COMSATS University Islamabad, Abbottabad Campus

Department of Computer Science

CSC392 Object-Oriented Software Engineering

**Project Proposal**

**HOSTEL MANAGEMENT SYSTEM**

|  |
| --- |
| **Group Members** |
| FA21-BSE-015  Hanzla Nouman |
| FA21-BSE-019  Laiba Binta Tahir |
| FA21-BSE-027  Irfan |
| FA21-BSE-080  Arfah Ali |

…

Table of Contents

[CHAPTER 1 PROJECT PROPOSAL 3](#_Toc131615357)

[Introduction 3](#_Toc131615358)

[Vision and Business Case 4](#_Toc131615359)

[Use-Case Model 5](#_Toc131615360)

[Supplementary Specification 6](#_Toc131615361)

[Glossary 7](#_Toc131615362)

[Risk List & Risk Management Plan 7](#_Toc131615363)

[CHAPTER 2: USE CASES 9](#_Toc131615364)

[Use Case Diagram 9](#_Toc131615365)

[Use Cases Distribution 10](#_Toc131615366)

[Brief Level Use Cases 12](#_Toc131615367)

[*Laiba binta tahir (FA21-BSE-019 – 4A)* 12](#_Toc131615368)

[**Use Case: Delete student.** 12](#_Toc131615369)

[**Use Case: login.** 12](#_Toc131615370)

[Fully Dressed Use Cases 13](#_Toc131615371)

[*Laiba binta tahir (FA21-BSE-019 – 4A)* 13](#_Toc131615372)

[**Use Case: login.** 13](#_Toc131615373)

[**Use Case :delete student.** 14](#_Toc131615374)

# CHAPTER 1 PROJECT PROPOSAL

## Introduction

1.1 PROJECT OVERVIEW

The hostel management system is web-based software to provide university students accommodation to the university hostel more efficiently. This project also keeps details of the hostellers and applied students. It is headed by multiple administrators . This document is intended to minimize human works and make hostel allocation an easier job for students and hostel authorities by providing online application for hostel, automatically select the students from the waiting list and mess calculation, complaint registration, etc. Students will get approval notification in their mails. Hostellers can view hostel fee and mess menu by login into the online system.

1.2 PROBLEM STATEMENT

There are a lot of drawbacks in keeping and maintaining a hostel. Especially with a manual system. Since most hostels are being run by only one hostel manager, the number of students in a room are sometimes not known by the warden.

This particular project deals with the problems on managing a hostel and avoids the problems which occur when carried manually Identification of the drawbacks of the existing system leads to the designing of computerized system that will be compatible to the existing system with the system which is more user friendly and more GUI oriented.

We can improve the efficiency of the system, thus overcome the drawbacks of the existing system.

1.3 OBJECTIVES

• To make it easier for data collection, storage and referencing reliable.

• To maintain the students as hostellers and waiting list students separately.

• To process allotment list.

## 

## Vision and Business Case

***Vision:***

Our hostel management system aims to provide a streamlined and efficient solution for managing hostel operations, improving the student experience, and increasing overall productivity. By leveraging modern technologies, we seek to transform hostel management into an automated and user-friendly process that benefits both students and administrators alike.

***Business Case:***

Hostel management is a complex and time-consuming process that requires significant resources and effort to manage effectively. Traditional methods of managing hostels often involve manual processes such as paper-based record keeping, which can be error-prone, time-consuming, and difficult to manage.

Our hostel management system seeks to address these challenges by providing a comprehensive solution that automates many of the tasks involved in hostel management. This includes features such as room allocation, meal plans, student profiles, visitor management, and maintenance tracking. By automating these tasks, we aim to save time and increase the efficiency of hostel operations.

Additionally, our system includes features that enhance the student experience, such as the ability to view their room details, meal plans, and other relevant information through a user-friendly interface. This not only improves the overall student experience but also reduces the workload of administrators by reducing the number of inquiries and requests they receive from students.

Overall, our hostel management system aims to provide a cost-effective and efficient solution for managing hostel operations while also improving the student experience. By leveraging modern technologies and automation, we believe that our system will help hostels to run more smoothly, improve productivity, and ultimately increase profitability.

## Use-Case Model

Functional Requirements for Hostel Management System

* Allocaton /Booking
* record reservations
* record the student first name
* record the student’s last name
* record the number of students
* record the room number
* display the default room fee display whether or not the room is guaranteed.
* generate a unique confirmation number for each.
* record the expected check-in date and time
* The system shall record the expected checkout date and time
* The system shall record customer feedback

2.  Food

* The system shall track all meals purchased in the hotel .
* The system shall record payment and payment type for meals
* The system shall bill the current room if payment is not made at time of service
* The system shall accept reservations for services.

3. Management

* display the hotel occupancy for a specified period of time (days; including past, present, and future dates).
* display projected occupancy for a period of time (days).
* display room revenue for a specified period of time (days).
* display food revenue for a specified period of time (days).
* display an exception report, showing where default room and food prices have been overridden
* allow for the addition of information, regarding rooms, rates, menu items, prices, and user profiles
* allow for the deletion of information, regarding rooms, rates, menu items, prices, and user profiles
* allow for the modification of information, regarding rooms, rates, menu items, prices, and user profiles
* allow managers to assign user passwords

## Supplementary Specification

There are a lot of software requirements specifications included in the non-functional requirements of the Hostel Management System, which contains various processes, namely Security, Performance, Maintainability, and Reliability.

**Security:**

● user Identification: The system needs the user to recognize herself or himself using the phone.

● Logon ID: Any users who make use of the system need to hold a Logon ID and password.

● Modifications: Any modifications like insert, delete, update, etc. for the database can be synchronized quickly and executed only by the ward administrator.

● Front Desk Staff Rights: The staff at the front desk can view any data in the Hostel Management system.

● Administrator rights: The administrator can view as well as alter any information in the Hostel Management System.

**Performance:**

● Response Time: The system provides acknowledgment in just one second once the 'user's information is checked.

● Capacity: The system needs to support at least 1000 people at once.

● User-Interface: The user interface acknowledges within five seconds.

● Conformity: The system needs to ensure that the guidelines of the Microsoft accessibilities are followed.

**Maintainability:**

● Back-Up: The system offers efficiency for data backup.

● Errors: The system will track every mistake as well as keep a log of it.

**Reliability:**

● Availability: The system is available all the time.

## Glossary

|  |  |
| --- | --- |
| *HMS* | Hostel management system |
| *DFD* | Data flow diagram |
|  |  |
|  |  |

## 

## Risk List & Risk Management Plan

**Risk List:**

***Technical Risks*** *-* These risks include system failures, software bugs, and other technical issues that may disrupt the system's operations or cause data loss.

***Security Risks*** *-* These risks include data breaches, unauthorized access to the system, and other security threats that may compromise the system's integrity and the safety of students' information.

***Human Risks*** *-* These risks include human error, misconduct, and other issues that may arise from the actions of administrators or other users of the system.

***Operational Risks*** *-* These risks include issues related to the management of the hostel, such as room allocation, billing and payments, maintenance tracking, and other operations that may affect the system's performance.

**Risk Management Plan:**

***Technical Risks*** *-* To mitigate technical risks, the system will be regularly tested for bugs and other technical issues. A backup system will also be implemented to ensure that data is not lost in case of a system failure.

***Security Risks*** *-* To mitigate security risks, the system will be protected by advanced security protocols such as encryption, firewalls, and regular vulnerability testing. Access to the system will be restricted to authorized users only, and data will be regularly backed up and secured.

***Human Risks*** *-* To mitigate human risks, the system will be designed to limit the potential for user error, such as providing clear and intuitive user interfaces and requiring authorization for critical functions. Administrators will also be required to undergo training and follow strict protocols to ensure the safety and integrity of student data.

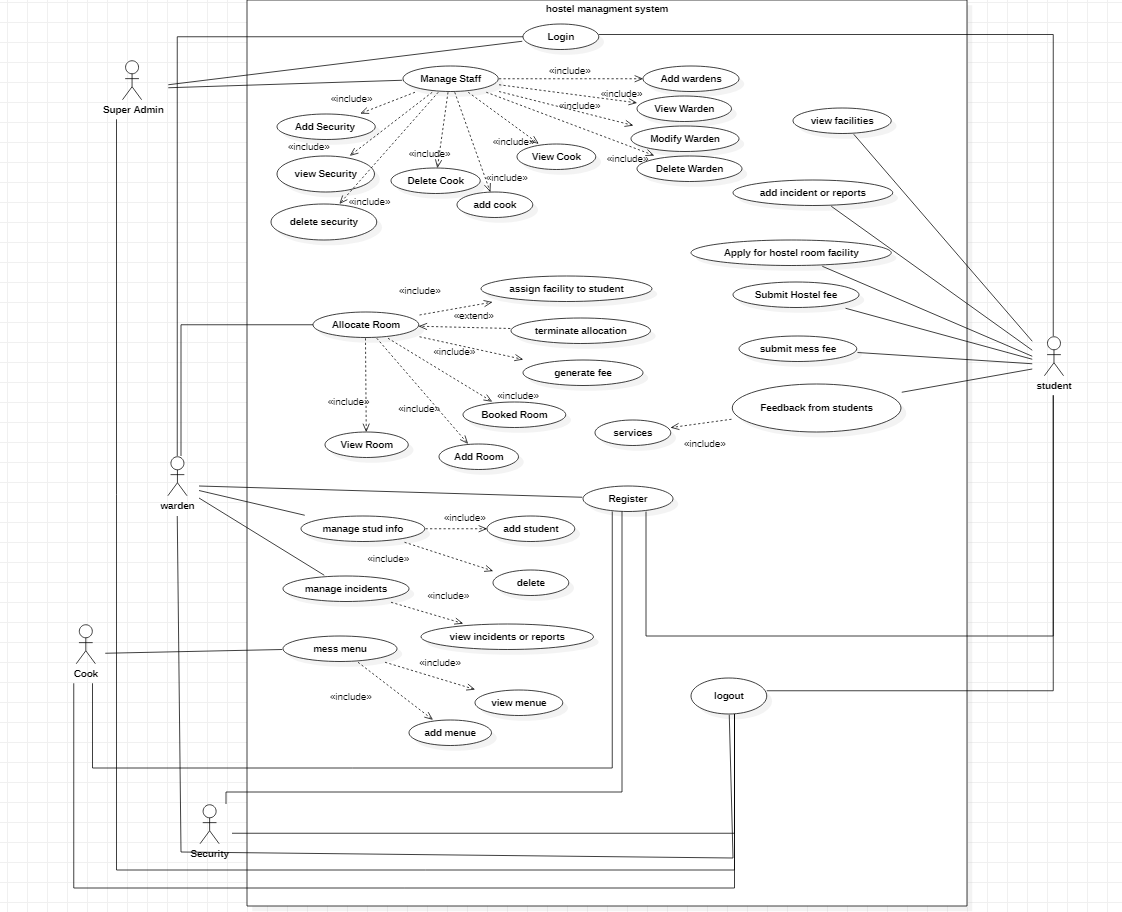
***Operational Risks*** *-* To mitigate operational risks, the system will be designed to automate and streamline hostel operations, such as room allocation and maintenance tracking. The system will also provide detailed analytics and reports to help administrators identify and address operational issues before they become major problems.

Overall, by implementing a comprehensive risk management plan, the hostel management system can minimize the potential for risk and ensure the smooth and efficient operation of the system.

# 

# CHAPTER 2: USE CASES

## Use Case Diagram



## Use Cases Distribution

|  |  |  |
| --- | --- | --- |
| S#. | Group Member | Assigned Use Cases |
| 1 | FA21-BBSE-0  Hanzala noman | Allocate rooms:   * Add room * View room * Generate fee * Terminate allocation * Assign facilities to students * Booked room |
| 2 | FA21-BSE-019  Laiba binta tahir | Mess menu  Login  Logout  Manage students   * Add student * View student * Delete student * modify student |
| 3 | FA21-BSE-027  Irfan | Register  Manage warden.   * Add warden. * View warden. * Delete warden. * modify warden.   Manage staff   * Add security * View security * Delete security * Add cook * View cook * Delete cook |
| 4 | FA21-BSE-080  Arfah Ali | View facilities  Apply for hostel  Add incidents  Submit fee(hostel/mess)  View incidents and reports.  Feedback from students |

## Brief Level Use Cases

### *Laiba binta tahir (FA21-BSE-019 – 4A)*

#### **Use Case: Delete student.**

The admin logs into the hostel management system and selects the "Delete *student*" option from the main menu. The system retrieves the list of all registered *students* and displays their details, such as name, contact information, on the screen. The admin selects the *student* to delete from the list and confirms the action. The system removes the selected *student* from the system, revokes their access to the system, and updates the records accordingly. The system then displays a confirmation message to the admin.

#### **Use Case: login.**

The user selects the login option in the hostel management system. The user enters their username and password. The system verifies the user's credentials. If the user's credentials are valid, the system logs the user into the system and displays the appropriate dashboard based on their role (student or administrator). If the user's credentials are invalid, the system displays an error message and prompts the user to re-enter their credentials.

## Fully Dressed Use Cases

### *Laiba binta tahir (FA21-BSE-019 – 4A)*

#### **Use Case: login.**

**Use Case Name:** Login

**Scope:** Hostel Management System

**Primary Actor:** Administrator or Student

Goal in Context: The primary goal of this use case is to allow administrators or students to log into the hostel management system to access their respective features and functionalities.

**Preconditions:**

The user has a valid account with the hostel management system.

The user has a device with internet access.

The user has their login credentials, including their username and password.

**Trigger:**

The user attempts to log into the hostel management system by entering their login credentials.

**Main Success Scenario:**

The user navigates to the login page of the hostel management system.

The system prompts the user to enter their login credentials.

The user enters their username and password.

The system validates the user's credentials.

The system grants the user access to their respective features and functionalities within the system.

**Extensions:**

If the user enters an incorrect username or password, the system will display an error message and prompt the user to re-enter their credentials.

If the user forgets their password, they can request a password reset link or contact the administrator for assistance.

**Alternate Flow:**

If the user does not have a valid account with the hostel management system, they can create an account by clicking the "register" button on the login page.

**Postconditions:**

The user is successfully logged into the hostel management system and has access to their respective features and functionalities.

The system logs the user's login activity for security and audit purposes.

**Exceptions:**

If the system experiences technical issues, the user may not be able to log in. In this case, the user can contact technical support or try again later.

#### **Use Case :delete student.**

**Use Case Name:** Delete student.

**Primary Actor:** Administrator

**Goal in Context**: The primary goal of this use case is to allow the administrator to delete a student from the hostel management system and associated records.

**Preconditions**

The administrator is logged in to the hostel management system.

The administrator has the appropriate privileges to delete a student record.

The student record to be deleted exists in the system.

**Trigger:**

The administrator selects the option to delete a student record from the system.

**Main Success Scenario:**

The administrator navigates to the student management section of the hostel management system.

The administrator selects the student record to be deleted.

The system prompts the administrator to confirm the deletion of the student record.

The administrator confirms the deletion of the student record.

The system deletes the student record and associated records from the system.

**Extensions:**

If the student record to be deleted has pending fees or dues, the system will prompt the administrator to settle the outstanding balance before proceeding with the deletion.

If the student record to be deleted is linked to other records, such as room allocations or meal plans, the system will prompt the administrator to reassign these records to another student or delete them along with the student record.

**Alternate Flow:**

If the administrator accidentally selects the wrong student record for deletion, they can cancel the deletion and select the correct record.

**Postconditions:**

The student record and associated records are successfully deleted from the hostel management system.

The system logs the administrator's action for audit purposes.

**Exceptions:**

If the administrator does not have the appropriate privileges to delete a student record, the system will display an error message and prevent the deletion from proceeding.

If the student record to be deleted does not exist in the system, the system will display an error message and prevent the deletion from proceeding.