Software Requirements Specification

Version 3.0

3 April 2016

Guest House Management System

Submitted in partial fulfillment

Of the requirements of

CS 223 Software Engineering

This work is based upon the submissions of the course Software Engineering (CS223). The students who submitted this team projects were:

Nithin V ug201310022

Kaustubh Kumar Pandey ug201310042

Vivek Laata ug201310041

Upendra Singh Chauhan ug201310038

# Table of Contents

[Table of Contents i](#_Toc440664050)

[List of Figures ii](#_Toc440664051)

[1.0. Introduction 1](#_Toc440664052)

[1.1. Purpose 1](#_Toc440664053)

[1.2. Scope of Project 1](#_Toc440664054)

[1.3 Constraints 1](#_Toc440664055)

[1.4 Assumptions and Dependencies 2](#_Toc440664056)

[1.3. Glossary 2](#_Toc440664057)

[1.4. References 2](#_Toc440664058)

[1.5. Overview of Document 2](#_Toc440664059)

[2.0. Overall Description 3](#_Toc440664060)

[2.1 System Environment 3](#_Toc440664061)

[2.2 Functional Requirements Specification 3](#_Toc440664062)

[2.2.1 Use case 1 3](#_Toc440664063)

[Use case: 3](#_Toc440664064)

[2.3 User Characteristics 3](#_Toc440664065)

[2.4 Non-Functional Requirements 3](#_Toc440664066)

[3.0. Requirements Specification 4](#_Toc440664067)

[3.1 Functional Requirements 4](#_Toc440664068)

[3.1.1 Logging in and authentification 4](#_Toc440664069)

[3.1.2 Requesting for a room available……………………………………………………………...3](#_Toc440664070)

[3.1.3 Booking a room 4](#_Toc440664070)

3.1.4 Billing………………………………………………………………………………………...3

[3.3 Detailed Non-Functional Requirements 5](#_Toc440664071)

[***3.4*** ***Logical Structure of the Data*** 5](#_Toc440664072)

[4.0 Supporting information 5](#_Toc440664073)

[4.1 Table of contents and index 5](#_Toc440664074)

[4.2 Appendixes 5](#_Toc440664075)

# List of Figures

**No table of figures entries found.**

# 1.0. Introduction

## 1.1. Purpose

The purpose of this document is to present a complete description of the Guest House Management System and an overview of our software , its uses , users , goal and its scalability, to be built for any academic institution’s online guest house booking, allotment and billing portal for aiding the institution to manage these specified tasks. The document describes the software and hardware requirements, the features that are included and the user interface .Its purpose is to help the designer and the developer in the process of software development.

## 1.2. Scope of Project

* Guest house management system is basically a portal that would be used for booking Guest house in any Academic Institute through both online and offline means.
* The users of this software will include the registered members of the institute who are the students, faculty and staff. It can also be used by an outsider who wish to book a guest house in the institute.
* The software will be used and maintained by an admin who receives all the requests for booking a guest house. He checks for any clashes (the registered members have more priority than guests) and allocates the rooms accordingly.

## 1.3 Constraints

1. The booking will not be confirmed immediately at the time of request.

2. Only the Admin will be able to see the Requests.

3. A reference of any registered member has to be provided.

4. Type of rooms are AC, non-AC, Deluxe AC, and Deluxe non-AC.

## 1.4 Assumptions and Dependencies

1. Anyone with a proper reference can request for the room in the guest house.

2. Any interrupt or failure during the process of booking, it has to roll back.

3. The features inside the room like TVs and fridge are filled into the database by the Admin.

4. Maximum duration for which a guest can stay is decided by the Admin.

5. The rooms can be given to special Interns under the permission of the faculty mentor.

6. The specified details in the form of database, for example registered user, are available before hand.

## 1.3. Glossary

|  |  |
| --- | --- |
| **Term** | **Definition** |
| Registered member | A student, staff or faculty |
| Admin | A person having access of guest house Database |
| Staff | A person registered via institute ID as an employee. |
| Faculty | A staff member who is engaged in teaching. |
| Student | A person registered via institute ID and hostel. |
| Reference | An ID and password of a registered member. |

## 1.4. References

IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications.* IEEE Computer Society, 1998.

## 1.5. Overview of Document

The rest of the document is designed in the following way:

# 2.0. Overall Description

## 2.1 System Environment

<< Keep blank for the time being >>

## 2.2 Functional Requirements Specification

. << Keep blank for the time being >>

### 2.2.1 Use case

#### Use case:

**Diagram:**

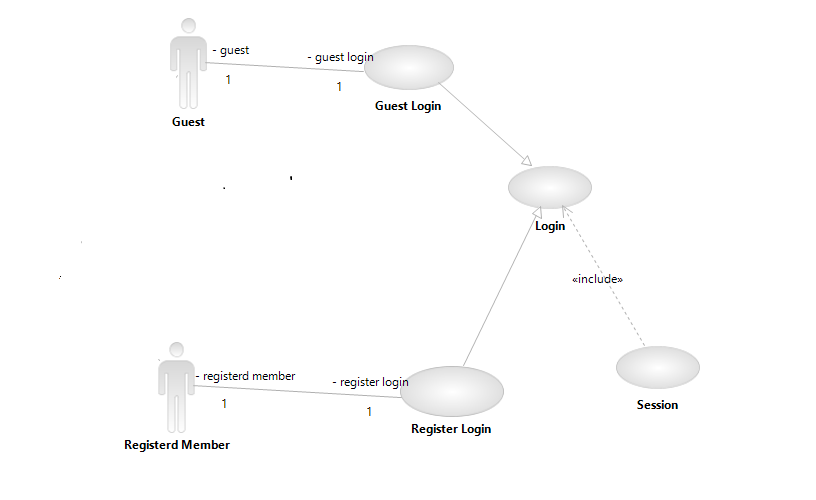


Figure 1 : Login

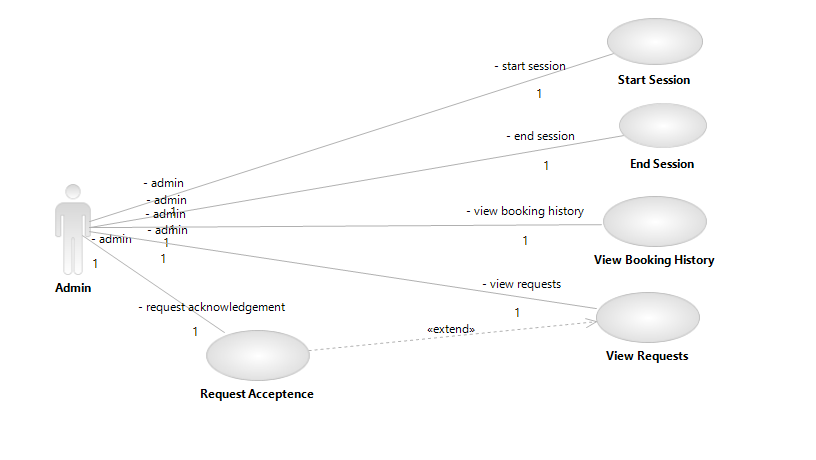


Figure 2 : Admin Login

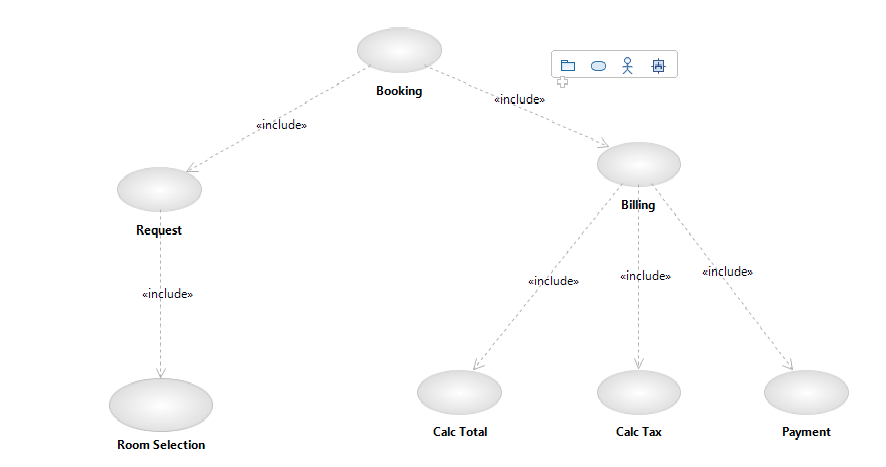


Figure 3 : Booking

### 2.2.1 Class

#### Class:

**Diagram:**

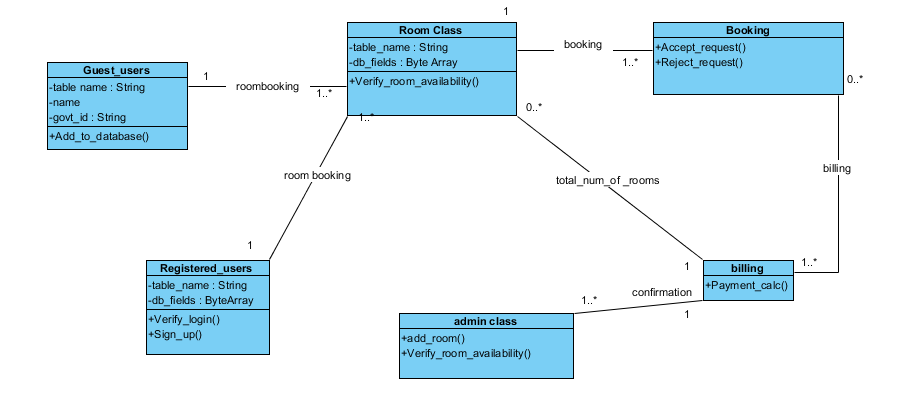


Figure 4 : Class Diagram

### 2.2.1 Sequence

#### Sequence:

**Diagram:**

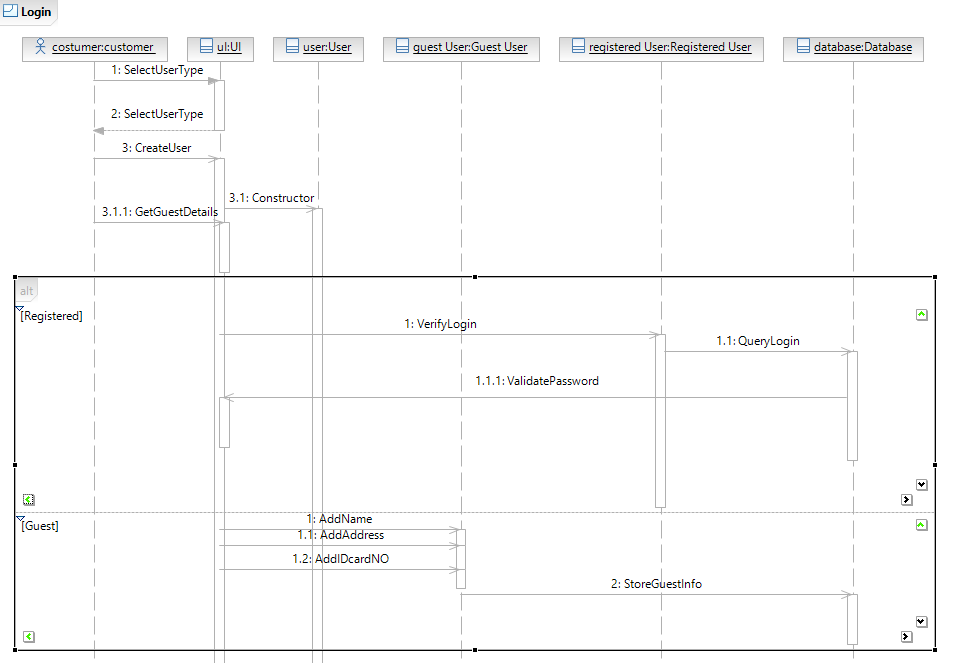


Figure 5 : Sequence Diagram 1

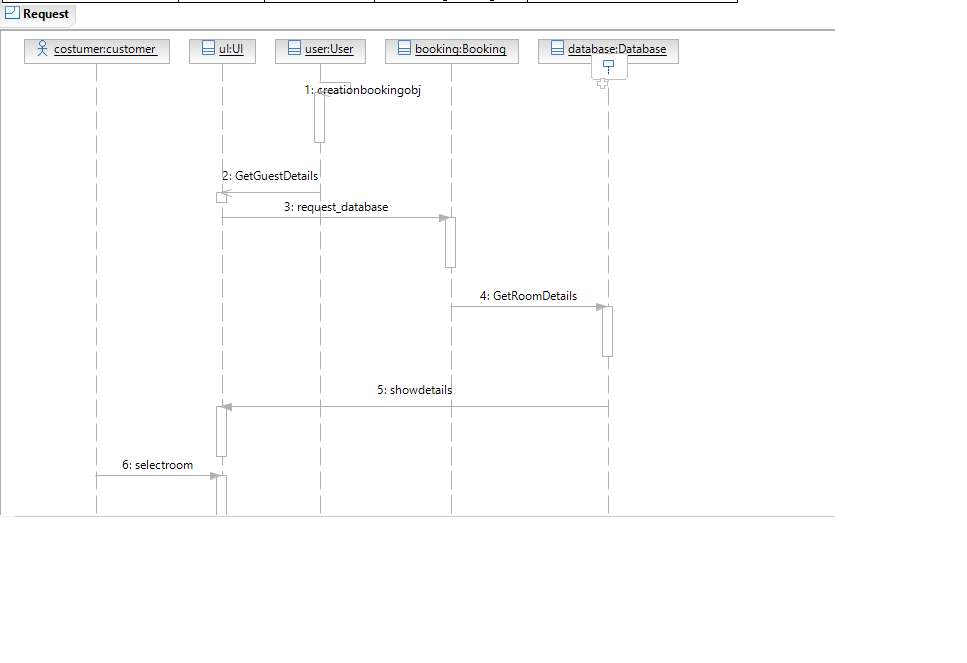


Figure 6 : Sequence Diagram 2

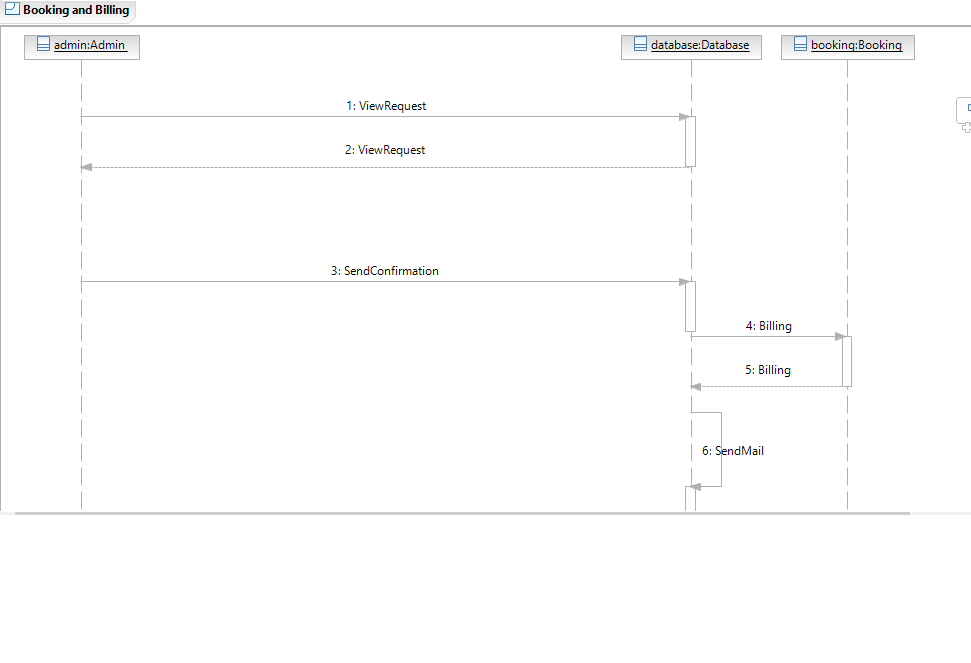


Figure 7 : Sequence Diagram 3

### 2.2.1 Activity

#### Activity:

**Diagram:**

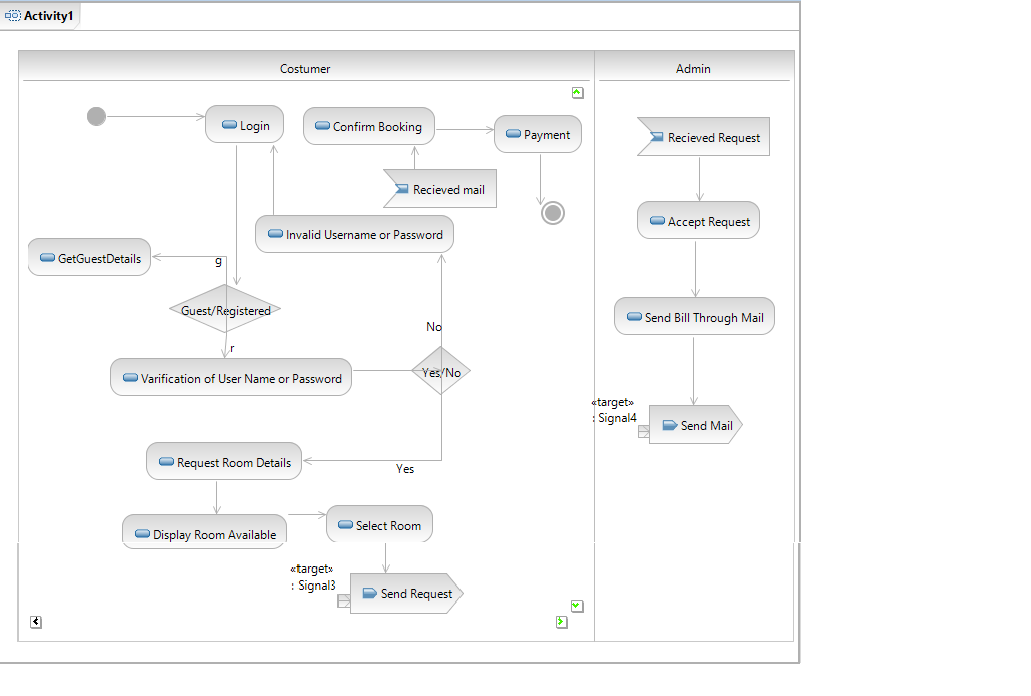


Figure 8 : Activity Diagram

**Brief Description**

**Initial Step-By-Step Description**

## 2.3 User Characteristics

The user of this software system requires the following skills to use this software

## 2.4 Non-Functional Requirements

# 3.0. Requirements Specification

## 3.1 Functional Requirements

### 3.1.1 Entry through verification to the portal

|  |  |
| --- | --- |
| **Use Case Name** | Logging in |
| **Trigger** | Opening of the portal. |
| **Precondition** | A person is interested in booking a room. |
| **Basic Path** | 1) For the admin and the registered members they must be able to provide their valid user name and password.  2) For the guest they must give details of any of their government issued identification card for the factor of safety. |
| **Alternative Paths** | The person in need of the room can directly meet the admin to get the details of the room . |
| **Postcondition** | The person logged in must be able to get the information of room availability. |
| **Exception Paths** | Authentication failure or wrong entry of identity information. |
| **Other** | A guest user must provide a govt. issued photo identity card information for security purpose. |

### 3.1.2 Request for booking a Room

|  |  |
| --- | --- |
| **Use Case Name** | Requesting for a room. |
| **Trigger** | A person logged in wants to book a room. |
| **Precondition** | Availability of vacant room. |
| **Basic Path** | A person logged in must be able to see a particular type of room ( AC or non AC , Double or single bedroom ) in the dates required and send a request to the admin for booking the same. |
| **Alternative Paths** | A person can go in person to the admin requesting for the availability of a room in the dates required. |
| **Postcondition** | Admin will receive the request and then can decide the allocation of the room. |
| **Exception Paths** |  |
| **Other** | << Any other important information>> |

### 3.1.2 Booking Confirmtion by Admin

|  |  |
| --- | --- |
| **Use Case Name** | Booking Confirmtion by Admin. |
| **Trigger** | The Admin logged in searching through all the requests. |
| **Precondition** | Availability of vacant room. |
| **Basic Path** | The Admin logged in must be able to select a particular request of room ( AC or non AC , Double or single bedroom ) by a guest in the dates required and send a confirmation to the guest user for the room. |
| **Alternative Paths** |  |
| **Postcondition** |  |
| **Exception Paths** |  |
| **Other** | << Any other important information>> |

### 3.1.2 Billing

|  |  |
| --- | --- |
| **Use Case Name** | Billing. |
| **Trigger** | A person logged in and select the billing option. |
| **Precondition** | Room confirmed by the Admin. |
| **Basic Path** | A person logged in must be able to select the check-in/out dates required and receive the payable amount. |
| **Alternative Paths** |  |
| **Postcondition** | Reciept generated. |
| **Exception Paths** |  |
| **Other** | << Any other important information>> |

## 3.3 Detailed Non-Functional Requirements

### ***3.4 Logical Structure of the Data***

<< Keep this blank for the time being>>

# 4.0 Supporting information

## 4.1 Table of contents and index

## 4.2 Appendixes