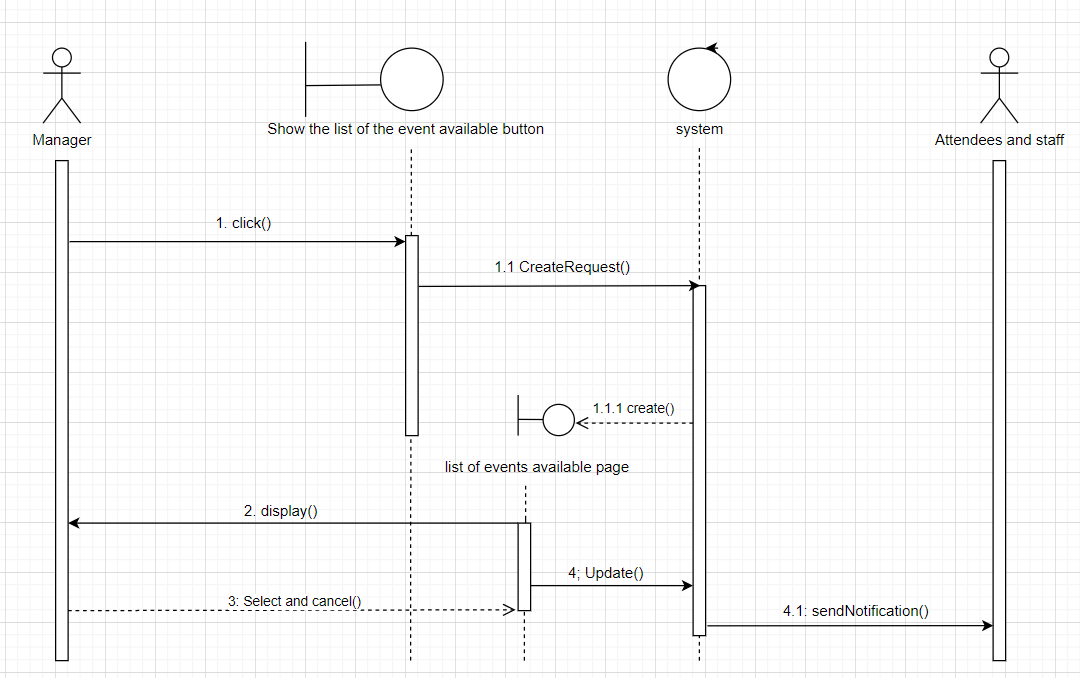


Figure 8. Cancel an Event Sequence Diagram

Manager can change the room details.

### 3.7.2. Functional Requirements

*Request for cancel an event*

*Introduction*

•The Manager clicks on the buuton to cancel an event.

*Processing*

•The Hotel system create an list of event and wait for an cancelling request.

## 3.8. Edit an employee’s tasks.

### 3.8.1. Description

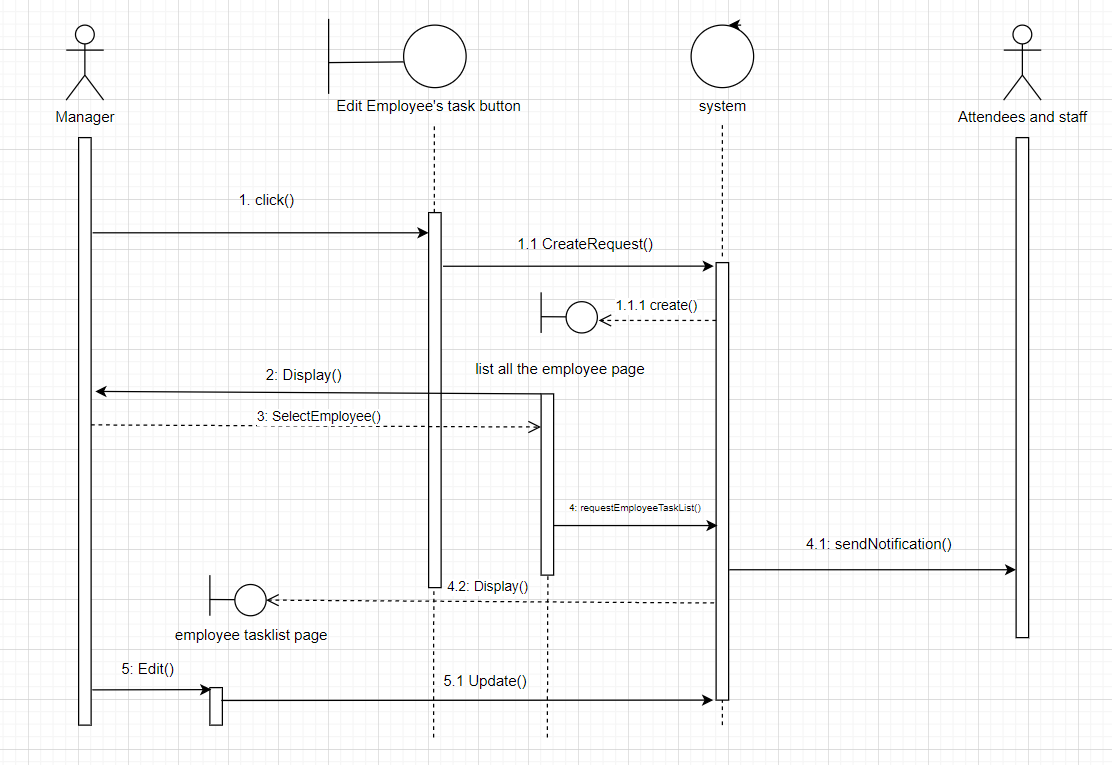


Figure 9. Edit an Emplyee’s task Sequence Diagram

Manager can edit an employee's task.

### 3.8.2. Functional Requirements

*Request for Editing employee task.*

*Introduction*

•Manager can edit an employee's task

*Inputs*

•Manager click

*Processing*

•The hotel system display employee task list and accpecting for and editing.

## 3.9.Log into the system.

### 3.9.1. Description

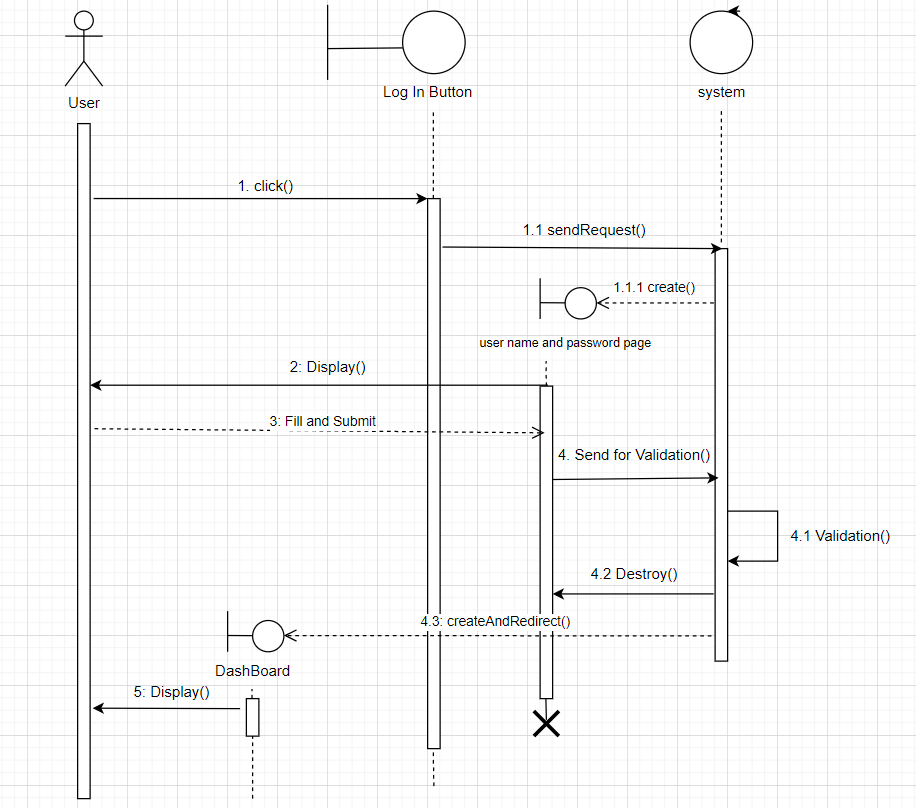


Figure 9. Log into the system Sequence Diagram

Any User can log into the system.

### 3.9.2. Functional Requirements

*Request for Logging into the system.*

*Introduction*

•User click the log in button

*Inputs*

•User ,click

*Processing*

•The hotel system display employee task list and accpecting for and editing.

## 3.10. Get report of employee’s task.

### 3.10.1. Description

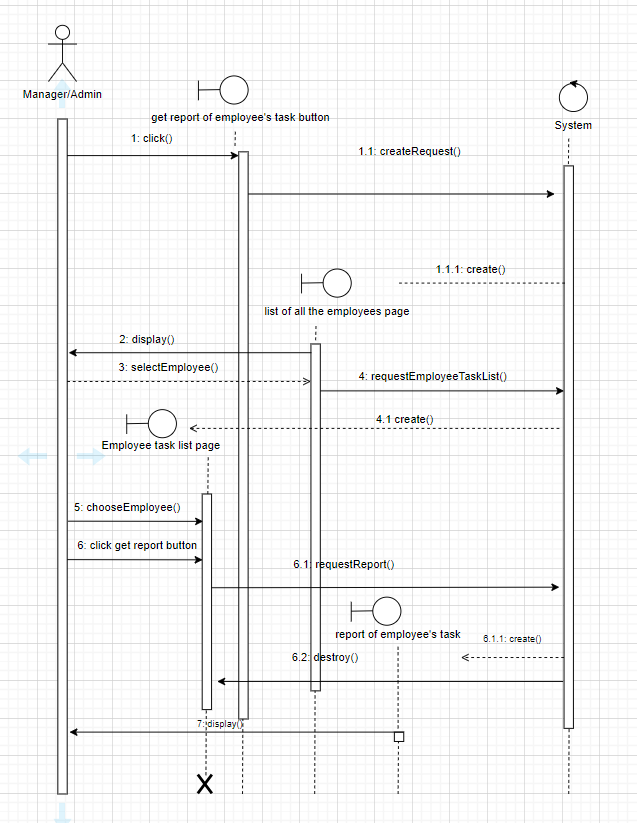


Figure 10. Get report of an Emplyee’s task Sequence Diagram

Manager/admin can get an employee's task.

### 3.10.2. Functional Requirements

*Request for Getting report of an employee task.*

*Introduction*

•Manager/Admin click the get report of an employee task button.

*Inputs*

•Manager click

*Processing*

•The Hotel system create a list of report of employee task.

## 3.11. Request day off.

### 3.11.1. Description

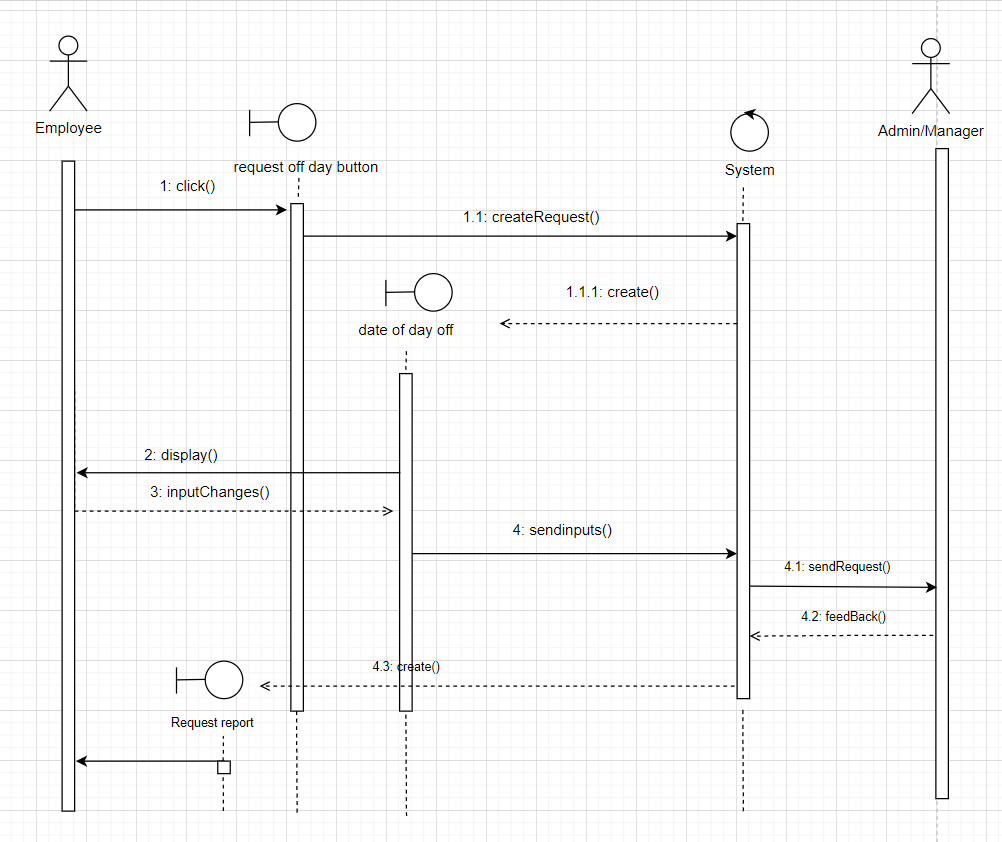


Figure 11. Request day off Sequence Diagram

Employee can request day off.

### 3.11.2. Functional Requirements

*Request for Requesting day off.*

*Introduction*

•Employee click the request off dat button

*Inputs*

•Emplyee click

*Processing*

•The hotel system display data of day off and wait for input.

## 3.12. Create employee shift.

### 3.12.1. Description

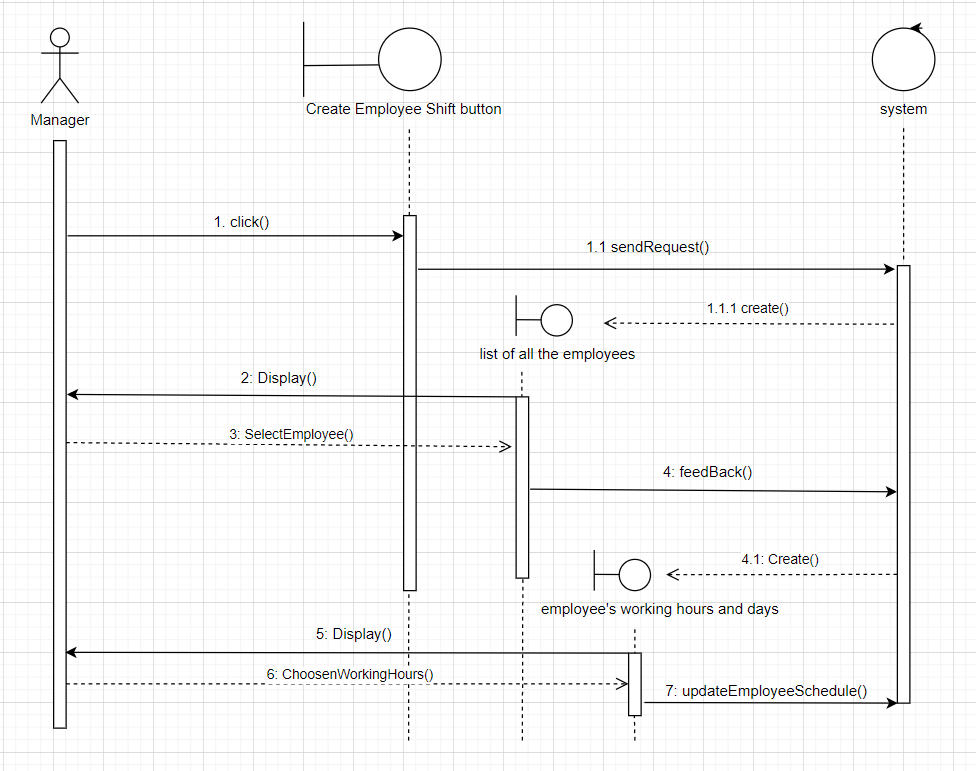


Figure 12. Create emplyee shift Sequence Diagram

Manager can create emplyee shift.

### 3.12.2. Functional Requirements

*Request for Creating employee shift.*

*Introduction*

•Manager click the create emplyee shift button.

*Inputs*

•Manager click

*Processing*

•The hotel system display list of employee and wait for input.

## 3.13. Check employee’s shift.

### 3.13.1. Description

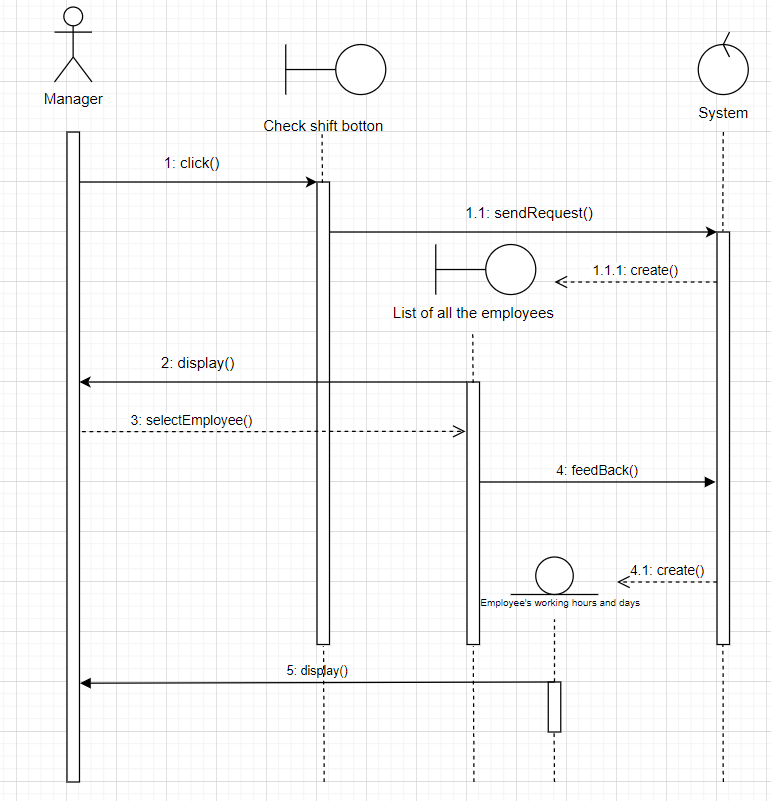


Figure 13. Check emplyee’s shift Sequence Diagram

Manager can check an employee's shift.

### 3.13.2. Functional Requirements

*Request for Checking employee shift.*

*Introduction*

•Manager can check employee's shift

*Inputs*

•Manager click

*Processing*

• The hotel system display list of employee and wait for input.

## 3.14. Check weekly schedule.

### 3.14.1. Description

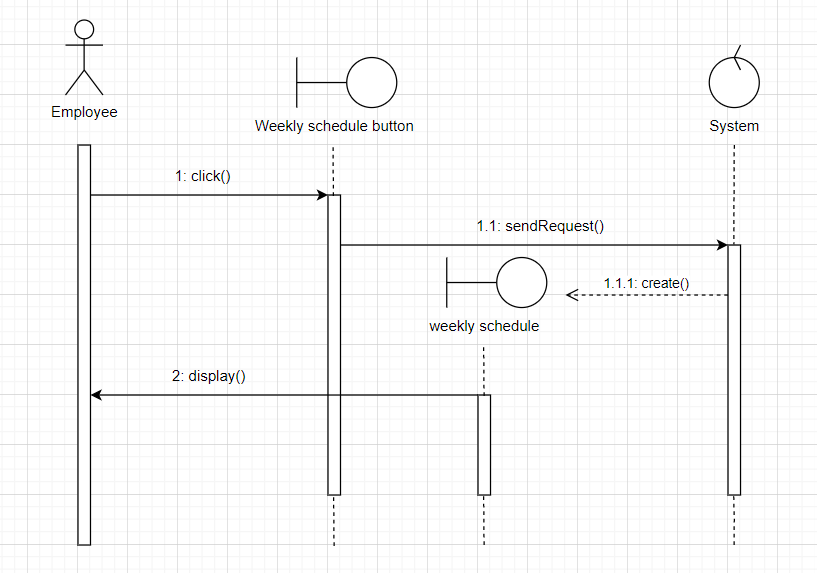


Figure 14: Check weekly schedule Sequence Diagram

Employee can check weekly schedule.

### 3.14.2. Functional Requirements

*Request for Checking weekly schedule.*

*Introduction*

•Employee click the weekly schedule button

*Inputs*

•Employee click

*Processing*

•The hotel system display weekly schedule.

## 3.15. Create guest account.

### 3.15.1. Description

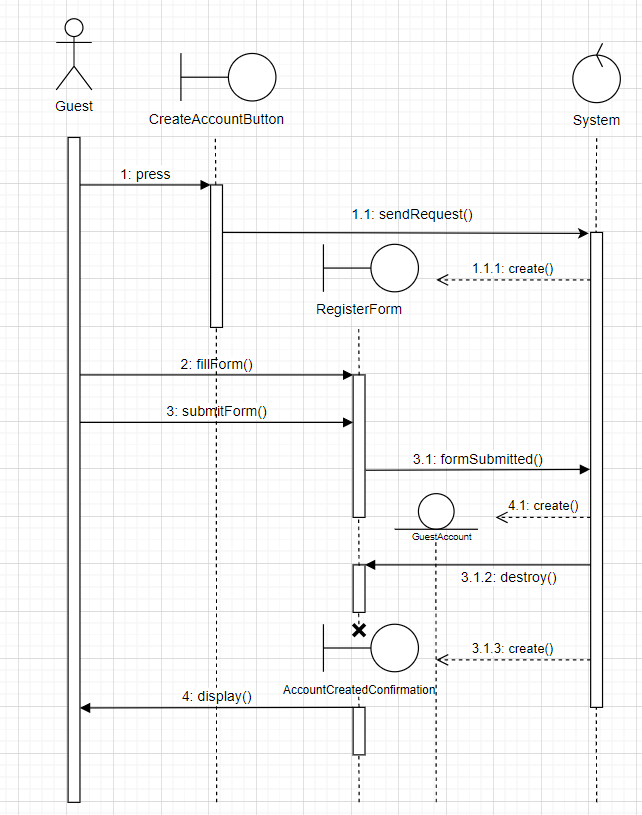


Figure 15. Create Guest account Sequence Diagram

Guest can create their account.

### 3.15.2. Functional Requirements

*Request for Creating guest account.*

*Introduction*

•Guest click the create guest account button

*Inputs*

•Guest click

*Processing*

•The hotel system require account information and waiting for Guest’s input.

## 3.16. Edit employee’s information.

### 3.16.1. Description

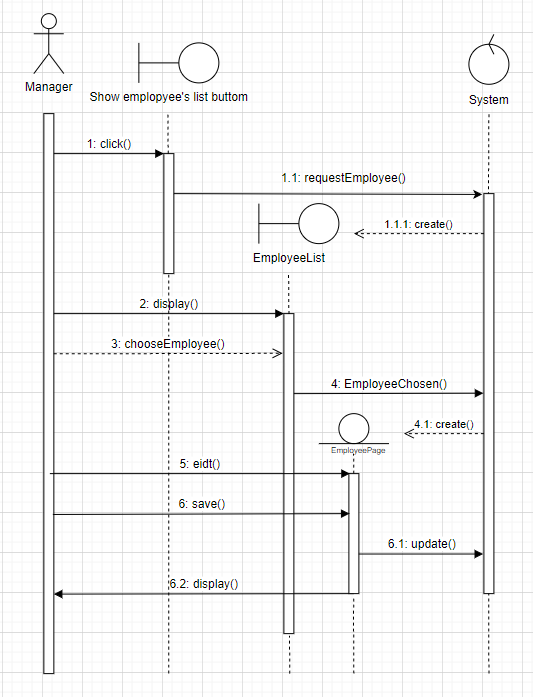


Figure 16. Edit an Emplyee’s task Sequence Diagram

Manager can edit an employee's information.

### 3.16.2. Functional Requirements

*Request for Editing employee information.*

*Introduction*

•Manager click show employee in button.

*Inputs*

•Manager click

*Processing*

•The hotel system display employee information and accpecting for an editing.

## 3.17. Refund payment.

### 3.17.1. Description

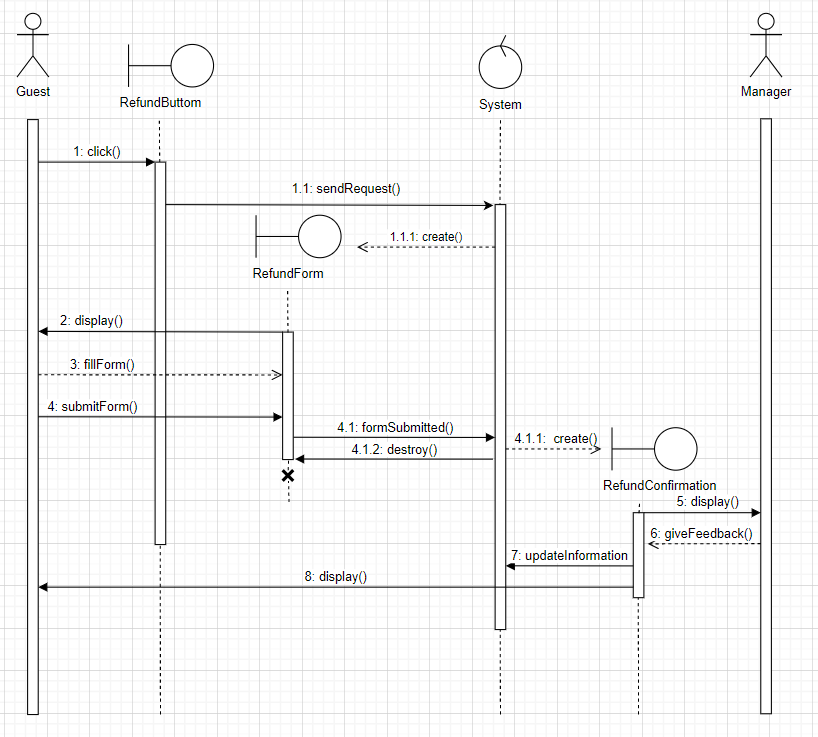


Figure 17. Refund payment Sequence Diagram

Guest can request for refunding payment.

### 3.17.2. Functional Requirements

*Request for Refunding payment.*

*Introduction*

•Guest click refund button

*Inputs*

•Guest click

*Processing*

•The hotel system send refund form and wait for an input.

## 3.18. Request room service.

### 3.18.1. Description

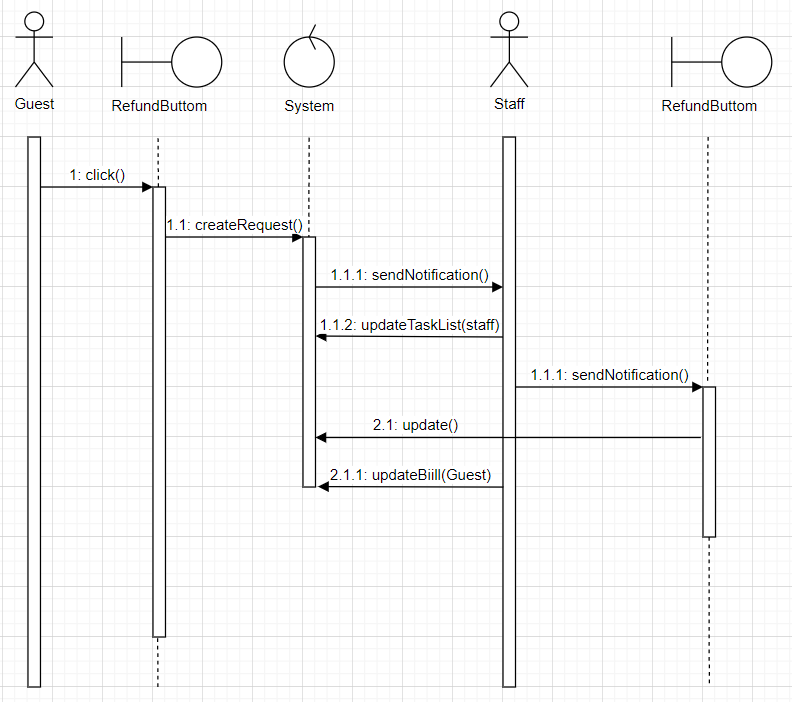


Figure 18. Room service Sequence Diagram

Guest can request room service.

### 3.18.2. Functional Requirements

*Request for Requesting room service.*

*Introduction*

•Guest click request room service button

*Inputs*

•Guest click

*Processing*

•The hotel system received room services request and update.

## 3.19. Request Bill.

### 3.19.1. Description

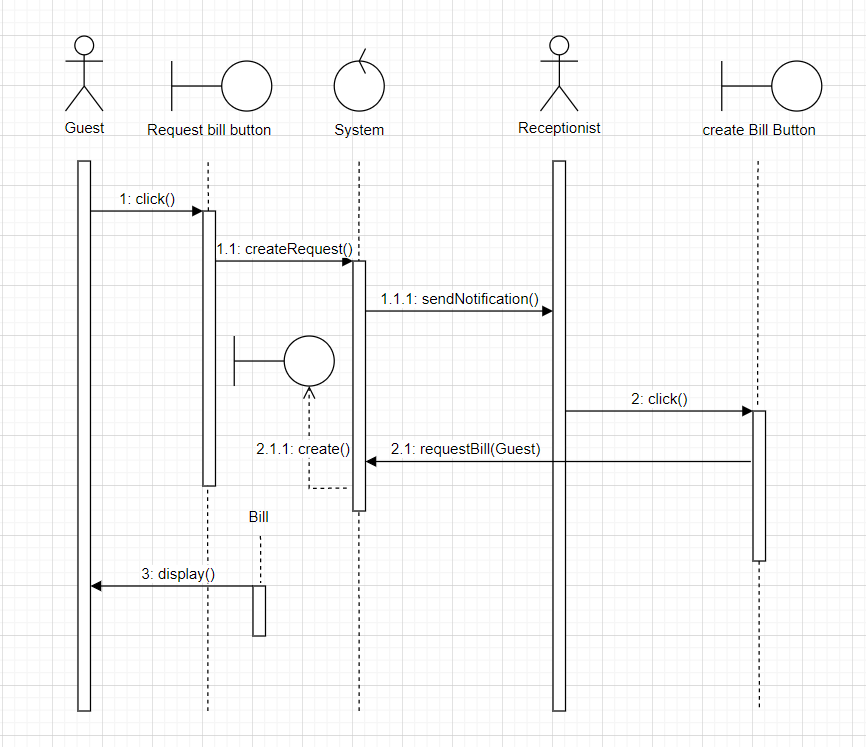


Figure 19. Request Bill Sequence Diagram

Guest can request bill.

### 3.19.2. Functional Requirements

*Request for requesting Bill.*

*Introduction*

•Guest click the request bill button

*Inputs*

•Guest click

*Processing*

•The hotel system received an request and display bill.

# **4. Data Requirements**

## 4.1. Logical Data Model

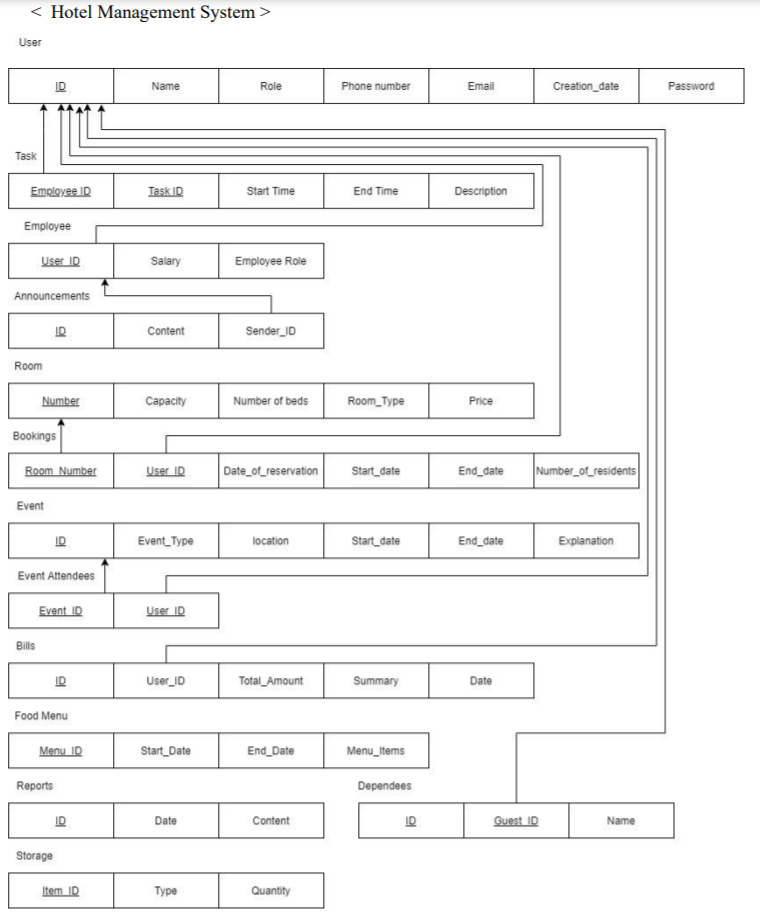


Figure 10. ERD

All the data is saved in the database:

* Guest information
* User information
* Admin information
* Employee information
* Manager information
* Receptionist information
* Staff information
* Storage Item
* Bill
* Food menu
* Task
* Event
* Room
* Report
* Announcement

The database allows concurrent access by various employees and is kept consistent at all the times requiring a good database design.

## 4.2. Data Dictionary

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Entity | Data Element | Description | Composition or Data Type | Length | Values |
| User | ID | ID of the User account | int | 20 |  |
| first\_name | First name of User account | string | 50 |  |
| Last\_name | Last name of User account | string | 50 |  |
| role | Role of user account | String | 20 |  |
| phone\_number | Phone number of user | int | 20 |  |
| email | Email of user | string | 50 |  |
| creation\_date | date of creating account | date | 20 |  |
| password | Password of user account | string | 20 |  |
| Report | id | ID of report | int | 20 |  |
| date | Date of report | date | 20 |  |
| content | Content of report | String | 100 |  |
| Anouncement | id | ID of anouncement | int | 20 |  |
| Content | Content of anouncement | string | 100 |  |
| Sender\_id | Id of sender | int | 20 |  |
| Room | id | Id of the room | int | 20 |  |
| capacity | The capacity of the room | int | 10 |  |
| number\_of\_beds | Number of bed of room | int | 10 |  |
| room\_type | The type of room | string | 10 |  |
| price | The price of room | int | 20 |  |
| Task | employee\_id | Id employee | int | 20 |  |
| task\_id | Id of task | int | 20 |  |
| start\_time | Time of starting | date | 10 |  |
| end\_time | Time of ending | date | 10 |  |
| description | The full description | string | 200 |  |
| Employee | salary | Employee’s salary | int | 15 |  |
| employee\_role | The role of employee | string | 20 |  |
| Event | id | ID of event | int | 10 |  |
| event\_type | Type of event | string | 20 |  |
| location | Location of event | string | 20 |  |
| start\_date | Time of starting | date | 20 |  |
| end\_date | Time of ending | date | 20 |  |
| explanation | Event explanation | string | 100 |  |
| Bill | id | ID of bill | int | 20 |  |
| user\_id | Id of user | int | 20 |  |
| total\_amount | The overall amount | int | 15 |  |
| summary | The summary of bill | string | 100 |  |
| date | Date of releasing bill | date | 20 |  |
| Food menu | menu\_id | ID of menu | Int | 20 |  |
| Start\_date | Start date of menu | date | 20 |  |
| End\_date | End date of menu | date | 20 |  |
| Menu\_items | All items from menu | string | 200 |  |
| Referee Country | Country of referee | string | 20 |  |

## 4.3. Data Integrity, Retention, and Disposal

DI-1: The HSM shall retain customer information for 3 days after the customer check out.

DI-2: The HSM shall retain room booked information for a day after the room booked.

# **5. External Interface Requirements**

## 5.1. User Interfaces

Customer Interface

The HSM screen displays an interface for the user to choose features for him to review hotel room or service information.

Manager Interface

The HSM screen displays an interface to communicate with HSM system.

## 5.2. Software Interfaces

Booking room Interface

The HSM screen displays an interface for the user to type in room information like number of bed, room type, price to book room.

See guest information Interface

The HSM screen displays an interface for the user to see guest information.

Check dairy task Interface

The HSM screen displays an interface for the admin see employee’s task and an interface for employee to their own task.

Check the event Interface

The HSM screen displays an interface for the guest to see a full list of hotel event schedule.

Change salary Interface

The HSM screen displays an interface for the admin/manager to have a list of employee’s salary and have permittion to change it.

Change room detail Interface

The HSM screen displays an interface for the admin to change each room detail and wait for the changing.

Cancel event Interface

The HSM screen displays the list of hotel events and buttom to cancel event.

Edit an employee’s task Interface

The HSM screen displays an interface for the admin to view all the current task of all employee and an interface for the admin to change task.

Log into system Interface

The HSM screen displays an interface for the user to type in information to log in.

Get report from employee’s task Interface

The HSM screen displays an interface for the admin see report of employee’s task.

Request day off Interface

The HSM screen displays an interface for the employee to fill in the day off and submit.

Create employee shift Interface

The HSM screen displays an interface for the admin to change a list of employee’s working time and set up for a new one.

Check employee’s shift Interface

The HSM screen displays an interface for the admin/employee to see the shift of employee.

Check weekly schedule Interface

The HSM screen displaysan interface for the admin/employee to see the schedule of employee.

Create guest account Interface

The HSM screen displays an interface for the guest to type in information like name, age, phone number, email, address to create account.

Edit employee’s information Interface

The HSM screen displays an interface for the admin to see the information of employee and interface to change the information.

Refund payment Interface

The HSM screen displaysan interface for the guest to see the payment of room booked and an interface to refund it.

Request room service Interface

The HSM screen displays an interface for the guest a list of available service to choose.

Request bill Interface

The HSM screen displays an interface for the guest the list of items in bill and price.

## 5.3. Hardware Interfaces

No hardware interfaces have been identified.

## 5.4. Communications Interfaces

|  |  |
| --- | --- |
| CI-1 | The HSM shall send an email to the User to confirm his register request. |

# **6. Quality Attributes**

## 6.1. Usability Requirements

|  |  |
| --- | --- |
| USE-1 | The HSM shall allow an User to view hotel room information with a single interaction. |
| USE-2 | 99% of new Users shall be able to successfully view information without errors on their first try. |

## 6.2. Performance Requirements

High level of performance requires high speed network and high level of connectivity.

## 6.3. Reliability Requirements

The available server must be reliable and the network connectivity in the supermarket should be proper for smooth flow of all operations and data.

## 6.4. Security Requirements

|  |  |
| --- | --- |
| SEC-1 | Every user of the software is provided a unique login username and a password which is stored in the SQLite Library database. |
| SEC-2 | Users shall be required to log on to the HSM for all operations. |
| SEC-3 | Only admin/manager shall be permitted to work with the system. |
| SEC-4 | The system shall permit Users to view only what they choose to view. |

## 6.5. Availabilty Requirements

The software is available for use all the time the hotel is still working.

## 6.6. Inverse Requirements

|  |  |
| --- | --- |
| INV-1 | The software does not allow the customer with lack of information who book room. |
| INV-2 | The software does not allow any other person except the managers or admin to change the system information. |