BONAFIDE

This is to certify that 18CSC209J - **DATABASE MANAGEMENT SYSTEM AND CLOUD INTEGRATION SERVICES LABORATORY Mini Project report**

titled “HOSTEL MANAGEMENT SYSTEM ”is the bonafide work of

**G.JAYACHANDRAN(RA2111028010013), MEGHASYAM(RA2111028010059)**

who undertook the task of completing the project within the allotted time.

**SIGNATURE SIGNATURE**

**DR. T.Preethiya DR.Annapurani.K**

Assistant Professor Professor and Head

Department of Networking Department of Networking

and Communication and Communication

SRM Institute of Science SRM Institute of Science

and Technology and Technology

# CHAPTER 1 INTRODUCTION

## INTRODUCTION

The Hostel Management System is a database management system built using SQL developer, with the aim of simplifying the process of accessing and retrieving information related to hostel management. The system provides a user-friendly interface to hostel staff and residents, enabling them to efficiently manage and access information on room allocation, student records, maintenance, and billing. The system uses SQL queries to retrieve and manage data, ensuring data consistency and integrity. It also supports features such as online booking, payment, and messaging to enhance the user experience. The Hostel Management System provides efficient data handling and automation, making it easier for hostel staff to manage day-to-day operations and improve the overall hostel management process. With its easy-to-use interface and powerful SQL-based database management, the system can help hostels provide better services to their residents.

## PROBLEM STATEMENT

Hostel management is a complex and time-consuming process that involves several tasks such as room allocation, student records management, maintenance, and billing. The manual management of these tasks is often inefficient, leading to errors and delays in the management process.

Problems that the Hostel Management System aims to address are:-

* + 1. *Separate System for Booking and Management*

The current system for booking rooms and managing hostel operations are separate, leading to inefficiencies and difficulties in coordinating tasks.The lack of integration between booking and management systems makes it challenging to manage reservations, billing, and other tasks effectively.

* + 1. *No Separate Page for issues*

The current system lacks a portal for reporting and tracking issues such as maintenance requests or complaints.The lack of a centralized system for managing issues can result in delays in addressing issues and can lead to dissatisfaction among residents.

* + 1. *Inefficient Billing Process*

The manual billing process is often prone to errors and can result in delays in the payment process.The lack of a centralized system makes it difficult to manage and track billing information, which can lead to inaccuracies and confusion.

* + 1. *System for reallocation of rooms*

The current system lacks an efficient method for re-allocating rooms when necessary, such as when a resident leaves or when changes need to be made to room assignments.The lack of a centralized system for room re- allocation can result in delays and confusion among residents and staff.

* + 1. *Separate page for mess information*

The current system lacks a portal for managing and sharing mess-related information such as menus, schedules, and feedback.The lack of a centralized system for mess information can lead to confusion among residents and staff and can result in inefficiencies in managing the mess.

* + 1. *Separate page for laundry information*

The current system lacks a dedicated page for managing and sharing laundry-related information such as schedules, fees, and feedback.The lack of a centralized system for laundry information can lead to confusion among residents and staff and can result in inefficiencies in managing the laundry facilities.

## OBJECTIVE

The objective of the Hostel Management System project is to provide a comprehensive solution to manage various activities in the hostel. The system aims to solve the problems of separate systems for booking and management, no separate page for issues, inefficient billing process, system for reallocation of rooms, separate page for mess information, and separate page for laundry information. The system will provide online access to students and management to manage their respective tasks. The system will automate the allocation of rooms, manage payment details, and provide a separate page for mess and laundry information. The system will also provide a separate page for students to report any issues they face in the hostel. The system will be developed using Database Management System (DBMS) which will allow users to define, create, maintain and control access to the database. The system will be user-friendly and efficient, reducing the workload of hostel administrators and providing better service to the students. The system will maintain accurate records of student information, room allocation, payment details, and other related information, making the management of hostel activities more efficient and effective.

## SCOPE AND APPLICATION

The scope of the Hostel Management System project is to develop an online application that will capture all the management activities involved with the hostels. The system will manage various activities in the hostel, including room allocation, payment details, mess information, laundry information, and reallocation of rooms. The system will provide separate interfaces for students and management to manage their respective tasks. The system will be developed using Database Management System (DBMS) which will allow users to define, create, maintain and control access to the database. The application will be user-friendly and efficient, reducing the workload of hostel administrators and providing better service to the students. The system will maintain accurate records of student information, room allocation, payment details, and other related information, making the management of hostel activities more efficient and effective. The application will be designed to be scalable and adaptable to different educational institutions. The Hostel Management System project will help to ease hostel management and create a convenient and easy-to-use system for both students and hostel administrators.

## GENERAL AND UNIQUE SERVICES IN DATABASE APPLICATION

* + 1. **GENERAL SERVICES**

**Room management**: The system allows hostel administrators to manage and allocate rooms to students based on their preferences, availability, and other factors.

**Meal management**: The system enables students to book meals and manage their meal plans, while administrators can monitor and manage the meal service, such as menu planning, inventory management, and billing.

**Fee management**: The system provides a platform for students to pay their hostel fees online and allows administrators to manage fee collection, generate fee reports, and track payment histories.

**Complaint management**: The system allows students to report issues and complaints regarding hostel facilities or services, while administrators can track, manage, and resolve complaints through the system.

**Attendance management**: The system enables hostel administrators to track student attendance and manage attendance records, making it easier to monitor student activities and take necessary actions when needed.

**Visitor management**: The system enables administrators to manage visitor access to the hostel, such as issuing visitor passes, recording visitor information, and monitoring visitor activities.

**Inventory management**: The system enables administrators to manage hostel inventory, such as beddings, furniture, and other amenities, ensuring adequate supply and efficient usage of resources.

**Security management:** The system allows administrators to monitor hostel security, such as access control, CCTV surveillance, and emergency response, ensuring a safe and secure environment for students.

**Communication management:** The system provides a platform for administrators to communicate with students, such as sending notifications, updates, and reminders, and enabling students to communicate with each other and the management.

**Reporting:** The system enables administrators to generate various reports, such as occupancy reports, fee collection reports, and complaint reports, providing valuable insights into hostel operations and performance**.**

## UNIQUE SERVICES

**Customized dashboard:** The system can provide a customized dashboard for each user, such as students and administrators, with personalized information and features based on their roles and preferences.

**AI-powered chatbot:** The system can integrate an AI-powered chatbot to provide instant support to students and respond to their queries and concerns, saving time and enhancing user experience.

**Smart energy management:** The system can incorporate smart energy management features, such as automatic light and air-conditioning controls, energy consumption tracking, and optimization, reducing energy costs and promoting sustainability.

**Social media integration:** The system can integrate with social media platforms to enable students to connect and engage with each other, share updates, and participate in hostel activities and events.

**Analytics and prediction:** The system can use data analytics and prediction models to analyze student data, such as attendance, fee payment, and complaints, to identify patterns and make informed decisions.

**Mobile app:** The system can provide a mobile app for students and administrators to access the system's features and services, providing convenience and mobility.

**Integration with other systems:** The system can integrate with other campus systems, such as the student information system and finance system, to provide a seamless experience for users and enable efficient data exchange between systems.

**Feedback analysis:** The system can use natural language processing and sentiment analysis to analyze student feedback and provide insights into the quality of services and facilities provided by the hostel.

**Alumni network:** The system can provide a platform for hostel alumni to connect and engage with each other, share experiences, and provide mentorship to current students.

**Virtual tour:** The system can provide a virtual tour of the hostel facilities, enabling prospective students to get a glimpse of the hostel's amenities and features, promoting better decision-making.

**Maintenance management:** The system enables administrators to manage hostel maintenance, such as scheduling repairs, tracking maintenance requests, and ensuring timely maintenance of hostel facilities.

**Laundry management:** The system allows students to book and manage their laundry services, while administrators can monitor and manage laundry operations, such as machine usage, maintenance, and billing.

**Transportation management:** The system enables administrators to manage hostel transportation, such as scheduling bus services, managing transport routes, and tracking transport usage by students.

**Events management:** The system allows administrators to organize and manage hostel events, such as cultural programs, sports events, and other activities, ensuring smooth coordination and efficient management of resources.

**Feedback management:** The system enables students to provide feedback on hostel facilities and services, while administrators can track and analyze feedback to improve the quality of services and facilities provided by the

## SOFTWARE REQUIREMENT

* + 1. **FUNCTIONAL REQUIREMENT**

Functional requirements for a Hostel Management System:-

**User management:** The system should provide user management features, such as registration, login, and password management, for both students and administrators.

**Room management:** The system should allow administrators to manage and allocate rooms to students, such as room assignment, room transfer, and room availability management.

**Meal management:** The system should provide features to manage the meal service, such as menu planning, inventory management, meal booking, and billing.

**Fee management:** The system should provide features for managing hostel fees, such as fee collection, payment tracking, and generating fee reports.

**Complaint management:** The system should provide features to report and manage complaints, such as complaint registration, complaint tracking, and complaint resolution.

**Attendance management**: The system should provide features to manage student attendance, such as attendance tracking, attendance reports, and attendance management.

**Visitor management:** The system should provide features to manage visitor access, such as visitor registration, visitor pass issuance, and visitor monitoring.

**Inventory management:** The system should provide features to manage hostel inventory, such as asset tracking, inventory management, and inventory reports.

**Security management:** The system should provide features to manage hostel security, such as access control, CCTV surveillance, and emergency response.

**Communication management:** The system should provide features for communication between students and administrators, such as notifications, announcements, and messaging.

**Reporting:** The system should provide features for generating various reports, such as occupancy reports, fee collection reports, complaint reports, and other reports as required.

**Data management:** The system should provide features to manage and store student and hostel data securely and ensure data privacy and confidentiality.

**Integration with other systems:** The system should provide features to integrate with other campus systems, such as the student information system and finance system, to enable efficient data exchange between systems.

**Scalability and customization:** The system should be scalable to accommodate future growth and customizable to meet the specific needs of the hostel and its users.

## NON FUNCTIONAL REQUIREMENT

**Performance:** The system should have a fast response time, handle multiple requests simultaneously, and perform efficiently under heavy loads.

**Security:** The system should ensure data privacy and confidentiality and protect against unauthorized access, hacking, and other security threats.

**User interface:** The system should have a user-friendly interface, easy navigation, and clear instructions to enable easy usage by all types of users.

**Reliability:** The system should have a high degree of reliability, ensuring that it is available at all times and minimizing downtime due to maintenance or other reasons.

**Scalability:** The system should be scalable to accommodate future growth and an increasing number of users, features, and data.

**Compatibility:** The system should be compatible with various devices, browsers, and operating systems, enabling users to access it from any device or location.

**Accessibility:** The system should be accessible to all types of users, including those with disabilities, and comply with accessibility standards.

**Availability:** The system should be available 24/7, with minimal downtime due to maintenance or other reasons, to ensure uninterrupted access to hostel services.

**Documentation:** The system should have comprehensive documentation, such as user manuals, system manuals, and help files, to enable users to understand the system's features and functionalities.

**Maintenance:** The system should have easy maintenance procedures, such as backup and restore, version control, and error logging, to ensure smooth maintenance and troubleshooting.

**Integration:** The system should be easily integrable with other campus systems and third-party tools to enable efficient data exchange and maximize the system's functionality.

**Performance monitoring:** The system should have performance monitoring features, such as system logs, monitoring tools, and performance reports, to ensure optimal system performance and detect and resolve issues promptly.

**Disaster recovery:** The system should have disaster recovery features, such as backup and restore procedures, to ensure minimal data loss and system downtime in case of disasters or emergencies.

**Internationalization:** The system should support multiple languages, currencies, and time zones, to enable usage by international students and staff.

## HARDWARE REQUIREMENTS

Insufficient random access memory may affect adversely on the speed andefficiency of the entire system. The process should be powerful to handle the entireoperations. The hard disk should have sufficient capacity to store the file and application.

**Processor:** Pentium IV and above

**Processor speed:** 1.4 GHz Onwards

**System memory:** 128 MB minimum (256 MB recommended)

**Cache size:** 512 KB

**RAM:** 512 MB (Minimum)

**Network card:** Any card can provide a 100mbps speed **Network connection:** UTP or Coaxial cable connection **Printer:** Inkjet/Laser Colour printer provides at least 1000 Dpi **Hard disk:** 80 GB

**Monitor:** SVGA Colour 15”

**Mouse:** 104 keys US Key Serial, USB or PS/2

## SOFTWARE REQUIREMENTS

Selected software should be acceptable by the firm and one user as well as it should be feasible for the system.

**Technology Implemented:** Apache Server **Language Used:** PHP 5.3 or newer versions **Database:** My SQL 5.5 or newer

**User Interface:** HTML, AJAX

**Web Browser:** Mozilla, Chrome or Internet Explorer 8(or newer)

**Software:** XAMPP or WAMP Server

**Operating System:** Windows XP or higher versions.

* 1. **EXISTING SYSTEM**

# CHAPTER 2 LITERATURE SURVEY

The existing system is manual based and need lot of efforts and consume enough time. In the existing system we can apply for the hostels online but the allotment processes are done manually. It may lead to corruptions in the allocation process as well as hostel fee calculation. The existing system does not deal with mess calculation and complaintregistration.

* + 1. *DISADVANTAGE:-*
       - More human power

More strength and strain of manual labour needed

* + - * Repetition of same procedure.
      * Low security.
      * Data redundancy.
      * Difficulty to handle.
      * Difficulty to update data.
      * Record keeping is difficult.
      * Backup data can be easily generated.

## PROPOSED SYSTEM

This project is aimed at developing a system for keeping records and showing information about or in a hostel. This system will help the hostel officer to be able to manage the affairs of the hostel. This system will provide full information about a student in the hostel. It will show rooms available or not and number of people in a particular room. This will also provide information on students who have paid in full or still owe. This system will also provide a report on the summary detail regarding fees and bills students are owing. Also included is a user module for employees or the hostel officer. There will also be an administrator module which will access by the administrator and has the ability to delete, add and edit employee records. This system will be developed based on Software Development Life Cycle (SDLC) with PHP and My SQL server. PHP is good for the development and design of web based programs whiles My SQL is good for databases because of its security and its advanced features and properties.

## EXISTING SYSTEM *VS* PROPOSED SYSTEM

***(Table 2.1)***

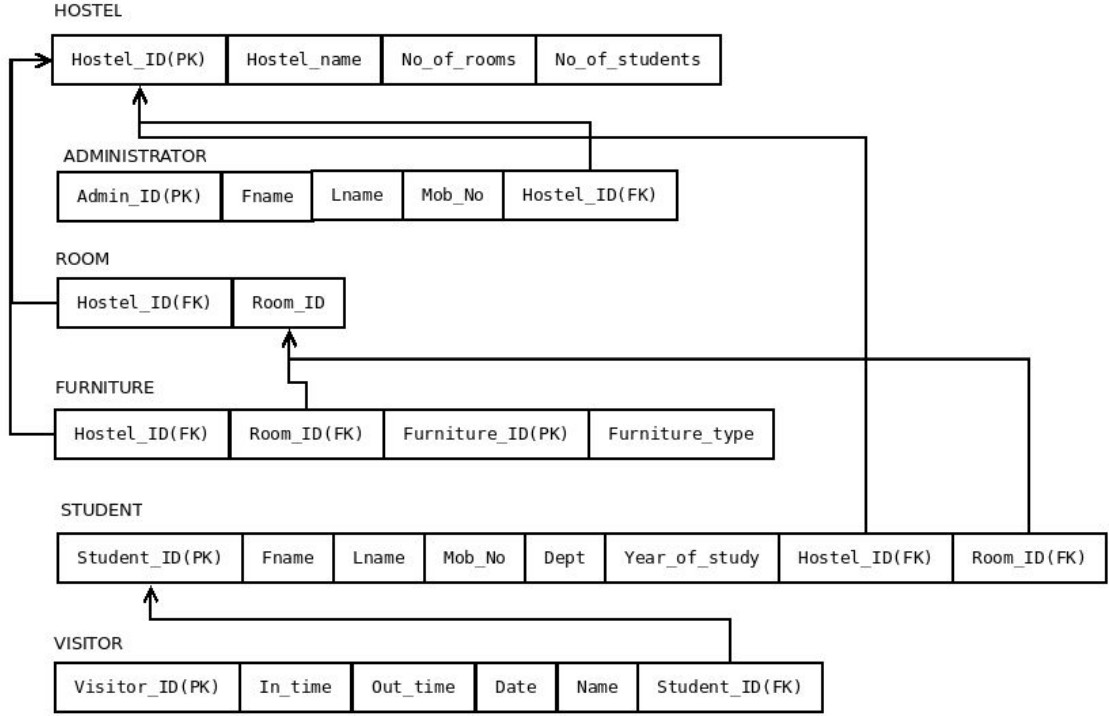
|  |  |
| --- | --- |
| **EXISTING SYSTEM** | **PROPOSED SYSTEM** |
| Uses manual Database and Records No facility for hostel room reallocation Separate System for Mess Selection  Separate System for laundry management  Separate Payment gateway site  Separate Interface for Hostel booking and management. | Uses Database Management System and SQL.  Includes separate interface for room reallocation  Incorporates mess management system with hostel management.  Incorporates laundry management with hostel management system itself.  Has payment gateway installed in the system itself  Incorporates booking system and room availability in the same system. |

In conclusion, the literature survey reveals that Hostel Management Systems are widely used and have been developed using various programming languages and databases. These systems effectively manage the day-to-day operations of hostels, including student registration, room allocation, fee collection, attendance tracking, and mess management. These systems also have features such as scalability, security, and user-friendliness. Further research can be done to improve the efficiency and effectiveness of Hostel Management Systems.

# CHAPTER 3

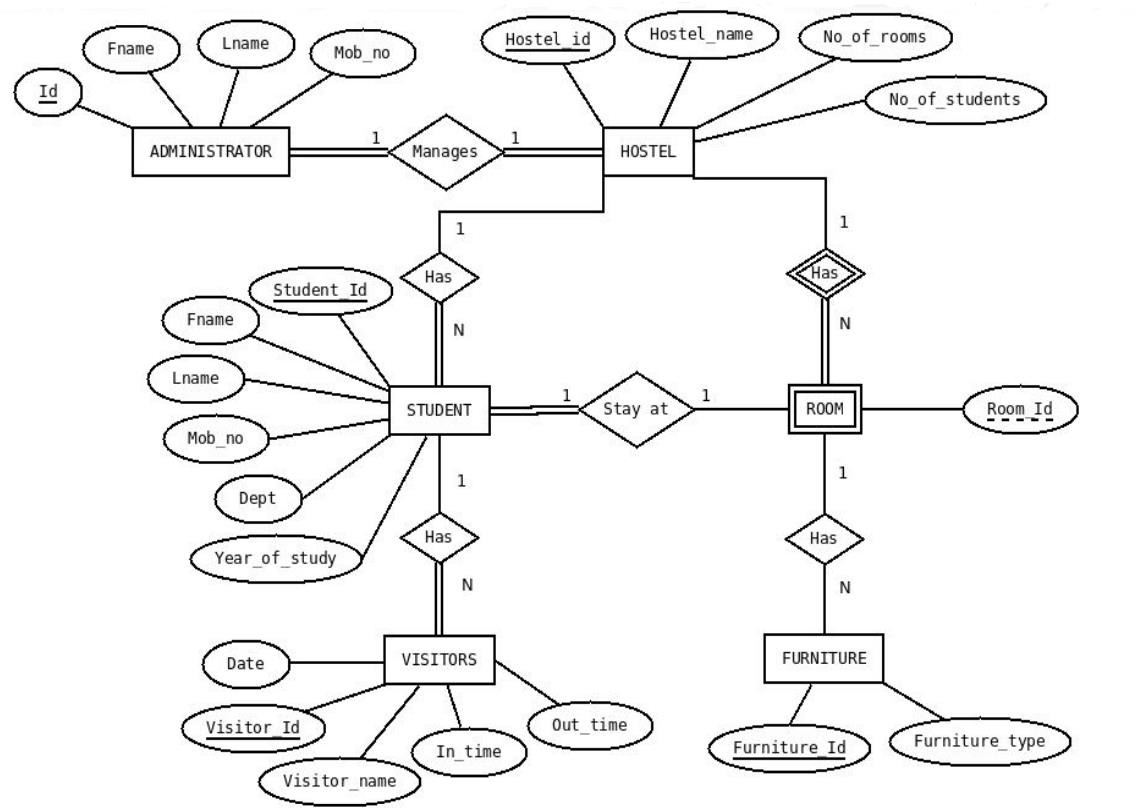
**SYSTEM ARCHITECTURE AND DESIGN**

## ARCHITECTURE DIAGRAM



***(Fig. 3.1)***

## ER DIAGRAM



***(Fig. 3.2)***

# CHAPTER 4

**MODULES AND FUNCTIONALITIES**

**Student Registration Module:** This module is used for registering new students and maintaining their personal information, including name, address, contact details, and parent/guardian information.

**Room Allocation Module:** This module is used for assigning rooms to students based on their preferences, availability, and other criteria. It also helps manage the inventory of available rooms, room types, and room tariffs.

**Attendance Tracking Module:** This module is used for tracking the attendance of students in the hostel, including attendance in mess and classes. It also generates reports on attendance trends and patterns.

**Fee Collection Module:** This module is used for collecting and managing hostel fees from students, including room rent, mess charges, and other fees. It also generates fee receipts and reports on fee collections and outstanding dues.

**Mess Management Module:** This module is used for managing mess-related activities, including meal plans, menu planning, inventory management, and kitchen management. It also generates reports on food consumption, wastage, and expenses.

**Complaint Management Module:** This module is used for receiving and addressing complaints from students regarding various issues, including maintenance, cleanliness, and security. It also generates reports on complaint trends and patterns.

**Visitor Management Module:** This module is used for managing visitors to the hostel, including recording visitor details, issuing visitor passes, and monitoring visitor activity.

**Staff Management Module:** This module is used for managing hostel staff, including their personal information, attendance, and payroll. It also generates reports on staff performance, attendance, and salary.

**Security Management Module:** This module is used for managing the security of the hostel, including monitoring access, managing security personnel, and addressing security-related issues.

**Reporting Module:** This module is used for generating reports on various aspects of hostel operations, including attendance, fee collections, room occupancy, and complaints.

**Laundry Management Module:** This module is used for managing the laundry services provided by the hostel, including tracking laundry orders, managing laundry staff, and monitoring laundry expenses. It also generates reports on laundry usage, expenses, and trends.

**Manager Appointment Module:** This module is used for managing the appointment of hostel managers and other staff, including recording their personal information, work experience, and job responsibilities. It also generates reports on staff performance, attendance, and salary.

**Room Reallocation Module:** This module is used for reallocating rooms to students based on their changing needs and preferences, and it also updates the room inventory in real-time. It helps in streamlining the room allocation process and maintaining accurate records of room occupancy.

**Room Layout Module:** This module is used for managing the layout of the rooms in the hostel, including the placement of furniture, electrical fixtures, and other amenities. It also helps in optimizing the use of available space and resources.

# CHAPTER 5 CODING AND TESTING

## PHP and SQL code PHP CODE-

php codes can be found here-

https://docs.google.com/document/d/1ki8jvNK6fVOo0ppaq6bd4yPOfAzClVMi2QxW 3YXiHRQ/edit?usp=sharing

## Database Hostel\_management\_system\_application.sql

-- MySQL dump 10.13 Distrib 5.7.23, for Linux (x86\_64)

--

-- Host: localhost Database: hostel\_management\_system

-- Server version 5.7.23-0ubuntu0.16.04.1

/\*!40101 SET

@OLD\_CHARACTER\_SET\_CLIENT=@@CHARACTER\_SET\_CLIENT \*/;

/\*!40101 SET

@OLD\_CHARACTER\_SET\_RESULTS=@@CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET

@OLD\_COLLATION\_CONNECTION=@@COLLATION\_CONNECTION \*/;

/\*!40101 SET NAMES utf8 \*/;

/\*!40103 SET @OLD\_TIME\_ZONE=@@TIME\_ZONE \*/;

/\*!40103 SET TIME\_ZONE='+00:00' \*/;

/\*!40014 SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0 \*/;

/\*!40014 SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0 \*/;

/\*!40101 SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='NO\_AUTO\_VALUE\_ON\_ZERO' \*/;

/\*!40111 SET @OLD\_SQL\_NOTES=@@SQL\_NOTES, SQL\_NOTES=0 \*/;

--

-- Table structure for table `Application`

--

DROPTABLEIFEXISTS`Application`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

/\*!40101 SET character\_set\_client = utf8 \*/; CREATETABLE `Application` (

`Application\_id`int(100) NOT NULL AUTO\_INCREMENT,

`Student\_id`varchar(255) NOT NULL,

`Hostel\_id`int(10) NOT NULL,

`Application\_status`tinyint(1) DEFAULTNULL,

`Room\_No`int(10) DEFAULTNULL,

`Message`varchar(255) DEFAULTNULL, PRIMARY KEY (`Application\_id`), KEY`Student\_id` (`Student\_id`), KEY`Hostel\_id` (`Hostel\_id`),

CONSTRAINT`Application\_ibfk\_1`FOREIGN KEY (`Student\_id`) REFERENCES`Student` (`Student\_id`),

CONSTRAINT`Application\_ibfk\_2`FOREIGN KEY (`Hostel\_id`) REFERENCES`Hostel` (`Hostel\_id`)

) ENGINE=InnoDB AUTO\_INCREMENT=2DEFAULT CHARSET=latin1;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Dumping data for table `Application`

--

LOCK TABLES `Application` WRITE;

/\*!40000 ALTER TABLE `Application` DISABLE KEYS \*/;

INSERT INTO`Application`VALUES (1,'B160497CS',1,1,NULL,'I am a handicapped, so I would like to have a room at ground floor ');

/\*!40000 ALTER TABLE `Application` ENABLE KEYS \*/; UNLOCK TABLES;

/\*!40103 SET TIME\_ZONE=@OLD\_TIME\_ZONE \*/;

/\*!40101 SET SQL\_MODE=@OLD\_SQL\_MODE \*/;

/\*!40014 SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS \*/;

/\*!40014 SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS \*/;

/\*!40101 SET CHARACTER\_SET\_CLIENT=@OLD\_CHARACTER\_SET\_CLIENT

\*/;

/\*!40101 SET

CHARACTER\_SET\_RESULTS=@OLD\_CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET

COLLATION\_CONNECTION=@OLD\_COLLATION\_CONNECTION \*/;

/\*!40111 SET SQL\_NOTES=@OLD\_SQL\_NOTES \*/;

-- Dump completed on 2018-10-15 14:14:13

Hostel\_management\_system\_hostel\_manager.sql

-- MySQL dump 10.13 Distrib 5.7.23, for Linux (x86\_64)

--

-- Host: localhost Database: hostel\_management\_system

-- Server version 5.7.23-0ubuntu0.16.04.1

/\*!40101 SET

@OLD\_CHARACTER\_SET\_CLIENT=@@CHARACTER\_SET\_CLIENT \*/;

/\*!40101 SET

@OLD\_CHARACTER\_SET\_RESULTS=@@CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET

@OLD\_COLLATION\_CONNECTION=@@COLLATION\_CONNECTION \*/;

/\*!40101 SET NAMES utf8 \*/;

/\*!40103 SET @OLD\_TIME\_ZONE=@@TIME\_ZONE \*/;

/\*!40103 SET TIME\_ZONE='+00:00' \*/;

/\*!40014 SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0 \*/;

/\*!40014 SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0 \*/;

/\*!40101 SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='NO\_AUTO\_VALUE\_ON\_ZERO' \*/;

/\*!40111 SET @OLD\_SQL\_NOTES=@@SQL\_NOTES, SQL\_NOTES=0 \*/;

--

-- Table structure for table `Hostel\_Manager`

--

DROPTABLEIFEXISTS`Hostel\_Manager`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

/\*!40101 SET character\_set\_client = utf8 \*/; CREATETABLE `Hostel\_Manager` (

`Hostel\_man\_id`int(10) NOT NULL AUTO\_INCREMENT,

`Username`varchar(255) NOT NULL,

`Fname`varchar(255) NOT NULL,

`Lname`varchar(255) NOT NULL,

`Mob\_no`varchar(255) NOT NULL,

`Hostel\_id`int(10) NOT NULL,

`Pwd` LONGTEXT NOT NULL,

`Isadmin`tinyint(1) DEFAULT'0', PRIMARY KEY (`Hostel\_man\_id`), UNIQUE (`Username`), KEY`Hostel\_id` (`Hostel\_id`),

CONSTRAINT`Hostel\_Manager\_ibfk\_1`FOREIGN KEY (`Hostel\_id`) REFERENCES`Hostel` (`Hostel\_id`)

) ENGINE=InnoDBDEFAULT CHARSET=latin1;

## Testing:

**SYSTEM TESTING**

As the part of system testing we execute 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 are unit testing, integration testing and user

acceptance testing.

## UNIT TESTING

The software units in the system is are modules and routines that are assembled and

integrated to perform a specific function. As a part of unit testing we executed the program

for individual modules independently. This enables, to detect errors in coding and logic that

are contained within each of the three module. This testing includes entering data that is

filling forms and ascertaining if the value matches to the type and entered into the database.

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, sub

functions 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 tested modules and build a program structure. All the modules are combined and tested

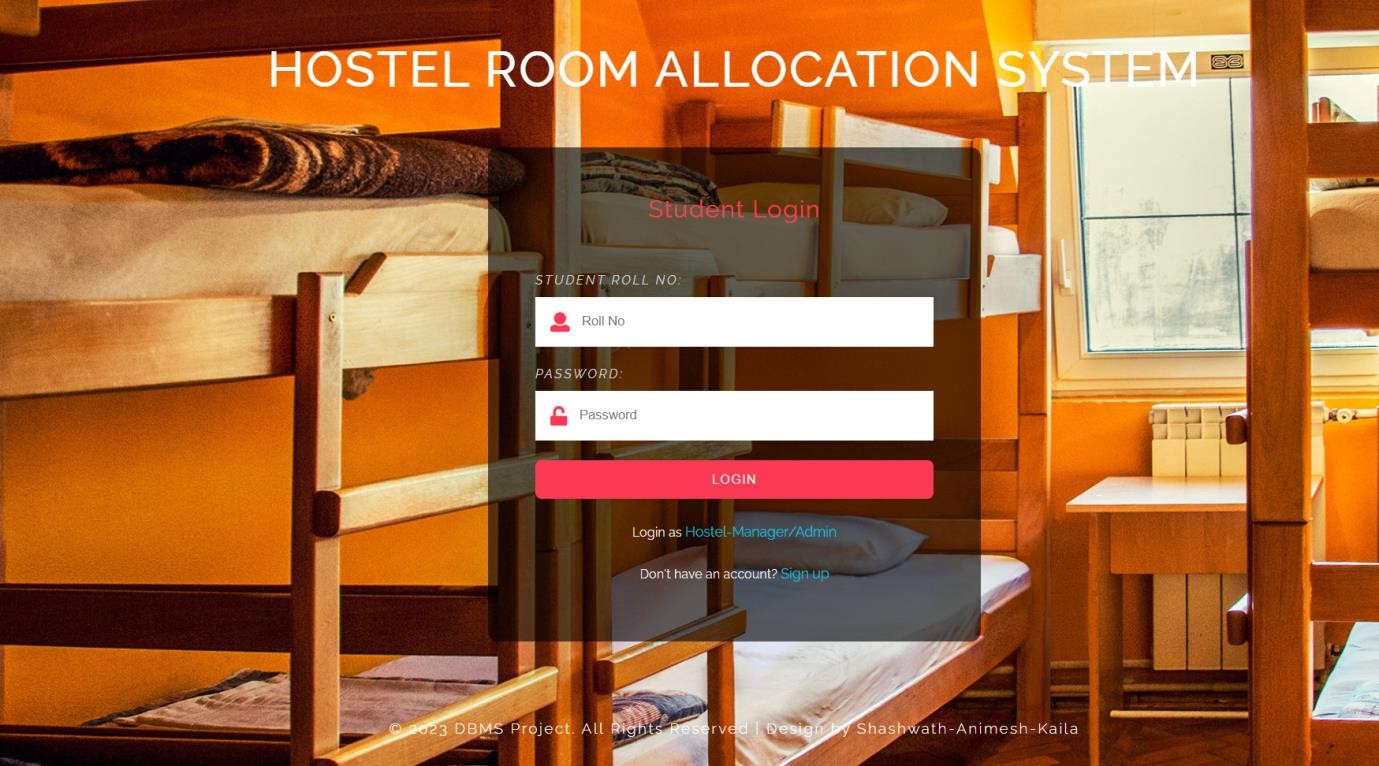
as a whole. Here the admin module, employee module and student module options are integrated and tested.

This testing provides the assurance that the application is well integrated functional unit with

smooth transition of data.

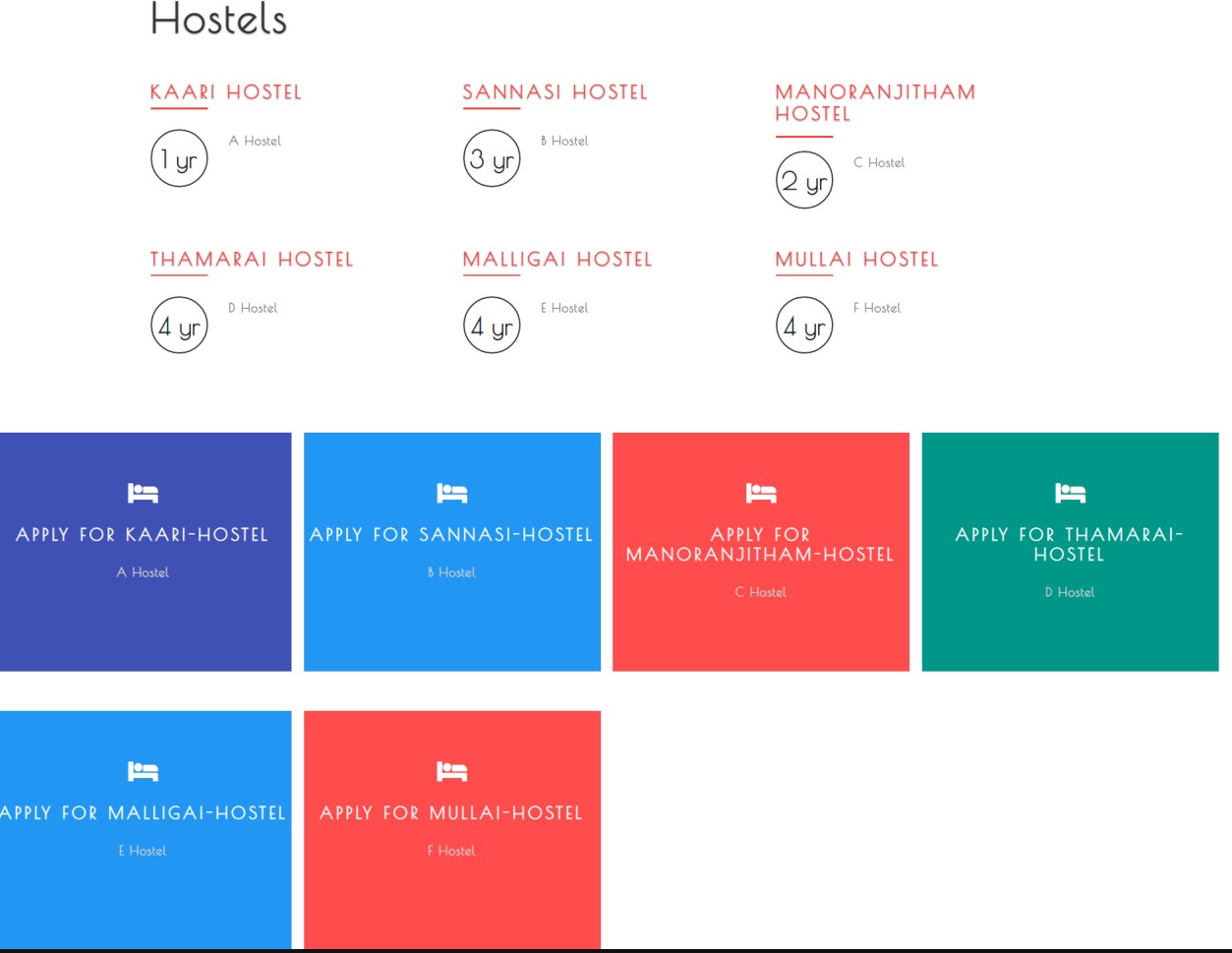
## OUPUTS

**STUDENT SIGN IN**- **An interface for signing in to the interface**



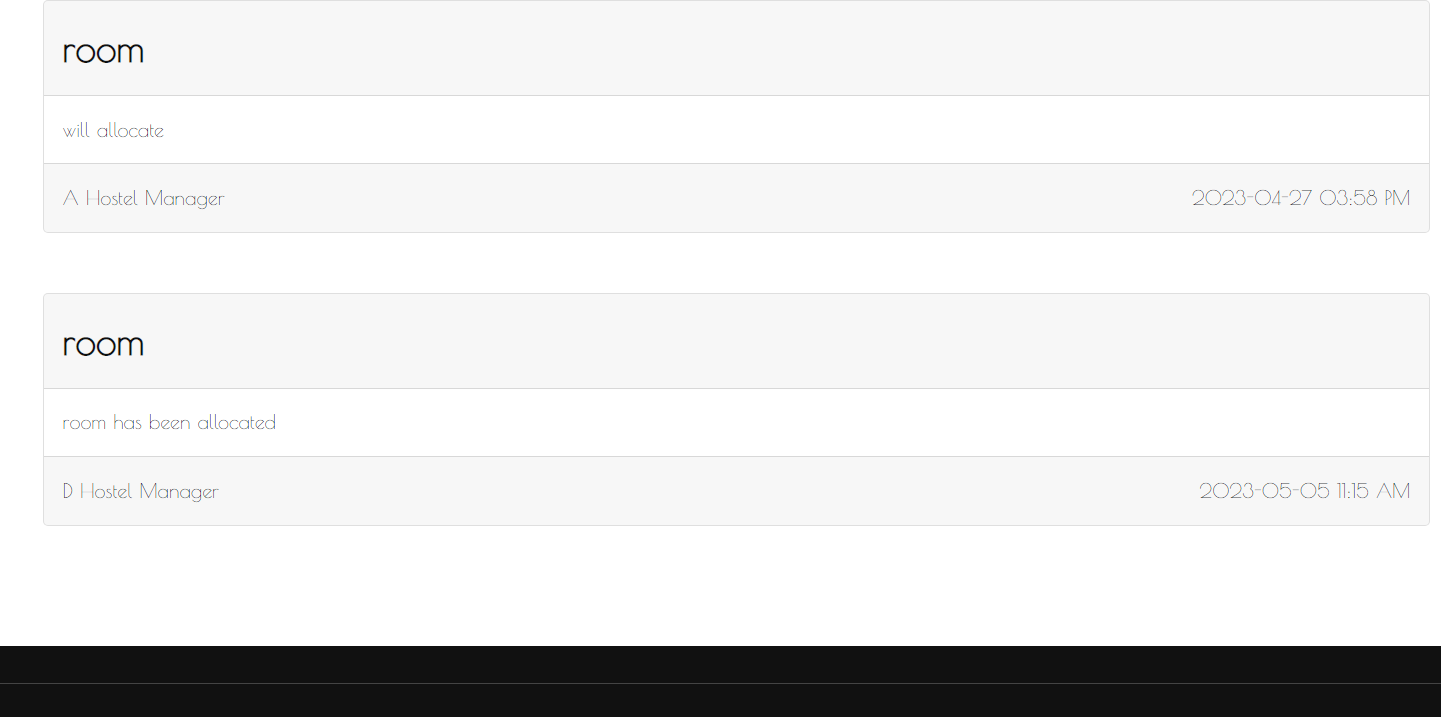
**STUDENT HOME PAGE**- **Home page displaying the different services**

**HOSTELS**- **Various hostels that we can choose from**

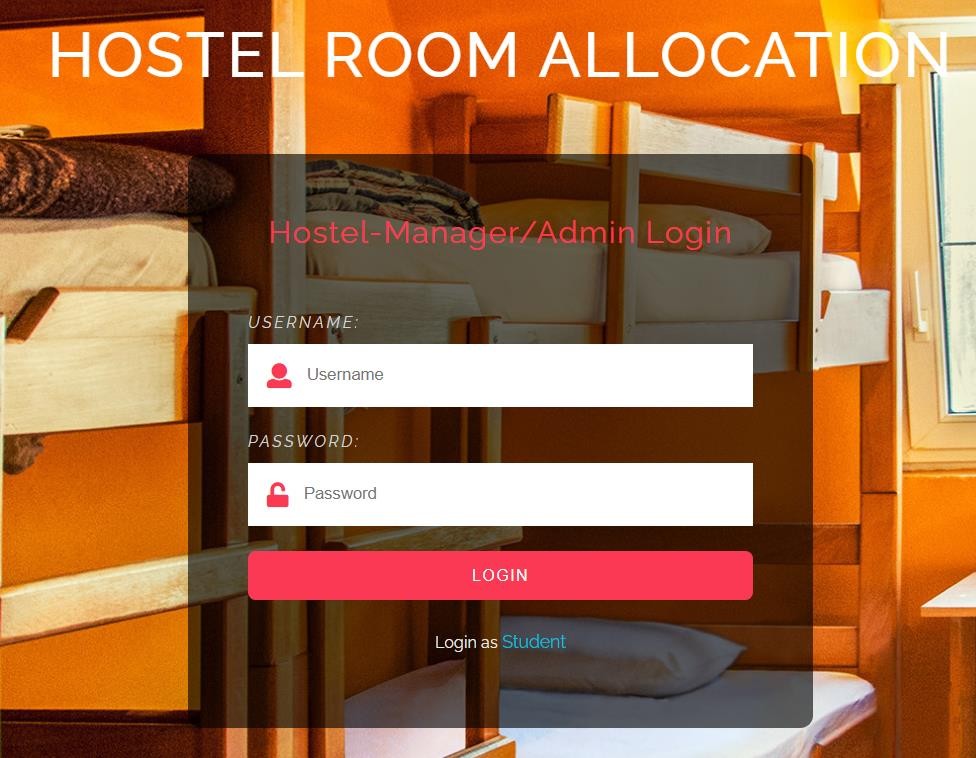


**STUDENT PESONAL INFO**- **Profile describing the entire details of the student**

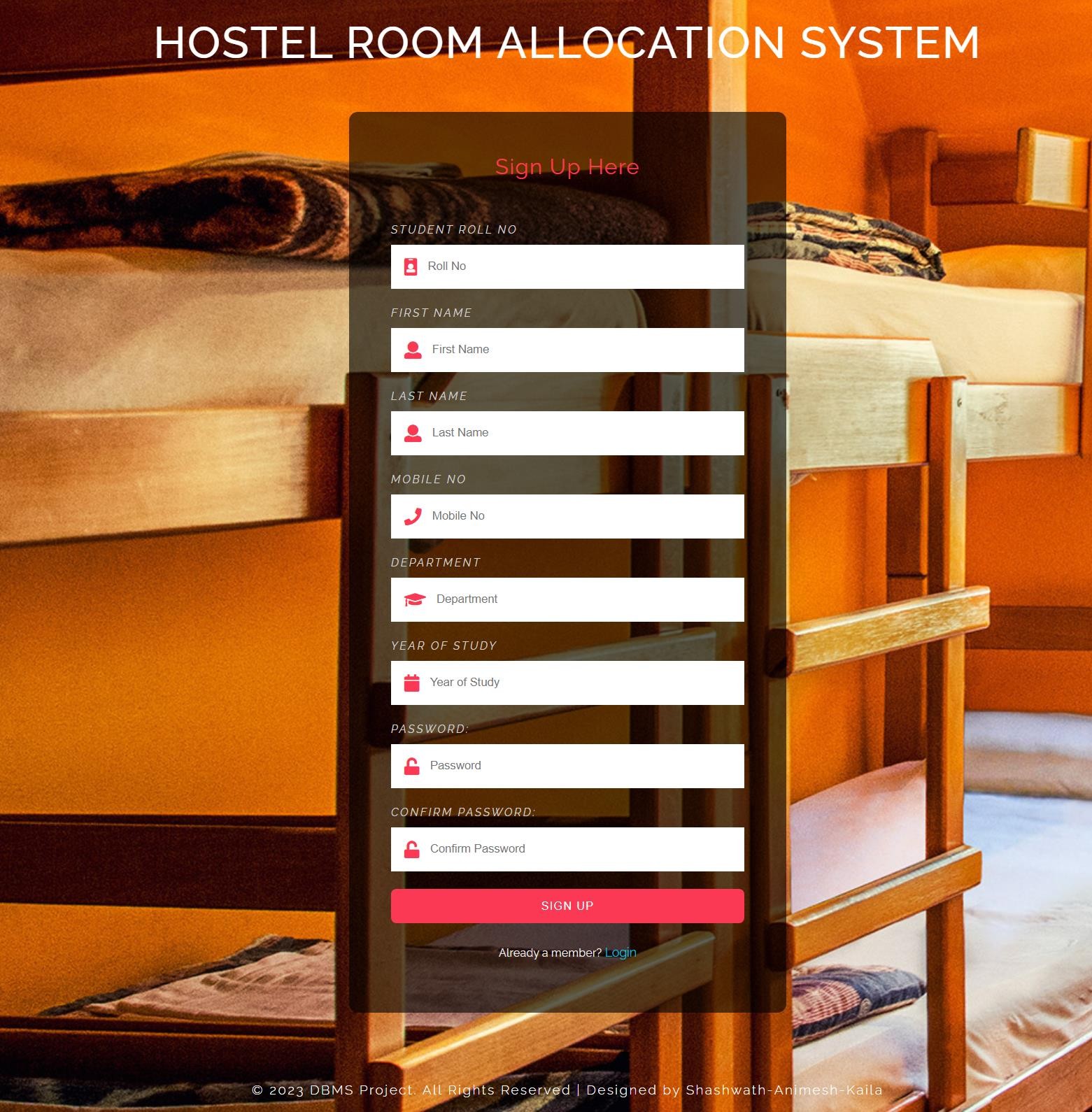
**MESSAGE RECEIVED- message received from the manager or message interface**

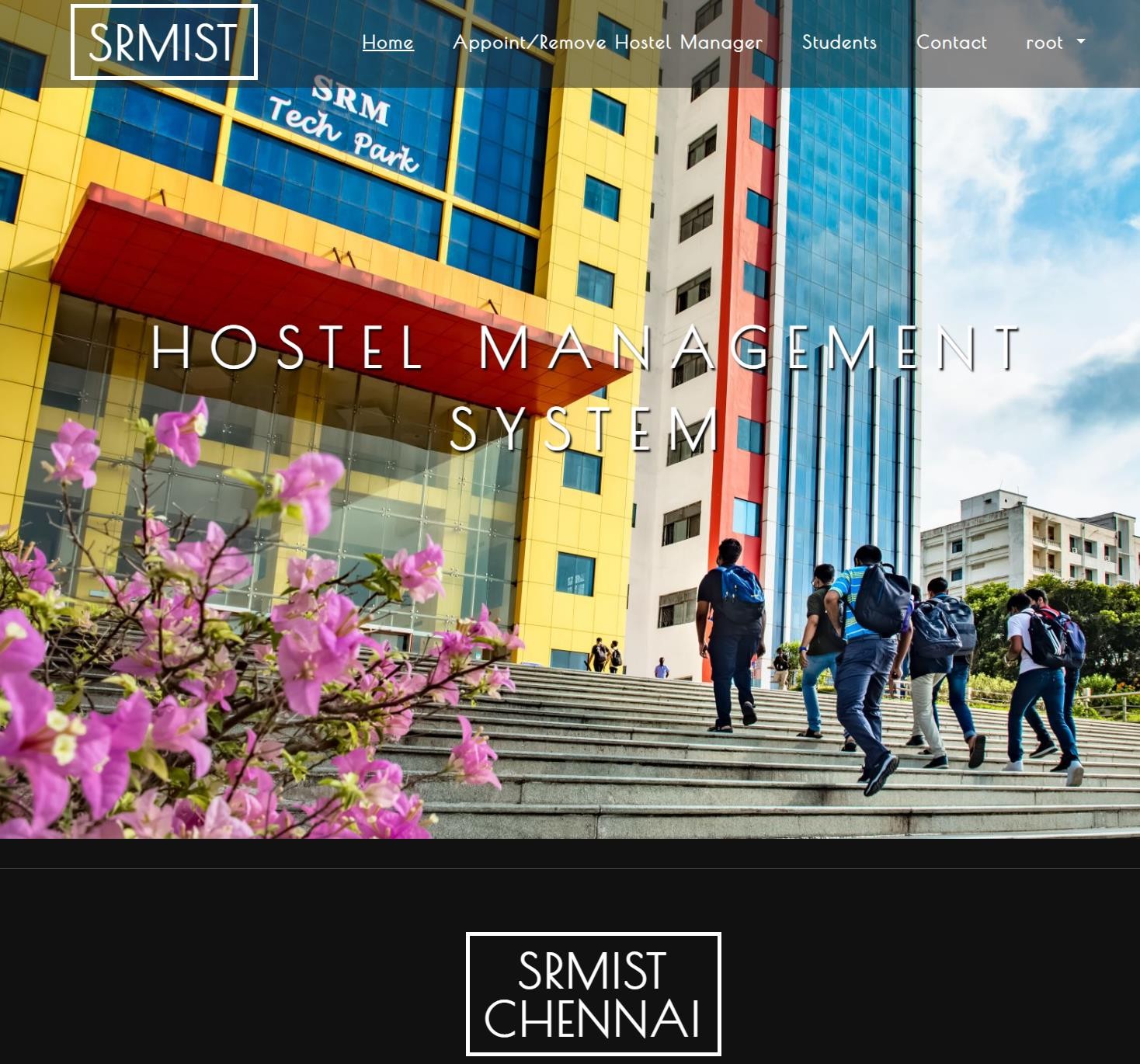


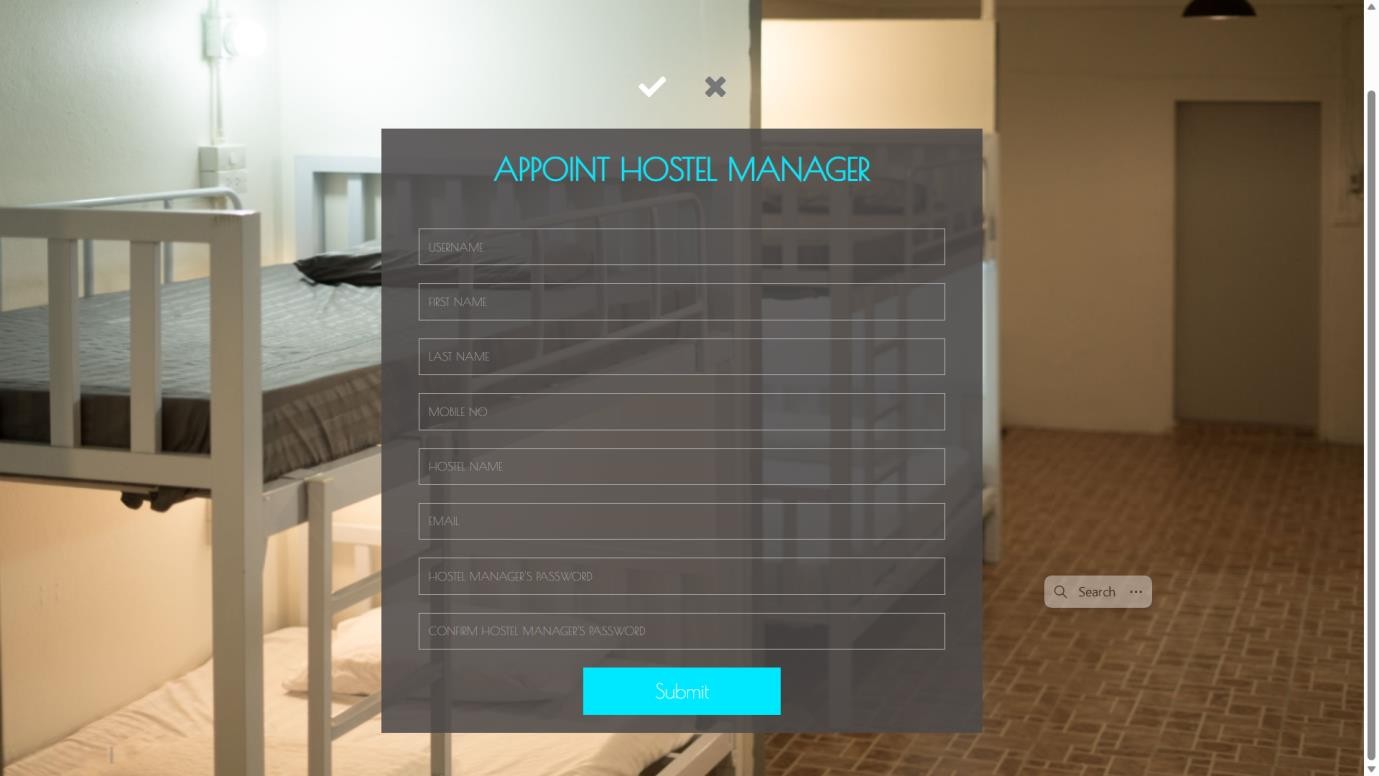
**HOSTEL-MANAGER/ADMIN SIGN IN-Sign in interface for hostel manager/admin**



**STUDENT SIGN UP- sign up page for the student**

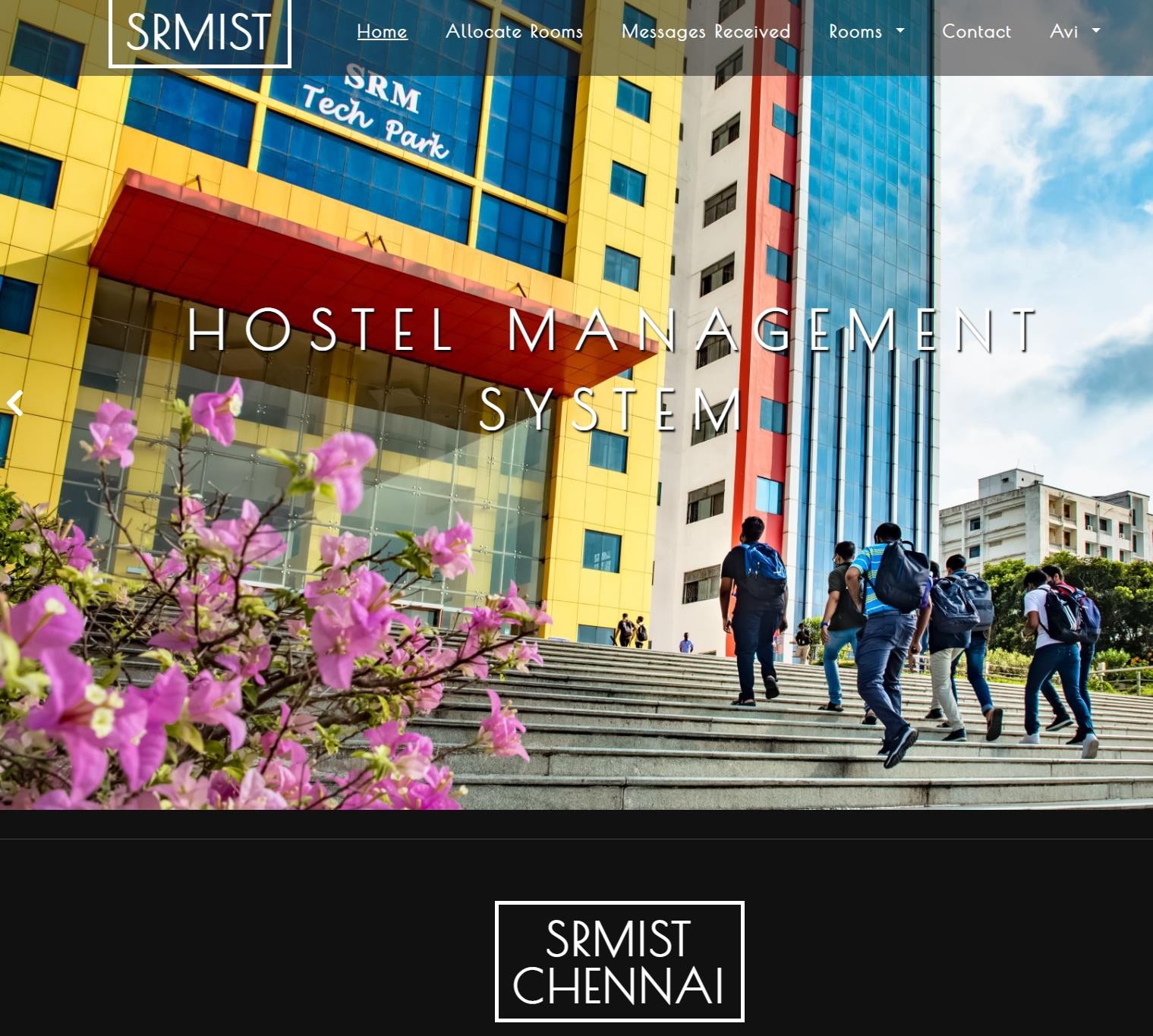


**ADMIN HOME PAGE- Admin services and home page**

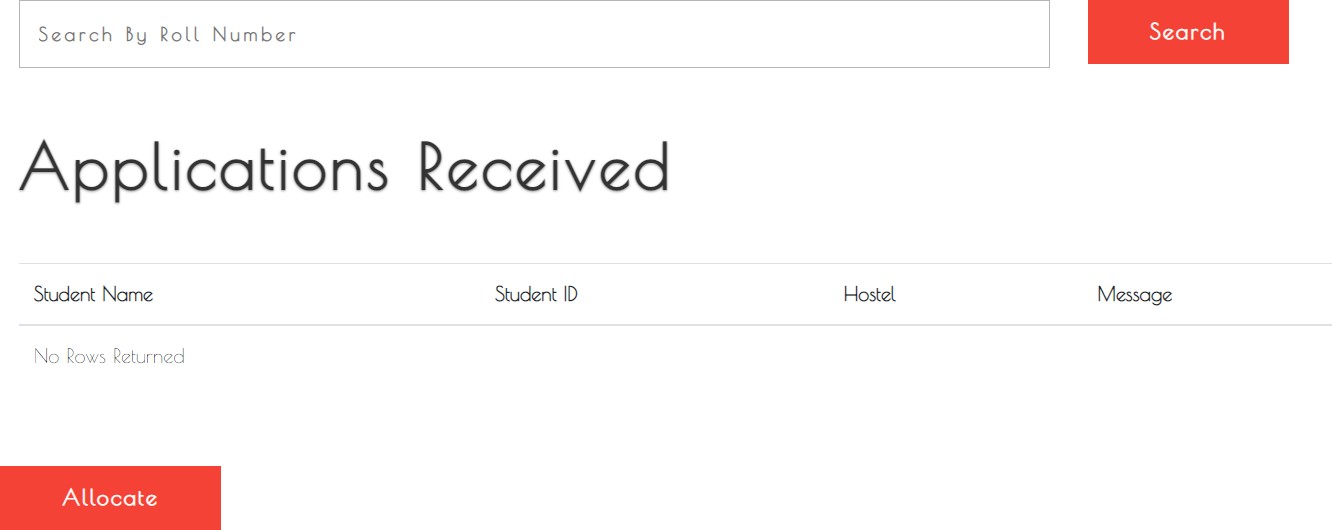
**APPOINT HOSTEL MANAGER and REMOVE MANAGER-Appoint and remove manager**

**ROOT PERSONAL INFO**

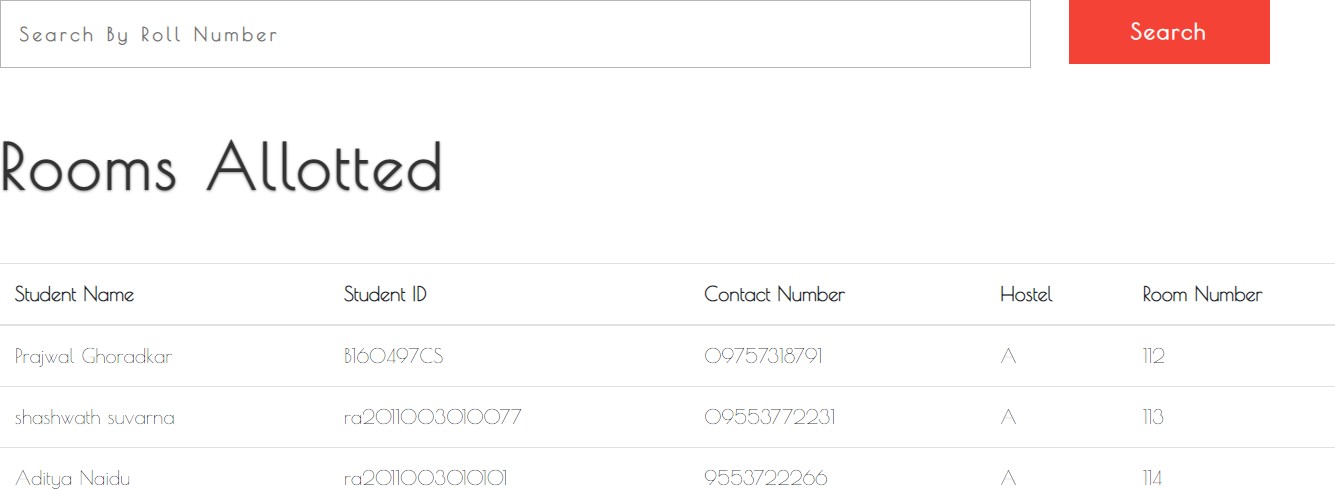


**HOSTEL MANAGER HOME PAGE- Manager home page**

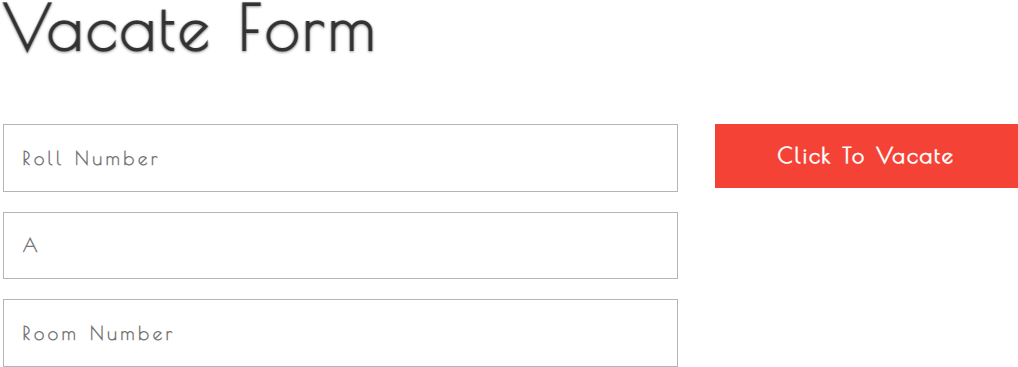
**APPLICATIONS RECEIVED-Applications received for room allotment**



**ROOMS ALLOTED-rooms already alloted**

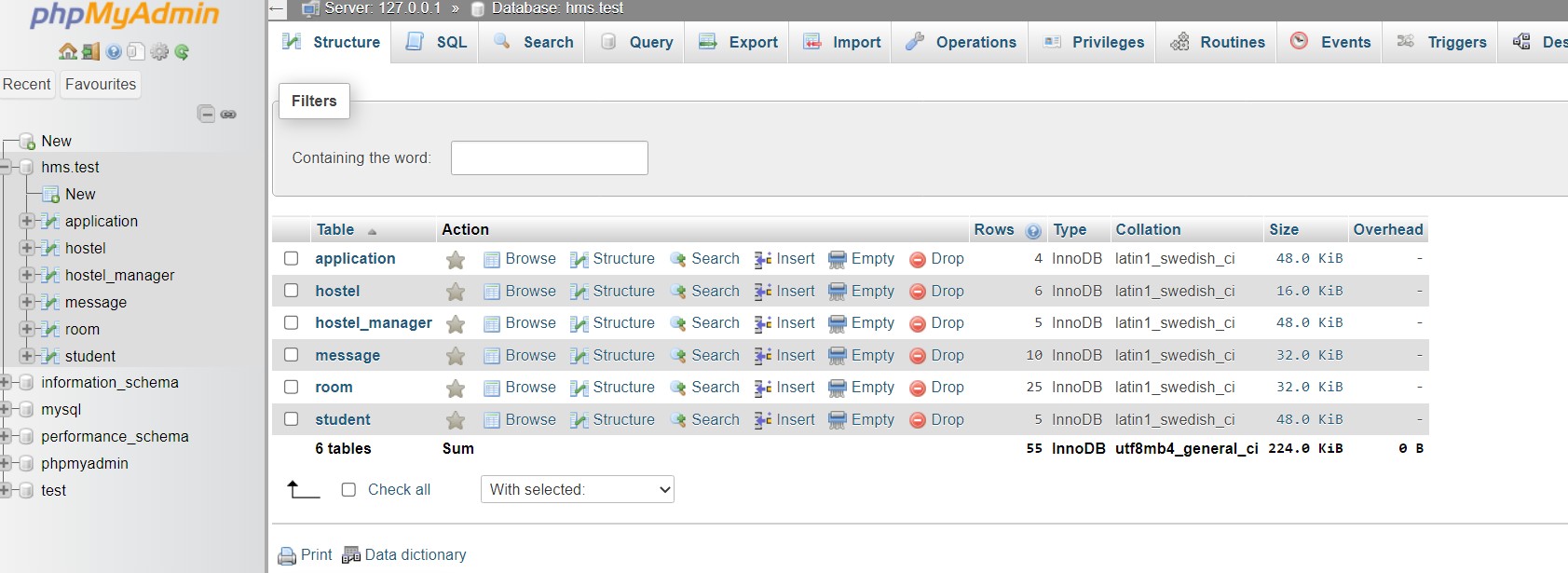


**ROOMS VACATED-rooms vacated**

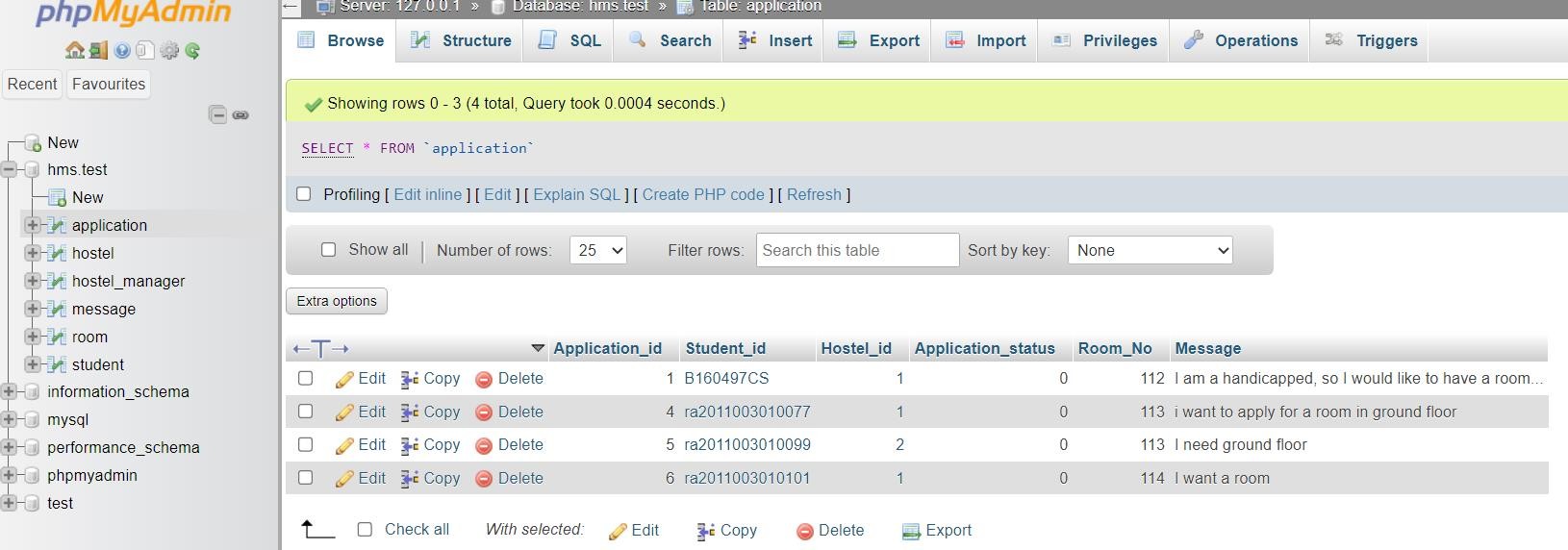


**HOSTEL MANAGER PERSONAL INFO- Personal details of the hostel manager**

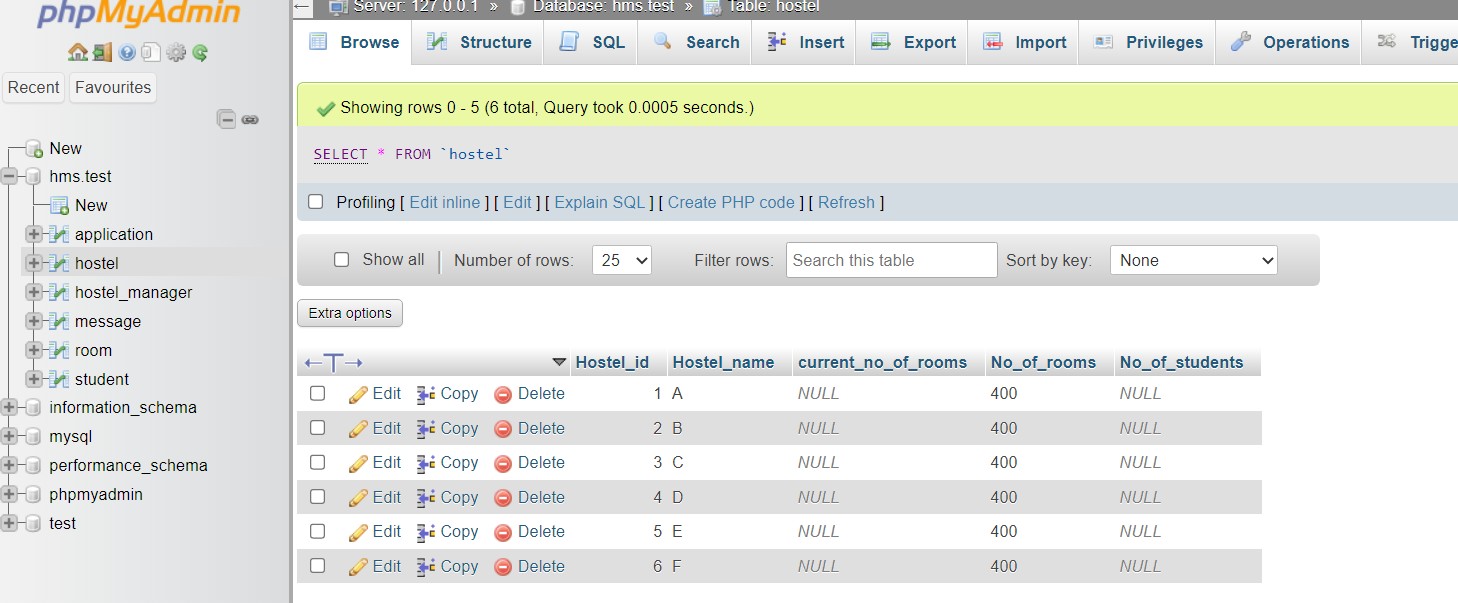
**DATABASE**



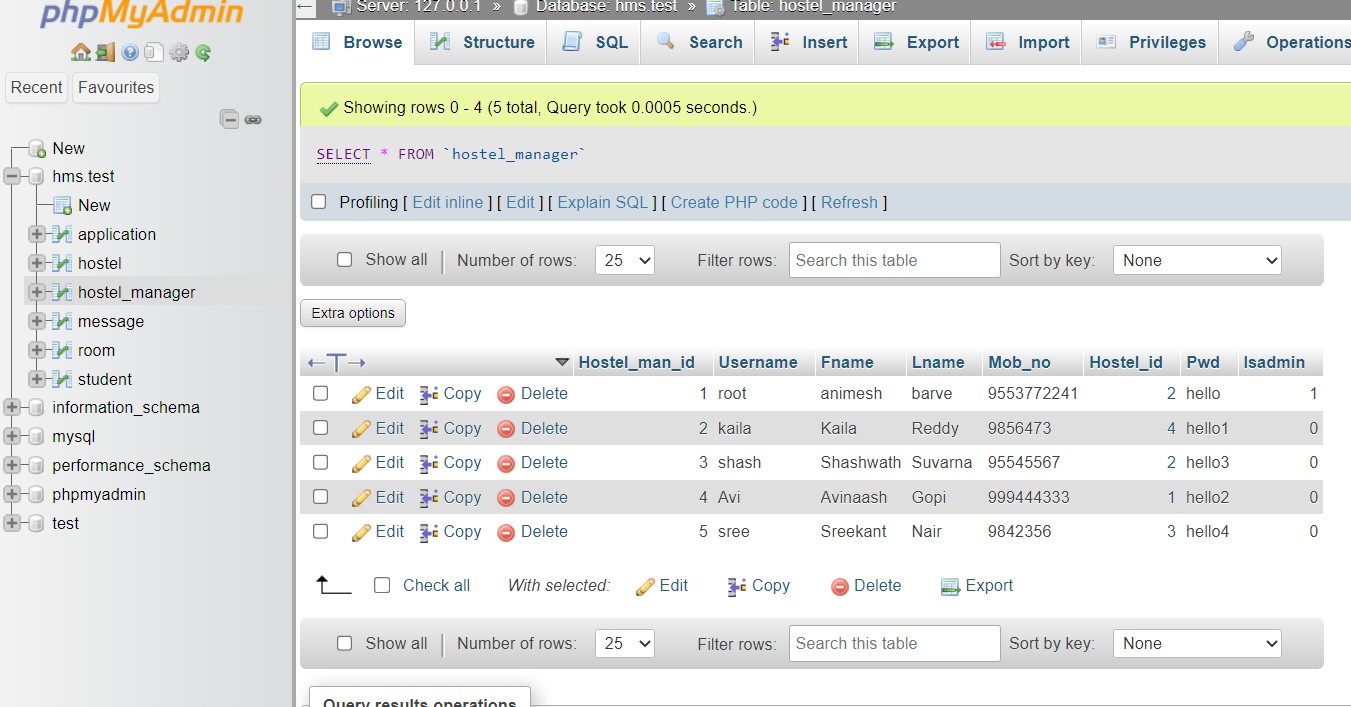
**APPLICATION TABLE**



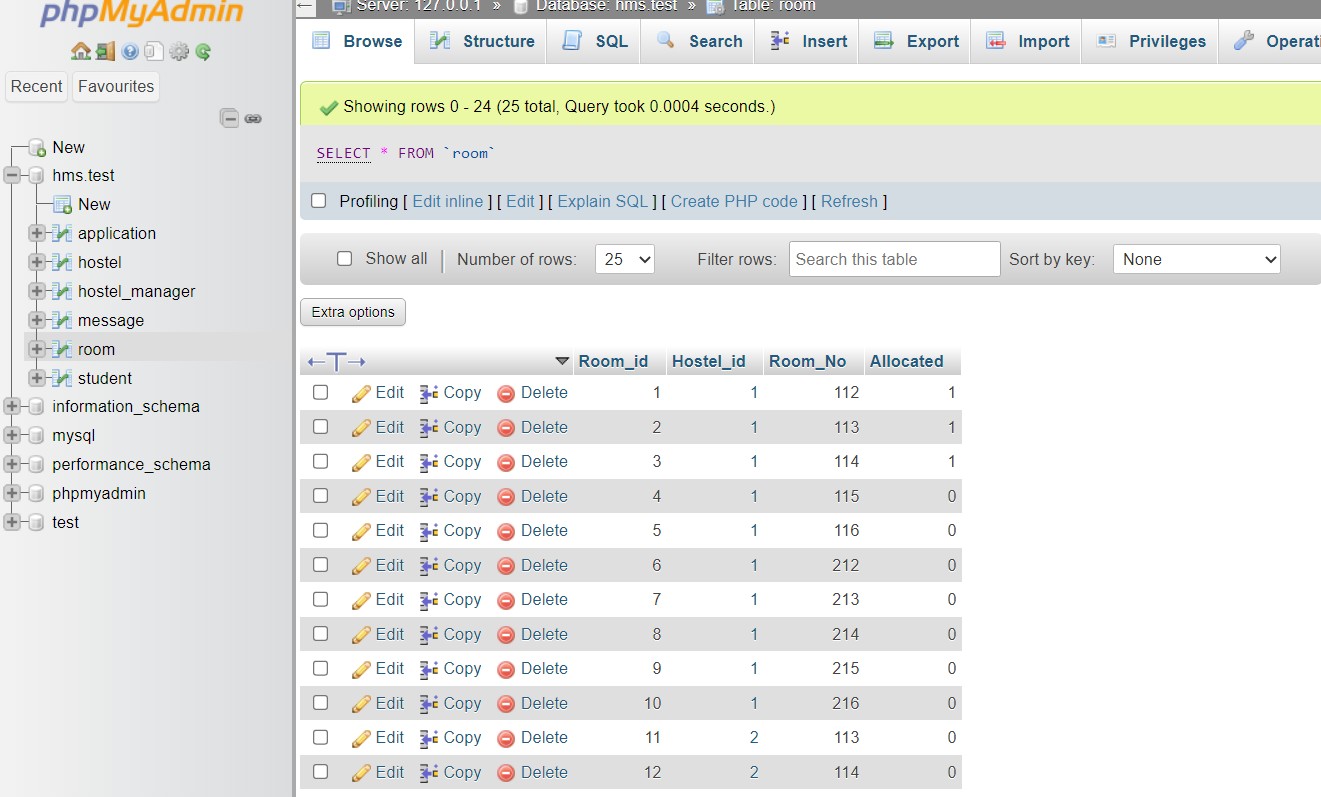
**HOSTEL TABLE**



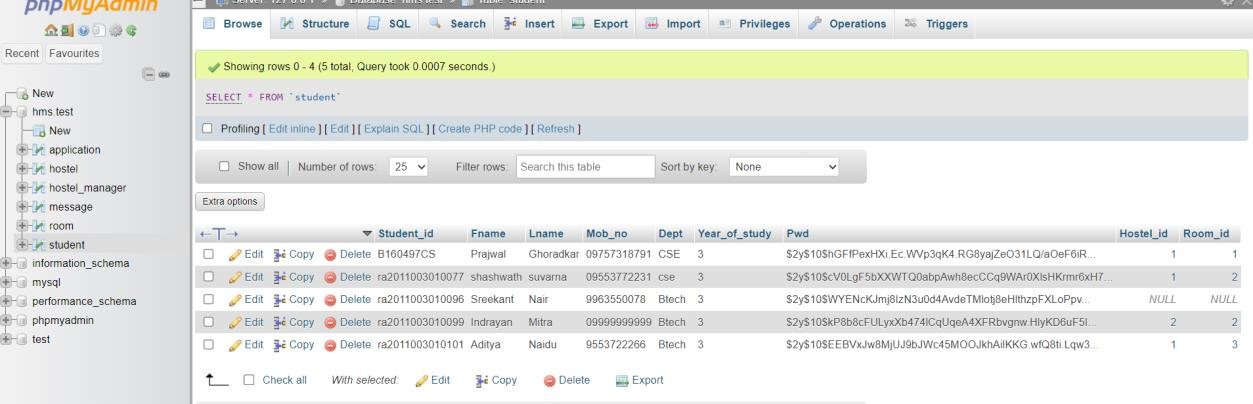
**HOSTEL MANAGER TABLE**



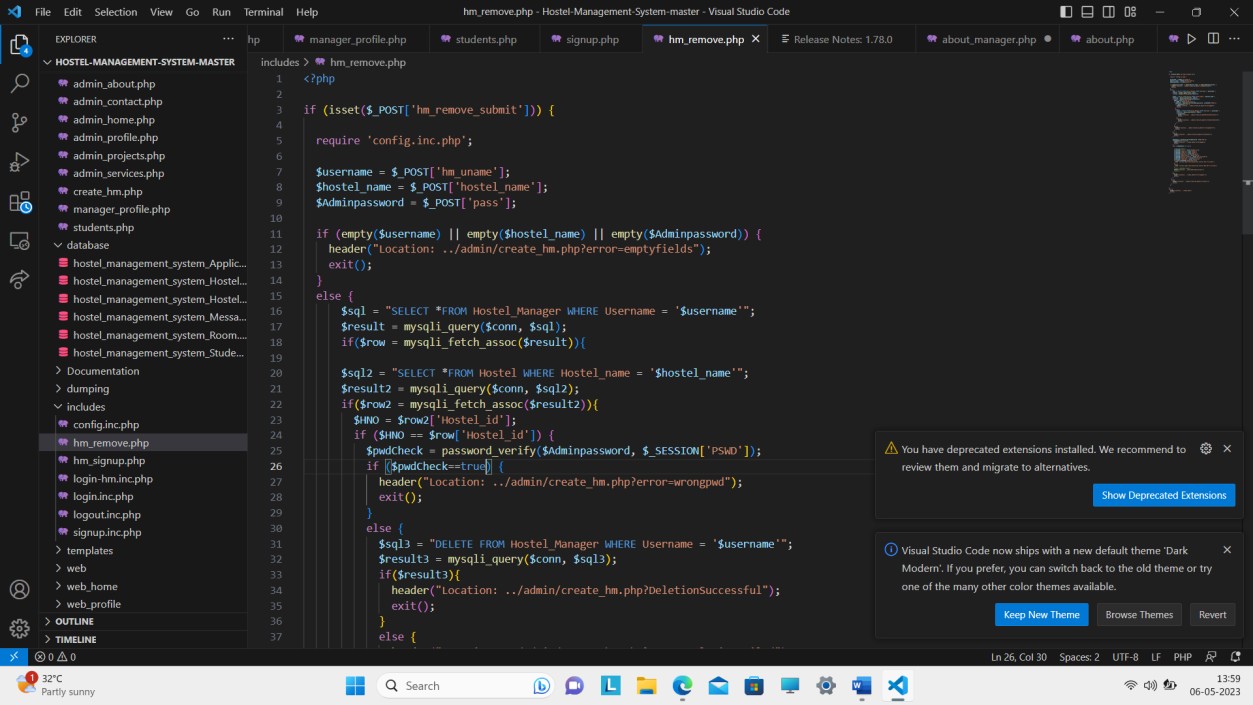
**ROOM TABLE**



**STUDENT TABLE**



**PHP FILE FOR FRONTEND IN VSCODE**



# CHAPTER 6 RESULT AND DISCUSSION

This project is aimed at developing a system for keeping records and showing information about or in a hostel. This system will help the hostel officer to be able to manage the affairs of the hostel. This system will provide full information about a student in the hostel. It will show rooms available or not and number of people in a particular room. This will also provide information on students who have paid in full or are still owing. This system will also provide a report on the summary detail regarding fees and bills students are owing. Also included is a user module for employees or the hostel officer.

# CHAPTER 7

**CONCLUSION AND FURTHER ENHANCEMENT**

To conclude the description about the project, the project, developed using PHP with My SQL is based on the requirement specification of the user and the analysis of the existing system, with flexibility for future enhancement. HOSTEL MANAGEMENT SYSTEM is very useful for hostel allotment and mess fee calculation. This hostel management software is designed for people who want to manage various activities in the hostel. For the past few years the numbers of educational institutions are increasing rapidly. Thereby the numbers 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 are running the hostel and software’s are not usually used in this context. 29 This particular project deals with the problems on managing a hostel and avoids the problems which occur when carried manually. Identification of the drawbacks of the existing system leads to the designing of computerized system that will be compatible to the existing system with the system which is more user friendly and more GUI oriented.

# REFERENCES

"Hostel Management System" by Khwaja Hamid Ali, published in the International Journal of Scientific and Research Publications, Volume 4, Issue 4, April 2014.

"Design and Implementation of Hostel Management System using PHP and MySQL" by Akinwande Akindele Oluwaseun, published in the Journal of Software Engineering and Applications, Volume 7, Issue 8, August 2014.

"Hostel Management System Using Android Application" by V.S. Raja, published in the International Journal of Innovative Research in Computer and Communication Engineering, Volume 4, Issue 6, June 2016.

"A Review of Hostel Management System" by Saba Sadiq and Munir Ahmad, published in the International Journal of Advanced Computer Science and Applications, Volume 6, Issue 12, December 2015.

"Design and Development of Hostel Management System using Visual Basic" by Shobhit Gupta and Ankit Taneja (2016):

"Hostel Management System" by M. Siva Sankar and K. Naga Vignesh (2016)

"Development of Hostel Management System using PHP and MySQL" by Rajesh Kumar and Manoj Kumar (2015)

"Design and Implementation of Hostel Management System using Java" by S. Vengatesan, K. Sathish Kumar, and K. Srinivasan (2017)

"Development of Hostel Management System using ASP.NET and C#" by N. Panchapakesan, S. Selvakumar, and R. Saravanakumar (2016)