|  |
| --- |
|  |
| Cristor |
| CS 2062 – Object Oriented Software Development |
|  |
|  |
|  |

|  |  |
| --- | --- |
| A.B.P.R.Lakshani | : Team leader |
| H.L.M. Hansini | : Document manager: |
| S.G. Kumarawadu | : System Analyst |
| C.M.N.D. Pathirana | : Chief Developer |

Contents

[Introduction 2](#_Toc447658504)

[Client 3](#_Toc447658505)

[Business Problem 4](#_Toc447658506)

[Vision 5](#_Toc447658507)

[System objectives: 5](#_Toc447658508)

[Business benefits 5](#_Toc447658509)

[System capabilities 5](#_Toc447658510)

[System scope 5](#_Toc447658511)

[Inception Phase 6](#_Toc447658512)

[Introduction 6](#_Toc447658513)

[Stakeholder Analysis 6](#_Toc447658514)

[Risk Identification and Feasibility analysis 7](#_Toc447658515)

[Requirement analysis 9](#_Toc447658516)

[Requirement gathering methods 9](#_Toc447658517)

[Essential Use Case list 10](#_Toc447658518)

[Non-Functional requirements 11](#_Toc447658519)

[Elaboration Phase 12](#_Toc447658520)

[Introduction 12](#_Toc447658521)

[Domain Modeling 13](#_Toc447658522)

[Event table 13](#_Toc447658523)

[Use case diagram 16](#_Toc447658524)

[Use case detailed descriptions 17](#_Toc447658525)

[Usecase Detailed Descriptions for non-essential usecases 30](#_Toc447658526)

[Activity Diagrams 33](#_Toc447658527)

[System Sequence Diagrams 43](#_Toc447658528)

[Domain class diagram 49](#_Toc447658529)

[Conclusion 50](#_Toc447658530)

[Tools Used 50](#_Toc447658531)

[Project Status 53](#_Toc447658535)

[Problems encountered 54](#_Toc447658536)

# Introduction

**Composition and responsibilities of each team player:**

* Team Leader: A.B.P.R. Lakshani
  + Coordinating with internal and external communication during the project
  + Responsible for overall project
* Document manager: H.L.M. Hansini
  + Responsible for maintenance of project documentation and final preparation
* System Analyst: S.G. Kumarawadu
  + Dealing with the business environment and analysing the requirements of the client
* Chief Developer: C.M.N.D. Pathirana
  + Responsible for guiding and maintain technical side of the project while developing the software systems in accordance with the client’s requirements

## Client

**Name of the business**: “Hotel T.K. Green Garden”

**Register Number** : MA/1/227

**Contact Details**

1. Name of the client: Mr. Tennyson Kumarawadu
2. Position hold/relationship to the owner Proprietor of the business
3. Postal Addres Hotel T.K. Green Garden,

Polhena,

Matara

1. Telephone 041-2222603

**Company structure**

The Hotel consists of 16 rooms with A/C and non A/C facilities including 5 exceptional luxury rooms, 1 wedding hall which has the capacity of nearly 300 people. In addition, the selected organization is expecting to extend their service to start a catering service for the area.

**Staff details**

Figure 1.1: Hierarchical view of the staff

## Business Problem

The client expects an automated information system to keep track of the bookings (room and hall) and two simple integrated subsystems to keep track of details of the staff members and transaction details for each day.

Furthermore in order to achieve the goal of the expansion plan for next 3 years, client wishes to have a website for the selected organization which will help in providing the facility of advertising and online room bookings, in order to a give broad exposure for the company.

## Vision

### System objectives:

Increase the efficiency and reliability in the process of information management in the business by replacing the manual information system which they currently use.

### Business benefits

* Replace manual methods by computerized system.
* Provide online boking facility for customers.
* Provide a user friendly and centralized information system with controlled access to each party.
* Publicize services and facilities provided by the hotel.

### System capabilities

* Supportive for booking and transactions recording processes.
* Support traditional telephone room bookings
* Monthly/ annual reports should be generated by the system itself and should be accessible by the manager or the relevant party.
* Customer logs and details will be stored with secured access.
* System is capable of storing staff details • Maintain all the databases and keep information history.

### System scope

* Manager of the hotel is provided with an interface which has access to all the details and the ability to enter details.
* Receptionist should be provided with an interface which has the ability to access information about the available rooms and wedding halls and enter details on bookings.
* System should have a database which maintains payment information for each customer and should be accessible to check the current bill.

# Inception Phase

## Introduction

## Stakeholder Analysis

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Stake Holder Category** | **Stake Holder name** | **Project role** | **Product focus** | **Schedule focus** | **Cost focus** |
|  |  |  |  |  |  |
| Executive | Tennyson Kumarawadu | Steering committee | High | High | Hight |
| Operational Staff | Ishara  Karunatilake | Requirements definition | Low | Low | Low |
| Business users |  | Requirement definition | Low | Low | Low |
|  |  |  |  |  |  |

## Risk Identification and Feasibility analysis

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Risk description** | **Potential impact on project** | **Likelihood of Occurrence** | **Difficulty f timely Anticipation** | **Overall threat** |
| Critical team member not available | High | Low | Low | Medium |
| Organizational employees not compute savvy | Medium | Low | Low | Low |

**Identified risks in the system**

* A low level of computer competency
* Substantial computer phobia
* Fear of loss of employment due to increases automation.

* **Cultural feasibility**

In our current project we identified Manager and Receptionist as the people who will interact with the system highly. Here the Manager is computer literate and is a person with basic computer knowledge and also someone capable of adapting to new environments quickly.

But when it comes to the Receptionist, we recognize she has poor computer knowledge but she as the ability to grasp new things and change accordingly. In order to minimize the risk and to establish the system successfully we are planning on conducting session.

* Resource feasibility

Here the owner of the hotel is capable of supplying required resources to the system.

* Schedule feasibility

There are few risks which may cause a huge latency to complete the project before the deadline. This happens due to that the working team has to manage their studies and exams. In order to minimize that we have divided the work in to several stages and scheduled each work in order to make sure we can complete the project in time.

## Requirement analysis

### Requirement gathering methods

In order to gather detailed information required to build the project we mainly used following methods.

* Using interviews
* Reviewing previous manually created documents
* Reviewing booking websites currently in operation

**Interviews**

Interviews were conducted with the Manger who is the owner of the Hotel and the main stake holder of the project. And also we conducted several discussions with the receptionist who will actively engage with the deployed system.

**Reviewing previously created documents.**

As we are focused on replacing the manually operated system (currently on operation) using an automated system. We got information related to the current process from the documents maintained by the manager and the Receptionist.

**Reviewing booking websites currently in operation**

As we are new to the field of website development, we launched a research to identify the factors which are required to building an eye catching website to promote the Hotel as well as to facilitate online booking. For that we reviewed websites of Famous hotels in Sri Lanka.

### Essential Use Case list

1. Reserving rooms (General)
2. Reserving rooms (online)
3. Customer check availability details
4. Hall booking
5. Register payments
6. Adding expenditure of the day
7. Adding details of total gain of the day
8. Add new staff member
9. Updating details of staff members
10. System update request

### Non-Functional requirements

* **Security requirements:**

The deployed system is needed to have controlled access to different parts of the system by different parties. As an example the Manager will be given full access to the system while the receptionist is capable of accessing booking section.

* **Technical requirements**

As we have identified, it is required to add a new desktop PC to the hotel with Windows operating system which can acts as the sever for the website.

* **Usability requirements**

As we have identified, Interface elements (e.g. menus) should be easy to understand and the user documentation and help should be complete, consistent, and context sensitive and explain how to achieve common task. And also the system should be easy to learn

# Elaboration Phase

## Introduction

|  |  |
| --- | --- |
| **Team Player** | **Contributions** |
| A.B.P.R.Lakshani | Guiding team members  Preparing event table  Preparing system sequence diagrams  Preparing activity diagrams |
| H.L.M. Hansini | Preparing activity diagrams  Preparing use case diagrams  Preparing system sequence diagrams  Preparing Use case detailed descriptions |
| S.G. Kumarawadu | Preparing domain class diagram for the system.  Preparing activity diagrams  Preparing system sequence diagrams  Preparing Use case detailed descriptions |
| C.M.N.D. Pathirana | Designing logo of the company  Preparing activity diagrams  Preparing system sequence diagrams  Preparing Use case detailed descriptions |

**Note:** Here the responsibilities of the project were assigned in a vertical responsibility division manner. Hence there was contribution from everyone while completing a single task.

## Domain Modeling

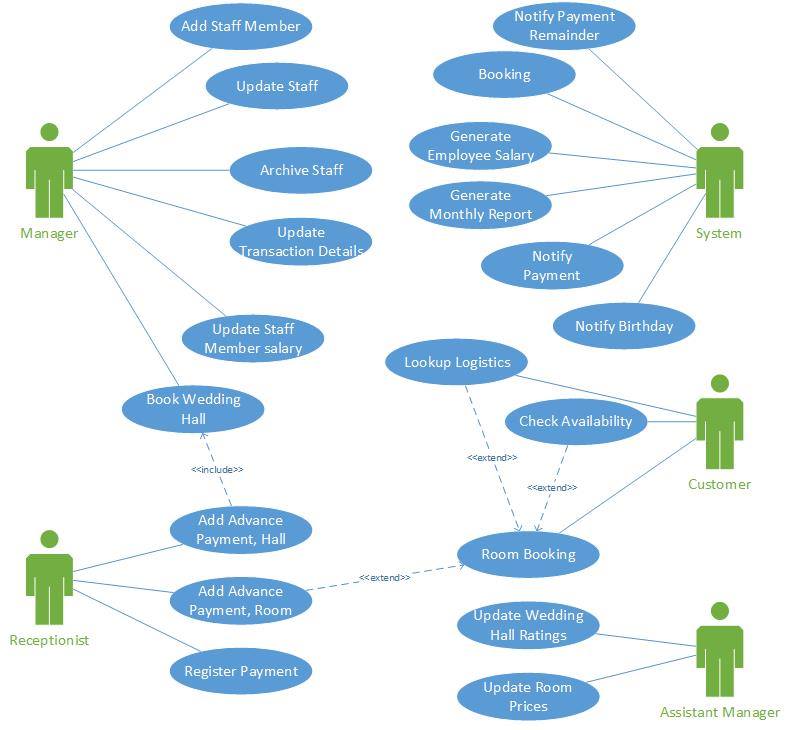
### Event table

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Event** | **Triggered** | **Use case** | **Source** | **Response** | **Destination** |
| 1. Room booking by customers | Booking request | Reserving room | Customer | Message of confirmation | Customer |
| 1. Customer Check availability of rooms | Room detail inquiry | Look up room details | Customer | Message of availability | Customer |
| 1. Customer asking for details | Logistic detail inquiry | Look up logistic details | Customer |  | Customer |
| 1. Wedding hall bookings by the customer | Wedding hall booking Request. | Book wedding hall | Manager  Or receptionist | Message of confirmation |  |
| 1. Updating the room prices and special Offers. | System update Request | Updating rates | Admin | Message of confirmation | Admin |
| 1. Update wedding hall ratings | System update Request | Updating rates | Assistant Manager | Message of confirmation |  |
| 1. Adding details of advance payments by the Manager | Transaction detail update request | Update transaction details | Receptionist | Message of confirmation |  |
| 1. Notify about wedding hall bookings | “2 days before the booked date” | Notification for Assistant Manager and Manager |  | Notification | Manager Assistant Manager |
| 1. Receptionist register payments | Transaction update request | Update payment | Receptionist |  |  |
| 1. Notify remainder to pay | 12.00am on the day of departure | Notify receptionist |  | Balance notification | Receptionist |
|  | | | | | |
| 1. Adding a staff member to the system by Manager | Add employee request | Adds a new employee | Manager | Message of confirmation |  |
| 1. Updating details of staff members Manager | Edit request | Update employee details | Manager | Message of confirmation |  |
| 1. Archiving staff member details by Manager | Staff detail change request | Updating archived staff table in database | Manager | Staff member detail being archived |  |
| 1. Generate employee salary report | “End of month” | Calculating salary of employees by the system |  | Employee salary report | Manager  Assistant manager |
| 1. Adding details of extra payments for employees by Manager | Salary update request | Update employee salary details | Manager  Or assistant manager | Message of confirmation |  |
| 1. Notify birthdays of employees | “1 hour before birthday” | System notification |  | Notification or a reminder | Assistant manager |
|  | | | | | |
| 1. Adding details  of the total gain for the day by Manager | Update transaction details | Transaction detail update request | Manager | Message of confirmation |  |
| 1. Adding details of expenditure of the day Manager | Update transaction details | Transaction detail update request | Manager | Message of confirmation |  |
| 1. Generate Monthly report by the system | Generating monthly report | “End of the month” |  | Monthly report | Manager |

**Event based approach**

### Use case diagram

Figure 3.2.3.0-Usecase diagram for the project



### Use case detailed descriptions

* **Room Booking (General)**

|  |  |  |
| --- | --- | --- |
| Use Case name | Reserving the room | |
| Scenario | Room booking by the customer | |
| Triggering event | Booking request | |
| Brief Description | When customer request for a room with a specified condition (single, double or triple), receptionist check whether the rooms are available or not through the system by retrieving room details from the database. If the rooms are available receptionist give a set of room numbers and then customer select a room and confirm booking. Receptionist adds a new booking with customer details and other advance payment details. If the rooms are not available, customer can either change the room details or cancel the booking. | |
| Actors | Receptionist | |
| Related use cases | Includes: Check room availability | |
| Stakeholders | Staff members: To arrange rooms for customers  Manager: To verify the total gain for the month | |
| Preconditions | Customer must exist  Requested rooms must exit | |
| Post conditions | New booking should be added to the database  Advance payment details should be updated if exist  Database should be up to date with vacant rooms | |
| Flow of events | Actor | System |
| 1. Customer come to the hotel and meet receptionist 2. Customer request for a room with the room condition 3. Receptionist checks for availability of the rooms 4. Receptionist adds a new room booking 5. Customer pays advance payment 6. Customer indicates the end of the booking | 3.1 Display available rooms    4.1 Add a new room booking to the database  5.1 Update booking details by adding advance payment  6.1 Complete the order |
| Exception conditions | 2.1  if the requested room is not available customer can  a. cancel the booking, or  b. request for a another room | |

* **Room Booking (Online)**

|  |  |  |
| --- | --- | --- |
| Use Case name | Room Booking | |
| Scenario | Online room booking | |
| Triggering event | Customer makes an online room booking request | |
| Brief Description | When a customer makes an online room booking request and enters the date + no. of people, the system checks the availability  If it is possible the customer can confirm the booking and provide more information, or skip the booking.  Else the customer can either change the date of the booking or cancel the booking | |
| Actors | Customer | |
| Related use cases | Check availability by the customer<<extends>> | |
| Stakeholders | Manager: To verify the income of the business per month  Receptionist: To verify the bookings per day  Customer: To verify the booking | |
| Preconditions | Room charges must be up to date in the website  Availability of the rooms must be updated in the website | |
| Post conditions | If the booking is confirmed, it must be recorded in the system calendar | |
| Flow of events | Actor   1. Customer makes an online room booking request   3. enter date + number of people  6.enters more information | System   1. Prompts for date + number of people   4. checks availability of the rooms  5.prompts more information  7. displays message of confirmation |
| Exception conditions | 4.1 if the room is not available on that particular day, the customer can either change or cancel the booking  If the customer changes the day the steps 2,3,4 will repeat until the booking is available to the entered day  5.1 if the customer doesn’t want to proceed with the booking, he can skip the booking after checking the availability. | |

* **Customer check availability details**

|  |  |  |
| --- | --- | --- |
| Use case name | Lookup room details | |
| Scenario | Customer checks availability of rooms online | |
| Triggering event | Customer enters date and no. of people to check room availability | |
| Brief Description | When a customer makes an online room booking request and enters the date + no. of people, the system checks the availability | |
| Actors | customer | |
| Related use cases | Online room booking <<extends>> | |
| Stakeholders | Customer: To verify the availability of the room | |
| Preconditions | The website should be updated with the latest room availabilities. | |
| Post conditions | none | |
| Flow of events | Actor   1. Customer makes an online room booking request   3. enter date + number of people | System   1. Prompts for date + number of people   4. checks availability of the rooms |
| Exception conditions | 4.1 customer can either proceed with the room booking or skip from the process of online room booking | |

* **Wedding hall booking**

|  |  |  |
| --- | --- | --- |
| Use case name | Wedding Hall Booking by the customer | |
| Scenario | Wedding Hall Booking by the customer | |
| Triggering event | Customer talks to the receptionist to book the hall | |
| Brief Description | When a customer talks to the receptionist to book a hall, the receptionist enter a new booking request in the system, provide the necessary information, system checks the availability of the particular booking, if it’s available more details are entered into the system and confirm the booking  If the hall is not available the system displays a message and cancels the booking request. | |
| Actors | Receptionist | |
| Related use cases | Check availability<<includes>>  Register payments by the receptionist<<extends>> | |
| Stakeholders | Manager: To verify the income of the business per month  Customer: To verify the hall booking | |
| Preconditions | Wedding hall booking rates must be up to date. | |
| Post conditions | If the booking was confirmed the hall must be allocated in the particular day for the customer and must be marked in the system calendar.  Else the booking should be deleted. | |
| Flow of events | Actor   1. Receptionist verify the booking request from the customer 2. Receptionist enter a hall booking request to the system 3. Customer provides the information 4. Receptionist enter the information   7. receptionist enter package + no.of people  8.receptionist enter modifications to the booking | System  2.1 A new booking is created   1. checks the availability of the booking 2. prompt for package + no.of people   9. display a message of confirmation |
| Exception conditions | 5.1 if the booking is not available the booking is cancelled  8.1 if there are no modifications to be done the system displays a message of confirmation | |

* **Adding expenditure of the day**

|  |  |  |
| --- | --- | --- |
| Use Case name | Enter details of expenditure | |
| Scenario | Enter details of expenditure of the day by the manager | |
| Triggering event | Expenditure detail update request | |
| Brief Description | On every expenditure(apart from the salary payments) done by the business, the manager make a request to the system to keep track of the expenditures(the amount and the purpose), Finally update the database of the system | |
| Actors | Manager | |
| Related use cases | None | |
| Stakeholders | Manager: To verify the income of the business per month | |
| Preconditions | None | |
| Post conditions | Record the payments in the database | |
| Flow of events | Actor   1. Manager verifies the expenditure 2. Manager enters the amount and the purpose to the database 3. Manager do the payments for the external party 4. Repeat steps 1,2,3 for every expenditure in the business | System  2.1 Database get updated accordingly |
| Exception conditions | 3.1 if the payment is above Rs.10,000 the receptionist must get the permission from the proprietor | |

* **Adding total gain of the day**

|  |  |  |
| --- | --- | --- |
| Use case name | Add gain of the day by the Manager | |
| Scenario | Add gain of the day by the Manager | |
| Triggering event | Manager makes an “add gain of the day request” to the system. | |
| Brief Description | At the end of the day, manager makes a “add gain of the day request” to the system, and if there are any additional payments for the day, manager enter them into the system.  Finally manager verify the accuracy of the report. | |
| Actors | Manager | |
| Related use cases | Calculate the monthly report by the system <<extends>>  Register payments by the receptionist <<extends>> | |
| Stakeholders | Manager: to verify the monthly income of the business | |
| Preconditions | None | |
| Post conditions | Update the database | |
| Flow of events | Actor   1. The manager makes a “update gain of the day” request to the system   4. enters additional payments  7. manager checks the accuracy of the report | System  2. opens a new window displaying the total payments  3. system prompts for additional payments of the day  5. calculate the total gain of the day  6. display the detailed report to the manager    8. display the message of confirmation |
| Exception conditions | 3.1 if there are no any additional payments the system will calculate the income directly by the payments displayed in the window  7.1 if the manager encounters any faults in the report the system deducts the additional gains and steps 3,4,5,6 will repeat | |

* **Add a staff member**

|  |  |  |
| --- | --- | --- |
| Use Case name | Adding a new staff member to the system | |
| Scenario | Adding a new staff member to the system by the manager | |
| Triggering event | Manager decides to recruit a new staff member to the business. | |
| Brief Description | When the manager decides to recruit a new member to the staff, manager makes a “Add a new staff member” request to the system, then the system creates a new staff member account which will link the “Staff member” database.  The account will exist in the system until the staff member is vacated. | |
| Actors | Manager | |
| Related use cases | Generate employee salary report << extends > >  Notify birthdays of employees<<extends>>  Updating details of staff members Manager <<extends>>  Adding details of extra payments for employees by the manager<<extends>> | |
| Stakeholders | Staff member: To verify his recruitment and salary payments  Manager : To verify the income of the business per month | |
| Preconditions | No duplicate accounts must be exist in the system | |
| Post conditions | Once a new staff member is added a new row in the “Staff member” database should be allocated to the particular Staff member account.  When the member is vacated the particular account should be deleted from the system but the database will contain the information until manager declares it is unwanted. | |
| Flow of events | Actor   1. Manager makes a “Add a new staff member” to the system. 2. Staff member provides his information 3. Manager enters the information | System   * 1. system creates a new “Staff member” account which reflects to the “Staff member” database.  1. finalize the account |
| Exception conditions | None | |

* **Update staff member details**

|  |  |  |
| --- | --- | --- |
| Use case name | Update employee details | |
| Scenario | Update employee details by Manager | |
| Triggering event | Staff member informs manager about the change of his details | |
| Brief Description | When information of a staff member (ex: address, telephone number etc.) is changed, he informs manager. Then the manager logs into the system and accesses the staff member’s account in order to change details. The system will display the current information and Manager can edit that information. Then the manager asks for the changed details of the employee. Manager updates the relevant employee’s account according to the details he provides. | |
| Actors | Manager | |
| Related use cases | - | |
| Stakeholders | Manager: To keep records of the non-professional details of employees for contact purposes etc.  Employee: To get contact information of other employees for work purposes. | |
| Preconditions | Relevant employee should have been added to the system before updating his details. | |
| Post conditions | Employee details database should be updated accordingly.  Old details should be deleted. | |
| Flow of events | Actor   1. Staff member informs the manager about the change of his details 2. Manger log into the relevant staff members’ account   4. update the details of the staff member | System  3.displays the old information  5.displays a messageof confirmation |
| Exception conditions | None | |

### Usecase Detailed Descriptions for non-essential usecases

* **Generate employee salary reports**

|  |  |  |
| --- | --- | --- |
| Use Case name | Generate employee salary reports | |
| Scenario | Generate the employee salary reports on a fixed date of the month | |
| Triggering event | “End of month” | |
| Brief Description | At the end of the month the system connects to the employee database, retrieves the data from the employee accounts, Calculates the salary accordingly, Create new salary reports separately as per the number of employee accounts, Verify the salary payments, Record the payments in the database | |
| Actors | Manager/Administrator | |
| Related use cases | Validate employee account <<includes>>  Check for additional payments(new year allowance etc.) <<includes>> | |
| Stakeholders | Manager: To verify the income of the business per month | |
| Preconditions | When a new employee is recruited, a new account is created to the employee with all the information needed  Accounts must be up to date with the working days of the employees  When an employee is vacated the account should be deleted.(hence the number of employees = number of employee accounts in the system) | |
| Post conditions | The salary payment should be done within 24 hours after the report has been generated.  Record the payments in the database | |
| Flow of events | Actor   1. Administrator sets a date to generate the employee salary reports | System   1. Connects to an employee account 2. Retrieves the data(Name, month, no. of worked days etc.) from the employee account 3. a.) If the employee has taken more holidays than allowed, then calculate the salary according to the information from the employee account   b.) If the employee has not taken any holidays do not process any calculations   1. if an employee has taken a loan/ an advance payment deduct it from the calculated salary 2. Check for the additional payments and amount to be paid at the end 3. Generate the employee salary report and display it 4. Repeat step 2,3,4,5 as per the no. of employee accounts in the system 5. System indicates the end of the process |
| Exception conditions | * 1. if an employee has worked more than the allowed work days, notify the manager before generating the salary report | |

* **Notify birthdays of employees**

|  |  |  |
| --- | --- | --- |
| Use Case name | Notify birthdays of employees | |
| Scenario | Notify birthdays of employees | |
| Triggering event | “At 12.00 am on the birthday” | |
| Brief Description | With the creation of a new employee account, the system keep track of the birthday details and notify the manger exactly on the birthday | |
| Actors | Administrator | |
| Related use cases | Add a new employee<<includes>> | |
| Stakeholders | Manager: To maintain a friendly environment with staff | |
| Preconditions | None | |
| Post conditions | None | |
| Flow of events | Actor   1. Administrator sets a date to notify | System  2.Notify the manager on the birthday |
| Exception conditions | None | |

### Activity Diagrams

* **Room Booking**

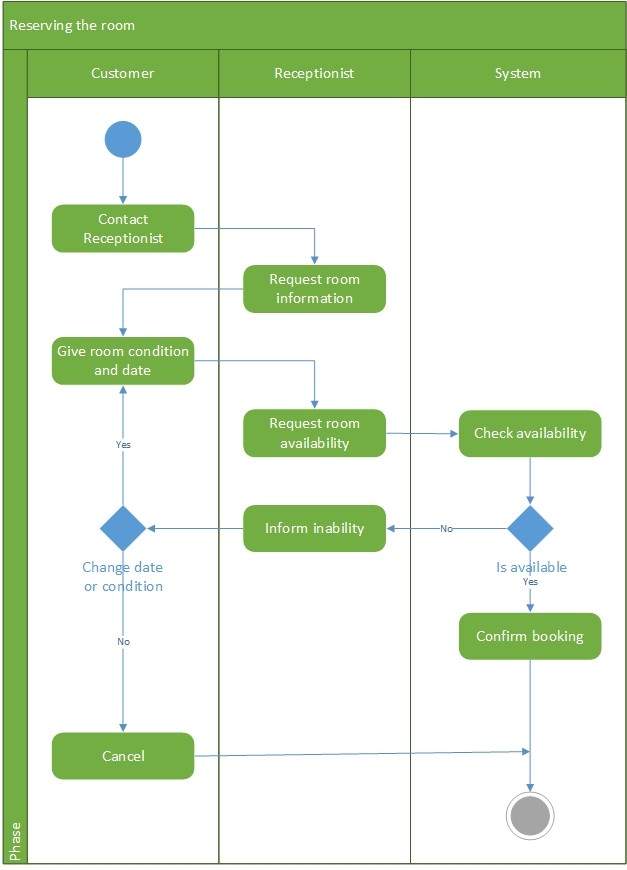
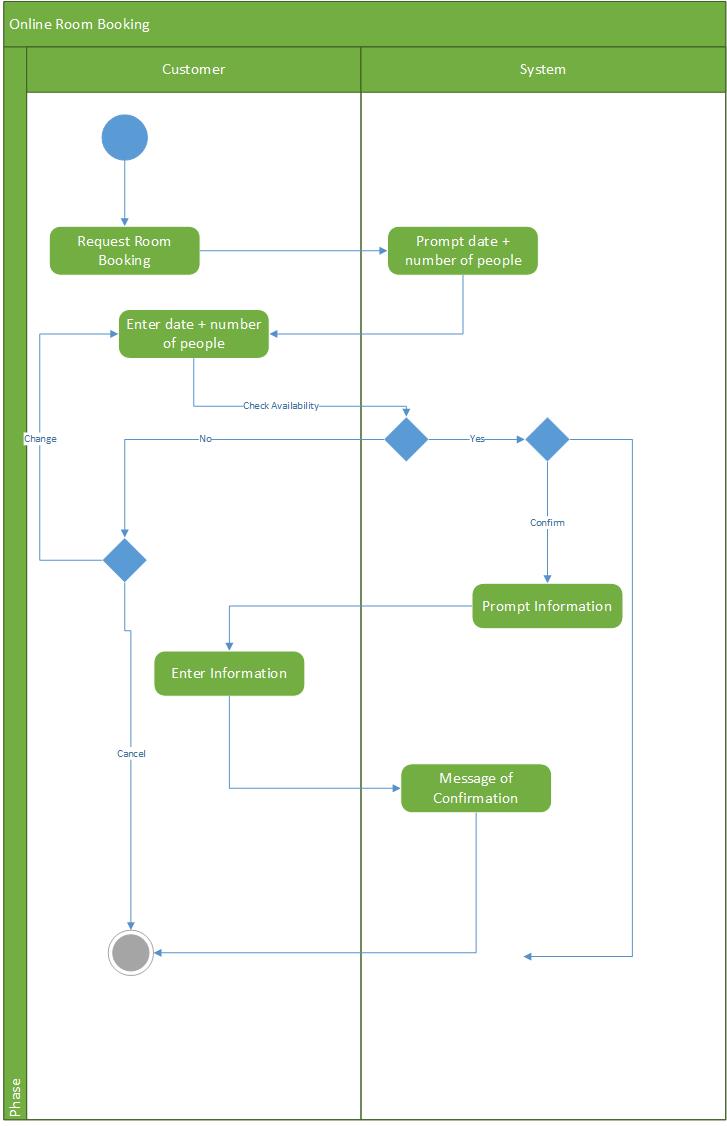


Figure 3.2.4.0-Activity diagram for Reserving rooms(General) usecase

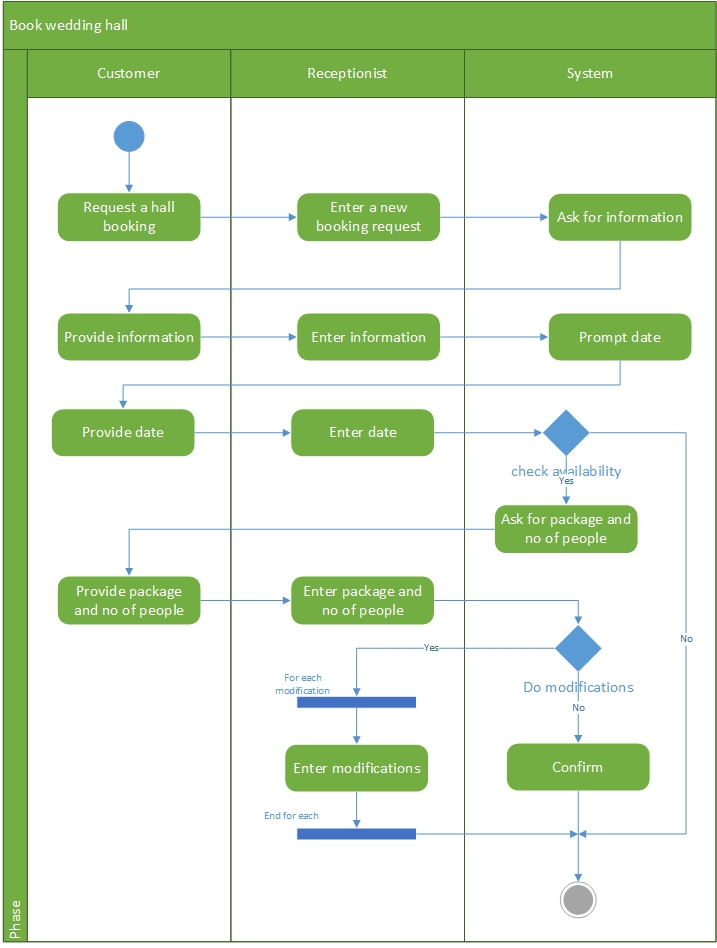
* **Reserving rooms (online)**

Figure 3.2.4.1-Activity diagram for Online room booking



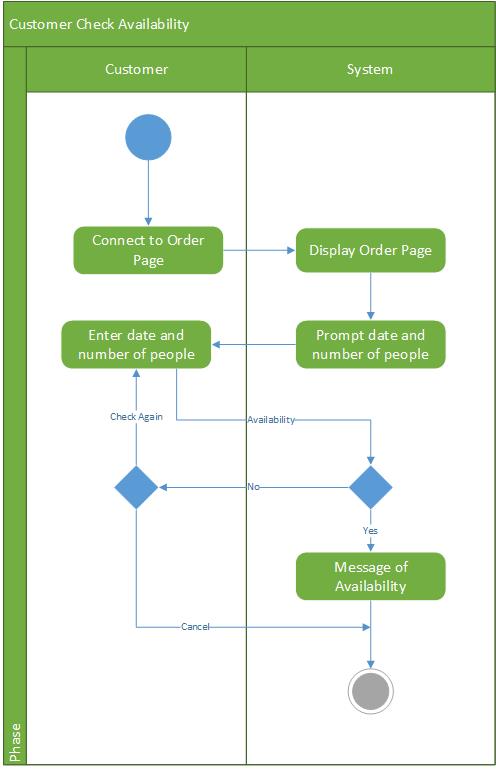
* **Wedding hall booking**

Figure 3.2.4.2- Activity diagram for booking wedding hall usecase



* **Customer check availability details**

Figure 3.2.4.3- Activity diagram for customer check availability



* **Register payments**

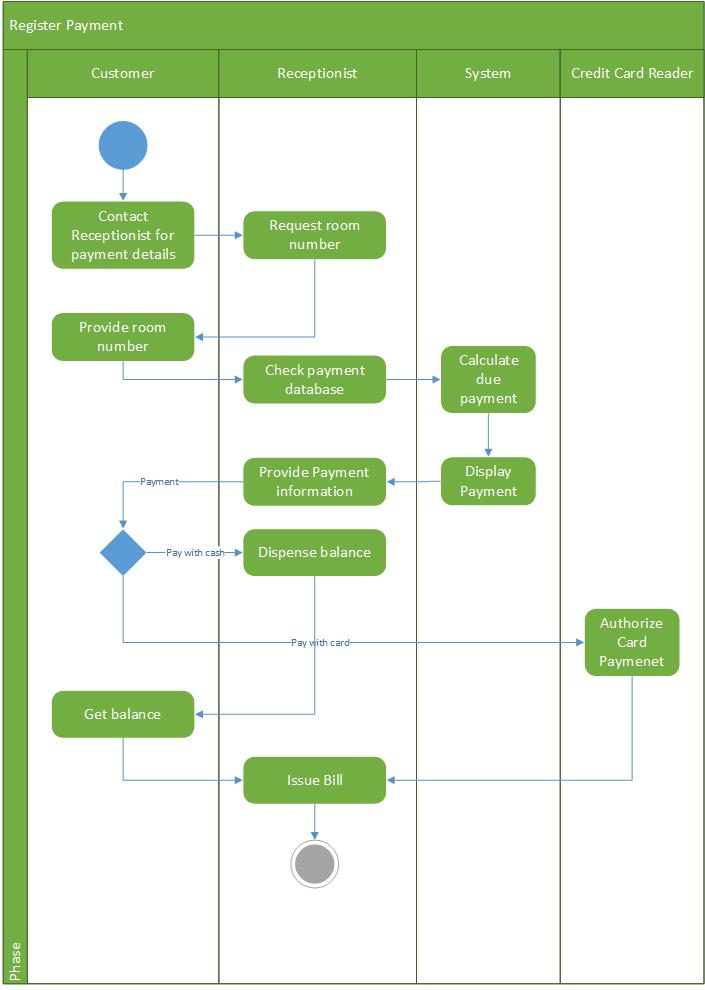
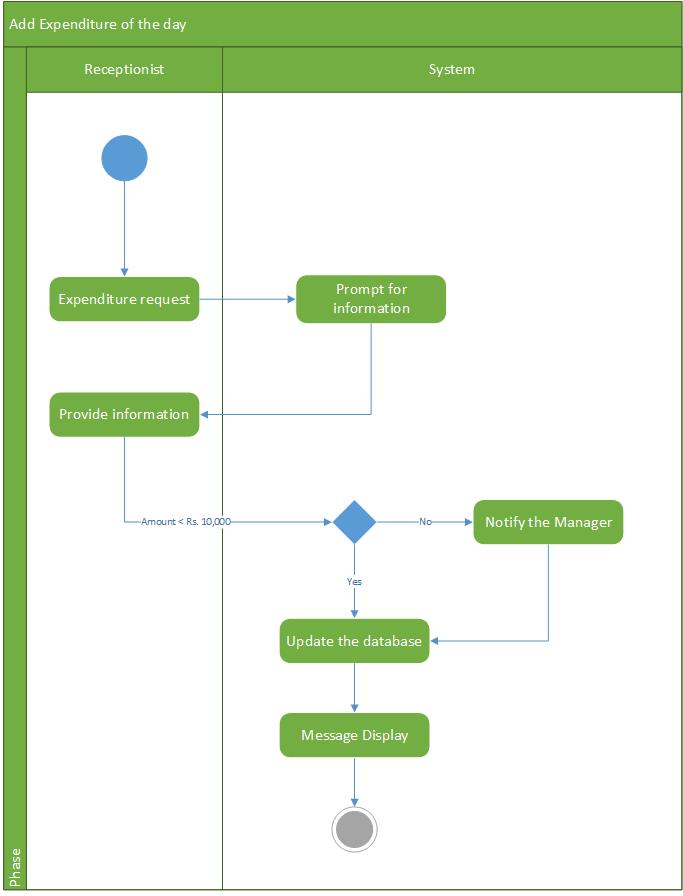


Figure 3.2.4.4- Activity diagram for register payments

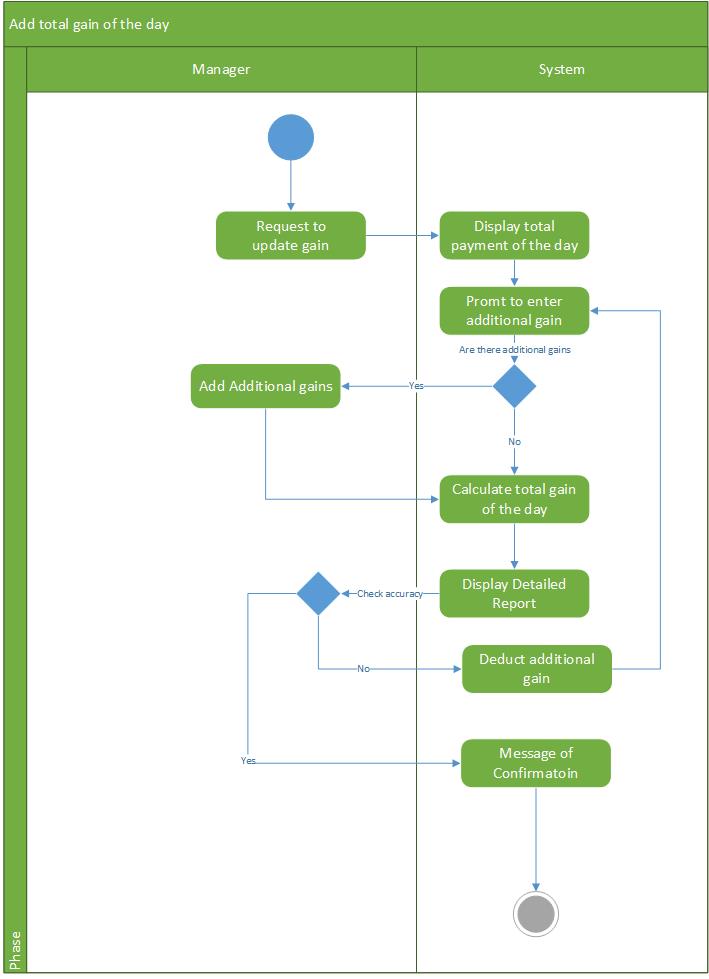
* **Adding expenditure of the day**

Figure 3.2.4.5- Activity diagram for adding expenditure of the day



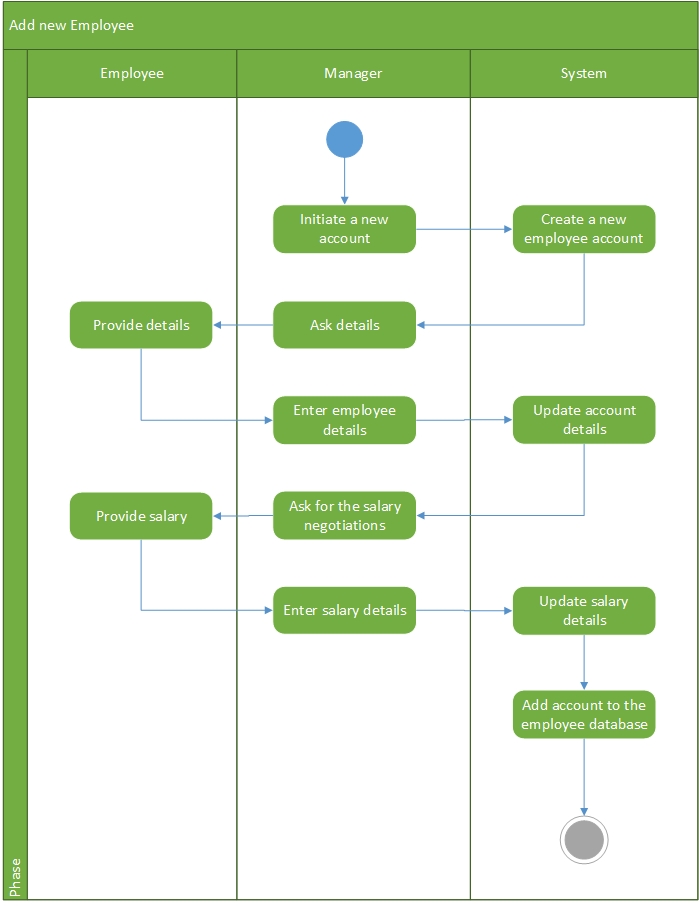
* **Adding total gain of the day**

Figure 3.2.4.6-Activity diagram for adding total gain of the day



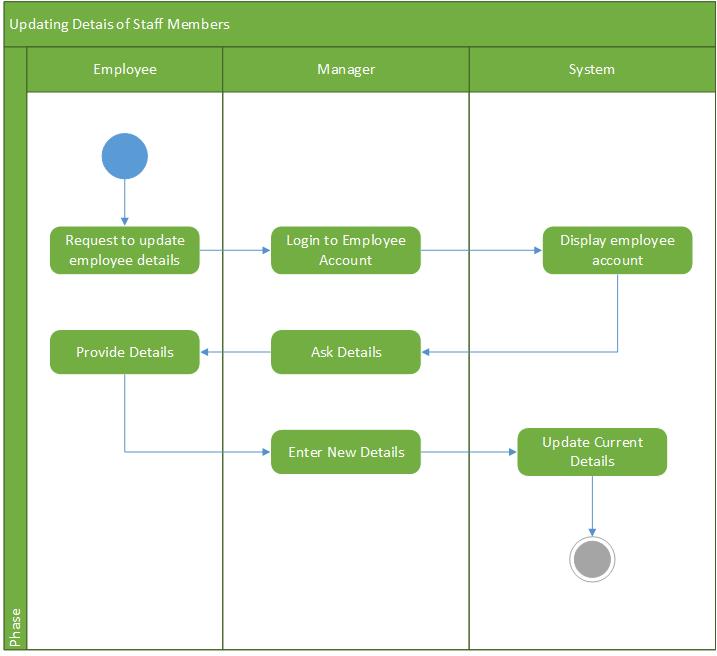
* **Add a new staff member**

Figure 3.2.4.7- Activity diagram for add a new employee



* **Update staff member details**

Figure 3.2.4.8- Activity diagram for update staff member



* **System update request**

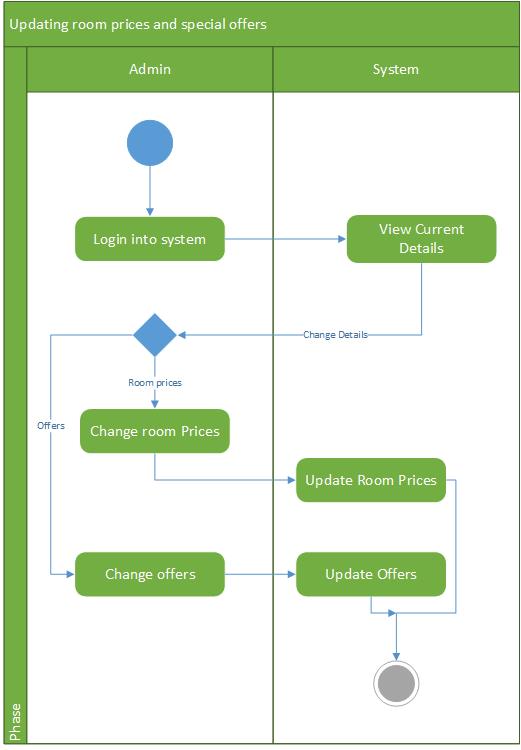
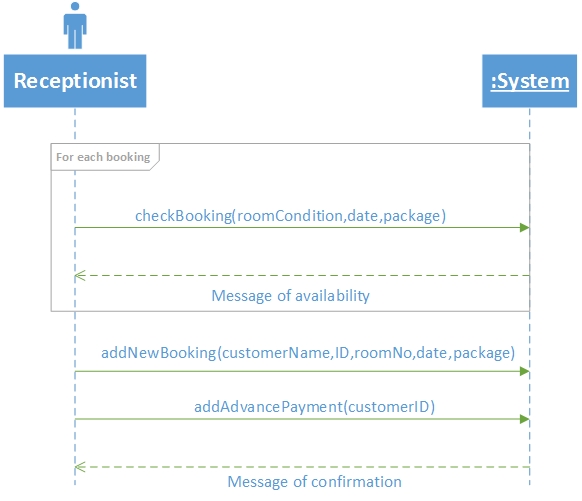


Figure 3.2.4.9- Activity diagram for updating room prices and special offers

### System Sequence Diagrams

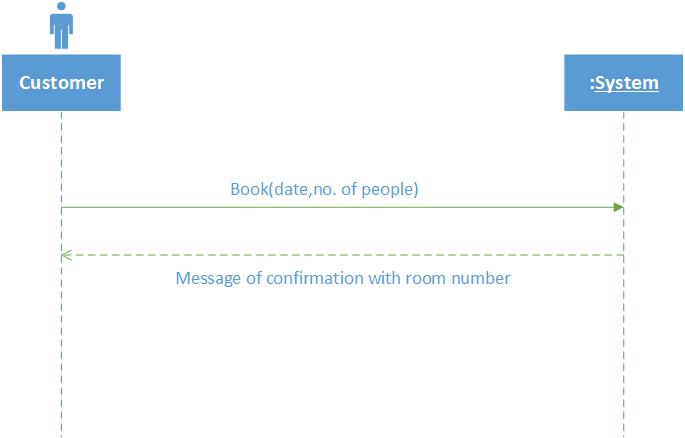
1. **Reserving rooms**

Figure 3.2.5.0 –System sequence diagram for general room booking



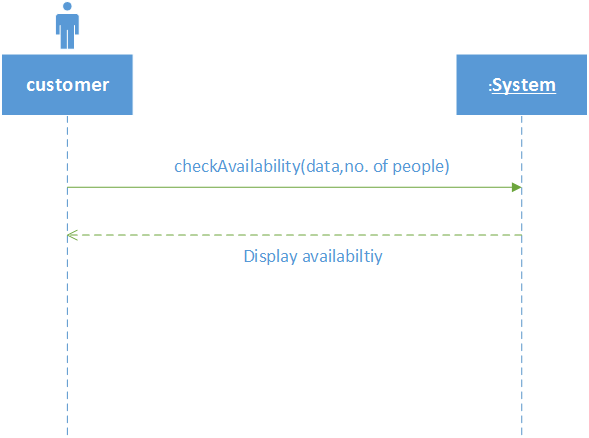
1. **Reserving rooms (online)**

Figure 3.2.5.1 System sequence diagram for online room booking



1. **Customer check availability**

Figure 3.2.5.2- System sequence diagram for check availability



1. **Wedding hall booking**

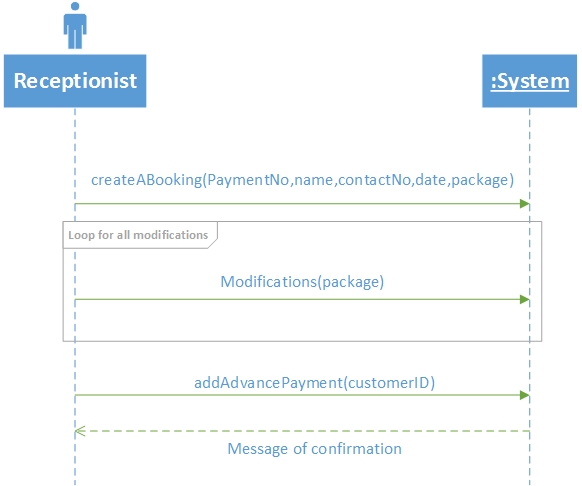
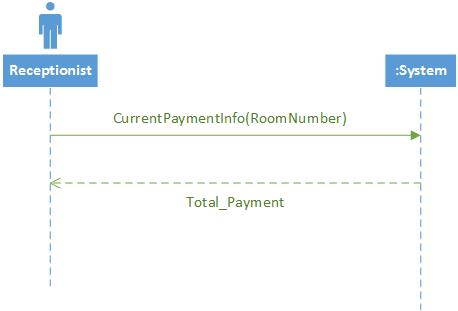


Figure 3.2.5.3- System sequence diagram for Wedding hall booking

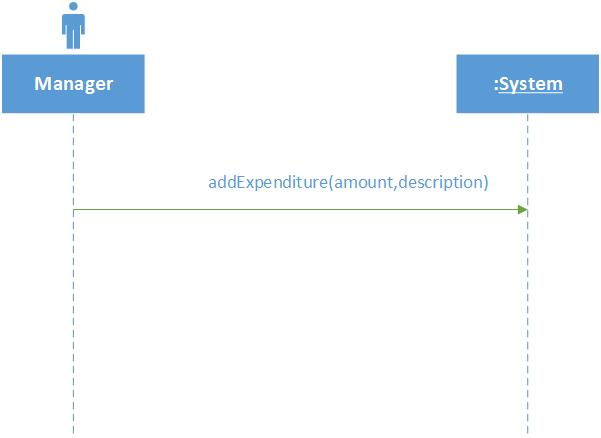
1. **Register payments**

Figure 3.2.5.4- System sequence diagram for register payments



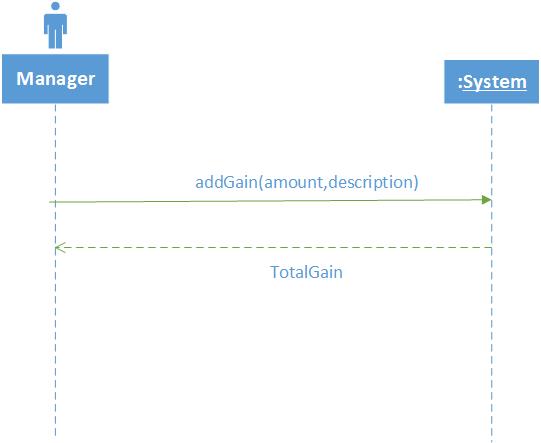
1. **Adding expenditure of the day**

Figure 3.2.5.5-System sequence diagram for adding expenditure



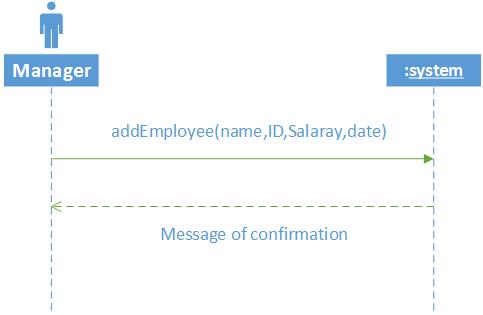
1. **Adding details of total gain of the day**

Figure 3.2.5.6- System sequence diagram for adding gain of the day



1. **Add new staff member**

Figure 3.2.5.7-System sequence diagram for add a new employee



h

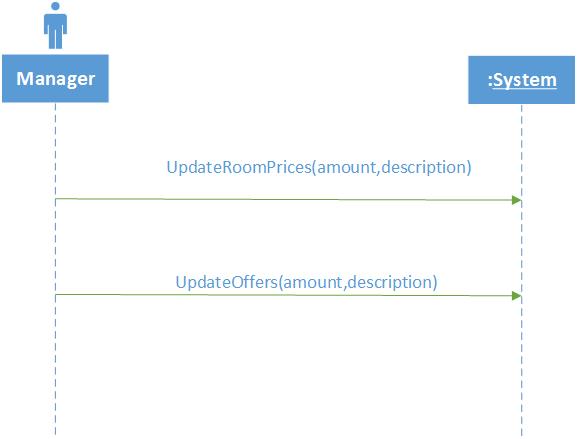
1. **Update staff member details**

Figure 3.2.5.8- System sequence diagram for updating staff member details



1. **System update request**

Figure 3.2.5.9- System sequence diagram for updating Room prices and offers



**Entity based approach**

### Domain class diagram

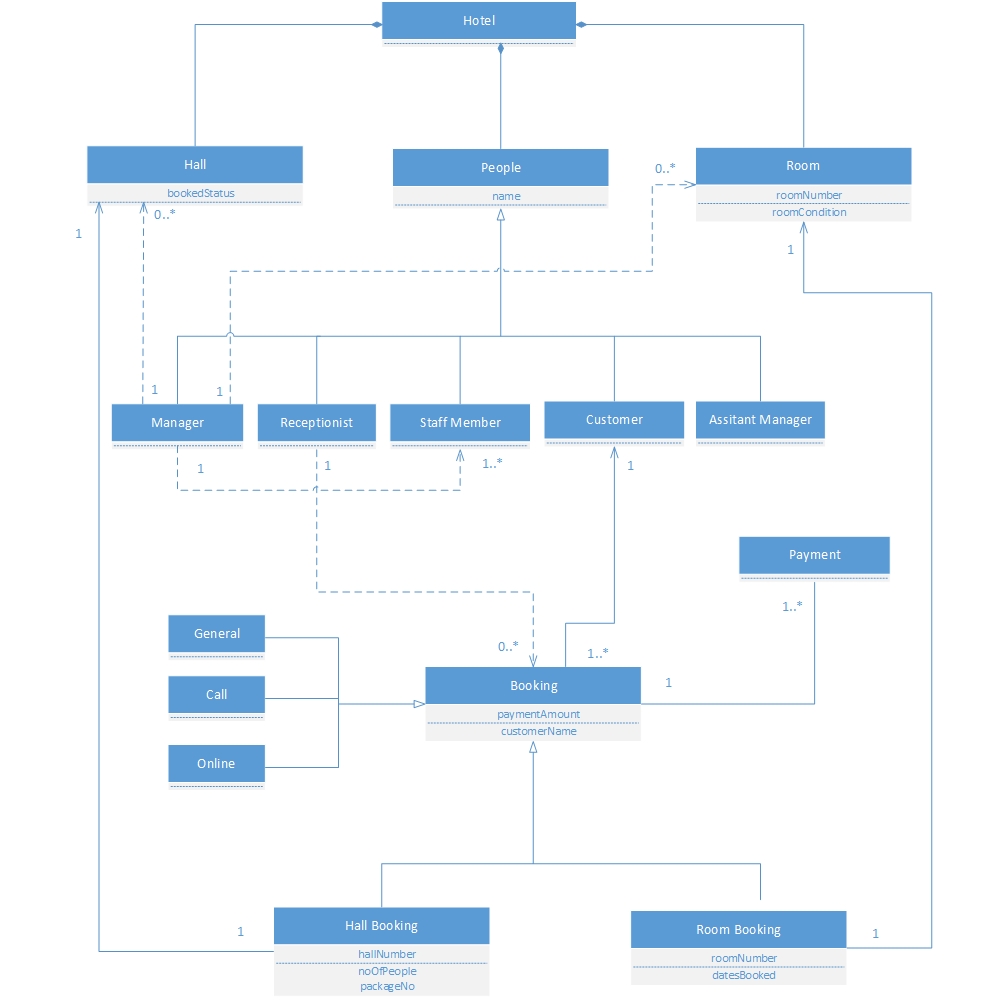


Figure 3.2.6.0- Domain class diagram for the project

# Conclusion

## Tools Used

* **Microsoft Visio**

Microsoft Viso is a diagramming and vector graphics application and is part of the Microsoft Office family. The product was first introduced in 1992, made by the Shapeware Corporation. It was acquired by Microsoft in 2000. Visio offers thousands of shapes that meet industry standards, including BPMN 2.0, UML 2.4 and IEEE (new). Users can easily create UML use case diagrams using Microsoft Visio. There are special communication shapes to indicate relationships between use cases and actors. The application supports for several file types so that the user can save the diagram in many file formats.

* **Microsoft project**

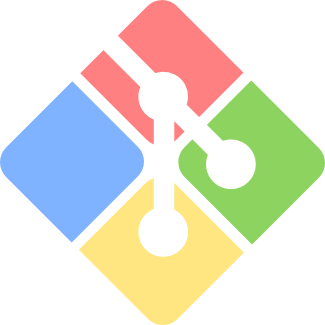
Microsoft Project is a project management software program, developed and sold by Microsoft, that is designed to assist a project manager in developing a plan, assigning [resources](https://en.wikipedia.org/wiki/Resource_(project_management)) to tasks, tracking progress, managing the [budget](https://en.wikipedia.org/wiki/Budget), and analysing workloads.



For our project we used Microsoft Project to build the Work breakdown system which lists all the activities and tasks needed to be completed in the project.

* **Git**

Git is a [free and open source](https://git-scm.com/about/free-and-open-source) distributed version control system designed to handle everything from small to very large projects with speed and efficiency.



In our project we started using Git to record changes to files that we are working on. As we are a 4 member team it helps us to compare changes over time, see who last modified something that might be causing a problem, who introduced an issue and when.

* **Adobe Photoshop**

Adobe Photoshop is a [raster graphics editor](https://en.wikipedia.org/wiki/Raster_graphics_editor) developed and published by [Adobe Systems](https://en.wikipedia.org/wiki/Adobe_Systems) for [Windows](https://en.wikipedia.org/wiki/Microsoft_Windows) and [OS X](https://en.wikipedia.org/wiki/OS_X).

In our project we used Adobe Photoshop for designing purposes.

* **Netbeans:**

NetBeans is a [software development](https://en.wikipedia.org/wiki/Software_development) platform written in [Java](https://en.wikipedia.org/wiki/Java_(programming_language)). The NetBeans [Platform](https://en.wikipedia.org/wiki/Platform_(computing)) allows applications to be developed from a set of modular [software components](https://en.wikipedia.org/wiki/Software_component) called modules.



In our project we started using Netbeans IDE as it is a Fast & Smart Code Editing tool which facilitates Unit testing needed to verify the functionality of software class methods. Netbeans IDE is also known for Rapid User Interface Development and as an Easy & Efficient Project Management tool.

* **Atom**

Atom is a [free and open-source](https://en.wikipedia.org/wiki/Free_and_open-source_software) [text](https://en.wikipedia.org/wiki/Text_editor) and [source code editor](https://en.wikipedia.org/wiki/Source_code_editor) for [OS X](https://en.wikipedia.org/wiki/OS_X), [Linux](https://en.wikipedia.org/wiki/Linux), and [Windows](https://en.wikipedia.org/wiki/Windows) with support for [plug-ins](https://en.wikipedia.org/wiki/Plug-in_(computing)) written in [Node.js](https://en.wikipedia.org/wiki/Node.js), and embedded [Git Control](https://en.wikipedia.org/wiki/Git_(software)), developed by [GitHub](https://en.wikipedia.org/wiki/GitHub). Atom is a desktop application built using web technologies. Most of the extending packages have [free software licenses](https://en.wikipedia.org/wiki/Free_software_license) and are community-built and maintained.



Atom text editor was used to view source code of web templates.

## Project Status

Currently we are in the end of Elaboration Phase. We started implementing key parts of the system. For the information system, we started building software classes from the domain class diagram.

As interface of the system plays a key role for the success of the project. Storyboarding is on progress to convince the client about the final outcome from the project.

In order to facilitate the Hotel with an online booking service, we are planning on launching a website. Hence our team players are on research to identify Web site templates which are suitable for the hotel.

With that current state in our project, after analysing feedback from the client about the interface, we are planning on building the interface for the system simultaneously while implementing business logic.

## Problems encountered

* Difficulty in scheduling interviews with client.

As the hotel is situated in Matara and due to the busy schedule of Mr.Kumarawadu it was difficult to allocate time for meet-ups. So that we started discussions via telephone.

* Difficulty in understanding the scope of the Business.

At the beginning of the project, we had difficulties in understanding functions and requirements of the system. So that after having interviews with the client we had long discussions with project members and consulted clients again to verify problems encountered.