Daffodil International University

Assignment

Course Name: Object Oriented Software Development Lab.

Course code: SWE-331

Final Document

Submitted By

Shoumen Mondal

ID:151-35-853

Sec:B

Submitted To

M.D.Alamgir Kabir

Lecturer

Department of Software Engineering

Daffodil International University

[ROOM MANAGEMENT APPLIENCE SYSTEM]

[Web application of room management system]

ABSTRACT

[This the system for managing shared office ,meeting, laboratory and teaching space is to be developed .This system is web based scheduling system but the status of each room can be checked at location.]

Shoumen mondal

[Object Oriented Software Development With Lab]

Acknowledgement

The outcome of this project required a lot of guidance and assistance from many people and we are extremely fortunate to have got this all along the completion of our project work. Whatever we have done is only due to such guidance and assistance and we could not forget to thank them. We convey our respect and gratitude to Mr. MD . Alamgir Kabir for giving us an opportunity to do this project and providing us all support and guidance which made us complete the project on time, we are extremely grateful to him for providing such a nice support and guidance.

We are also very grateful to our department's head Dr Touhid Bhuiyan for the resources he provided us. Without this we couldn't complete this.

Table of Content

Table of Figure

Chapter 1
Introduction

1.1 About the System

System Overview:

A system for managing shared office, meeting, laboratory, and teaching space is to be developed. A networked display appliance is situated outside each room. The appliance indicates who has reserved the space and for what period. The appliance allows users to reserve a room and check the availability of other rooms.

The system is a web based scheduling system but the status of each room can be checked at location. It incorporates the use of small flat panel displays to eliminate scheduling conflicts, interruptions, and encourage better space utilization.

The system can:

- Reserve the appropriate room when needed from any location on the network
- Display meeting information outside each room
- Easily extend room reservation with the touch screen
- Release a room if meeting finishes early
- Quickly identify occupied and available rooms by a red or green light
- Grab a room with the touch screen for an uninterrupted impromptu meeting

Web interface

All the room displays are networked so then can be accessed via a web interface (a single URL). Every room's schedule can then be viewed from the web interface and maintained by an administrator. A user should be able to check the availability of any room from the room display and the web interface. The web interface allows for reserving rooms along with searching tools to find available rooms at given times. All significant management of the building's space information will be done through this interface.

1.2 Purpose

The system can:

- Reserve the appropriate room when needed from any location on the network
- Display meeting information outside each room
- Easily extend room reservation with the touch screen
- Release a room if meeting finishes early
- Quickly identify occupied and available rooms by a red or green light
- Grab a room with the touch screen for an uninterrupted impromptu meeting.

1.3 Scope

In this system the registered user confirm about their room reservation with time. They can reserve the existing room for their purpose by maintaining the system. They have to login by their account and they can also cancel their reservation but after payment they are not be able to cancel reservation. This system such a great dill for them who need the meeting room.

1.4 Vision

For reserving a room we have to going or contact here and there for getting a room. But maintaining this system we have not to go anywhere for meeting room. This system will be manage the room for appliance by online payment.

1.5 Why this System Necessary

This system is necessary because this system is important for the user. Now all the system is web base so this system is web base system. For this Reason minimize the collision of room time and schedule or reservation for meeting or teaching in a room.

1.6 Proposed Solution

- Reservation with touch screen
- Identified room by using green or red light
- Check availability with view web interface

Chapter 2
System Analysis and Design

2.1 Use case model

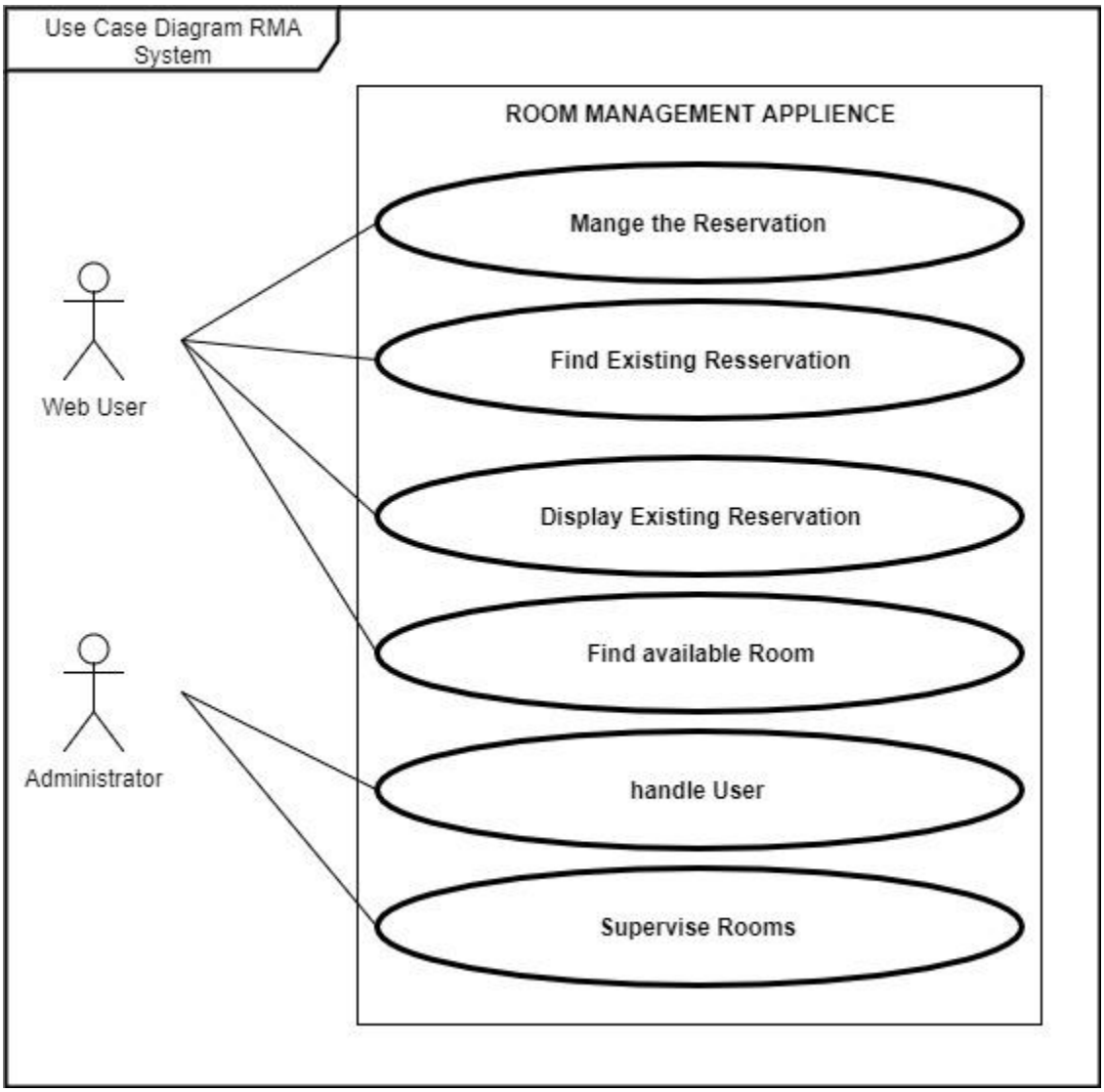

Figure 1: Use Case Model

2.2 Use Case Description (Briefly)

2.2.1 Description of Use case Manage The Reservation.

<table>
<tr><td>Scope</td><td>Management the all Reservation.</td></tr>
<tr><td>Level</td><td>User level</td></tr>
<tr><td>Primary Actor</td><td>Web User</td></tr>
<tr><td>Precondition</td><td>User must login first. And Verify the authentication</td></tr>
<tr><td>Post condition</td><td>Person first & last names, address, city, state, phone number must be recorded.</td></tr>
<tr><td>Main flow</td><td>
<table>
<tr><th>Actor</th><th>System</th></tr>
<tr><td>1. Get log in after registration. Get (user name, password) for log in in system.
2. Get entry information after collection of person information or entries. Entry Info (Person first and last name, address, city, state , phone number).</td><td>1.1. System returns log in successfully.

2.1. The system return entry information successfully.</td></tr>
</table>
</td></tr>
<tr><td>Alternative scenario</td><td>
<table>
<tr><th>Actor</th><th>System</th></tr>
<tr><td>1. Get log in after registration. Get (user name, password) for log in in system.</td><td>1.1. System returns wrong user name.</td></tr>
</table>
</td></tr>
</table>

2.2.2 Description of Use case Find Existing Reservation.

<table>
<tr><td>Scope</td><td>Finding all the existing Reservation</td></tr>
<tr><td>Level</td><td>User level</td></tr>
<tr><td>Primary Actor</td><td>Web User</td></tr>
<tr><td>Precondition</td><td>User must Reservation first. And finding the existing reservation.</td></tr>
<tr><td>Post condition</td><td>After reservation Person get Reservation details. Then the person existing the reservation the room.</td></tr>
<tr><td>Main flow</td><td>
<table>
<tr><th>Actor</th><th>System</th></tr>
<tr><td>1. Access the system.
2. Reserve the room(input reservation details)</td><td>1.1 System returns log in successfully.

1.2 The system return entry information successfully.</td></tr>
</table>
</td></tr>
</table>

Alternative scenario	<table><tr><th>Actor</th><th>System</th></tr><tr><td>3. If there are not Get reservation details then the person existing the room</td><td>3.1 System returns existing reservation details.</td></tr></table>

2.2.3 Description of Use case Display the existing Reservation.

Scope	Display the all Existing Reservation.
Level	User level
Primary Actor	Web User
Precondition	For display exiting reservation must be reserve the room
Post condition	Person first & last names, address, city, state, phone, number room number must be recorded.
Main flow	<table><tr><th>Actor</th><th>System</th></tr><tr><td>1. Get access the system after login and display all the reservation. 2. Find available room for reserve room.</td><td>1.1. System returns Reservation successfully. 2.1 The system return the all available room for reservation.</td></tr></table>
Alternative scenario	<table><tr><th>Actor</th><th>System</th></tr><tr><td>3. If All reservation is not display.</td><td>3.1. System returns Display all the existing reservation.</td></tr></table>

2.2.4 Description of Use case Find Available Room

Scope	Finding the available room
Level	User level
Primary Actor	Web user
Precondition	User must login first.

<table>
<tr><td>Post condition</td><td>Person first & last names, address, city, state, phone number must be recorded.</td></tr>
<tr><td>Main flow</td><td>
<table>
<tr><td>Actor</td><td>System</td></tr>
<tr><td>4. Get log in after registration. Get (user name, password) for log in in system.
5. Get entry information after collection of person information or entries. entry Info (person first and last name, address, city, state , phone number).</td><td>1.2. System returns log in successfully.
a. The system return entry information successfully.</td></tr>
</table>
</td></tr>
<tr><td>Alternative scenario</td><td>
<table>
<tr><td>Actor</td><td>System</td></tr>
<tr><td>6. If no available room</td><td>a. System returns not available room.</td></tr>
</table>
</td></tr>
</table>

2.2.5 Description of use case Handle User.

Scope	Handle all the Users.
Level	Administrator level.
Primary Actor	Administrator.
Precondition	User must login first.
Post condition	Person first & last names, address, city, state, phone number must be recorded.

<table>
<tr><td>Main flow</td><td>
<table>
<tr><th>Actor</th><th>System</th></tr>
<tr><td>7. Get log in after registration. Get (user name, password) for log in in system.
8. Get entry information after collection of person information or entries. entry Info (person first and last name, address, city, state , phone number).</td><td>1.3. System returns log in successfully.
a. The system return entry information successfully.</td></tr>
</table>
</td></tr>
<tr><td>Alternative scenario</td><td>
<table>
<tr><th>Actor</th><th>System</th></tr>
<tr><td>9. Get log in after registration. Get (user name, password) for log in in system.</td><td>a. System returns wrong user name.</td></tr>
</table>
</td></tr>
</table>

2.2.6 Description of Use case Supervise the Room.

Scope	Supervising all the room
Level	Administrator level.
Primary Actor	Administrator.
Precondition	User must login first.
Post condition	Person first & last names, address, city, state, phone number must be recorded.

<table>
<tr><td>Main flow</td><td>
<table>
<tr><td>Actor</td><td>System</td></tr>
<tr><td>10. Input room Details. Get (number , id ,location) for log in in system.
11. Get entry information after collection of room information or entries.</td><td>1.4. System returns log in successfully.

a. The system return entry information successfully.</td></tr>
</table>
</td></tr>
<tr><td>Alternative scenario</td><td>
<table>
<tr><td>Actor</td><td>System</td></tr>
<tr><td>Not input room details.</td><td>b. System returns no available room.</td></tr>
</table>
</td></tr>
</table>

2.3 Use Case Description (Detailed)

2.3.1 Use Case of Web User

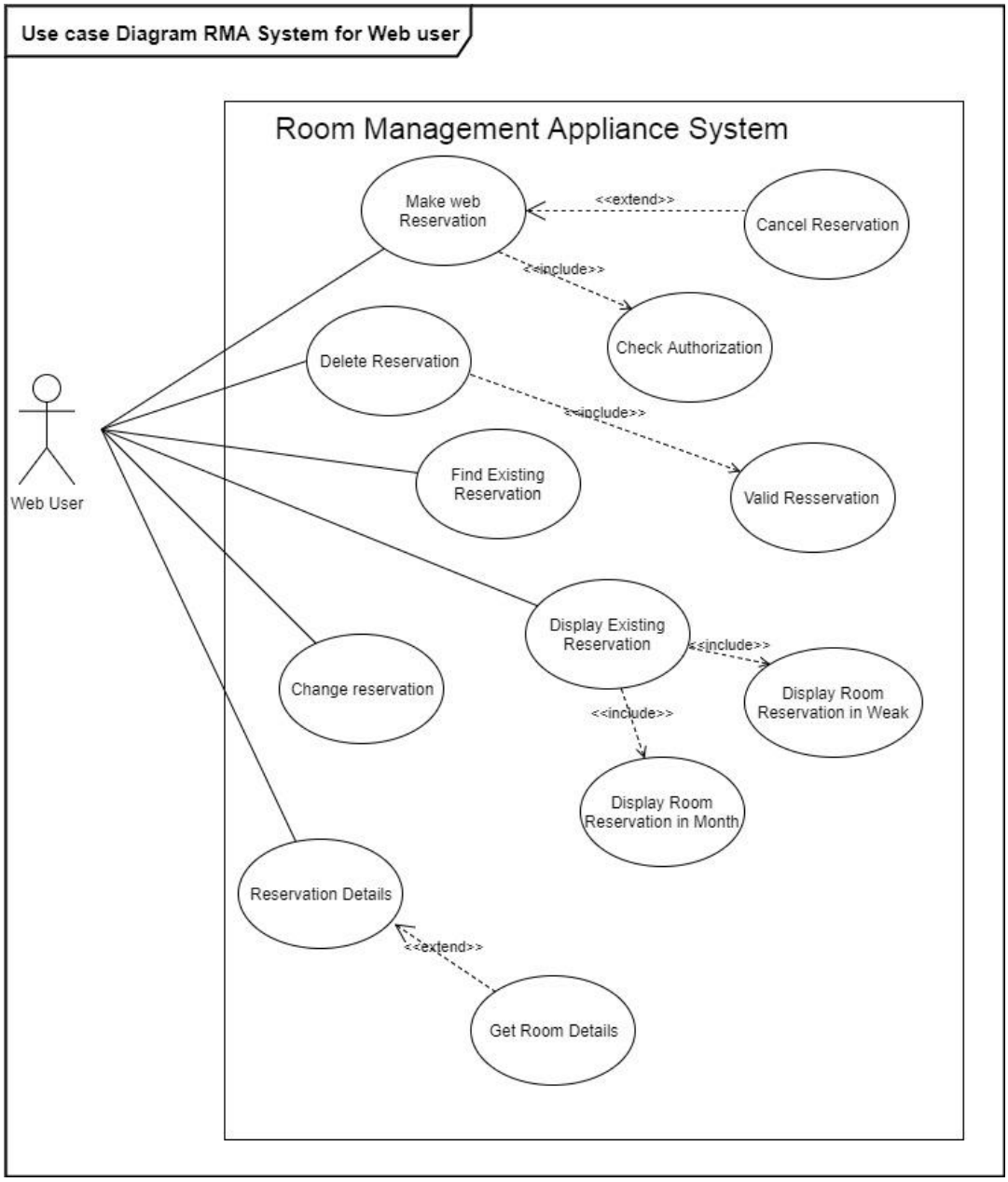

Figure 2: Use Case Model Web User

2.3.2 Use Case of Administrator

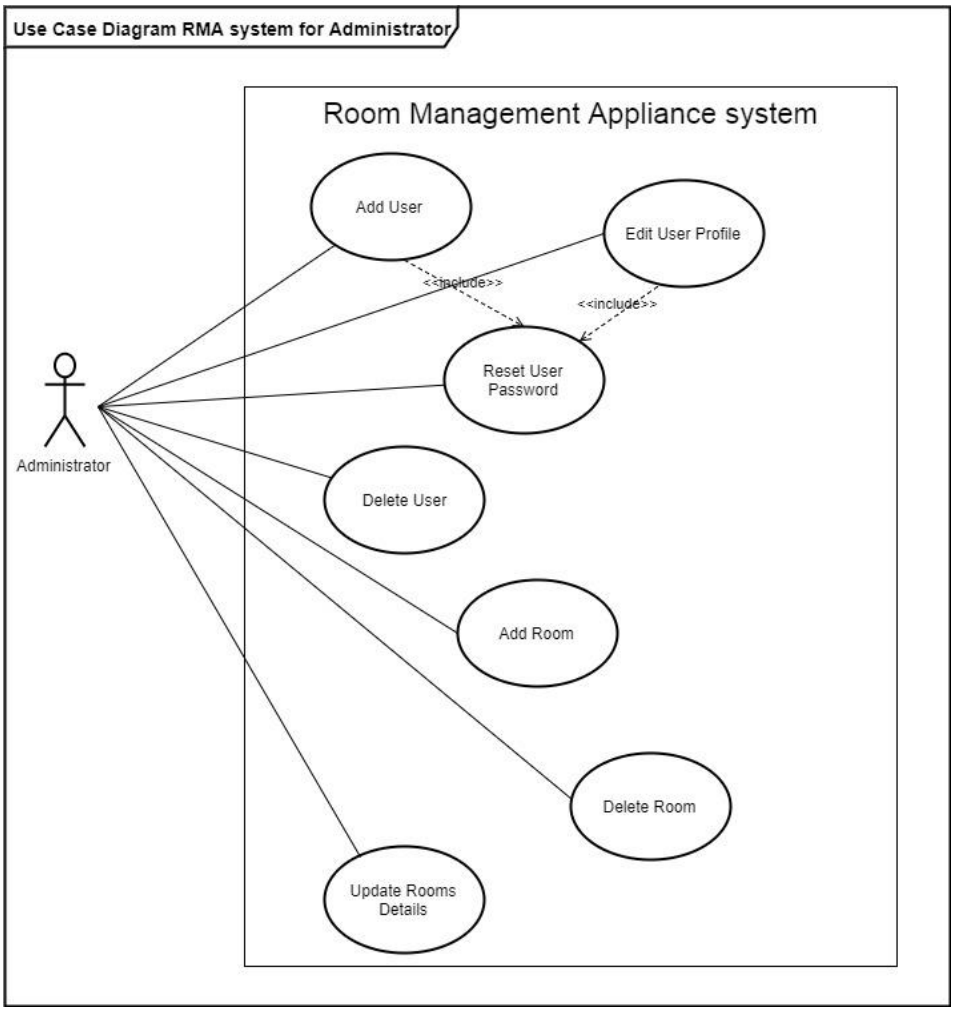

Figure 3: Use Case Model of administrator

2.5 System Sequence Diagram

2.4.1 System Sequence of manage Reservation

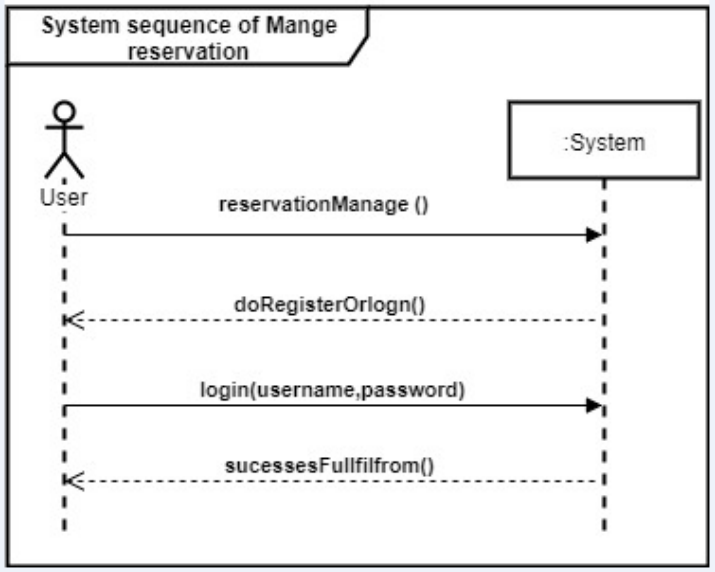

Figure 4: System sequence Diagram manage Reservation

2.4.2 System Sequence of Existing reservation

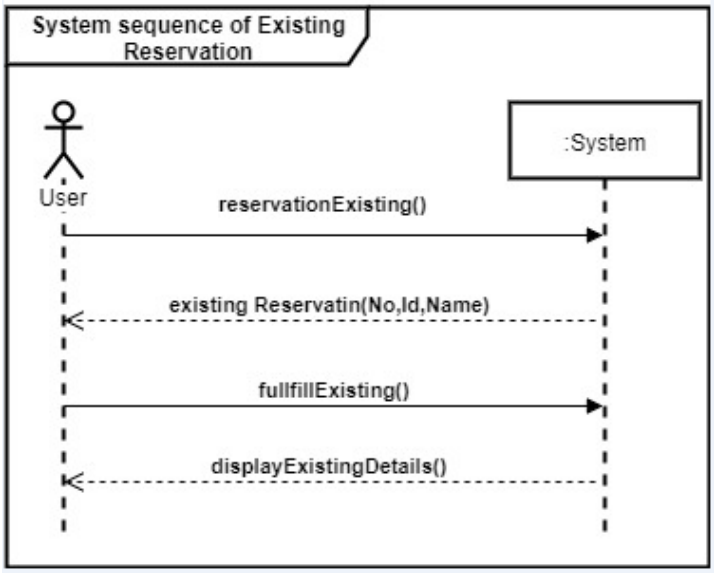

Figure 5: System sequence Diagram Existing Reservation

2.4.3 System Sequence of Display reservation

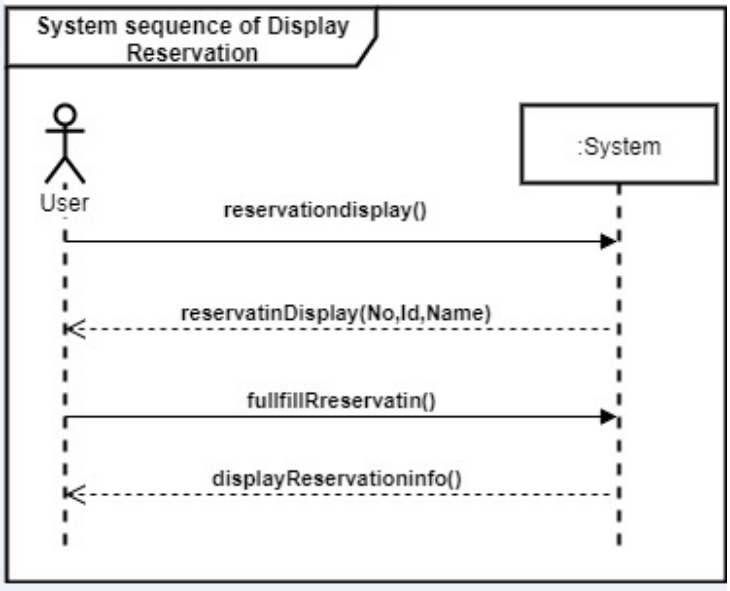

Figure 6: System sequence Diagram Display Reservation

2.4.4 System Sequence of Available Room

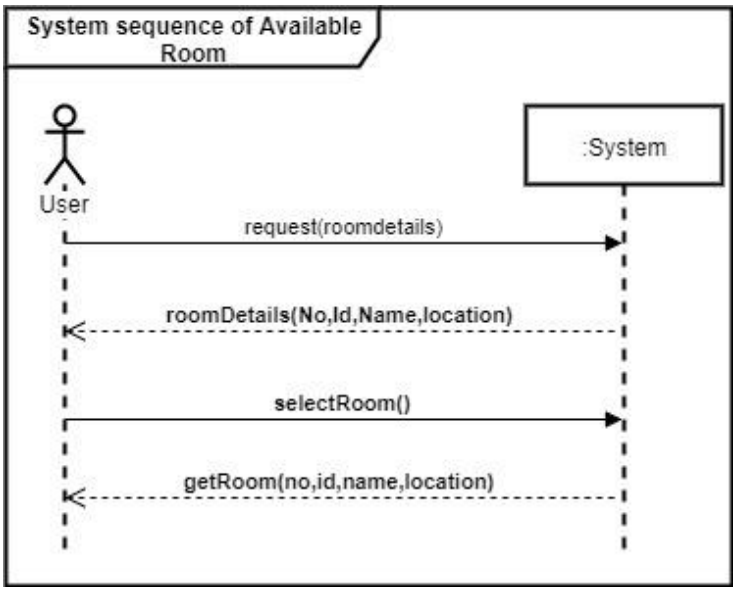

Figure 7: System sequence Diagram Available Room

2.4.5 System Sequence of Handle User

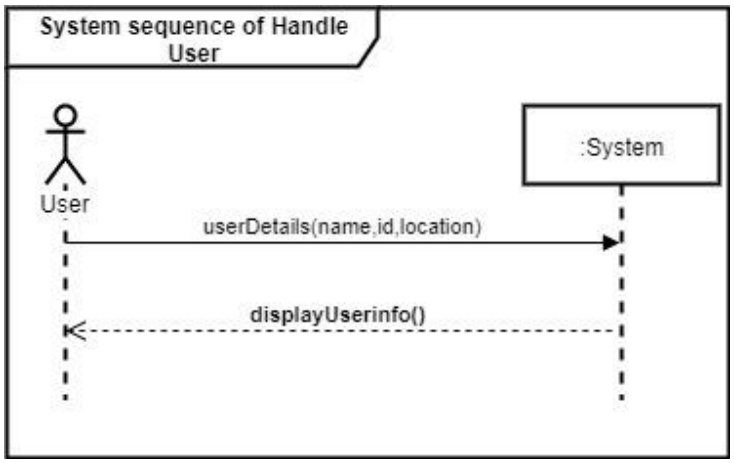

Figure 8: System sequence Diagram handle user

2.4.6 System Sequence of supervisor room

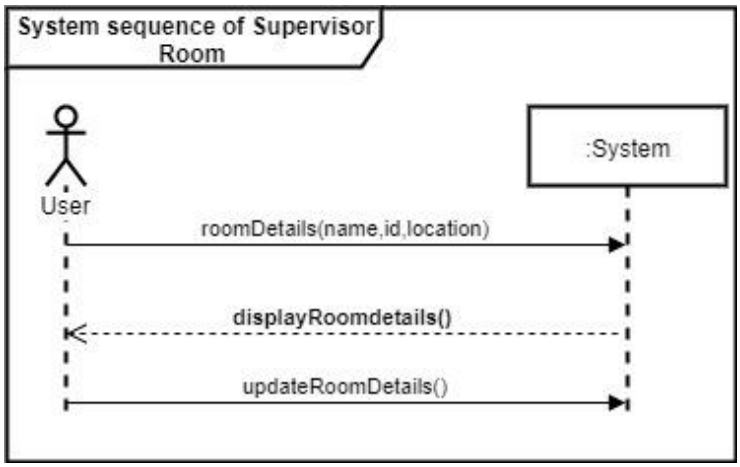

Figure 9: System sequence Diagram Supervise Room

2.5 Domain Model/Conceptual Model

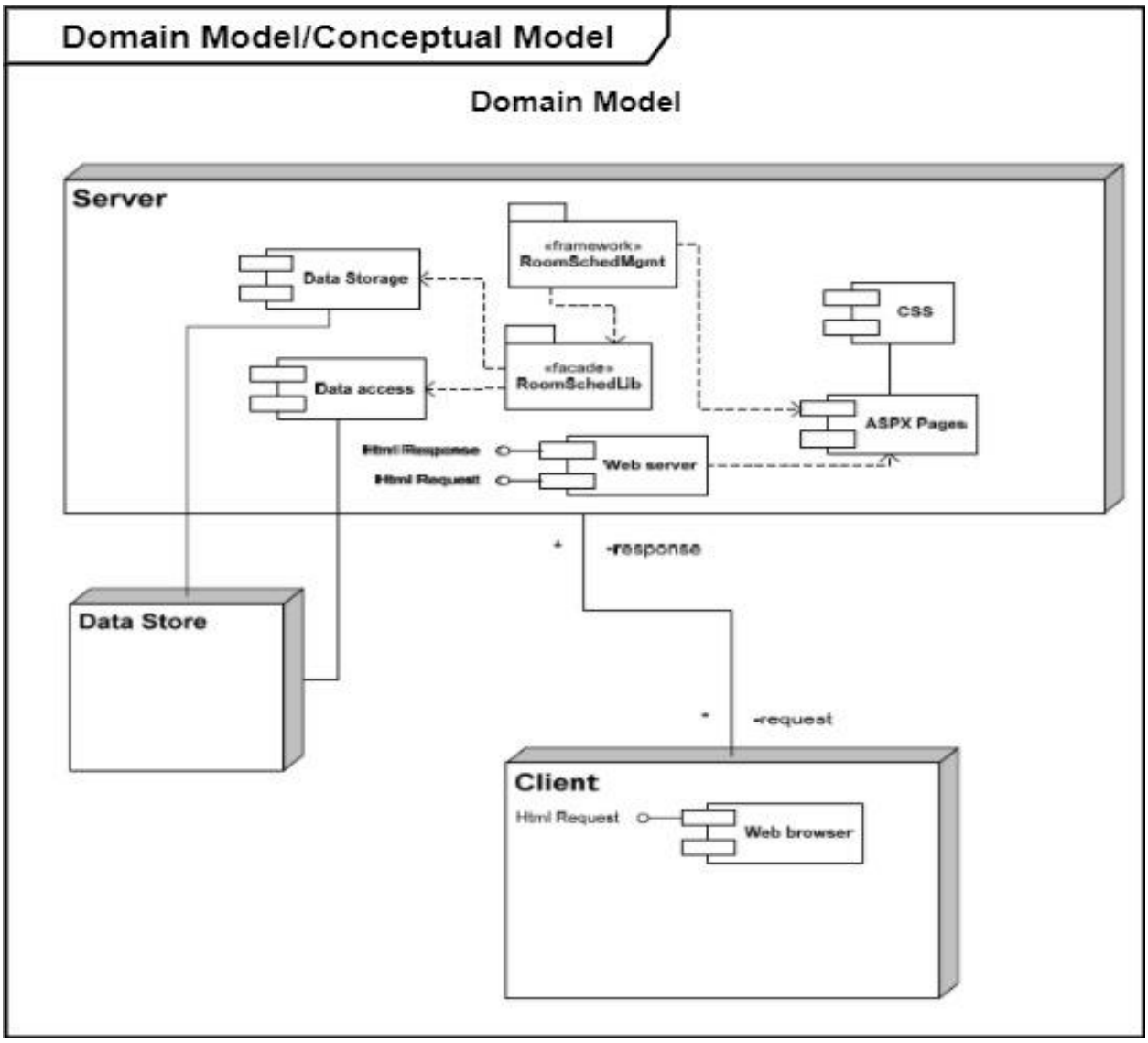

Figure 10: Domain Model

2.6 Activity Diagram

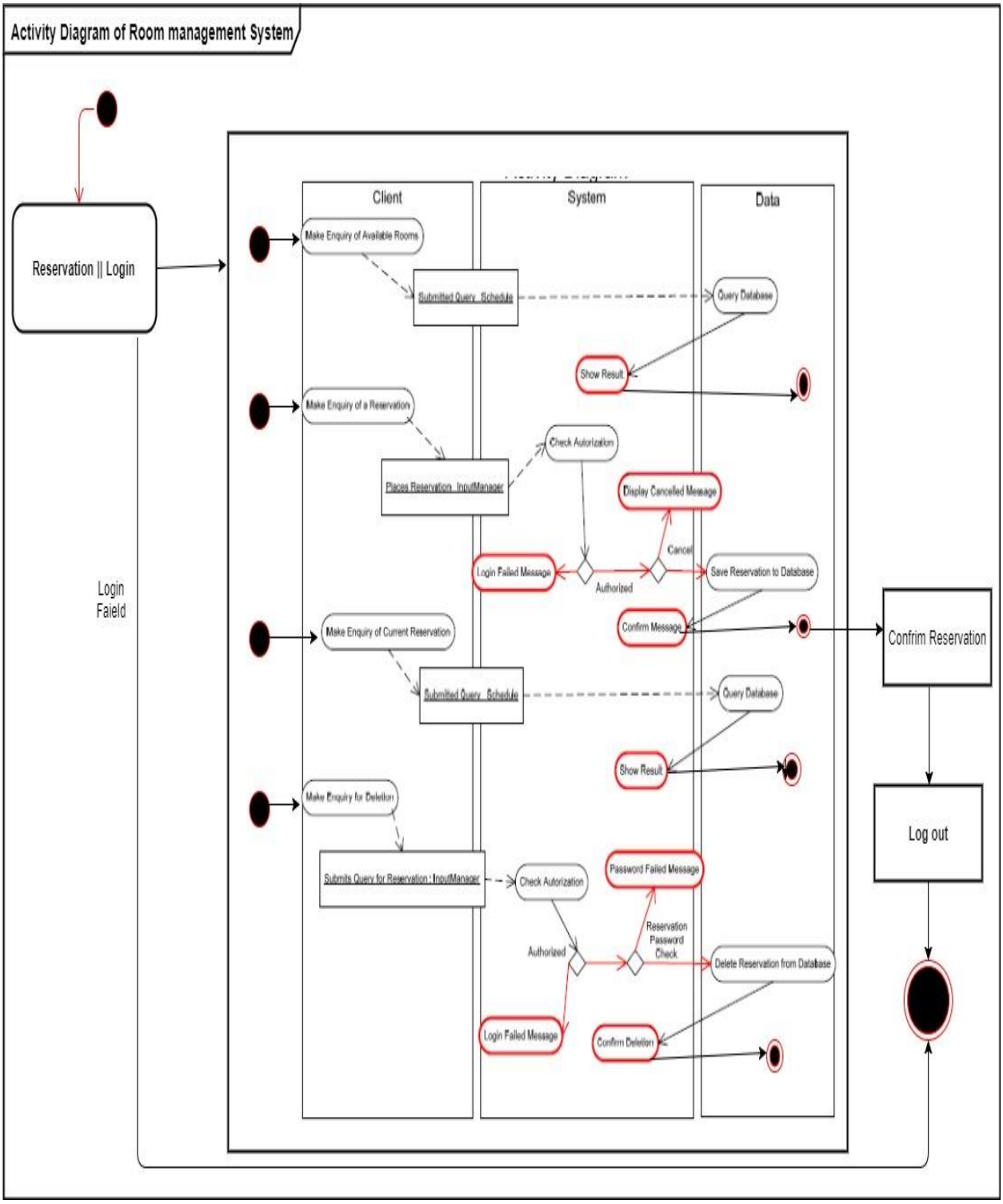

Figure 11: Activity Diagram

Chapter 3
System Design

3.1 Sequence Diagram

3.1.1 Sequence of Manage the Reservation

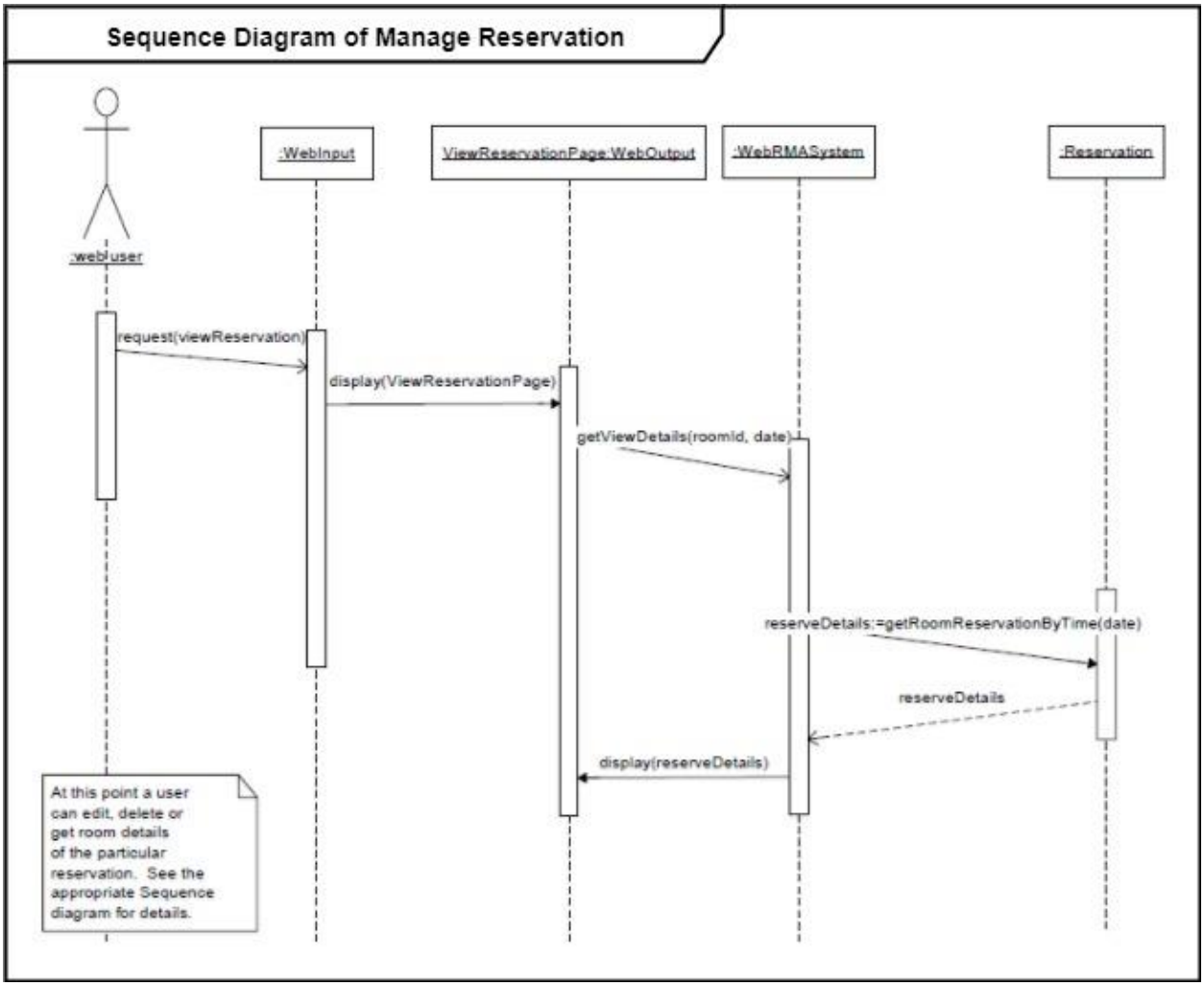

Figure 12: Sequence of Manage the Reservation

3.1.2 System Sequence of Find Existing reservation

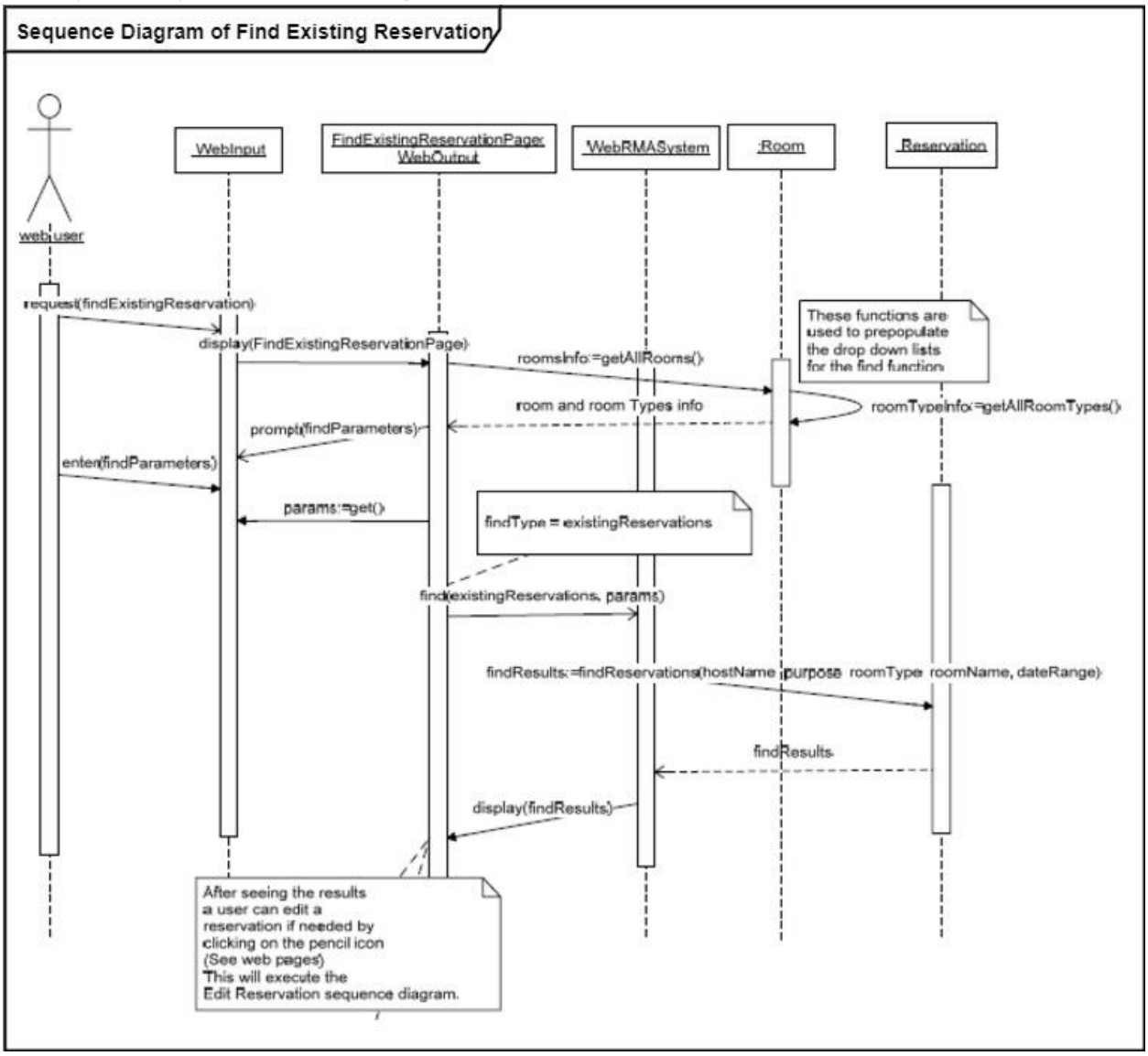

Figure 13: Sequence of Find Existing Room

3.1.3 System Sequence of Display Existing reservation

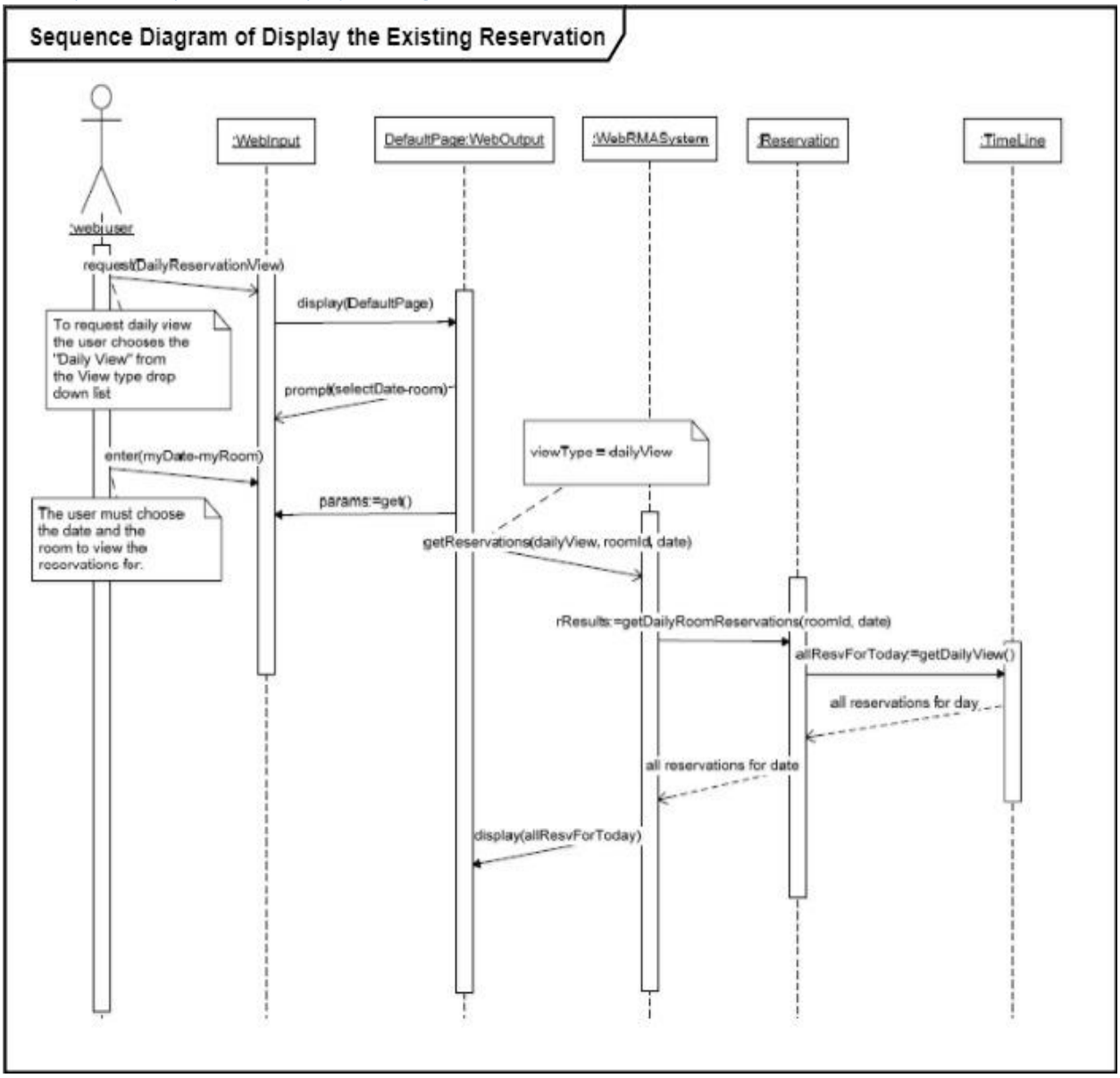

Figure 14: Sequence of Display Existing Room

3.1.4 System Sequence of Find available Room

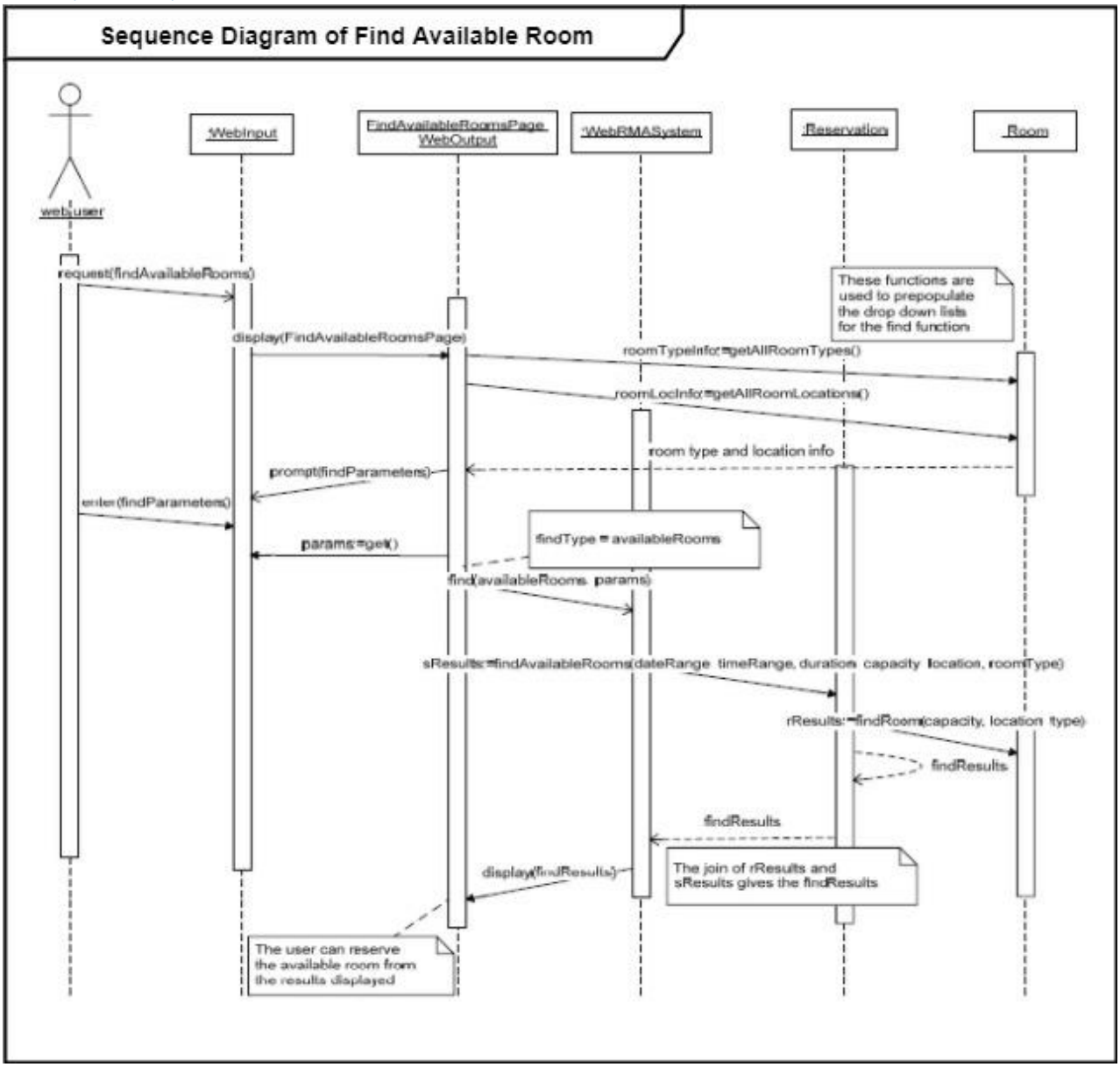

Figure 15: Sequence of Find available Room

3.1.5 System Sequence of Handle the User

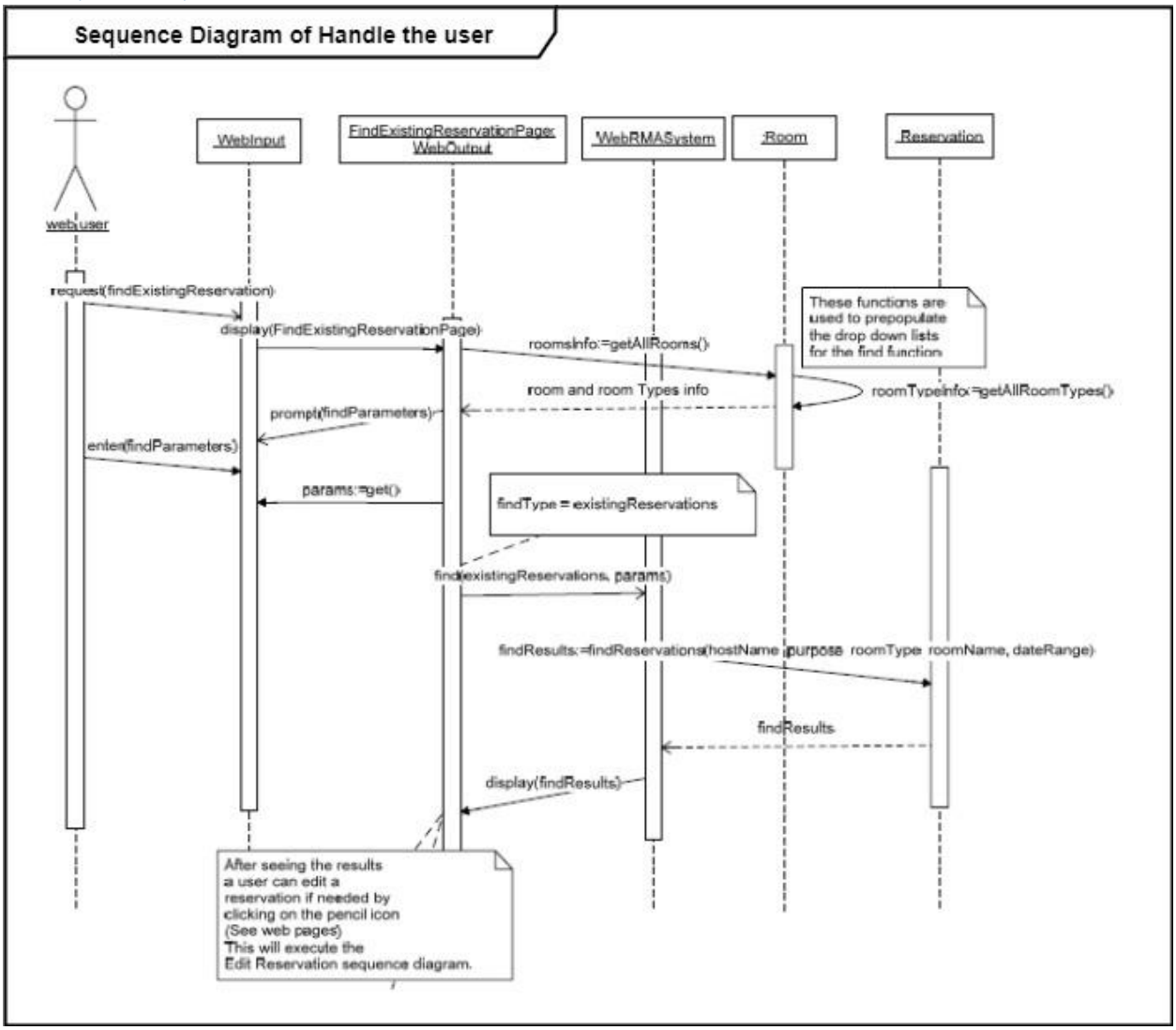

Figure 16: Sequence of Handle user

3.1.6 System Sequence of Supervisor The Room

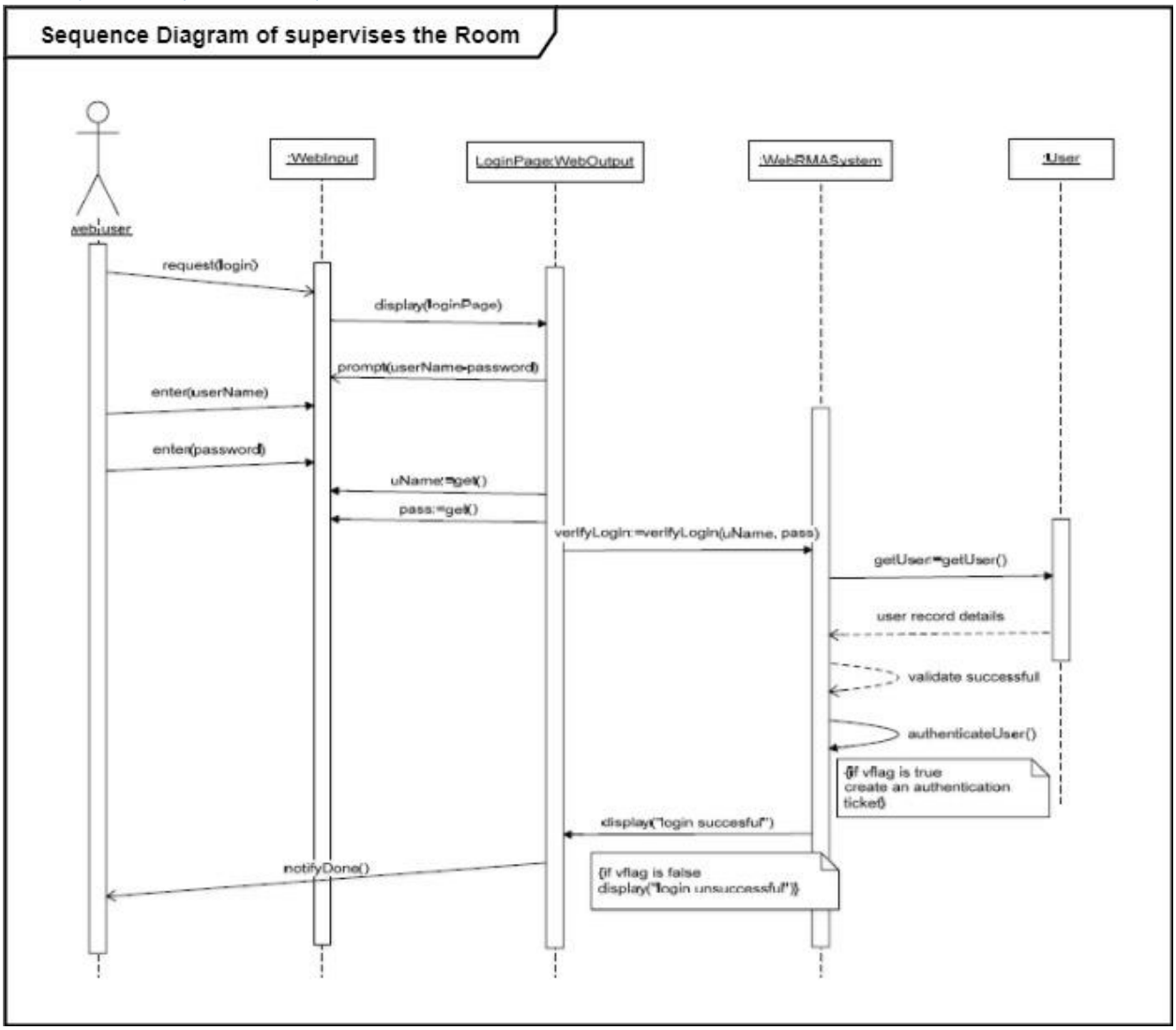

Figure 17: Sequence of Supervise Room

3.2 Class diagram

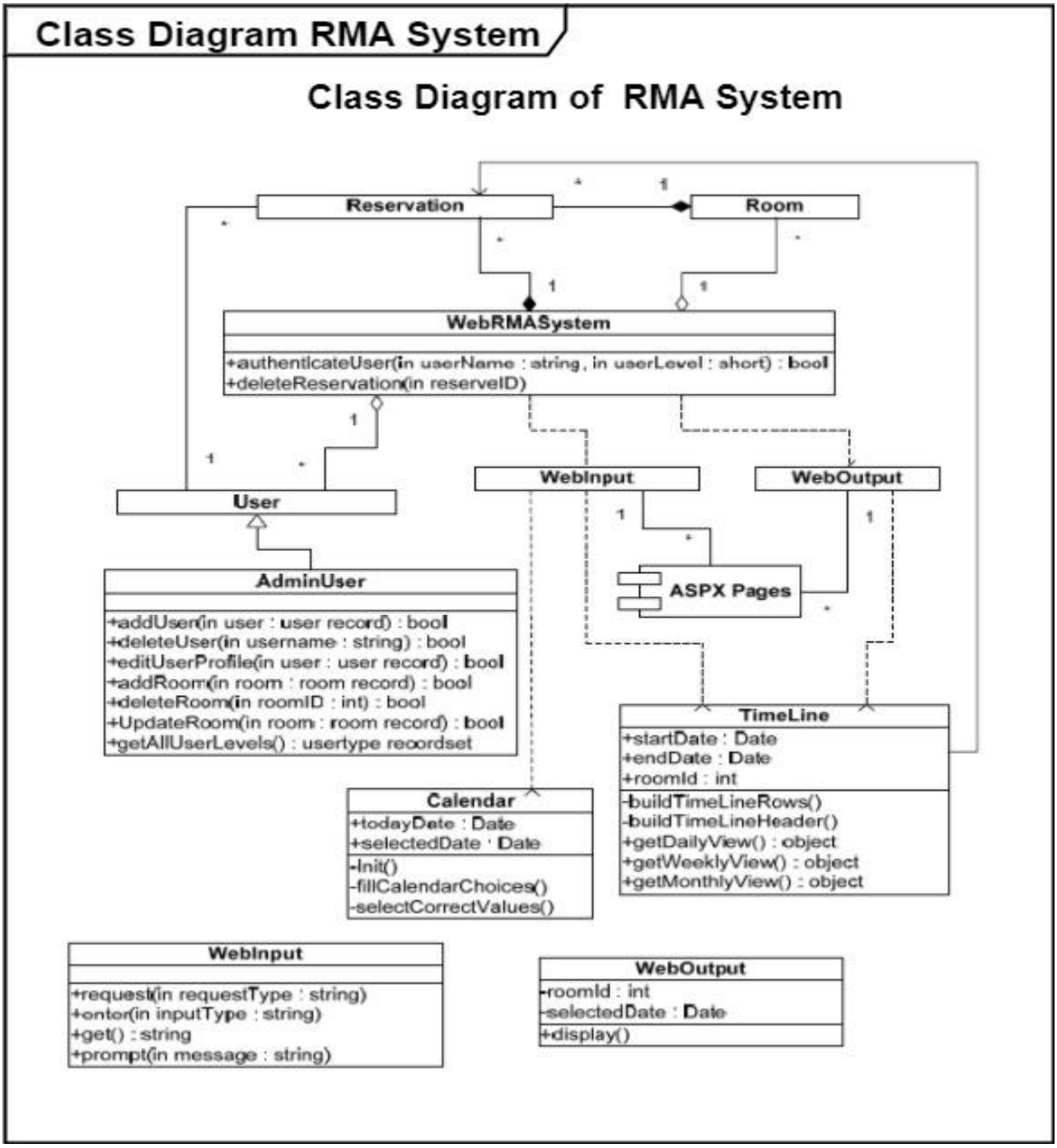

Figure 18: Class Diagram

3.2 Entity Relationship Diagram

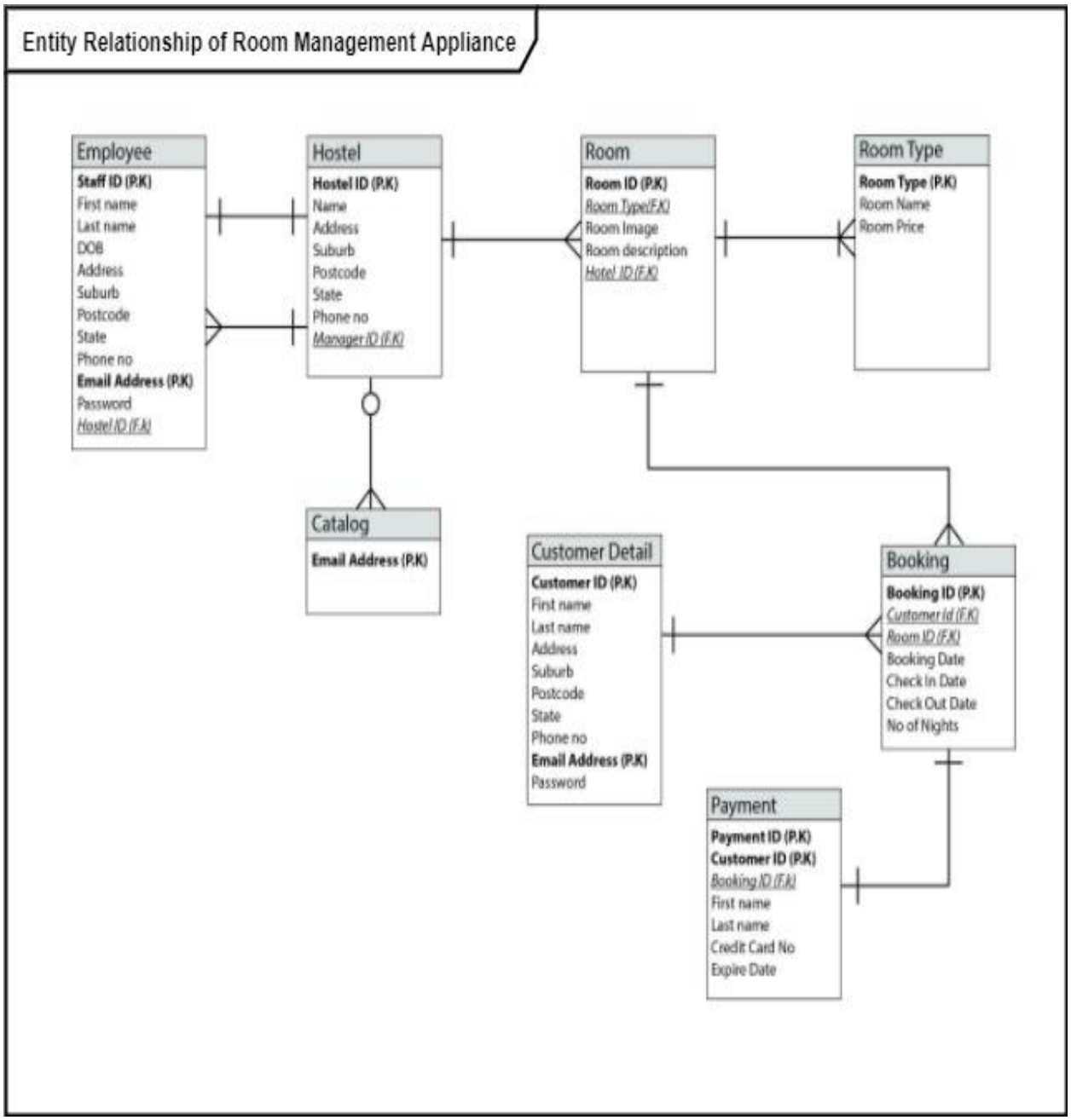

Figure 19: Entity Relationship Model

Chapter 4
Implementation

4.1 Tools and Technologies

Following are the tools and technologies used in development of this project:

Microsoft Visual Studio 2012

ASP.NET Framework

Microsoft SQL Server 2008

Telluric reports

Microsoft Visio

Microsoft Architect

HTML5, CSS, JavaScript, J-query, Twitter bootstrap