



# INDEX

<b>Sl. No.</b>	<b>Topics</b>	<b>Page No.</b>
<b>I</b>	<b>INTRODUCTION</b>	<b>7</b>
1.1	Introduction	8
1.2	Objective of the Project	9
1.3	System Study	10
	1.3.1 Study of Existing System	10
	1.3.2 Literature Survey	11
	1.3.3 Demerits of Existing System	12
	1.3.4 Study on Proposed System	12
	1.3.5 Merits of Proposed System	13
	1.3.6 Advantages of Proposed System	14
<b>II</b>	<b>SYSTEM ANALYSIS</b>	<b>17</b>
2.1	User Requirement Specifications	18
2.2	Software Requirement Specification (SRS)	23
	2.2.1 Functional System Requirements	23
	2.2.2 Non Functional System Requirements	24
2.3	System Specifications	24
	2.3.1 Hardware Specifications	24
	2.3.2 Software Specifications	25

2.4	Final Outline of Proposed System	25
<b>III</b>	<b>DESIGN AND DEVELOPMENT PROCESS</b>	<b>28</b>
3.1	Fundamental Design Concepts	29
	3.1.1 Modules and Functionalities	29
	3.1.2 Description of Modules	29
3.2	Design Notations	34
	3.2.1 Data Flow Diagram Symbol	34
	3.2.2 Data Flow Diagram	35
	3.2.3 ER Diagram	37
3.3	Design Process	39
	3.3.1 Database Design	39
	3.3.2 Schema Diagram	42
3.4	Development Approach	43
	3.4.1 Development Model / Approach	43
<b>IV</b>	<b>PLANNING</b>	<b>45</b>
4.1	Gantt Chart	46
<b>V</b>	<b>TESTING AND IMPLEMENTATION</b>	<b>47</b>
5.1	Testing	48

	5.1.1 Testing Methodologies	48
	5.1.2 Test Cases	48
5.2	System Implementation	49
	5.2.1 Implementation Procedures	49
	5.2.2 User Training	49
5.3	System Maintenance	50
<b>VI</b>	<b>CONCLUSION</b>	<b>51</b>
6.1	Conclusion	52
6.2	Scope for Future Enhancements	52
	<b>ANNEXURE- A</b>	<b>54</b>
	Sample Code	55-92
	<b>ANNEXURE-B</b>	<b>93</b>
	Snapshots	94-105
	<b>BIBLIOGRAPHY</b>	<b>106</b>

# **CHAPTER-I**

## **INTRODUCTION**

## 1.1 INTRODUCTION

Hostel Management System is a software application designed to streamline and automate the administrative tasks and processes involved in managing a hostel or similar accommodation facility. It provides an efficient and organized approach to managing room allocation, Hostel Manager registration, Student Registration and other essential operations. This system is widely used in educational institutions, corporate organizations, and other establishments that provide hostel facilities.

The primary objective of a Hostel Management System is to simplify the overall management of a hostel, ensuring smooth operations and enhancing the experience of both hostel administrators and residents. By replacing manual paperwork and traditional record-keeping methods, it eliminates the risk of errors, reduces paperwork, and saves time.

One of the key features of the system is room allocation and management. It allows administrators to allocate rooms to residents based on their preferences, such as room type, occupancy, and special requirements. The system maintains a centralized database of all rooms, along with their availability status, ensuring efficient room allocation and utilization. This feature also enables easy tracking of room vacancies and helps to streamline the check-in and check-out processes.

The guest registration module simplifies the process of capturing and managing resident information. Hostel administrators can easily record details such as resident names, contact information, identification documents, and emergency contacts. This information can be readily accessed whenever required, facilitating efficient communication and ensuring the safety and security of the residents.

Communication plays a crucial role in hostel management, and the system provides tools to facilitate effective communication between administrators and residents. It includes features such as messaging systems, notice boards, and email notifications, enabling administrators to disseminate important information, announcements, and updates efficiently.

Reporting and analytics capabilities are integral to the Hostel Management System. Administrators can generate various reports related to room occupancy, resident information, financial transactions, and other relevant data. These reports provide valuable insights for decision-making, enabling administrators to make informed choices and optimize hostel operations.

In conclusion, a Hostel Management System is a comprehensive software application that simplifies and automates various administrative tasks involved in managing a hostel or similar accommodation facility. It streamlines processes such as room allocation, guest registration, billing, inventory management, and communication. By replacing manual methods, it enhances efficiency, accuracy, and overall hostel management. Implementing a Hostel Management System can greatly benefit educational institutions, corporate organizations, and other establishments by providing a seamless and well-organized system for managing their hostel facilities.

## 1.2 OBJECTIVE OF THE PROJECT

The objective of a Hostel Management System is to provide an efficient and effective solution for managing and overseeing the operations of a hostel or similar accommodation facility. The system aims to achieve the following objectives:

1. **Streamline Administrative Tasks:** The primary objective is to streamline administrative tasks and processes involved in managing a hostel. The system automates tasks such as room allocation, guest registration, billing, inventory management, and communication, reducing manual effort, minimizing errors, and saving time.
2. **Efficient Room Allocation:** The system aims to facilitate efficient room allocation based on resident preferences, room availability, and special requirements. It helps administrators optimize room occupancy, avoid overbooking, and ensure fair and effective distribution of rooms among residents.
3. **Simplify Guest Registration:** The system simplifies the process of guest registration by providing a centralized platform to record and manage resident information. It aims to capture essential details accurately, such as resident names, contact information, identification documents, and emergency contacts, ensuring efficient communication and ensuring resident safety and security.
4. **Communication and Information Sharing:** The system aims to enhance communication between administrators and residents. It provides tools such as messaging systems, notice boards, and email notifications to disseminate

important information, announcements, and updates. This objective ensures effective communication and facilitates timely sharing of relevant information.

5. **Reporting and Analytics:** The system aims to provide comprehensive reporting and analytics capabilities. Administrators can generate various reports related to room occupancy, resident information, financial transactions, and other relevant data. This objective helps administrators make informed decisions, identify trends, and optimize hostel operations.
6. **To deal with Hostel Management System in an easy and an efficient manner.**
7. **Create a strong and secret database** that allows for any connection in a secret way, to prevent any outside or inside attacks.

Overall, the objective of a Hostel Management System is to optimize the management and operations of a hostel, improve efficiency, accuracy, and resident satisfaction. It aims to simplify administrative tasks, streamline processes, and provide a centralized platform for effective communication and decision-making.

## 1.3 SYSTEM STUDY

### 1.3.1 EXISTING SYSTEM

In the existing system for Hostel Management, manual methods and traditional paperwork are typically employed to manage the operations of a hostel. These methods involve a range of manual tasks and processes, which can be time-consuming, prone to errors, and inefficient. Some aspects of the existing system include:

1. **Manual Room Allocation:** In the existing system, room allocation is done manually by the hostel staff. They maintain physical records or spreadsheets to keep track of room availability and allocate rooms based on resident preferences and requirements. This process can be cumbersome and may lead to errors or inconsistencies in room allocation.
2. **Paper-based Guest Registration:** Guest registration in the existing system involves filling out paper-based registration forms. Hostel staff manually record resident details such as names, contact information, identification documents, and emergency contacts. Storing and retrieving this information becomes challenging, and there is a risk of misplacing or losing important documents.

3. **Communication Challenges:** Communication between hostel staff and residents in the existing system is primarily done through physical notices, bulletin boards, or verbal communication. This method may lead to miscommunication, delayed information sharing, or missed announcements. It can be challenging to disseminate important information to all residents effectively.
4. **Limited Reporting and Analytics:** In the existing system, generating reports and analyzing data related to hostel operations can be time-consuming and labor-intensive. Staff often need to manually compile data from various sources to generate reports on room occupancy, financial transactions, and other metrics. This process lacks real-time insights and may not provide a comprehensive view of the hostel's performance.

Overall, the existing system for Hostel Management relies heavily on manual processes, paperwork, and spreadsheets. These methods are prone to errors, time-consuming, and can lead to inefficiencies in managing hostel operations. To overcome these challenges, organizations are increasingly adopting modern Hostel Management Systems that automate and streamline these processes, improving efficiency, accuracy, and resident satisfaction.

### 1.3.2 LITERATURE SURVEY

Hostel Management Systems (HMS) have emerged as indispensable tools for efficiently managing hostel operations in educational institutions, corporate organizations, and other establishments. This literature survey aims to provide a comprehensive overview of the current state of research, advancements, challenges, and best practices in the field of HMS. By reviewing relevant research articles, conference papers, theses, and other scholarly sources, this survey presents key insights into the features and functionalities of HMS, technological platforms, user experience, security and privacy considerations, implementation challenges, benefits, and future research directions. The findings of this survey contribute to a deeper understanding of HMS and provide valuable guidance for researchers, practitioners, and decision-makers involved in the design, development, and implementation of hostel management solutions.

### **1.3.3 DEMERITS OF EXISTING SYSTEM**

1. More prone to human error
2. Increased strength and strain of manual labor
3. Less security
4. Data redundancy
5. Data inconsistency
6. Difficult and Complex to handle
7. Data updating is an hectic task
8. Hard to maintain record
9. Backup data cannot be generated.

### **1.3.4 STUDY ON PROPOSED SYSTEM**

This study examines a proposed system for Hostel Management to address the existing challenges and enhance the efficiency of hostel operations. The objective is to evaluate the feasibility and potential benefits of implementing the proposed system in a real-world hostel setting. The study focuses on analyzing the system's features, functionalities, user experience, security measures, implementation process, and expected impacts. Through a combination of literature review, interviews, and user feedback, this study aims to provide valuable insights into the proposed system's effectiveness and potential for improving hostel management.

Through a comprehensive study of the proposed system, this research aims to provide an in-depth understanding of its potential to address the challenges faced in hostel management. The findings will help decision-makers and stakeholders assess the feasibility and benefits of implementing the proposed system in their respective hostels. Furthermore, this study contributes to the growing body of knowledge on hostel management systems and serves as a foundation for future research and system improvements.

This paper presents a proposed system for Hostel Management that aims to streamline and optimize hostel operations through the integration of modern technologies and efficient processes. The proposed system encompasses various modules, including room allocation, resident registration, communication tools, and reporting capabilities. By leveraging advancements in technology and addressing the specific needs of hostel management, this system aims to enhance the overall efficiency, accuracy, and

resident satisfaction in hostel operations. The proposed system serves as a comprehensive framework for institutions and organizations seeking to implement an advanced Hostel Management System that caters to their unique requirements.

By presenting a comprehensive proposed system for Hostel Management, this paper provides a framework that institutions and organizations can utilize to enhance their hostel operations. The integration of advanced technologies, efficient processes, and user-friendly interfaces is expected to improve the overall efficiency, accuracy, and resident satisfaction in hostel management. The proposed system serves as a blueprint for implementation and customization, ensuring it aligns with the unique requirements and objectives of each institution or organization.

### **1.3.5 MERITS OF PROPOSED SYSTEM**

The proposed system for Hostel Management encompasses several merits that contribute to its effectiveness and potential benefits. Here are some key merits of the proposed system:

- 1. Enhanced Operational Efficiency:** The proposed system streamlines hostel operations by automating processes such as room allocation, resident registration, billing, inventory management, and communication. This automation reduces manual effort, minimizes errors, and enables faster and more efficient execution of tasks, leading to overall improved operational efficiency.
- 2. Improved Resident Satisfaction:** With features like intelligent room allocation, resident preferences management, and effective communication tools, the proposed system enhances the resident experience. Residents can easily access and update their information, receive timely notifications and announcements, and have their preferences considered, leading to higher satisfaction levels and a better overall hostel experience.
- 3. Data-Driven Decision Making:** The reporting and analytics module provides comprehensive reports and data visualization tools that offer insights into occupancy rates, finances, and resident information. These analytics support informed decision-making, enabling hostel administrators to identify trends, allocate resources effectively, and plan for future requirements.

4. **Scalability and Customization:** The proposed system is designed to be scalable and customizable to suit the specific needs of different institutions and organizations. It can accommodate various hostel sizes, multiple campuses, and diverse operational requirements. This flexibility allows for seamless implementation and adaptation to different environments.
5. **Integration with Institutional Systems:** The proposed system can integrate with other institutional systems such as student information systems and finance systems. This integration ensures smooth data exchange, reduces redundancy, and improves overall efficiency in managing student-related information and financial processes.
6. **Enhanced Security and Privacy:** The proposed system prioritizes security and privacy considerations. It incorporates data encryption, access controls, user authentication, and compliance with data protection regulations. These measures protect resident information, ensuring confidentiality and building trust among residents.
7. **Cost Savings:** By optimizing resource utilization, automating processes, and minimizing manual errors, the proposed system can result in cost savings for institutions. Reduction in administrative workload, improved financial management, and efficient inventory control contribute to cost efficiency and financial sustainability.

Overall, the proposed system for Hostel Management offers numerous merits, including enhanced operational efficiency, improved resident satisfaction, accurate billing and payment processing, efficient inventory management, data-driven decision making, scalability, customization, enhanced security and privacy, and cost savings. These merits collectively contribute to a well-managed and resident-centric hostel environment, benefiting both administrators and residents alike.

### 1.3.6 ADVANTAGES OF PROPOSED SYSTEM

1. Less human error
  2. Strength and strain of manual labor can be reduced
  3. High security
  4. Data redundancy can be avoided to some extent
-

5. Data consistency
6. Easy to handle
7. Easy data updating
8. Easy record keeping
9. Backup data can be easily generated.

The proposed system for Hostel Management (HMS) offers several advantages that can significantly improve the efficiency and effectiveness of hostel operations. Here are some key advantages of the proposed system:

1. **Automation and Efficiency:** The proposed system automates various manual tasks and processes involved in hostel management, such as room allocation, resident registration, billing, inventory management, and communication. This automation eliminates time-consuming manual efforts, reduces errors, and improves overall operational efficiency.
2. **Streamlined Processes:** With the proposed system, hostel processes become streamlined and standardized. Tasks like room allocation, resident registration, and billing follow predefined workflows, ensuring consistency and minimizing confusion. This leads to smoother operations and better utilization of resources.
3. **Improved Accuracy and Data Integrity:** By automating data entry and management, the proposed system reduces the risk of human errors. It ensures accurate and up-to-date information related to resident details, room availability, billing records, and inventory levels. This enhances data integrity and eliminates discrepancies.
4. **Enhanced Resident Experience:** The proposed system focuses on improving the resident experience by providing convenient features and functionalities. Residents can easily access and update their information, view room availability, make payments online, receive timely notifications and announcements, and communicate with hostel staff. This leads to higher resident satisfaction and a positive hostel experience.
5. **Real-time Information and Reporting:** The proposed system provides real-time information about room availability, occupancy rates, billing status, and inventory levels. Administrators can generate comprehensive reports and

analytics, enabling data-driven decision-making. This real-time information helps administrators to make informed decisions promptly.

6. **Customization and Scalability:** The proposed system can be customized to meet the specific requirements of different hostels and institutions. It can accommodate various hostel sizes, layouts, and operational needs. Additionally, the system is designed to be scalable, allowing it to handle increasing volumes of data and accommodate future growth.
7. **Integration with Existing Systems:** The proposed system can integrate with other existing institutional systems, such as student information systems, finance systems, or access control systems. This integration ensures seamless data exchange and avoids duplication of effort. It improves overall efficiency and provides a unified view of hostel operations.
8. **Data Security and Privacy:** The proposed system incorporates robust security measures to protect resident information and maintain privacy. It includes features such as data encryption, access controls, and user authentication. Compliance with data protection regulations ensures the confidentiality and integrity of resident data.
9. **Cost Savings and Resource Optimization:** The automation and efficiency achieved through the proposed system result in cost savings and resource optimization. Reduced manual effort, improved inventory management, accurate billing, and streamlined processes contribute to financial savings and optimal resource utilization.
10. **Future Expansion and Upgrades:** The proposed system is designed to accommodate future expansions and technological advancements. It can incorporate emerging technologies such as artificial intelligence, machine learning, or Internet of Things (IoT) to further enhance hostel management capabilities.

Overall, the proposed system for HMS offers advantages such as automation, streamlined processes, improved accuracy, enhanced resident experience, real-time information and reporting, customization, integration with existing systems, data security and privacy, cost savings, resource optimization, and scalability. These advantages collectively contribute to a more efficient, effective, and resident-centric hostel management environment.

---

## **CHAPTER-II**

# **SYSTEM ANALYSIS**

## 2.1 USER REQUIREMENT SPECIFICATION

The user requirement for this system is to make the system fast, flexible, less prone to error, reduce expenses and save time.

1. Less human error
2. Strength and strain of manual labor can be reduced
3. High security
4. Data redundancy can be avoided to some extent
5. Data consistency
6. Easy to handle
7. Easy data updating
8. Easy record keeping
9. Backup data can be easily generated.

A Hostel Management System (HMS) consists of various modules that collectively handle different aspects of hostel operations. Here are descriptions of some key modules in an HMS :

### 1. Student Module :

The student module in a hostel management system (HMS) is a software application that allows hostel administrators to track and manage student information, including:

1. **Personal information:** This includes the student's name, date of birth, contact information, and emergency contact information.
2. **Academic information:** This includes the student's academic records, such as grades, attendance, and transcripts.
3. **Medical information:** This includes the student's medical history, allergies, and medications.
4. **Financial information:** This includes the student's tuition, fees, and financial aid information.
5. **Other information:** This may include the student's extracurricular activities, housing preferences, and dietary restrictions.

The student module in an HMS can be used to:

1. **Generate reports:** The hostel administrator can use the student module to generate reports on student academic performance, financial aid status, and other areas.
2. **Communicate with students:** The hostel administrator can use the student module to communicate with students about important information, such as upcoming events, changes to policies, and deadlines.
3. **Provide students with access to their information:** The student module can be used to provide students with access to their personal information, such as grades, attendance, and financial aid information.

The student module in an HMS can help hostel administrators to improve the efficiency and effectiveness of their operations. By automating tasks, such as tracking attendance and generating reports, the student module can free up hostel administrators' time so they can focus on other important tasks.

Additionally, the student module can help hostel administrators to improve communication with students and provide them with access to their information.

**Here are some of the benefits of using a student module in an HMS:**

1. **Improved efficiency:** The student module can help hostel administrators to automate tasks, such as tracking attendance and generating reports. This can free up hostel administrators' time so they can focus on other important tasks.
2. **Improved communication:** The student module can help hostel administrators to improve communication with students. Hostel administrators can use the student module to send messages to students, update them on important information, and answer their questions.
3. **Improved student satisfaction:** Students can access their information through the student module. This can help students to stay organized and on top of their studies.

If you are considering implementing an HMS, it is important to choose a system that has a student module. The student module can be a valuable tool for hostel administrators and students alike.

## 2. Admin Module :

The admin module is a critical part of any hostel management system. It allows the administrator to manage all aspects of the hostel, including:

1. **Student records:** The admin can create, edit, and delete student records. This includes information such as the student's name, contact information, academic records, and hostel room assignment.
2. **Room assignments:** The admin can assign students to rooms. This can be done manually or automatically, based on factors such as the student's year of study, academic standing, and roommate preferences.
3. **Maintenance:** The admin can track and manage maintenance requests. This includes requests for repairs, cleaning, and other services.
4. **Reporting:** The admin can generate reports on various aspects of the hostel, such as student attendance, room occupancy, and fee collection.

The admin module is a powerful tool that can help to improve the efficiency and effectiveness of any hostel management system. By giving the administrator the ability to manage all aspects of the hostel from a single location, the admin module can help to ensure that the hostel is running smoothly and efficiently.

**Here are some of the benefits of having an admin module in a hostel management system:**

1. **Improved efficiency:** The admin module can help to improve the efficiency of the hostel by automating many of the tasks that were previously done manually. This can free up the administrator's time to focus on other important tasks.
2. **Increased accuracy:** The admin module can help to improve the accuracy of data by providing a single source of truth for all hostel records. This can help to reduce errors and improve decision-making.
3. **Enhanced security:** The admin module can help to enhance the security of the hostel by providing a centralized system for managing access control, video surveillance, and fire safety. This can help to protect students and staff from harm.

4. **Improved communication:** The admin module can help to improve communication between the administrator and students, staff, and parents. This can help to resolve issues quickly and efficiently.

Overall, the admin module is a valuable tool that can help to improve the efficiency, accuracy, security, and communication of any hostel management system.

### **3. Hostel Manager Module :**

The hostel manager module in a hostel management system is responsible for room allocation. This module typically includes the following features:

1. **Room availability:** The module tracks the availability of rooms in the hostel. This information is used to determine which students can be assigned to a room.
2. **Room preferences:** The module allows students to specify their room preferences. This information is used to try to assign students to rooms that meet their preferences.
3. **Room allocation rules:** The module can be configured with rules for assigning students to rooms. These rules can be based on factors such as the student's year of study, academic standing, and roommate preferences.
4. **Room assignment notifications:** The module can send notifications to students when they have been assigned a room.

The hostel manager module is an important part of any hostel management system. It allows the hostel manager to efficiently and fairly allocate rooms to students.

**Here are some of the benefits of having a hostel manager module in a hostel management system:**

1. **Efficiency:** The hostel manager module can help to improve the efficiency of the room allocation process by automating many of the tasks that were previously done manually. This can free up the hostel manager's time to focus on other important tasks.

2. **Fairness:** The hostel manager module can help to ensure that the room allocation process is fair by using rules that are based on objective factors. This can help to reduce the likelihood of students being assigned to rooms that they do not want.
3. **Transparency:** The hostel manager module can help to improve transparency by providing students with information about the room allocation process. This can help to reduce the likelihood of students feeling that they have been treated unfairly.

Overall, the hostel manager module is a valuable tool that can help to improve the efficiency, fairness, and transparency of the room allocation process in any hostel.

**Here are some additional tips for using the hostel manager module to improve room allocation:**

1. **Set clear rules for room allocation:** The hostel manager should set clear rules for room allocation. These rules should be based on objective factors, such as the student's year of study, academic standing, and roommate preferences.
2. **Communicate the rules to students:** The hostel manager should communicate the rules for room allocation to students. This can be done through the hostel's website, email, or in person.
3. **Allow students to specify their room preferences:** The hostel manager should allow students to specify their room preferences. This information can be used to try to assign students to rooms that meet their preferences.
4. **Monitor the room allocation process:** The hostel manager should monitor the room allocation process to ensure that it is fair and efficient. This can be done by reviewing the rules for room allocation, communicating with students, and resolving any issues that arise.

## 2.2 SOFTWARE REQUIREMENT SPECIFICATION

The Software Requirements Specification (SRS) will provide a detailed description of requirements for the Hostel Management System (HMS). This SRS will be helpful for complete understanding what is to be expected from the newly introduced system which is to be constructed. The clear understanding of the system and its functionality will allow for the correct software to be developed for the end user and will be used for the development of the future stages of the project. This SRS will provide the foundation of the project. From this SRS, the Hostel Management Systems can be designed, constructed and finally tested. The Project team will use the SRS to fully understand the expectations of the HMS to construct the appropriate software. The hostel end users will be able to use the SRS as a “test” to see if the constructing team will be constructing the system to their expectations.

**The Web Application has two main parts:**

- 1) Hostel Administrators
- 2) Students

The **student** can select among the allocated hostel to a specified batch and the **Hostel Administrator** can assign the room number in the specific hostel that the student has selected upon the availability.

### 2.2.1 FUNCTIONAL SYSTEM REQUIREMENTS

This section gives a functional requirement that is applicable to the HMS.

These are sub modules in this phase.

1. Administrator module.
2. User Module

**The functionality of each module is as follows:**

#### **1. Administrator module:**

The Administrator can :

1. Allot different students to the different hostels.
2. Vacate the students from the hostels.
3. Edit the details of the students & modify the student records.

#### **2. User module:**

1. Can submit the application form
2. Can view the notice board
3. Can submit the vacating form.

## 2.2.2 NON-FUNCTIONAL SYSTEM REQUIREMENTS

### 1. Performance Requirements

Some Performance requirements identified is listed below:

1. The database shall be able to accommodate around thousand records to store.
2. The software shall support use of multiple users at a time.

### 2. Safety Requirements

The database may get crashed at any certain time due to virus or operating system failure. Therefore, it is required to take the database backup.

### 3. Security Requirements

Some of the factors that are identified to protect the software from accidental or malicious access, use, modification, destruction, or disclosure are described below. Keep specific log or history data sets

1. Assign certain functions to different modules
2. Restrict communications between some areas of the program
3. Check data integrity for critical variables
4. Later version of the software will incorporate encryption
5. techniques in the user/license authentication process

## 2.3 SYSTEM SPECIFICATION

### 2.3.1 HARDWARE SPECIFICATION

<b>Ram (Min)</b>	512 MB
<b>Rom (Min)</b>	1 GB
<b>Processor</b>	Pentium or greater
<b>Accessories</b>	Mouse, Keyboard

## 2.3.2 SOFTWARE SPECIFICATION

<b>Platform/OS</b>	Windows 11
<b>Software</b>	Xampp Server, VS Code
<b>Language</b>	HTML, CSS, PHP
<b>Database</b>	My SQL
<b>Server</b>	Apache
<b>Tools</b>	PhpMyAdmin
<b>Browser</b>	Google Chrome, Brave

## 2.4 FINAL OUTLINE OF PROPOSED SYSTEM

### System Overview

The proposed system of HMS is a web-based application that will be used to manage the day-to-day operations of a hostel. The system will provide a centralized platform for managing student records, room assignments, fees, maintenance requests, security, and reporting.

### System Modules

The proposed system will consist of the following modules:

- 1. Admin Module:** The Admin Module will be used by the hostel administrator to manage all aspects of the hostel. This includes managing student records, room assignments, fees, maintenance requests, security, and reporting.
- 2. Hostel Manager Module:** The Hostel Manager Module will be used by the hostel manager to manage the day-to-day operations of the hostel. This includes managing room availability, room preferences, room allocation rules, and room assignment notifications.
- 3. Student Module:** The Student Module will be used by students to view their own records, room assignments, fees, and maintenance requests.

## System Features

The proposed system will offer the following features:

1. **Student Records:** The system will allow the administrator to create, edit, and delete student records. This includes information such as the student's name, contact information, academic records, and hostel room assignment.
2. **Room Assignments:** The system will allow the hostel manager to assign students to rooms. This can be done manually or automatically, based on factors such as the student's year of study, academic standing, and roommate preferences.
3. **Maintenance:** The system will allow the administrator to track and manage maintenance requests. This includes requests for repairs, cleaning, and other services.
4. **Reporting:** The system will allow the administrator to generate reports on various aspects of the hostel, such as student attendance, room occupancy, and fee collection.

## System Implementation

The proposed system will be implemented in the following phases:

1. **Phase 1:** The system will be developed and tested.
2. **Phase 2:** The system will be deployed to the hostel.
3. **Phase 3:** The system will be used by the hostel staff and students.

## System Maintenance

The proposed system will be maintained by the hostel administrator. The administrator will be responsible for updating the system with new features and bug fixes.

## System Security

The proposed system will be secure. The system will use a variety of security measures to protect data, including:

1. **Authentication:** Users will be authenticated using a username and password.

2. **Authorization:** Users will be authorized to access only the data that they need to access.
3. **Encryption:** Data will be encrypted when it is stored or transmitted.
4. **Firewalls:** Firewalls will be used to protect the system from unauthorized access.

## System Benefits

The proposed system will provide a number of benefits to the hostel, including:

1. **Improved efficiency:** The system will improve the efficiency of the hostel by automating many of the tasks that were previously done manually. This will free up the hostel staff to focus on other important tasks.
2. **Improved accuracy:** The system will improve the accuracy of data by providing a single source of truth for all hostel records. This will help to reduce errors and improve decision-making.
3. **Enhanced security:** The system will enhance the security of the hostel by providing a centralized system for managing access control, video surveillance, and fire safety. This will help to protect students and staff from harm.
4. **Improved communication:** The system will improve communication between the hostel staff and students, parents, and other stakeholders. This will help to resolve issues quickly and efficiently.

# **CHAPTER-III**

## **DESIGN AND**

## **DEVELOPMENT PROCESS**

## 3.1 FUNDAMENTAL DESIGN CONCEPTS

### 3.1.1 MODULE AND FUNCTIONALITIES

These are two modules in this phase.

1. Administrator module.
2. User Module

**The functionality of each module is as follows:**

#### **1. Student module:**

1. Can submit the application form
2. Can view the notice board
3. Can submit the vacating form.

#### **2. Administrator module:**

The Administrator can :

1. Allot different students to the different hostels.
2. Vacate the students from the hostels.
3. Edit the details of the students & modify the student records.

#### **3. Hostel Manager Module**

1. Can submit the application form
2. Can view the notice board
3. Can submit the vacating form.

### 3.1.2 DESCRIPTION OF MODULES

A Hostel Management System (HMS) consists of various modules that collectively handle different aspects of hostel operations. Here are descriptions of some key modules in an HMS :

#### **1. Student Module :**

The student module in a hostel management system (HMS) is a software application that allows hostel administrators to track and manage student information, including:

1. **Personal information:** This includes the student's name, date of birth, contact information, and emergency contact information.

2. **Academic information:** This includes the student's academic records, such as grades, attendance, and transcripts.
3. **Medical information:** This includes the student's medical history, allergies, and medications.
4. **Financial information:** This includes the student's tuition, fees, and financial aid information.
5. **Other information:** This may include the student's extracurricular activities, housing preferences, and dietary restrictions.

The student module in an HMS can be used to:

1. **Generate reports:** The hostel administrator can use the student module to generate reports on student academic performance, financial aid status, and other areas.
2. **Communicate with students:** The hostel administrator can use the student module to communicate with students about important information, such as upcoming events, changes to policies, and deadlines.
3. **Provide students with access to their information:** The student module can be used to provide students with access to their personal information, such as grades, attendance, and financial aid information.

The student module in an HMS can help hostel administrators to improve the efficiency and effectiveness of their operations. By automating tasks, such as tracking attendance and generating reports, the student module can free up hostel administrators' time so they can focus on other important tasks.

Additionally, the student module can help hostel administrators to improve communication with students and provide them with access to their information.

**Here are some of the benefits of using a student module in an HMS:**

1. **Improved efficiency:** The student module can help hostel administrators to automate tasks, such as tracking attendance and generating reports. This can free up hostel administrators' time so they can focus on other important tasks.
2. **Improved communication:** The student module can help hostel administrators to improve communication with students. Hostel

administrators can use the student module to send messages to students, update them on important information, and answer their questions.

3. **Improved student satisfaction:** Students can access their information through the student module. This can help students to stay organized and on top of their studies.

If you are considering implementing an HMS, it is important to choose a system that has a student module. The student module can be a valuable tool for hostel administrators and students alike.

## 2. Admin Module :

The admin module is a critical part of any hostel management system. It allows the administrator to manage all aspects of the hostel, including:

1. **Student records:** The admin can create, edit, and delete student records. This includes information such as the student's name, contact information, academic records, and hostel room assignment.
2. **Room assignments:** The admin can assign students to rooms. This can be done manually or automatically, based on factors such as the student's year of study, academic standing, and roommate preferences.
3. **Maintenance:** The admin can track and manage maintenance requests. This includes requests for repairs, cleaning, and other services.
4. **Reporting:** The admin can generate reports on various aspects of the hostel, such as student attendance, room occupancy, and fee collection.

The admin module is a powerful tool that can help to improve the efficiency and effectiveness of any hostel management system. By giving the administrator the ability to manage all aspects of the hostel from a single location, the admin module can help to ensure that the hostel is running smoothly and efficiently.

**Here are some of the benefits of having an admin module in a hostel management system:**

1. **Improved efficiency:** The admin module can help to improve the efficiency of the hostel by automating many of the tasks that were

previously done manually. This can free up the administrator's time to focus on other important tasks.

2. **Increased accuracy:** The admin module can help to improve the accuracy of data by providing a single source of truth for all hostel records. This can help to reduce errors and improve decision-making.
3. **Enhanced security:** The admin module can help to enhance the security of the hostel by providing a centralized system for managing access control, video surveillance, and fire safety. This can help to protect students and staff from harm.
4. **Improved communication:** The admin module can help to improve communication between the administrator and students, staff, and parents. This can help to resolve issues quickly and efficiently.

Overall, the admin module is a valuable tool that can help to improve the efficiency, accuracy, security, and communication of any hostel management system.

### 3. Hostel Manager Module :

The hostel manager module in a hostel management system is responsible for room allocation. This module typically includes the following features:

1. **Room availability:** The module tracks the availability of rooms in the hostel. This information is used to determine which students can be assigned to a room.
2. **Room preferences:** The module allows students to specify their room preferences. This information is used to try to assign students to rooms that meet their preferences.
3. **Room allocation rules:** The module can be configured with rules for assigning students to rooms. These rules can be based on factors such as the student's year of study, academic standing, and roommate preferences.
4. **Room assignment notifications:** The module can send notifications to students when they have been assigned a room.

The hostel manager module is an important part of any hostel management system. It allows the hostel manager to efficiently and fairly allocate rooms to students.

**Here are some of the benefits of having a hostel manager module in a hostel management system:**

1. **Efficiency:** The hostel manager module can help to improve the efficiency of the room allocation process by automating many of the tasks that were previously done manually. This can free up the hostel manager's time to focus on other important tasks.
2. **Fairness:** The hostel manager module can help to ensure that the room allocation process is fair by using rules that are based on objective factors. This can help to reduce the likelihood of students being assigned to rooms that they do not want.
3. **Transparency:** The hostel manager module can help to improve transparency by providing students with information about the room allocation process. This can help to reduce the likelihood of students feeling that they have been treated unfairly.

Overall, the hostel manager module is a valuable tool that can help to improve the efficiency, fairness, and transparency of the room allocation process in any hostel.

**Here are some additional tips for using the hostel manager module to improve room allocation:**

1. **Set clear rules for room allocation:** The hostel manager should set clear rules for room allocation. These rules should be based on objective factors, such as the student's year of study, academic standing, and roommate preferences.
2. **Communicate the rules to students:** The hostel manager should communicate the rules for room allocation to students. This can be done through the hostel's website, email, or in person.
3. **Allow students to specify their room preferences:** The hostel manager should allow students to specify their room preferences. This information can be used to try to assign students to rooms that meet their preferences.
4. **Monitor the room allocation process:** The hostel manager should monitor the room allocation process to ensure that it is fair and efficient.

This can be done by reviewing the rules for room allocation, communicating with students, and resolving any issues that arise.

## 3.2 DESIGN NOTATIONS

### 3.2.1 DFD SYMBOLS

**DFD Symbols:** A data flow diagram shows how data is processed within a system based on inputs and outputs. Visual symbols are used to represent the flow of information, data sources and destinations, and where data is stored. Data flow diagrams are often used as a first step toward redesigning a system. They provide a graphical representation of a system at any level of detail, creating an easy-to-understand picture of what the system does. A general overview of a system is represented with a context diagram, also known as a level 0 DFD, which shows a system as a single process. A level 1 diagram provides greater detail, focusing on a system's main functions. Diagrams that are level 2 or higher illustrate a system's functioning with increasing detail. It's rare for a DFD to go beyond level 2 because of the increasing complexity, which makes it less effective as a communication tool.

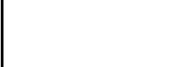
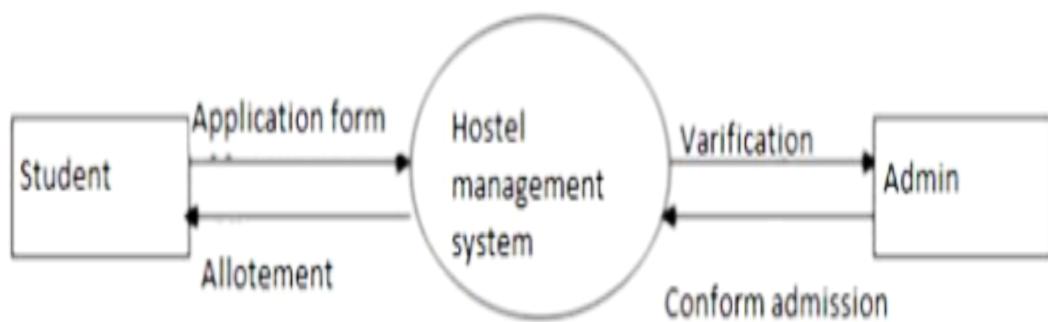
	<b>dataflow</b>	<b>Arrows showing direction of flow</b>
	<b>process</b>	<b>Ovals</b>
	<b>Data-source, sink</b>	<b>Rectangular box</b>

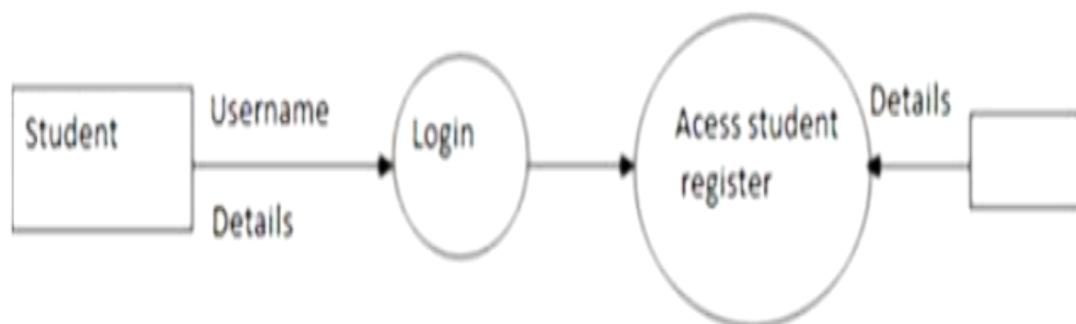
Table : DFD Symbols

### 3.2.2 DATA FLOW DIAGRAM

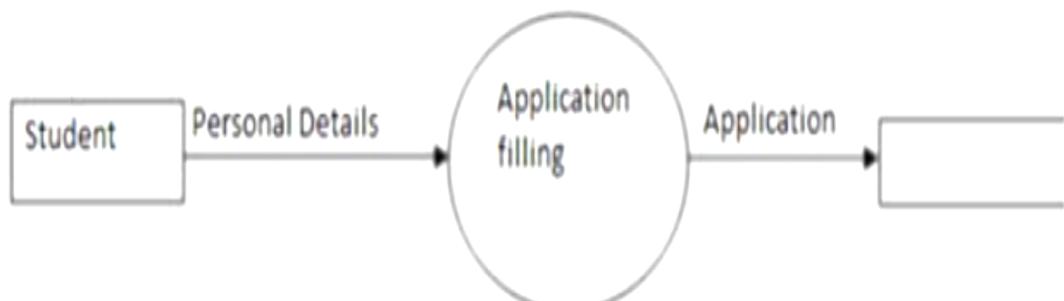
Data Flow Diagram for Allotment Process



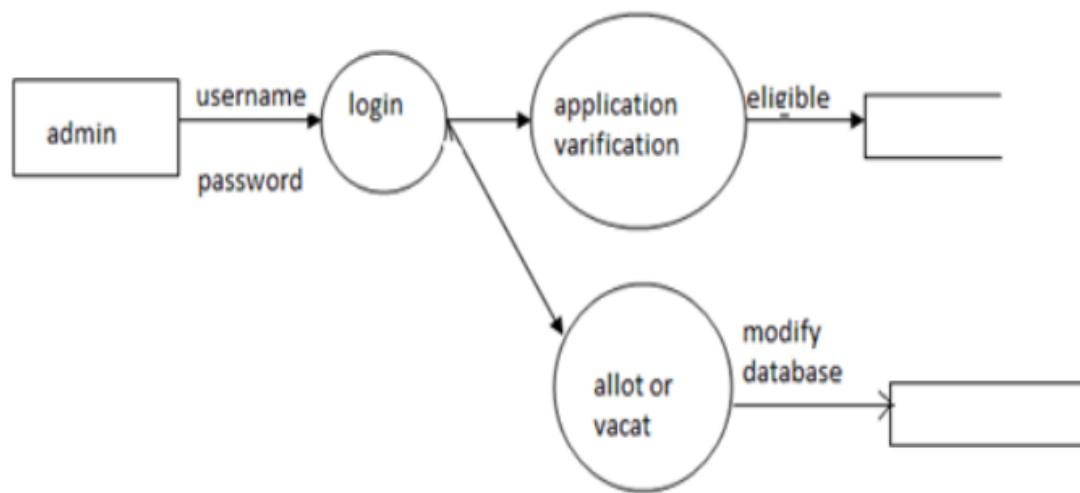
Data Flow Diagram For Student Module



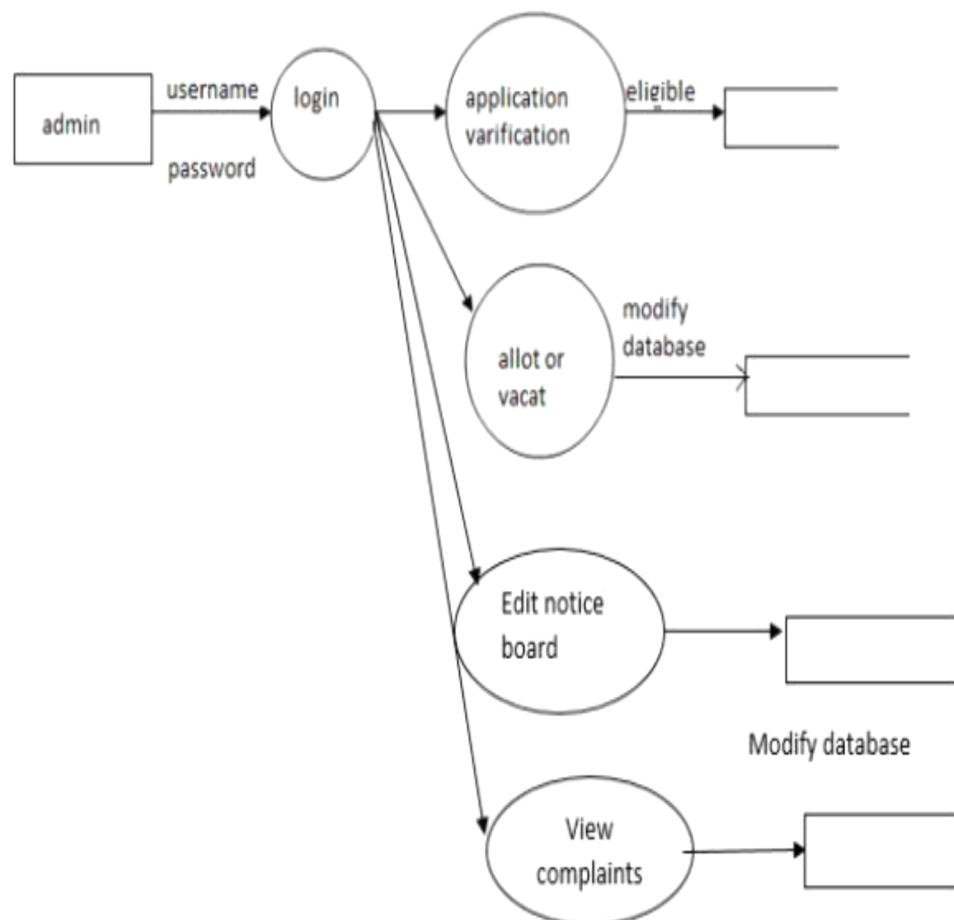
Data Flow Diagram For Student Registration



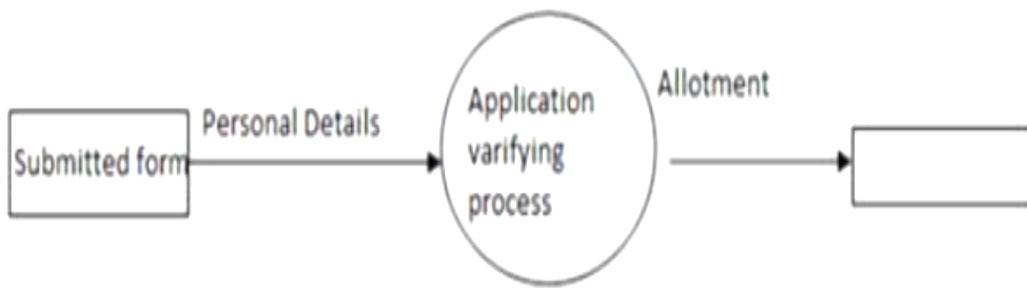
### Data Flow Diagram For Hostel Manager/ Admin Module



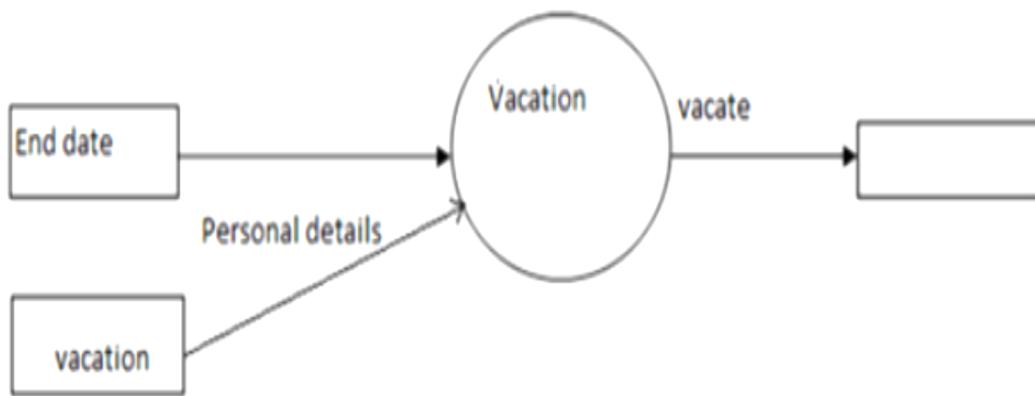
### Data Flow Diagram For Admin Module



### Data Flow Diagram For Allotment Process



### Data Flow Diagram For Vacation Process



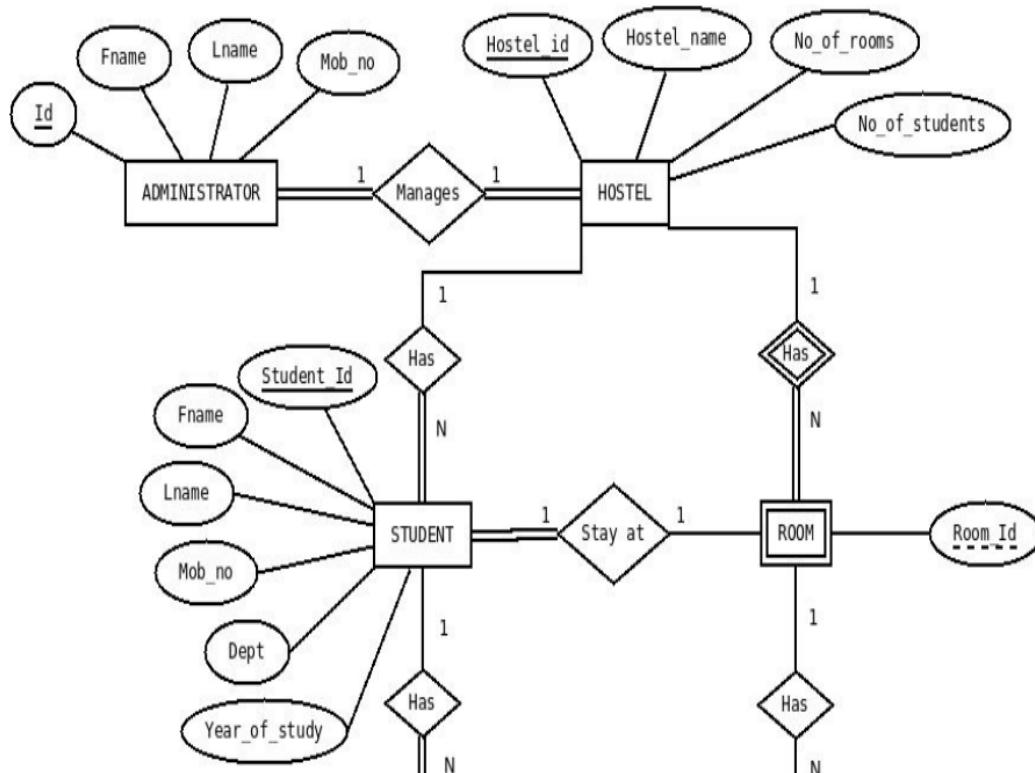
### 3.2.3 ER DIAGRAM

E-R (Entity-Relationship) Diagram is used to represent the relationship between entities in a table.

E-R diagram means Entity Relationship diagram. Entity means object of a system, generally we refer entity as database table , the e-r diagram represents the relationship between each table of the database. E-R diagrams represent entities with attributes, attributes are the properties of an entity. If we assume an entity is a database table then all the columns of the table are treated as attributes.

**E-R Diagram Symbols :-**

Name	Symbol	Meaning
Oval	○	Shows different attributes
Rectangle	□	Shows entity set
Diamond	◇	Show relationship among entity set
Line	—	Links entity set to attributes & entity set to relationship

**ER DIAGRAM FOR HOSTEL MANAGEMENT SYSTEM**

### 3.3 DESIGN PROCESS

#### 3.3.1 DATABASE DESIGN

##### DATABASE DESIGN TABLE STRUCTURE

###### 1. APPLICATION TABLE

Column	Type	Attributes	Null	Default	Extra	Links to	Comments	MIME
Application_id	int(100)		No		auto_increment			
Student_id	varchar(255)		No			-> student.Student_id ON UPDATE RESTRICT ON DELETE RESTRICT		
Hostel_id	int(10)		No			-> hostel.Hostel_id ON UPDATE RESTRICT ON DELETE RESTRICT		
Application_status	tinyint(1)		Yes	NULL				
Room_No	int(10)		Yes	NULL				
Message	varchar(255)		Yes	NULL				

###### 2. HOSTEL TABLE

Column	Type	Attributes	Null	Default	Extra	Links to	Comments	MIME
Hostel_id	int(10)		No		auto_increment			
Hostel_name	varchar(255)		No					
current_no_of_rooms	varchar(255)		Yes	NULL				
No_of_rooms	varchar(255)		Yes	NULL				
No_of_students	varchar(255)		Yes	NULL				

### 3. HOSTEL MANAGER TABLE

Column	Type	Attributes	Null	Default	Extra	Links to	Comments	MIME
Hostel_man_id	int(10)		No		auto_increment			
Username	varchar(255)		No					
Fname	varchar(255)		No					
Lname	varchar(255)		No					
Mob_no	varchar(255)		No					
Hostel_id	int(10)		No			-> hostel.Hostel_id ON UPDATE RESTRICT ON DELETE RESTRICT		
Email	varchar(30)		No					
Pwd	longtext		No					
Isadmin	tinyint(1)		Yes	0				

### 4. MESSAGE TABLE

Column	Type	Attributes	Null	Default	Extra	Links to	Comments	MIME
msg_id	int(10)		No		auto_increment			
sender_id	varchar(255)		Yes	NULL				
receiver_id	varchar(255)		Yes	NULL				
hostel_id	int(10)		Yes	NULL		-> hostel.Hostel_id ON UPDATE RESTRICT ON DELETE RESTRICT		
subject_h	varchar(255)		Yes	NULL				
message	varchar(255)		Yes	NULL				
msg_date	varchar(255)		Yes	NULL				
msg_time	varchar(255)		Yes	NULL				

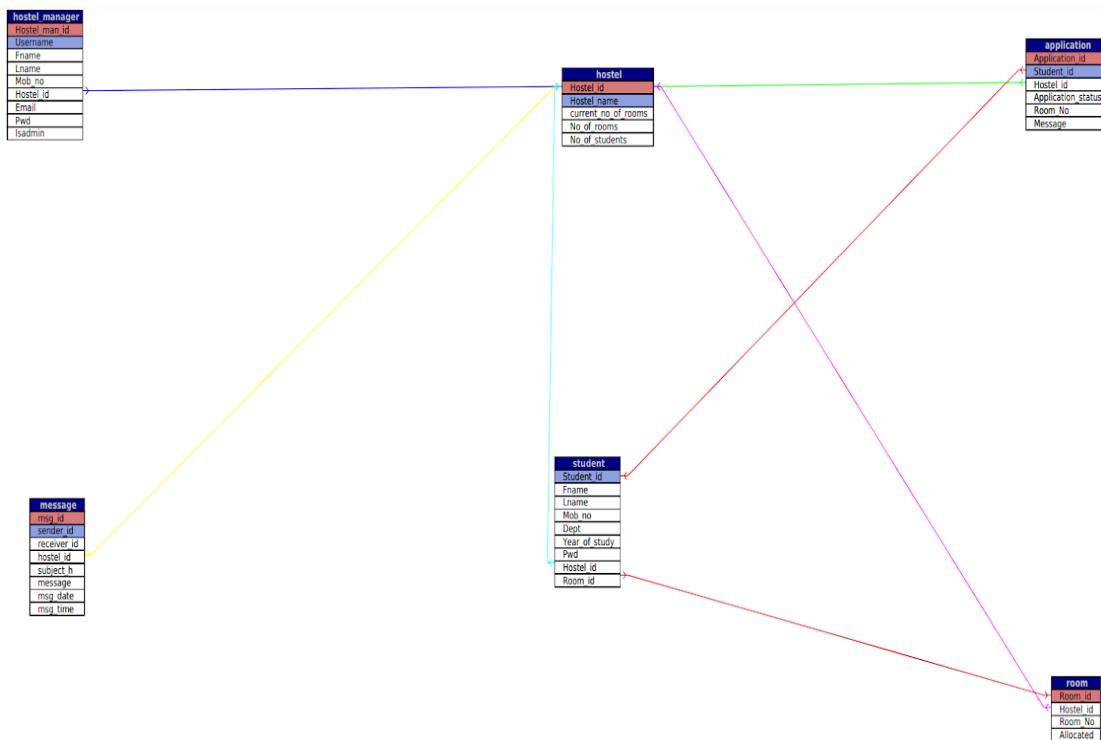
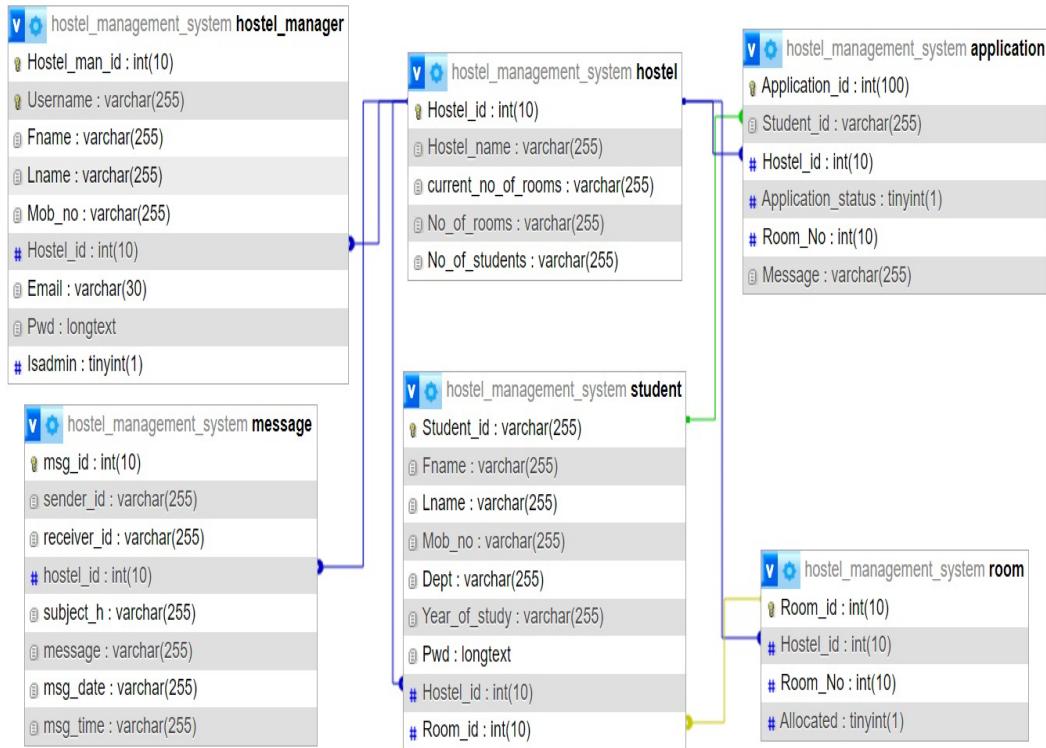
## 5. ROOM TABLE

Column	Type	Attributes	Null	Default	Extra	Links to	Comments	MIME
Room_id	int(10)		No		auto_increment			
Hostel_id	int(10)		No			-> hostel.Hostel_id ON UPDATE RESTRICT ON DELETE RESTRICT		
Room_No	int(10)		No					
Allocated	tinyint(1)		Yes	0				

## 6. STUDENT TABLE

Column	Type	Attributes	Null	Default	Extra	Links to	Comments	MIME
Student_id	varchar(255)		No					
Fname	varchar(255)		No					
Lname	varchar(255)		No					
Mob_no	varchar(255)		No					
Dept	varchar(255)		No					
Year_of_study	varchar(255)		No					
Pwd	longtext		No					
Hostel_id	int(10)		Yes	NULL		-> hostel.Hostel_id ON UPDATE RESTRICT ON DELETE RESTRICT		
Room_id	int(10)		Yes	NULL		-> room.Room_id ON UPDATE RESTRICT ON DELETE RESTRICT		

### 3.3.2 SCHEMA DIAGRAM



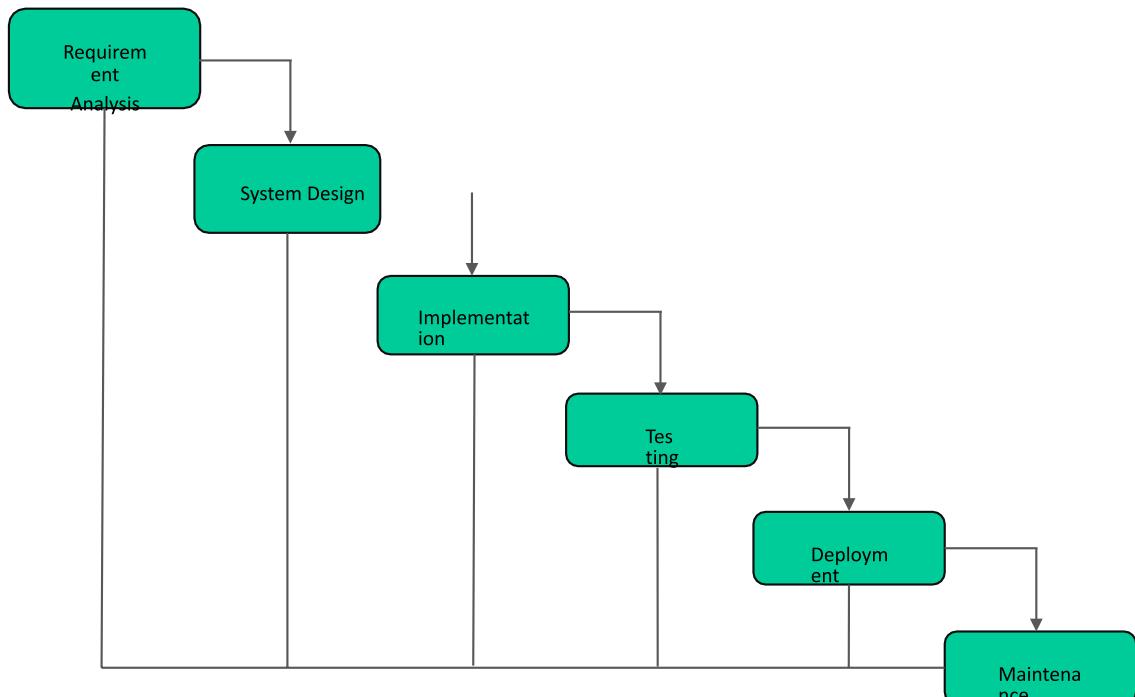
## 3.4 DEVELOPMENT APPROACH

### 3.4.1 DEVELOPMENT MODEL/APPROACH

We are using the waterfall model to build our project. As our project is not highly complicated and big, we are using the waterfall model for the development of our project.

#### Waterfall Model

Waterfall approach was first SDLC Model to be used widely in Software Engineering to ensure success of the project. In "The Waterfall" approach, the whole process of software development is divided into separate phases. In this Waterfall model, typically, the outcome of one phase acts as the input for the next phase sequentially.



The sequential phases in Waterfall model are –

1. **Requirement Gathering and analysis** – All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification document. We understood the different concepts of our project and gathered all the required software, modules and codes that was needed for the project.
2. **System Design** – The requirement specifications from first phase are studied in this phase and the system design is prepared. This system design helps in specifying hardware and system requirements and helps in defining the overall

system architecture. We studied the requirements of the system for our project and designed a blue print for our project.

3. **Implementation** – With inputs from the system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality, which is referred to as Unit Testing. We build our project step-by-step. Firstly we build small units of our project and then we merged it into one final software.
4. **Integration and Testing** – All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures. After successfully building up the project we tested each and every part of the software in all possible ways.
5. **Deployment of system** – Once the functional and nonfunctional testing is done; the product is deployed in the customer environment or released into the market. After the testing phase was completed we deployed the software in another system and tested whether it can run smoothly or not.
6. **Maintenance** – There are some issues which come up in the client environment. To fix those issues, patches are released. Also to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment. When the software worked properly we tested it for any issues. There was some minor errors and changes that we fixed to run the software into another system.

# **CHAPTER-1V**

## **PLANNING**

## 4.1 GANTT CHART

Gantt charts are useful for planning and scheduling projects. They help you assess how long a project should take, determine the resources needed, and plan the order in which tasks have to be completed.

Start Date – **13 Feb 2023**

End Date – **09 June 2023**



# **CHAPTER-V**

## **TESTING AND**

## **IMPLEMENTATION**

## 5.1 TESTING

### 5.1.1 TESTING METHODOLOGIES

System testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before live operation commences. Testing is the process of executing the program with the intent of finding errors and missing operations and also a complete verification to determine whether the objectives are met and the user requirements are satisfied. The ultimate aim is quality assurance.

Tests are carried out and the results are compared with the expected document. In the case of erroneous results, debugging is done. Using detailed testing strategies a test plan is carried out on each module. The various tests performed in “Network Backup System” are unit testing, integration testing and user acceptance testing.

### 5.1.2 TEST CASES

#### Unit Testing

The software units in a system are modules and routines that are assembled and integrated to perform a specific function. Unit testing focuses first on modules, independently of one another, to locate errors. This enables, to detect errors in coding and logic that are contained within each module. This testing includes entering data and ascertaining if the value matches to the type and size supported by java. The various controls are tested to ensure that each performs its action as required.

#### Integration Testing

Data can be lost across any interface, one module can have an adverse effect on another, subfunctions when combined, may not produce the desired major functions. Integration testing is a systematic testing to discover errors associated within the interface. The objective is to take unit test modules and build a program structure. All the modules are combined and tested as a whole. Here the Server module and Client module options are integrated and tested. This testing provides the assurance that the application is a well integrated functional unit with smooth transition of data.

#### User Acceptance Testing

User acceptance of a system is the key factor for the success of any system. The system under consideration is tested for user acceptance by constantly keeping in

touch with the system users at time of developing and making changes whenever required.

## 5.2 SYSTEM IMPLEMENTATION

### 5.2.1 IMPLEMENTATION PROCEDURES

Implementation is the stage in the project where the theoretical design is turned into a working system and is giving confidence on the new system for the users that it will work efficiently and effectively. It involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the change over, and evaluation of change over methods. Apart from planning, major tasks of preparing the implementation are education and training of users. The implementation process begins with preparing a plan for the implementation of the system. According to this plan, the activities are to be carried out, discussions made regarding the equipment and resources and the additional equipment has to be acquired to implement the new system. In a network backup system no additional resources are needed.

Implementation is the final and the most important phase. The most critical stage in achieving a successful new system is giving the users confidence that the new system will work and be effective. The System can be implemented only after thorough testing is done and if it is found to be working according to the specification. This method also offers the greatest security since the old system can take over if the errors are found or inability to handle certain type of transactions while using the new system.

### 5.2.2 USER TRAINING

After the system is implemented successfully, training of the user is one of the most important subtasks of the developer. For this purpose user manuals are prepared and handled over to the user to operate the developed system. Thus the users are trained to operate the developed system. Both the hardware and software securities are made to run the developed systems successfully in future. In order to put new application system into use, the following activities were taken care of:-

1. Preparation of user and system documentation.
2. Conducting user training with demo and hands on uses.

3. Test run for some period to ensure smooth switching over the system

The users are trained to use the newly developed functions. User manuals describing the procedures for using the functions listed on the menu are circulated to all the users. It is confirmed that the system is implemented up to the user's needs and expectations.

### **5.3 SYSTEM MAINTENANCE**

Maintenance involves the software industry captive, typing up system resources .It means restoring something to its original condition. Maintenance follows conversion to the extent that changes are necessary to maintain satisfactory operations relative to changes in the user's environment. Maintenance often includes minor enhancements or corrections to problems that surface in the system's operation. Maintenance is also done based on fixing the problems reported, changing the interface with other software or hardware enhancing the software.

Any system developed should be secured and protected against possible hazards.

Security measures are provided to prevent unauthorized access of the database at various levels. An uninterrupted power supply should be so that the power failure or voltage fluctuations will not erase the data in the files.

Password protection and simple procedures to prevent unauthorized access are provided to the users .The system allows the user to enter the system only through proper username and password.

## **CHAPTER-VI**

## **CONCLUSION**

## 6.1 CONCLUSION

To conclude the description about the project: The project, developed using PHP and MySQL is based on the requirement specification of the user and the analysis of the existing system, with flexibility for future enhancement.

The expanded functionality of today's software requires an appropriate approach towards software development. This hostel management software is designed for people who want to manage various activities in the hostel. For the past few years the number of educational institutions has been increasing rapidly.

Thereby the number of hostels are also increasing for the accommodation of the students studying in this institution. And hence there is a lot of strain on the person who is running the hostel and software's are not usually used in this context. 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 a computerized system that will be compatible with the existing system with the system which is more user friendly and more GUI oriented.

## 6.2 SCOPE OF FUTURE ENHANCEMENT

1. **Automate tasks:** Many tasks in hostel management, such as room allocation, bill generation, and complaint handling, can be automated. This will free up staff time to focus on other tasks, such as student support.
2. **Improve communication:** A good hostel management system should provide a way for students, staff, and parents to communicate with each other easily. This can be done through a messaging system, a forum, or a social media platform.
3. **Provide more data insights:** A good hostel management system should provide data insights that can help the hostel improve its operations. This data could include information on student attendance, room occupancy, and food consumption.
4. **Make the system mobile-friendly:** More and more people are using mobile devices, so it's important to make the hostel management system mobile

friendly. This will allow students, staff, and parents to access the system from anywhere.

5. **Integrate with other systems:** A hostel management system can be integrated with other systems, such as a student information system or a financial management system. This will allow for seamless data transfer and sharing.
6. **Use artificial intelligence (AI):** AI can be used to automate tasks, improve decision-making, and personalize the user experience. For example, AI could be used to automatically generate room assignments, identify potential problems, or recommend resources to students.
7. **Use big data analytics:** Big data analytics can be used to gain insights into student behavior, trends, and preferences. This information can be used to improve the hostel experience and make better decisions about operations.

## **ANNEXURE -A**

## **SAMPLE CODE**

# Index.php(Student login)

```
<!DOCTYPE html>
<html lang="en">

<head>
    <title>HMS</title>
    <!-- meta tags -->
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width,
initial-scale=1.0" />
    <!-- /meta tags -->
    <!-- custom style sheet -->
    <link href="web/css/style.css" rel="stylesheet" type="text/css"
/>
    <!-- /custom style sheet -->
    <!-- fontawesome css -->
    <link href="web/css/fontawesome-all.css" rel="stylesheet" />
    <!-- /fontawesome css -->
    <!-- google fonts-->
    <link
        href="//fonts.googleapis.com/css?family=Raleway:100,100i,200,200i,300
,300i,400,400i,500,500i,600,600i,700,700i,800,800i,900,900i"
        rel="stylesheet">
    <!-- /google fonts-->

</head>
<body>
    <h1>Hostel Room Allocation System</h1>
    <div class=" w3l-login-form">
        <h2>Student Login</h2>
        <form action="includes/login.inc.php" method="POST">

            <div class=" w3l-form-group">
                <label>Student Roll No:</label>
                <div class="group">
                    <i class="fas fa-user"></i>
                    <input type="text" class="form-control"
name="student_roll_no" placeholder="Roll No" required="required" />
                </div>
            </div>
            <div class=" w3l-form-group">
                <label>Password:</label>
                <div class="group">
                    <i class="fas fa-unlock"></i>
```

```

        <input type="password" class="form-control" name="pwd" placeholder="Password" required="required" />
            </div>
        </div>
        <button type="submit" name="login-submit">Login</button>
    </form>
    <p class="w3l-register-p">Login as<a href="login-hostel_manager.php" class="register">Hostel-Manager/Admin</a></p>
    <p class="w3l-register-p">Don't have an account?<a href="signup.php" class="register"> Sign up</a></p>
    </div>
    <footer>
        <p class="copyright-agileinfo"> © 2023 Project. All Rights Reserved | Design by Rohit-Aman-Anurag-Saniya</a></p>
    </footer>
</body>
</html>

```

## Signup.php (Student Signup)

```

<!DOCTYPE html>
<html lang="en">
<head>
    <title>SIGNUP PAGE</title>
    <!-- meta tags -->
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <meta name="keywords" content="" />
    <!-- /meta tags -->
    <!-- custom style sheet -->
    <link href="web/css/style.css" rel="stylesheet" type="text/css" />
    <!-- /custom style sheet -->
    <!-- fontawesome css -->
    <link href="web/css/fontawesome-all.css" rel="stylesheet" />
    <!-- /fontawesome css -->
    <!-- google fonts-->
    <link href="https://fonts.googleapis.com/css?family=Raleway:100,100i,200,200i,300,300i,400,400i,500,500i,600,600i,700,700i,800,800i,900,900i" rel="stylesheet">
    <!-- /google fonts-->

```

```
</head>
<body>
    <h1>Hostel Room Allocation System</h1>
    <div class=" w3l-login-form">
        <h2>Sign Up Here</h2>
        <form action="includes/signup.inc.php" method="POST">

            <div class=" w3l-form-group">
                <label>Student Roll No</label>
                <div class="group">
                    <i class="fas fa-id-badge"></i>
                    <input type="text" class="form-control" name="student_roll_no" placeholder="Roll No" required="required" />
                </div>
            </div>
            <div class=" w3l-form-group">
                <label>First Name</label>
                <div class="group">
                    <i class="fas fa-user"></i>
                    <input type="text" class="form-control" name="student_fname" placeholder="First Name" required="required" />
                </div>
            </div>
            <div class=" w3l-form-group">
                <label>Last Name</label>
                <div class="group">
                    <i class="fas fa-user"></i>
                    <input type="text" class="form-control" name="student_lname" placeholder="Last Name" required="required" />
                </div>
            </div>
            <div class=" w3l-form-group">
                <label>Mobile No</label>
                <div class="group">
                    <i class="fas fa-phone"></i>
                    <input type="text" class="form-control" name="mobile_no" placeholder="Mobile No" required="required" />
                </div>
            </div>
            <div class=" w3l-form-group">
                <label>Department</label>
                <div class="group">
                    <i class="fas fa-graduation-cap"></i>
                    <input type="text" class="form-control" name="department" placeholder="Department" required="required" />
                </div>
            </div>
        </form>
    </div>

```

```
        </div>
    </div>
    <div class=" w3l-form-group">
        <label>Year of Study</label>
        <div class="group">
            <i class="fas fa-calendar"></i>
            <input type="text" class="form-control"
name="year_of_study" placeholder="Year of Study" required="required"
/>
        </div>
    </div>
    <div class=" w3l-form-group">
        <label>Password:</label>
        <div class="group">
            <i class="fas fa-unlock"></i>
            <input type="password" class="form-control"
name="pwd" placeholder="Password" required="required" />
        </div>
    </div>

    <div class=" w3l-form-group">
        <label>Confirm Password:</label>
        <div class="group">
            <i class="fas fa-unlock"></i>
            <input type="password" class="form-control"
name="confirmpwd" placeholder="Confirm Password" required="required"
/>
        </div>
    </div>
    <button type="submit" name="signup-submit">Sign
Up</button>
</form>
<p class=" w3l-register-p">Already a member?<a
href="index.php" class="register"> Login</a></p>
</div>
<footer>
    <p class="copyright-agileinfo"> &copy; 2023 Project. All
Rights Reserved | Design by Rohit-Aman-Anurag-Saniya</a></p>
</footer>
</body>
</html>
```

# Login-Hostel\_manager.php (Hostel Manager/ Admin Login)

```
<!DOCTYPE html>
<html lang="en">

<head>
    <title>HMS</title>
    <!-- meta tags -->
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width,
initial-scale=1.0" />
    <meta name="keywords" content="" />
    <!-- /meta tags -->
    <!-- custom style sheet -->
    <link href="web/css/style.css" rel="stylesheet" type="text/css" />
    <!-- /custom style sheet -->
    <!-- fontawesome css -->
    <link href="web/css/fontawesome-all.css" rel="stylesheet" />
    <!-- /fontawesome css -->
    <!-- google fonts-->
    <link
        href="https://fonts.googleapis.com/css?family=Raleway:100,100i,200,200i,300
,300i,400,400i,500,500i,600,600i,700,700i,800,800i,900,900i"
        rel="stylesheet">
    <!-- /google fonts-->

</head>
<body>
    <h1>Hostel Room Allocation System</h1>
    <div class=" w3l-login-form">
        <h2>Hostel-Manager/Admin Login</h2>
        <form action="includes/login-hm.inc.php" method="POST">

            <div class=" w3l-form-group">
                <label>Username:</label>
                <div class="group">
                    <i class="fas fa-user"></i>
                    <input type="text" class="form-control"
name="username" placeholder="Username" required="required" />
                </div>
            </div>
        </form>
    </div>
</body>

```

```

<div class=" w3l-form-group">
    <label>Password:</label>
    <div class="group">
        <i class="fas fa-unlock"></i>
        <input type="password" class="form-control" name="pwd" placeholder="Password" required="required" />
    </div>
</div>
<button type="submit" name="login-submit">Login</button>
</form>
<p class=" w3l-register-p">Login as<a href="index.php" class="register"> Student</a></p>
</div>
<footer>
    <p class="copyright-agileinfo"> &copy; 2023 Project. All Rights Reserved | Design by Rohit-Aman_Anurag-Saniya</a></p>
</footer>
</body>
</html>

```

## Admin\_home.php (Admin Home page)

```

<?php
    require '../includes/config.inc.php';
?>
<!DOCTYPE html>
<html lang="en">
<head>
<title>Admin Home</title>
    <!-- Meta tag Keywords -->
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta charset="utf-8">
    <script type="application/x-javascript">
        addEventListener("load", function () {
            setTimeout(hideURLbar, 0);
        }, false);

        function hideURLbar() {
            window.scrollTo(0, 1);
        }
    </script>
    <!--// Meta tag Keywords -->

```

```

        <link href="../../web_home/css_home/slider.css" type="text/css"
rel="stylesheet" media="all">

        <!-- css files -->
        <link rel="stylesheet" href="../../web_home/css_home/bootstrap.css">
<!-- Bootstrap-Core-CSS -->
        <link rel="stylesheet" href="../../web_home/css_home/style.css"
type="text/css" media="all" /> <!-- Style-CSS -->
        <link rel="stylesheet"
href="../../web_home/css_home/fontawesome-all.css"> <!--
Font-Awesome-Icons-CSS -->
        <!-- //css files -->

        <!-- testimonials css -->
        <link rel="stylesheet" href="../../web_home/css_home/flexslider.css"
type="text/css" media="screen" property="" /><!-- flexslider css -->
        <!-- //testimonials css -->

        <!-- web-fonts -->
        <link
href="//fonts.googleapis.com/css?family=Poiret+One&subset=cyrilli
c,latin-ext" rel="stylesheet">
        <!-- //web-fonts -->

</head>

<body>

<!-- banner -->
        <div class="banner" id="home">
            <div class="cd-radial-slider-wrapper">

<!--Header-->
<header>
        <div class="container agile-banner_nav">
            <nav class="navbar navbar-expand-lg navbar-light bg-light">

                <h1><a class="navbar-brand" href="admin_home.php">NTTF
<span class="display"></span></a></h1>
                <button class="navbar-toggler" type="button"
data-toggle="collapse" data-target="#navbarSupportedContent"
aria-controls="navbarSupportedContent" aria-expanded="false"
aria-label="Toggle navigation">
                    <span class="navbar-toggler-icon"></span>
                </button>

```

```
        <div class="collapse navbar-collapse
justify-content-center" id="navbarSupportedContent">
        <ul class="navbar-nav ml-auto">
            <li class="nav-item active">
                <a class="nav-link"
                href="admin_home.php">Home <span class="sr-only">(current)</span></a>
            </li>
            <li class="nav-item">
                <a class="nav-link"
                href="create_hm.php">Appoint/Remove Hostel Manager</a>
            </li>

            <li class="nav-item">
                <a class="nav-link"
                href="students.php">Students</a>
            </li>
            <!-- <li class="nav-item">
                <a class="nav-link"
                href="admin_contact.php">Contact</a>
            </li> -->
            <li class="dropdown nav-item">
                <a href="#" class="dropdown-toggle nav-link"
data-toggle="dropdown"><?php echo $_SESSION['username']; ?>
                    <b class="caret"></b>
                </a>
                <ul class="dropdown-menu
agile_short_dropdown">
                    <li>
                        <a href="admin_profile.php">My
Profile</a>
                    </li>
                    <li>
                        <a
                href="..../includes/logout.inc.php">Logout</a>
                    </li>
                </ul>
            </li>
        </ul>
    </div>
</header>
<!--Header-->
```

```
        <ul class="cd-radial-slider" data-radius1="60"
data-radius2="1364" data-centerx1="110" data-centerx2="1290">
            <li class="visible">
                <div class="svg-wrapper">
                    <svg viewBox="0 0 1400 800">
                        <title>Animated SVG</title>
                        <defs>
                            <clipPath id="cd-image-1">
                                <circle id="cd-circle-1" cx="110"
cy="400" r="1364"/>
                            </clipPath>
                        </defs>
                        <image height='800px' width="1400px"
clip-path="url(#cd-image-1)"
xlink:href="../web_home/images/1.png"></image>
                    </svg>
                </div> <!-- .svg-wrapper -->
                <div class="cd-radial-slider-content">
                    <div class="wrapper">
                        <div class="text-center">
                            <h2>Hostel Management </h2>
                            <h3> System </h3>

                        </div>
                    </div>
                </div> <!-- .cd-radial-slider-content -->
            </li>
            <li class="next-slide">
                <div class="svg-wrapper">
                    <svg viewBox="0 0 1400 800">
                        <title>Animated SVG</title>
                        <defs>
                            <clipPath id="cd-image-2">
                                <circle id="cd-circle-2"
cx="1290" cy="400" r="60"/>
                            </clipPath>
                        </defs>
                        <image height='800px' width="1400px"
clip-path="url(#cd-image-2)"
xlink:href="../web_home/images/2.png"></image>
                    </svg>
                </div> <!-- .svg-wrapper -->
                <div class="cd-radial-slider-content
text-center">
```

```
<div class="wrapper">
  <div class="text-center">
    <h3>Rooms</h3>

    </div>
  </div>
</div> <!-- .cd-radial-slider-content -->
</li>
<li>
  <div class="svg-wrapper">
    <svg viewBox="0 0 1400 800">
      <title>Animated SVG</title>
      <defs>
        <clipPath id="cd-image-3">
          <circle id="cd-circle-3" cx="110"
                cy="400" r="60"/>
        </clipPath>
      </defs>
      <image height='800px' width="1400px"
            clip-path="url(#cd-image-3)"
            xlink:href="../../web_home/images/3.png"></image>
    </svg>
  </div> <!-- .svg-wrapper -->
  <div class="cd-radial-slider-content
text-center">
    <div class="wrapper">
      <div class="text-center">
        <h3>Hostels </h3>

        </div>
      </div>
    </div> <!-- .cd-radial-slider-content -->
  </li>

</ul> <!-- .cd-radial-slider -->
<ul class="cd-radial-slider-navigation">
  <li><a href="#" class="next"><i class="fas
fa-chevron-right"></i></a></li>
  <li><a href="#" class="prev"><i class="fas
fa-chevron-left"></i></a></li>
</ul> <!-- .cd-radial-slider-navigation -->
</div> <!-- .cd-radial-slider-wrapper -->
</div>
```

```
<!-- //banner -->

<footer class="py-5">
    <div class="container py-md-5">
        <div class="footer-logo mb-5 text-center">
            <a class="navbar-brand" href="https://www.nttftrg.com/" target="_blank">NTTF<span class="display"> JAMSHEDPUR</span></a>
        </div>
        <div class="footer-grid">

            <div class="list-footer">
                <ul class="footer-nav text-center">
                    <li>
                        <a href="admin_home.php">Home</a>
                    </li>

                    <li>
                        <a href="create_hm.php">Appoint</a>
                    </li>
                    <li>
                        <a href="students.php">Student</a>
                    </li>
                    <li>
                        <a href="admin_profile.php">Profile</a>
                    </li>
                </ul>
            </div>
        </div>
    </div>
</footer>
<!-- footer -->
<!-- js-scripts -->
<!-- js -->
<script type="text/javascript"
src="../web_home/js/jquery-2.2.3.min.js"></script>
<script type="text/javascript"
src="../web_home/js/bootstrap.js"></script> <!-- Necessary-JavaScript-File-For-Bootstrap -->
<!-- //js -->
<!-- banner js -->
<script src="../web_home/js/snap.svg-min.js"></script>
<script src="../web_home/js/main.js"></script> <!-- Resource jQuery -->
<!-- //banner js -->
```

```
<!-- flexSlider --><!-- for testimonials -->
<script defer src="../web_home/js/jquery.flexslider.js"></script>
<script type="text/javascript">
    $(window).load(function() {
        $('.flexslider').flexslider({
            animation: "slide",
            start: function(slider) {
                $('body').removeClass('loading');
            }
        });
    });
</script>
<!-- //flexSlider --><!-- for testimonials -->

<!-- start-smoth-scrolling -->
<script src="../web_home/js/SmoothScroll.min.js"></script>
<script type="text/javascript"
src="../web_home/js/move-top.js"></script>
<script type="text/javascript"
src="../web_home/js/easing.js"></script>
<script type="text/javascript">
    jQuery(document).ready(function($) {
        $(".scroll").click(function(event){
            event.preventDefault();

            $('html,body').animate({scrollTop:$(this.hash).offset().top},1000);
        });
    });
</script>
<!-- here stars scrolling icon -->
<script type="text/javascript">
    $(document).ready(function() {

        $.UItoTop({ easingType: 'easeOutQuart' });

    });
</script>
<!-- //here ends scrolling icon -->
<!-- start-smoth-scrolling -->
<!-- //js-scripts -->
</body>
</html>
```

## Create\_hm.php (appoint hostel manager)

```
<?php
    require '../includes/config.inc.php';
?>
<!DOCTYPE html>
<html lang="en">
<head>
<title>User Profile</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<script type="application/x-javascript"> addEventListener("load",
function() { setTimeout(hideURLbar, 0); }, false);
    function hideURLbar(){ window.scrollTo(0,1); } </script>
<!-- js -->
<script src="../web_profile/js/jquery-2.1.3.min.js"
type="text/javascript"></script>
<script type="text/javascript"
src="../web_profile/js/sliding.form.js"></script>
<!-- //js -->
<link href="../web_profile/css/style.css" rel="stylesheet"
type="text/css" media="all" />
<link rel="stylesheet" href="../web_profile/css/font-awesome.min.css"
/>
<link rel="stylesheet" href="../web_profile/css/smoothbox.css"
type='text/css' media="all" />
<link href="//fonts.googleapis.com/css?family=Pathway+Gothic+One"
rel="stylesheet">
<link
href='//fonts.googleapis.com/css?family=Open+Sans:400,300,300italic,4
00italic,600,600italic,700,700italic,800,800italic' rel='stylesheet'
type='text/css'>

<script type="application/x-javascript">
    addEventListener("load", function () {
        setTimeout(hideURLbar, 0);
    }, false);

    function hideURLbar() {
        window.scrollTo(0, 1);
    }
</script>
<!--// Meta tag Keywords -->
```

```
<link href="../web_home/css_home/slider.css" type="text/css"
rel="stylesheet" media="all">

<!-- css files -->
<link rel="stylesheet" href="../web_home/css_home/bootstrap.css">
<!-- Bootstrap-Core-CSS -->
<link rel="stylesheet" href="../web_home/css_home/style.css"
type="text/css" media="all" /> <!-- Style-CSS -->
<link rel="stylesheet"
href="../web_home/css_home/fontawesome-all.css"> <!--
Font-Awesome-Icons-CSS -->
<!-- //css files -->

<!-- testimonials css -->
<link rel="stylesheet" href="../web_home/css_home/flexslider.css"
type="text/css" media="screen" property="" /><!-- flexslider css -->
<!-- //testimonials css -->

<!-- web-fonts -->
<link
href="//fonts.googleapis.com/css?family=Poiret+One&subset=cyrilli
c,latin-ext" rel="stylesheet">
<!-- //web-fonts -->

</head>
<body>
    <!-- banner -->
    <div class="banner" id="home">
        <div class="cd-radial-slider-wrapper">

            <!--Header-->
            <header>
                <div class="container agile-banner_nav">
                    <nav class="navbar navbar-expand-lg navbar-light
bg-light">

                        <h1><a class="navbar-brand"
href="admin_home.php">NTTF <span class="display"></span></a></h1>
                        <button class="navbar-toggler" type="button"
data-toggle="collapse" data-target="#navbarSupportedContent"
aria-controls="navbarSupportedContent" aria-expanded="false"
aria-label="Toggle navigation">
                            <span class="navbar-toggler-icon"></span>
                        </button>
```

```
        <div class="collapse navbar-collapse justify-content-center" id="navbarSupportedContent">
            <ul class="navbar-nav ml-auto">
                <li class="nav-item active">
                    <a class="nav-link"
                        href="admin_home.php">Home <span class="sr-only">(current)</span></a>
                </li>
                <li class="nav-item">
                    <a class="nav-link"
                        href="create_hm.php">Appoint Hostel Manager</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link"
                        href="students.php">Students</a>
                </li>
                <li class="dropdown nav-item">
                    <li>
                        <a
                            href="..../includes/logout.inc.php" class="nav-link">Logout</a>
                    </li>
                </ul>
            </li>
        </ul>
    </div>

    </nav>
</div>
</header>
<!--Header-->
<br><br><br><br><br>
<div class="main">
    <div id="navigation" style="display:none;" class="w3_agile">
        <ul>
            <li class="selected">
                <a href="#"><i class="fa fa-check" aria-hidden="true"></i><span>Appoint</span></a>
            </li>
            <li>
                <a href="#"><i class="fa fa-close" aria-hidden="true"></i><span>Remove</span></a>
            </li>
        </ul>
    </div>
    <div id="navigation" style="display:none;" class="w3_agile">
```

```
</div>
<div id="wrapper" class="w3ls_wrapper w3layouts_wrapper">
  <div id="steps" style="margin:0 auto;" class="agileits
w3_steps">
    <form id="formElem" name="formElem"
action="../includes/hm_signup.php" method="post" class="w3_form
w3l_form_fancy">

      <fieldset class="step w3_agileits">
        <legend>Appoint Hostel Manager</legend>
        <div class="agilecontactw3ls-grid">
          <div class="agile-con-centre">
            <form action="../includes/hm_signup.php"
method="POST">
              <input type="text" name="hm_uname"
placeholder="USERNAME" required="required">
              <input type="text" name="hm_fname"
placeholder="FIRST NAME" required="required">
              <input type="text" name="hm_lname"
placeholder="LAST NAME" required="required">
              <input type="text" name="hm_mobile"
placeholder="MOBILE NO" required="required">
              <input type="text" name="hostel_name"
placeholder="HOSTEL NAME" required="required">
              <input type="email" name="Email"
placeholder="EMAIL" required="required">
              <input type="password" name="pass"
placeholder="HOSTEL MANAGER'S PASSWORD" required="required">
              <input type="password" name="confpass"
placeholder="CONFIRM HOSTEL MANAGER'S PASSWORD" required="required">

              <div class="send-button">
                <input type="submit" name="hm_signup_submit">
              </div>
            </form>
          </div>
        <div class="clear"></div>
      </div>
    </fieldset>

    </form>
    <form id="formElem" name="formElem"
action="../includes/hm_remove.php" method="post" class="w3_form
w3l_form_fancy">
```

```

        <fieldset class="step w3_agileits">
            <legend>Remove Hostel Manager</legend>
            <div class="agilecontactw3ls-grid">
                <div class="agile-con-centre">
                    <form action="../includes/hm_remove.php"
method="POST">
                        <input type="text" name="hm_uname"
placeholder="USERNAME" required="required">
                        <input type="text" name="hostel_name"
placeholder="HOSTEL NAME" required="required">
                        <input type="password" name="pass"
placeholder="ADMIN PASSWORD" required="required">
                        <div class="send-button">
                            <input type="submit" name="hm_remove_submit">
                        </div>
                    </form>
                </div>
                <div class="clear"></div>
            </div>
        </fieldset>
        </form>

        </div>
    </div>
</script type="text/javascript"
src="../web_profile/js/smoothbox.jquery2.js"></script>
</body>
</html>

```

## Admin\_profile.php(Admin Profile)

```

<?php
    require '../includes/config.inc.php';
?>
<!DOCTYPE html>
<html lang="en">
<head>
    <title>User Profile</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <script type="application/x-javascript"> addEventListener("load",
function() { setTimeout(hideURLbar, 0); }, false);
        function hideURLbar(){ window.scrollTo(0,1); } </script>
<!-- js -->

```

```
<script src="../web_profile/js/jquery-2.1.3.min.js"
type="text/javascript"></script>
<script type="text/javascript"
src="../web_profile/js/sliding.form.js"></script>
<!-- //js -->
<link href="../web_profile/css/style.css" rel="stylesheet"
type="text/css" media="all" />
<link rel="stylesheet" href="../web_profile/css/font-awesome.min.css"
/>
<link rel="stylesheet" href="../web_profile/css/smoothbox.css"
type='text/css' media="all" />
<link href="//fonts.googleapis.com/css?family=Pathway+Gothic+One"
rel="stylesheet">
<link
href='//fonts.googleapis.com/css?family=Open+Sans:400,300,300italic,4
00italic,600,600italic,700,700italic,800,800italic' rel='stylesheet'
type='text/css'>

<script type="application/x-javascript">
    addEventListener("load", function () {
        setTimeout(hideURLbar, 0);
    }, false);

    function hideURLbar() {
        window.scrollTo(0, 1);
    }
</script>
<!--// Meta tag Keywords -->

<link href="../web_home/css_home/slider.css" type="text/css"
rel="stylesheet" media="all">

<!-- css files -->
<link rel="stylesheet" href="../web_home/css_home/bootstrap.css">
<!-- Bootstrap-Core-CSS -->
<link rel="stylesheet" href="../web_home/css_home/style.css"
type="text/css" media="all" /> <!-- Style-CSS -->
<link rel="stylesheet"
href="../web_home/css_home/fontawesome-all.css"> <!--
Font-Awesome-Icons-CSS -->
<!-- //css files -->

<!-- testimonials css -->
<link rel="stylesheet" href="../web_home/css_home/flexslider.css"
type="text/css" media="screen" property="" /><!-- flexslider css -->
```

```
<!-- //testimonials css -->

<!-- web-fonts -->
<link
  href="//fonts.googleapis.com/css?family=Poiret+One&subset=cyrilli
c,latin-ext" rel="stylesheet">
<!-- //web-fonts -->

</head>
<body>
  <!-- banner -->
  <div class="banner" id="home">
    <div class="cd-radial-slider-wrapper">

      <!--Header-->
      <header>
        <div class="container agile-banner_nav">
          <nav class="navbar navbar-expand-lg navbar-light
bg-light">

            <h1><a class="navbar-brand"
href="admin_home.php">NTTF <span class="display"> </span></a></h1>
            <button class="navbar-toggler" type="button"
data-toggle="collapse" data-target="#navbarSupportedContent"
aria-controls="navbarSupportedContent" aria-expanded="false"
aria-label="Toggle navigation">
              <span class="navbar-toggler-icon"></span>
            </button>

            <div class="collapse navbar-collapse
justify-content-center" id="navbarSupportedContent">
              <ul class="navbar-nav ml-auto">
                <li class="nav-item active">
                  <a class="nav-link"
href="admin_home.php">Home <span class="sr-only">(current)</span></a>
                </li>
                <li class="nav-item">
                  <a class="nav-link"
href="create_hm.php">Appoint/Remove Hostel Manager</a>
                </li>
                <li class="nav-item">
                  <a class="nav-link"
href="students.php">Students</a>
                </li>
              </ul>
            </div>
          </nav>
        </div>
      </header>
    </div>
  </div>
</body>
```

```

        </li>
        <!-- <li class="nav-item">
            <a class="nav-link"
href="admin_contact.php">Contact</a>
        </li> -->
        <li class="dropdown nav-item">
            <li>
                <a
href="..../includes/logout.inc.php" class="nav-link">Logout</a>
            </li>
        </ul>
    </li>
</ul>
</div>

</nav>
</div>
</header>
<!--Header-->
<br><br><br><br><br>
<div class="main">
    <div id="navigation" style="display:none;" class="w3_agile">

    </div>
    <div id="wrapper" class="w3ls_wrapper w3layouts_wrapper">
        <div id="steps" style="margin:0 auto;" class="agileits
w3_steps">
            <form id="formElem" name="formElem" action="#" method="post" class="w3_form w3l_form_fancy">
                <fieldset class="step agileinfo w3ls_fancy_step">
                    <legend>Personal Info</legend>
                    <div class="abt-agile">
                        <div class="abt-agile-left">
                        </div>
                        <div class="abt-agile-right">

                            <h3><?php echo $_SESSION['fname'] . "
".$_SESSION['lname'] ; ?></h3>
                            <h5>Admin</h5>
                            <ul class="address">
                                <li>
                                    <ul class="address-text">
                                        <li><b>Username </b></li>
                                        <li>: <?php echo
$_SESSION['username'] ; ?></li>
                                    </ul>
                                </li>
                            </ul>
                        </div>
                    </div>
                </fieldset>
            </form>
        </div>
    </div>
</div>

```

```

        </ul>
    </li>
    <li>
        <ul class="address-text">
            <li><b>PHONE </b></li>
            <li>: <?php echo
$_SESSION['mob_no']; ?></li>
        </ul>
    </li>
    <li>
        <ul class="address-text">
            <li><b>Email </b></li>
            <li>: <?php echo
$_SESSION['email']; ?></li>
        </ul>
    </li>
</ul>
</div>
<div class="clear"></div>
</div>
</fieldset>

</form>
</div>
</div>

</div>
<script type="text/javascript"
src="../web_profile/js/smoothbox.jquery2.js"></script>
</body>
</html>

```

## home\_manager.php(HostelManager Homepage)

```

<?php
    require 'includes/config.inc.php';
?>
<!DOCTYPE html>
<html lang="en">
<head>
<title> Home Manager</title>

<!-- Meta tag Keywords -->

```



```

        </li>
        <li>
            <a href="empty_rooms.php">Empty
Rooms</a>
        </li>
        <li>
            <a href="vacate_rooms.php">Vacate
Rooms</a>
        </li>
    </ul>
</li>
<li class="nav-item">
    <a class="nav-link"
href="contact_manager.php">Contact</a>
    </li>
    <li class="dropdown nav-item">
        <a href="#" class="dropdown-toggle nav-link"
data-toggle="dropdown"><?php echo $_SESSION['username'] ; ?>
            <b class="caret"></b>
        </a>
        <ul class="dropdown-menu
agile_short_dropdown">
            <li>
                <a
href="admin/manager_profile.php">My Profile</a>
            </li>
            <li>
                <a
href="includes/logout.inc.php">Logout</a>
            </li>
        </ul>
    </li>
</ul>
</div>

</nav>
</div>
</header>
<!--Header-->

<footer class="py-5">
    <div class="container py-md-5">
        <div class="footer-logo mb-5 text-center">
            <a class="navbar-brand" href="https://www.nttftrg.com/">
NTTF<span class="display">JAMESHEDPUR</span></a>
        </div>
    </div>
</footer>

```

```
        </div>
        <div class="footer-grid">

            <div class="list-footer">
                <ul class="footer-nav text-center">
                    <li>
                        <a href="home_manager.php">Home</a>
                    </li>
                    <li>
                        <a href="allocate_room.php">Allocate</a>
                    </li>
                    <li>
                        <a href="contact_manager.php">Contact</a>
                    </li>
                    <li>
                        <a href="manager_profile.php">Profile</a>
                    </li>
                </ul>
            </div>

        </div>
    </div>
</footer>
<!-- footer -->

<!-- js-scripts -->

<!-- js -->
<script type="text/javascript"
src="web_home/js/jquery-2.2.3.min.js"></script>
<script type="text/javascript"
src="web_home/js/bootstrap.js"></script> <!--
Necessary-JavaScript-File-For-Bootstrap -->
<!-- //js -->

<!-- banner js -->
<script src="web_home/js/snap.svg-min.js"></script>
<script src="web_home/js/main.js"></script> <!-- Resource jQuery
-->
<!-- //banner js -->

<!-- flexSlider --><!-- for testimonials -->
<script defer src="web_home/js/jquery.flexslider.js"></script>
<script type="text/javascript">
    $(window).load(function() {
```

```

        $('.flexslider').flexslider({
            animation: "slide",
            start: function(slider){
                $('body').removeClass('loading');
            }
        });
    });

```

```

</script>
<!-- //flexSlider --><!-- for testimonials -->
<!-- start-smoth-scrolling -->
<script src="web_home/js/SmoothScroll.min.js"></script>
<script type="text/javascript"
src="web_home/js/move-top.js"></script>
<script type="text/javascript"
src="web_home/js/easing.js"></script>
<script type="text/javascript">
    jQuery(document).ready(function($){
        $(".scroll").click(function(event){
            event.preventDefault();

        $('html,body').animate({scrollTop:$('this.hash').offset().top},1000);
    });
});

```

```

</script>
<!-- here stars scrolling icon -->
<script type="text/javascript">
    $(document).ready(function() {
        $.UItoTop({ easingType: 'easeOutQuart' });
    });

```

```

</script>
<!-- //here ends scrolling icon -->
<!-- start-smoth-scrolling -->
<!-- //js-scripts -->
</body>
</html>

```

## Home.php(Student Home)

```

<?php
    require 'includes/config.inc.php';
?>
<!DOCTYPE html>
<html lang="en">
<head>
<title>Student Home</title>

```

```
<!-- Meta tag Keywords -->
<meta name="viewport" content="width=device-width,
initial-scale=1">
<meta charset="utf-8">
<script type="application/x-javascript">
    addEventListener("load", function () {
        setTimeout(hideURLbar, 0);
    }, false);

    function hideURLbar() {
        window.scrollTo(0, 1);
    }
</script>
<!--// Meta tag Keywords -->

<link href="web_home/css_home/slider.css" type="text/css"
rel="stylesheet" media="all">
</head>

<body>

    <a class="nav-link" href="home.php">Home <span
class="sr-only">(current)</span></a>
    </li>

    <li class="nav-item">
        <a class="nav-link"
href="services.php">Hostels</a>
    </li>

    <li class="nav-item">
        <a class="nav-link"
href="contact.php">Contact</a>
    </li>
    <li class="nav-item">
        <a class="nav-link"
href="message_user.php">Message Received</a>
    </li>
    <li class="dropdown nav-item">
        <a href="#" class="dropdown-toggle nav-link"
data-toggle="dropdown"><?php echo $_SESSION['roll']; ?>
        <b class="caret"></b>
    </a>

```

```


- My Profile
- Logout

```

</nav>

</div>

</header>

<!--Header-->

<ul class="cd-radial-slider" data-radius1="60" data-radius2="1364" data-centerx1="110" data-centerx2="1290">

<li class="visible">

<div class="svg-wrapper">

<svg viewBox="0 0 1400 800">

<title>Animated SVG</title>

<defs>

<clipPath id="cd-image-1">

<circle id="cd-circle-1" cx="110" cy="400" r="1364"/>

</clipPath>

</defs>

<image height='800px' width="1400px"

clip-path="url(#cd-image-1)"

xlink:href="web\_home/images/1.png"></image>

</svg>

</div> <!-- .svg-wrapper -->

<div class="cd-radial-slider-content">

<div class="wrapper">

<div class="text-center">

<h2>Hostel Management </h2>

<h3> System </h3>

</div>

</div>

</div> <!-- .cd-radial-slider-content -->

```
</li>
<li class="next-slide">
  <div class="svg-wrapper">
    <svg viewBox="0 0 1400 800">
      <title>Animated SVG</title>
      <defs>
        <clipPath id="cd-image-2">
          <circle id="cd-circle-2"
            cx="1290" cy="400" r="60"/>
        </clipPath>
      </defs>
      <image height='800px' width="1400px"
        clip-path="url(#cd-image-2)"
        xlink:href="web_home/images/2.png"></image>
    </svg>
  </div> <!-- .svg-wrapper -->
  <div class="cd-radial-slider-content
text-center">
    <div class="wrapper">
      <div class="text-center">
        <h3>Rooms</h3>
      </div>
    </div>
  </div> <!-- .cd-radial-slider-content -->
</li>
<li>

<!-- footer -->
<footer class="py-5">
  <div class="container py-md-5">
    <div class="footer-logo mb-5 text-center">
      <a class="navbar-brand" href="https://www.nttftrg.com/">
        NTTF <span class="display"> JAMSHEDPUR</span>
      </a>
    </div>
    <div class="footer-grid">

      <div class="list-footer">
        <ul class="footer-nav text-center">
          <li>
            <a href="home.php">Home</a>
          </li>
        </ul>
      </div>
    </div>
  </div>
</footer>

```

```

<li>
    <a href="services.php">Hostels</a>
</li>
<li>
    <a href="contact.php">Contact</a>
</li>
<li>
    <a href="profile.php">Profile</a>
</li>
</ul>
</div>
</div>
</div>
<!-- footer -->

<!-- js-scripts -->

<!-- js -->
<script type="text/javascript"
src="web_home/js/jquery-2.2.3.min.js"></script>
<script type="text/javascript"
src="web_home/js/bootstrap.js"></script> <!--
Necessary-JavaScript-File-For-Bootstrap -->
<!-- //js -->
</body>
</html>

```

## Application\_form.php (Hostel Form)

```

<?php
    require 'includes/config.inc.php';
?>

<!DOCTYPE html>
<html lang="en">
<head>
<title> </title>

<!-- Meta tag Keywords -->
<meta name="viewport" content="width=device-width,
initial-scale=1">
<meta charset="utf-8">
</head>

```

```
<body>

<!-- banner -->
<div class="inner-page-banner" id="home">
    <!--Header-->
    <header>
        <div class="container agile-banner_nav">
            <nav class="navbar navbar-expand-lg navbar-light
bg-light">

                <h1><a class="navbar-brand" href="home.php">NTTF
<span class="display"></span></a></h1>
                <button class="navbar-toggler" type="button"
data-toggle="collapse" data-target="#navbarSupportedContent"
aria-controls="navbarSupportedContent" aria-expanded="false"
aria-label="Toggle navigation">
                    <span class="navbar-toggler-icon"></span>
                </button>

                <div class="collapse navbar-collapse
justify-content-center" id="navbarSupportedContent">
                    <ul class="navbar-nav ml-auto">
                        <li class="nav-item">
                            <a class="nav-link" href="home.php">Home
<span class="sr-only">(current)</span></a>
                        </li>
                        <li class="nav-item active">
                            <a class="nav-link"
href="services.php">Hostels</a>
                        </li>
                        <li class="nav-item">
                            <a class="nav-link"
href="contact.php">Contact</a>
                        </li>
                        <li class="nav-item">
                            <a class="nav-link"
href="message_user.php">Message Received</a>
                        </li>
                        <li class="dropdown nav-item">
                            <a href="#" class="dropdown-toggle nav-link"
data-toggle="dropdown">
                                <b class="caret"></b>
                            </a>
                            <ul class="dropdown-menu
agile_short_dropdown">

```

```
        <li>
            <a href="profile.php">My Profile</a>
        </li>
        <li>
            <a
href="includes/logout.inc.php">Logout</a>
        </li>
    </ul>
</li>
</ul>
</div>

</nav>
</div>

</div>
</section>

<!--footer-->
<footer class="py-5">
    <div class="container py-md-5">
        <div class="footer-logo mb-5 text-center">
            <a class="navbar-brand" href="https://www.nttftrg.com/" target="_blank">NTTF <span class="display"> JAMSHEDPUR</span></a>
        </div>
        <div class="footer-grid">

            <div class="list-footer">
                <ul class="footer-nav text-center">
                    <li>
                        <a href="home.php">Home</a>
                    </li>

                    <li>
                        <a href="services.php">Hostels</a>
                    </li>
                    <li>
                        <a href="contact.php">Contact</a>
                    </li>
                    <li>
                        <a href="profile.php">Profile</a>
                    </li>
                </ul>
            </div>
        </div>
    </div>
</footer>
```

```

        </div>
    </div>
</footer>

$query3 = "INSERT INTO Application
(Student_id,Hostel_id,Application_status,Message) VALUES
('{$roll}', '{$hostel_id}', true, '{$message}')";
$result3 = mysqli_query($conn,$query3);

if($result3) {
    echo "<script
type='text/javascript'>alert('Application sent
successfully')</script>";
}
}
}
}
else{
    echo "<script type='text/javascript'>alert('You have Already
applied for a Room')</script>";
}
else{
    echo "<script type='text/javascript'>alert('You have
Already been allotted a Room')</script>";
}
}
?>

```

## Allocate\_room.php

```

<?php
    require 'includes/config.inc.php';
?>
<!DOCTYPE html>
<html lang="en">
<head>
<title> Allocate Room</title>

<!-- Meta tag Keywords -->
<meta name="viewport" content="width=device-width,
initial-scale=1">
<meta charset="utf-8">
<script type="application/x-javascript">
    addEventListener("load", function () {
        setTimeout(hideURLbar, 0);

```

```
    }, false);

    function hideURLbar() {
        window.scrollTo(0, 1);
    }
</script>
<!--bootstrap -->

<!--// Meta tag Keywords -->

<!-- css files -->
<link rel="stylesheet"
href="web_home/css_home/bootstrap.css" > <!-- Bootstrap-Core-CSS
-->
<link rel="stylesheet" href="web_home/css_home/style.css"
type="text/css" media="all" /> <!-- Style-CSS -->
<link rel="stylesheet"
href="web_home/css_home/fontawesome-all.css" > <!--
class="nav-link" href="home_manager.php">Home <span
class="sr-only">(current)</span></a>
</li>
<li class="nav-item">
<a class="nav-link"
href="allocate_room.php">Allocate Room</a>
</li>
<li class="nav-item">
<a class="nav-link"
href="message_hostel_manager.php">Messages Received</a>
</li>
<li class="dropdown nav-item">
<a href="#" class="dropdown-toggle
nav-link" data-toggle="dropdown">Rooms
<b class="caret"></b>
</a>
<ul class="dropdown-menu
agile_short_dropdown">
<li>
<a
href="allocated_rooms.php">Allocated Rooms</a>
</li>
<li>
```

```
          <a href="empty_rooms.php">Empty
Rooms</a>
          </li>
          <li>
            <a href="vacate_rooms.php">Vacate
Rooms</a>
          </li>
        </ul>
      </li>
      <li class="nav-item">
        <a class="nav-link"
href="contact_manager.php">Contact</a>
      </li>
      <li class="dropdown nav-item">
        <a href="#" class="dropdown-toggle
nav-link" data-toggle="dropdown"><?php echo
$_SESSION['username']; ?>
          <b class="caret"></b>
        </a>
        <ul class="dropdown-menu
agile_short_dropdown">
          <li>
            <a
href="admin/manager_profile.php">My Profile</a>
          </li>
          <li>
            <a
href="includes/logout.inc.php">Logout</a>
          </li>
        </ul>
      </li>
    </ul>
  </div>
</nav>
</div>
</header>
<!--Header-->
</div>
<!-- //banner -->

<section class="contact py-5">
  <div class="container">
```



```

        echo
"<tr><td>{$student_name}</td><td>{$row_search['Student_id']}</td>
<td>{$hostel_name}</td><td>{$row_search['Message']}</td></tr>\n";

    }
}

?>
</tbody>
</table>
</div>
<?php
}

?>

<div class="container">
<h2 class="heading text-capitalize mb-sm-5 mb-4"> Applications
Received </h2>
<?php
$hostel_id = $_SESSION['hostel_id'];
$query1 = "SELECT * FROM Application where Hostel_id =
'$hostel_id' and Application_status = '1'";
$result1 = mysqli_query($conn,$query1);
//select the hostel name from hostel table
$query6 = "SELECT * FROM Hostel WHERE Hostel_id =
'$hostel_id'";
$result6 = mysqli_query($conn,$query6);
$row6 = mysqli_fetch_assoc($result6);
$hostel_name = $row6['Hostel_name'];
?>

<table class="table table-hover">
<thead>
<tr>
<th>Student Name</th>
<th>Student ID</th>
<th>Hostel</th>
<th>Message</th>
</tr>
</thead>
<tbody>
<?php
if(mysqli_num_rows($result1)==0) {
echo '<tr><td colspan="4">No Rows Returned</td></tr>';
}
}

```

```
        }
    </div>
<section class="contact py-5">
    <div class="container">
        <div class="mail_grid_w31">
            <form action="allocate_room.php" method="post">
                <div class="row">
                    <input type="submit" value="Allocate" name="submit">
                </div>
            </form>
        </div>
    </div>
</section>
<?php
if(isset($_POST['submit'])){
    $result1 = mysqli_query($conn,$query1);

    $query4 = "UPDATE Student SET Hostel_id =
'$hostel_id',Room_id = '$room_id' WHERE Student_id =
'$student_id';
    $result4 = mysqli_query($conn,$query4);
    if($result4){
        $query5 = "UPDATE Room SET Allocated = '1' where
Room_id = '$room_id'";
        $result5 = mysqli_query($conn,$query5);
        if($result5){
            echo "<script type='text/javascript'>alert('Rooms
Allocated Successfully')</script>";
        }
    }
    else{
        echo "<script type='text/javascript'>alert('Failed to
allocate Rooms')</script>";
    }
}
else{
    echo "<script type='text/javascript'>alert('Failed to
allocate Rooms')</script>";
}
}
?>
```

```
<br>
<br>
<br>

<!-- footer -->
<footer class="py-5">
    <div class="container py-md-5">
        <div class="footer-logo mb-5 text-center">
            <a class="navbar-brand" href="https://www.nttftrg.com/" target="_blank">NTTF<span class="display">JAMESHEDPUR</span></a>
        </div>
        <div class="footer-grid">

            <div class="list-footer">
                <ul class="footer-nav text-center">
                    <li>
                        <a href="home_manager.php">Home</a>
                    </li>
                    <li>
                        <a href="allocate_room.php">Allocate</a>
                    </li>

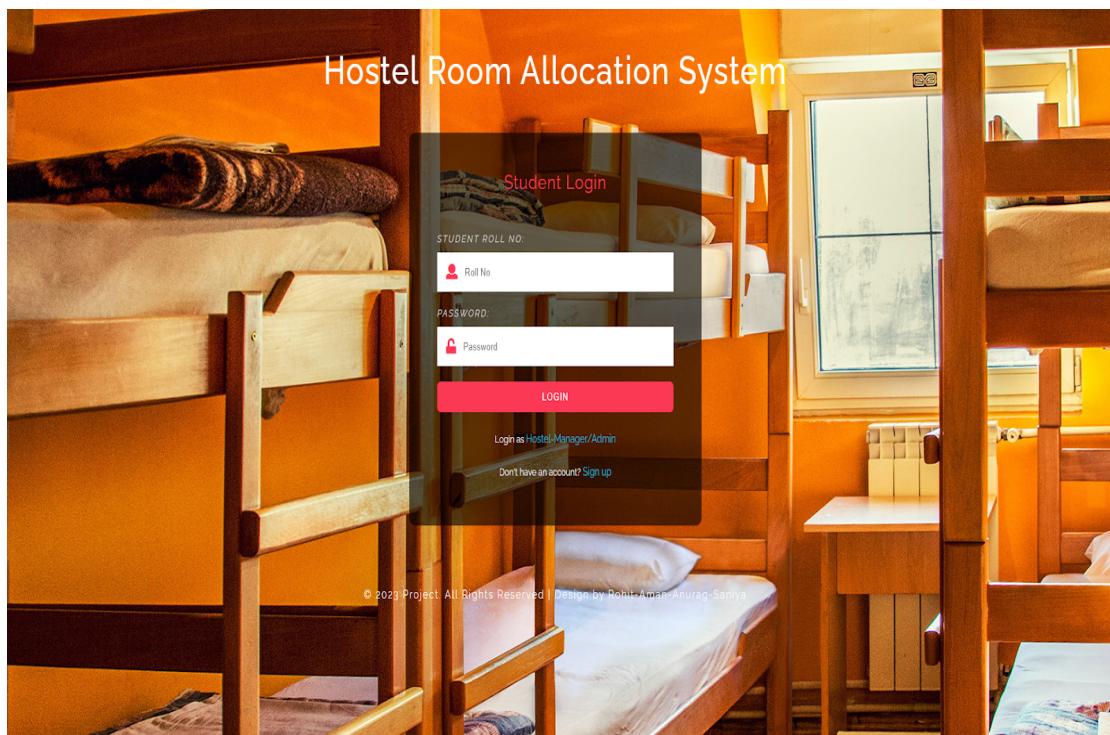
                    <li>
                        <a href="contact_manager.php">Contact</a>
                    </li>
                    <li>
                        <a href="admin/manager_profile.php">Profile</a>
                    </li>
                </ul>
            </div>

        </div>
    </div>
</footer>
<!-- footer -->
</body>
</html>
```

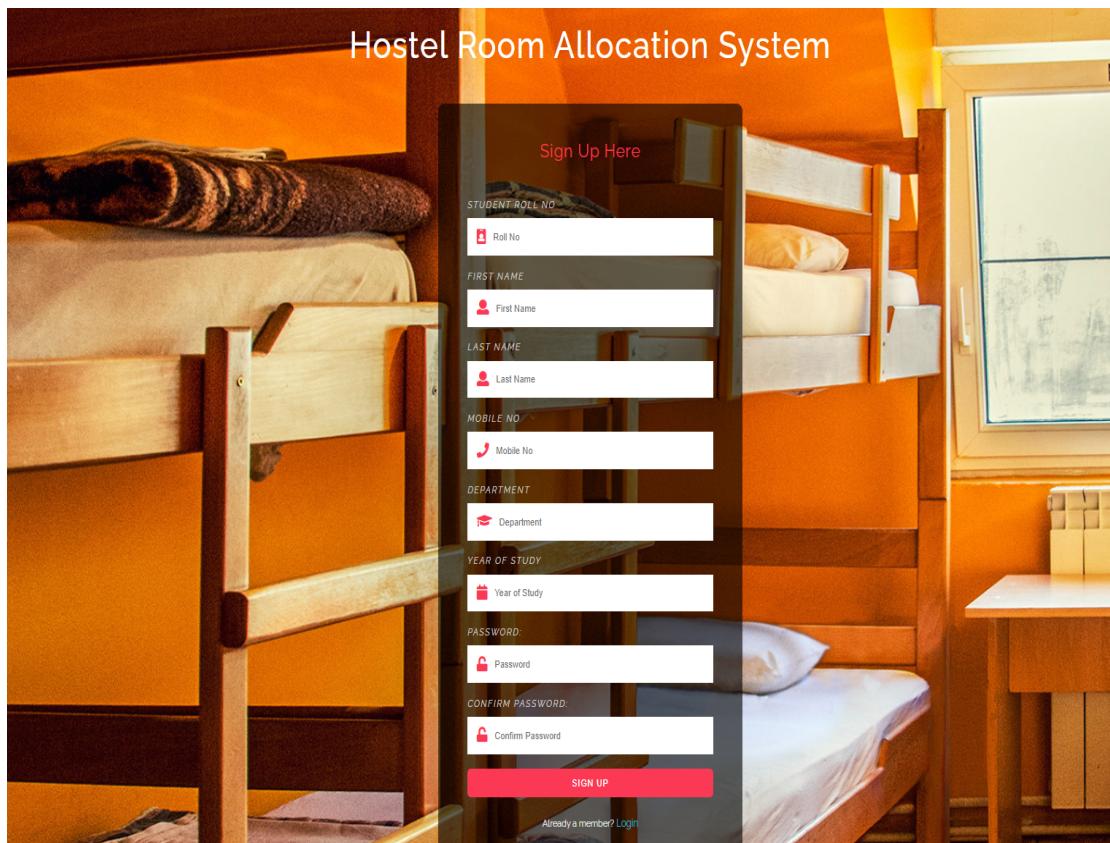
## ANNEXURE -B

## SNAPSHOT

## STUDENT SIGN-IN



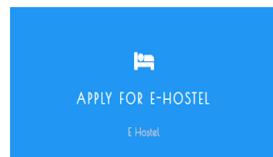
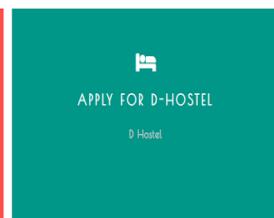
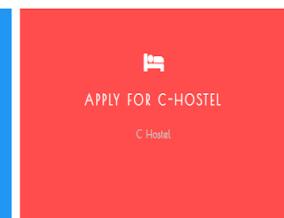
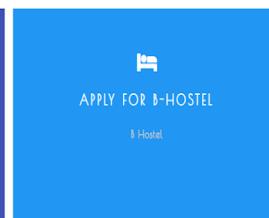
## STUDENT SIGN-UP



## STUDENT HOMEPAGE



## STUDENT HOSTEL

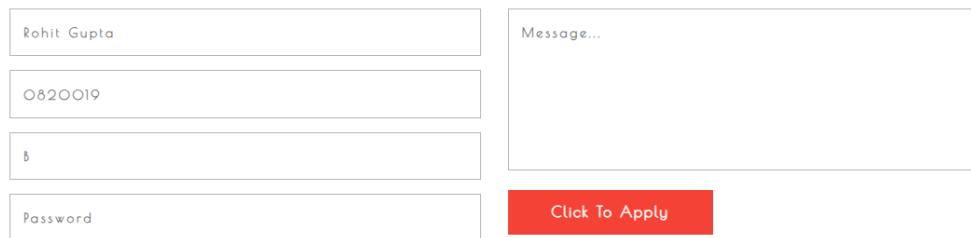
A HOSTELA Hostel  
1 yrB HOSTELB Hostel  
3 yrC HOSTELC Hostel  
2 yrD HOSTELD Hostel  
4 yrE HOSTELE Hostel  
4 yrF HOSTELF Hostel  
4 yr

## STUDENT HOSTEL APPLICATION FORM



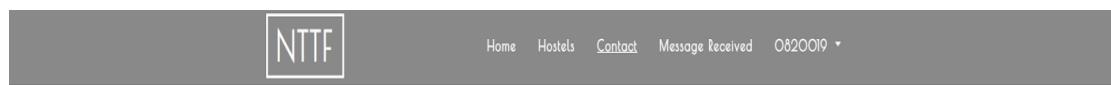
The header features the NTTF logo in a white box on the left. To the right are navigation links: Home, Hostels (underlined in red), Contact, Message Received, and a dropdown menu indicated by a downward arrow.

## Application Form



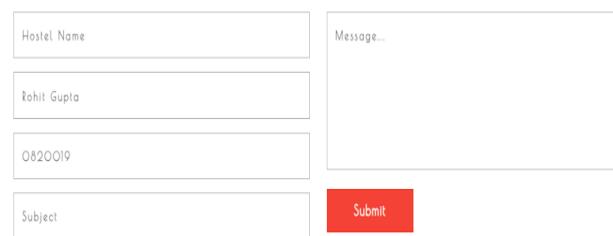
The form consists of four input fields on the left and a message area on the right. The input fields are: Name (Rohit Gupta), Hostel Number (0820019), Hostel Category (B), and Password. To the right is a large text area labeled 'Message...' with placeholder text 'Message...'. Below the message area is a red button labeled 'Click To Apply'.

## STUDENT CONTACT PAGE



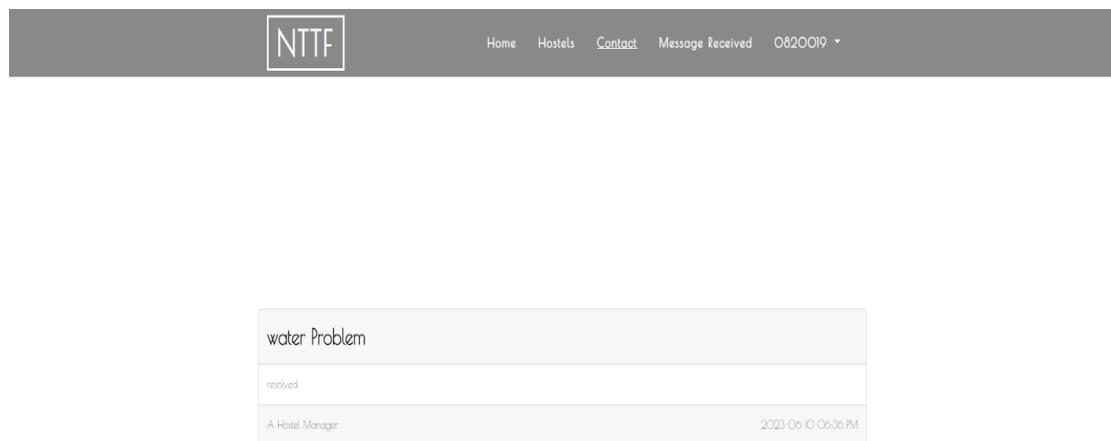
The header features the NTTF logo in a white box on the left. To the right are navigation links: Home, Hostels, Contact, Message Received, and a dropdown menu showing '0820019'.

## Contact Us



The form consists of four input fields on the left and a message area on the right. The input fields are: Hostel Name (Hostel Name), Name (Rohit Gupta), Hostel Number (0820019), and Subject. To the right is a large text area labeled 'Message...' with placeholder text 'Message...'. Below the message area is a red button labeled 'Submit'.

## STUDENT MESSAGE RECEIVED PAGE



water Problem

resolved

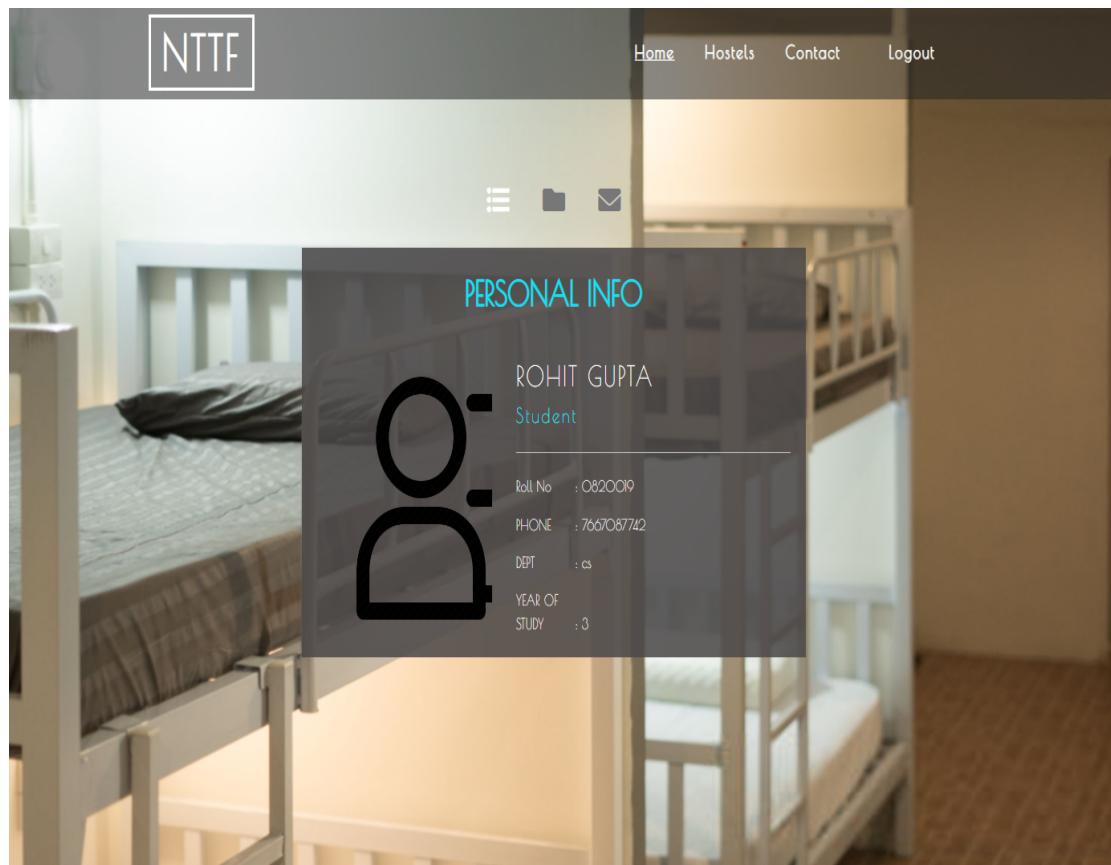
A Hostel Manager 2023-06-10 06:36 PM



NTTF  
JAMSHEDPUR

Home Hostels Contact Profile

## STUDENT PROFILE 1

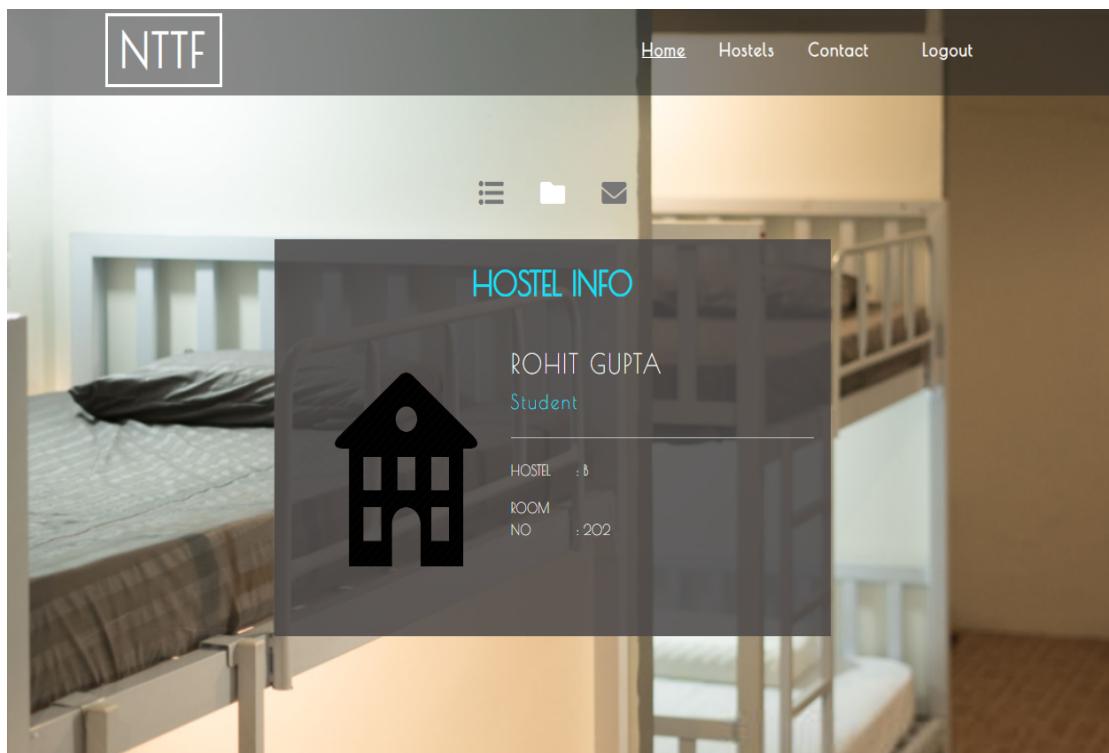


PERSONAL INFO

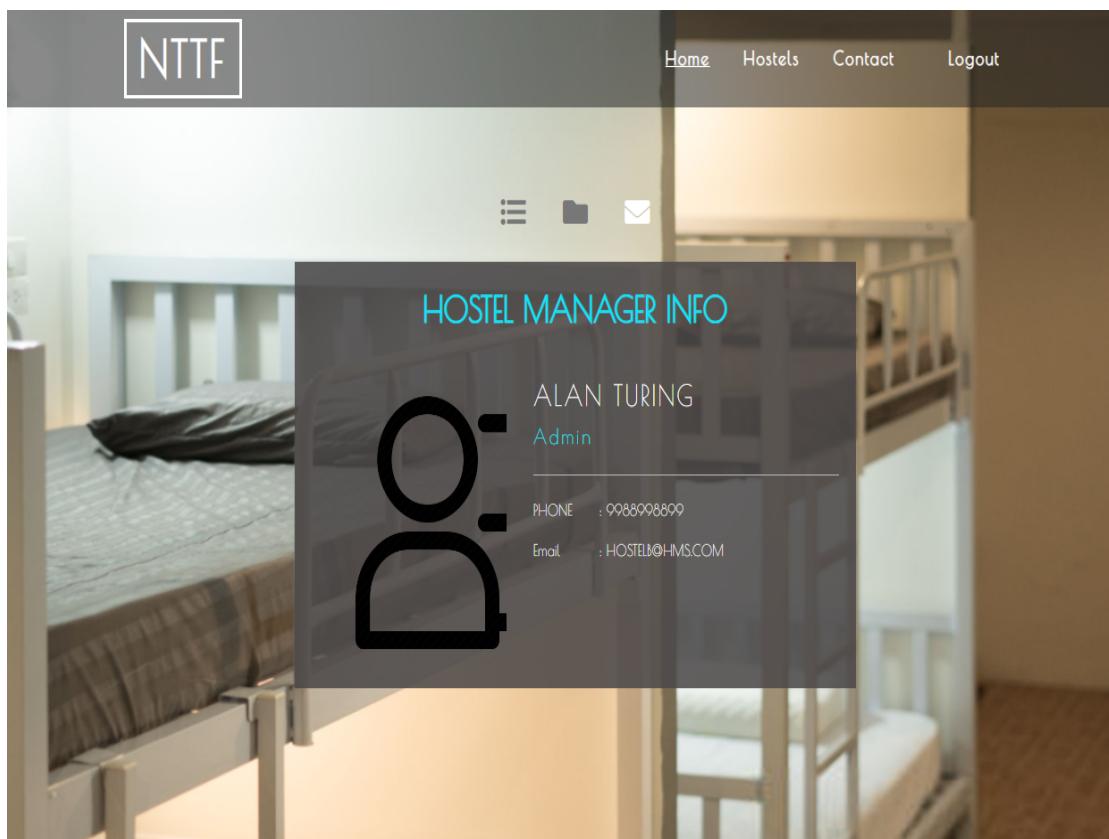
ROHIT GUPTA  
Student

Roll No : 0820019  
PHONE : 7667087742  
DEPT : cs  
YEAR OF STUDY : 3

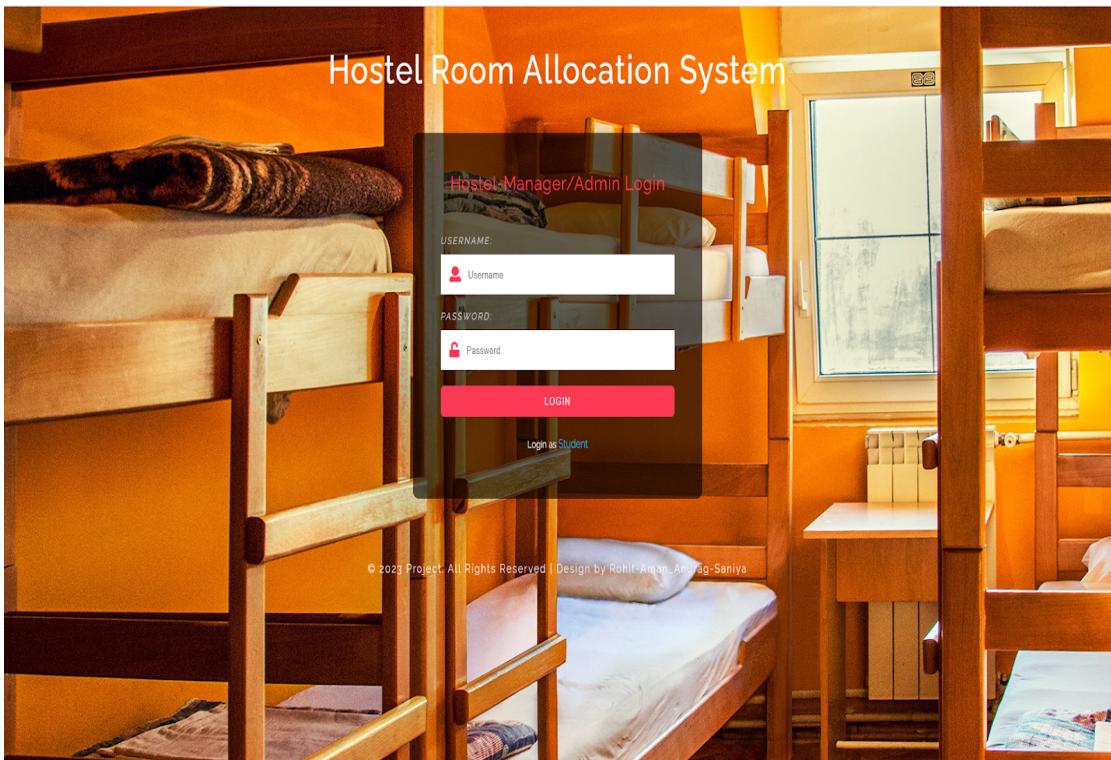
## STUDENT PROFILE 2



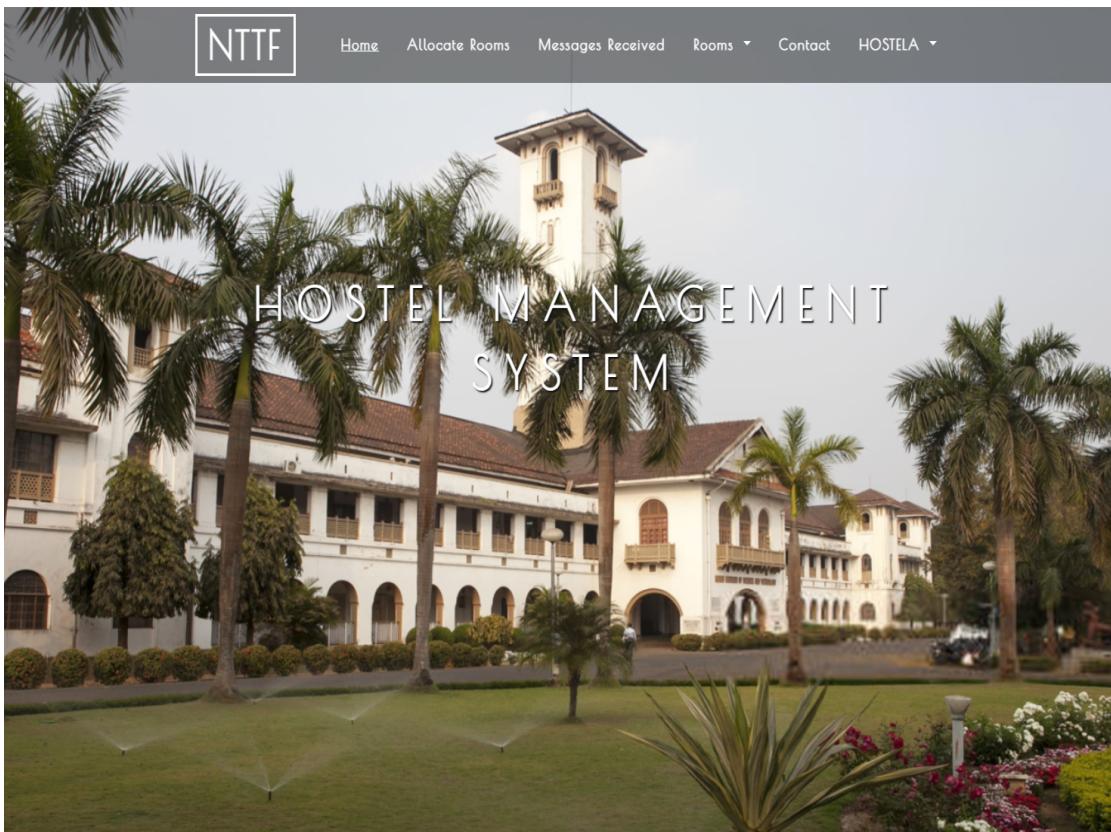
## STUDENT PROFILE 3



## HOSTEL MANAGER/ ADMIN LOGIN



## HOSTEL MANAGER HOME



## HOSTEL MANAGER ALLOCATE ROOM



NTTF

Home   Allocate Room   Messages Received   Rooms   Contact   HOSTELA

## Applications Received

Student Name	Student ID	Hostel	Message
No Rows Returned			

## HOSTEL MANAGER MESSAGE RECEIVED



NTTF

Home   Allocate Room   Messages Received   Rooms   Contact   HOSTELA

water Problem	
resolved	
12345	2023-06-10 06:36 PM

NTTF  
JAMESHEDPUR

Home   Allocate   Contact   Profile

## HOSTEL MANAGER ALLOCATED ROOMS



The header features the NTTF logo in a white box. To the right are navigation links: Home, Allocate Room, Messages Received, Rooms (with a dropdown arrow), Contact, and HOSTELB (with a dropdown arrow).

## Rooms Allotted

Student Name	Student ID	Contact Number	Hostel	Room Number
Rohit Gupta	082009	766087742	B	202
Rohit Gupta	RN1CO82009	766087742	B	201

## HOSTEL MANAGER EMPTY ROOMS



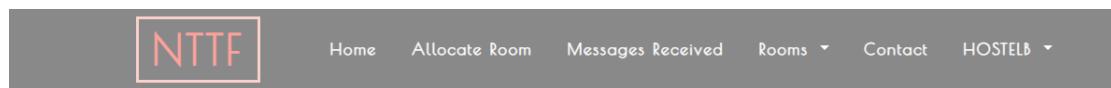
The header features the NTTF logo in a white box. To the right are navigation links: Home, Allocate Rooms, Messages Received, Rooms (with a dropdown arrow), Contact, and HOSTELB (with a dropdown arrow).

## Empty Rooms

Hostel Name	Room Number
B	203
B	204

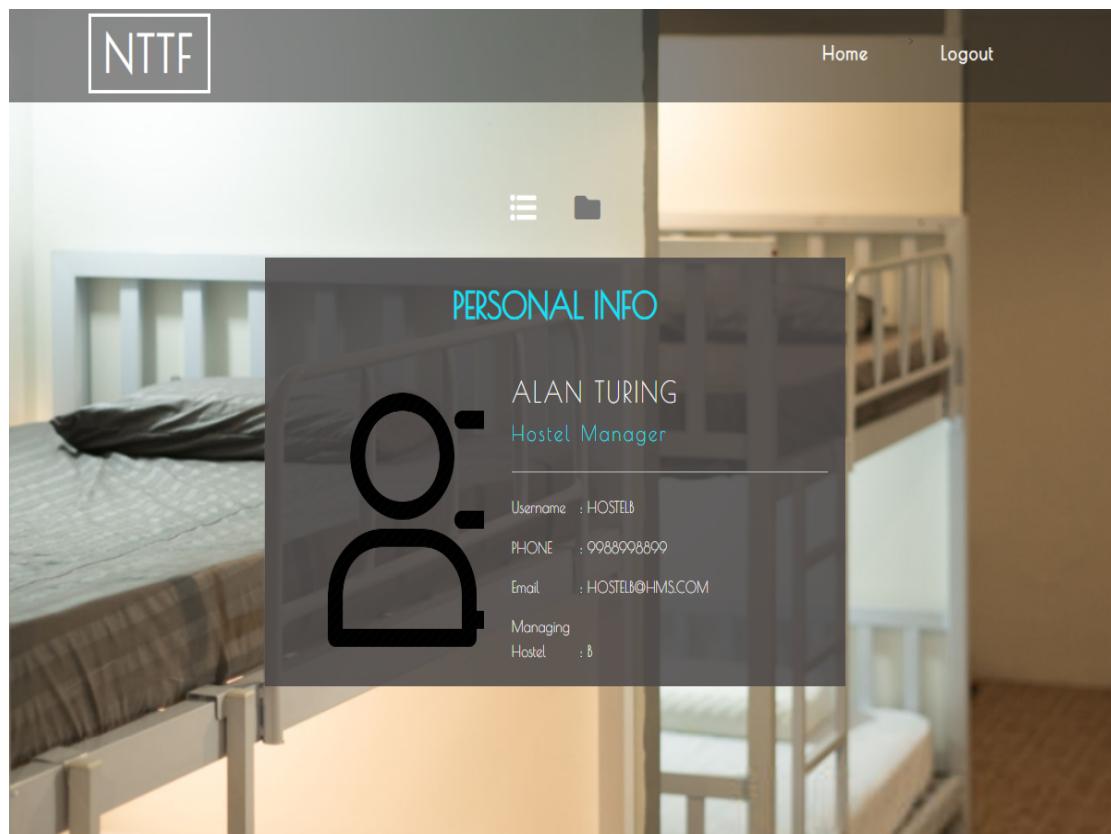
## HOSTEL MANAGER VACATE ROOM FORM



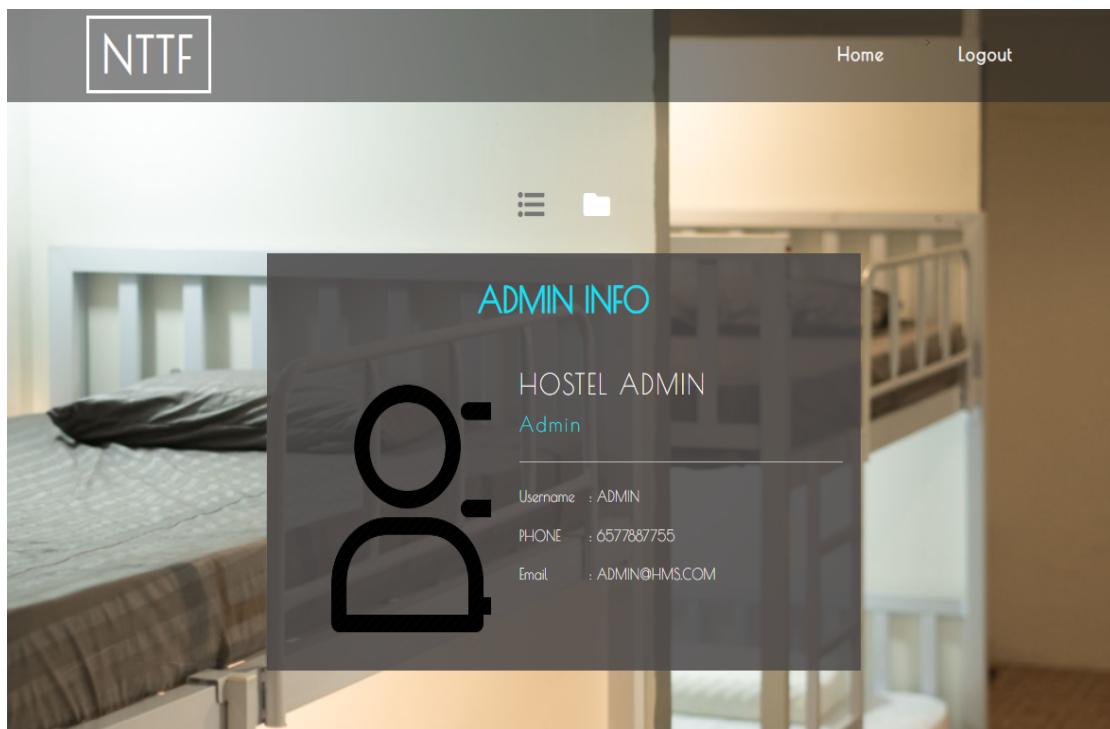
## Vacate Form

Roll Number	<input type="button" value="Click To Vacate"/>
Room Number	

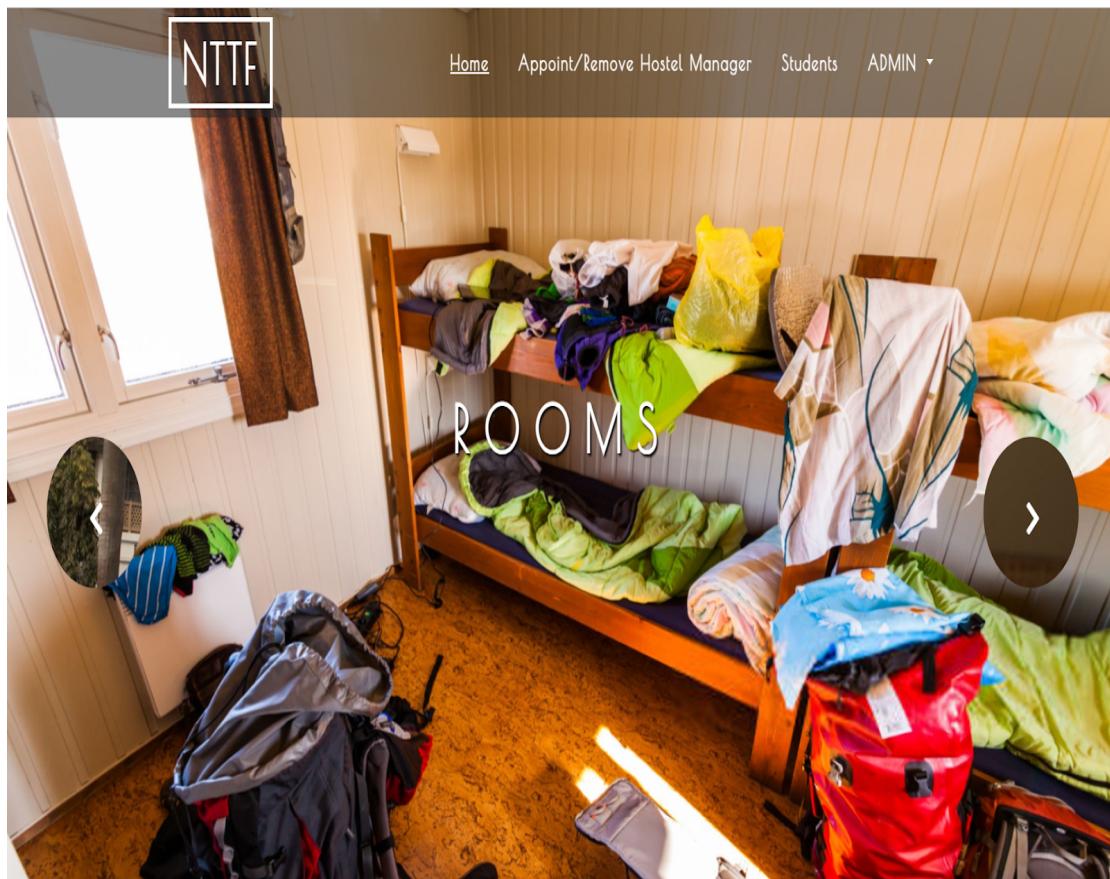
## HOSTEL MANAGER PROFILE 1



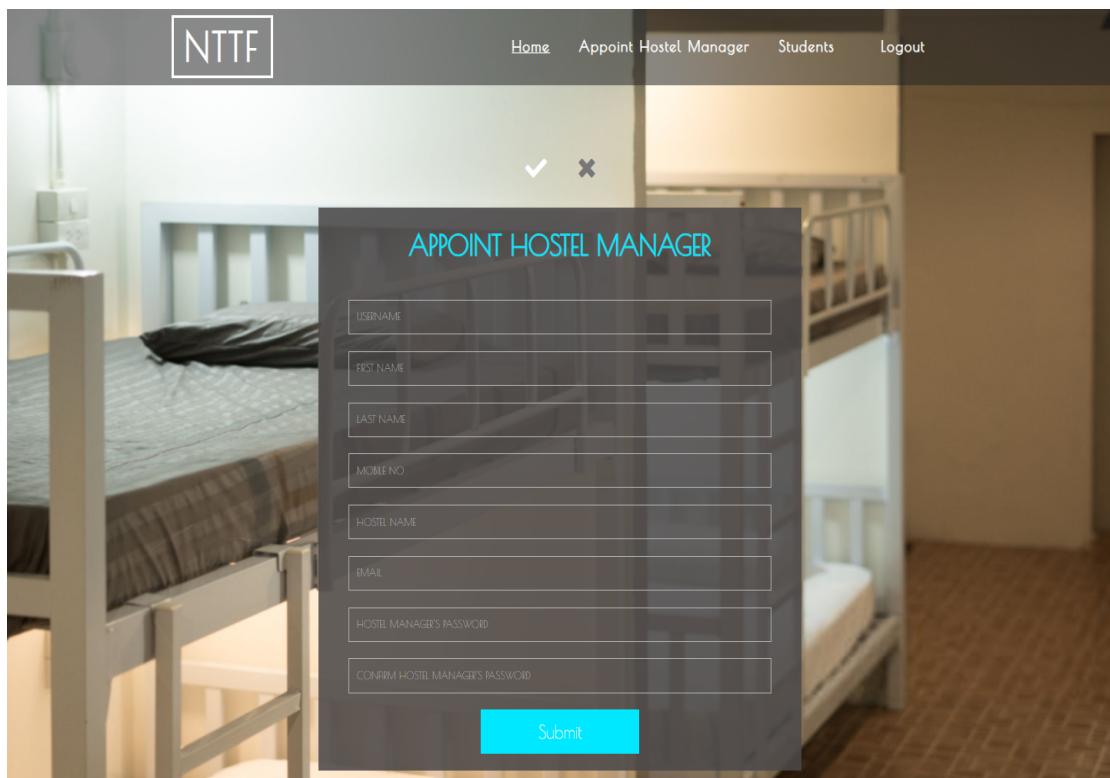
## HOSTEL MANAGER PROFILE 2



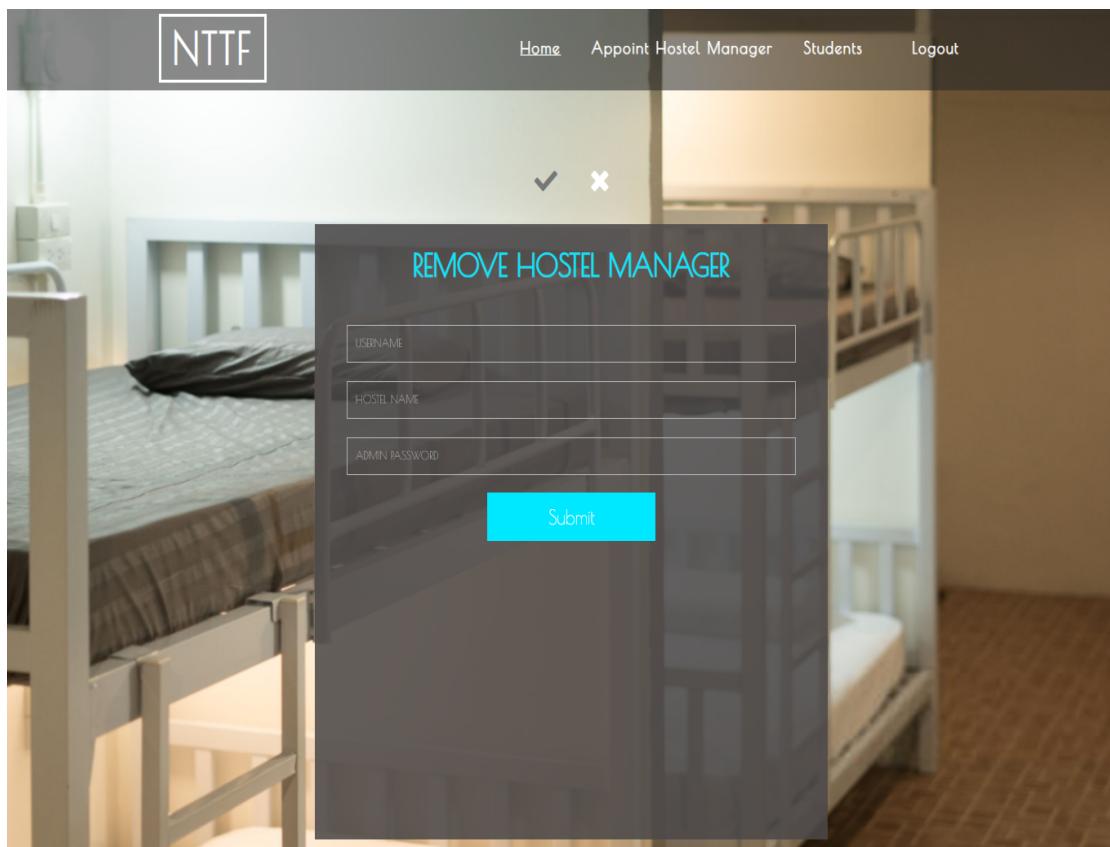
## ADMIN HOME



## ADMIN APPOINT HOSTEL MANAGER



## ADMIN REMOVE HOSTEL MANAGER



## ADMIN STUDENT DETAILS

082	<input type="button" value="Search"/>
-----	---------------------------------------

Student Name	Student ID	Contact Number	Hostel	Room Number
Ram Kumar	0820001	989797989		
a p	0820008	0000		
Rohit Gupta	0820019	7667087742	B	202
Rahul Gupta	08200192	+916203994432		

## Rooms Allotted

Student Name	Student ID	Contact Number	Hostel	Room Number
Ram Kumar	0820001	989797989	None	None
a p	0820008	0000	None	None
Rohit Gupta	0820019	7667087742	B	202
Rahul Gupta	08200192	+916203994432	None	None
anurag prasad	mtc0820008	5657754457	None	None
Rohit Gupta	RNICO0820019	7667087742	B	201

## ADMIN PROFILE

NTTF

Home Appoint/Remove Hostel Manager Students Logout

PERSONAL INFO

HOSTEL ADMIN

Admin

Username : ADMIN

PHONE : 6577887755

Email : ADMIN@HMS.COM

## BIBLIOGRAPHY

- [1] [www.w3schools.com](http://www.w3schools.com)
- [2] [in.php.net](http://in.php.net)
- [3] [en.wikipedia.org/wiki/PHP](http://en.wikipedia.org/wiki/PHP)
- [4] [www.mysql.com](http://www.mysql.com)
- [5] [www.youtube.com](http://www.youtube.com)
- [6] UK essays "Case Study on ZAS Hostel Management System"
- [7] [www.wowslder.com](http://www.wowslder.com)