WOLKITE UNIVERSITY

COLLEGE OF COMPUTING AND INFORMATICS

DEPARTMENT OF INFORMATION SYSTEMS

PROJECT TITLE: WEB-BASED HOTEL MANAGEMENT SYSTEM FOR SORESA HOTEL

BY

Group Member	**ID**
1. SEGNI MITIKU ……	CIR/179/11
2. DIBORA ASEFA …….	CIR/427/1
3. ABDI SEMALIGN ……..……	CIR/102/11

ADVISOR: Mr.Musa

Wolkite University, Wolkite, Ethiopia

March /2014 E.C

WOLKITE UNIVERSITY

COLLEGE OF COMPUTING AND INFORMATICS

DEPARTMENT OF INFORMATION SYSTEMS

PROJECT TITLE: WEB-BASED HOTEL MANAGEMENT SYSTEM FOR SORESA HOTEL

SUBMITTED TO THE DEPARTMENT OF INFORMATION SYSTEMS

IN PARTIAL FULFILLMENT FOR THE REQUIREMENT FOR

THE DEGREE OF BACHELOR OF SCIENCE IN INFORMATION SYSTEMS

BY

Group Member	**ID**
1. SEGNI MITIKU …..……	CIR/179/11
2. DIBORA ASEFA ……..…….	CIR/427/1
3. ABDI SEMALIGN ……..……	CIR/102/11

PROJECT ADVISOR: MR.MUSA

Wolkite University, Wolkite, Ethiopia

March 3/2014 E.C

DECLARATION

This is to declare that this project work is done under the guidance of Mr.Musa. And having the Title Web-based Hotel management system for Soresa Hotel for Wolkite University is the sole contribution of Abdi Semalign, Dibora Asefa, and Segni Mitiku.

Date: _____

Group Members:

Full name	Signature
1. Abdi Semalign	_____
2. Dibora Asefa	_____
3. Segni Mitiku	_____

APPROVAL FORM

This is to confirm that the project report entitled Web-based Hotel management system for Soresa Hotel. Submitted to Wolkite University, College of Computing and Informatics Department of Information System By: Segni Mitiku, Dibora Asefa, Abdi Semalign is approved for submission.

Advisor Name	Date	Signature
Department Head Name	Date	Signature
Examiner 1 Name	Date	Signature
Examiner 2 Name	Date	Signature
Examiner 3 Name	Date	Signature

ACKNOWLEDGEMENT

First of all, we would like to thank God who gave us the insight and patience to accomplish this work. Next, we would like to extend our thanks to our advisor Mr. Musa for his heartfelt advice and help to shape our project through review and criticism thoroughly. We would also like to thank the Information Systems Department Head Offices, Soresa Hotel workers and others who are willing for any interview and operation for what we want to form their staff to achievement of our project. Lastly, all those who have contributed to our industrial project have a special place in us.

Contents Page

LIST OF TABLES

LIST OF FIGURES

ABBREVIATION AND ACRONYMS

- ✓ CSS……Cascading style sheets
- ✓ HTML……Hypertext markup language
- ✓ MYSQL……. My structure query language
- ✓ OOA……Object-oriented analysis
- ✓ OOD…….....Object-oriented design
- ✓ PHP……..Hypertext preprocessor
- ✓ SDLC…….Software development life cycle
- ✓ UC……Use case
- ✓ GUI……Graphical user interface
- ✓ DB……..Data base
- ✓ UI……..User interface
- ✓ UML……..Unified Model Language

ABSTRACT

In our project, on "Web based Hotel Management System for Soresa Hotel**",** we have tried to show How the reservation in hotels is managed. This has been achieved by dividing the project into various modules. Customer is proved with different services like checking in, checking out, booking, and editing entries or payment. If the customer wants he/she can cancel his/her booking. Inquiry about any customer or employee can be made either by customer Id or customer name. Inquiry about rooms available can also be made. It will generate reports for customers, employees (working in the hotel), and Bill for the customers is generated when the customer will check out from the hotel. Also, we used an Iterative development life cycle to develop our project and ASP.NET in order to make the system secure, because of a web-based system.

CHAPTER ONE

1. Introduction

A Hotel is a place where services like food, bed, hall services are given to the customer and users. And it needs its own management process which deals with different actors. Our project on **"Hotel management system"** gives an idea about the management in hotels. The package gives information regarding the check-in or check-out facilities of the customer. It gives detail of the customer, the time of arrival and departure of the customer. The packages also provide the facility of searching the customer or employee working in that particular hotel by name/by the provided Id. It gives a detailed report of the customer and the room occupied by him/her. This system is very useful for departments to manage their activities. The status of the room is updated only after 12 hours depending upon the time period they have taken. They were facing problems in providing rooms to customers. In our project, we will solve this problem, and at any moment of time the status of the room that whether it is vacant or occupied is shown correctly. Developing this system has much scope. Our project provides various services to customers like a booking (advance/current), cancellation of advance booking, enquiring about any customer (by name/customer id), or room availability.

.

1.1 Background

Soresa Hotel is a well-established hotel found at a convenient location in the center of the city Wolkite. It was established in April 2002 E.C. with a few staff members. Currently, the hotel has around 50 staff members. The hotel is 13 kilometers (km) away from Wolkite main campus. Currently, the hotel has 70 bedrooms, which are four types (single, double, triple, and family) with different prices. All the rooms have a shower (both hot and cold), toilet, and mini-fridge.

1.2 Statement of the Problem

Sales and services are the fundamental tools in any business organization. The profit and loss of any business depend on detailed information on sales and services made to aid in decision making and implementation, if accountability is not checked, then the business is sure to collapse, as a result in any retail and hospitality business there is a need for a system that gives feedback to the management to aid decisions, this is where computerized hotel management system comes handy. We have too many problems associated with the manual system used which include:

- ✓ It takes more time to reserve a room. So the performance of the current system is slow.
- ✓ The current manual system uses paperwork and direct human language communication by mouth to manage the hotel. This delays information transmission in the hotel.
- ✓ Difficulty in handling data accurately, security of data, data, lost and viewing by an unauthorized person.
- ✓ Retrieving information like reports and queries is time-consuming and almost impossible practicably if time is considered.
- ✓ Searching records of individual guests or customers takes time. (Update, Delete, and Edit), these types of methods are not accessible using the manual method.
- ✓ Slow speed of operation.
- ✓ Conflicting information.
- ✓ Data redundancy.
- ✓ Day to day activities of the existing system is very costly.

- ✓ Poor communication**:-**due to poor communication between the departments, guests are often served with services they didn't order.

1.3 Objectives

The project has both general and specific objectives.

1.3.1. General objective

The main objective of this project is to design and develop a web-based hotel management system for Soresa hotel.

1.3.2. Specific objective

The specific objectives of the project are mentioned as follow:-

- ✓ Reviewing how the current system works and operates.
- ✓ Investigating how the existing system is operating.
- ✓ Design a new system that can overcome the problem of the current system ✓ Develop and implement the new system.

1.4 Feasibility study

1.4.1 Technical feasibility

In this, one has to test whether the system can be developed using existing technology or not. It is planned to implement the proposed system usually Php, MySQL server, micro media, dream weaver. It is evident that necessary hardware and software are available for the development and representation of the website. The solution is technically feasible.

1.4.2. Operational feasibility

It is a standard that ensures the entire operability without shifting completing and innovation among users to the benefit of the public both in tercostcoast and service quality. The proposed system is acceptable to the use So that the website is operationally feasible.

1.4.3. Economic feasibility

As a part of this, the costs and benefits associated with the proposed system are compared and the project is economically feasible only if tangible and intangible benefits outweigh the cost. The cost for the proposed hotel management system is outweighing the cost and effort involved in maintaining the register's books, files, and generations of various reports. The system also reduces the administrative and technical staff to do various jobs that single software can do so, this system s economically feasible.

1.4.4. Legal feasibility

Legal feasibility determines whether the proposed system conflicts with legal requirements.

Example The data protection act. Some legal advisors will do it**.**

1.4.5. Schedule feasibility

Schedule feasibility determines whether the proposed system will be completed at the given time or not. Whatever the scarcity of time given for the project by the internal motivation and potential of the team member of the project, we surely expect the project will be completed on time.

1.5 Scopes and Limitations of the project

1.5.1 Scope of the project

Although design concepts, outputs, and other components of the Project can be used for a different purpose, the implementation of this project is on a web-based hotel management system. The system will cover; booking, drinking, accommodation, meals, and accounts details. Moreover, special services such as laundry, ironing, and room service will be automated by the system also, not to forget the additional facilities information that will be efficiently handled by the system. In addition, the new system should provide the following services.

- ✓ Search Rooms information
- ✓ Add Room, Updates room, Delete room
- ✓ Schedule
- ✓ Search customer information by their name or Id

- ✓ Registration
- ✓ Update, Add /Delete employee
- ✓ Generate report
- ✓ Backup and Restore
- ✓ Add Booking
- ✓ View Booking
- ✓ Feedback
- ✓ Schedule
- ✓ Changing profile picture
- ✓ Forgot password using Recovery Key
- ✓ Use map
- ✓ Online payment

The scope of the project mainly deals with providing online book reservations, canceling res reservations, updating reservation information, and human resource parts like adding employees, terminating employees and updating employees and etc.

1.5.2 Limitation of the project

The main limitations of the project are mentioned as follow:-

- ✓ The people with eye impairment can not use the system.
- ✓ This project only function when there is internet availability.

1.6 Significance of the project

The advantages that are using this web-based hotel management system are so many as compared to manual labor.

- ✓ The system is efficient in improving time management. Time is very vital.
- ✓ The system enables easy and fast access to the guest files.
- ✓ The system provides better data management facilities.
- ✓ The system is important in monitoring the premises of a lodge.
- ✓ The system reduces workload.
- ✓ The operation is computerized hence able to perform some of the tasks.
- ✓ The usage of a program makes work easier.
- ✓ It also reduces the number of staff members that can be required to perform a task..
- ✓ The use of computers ensures that the making of errors is reduced.
- ✓ Easy update of the guest records.
- ✓ High customer service standards attract more guests to the hotel.
- ✓ Greatly reduce paper use at the hotel.

1.7 Beneficiary of the project

This project will benefit the management of the hotel. It reduces the workload and will save time for them. Also, it will benefit the user of the system since it is system dependent it will increase the quality of the service they get, and also it will satisfy the user of the system by providing good service.

To the Developer Team

- ✓ It has initiated has to read and refer a lot in order to get knowledge how to design and develop a new system.
- ✓ It also experienced has in order to solve any problem around .its only matter of interest and initiations.

1.8 Methodology

1.8.1. Data collection

There is the method of data collection for both functional and non-non-functionalquirements.

Interview

An interview is conservation or questioning, for the purpose of eliciting information for the publication of the available statement so elicited. To get the basic information and background information about the existing system the team member has interviewed the manager and some customers about the service that is to become given to them, and the problem associated with that environment. Interviews can be conducted face-to-face or by telephone. They can range from indepth, semi-structured to unstructured depending on the information being sought.

Observation

Observation is one of the ways to collect data that has been necessary for our web-based system project for the hotel. We have observed how the general property of the hotel accomplishes their task by getting in high-quality beds to their customers. These have brought many tangible requirements for us.

1.8.2. System analysis and design

Among the different methodologies available we plan to use the object-oriented design methodology for the development of our system

Object-Oriented Analysis (OOA): During the phase, we will look at the problem domain, with the aim of producing a conceptual model of the information that exists in the area which will be analyzed. And this model the function of the system (use case modeling), the objects, and also the relationship between them, and finally model the behavior of the objects.

Object-Oriented Design (OOD): During this phase model object interaction and behaviors that support use case scenario and finally update the object model to reflect the implementation environment. And also transform the conceptual model produced in object-oriented analysis to

take account of the constraints imposed on our system format, so that we will use this phase to refine the use case model to reflect the implementation environment.

1.8.3. System development model

Iterative model

The iterative model is a particular implementation of software development of life cycle(SDLC) that focuses on an initial, simplified implementation, which then progressively gains more complexity and a broader feature set until the final system is complete. The iterative model is the best thought of the cyclical process. After an initial planning phase, a small handful of stateside repeated over and over, with each completion of the cycle incrementally improving and iterating on the software. Enhancements can quickly be recognized and implemented throughout each iteration, allowing the next iteration to be at least marginally better than the last.

1.8.4 System testing methodology

As we used the iterative system development model we have to use all (unit, integration, system, and acceptance) testing methodology.

Unit testing

In this level of the testing process, we will test the different sub-procedures or functions.

- ✓ Check whether the return type of the functions is correct.
- ✓ Check how the subprocedures or functions are called correctly.
- ✓ Check if the correct output is produced for different inputs.
- ✓ Check the efficiency of the code concerning the memory and CPU time.

Integration testing

In this level of testing, we will examine how the different procedures work together to achieve the goal of the subsystem.

System testing

In this level of the testing process, we will examine how good the subsystems of the whole

Management system work together to achieve the desired goal. This is tested by the developer and user.

Security testing: - involves the testing of software, in order to identify and flow and gaps from a security and vulnerability point of view. The main aspects of security testing are:-

- ✓ Authentication
- ✓ Availability
- ✓ Input checking and validation
- ✓ Software data is secure and etc.

Portability testing:-Portability testing of software with intended that it should be reusable and can be removed from another software as well. Strategies that can be used for portability testing can be considered as one of the subparts of the system.

Acceptance Testing

In this level testing process, the system will be tested by the user who checks how we are and what the system looks like by their testing mechanism. This is the final testing in the system development.

1.8.5 Development tools and technologies

Front end Tools

- ✓ **J query: -** is a JavaScript library that simplifies JavaScript programming.

  Jquery is a JavaScript library designed to simplify HTML and manipulation, as well as event handling, CSS animation.

- ✓ **Bootstrap**: - is the most popular HTML and CSS JavaScript framework for developing responsive, mobile-first approaches.

Is the most popular **CSS framework** for developing responsive and mobile-first websites.

Back end Tools

- ✓ **MYSQ: -** MySQL is an open-source relational database management system. It is based on the structured query language (SQL), which is used for adding, removing, and modifying information in the database. Standard SQL commands, such as ADD, DROP, INSERT, and UPDATE used in MySQL. MySQL is used for a variety of applications but it is usually used on Web servers. A website that uses MySQL may include Web pages that access information from a database.
- ✓ **PHP**: - is a server scripting languag, and a powerful tool for making dynamic and interactive web pages.

  PHP is a great option for many reasons, so if you're interviewing a company throwing around the PHP slag, here are some reasons why the language may be right for you or your project:

1.9 Document organization

Chapter one contains the problem of statements, objectives, methodology, the significance of the project, and others. Chapter two contains abour existing system of drawback, function, and user. Chapter three contains a proposed system of functional and no functional requirements. Chapter four is all about system models such as class diagram, sequence diagram, statechart, use case diagram, use case description, and others. Chapter five contains the solution part of which is the design of the system to simplify the ask of problems such as subsystem decomposition, hardware/software mapping, packages, and some pseudo-codes.

CHAPTER TWO

2. Description of the Existing system

2.1 Introduction of Existing system

Analysis involved a defiled study of the current system leading to the specification of new system. The existing system happens to be a non-computerized operating system were all operations are done manually by the receptionist.

Analyzing the existing system ensures that new software that would deliver efficiency, effectiveness, accuracy, reliability, and accuracy in the management of a hotel's affairs will be designed and implemented. However, the aim of a comprehensive and thorough analysis of the operation of an existing system is to get the required data that will assist tremendously in the design of a new system.

2.2 Users of Existing System

- ✓ Customers
- ✓ Guests
- ✓ Managers
- ✓ Receptionist

2.3 Major functions of the Existing System

The major function of existing system:-

- ✓ Provide room reservation service to the user
- ✓ Provide food and drink service to the user
- ✓ Provide internet access
- ✓ Generate report

2.4 Forms and other documentation of the Existing Systems

One of the existing system in our hotel is customers are seen the price of meal and drink like this picture.

Figure 2. 1 Menu of current system

And the other one is if guests are coming from other country first it must be complete thisform.

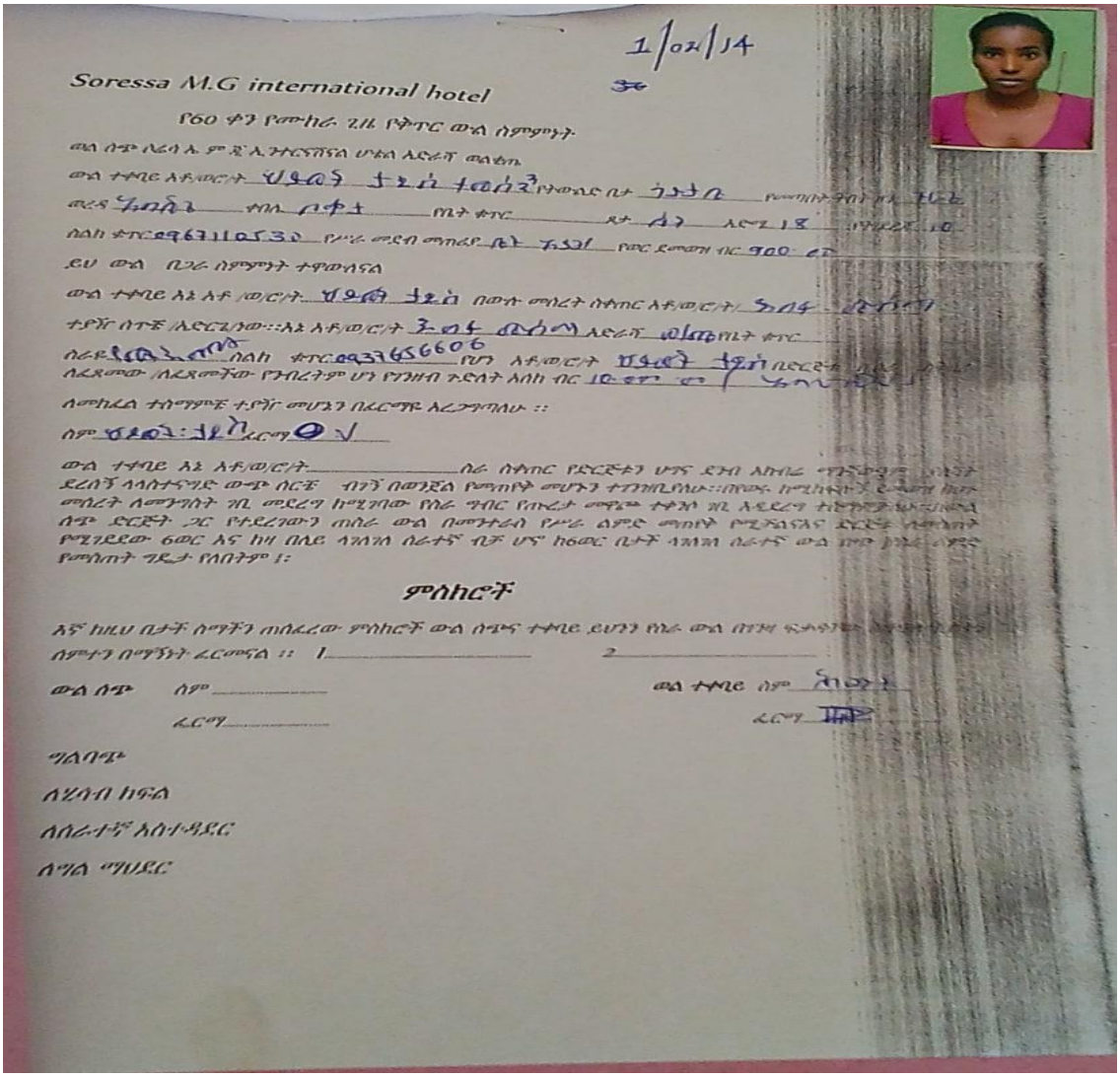

Figure 2. 2 Employee registration form of current system

2.5 Drawbacks of the existing system

It is important to keep pace with time with the increasing competition in the market and to stand on the present environment of the modern world. The various drawbacks of manually handling the operations in the hotel are:

- ✓ Slow speed of operation.
- ✓ Conflicting information
- ✓ Data redundancy
- ✓ Inability to analyze past data
- ✓ Inaccurate information
- ✓ Insatiable customers
- ✓ Day to day activities of the existing system is very costly.

It is well noted that an automated system of managing and reservation of rooms or other affairs of a hotel is more efficient and reliable then a manual system that performs the same operations. This in turn gave rise to a proffered solution to the underlying problems associated with managing the affairs of a hotel manually.

2.6 Business rule of the Existing system

Business rule states what type of rules and regulations are used by the organization to speed up the tasks which are performed within the hotel.

- The system may not accept customer, employee receptionist manager and system administrator without their username and password.
- The Customer table provides personal information for every customer that reserve a room with the hotel attribute includes first name, last name, ID, phone number and address.
- The customer won’t login into the system at the first time, i.e they only allow to registe

CHAPTER THREE

3. PROPOSED SYSTEM

Web based hotel management system is a system that provide management way with modern system, which improved the current problems of the current system by developing the new developed system.

The proposed system is capable of providing high security of data, the capability of all information in easy way of recording and accessing items information by well organized user friendly interface. Generally the proposed system improved the performance of the existing system and reduce this problems, give data security poor quality service delivery, inconsistency, wastage of time and other materials.

3.1 Functional Requirements

Functional requirements define the fundamental action that system must perform Functional requirements describe the relationship between the system and the user. Here the relationship means the direct or indirect interactions between the user and the system. Hence the system has the following stakeholders with different requirements [1].

The system allows the administrator to create account, update and deactivate

- ✓ The system allows the manager to generate a report
- ✓ The system allows the manager to see comments
- ✓ The system allows the manager to add, terminate employees
- ✓ The system allows the receptionist to add, update and delete room
- ✓ The system allows the receptionist to see for available room
- ✓ The system allows the receptionist to generate report
- ✓ The system allows the the customer to reserve room

- ✓ The system allows the customer to see for available room
- ✓ The system allows the customer to cancel reservation
- ✓ The system allows the customer to pay online
- ✓ The system allows the customer to write feedback for the hotel

3.2 Nonfunctional requirement

3.2.1 User interface and human factors

As the system is developed that follow good user interface guideline it provides an interface that is easy to use for the user. And user that means financial, student or any customer should only to have computer and browser to connect to the internet. So, there is no need of computer professions to use the system [1].

3.2.2 Hardware consideration

As the system is web application the hardware that should consider are server that enable user to access system through internet. And client computer or personal computer like laptop. That user use to access the system. The system is compatible to any computer that enables user to access internet.

3.2.3 Security issues

The system can allow only authorized user to entering and also able to encrypt user's password.

System also sends user password to his email if user changes his password.

3.2.4 Performance consideration

This system operates its function relatively in small amount of time which is less than one second and can be accessed and used by patients, doctors and other persons who can access the internet to get the information (they must have or use desktop, laptop and others, that have the application to able to access the internet (For example: Mozilla Firefox, Internet explorer, Chrome and so on)) at the same time or concurrently.

3.2.5 Error handling and validation

When a user interacts with the system errors may occur. To control this kind of inaccuracies the system will generate different user-friendly messages. To do this, most of the system execution buttons will be controlled according to the sequence which the user is expected to follow, or this can be done by generating different system responses to the input of the user. According to their work scope.

The systems can able handle exceptions that may happen while user uses the system. It handles exceptions of data duplication to save memory space error related to finance

3.2.6 Quality issues

We can say our system is reliable as we mentioned before the system will consistently perform its intended function. Unless there is an internet connection problem occurs, our system is available at any time. In the quality assessing the users will be involved by feedback mechanism in which they can give comment on the system.

3.2.7 Backup and recovery

- ✓ Make sure there is an additional server to back up the hospital data in case when the server is down.
- ✓ Use other external discs to back up data

3.2.8 Physical environment

The system is deployed or installed on the server-side script but for more feature we recommend that the system to deploy on Wolkite university server that is free from any disaster.

3.2.9 Resource issues

To use the system user should to have any browser on his computer that enables his/her to connect to the internet.

3.2.10 Documentation

In the process of interacting with this system the users and the users of system can be easly access the software using the following documentation type.

- ➢ User guides

- Documentation

CHAPTER FOUR

4. SYSTEM ANALYSIS

4.1 System Model

4.1.1 Use case model

In these sections, the interaction between the user/actor and the system in order to solve a problem should be described. It also defines the user's objective, the interactions between the system and the user, and the system's behavior required to meet these objectives. Various model elements are contained in the use-case model, such as actors, use cases, and the association between them.

4.1.1.1 Use case diagram

A use case describes a sequence of actions that provide a measurable value to an actor in other words it shows a way in which an actor interacts with the system at the same time use case describes the behavior of the system as seen from an actor's point view.

A use case describes a function provided by the system as a set of events that yield a visible result for the actors.

- In the analysis phase, they represent the functionality of the system.

No	Actor	Description
1	Admin	A person who manages the account.
2	Customers	A person who uses the system for reserve or to order food and beverage.
3	Receptionist	A person who register, reserve customer, or clients who come to reserve a room and take payment from the clients for the reservation.
4	Manager	A manager is a person who manages, plan, organize and direct overall activities in hotel services

- ✓ The actors and the use case that participate in our project are: -
- ✓ **Administrator**

The system allows the System Administrator:

- To login into the system.
- create account
- deactivate account
- View report
- Add,update,delete account
- To log out from the system.

✓ **Manager**

The system allows Manager:

- To log into the system.
- To view feedback.
- posting schedule
- generate report
- Add , Update,delete emplooye
- To view the report
- to logout from the system

✓ **Customer**

The system allows **Customer**:

- To log into the system.
- To search for available room.

- To make a reservation.
- To make payment
- To update reservation
- To see meal and drink list
- To cancel reservation.
- To see the location of the hotel.
- To give feedback to the manager.

✓ **Receptionist**

The system allows **Receptionists**:

- To login into the system.
- To search for available room
- To search customer information
- Generate report
- Check payment
- Add, update, and delete, rooms.
- Add, update, delete meal and drink
- To view customer information
- Log out from the system.

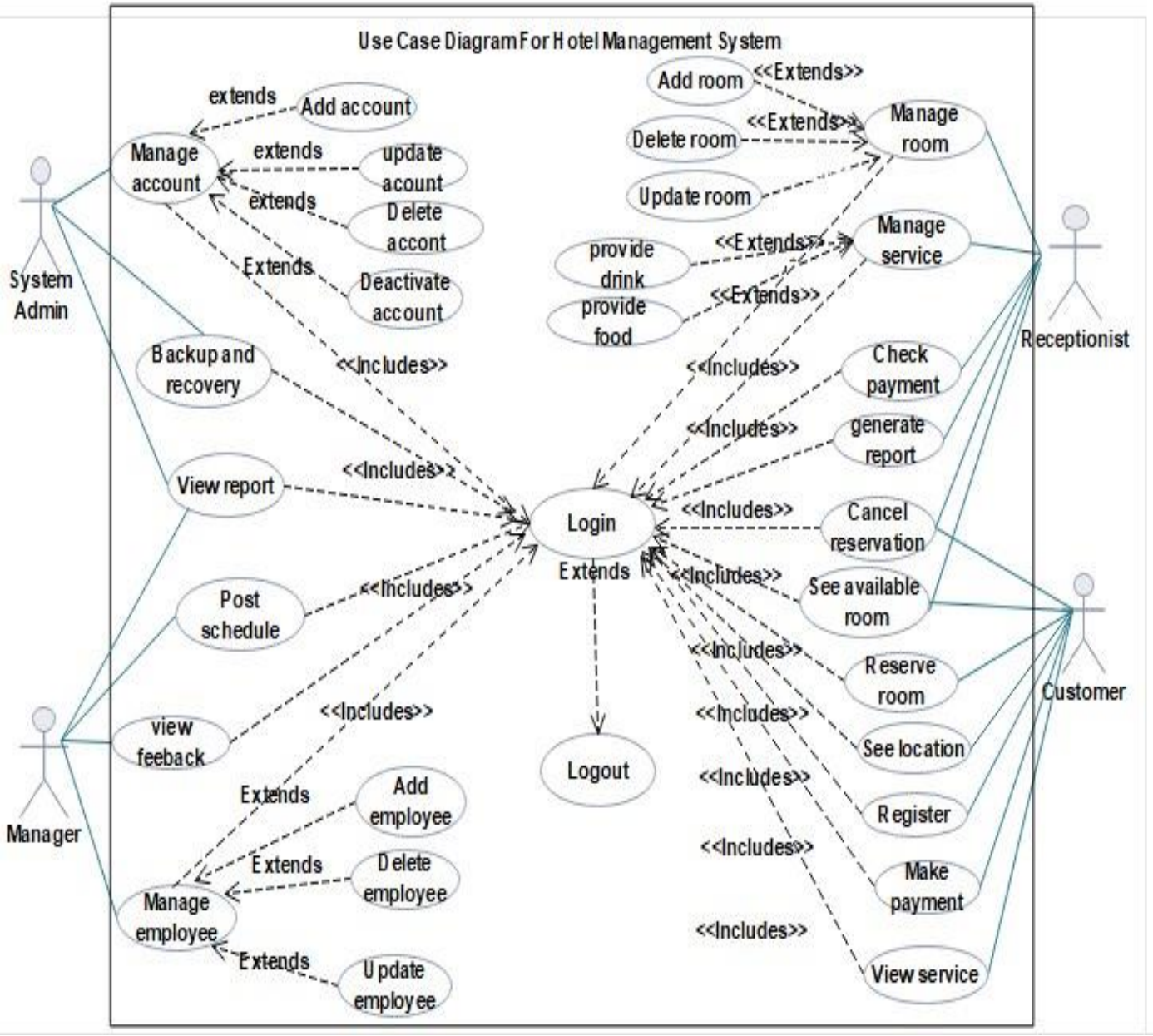

Figure 4. 1 use case diagram

4.1.1.2 Use case description

Table 4. 1 Use case description- **Search room**

Use case name	**Search room**	
Use case number	Uc1	
Actor	Customer and receptionist	
Description:	The receptionist searches for available rooms so does the customer.	
precondition	The customer should have internet access.	
Postcondition	The room information is displayed	
Basic course of action	**User action**	**System response**
	1. The user opens the 'Search available room' page. 3. The user chooses the type of room it wants to search on the form. 4. The user clicks the search button. 8. End-use case.	2. The system will display a 'search available room' form that helps the user to look for available rooms by accepting the room type from the user. 5. The system searches for an available room type that the user looking for. 6. If the type of room that the user wants is found the system displays the room's information on the result box.

An alternative course of action	7. If the type of room that the user looking for is not found go back to steps 3, 4, and, 5 of the basic course of action.

Table 4. 2 Use case description- Register

Use case name	Register	
Use case number	Uc2	
Actor	Customer	
Description:	The customer register online	
Precondition	The customer should have internet access and register for the first time	
Postcondition	The customer registered successfully if he/she fill the form	
Basic course of action	**User action**	**System response**

	1. The customer opens the home page. 2. The customer opens the Register membership page. 3. The customer files all the fields required 5. The Customer Clicks the register button 9. End-use case	4. The system will display a registration form that contains the following -First name -Last name -Address -Sex -Age -User Name -Password 6. The system checks all the form fields have been filled correctly. 7. If the form is filled correctly the system
		Display a successfully registered message.

An alternative course of action	8. If the form is not filled correctly go back to steps 3, 4, 5, and 6 of the basic course of action.

Table 4. 3 Use case description- Leave comment

Use case name	Leave comment	
Use case number	Uc3	
Actor	Customer	
Description:	The customer leaves a comment on the hotel's website for the manager as feedback.	
Precondition	The customer should visit the hotel first.	
Postcondition		
Basic course of action	**User action**	**System response**

	1. The customer opens the hotel's website. 2. The customer opens the comment page. 4. The user fills in the comment on the form. 5. The user clicks the 'send' button. 9. End-use case.	3. The system will display a 'comment' page that helps the user to leave his/her comment for the manager. 6. The system checks That the user fills all the required fields. 7. If all the required fields are filled properly by the user, the comment Will be sent to the manager.
An alternative course of action	8. If the comment form didn't fill properly go to steps 3, 4,5 and 6 of the basic course of action.	

Table 4. 4 Use case description- Search customer information

Use case name	Search customer information
Use case number	Uc4
Actor	Receptionist
Description:	The receptionist searches the customer's information that reserves a room in the hotel.

Precondition	A customer should reserve a room with the receptionist.	
Basic course of action	**User action**	**System response**
	1. The receptionist login to the system. 2. The receptionist opens the 'search customer' page. 4. The receptionist enters the customers on the form. 5. The receptionist clicks the search button. 9. The receptionist clicks on the result information and views the customer's information. 10. End-use case.	3. The system will display a 'Search customer' form that accepts the customer information to search. 6. The system checks for the customer's information on the database. 7. If the customer's information that the receptionist looking for is found the system displays the customer's information on the result box.
An alternative course of action	8. If the customer's information is not found go back to steps 3, 4, 5, and 6 of the basic course of action.	

Table 4. 5 Use case description- View comment

Use case name	View comment	
Use case number	Uc5	
Actor	Manager	
Description:	The manager will view comments sent from customers and receptionists.	
Precondition	There should be a comment sent from the customer and receptionist.	
Postcondition	View incoming comments	
Basic course of action	**User action**	**System response**
	1. The manager logs into the system. 2. The manager opens the 'view comments' page. 4. The manager selects and reads comments. 5. End-use case.	3. The system will display a page containing the list of comments sent from customers.

Table 4. 6 Use case description- Login

Use case name	Login	
Use case number	Uc6	
Actor	Manager, Receptionist, Customer and System administrator	
Description:	The user enters an authorized username and password in order to access the system.	
Precondition	The user should have a username and password.	
Basic course of action	**User action**	**System response**
	1. The user opens the 'login' page. 3. The user enters the username and password on the form. 4. The user clicks the login button. 8. End-use case.	2. The system will display a 'login' form that accepts the username and password from the user. 5. The system searches the username from the database and matches it with the password. 6. If correct the system opens the authorized page.

An alternative course of action	7. If the username and password are incorrect go to steps 2, 3, 4, and 5 of the basic course of action.	

Table 4. 7 Use case description- Logout

Use case name	Logout
Use case number	Uc7
Actor	Manager, Receptionist, customers, and System administrator
Description:	When the user logouts when he/she wants to exit from the system.
Precondition	The user should log in first.

Postcondition	Login button opened	
Basic course of action	**User action**	**System response**
	1. The user clicks the logout button. 3. End-use case.	2. The system will display the login form for the user.

Table 4. 8 Use case description- Add room

Use case name	Add room	
Use case number	Uc8	
Actor	Receptionist	
Description:	The receptionist will add a new room with new attributes to the system.	
Basic course of action	**User action**	**System response**

	1. The receptionist will log in to the system. 2. The receptionist will open the 'add room' page. 3. The receptionist will fill out the form. 4. End of use case.	3. The system will display the 'add room' form with fields such as: - Room name. - Room type. -Room price. 5. The system will check all fields of the form are filled or not.
		6. If all fields are filled correctly show successful message.
An alternative course of action	7. If the form is not filled correctly go to steps 3, 4, and 5 of the basic course of action.	

Table 4. 9 Use case description- Delete room

Use case name	Delete room
Use case number	Uc9
Actor	Receptionist

Description:	The receptionist will delete the room from the database.	
Basic course of action	**User action**	**System response**
	1. The receptionist will log in to the system. 2. The receptionist will open the 'delete room' page. 4. The receptionist will enter the information of the room to be deleted. 7. The receptionist selects the room from the result box and clicks the delete button.	3. The system will display the 'Delete room' form that helps the system admin choose the room to be deleted. 5. The system searches for the room that the system admin wants to delete. 5. If the room that the system admin wants to delete is found the system displays the room's information on the result box.
	10. End-use case.	8. The system will delete the room from the room's table. 9. The system will display a successful message.

Table 4. 10 Use case description- Update room

Use case name	Update room	
Use case number	Uc10	
Actor	Receptionist	
Description:	The receptionist updates the room's information.	
Basic course of action	**User action**	**System response**
	1. The receptionist will log into the system. 2. The receptionist will open the 'update room' page. 8. The receptionist will enter the information of the room to be updated.	3. The system will display the 'Update room' form that helps the system admin to choose the room to be updated. 5. The system searches for the room that the system admin wants to update. 6. If the room that the system admin wants to update is found

	8. The receptionist selects the room from the result box and clicks the update information button. 10. The receptionist will change the room's information. 11. The receptionist will click the save button. 14. End-use case.	The system displays the room's information on the result box. 9. The system will display a form that contains the selected rooms' information. 12. If all form fields are filled correctly the system will display a successful message.
An alternative course of action	7. If the room is not found go back to steps 3, 4, and 5 of the basic course of action. 13. If the form is not filled correctly go to steps 9, 10, and 11 of the basic course of action.	

Table 4. 11 Use case description- Add employee

Use case name	Add employee	
Use case number	Uc11	
Actor	Manager	
Description:	The manager will add a new room with new attributes to the system.	
Basic course of action	**User action**	**System response**

	1. The manager will log in to the system. 2. The manager will open the 'add employees' page. 4. The manager will fill out the form. 8. End of use case.	3. The system will display the 'add employees' form with fields such as: - Employee name. - Employee ID. -Employee salary. -Employee address. 5. The system will check all fields of the form are filled or not. 6. If all fields are filled correctly show a successful message.
An alternative course of action	7. If the form is not filled correctly go to steps 3, 4, and 5 of the basic course of action.	

Table 4. 12 Use case description- Update employees

Use case name	Update employee
Use case number	Uc12
Actor	Manager

Description:	The manager updates the employee's information.	
Basic course of action	**User action**	**System response**
	1. The manager will log in to the system. 2. The manager will open the 'update employee' page. 4. The manager will enter the information of the employee to be updated. 8. The manager selects the employee from the result box and clicks the update information button. 10. The manager will change the employee's information. 11. The manager will click the save button. 14. End-use case.	3. The system will display the 'Update employee' form that helps the system admin to choose the employee to be updated. 5. The system searches for the employee that the system admin wants to update. 6. If the employee that the system admin wants to update is found the system displays the employee's information on the result box. 9. The system will display a form that contains the selected employee's information. 12. If all form fields are filled correctly the system will display a successful message.
An alternative course of action	7. If the employee is not found go back to steps 3, 4, and 5 of the basic course of action. 13. If the form is not filled correctly go to steps 9, 10, and 11 of the basic course of action.	

4.1.1.3 Use case scenario

In this method, the actions that are required to enable or abandon a goal are represented. There are many paths that are discussed in these lessons. From those paths, a use case scenario is a single path through the use case. It describes all possible scenarios involving the described use case description. It shows the process/steps while the interaction b/n system & user.

Scenario name: login

Actor: all user

All the actors want to login into the system. They should enter the home page and then click the login button. After that, the user clicks the link, the system displays the login form. After that, the user fills the form with the appropriate username and password. Then the system checks the input data and if the user fills in the correct information the system allows the user to enter into the system and perform their activity. If the user enters invalid input the system generates the error message else user page is displayed next to perform his authenticated operation and logout.

Scenario name: add employee

Participating Actor: manager

If the user wants to add an employee first, he/she must log in to the system. After login into the system, the user browses the home page. Next to these, the user clicks the 'add employee' link from the dashboard. Then the system displays the add employee form that the user wants to fill. After that, the user fills the field and clicks the add employee button. If all the entry data are correct the data are stored in the database else the system display error message text. Finally, the user leaves the page.

Scenario name: create an account

Participating actor: administrator

If the administrator wants to create an account for a different user, first he/she must be login into the system. After that, he browses the home page and click the manage account button and

immediately click the create account link from the drop-down menu. Next to these the system displays the create account form field. Then the user fills out the form and clicks the create account button. After these, the system checks are the account was created before or not. If not created and the entry is correct the system displays the account is created message else the system displays an error message and the user tries again.

Scenario name: delete the account

Participating actor: administrator

If the administrator wants to delete the user account, he/she must be login into the system. After logging into the system, the user browses the home page and then selects/clicks the manage account button. After these, the user clicks the delete account link from the menu. With respect to these, the system displays the create account form and the user searches the account he/she want to delete by entering some key. After these, the system displays the search account if exists. Then the user clicks deletes account button to delete the account from the database. Next to these, the user views the account from the database, and if not exist the account is successfully deleted else try again and leave the page.

Scenario name: Reservation or booking

Participant actor Customer

If the customer wants to reserve a room first she/he would have a CBE account/hello cash account and run the system then the system displays the home page next customer open reservation page then click on the book now button. After that system displays the booking form the customer fill all required form including check-in and check-out date, then the system calculates and display the total amount she will stay. Then after the customer pays the calculated price through CBE/hello cash or transfer money to the hotel.

4.2 Object Model

4.2.1 Class diagram

Class modeling is used to model the static part of the systems. The design class model will reflect the wide variety of technology decisions you make. The main action shown on the class diagram is class, object, operation, and associations ( ). This model is derived from the analysis class diagram. The only difference is system class diagram includes an access modifier. There are two main access modifiers in our project [2].

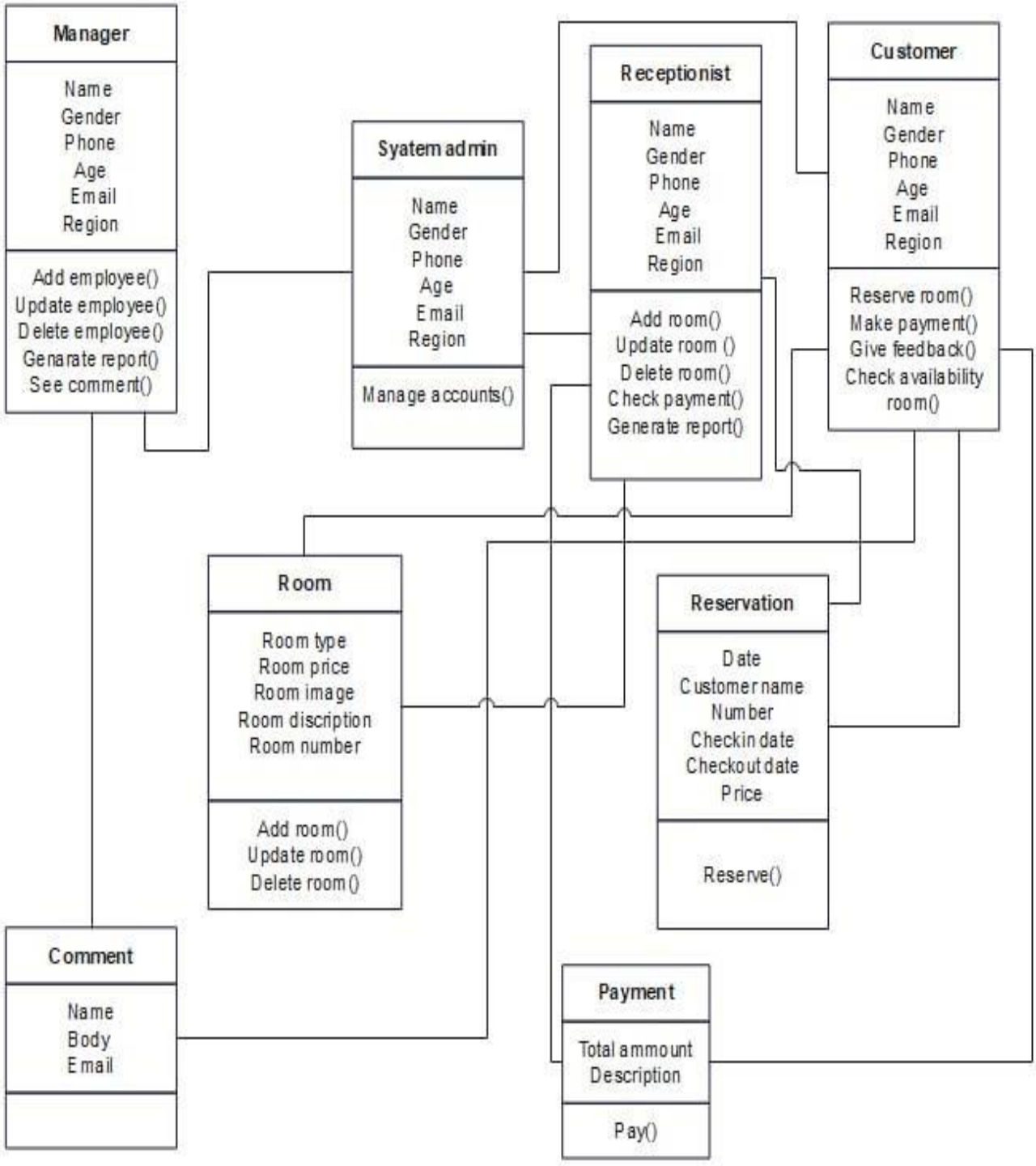

Figure 4. 2 Analysis Class Diagram

4.2.2 Data dictionary

As the system is able to save/store information about real-world entities into different data files we present a data dictionary of the system that describes which data is stored in which data file. The data dictionary is a description of the database table of the system such as functionality, data types, data size, contents item, and table relations.

It includes

- ✓ Name of the tables in the database
- ✓ Constraints of a table i.e., keys, relationships, etc.
- ✓ Columns of the tables that related to each other
- ✓ Owner of the table
- ✓ Last accessed information of the object
- ✓ Last updated information of the object.

Table 4. 13 Data dictionary for account details

Name: account details				
Attribute name	Datatype	Size	Domain	Constraint
user_id	Varchar	15	Letter, number, character	Primary key
First_name	Varchar	15	Letter	Not Null
Last_name	Varchar	15	Letter	Not Null
Age	Text	15	Letter, number	Not Null
UserName	Varchar	30	Letter, number, character	Unique
Password	Varchar	15	Letter, number, character	Unique

Date	Varchar	10	Letter, number, character	Not Null
Gender	Varchar	6	Letter	Not Null
Role	varchar	20	Letter	Not Null

Table 4. 14 Data dictionary for login details

Name: Login details				
Attribute name	Datatype	Size	Domain	Constraint
userName	Varchar	30	Letter, number, character	Not Null
Password	Varchar	15	Letter, number, character	Not Null
User type	Varchar	20	Letter	Not Null

Table 4. 15 Data dictionary Administrator Details

Admin details				
Name	Datatype	Size	Domain	Constraint
FirstName	Varchar	15	Letter	Not Null
LastName	Varchar	15	Letter	Not Null
Id	Varchar	15	Letter, number, character	Not Null
Address	Varchar	15	Letter and number	Not Null
Phone_no	Varchar	13	Character and number	Not Null
gender	Varchar	6	Letter	Not Null

Birthday	Varchar	15	Character and number	Not Null
User_name	Varchar	15	Number, letter, character	Not Null
Password	Varchar	15	Number, character, letter	Not Null

Table 4. 16 Data dictionary Manager Details

Manager details				
Name	Datatype	Size	Domain	Constraint
FirstName	Varchar	15	Letter	Not Null
LastName	Varchar	15	Letter	Not Null
Id	Varchar	15	Letter, number, character	Not Null
Address	Varchar	15	Letter and number	Not Null
Phone_no	Varchar	13	Character and number	Not Null
gender	Varchar	6	Letter	Not Null
Birthday	Varchar	15	Character and number	Not Null
User_name	Varchar	15	Number, letter, character	Not Null
Password	Varchar	15	Number, character, letter	Not Null

Table 4. 17 Data dictionary Receptionist details

Receptionist details				
Name	Datatype	Size	Domain	Constraint
FirstName	Varchar	15	Letter	Not Null
LastName	Varchar	15	Letter	Not Null
Id	Varchar	15	Letter, number, character	Not Null
Address	Varchar	15	Letter and number	Not Null
Phone_no	Varchar	13	Character and number	Not Null
Gender	Varchar	6	Letter	Not Null
Birthday	Varchar	15	Character and number	Not Null
User_name	Varchar	15	Number, letter, character	Not Null
Password	Varchar	15	Number, character, letter	Not Null

Table 4. 18 Data dictionary Customers Details

Name: Customers details				
Name	Datatype	Size	Domain	Constraint
FirstName	Varchar	15	Letter	Not Null
LastName	Varchar	15	Letter	Not Null
Id	Varchar	15	Letter, number, character	Not Null
Address	Varchar	15	Letter and number	Not Null

Phone_no	Varchar	13	Character and number	Not Null
Gender	Varchar	6	Letter	Not Null
Image	Img			Not Null
Birthday	Varchar	15	Character and number	Not Null
User_name	Varchar	15	Number, letter, character	Not Null
Password	Varchar	15	Number, character, letter	Not Null

Table 4. 19 Data dictionary Room details

Name: room details				
Name	Datatype	Size	Domain	Constraint
Room_no	Number	15	Number	Primary key
Room type	Varchar	15	Letter	Not Null
Blook_no	Number	15	Number	Not Null

Table 4. 20 Data dictionary Customers room detail

Name: Customers detail				
Name	Datatype	Size	Domain	Constraint
Room_no	Varchar	15	Letter, character, number	Foreign key

Room type	Varchar	15	Letter, character, number	Not Null
Patient_id	Varchar	15	Letter, character, number	Foreign key
Block_no	Varchar	15	Letter, character, number	Foreign key
Bed_no	Varchar	15	Letter, character, number	primary key

Table 4. 21 Data dictionary Food order

Name: Food order				
Name	Datatype	Size	Domain	Constraint
Order_id	Varchar	15	Letter, character, number	Primary key
food_id	Varchar	15	Letter, character, number	Foreign key
food_name	Varchar	20	Letter	Not Null
Receptionist ,id	Varchar	15	Letter, character, number	Foreign key
Unit	Varchar	20	Letter, character, number	Not Null
Quantity	Number	15	Number	Not Null
Req_Date	Varchar	10	Letter, character, number	Not Null

Table 4. 22 Data dictionary Feedback details

Name: feedback details				
Name	Datatype	Size	Domain	Constraint
Feedback_id	Varchar	15	Letter, character, number	Primary key
UserName	Varchar	20	Letter, character, number	Not Null
Message	Varchar	200	Letter, number	Not Null

4.3 Dynamic Model

In these section behaviors of the object model, in terms of sequence, activity, and state chart diagrams.

4.3.1 Sequence diagram

A sequence diagram describes interactions among classes in terms of an exchange of messages over time.

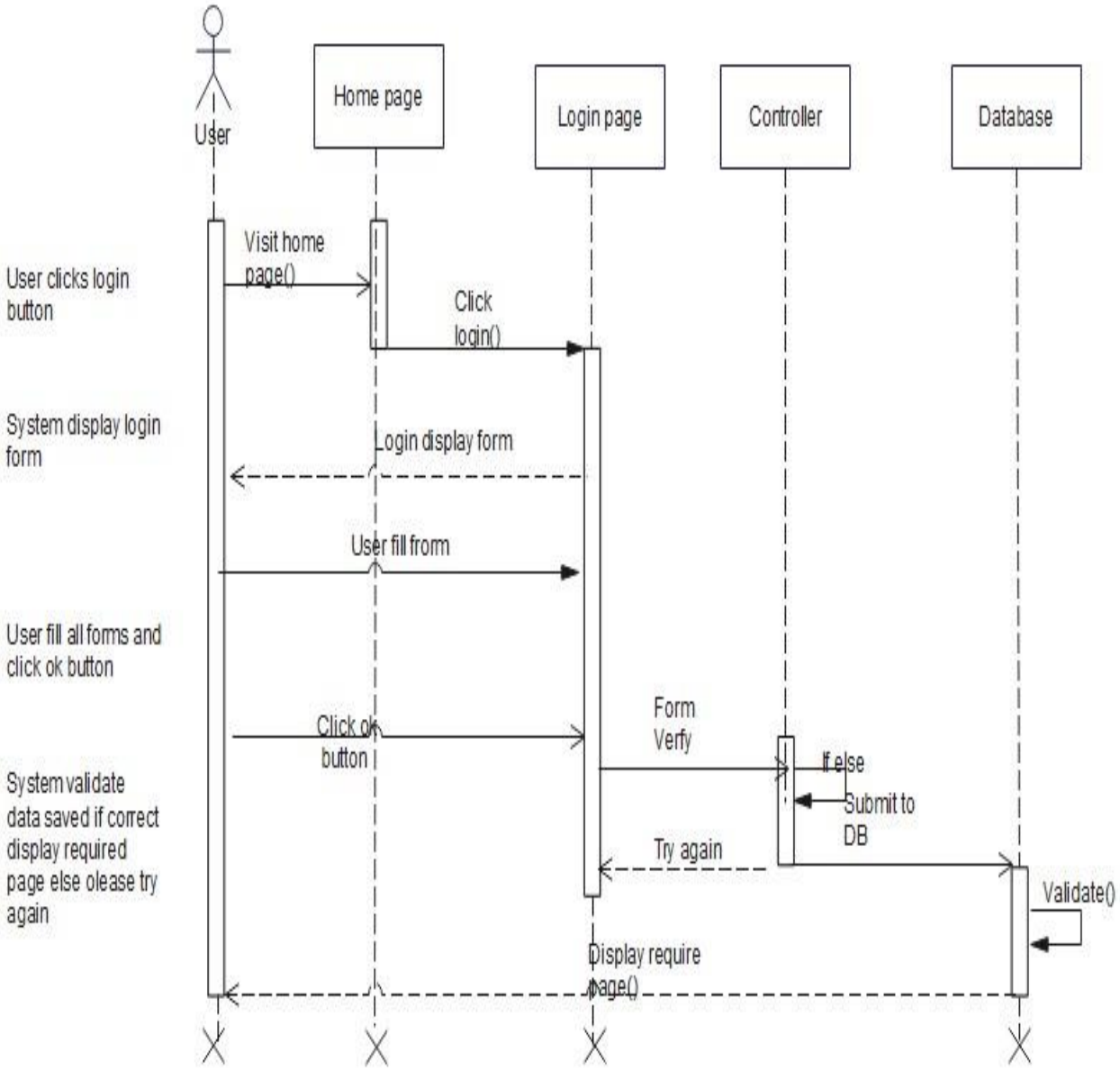

Figure 4. 3 Sequence diagram for Login

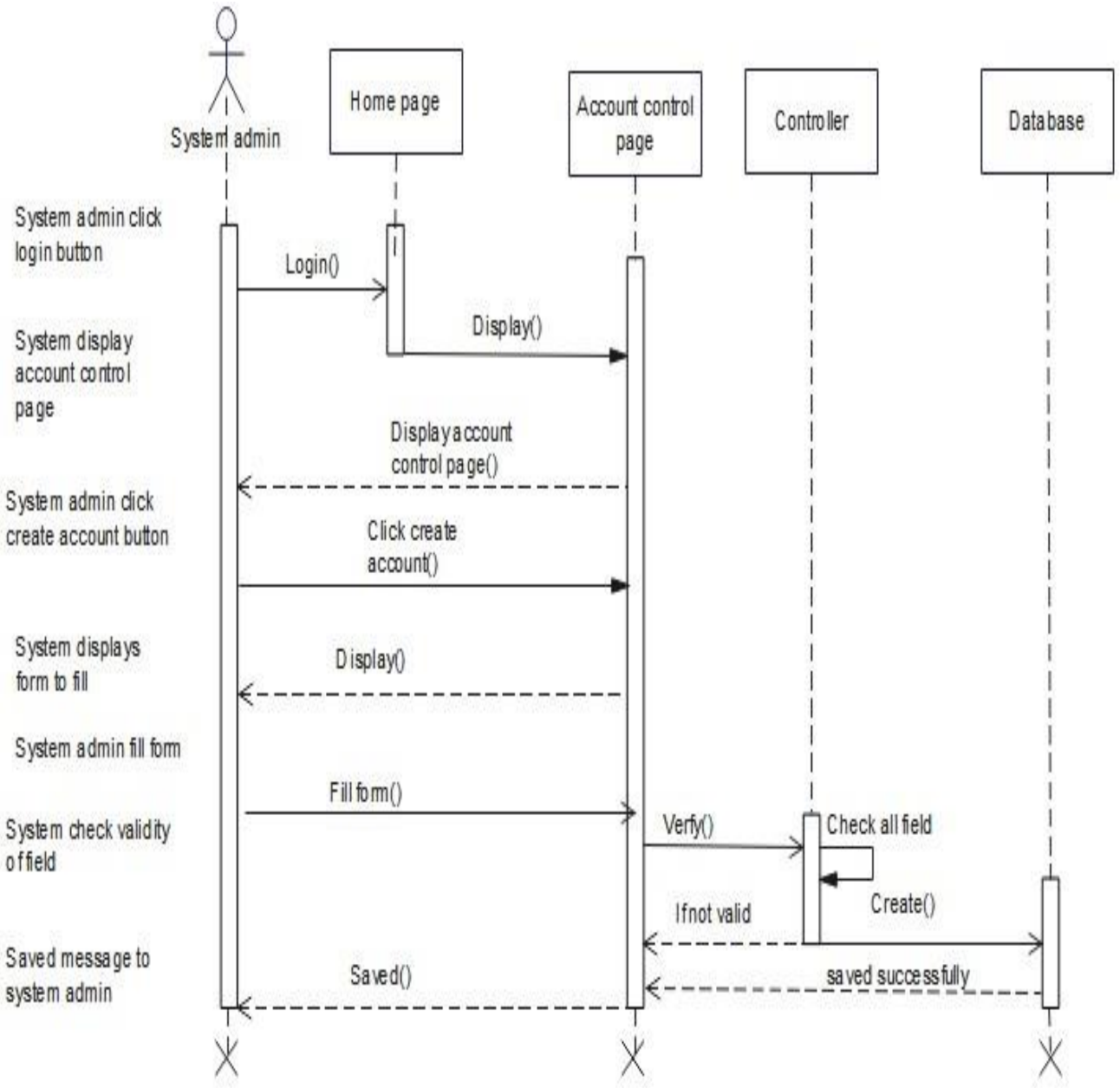

Figure 4. 4 Sequence diagram for creating an account

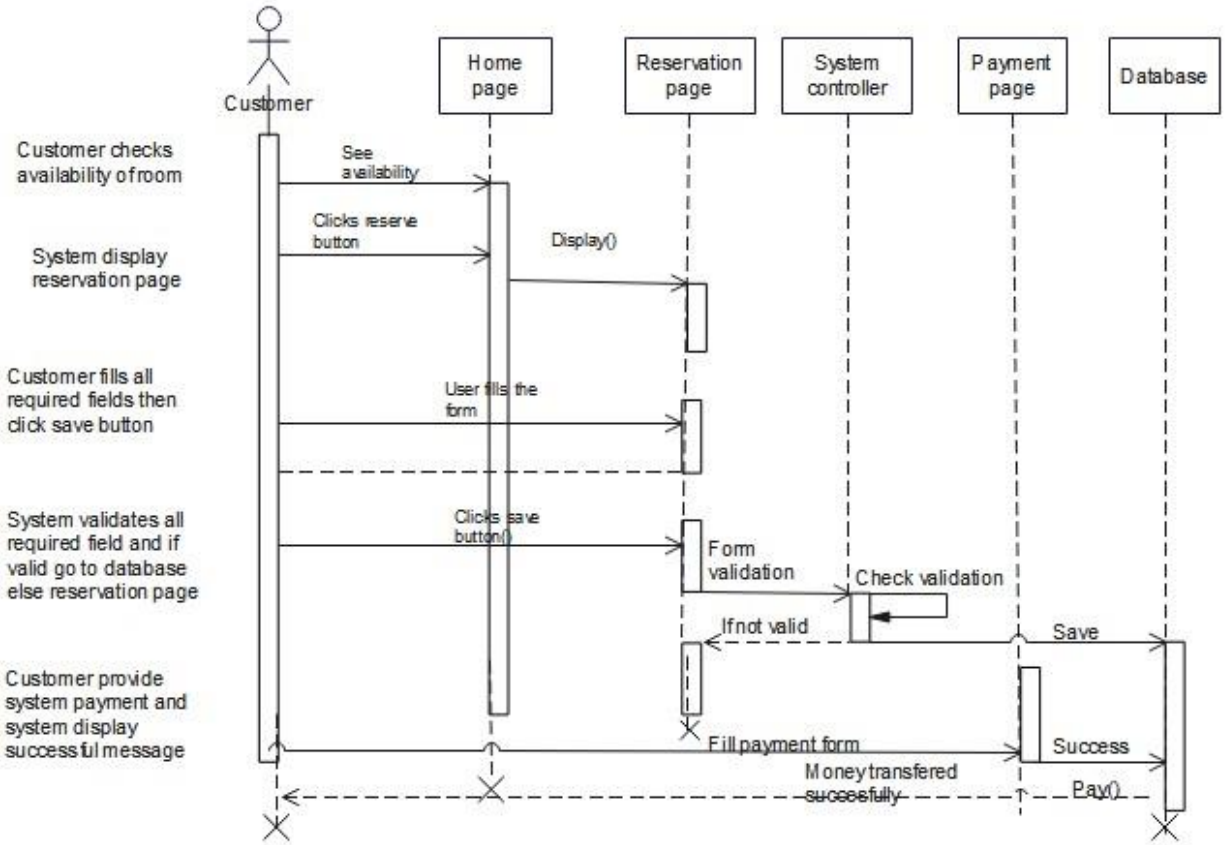

Figure 4. 5 Sequence diagram for Reservation

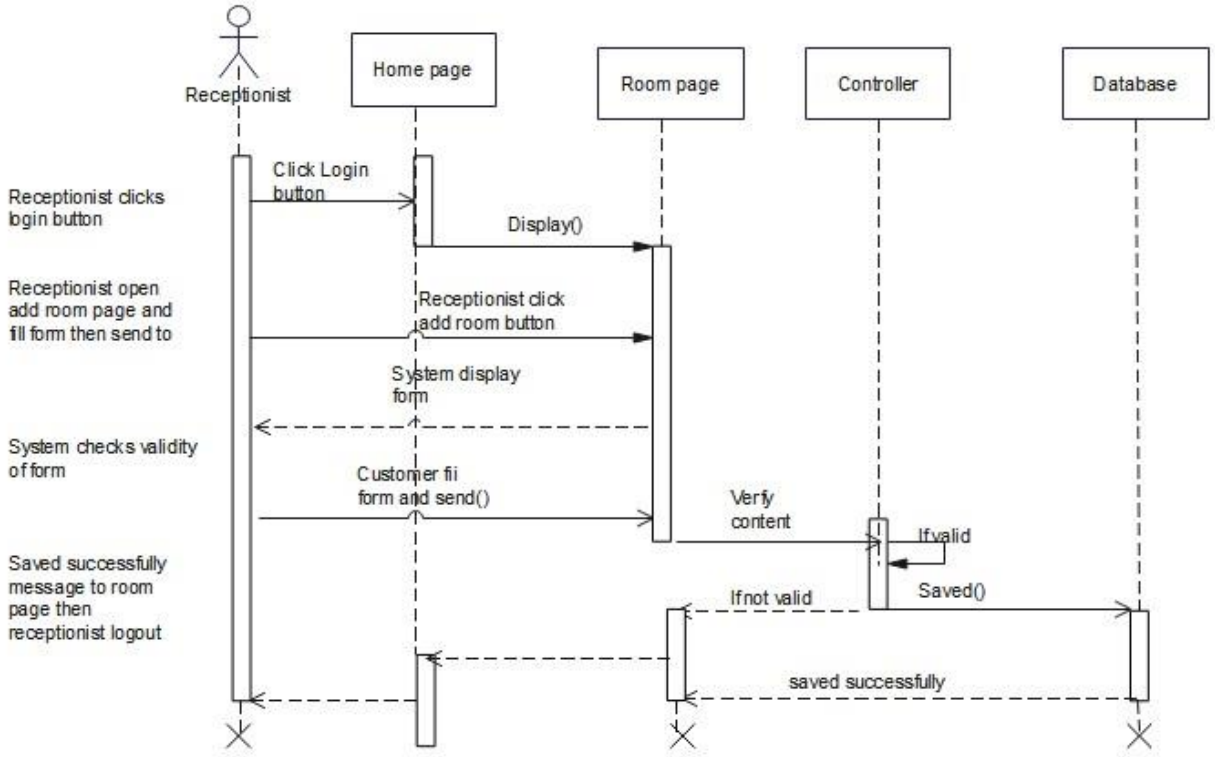

Figure 4. 6 sequence diagram for Add employee

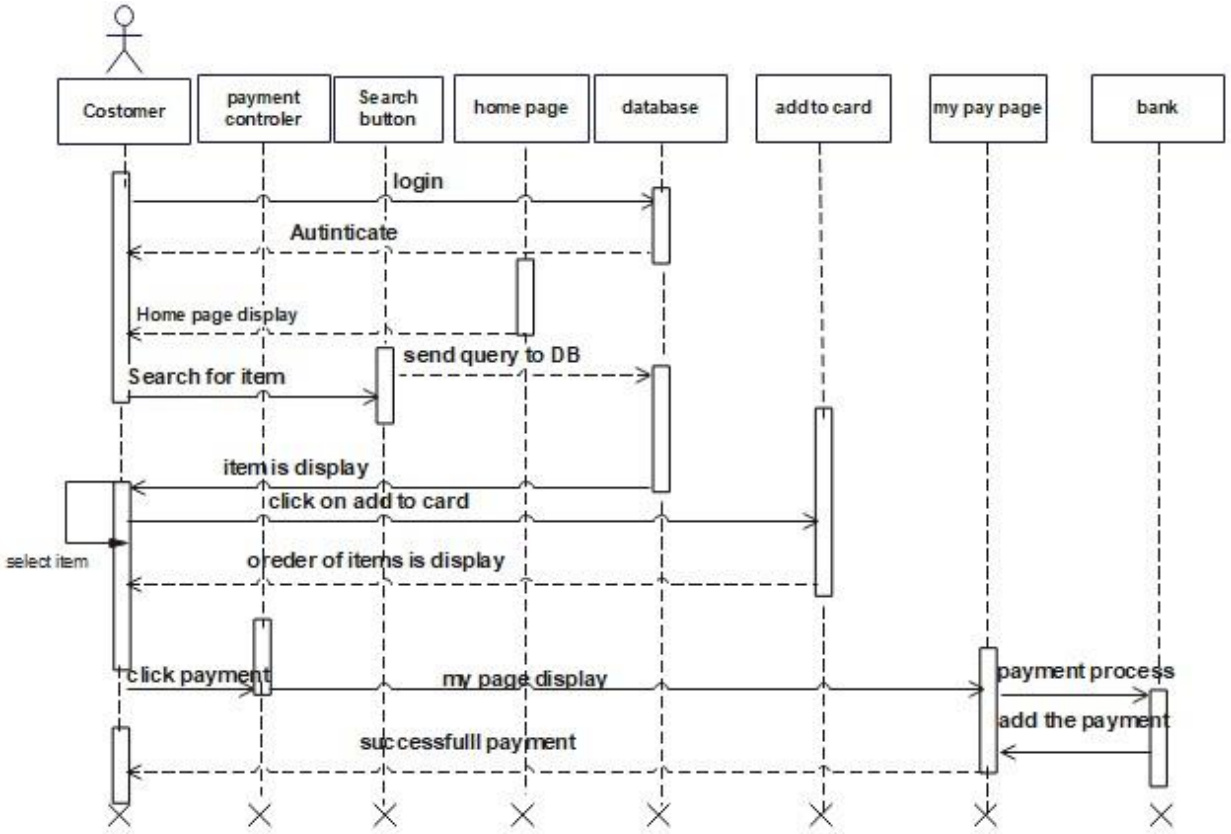

Figure 4. 7 sequence diagram for Make payment

4.3.2 Activity diagram

An activity diagram is basically a flow chart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent [3].

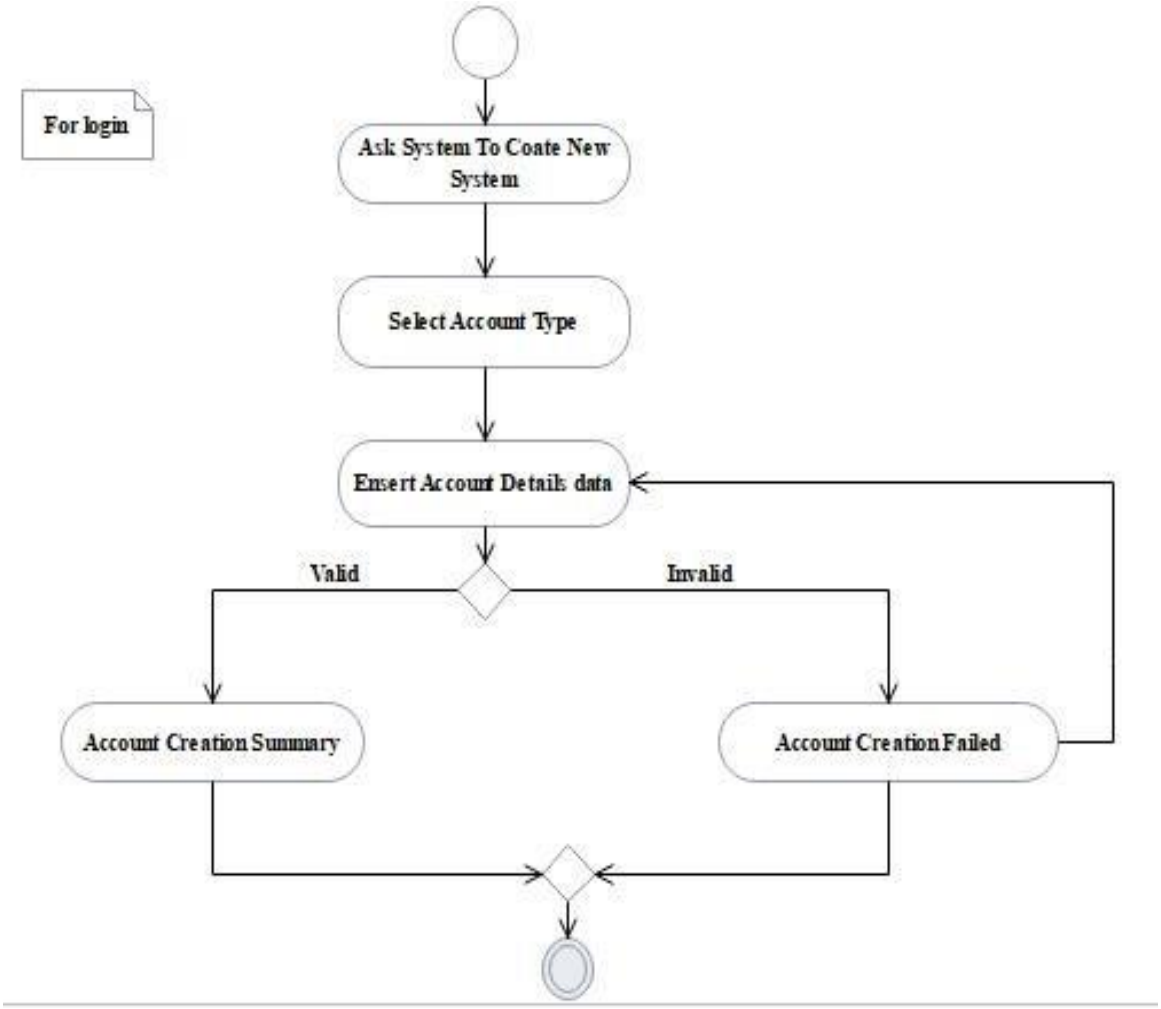

Figure 4. 8 Activity diagram for login

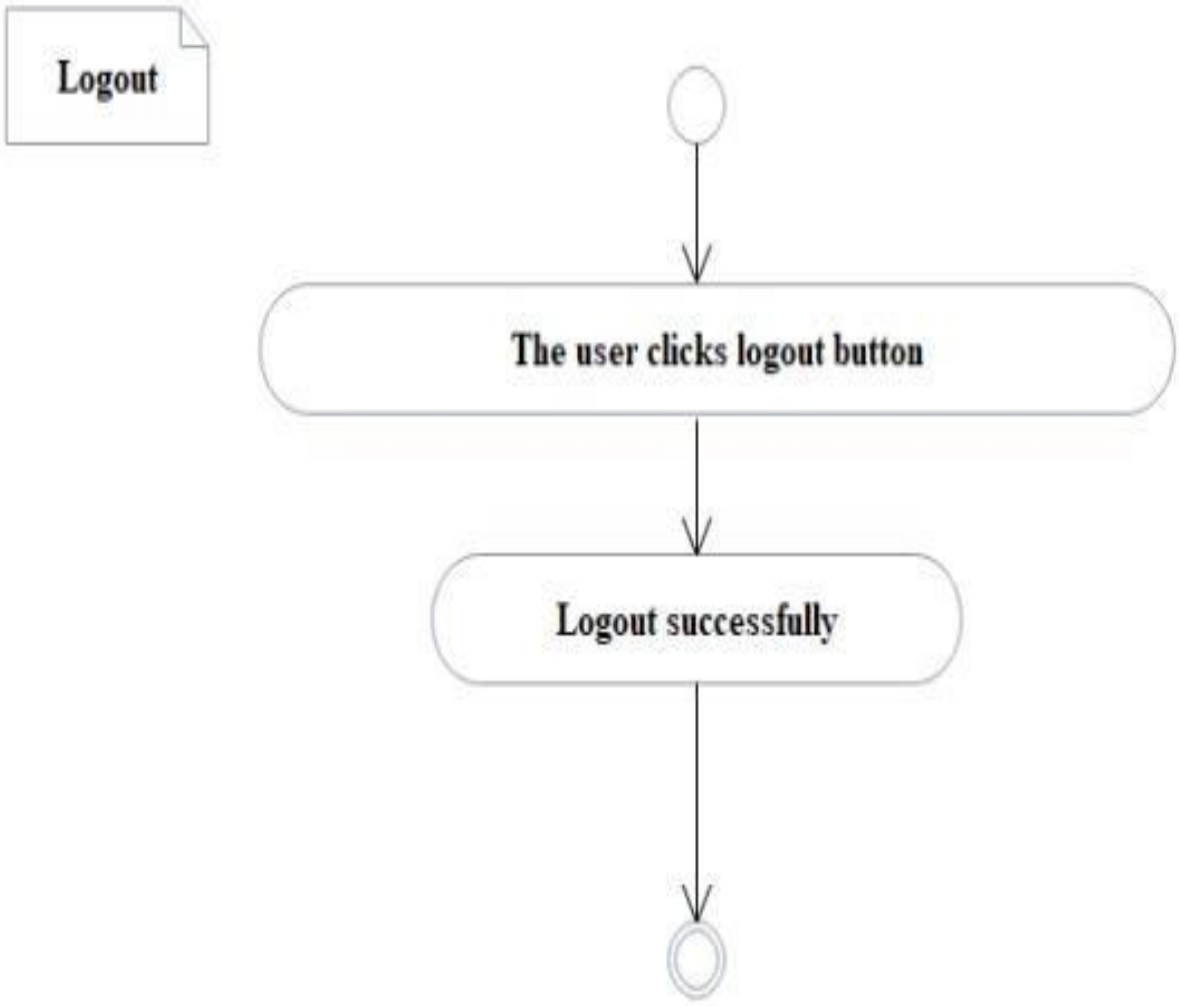

Figure 4. 9 Activity diagram for logout

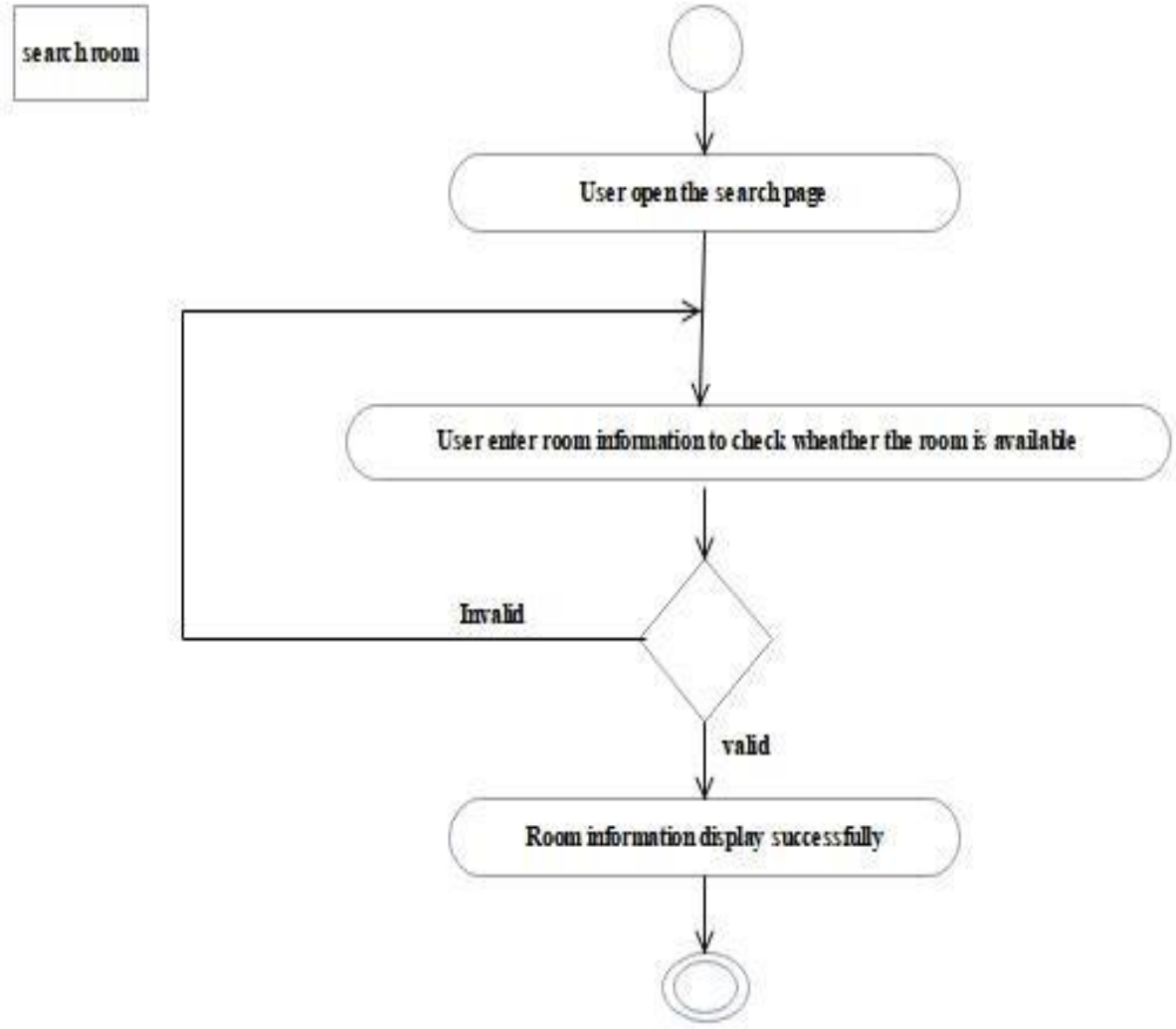

Figure 4. 10 Activity diagram for search room

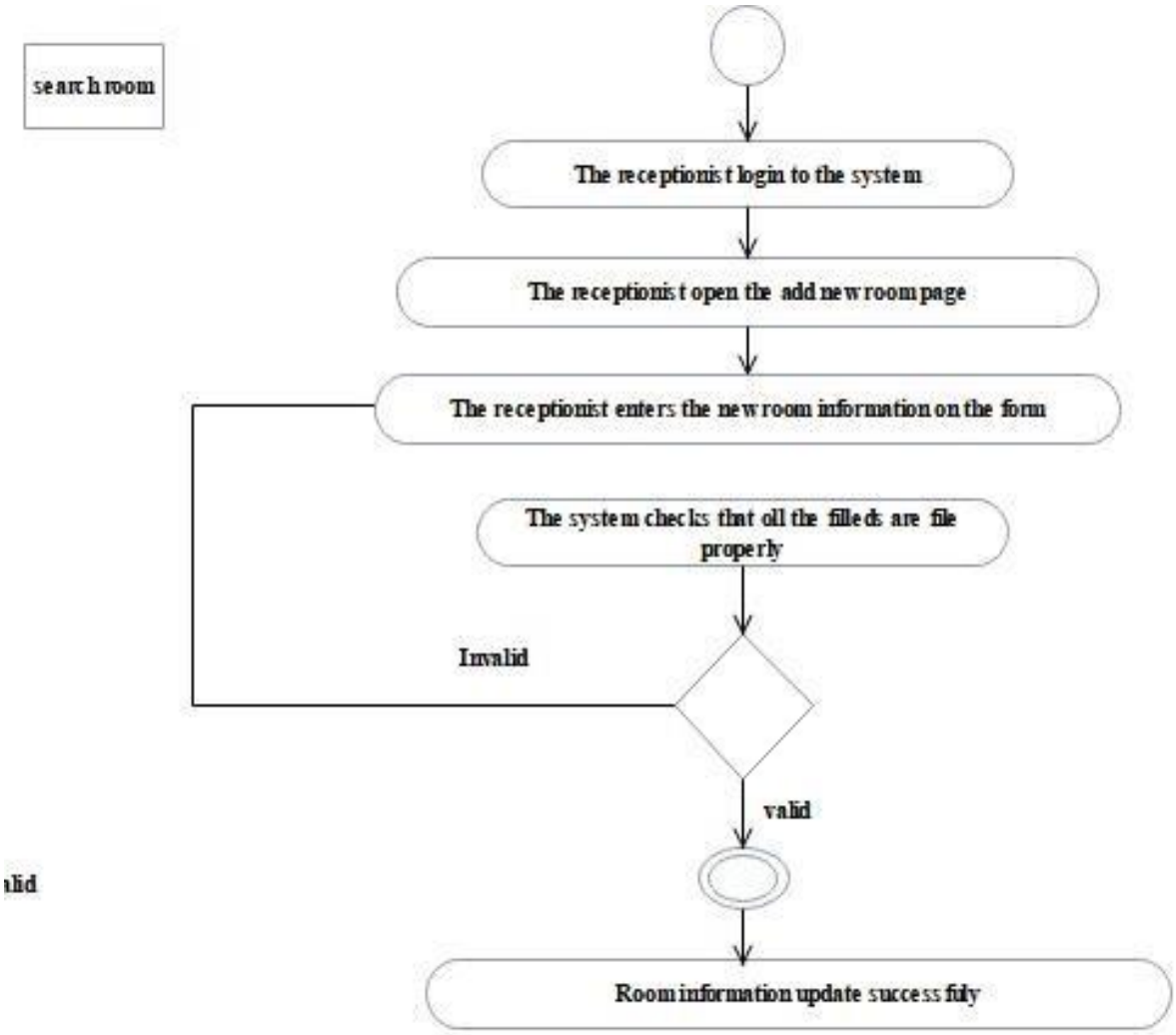

Figure 4. 11 Activity diagram for add room

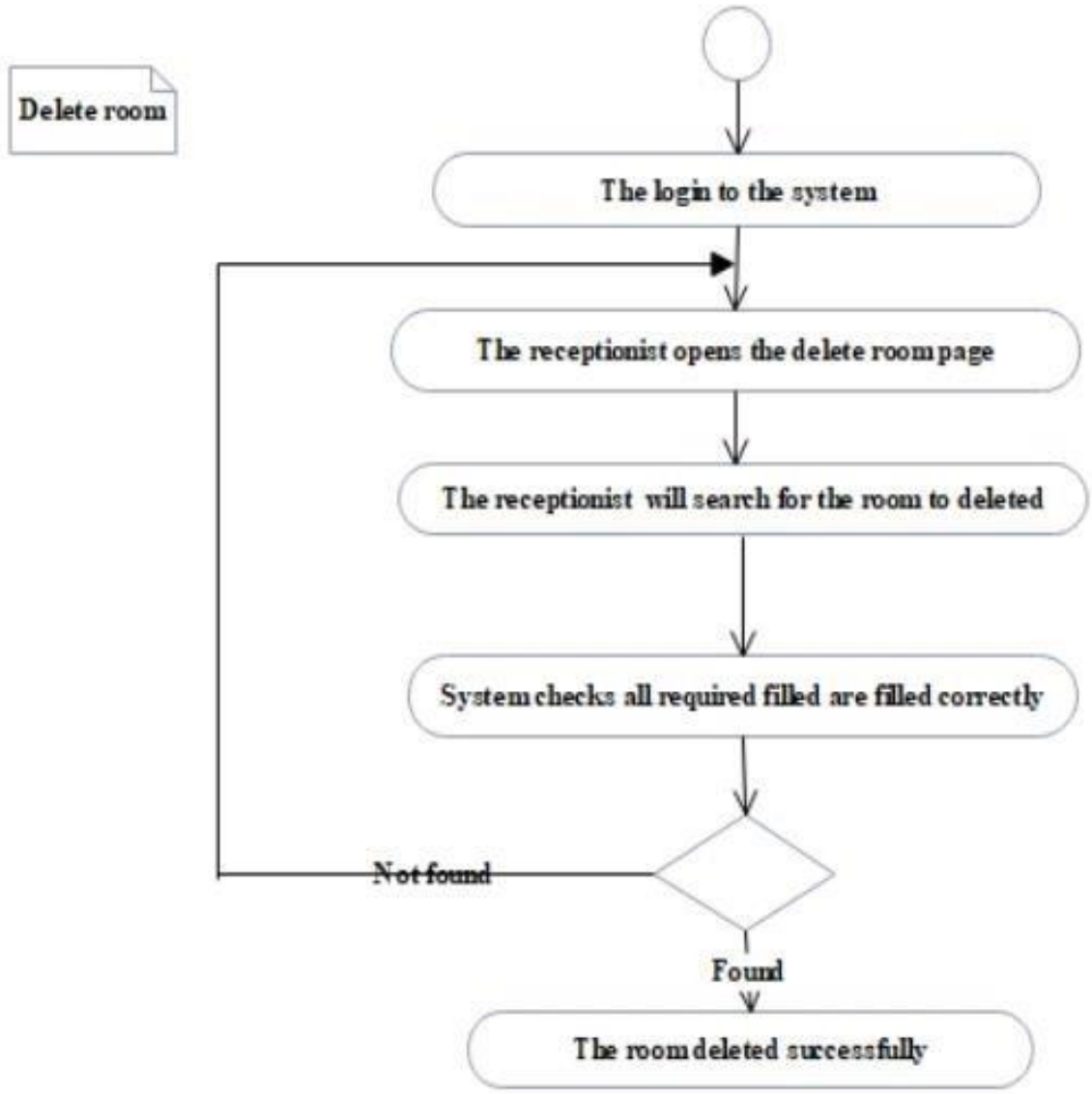

Figure 4. 12 Activity diagram for delete room

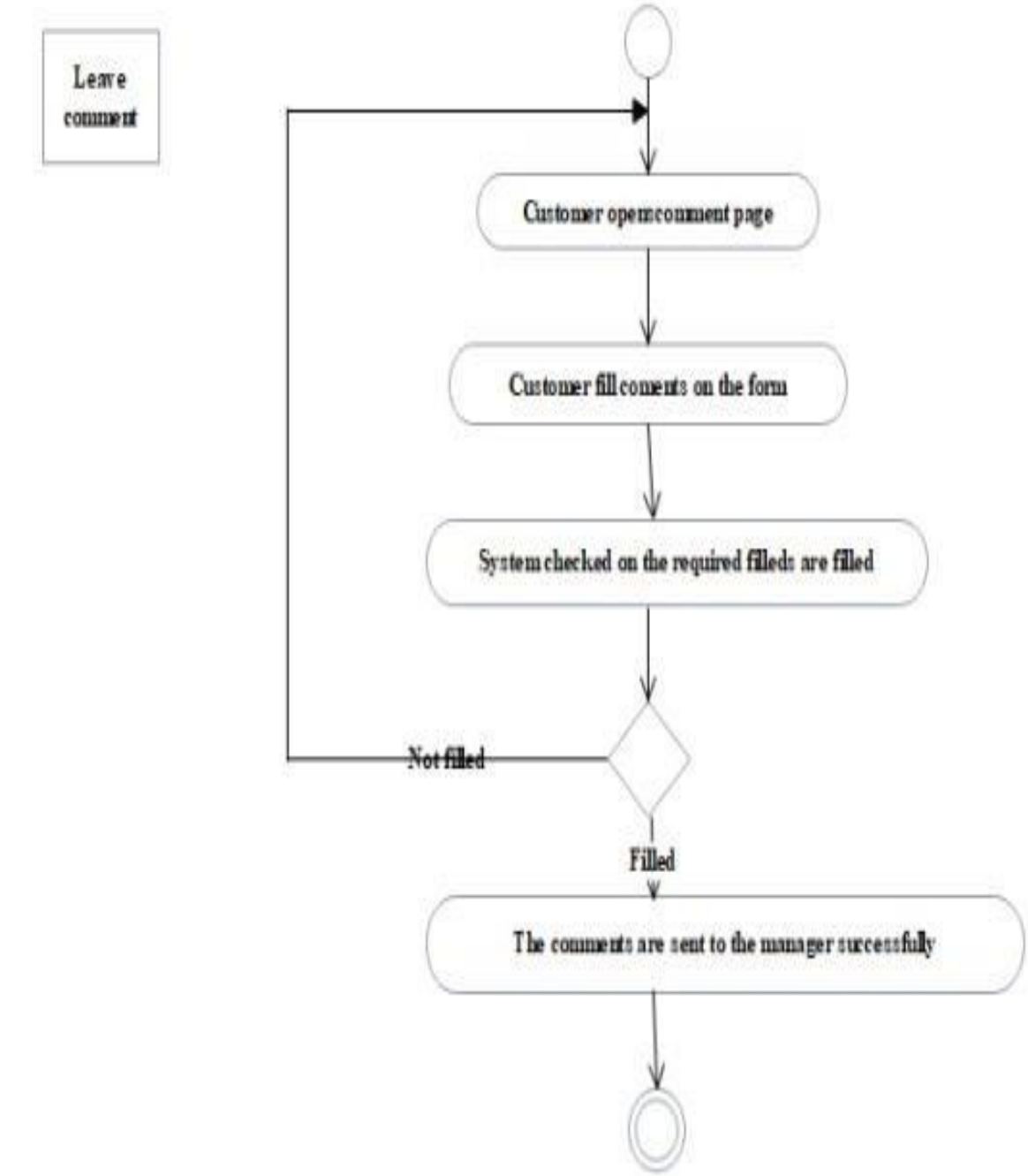

Figure 4. 13 Activity diagram for leave a comment

4.3.3 State chart diagram

In these topics, the various states that an object goes through, as well as the event that cause a transition from one state to another should be described. And also, various states an object may be in and the transitions between those states should be described. The common model elements that the statechart diagram contains are: -

- ✓ Start and end state
- ✓ Transitions
- ✓ State

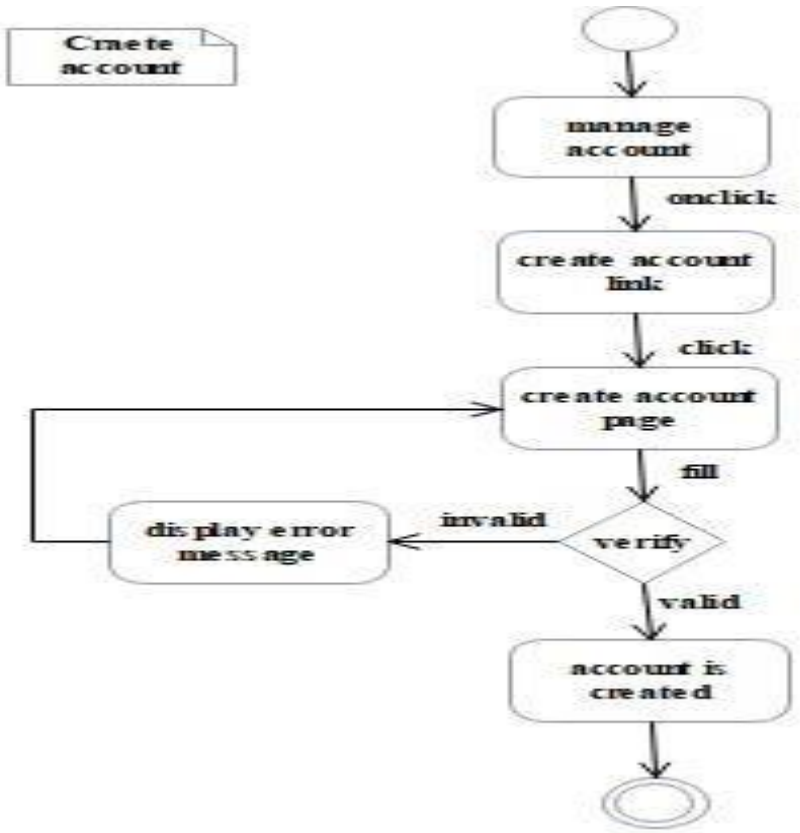

Figure 4. 14 State Chart diagram for creating an account

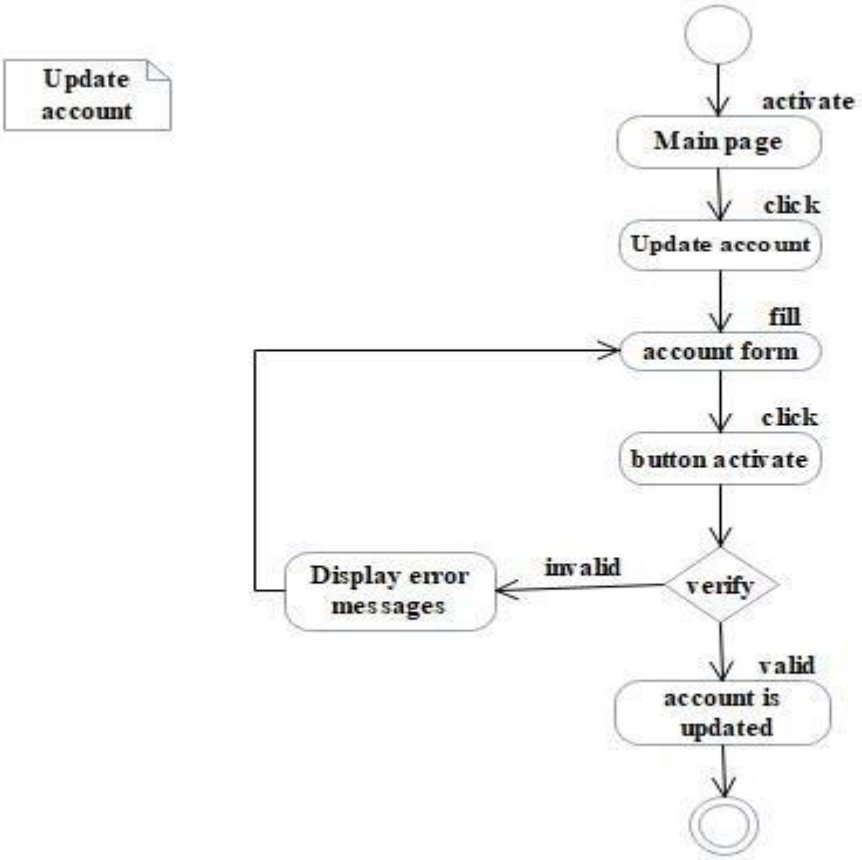

Figure 4. 15 State chart diagram for update account

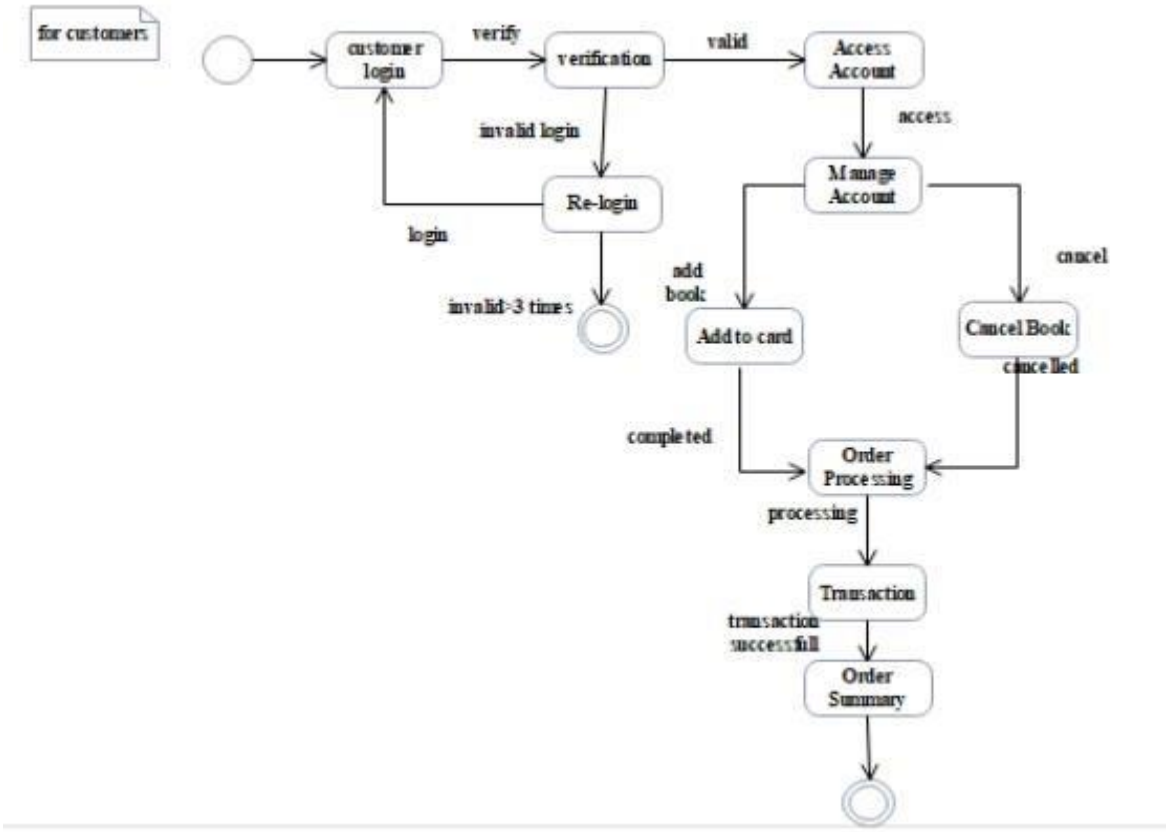

Figure 4. 16 State chart diagram for customers

CHAPTER FIVE

5. SYSTEM DESIGN

The purpose of designing is to show the direction of how the web page is built and to obtain clear and enough information needed to drive the actual implementation of the web page. It is based on an understanding of the model the web page built on system design also focuses on decomposing the system into manageable parts.

System design is the transformation of the analysis model into a system design model. System design is the first part to get into the solution domain in software development**.**

5.1 Design Goals

The goal of system design is to satisfy the functional and non-functional requirements as specified in the requirements specification document.

The following design goal promotes.

- ✓ **Security**: The system should be available for twenty-four hours of a day so that the users can access it at any time there the system should be designed to prompt the user with a password and user name this provides security in such a way that unauthorized users can not access the system's resources and information.
- ✓ **End-user:** The system should provide user-friendly and self-explanatory graphical there for the system design play a great role between user and system on the interface that interacts users with the system.
- ✓ **Readable**-the system is easy to understand since the system design shows the enduser in a simple way
- ✓ **Performance:** The response time to users' requests should be tolerable which is faster and more accurate.

- ✓ **Availability**: the system is available at any time as long as there is no power and internet access.
- ✓ **Adaptable**-it is adaptable and easily learn since the system design creates an image within end-users
- ✓ **Usability**: the system interface should be interactive with the users so as to enable them to use the system effectively.
- ✓ **Maintainability:** the system can handle errors and can be maintained if any system structure changed and the system should be easily modifiable.
- ✓ **Portable**- to be platforms independent.

The purpose of this design document is to provide the design models and methodologies that are developed and used to satisfy the requirements. This document starts with an introduction to the architecture and the design goals to be considered then it presents the proposed system architecture by describing the subsystem decomposition and the subsystem services also the hardware/software mapping is defined and the management of persistent data is explained. Access control and security issues are addressed. The global software control and boundary controls are described.

5.2 Proposed System Architecture

The proposed software architecture for Soresa hotel shows the decomposition of the system, since it is web-based it is easy to manage by looking at the architecture of the system because it is an automated system and the data management is well handled and stored in a database, as a result, an efficient and accurate data processing, consistent and integrated system architecture is easily designed this describes the overall system architecture of online hotel reservation with mobile banking and how it is decomposed**.**

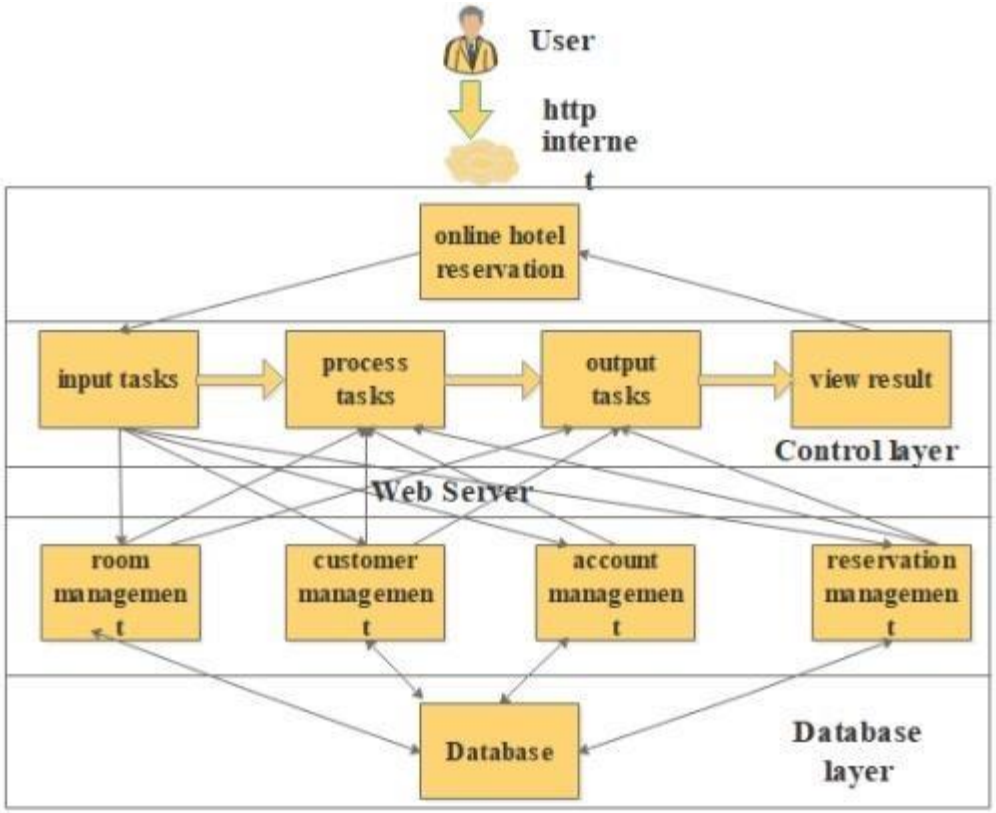

Figure 5. 1 System architecture

5.2.1 Subsystem Decomposition and Description

System decomposition refers to the process by which a complex problem or system is broken down into parts that are easier to conceive, understand, program, and maintain. To reduce the complexity of the solution domain, we decompose our system into simpler parts are called subsystems, which are made of a number of solution domain classes. In the case of complex subsystems, we recursively apply this principle and decompose a sub-system into simpler subsystems. We decompose our system into Four (4) subsystems. These are:

System administrator subsystem: - This subsystem perform this action

- Create account
- Deactivate Account
- Update Account

Manager subsystem: - This subsystem perform

- Register employee
- Generate Report ü
- Delete Employee
- View Report
- Update employee
- See feedback

Customer subsystem: - This subsystem perform this action

- Reserve room
- Cancel reservation
- Provide payment
- Provide feedback
- Check availability of room

Receptionist subsystem: - This subsystem perform this action

- Post available rooms
- See a list of reservation
- Generate report
- Check payment reservation
- See lists of reservation
- Add,update,delete rooms
- Add,update,delete food

The Database Subsystem is responsible for all insertion, reading of all data by all users. No data will be accessed in any way except through and by the Database Subsystem. Each user within the Subsystems has responsibilities related to their privileges like inserting, updating, and modifying their own data by utilizing the corresponding methods belonging to the database within the database Subsystem.

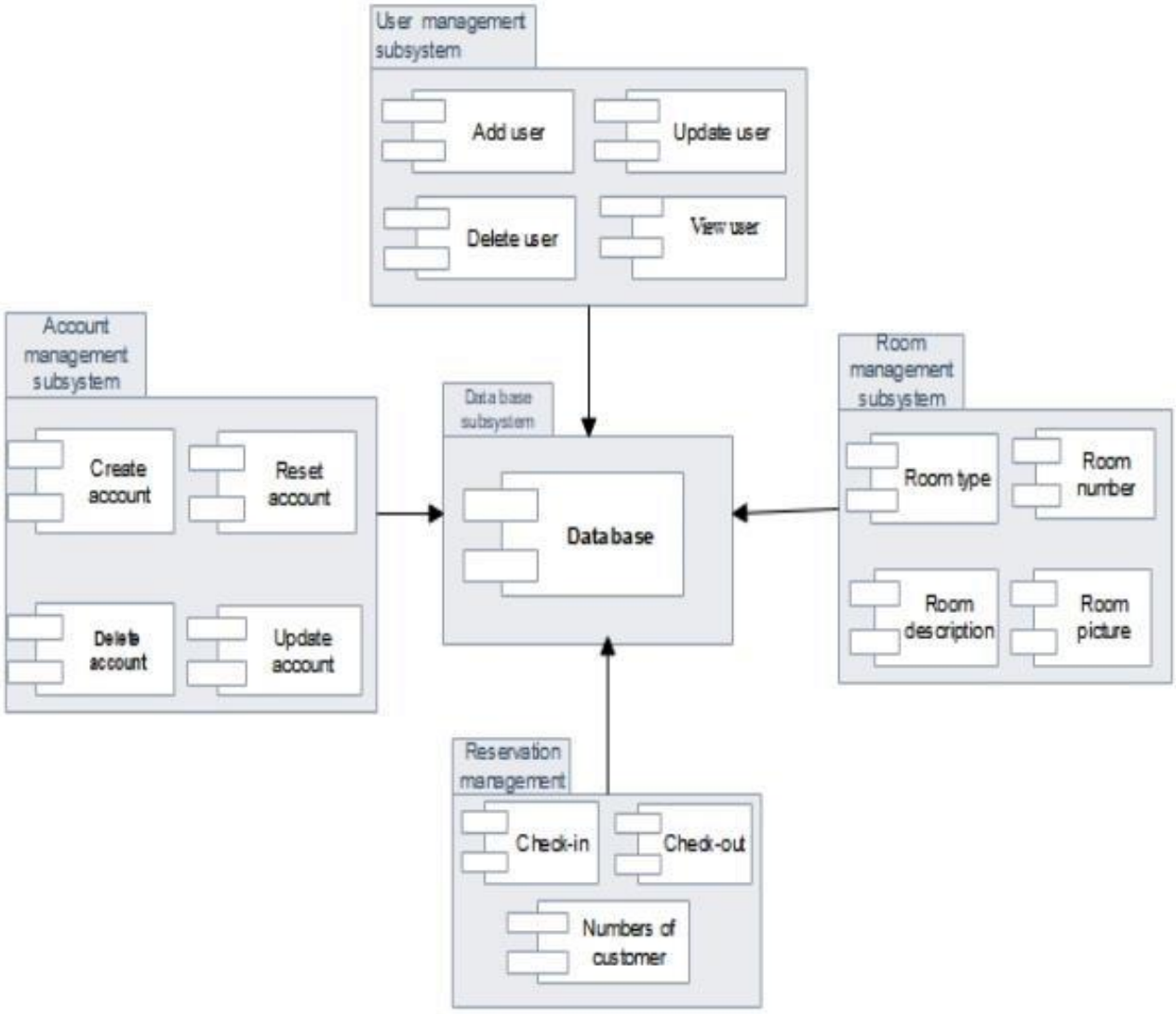

Figure 5. 2 System decomposition

5.2.2 Hardware and software mapping

Hardware or software mapping describes how subsystems are assigned to hardware and customized components. We, use a UML deployment diagram to diagrammatically illustrate the hardware/software mapping of our proposed system. [3]

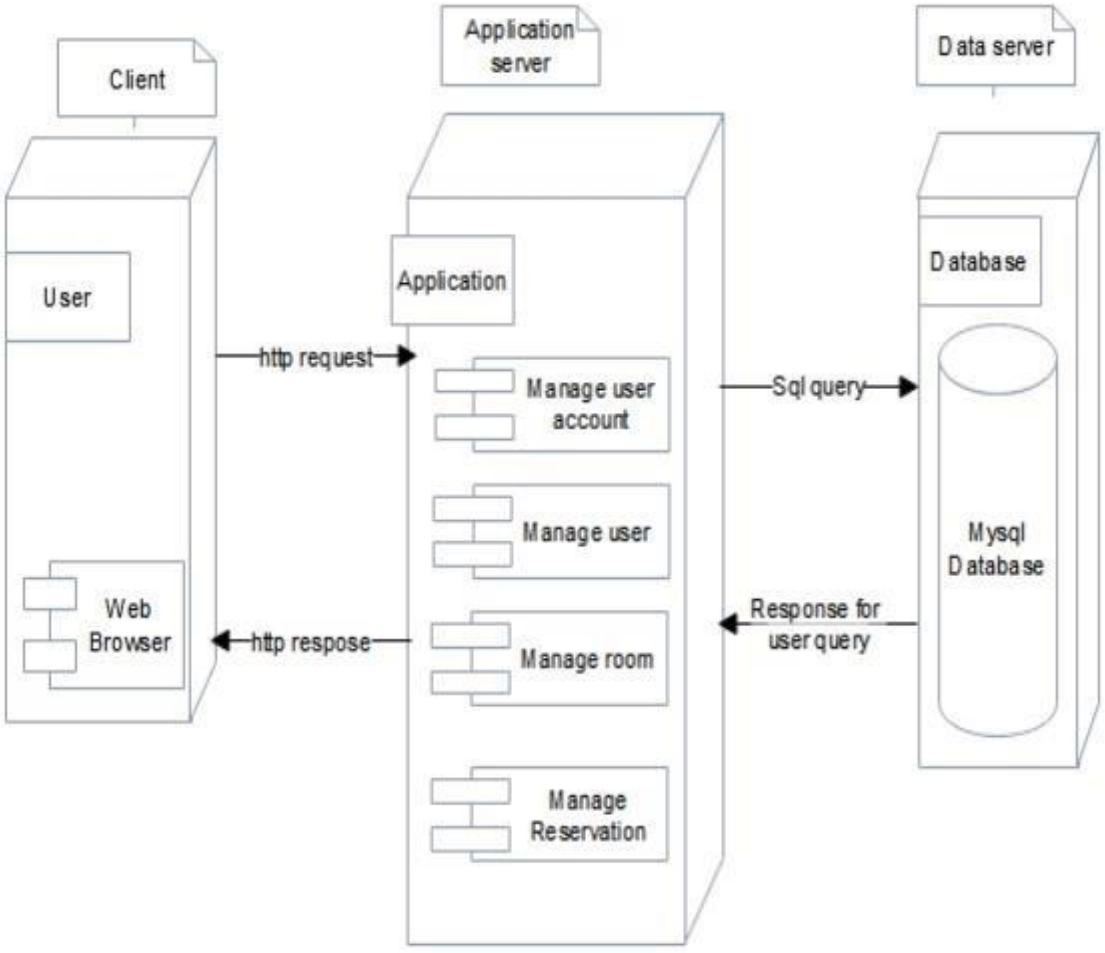

Figure 5. 3 Deployment diagram

5.2.3 Detailed Class Diagram

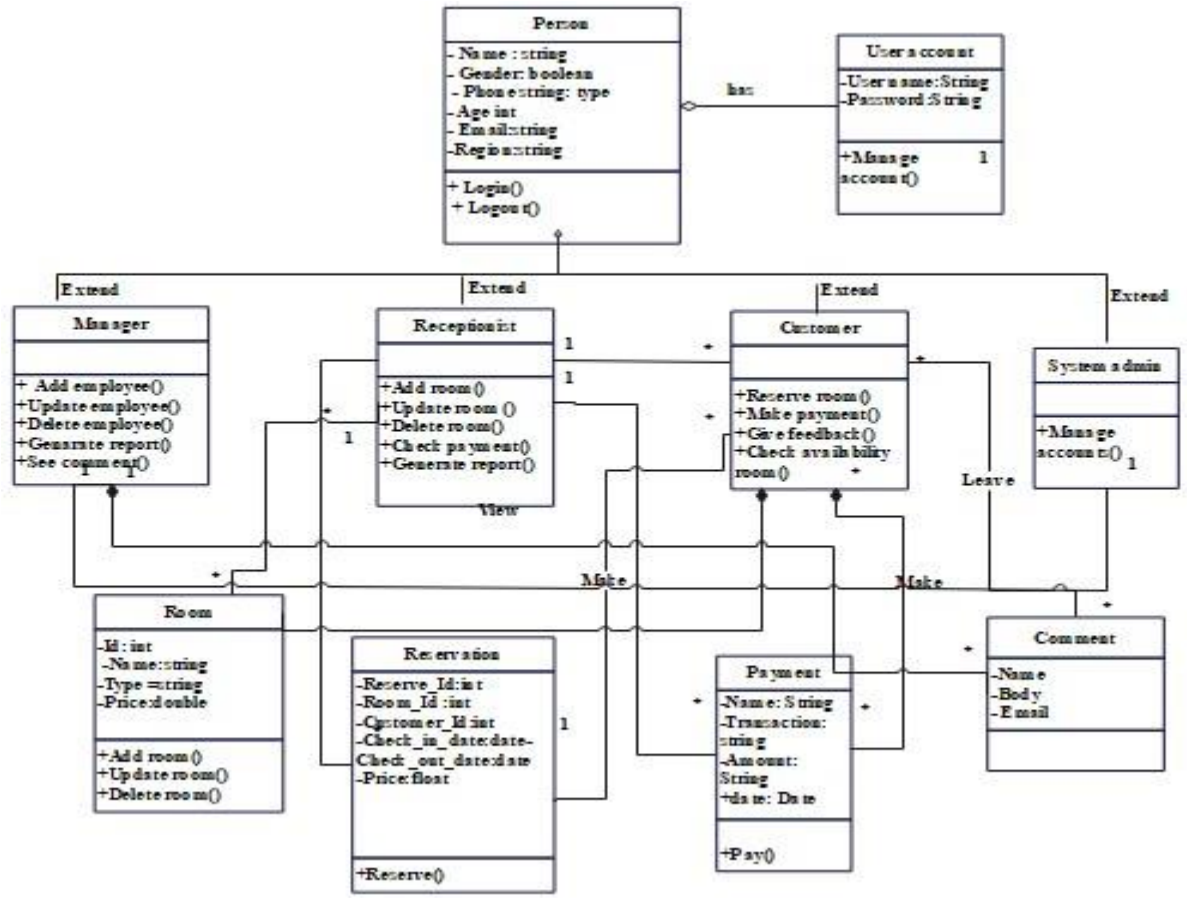

Figure 5. 4 Detailed class diagram

5.2.4 Persistent Data Management

Persistent data management describes the persistent data stored by the system and the system uses the MySQL database engine for storing data. This will allow the database to be easily integrated with and accessed by the rest of the system. Our system will largely depend on a relational database to perform operations and storing data there for data will be stored and manipulated through the Database Subsystem, which will ensure data integrity and consistency. And also database Subsystem will contain all necessary SQL queries that will be accessible by the rest of the Subsystems.

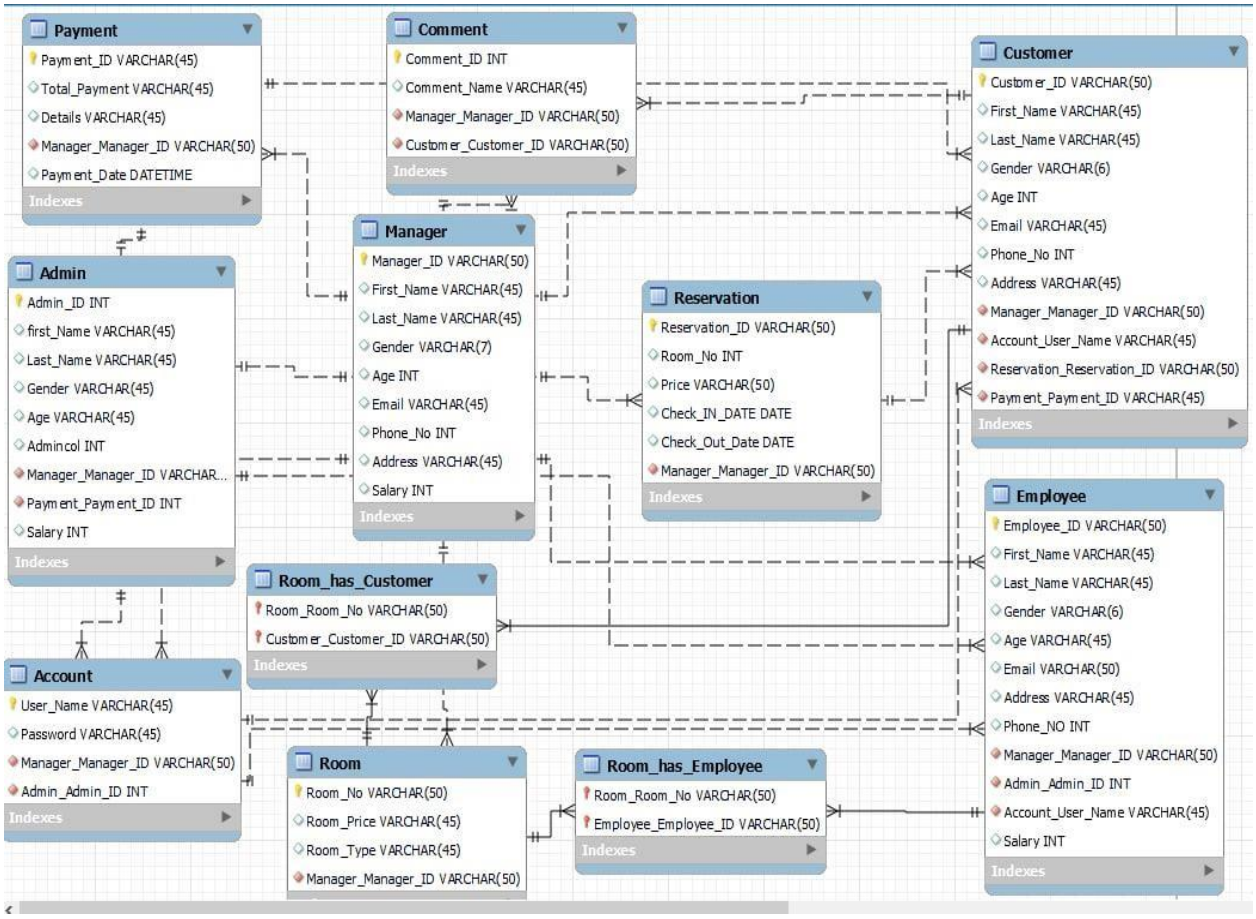

Figure 5. 5 Persistent diagram

5.2.5 Access Control and Security

In many multi-user websites or systems, different users and actors have access privilege to different functionalities and data processing. Based on this, we have designed the system application with different privileges to its users since authentication is the first security that protects our system from illegitimate users so our system has the following users these are admin, manager, customers and, a receptionist with their own privileges. The system admin have the privileges to manage account (creatan account, delete account, update account, change password account and manage users). The manager has the authority to manage the room (add room, delete room, update room and view reports). The customers have the right to create account, make reservation, cancel

reservation, and give feedback. The receptionist has the right to view reservation list, to approve the reservation of customers and to give feedback.

Table 5. 1 Control Matrix

	Administrator	manager	Customers	reception
Login	Yes	Yes	Yes	Yes
Delete account,	Yes	No	No	No
update account,	Yes	No	No	No
create account	Yes	Yes	Yes	Yes
Change password	Yes	Yes	Yes	Yes
Add employee	No	Yes	No	No
Update employee	No	Yes	No	No
Delete employee	No	Yes	No	No
Add room	No	No	No	Yes
Delete room	No	No	No	Yes
Update room	No	No	No	Yes
View report	Yes	Yes	No	
Search rooms	No	No	Yes	Yes
Making reservation	No	No	Yes	Yes
Cancel reservation	No		Yes	Yes
Give feedback	No		Yes	Yes
Logout	Yes	Yes	Yes	Yes

5.3 Packages

Packages diagram is a UML structure diagram that shows packages and dependencies between packages. Packages diagram enables you to gain a high-level understanding of the collaboration among model elements through analyzing the relationship among their parent packages [2].

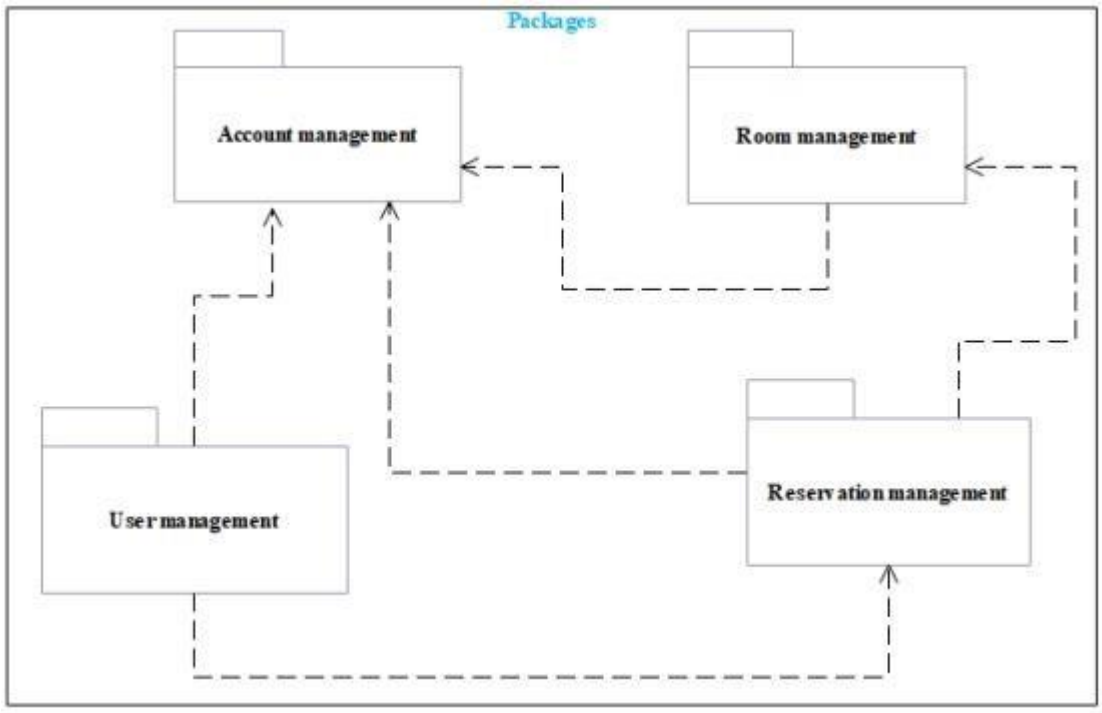

Figure 5. 6 Packages diagram

5.4 Algorithm Design

Pseudocode: - It's simply an implementation of an algorithm in the form of annotations and informative text written in plain English. It has no syntax like any of the programming languages and thus can't be compiled or interpreted by the computer. Improves the readability of any approach. It's one of the best approaches to start the implementation of an algorithm. The main goal of a pseudo-code is to explain what exactly each line of a program should do, hence making the code construction phase easier for the programmer.

Pseudocode: A Login form module

##Pseudocode: Reservation

1. Home page is displayed
2. See availability
3. Customer checks availability of room
4. System display reservation page
5. Customer click reserve button
6. Customer fills all required fields then click save button the system check validation
7. System validates all required field and
8. If the fill form is valid go to database
9. Else back to reservation page
10. If customer provide payment and system display successful message
11. Else the payment not provide successful message displayed

Pseudo code for add employee

1. click login button
2. Open add room page
3. Fill form of the room information
4. Verfy content
5. If valid save successfully
6. If not valid return to verify content

5.5 User Interface design

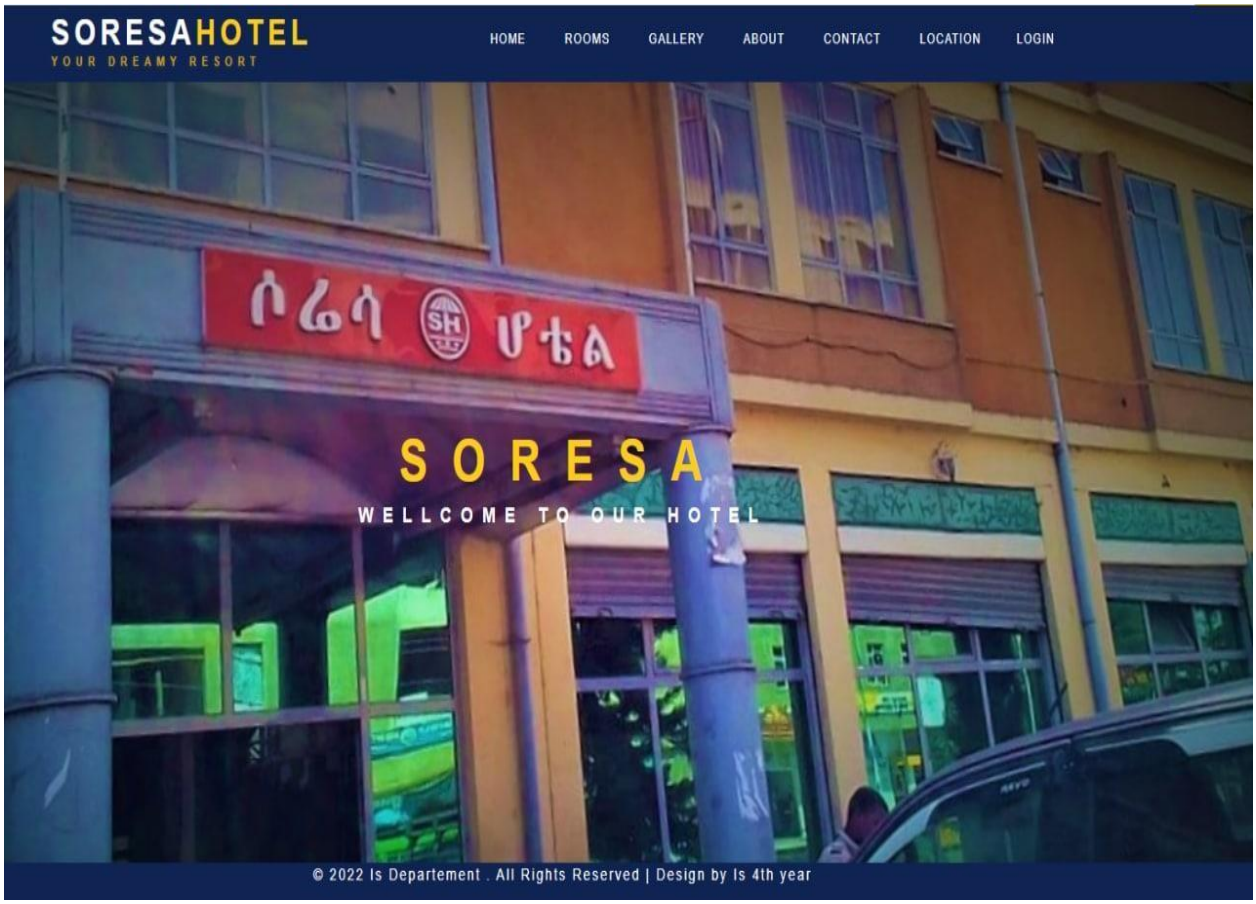

Figure 5. 7 Home page

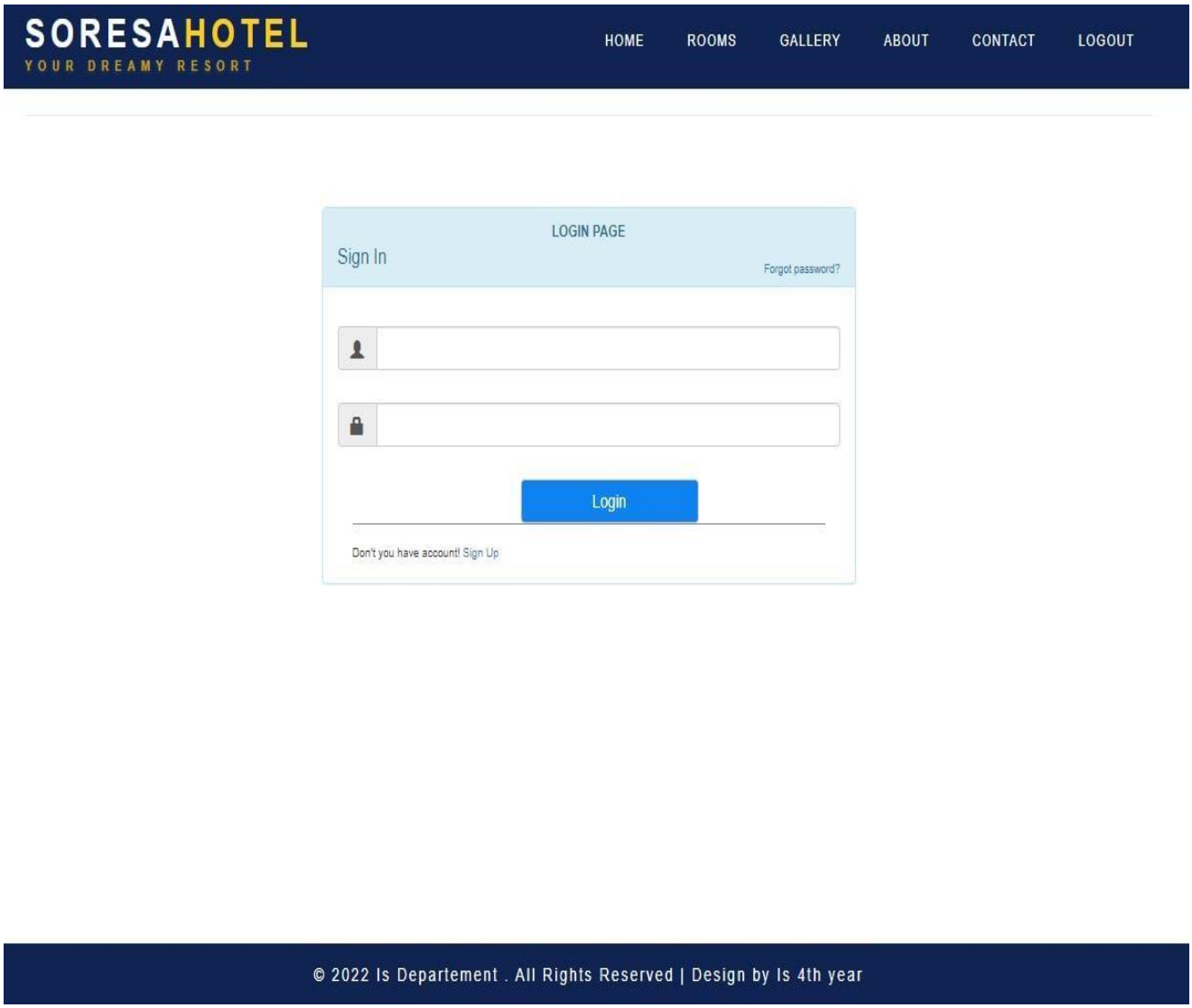

Figure 5. 8 Login Page

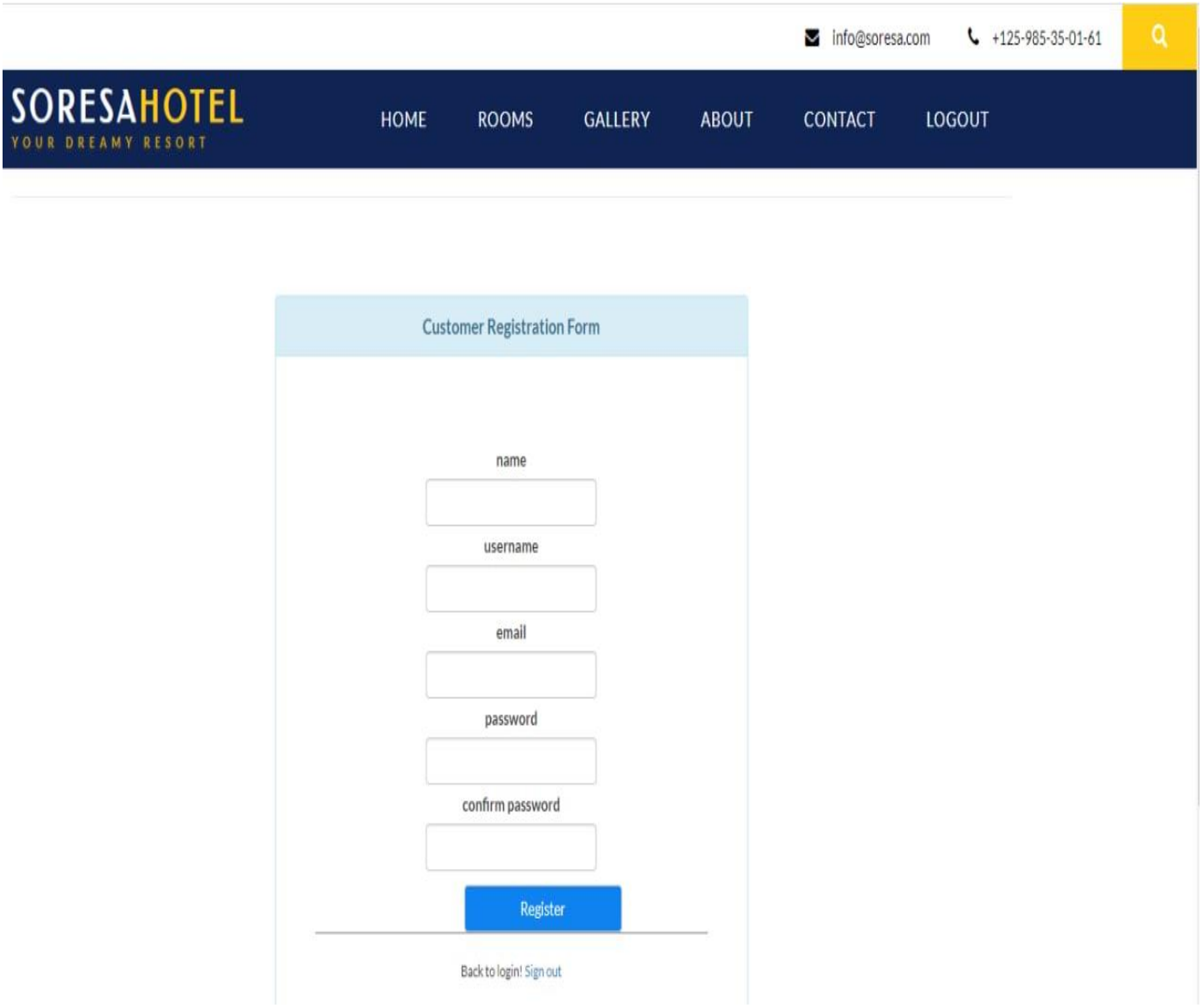

Figure 5. 9 Reservation Page

CHAPTER SIX

6. IMPLEMENTATION AND TESTING

This chapter basically highlights the issues deal with the implementation phases. Implementation is the phase where objectives of physical operations of the system turned into reality i.e. real working model. In this phase the coding convention has made it possible as it's the real phase of objectivity to reality. Then the code is tested until most of the errors have been detected and corrected. The goal of implementation is to introduce our system for the users in real sense that how they use this new system which is developed for their intended objectives

6.1 Implementation of the database

In order to implement the main application language is MYSQL that used to store database values and used to store it for long time. We have used MYSQL database because that can run any operating or browser application, Installation Is Streamlined, Security Features Are Better, Enhanced Performance, It's Important to Maintain an Environment That's Standardized. MYSQL is easy fast and can used for any type of database weather it is relational or simple database, large or small database.so MYSQL server for implementation our system or used to store our data.

6.2 Implementation of the class diagram

In this class diagram implementation we have implement the class diagram structure. That implements the view of an application, visualizing, describing and documenting different aspect of the system. And also implements the attribute and operation of the class

6.3 Configuration of the Application Server

We use ASP.NET server because ASP.NET is simple, lightweight Apache distribution it is extremely easy to create a local web server for testing and deployment purpose. Everything you needed is to set up a web server – server application (Apache), database and scripting language ASP.NET works equally well on Linux,Mac and Windows. Since it is suitable and the function we listed above we use ASP.NET application serve

6.4 Configuration of application security

Our system called Soresa Hotel management system is a reservation, services & payment issue validates all the inputs by returning error message and suggesting to try again when invalid input occurred. We implement encryption for user password by MD5 when the system admin creates a user account for manager, receptionist and costumer create their account or the user changes their password system encryption the password. The system has its security principles that control unauthorized authentication

6.5 Implementation of user interface

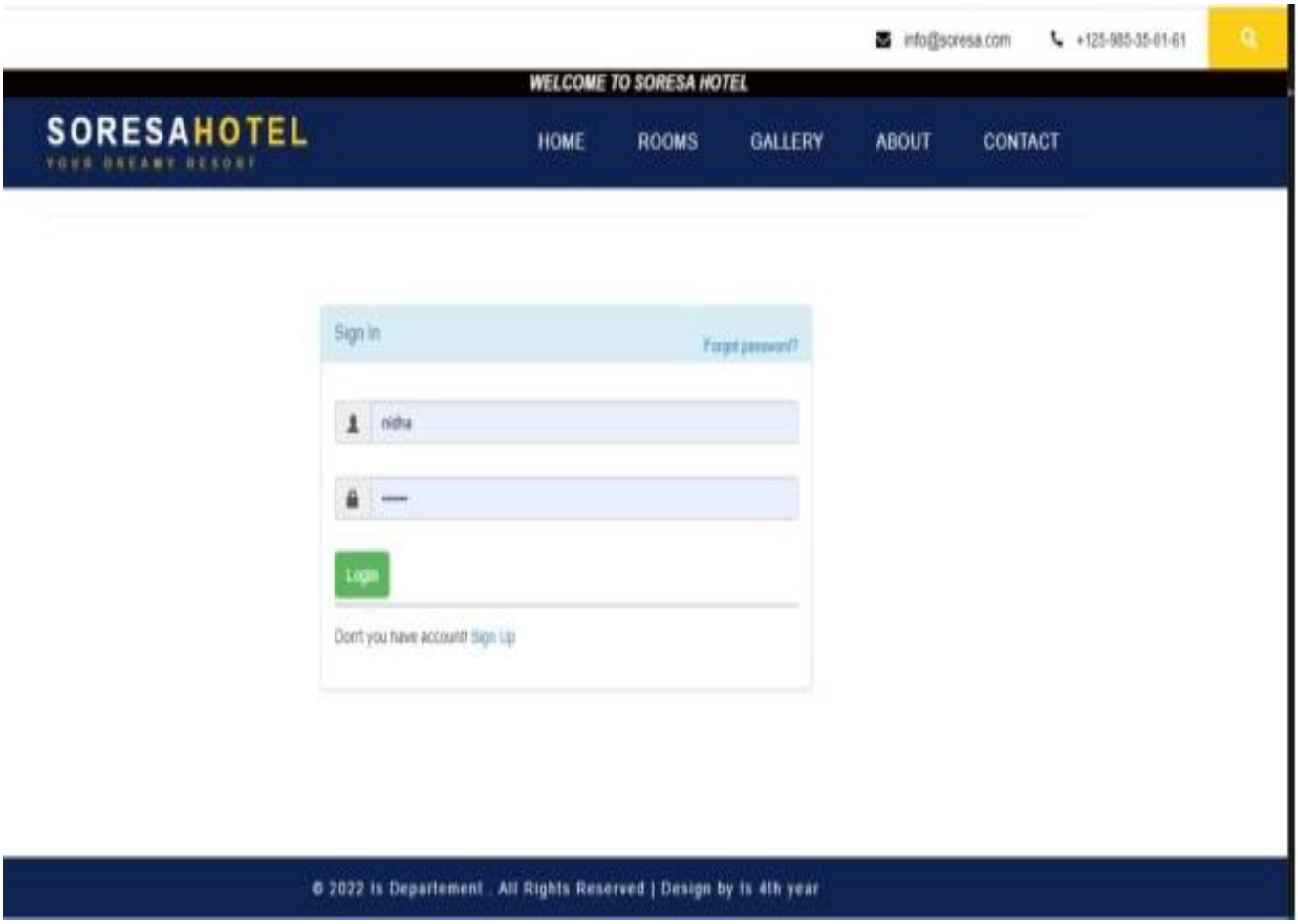

Figure 6. 1 Login

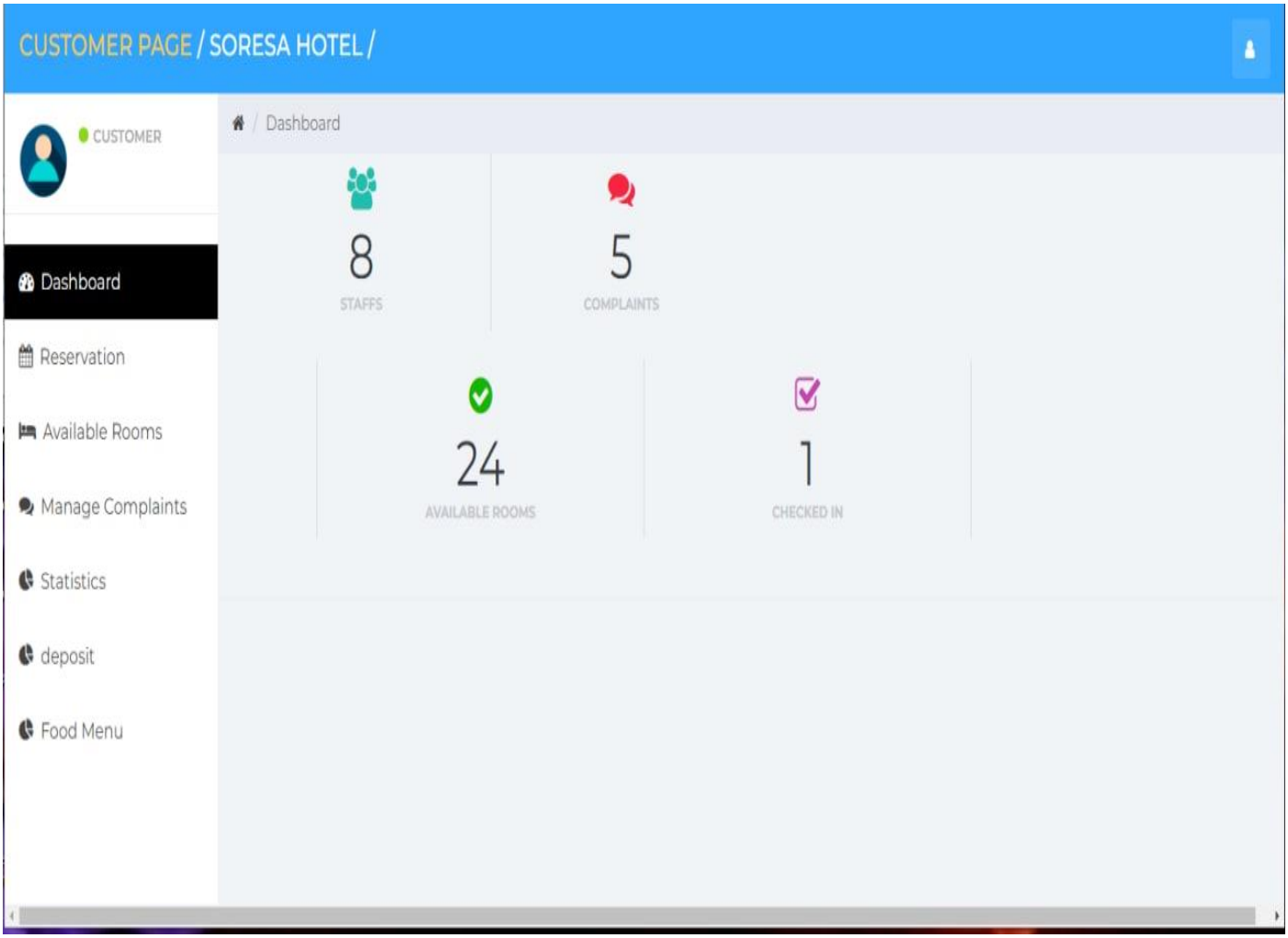

Figure 6. 2 Customer page

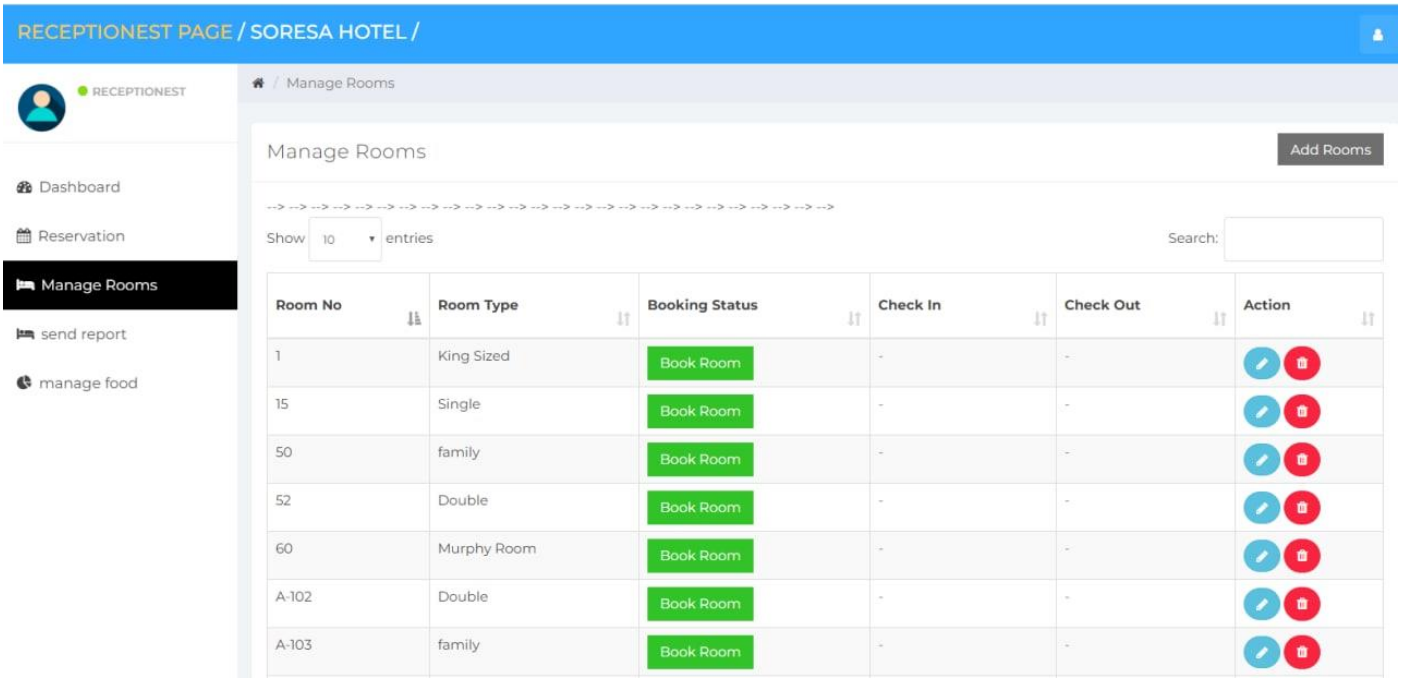

Figure 6. 3 Receptionist page

6.6 Testing

Testing evaluates a software product to ensure that it satisfies its planned purpose. A test that is modified to and consistent with development methodologies provides an observable and structured approach to verifying requirements and quantifiable performance

6.6.1 Testing criteria

Criteria are standards by which we evaluate our systems that help us to determine whether a test case passes or fail.

- **Fail Criteria:** when the system does not meet the all specific requirements of the

System and if the test case is said to fail the expected result is not satisfied by the

System that relates with its functionality.

- **Pass Criteria**: when the system meets the all specific requirements of the system and if the test case is said to pass the expected result is satisfied by the system that relates with its functionality

6.6.2 Test case

Is a set of actions executed to verify a particular feature or functionality of your software applications?

Test Case 1: Check result on entering Valid User ID & Password

Test Case 2: Check result on entering Invalid User ID& Password

Test Case 3: Check response when a User ID is empty & Login Button is pressed, and many more

A test case is a set of conditions or variables under which a tester will determine

Whether a system under test satisfies requirements or works correctly

Input	**Expected Result**	**Actual outcome**	**Pass/Fail**
Valid User Name and Valid Password	The user logs to the system successfully	The user logs to the system successfully	Pass
If inserted only user name, Without Password	The system displays an error message “please Fill out this field”	The system displays an error message “please Fill out this field”	Fail

If the user name or password Text Box or both are not filled	The system displays an error message "please fill user name and password"	The system displays an error message "please fill user name and password"	Fail
If the valid user name and invalid password	The system displays"User name or password doesn't match"	The system displays"User name or password doesn't match"	Fail

6.6.3 Unit testing

The main objective of unit testing to ensure that each individual part is working well and as it's supposed to work. The entire system will only be able to work well if the Individual part are working well. Unit testing is preformed by software developers themselves. In this level of testing process, the OHR system developers test different sub procedures, functions and tested by applying the black and white box testing.

- ✓ Cheek whether the return type of the function is correct.
- ✓ Cheek how the sub procedure or function are call correctly.
- ✓ Cheek if the correct output is produced for different inputs

6.6.4 System Testing

The test performs on the whole programs to ensure the whole functional and non-functional requirement specified on the system and also to decide the completeness of the system as a system that full fill perfectly the whole functional and non-functional requirements of the system

Since we have tested the overall features of our systems and we have conducted testing activities to check if there are errors as well as to correct them as not to occur after or later. The goal of the system testing phase is to evaluate all what we have been implemented in the previous development phase and to find errors in order to make corrections on the errors and these can not occur again

6.6.5 Integration Testing

As the name of the test indicates that the test performs to ensure the integration of the user interface module that interact the user with the system and the database module that store the transaction of the hotel and then in order to make a system as complete system the two module must be integrate to each other

6.6.6 Performance testing

The test performs to ensure the security of the system as a whole and also the test will be done by using black box testing methods that examines the functionalities of the application by focus only on input and expected output.

CHAPTER SEVEN

7. CONCLUSION AND RECOMMENDATION

7.1 Conclusion

The main purpose of this project is to establish a long lasting and effective communication between different users and also to introduce various features regarding Soresa Hotel management system. An effort has been made to study Soresa Hotel management system as partial fulfillment of BSC degree in Information systems. In doing the study the team has tried to follow object oriented system analysis and design methodology. Since the success and failure of any system depends on gathering the right information through different fact-finding techniques and user involvements, the team has made the best effort to gather requirements. After a detail review and study of the existing system of Soresa Hotel management system models have been designed to reflect the new system that are suppose to solve problems. In order to solve different problems existed the team has tried to propose a solution that at least reduce the existed problems and model the proposed system using different tools and methodologies. The team believe the different tools and techniques has helped us a lot in capturing real user requirements and model the right system for the users for their day to day transactions. Thus it should have the precedence in know-how and experience in collecting, processing and utilizing information. This project also focuses on online access to available information, order food, book a bed, make payments, and locate the hotel.

7.2 Recommendation

Since we are now living in a world that is led by technology and technology results, we need more and more applications to familiarize ourselves and also come up with the fast advancing technology. Thus, as we are beginner website developers, we recommend that other website designers, beginners or professionals, to create more dynamic pages that are very user friendly, more secure and also introduce the community as a whole to be familiar with the current technology. Online payment is difficult to the security purpose; we hope that this project will create some initiation for those people who wants to develop online payment.

REFERENCE

[1] M. Martin, "https://www.guru99.com/functional-requirement-specification-example.html.," guru99,, 18 December 2018.. [Online]. [Accessed [Accessed 2 February 2022].].

[2] Scot.W.Ambler, The Object Primer, 3rd Edition ed., Cambridge:: Camridge: University University Press, 2004.

[3] S. W. Ambler, The object primer,3rd Edition ed.,, Cambrigidge:: Cambrigidge University press,, 2004.

[4] D. B. a. G. W. Scragg, The Object Primer,, Cambridge:: Cambridge University Press,, 2004..

Appendix

Appendix I

Interview

Unstructured Interview

1. How the allover work is managed currently in this organization?
2. What are the limitations of the current hotel works?
3. Is there a central body that is responsible for hotel works?
4. How do you control the flow of information?
5. How do the employees generate reports?
6. How do they follow the problem in this hotel is solved?
7. How the owner of this hotel takes feedback whether the employee is safisfied or not?