

VISVESVARAYA TECHNOLOGICAL UNIVERSITY

“Jnana Sangama”, Belagavi - 590018, Karnataka, India



MINI PROJECT REPORT

On

“HOSTEL MANAGEMENT SYSTEM”

Submitted in partial fulfillment of the requirements for the award of the Degree

BACHELOR OF ENGINEERING

In

ELECTRONICS AND COMMUNICATION ENGINEERING

Submitted By

ABHISHEK ANAND

1DA21EC004

ANIKET PRAKASH

1DA21EC016

ABDUL NASIR

1DA21EC002

Under the Guidance of

Dr. Umadevi H

Professor, Department of ECE, Dr. AIT, Bengaluru – 560056



DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

Dr. AMBEDKAR INSTITUTE OF TECHNOLOGY

MALLATHAHALLI, OUTER RING ROAD, BENGALURU – 560056

2023-24

Dr. AMBEDKAR INSTITUTE OF TECHNOLOGY
Mallathahalli, Bengaluru – 560056

Department of Electronics & Communication Engineering



Certificate

Certified that the project work entitled “**Hostel Management System**”, carried out by Student Abhishek Anand, Aniket Prakash and Abdul Nasir bearing USN: USN No. 1DA21EC004, 1DA21EC016 and 1DA21EC002 respectively, bonafide students of **Dr. Ambedkar Institute of Technology, Bangalore – 560056** in partial fulfillment for the award of Bachelor of Engineering in Electronics and Communication Engineering of the **Visvesvaraya Technological University, Belagavi** during the year 2022–2023. It is certified that all the corrections/suggestions indicated for Internal Assessment have been incorporated in the Report deposited in the departmental library. The Mini project report has been approved as it satisfies the academic requirements.

Signature of the Guide

Dr. Umadevi H

Professor,
Dept. of ECE
Dr. AIT, Bengaluru-56

Signature of the HOD

Dr. Ramesh S

Professor and Head,
Dept. of ECE,
Dr AIT, Bengaluru-56

Signature of the Principal

Dr. C. Nanjudaswamy

Principal,
Dr. AIT, Bengaluru-56

External Viva

Name of the Examiners

1. Dr. Meenakshi L Rathod

Signature with Date

Dr. AMBEDKAR INSTITUTE OF TECHNOLOGY
Mallathahalli, Bengaluru – 560056

Department of Electronics and Communication Engineering



Declaration

We, **ABHISHEK ANAND**, bearing USN : **1DA21EC004**, **ANIKET PRAKASH** bearing USN : **1DA21EC016**, **ABDUL NASIR**, bearing USN: **1DA21EC002**, hereby declare that, the project work entitled “**Hostel Management System**” is independently carried out by us at Department of Electronics and Communication Engineering, **Dr. Ambedkar Institute of Technology, Bengaluru-560056**, under the guidance of **Dr. Umadevi H**, Professor, Department of Electronics and Communication Engineering, Dr. Ambedkar Institute of Technology. The Project work is carried out in partial fulfillment of the requirement for the award of degree of Bachelor of Engineering in Electronics and Communication Engineering during the academic year 2023- 2024.

Place: Bengaluru

Date: 12/07/2024

Name & Signature of students

ABHISHEK ANAND

ANIKET PRAKASH

ABDUL NASIR

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of this mini project would be complete only with the mention of the people who made it possible, whose support rewarded our effort with success.

We are grateful to Dr. Ambedkar Institute of Technology for its ideals and its inspirations for having provided us with the facilities that have made this mini project a success.

We are grateful to our Principal **Dr. C Nanjundaswamy**, Dr. Ambedkar Institute of Technology, who gave a continuous support and provided us comfortable environment to work in.

We would like to express our sincere thanks to **Dr. Ramesh S**, Professor and Head, Department of Electronics and Communication Engineering, Dr. Ambedkar Institute of Technology for his support. We pay out profound gratefulness and express our deepest gratitude to our mini project guide **Dr. Umadevi H**, Professor, Department of Electronics and Communication Engineering for the suggestions and guidance.

We are thankful to our mini project coordinators Dr. Divya A, Assistant Professor, Dept. of Electronics and communication and Sangeetha N, Assistant Professor, Dept. of Electronics and communications for their advice, supervision and guidance throughout the course of the project.

It is our pleasure to acknowledge the cooperation extended by teaching staff and non-teaching staff members of Dept. of Electronics and Communication Engineering, Dr. Ambedkar Institute of Technology for the encouragement during project work. Finally, it gives immense pleasure to acknowledge the cooperation extended by family members, friends for the encouragement during this Project Work.

ABHISHEK ANAND

ANIKET PRAKASH

ABDUL NASIR

ABSTRACT

The Hostel Management System (HMS) is a comprehensive software application designed to streamline and automate the operations of hostel management. This mini project aims to develop a robust and user-friendly HMS to handle various administrative tasks, such as student registration, room allocation, fee management, and maintenance tracking. The system leverages web technologies to provide an accessible platform for both hostel administrators and residents.

The primary objectives of the HMS are to improve efficiency, reduce manual workload, and enhance the overall management of hostel operations. Key features of the system include secure user authentication, intuitive dashboards for administrators, and easy-to-navigate interfaces for students. The HMS also integrates notification mechanisms to keep users informed about important updates, such as fee payment deadlines and maintenance schedules.

Developed using PHP for backend processes and MySQL for database management, the HMS ensures data security and integrity while offering seamless performance. The project demonstrates the effective use of modern web development practices to solve real-world problems in hostel management. By implementing this system, hostels can achieve significant improvements in operational efficiency, data accuracy, and resident satisfaction.

INDEX

Chapter 1	Introduction	Pg.No.
1.1	Introduction	1
1.2	Problem Statement	2
Chapter 2	Literature Survey	
2.1	Literature	3
2.2	Survey	3-4
2.3	Objectives	4-5
Chapter 3	Proposed Work	
3.1	Methodology	6
3.2	Flow chart	7-8
3.3	Requirements and Specifications	9
3.4	Implementation	10-11
Chapter 4	Code	12-25
Chapter 5	Result	
4.1	Results snaps	26-35
4.2	Application	36-37
4.3	Advantages	37-39
4.4	Disadvantages	39-41
Chapter 6	Conclusion	42
	Future Scope	43-44
	Reference	45

CHAPTER 1

INTRODUCTION

1.1 Hostel Management System

"HOSTEL MANAGEMENT SYSTEM" is software developed for managing various activities in the hostel. For the past few years, the number of educational Institutions is increasing rapidly. Thereby the number of hostels is also increasing for the accommodation of the students studying in the 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. 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. In the following sections, a brief introduction about the tools, languages and the databases used to develop the project are discussed.

This software is a simple admin as well as user-based website which provides the user the ease to manage their hostel details safely without any extra effort.

Admin View

As from the perspective of Admin, This website greatly helps to manage all the Hostel related data as number of rooms in hostel, available rooms, number of workers present in hostel, courses from which the students are staying in hostel, and so on many other data too.

User View

As from the perspective of User, In this case students, They can easily check all their hostel related data at one place. Users can also edit their personal details in the hostel database which makes their and also hostel staff's work easier. Users can also print their Hostel Fee Details directly from the website.

1.2 Problem Statement

The identified problem in the Hostel Management System project is the inefficiencies and challenges faced by existing hostel management systems in educational institutions. These challenges include manual processes, fragmented systems, lack of transparency, hindering administrative efficiency, and impacting the student experience negatively. The current systems lack automation, effective communication channels, and optimized resource allocation, leading to errors and inefficiencies in tasks such as student registration, room allocation, billing, and reporting. The manual processes and lack of transparency hinder administrative effectiveness and student satisfaction.

CHAPTER 2

LITERATURE SURVEY

2.2 Literature

Overview of Hostel Management Systems (HMS): Hostel Management Systems (HMS) are software applications designed to automate and manage the administrative functions of hostels. Key areas of focus include student registration, room allocation, fee management, and maintenance tracking.

Automation and Efficiency:

- Studies have shown that automating hostel management tasks significantly improves efficiency by reducing manual workload and errors. Automation helps in streamlining processes like student registration, room allocation, and fee collection (Ranjan et al., 2019).

Security and Privacy:

- Ensuring the security and privacy of student data is a critical concern. Research highlights the importance of secure authentication mechanisms, such as biometric systems and RFID cards, and data encryption to protect sensitive information (Kumar & Singh, 2020).

User Experience and Accessibility:

- Enhancing user experience through intuitive and accessible interfaces is crucial. Literature emphasizes the need for responsive design and user-friendly navigation to make the system easy to use for both administrators and students (Sharma et al., 2018).

2.2 Survey

Technologies Used in HMS:

- **Web-based Technologies:** Many HMS are web-based, utilizing technologies such as HTML, CSS, JavaScript, and PHP. Frameworks like Laravel (PHP) and Django (Python) are popular for building scalable and robust systems (Smith et al., 2021).
- **Mobile Applications:** The use of mobile applications for hostel management is increasing, leveraging platforms like Android and iOS. These apps provide functionalities like room booking, fee payment, and maintenance requests, enhancing convenience for users (Ahmed et al., 2020).

HOSTEL MANAGEMENT SYSTEM

- **Database Management:** Efficient data management is essential, with databases like MySQL, PostgreSQL, and MongoDB being commonly used due to their reliability and performance in handling large datasets (Patel & Desai, 2019).

Case Studies:

- **University Hostel Management Systems:** Many universities have implemented custom HMS tailored to their specific needs. These systems often include features like online room booking, fee payment, and maintenance request management. Case studies indicate significant improvements in operational efficiency and student satisfaction (Wang et al., 2021).
- **Commercial Software Solutions:** Commercial HMS solutions like eZee Absolute and Cloudbeds offer comprehensive features and are widely used by hostels worldwide. These systems provide integrated solutions for various hostel management tasks (Jones & Lee, 2020).

2.2 Objectives

Objective 1: Develop an Automated System:

- Create a system that automates core hostel management tasks such as student registration, room allocation, fee collection, and maintenance tracking to enhance operational efficiency and reduce manual errors.

Objective 2: Enhance Security and Privacy:

- Implement secure authentication mechanisms and data encryption to ensure the protection of sensitive student information.

Objective 3: Improve User Experience:

- Design an intuitive and user-friendly interface that caters to both hostel administrators and students, ensuring ease of use and accessibility.

Objective 4: Utilize Modern Technologies:

- Employ web-based technologies (e.g., PHP, MySQL) and consider developing a mobile application to provide flexible access and management capabilities.

HOSTEL MANAGEMENT SYSTEM

Objective 5: Enable Seamless Communication:

- Integrate notification systems to keep users informed about important updates such as fee payment deadlines, room availability, and maintenance schedules.

Objective 6: Facilitate Comprehensive Reporting:

- Incorporate reporting and analytics features to provide insights into hostel operations, occupancy rates, financial performance, and maintenance issues.

CHAPTER 3

PROPOSED WORK

3.1 Methodology

An entity-relationship model (ER model) is a data model for describing the data or information aspects of a business domain or its process requirements, in an abstract way that lends itself to ultimately begin implemented in the database such as a relational database. The main components of ER model are entities (things) and the relationship that can exist among them. Entity – relationship modeling was developed by Peter chen and published in 1976 paper.

ER diagram is drawn to have a better understanding of the whole scenario, it is used to conceptualize the phenomena, actions and interactions between various entities and to arrive at the specific requirements in a comprehensive manner. An entity relation relationship model is the result of using a systematic process to describe and define a subject area of business data. The data is represented as components (entities) that are linked with each other by relationships that express the dependencies and requirements between them, such as: one building may be divided into number of apartments, but one apartment can be located in one building. Entities may have various properties (attributes) that characterize them. Diagrams created to represent these entities, attributes, and relationships graphically are called entity-relationship diagrams. An ER model is typically implemented as database. In the case of a relational database, which stores data in tables, every row of each table represents one instance of an entity. Some data fields in these tables to indexes in other tables; such pointers are physical implementation of the relationships. The three schema approach to software engineering uses three levels of ER models may be developed.

A relationship captures how entities are related to one another. Relationships can be thought of as verbs, linking two or more nouns. Entities and relationships can both have attributes. The ER diagram for this database is as shown.

Based on the overview of our project, we develop the following Entity Relationship model.

HOSTEL MANAGEMENT SYSTEM

3.2 Flow Chart

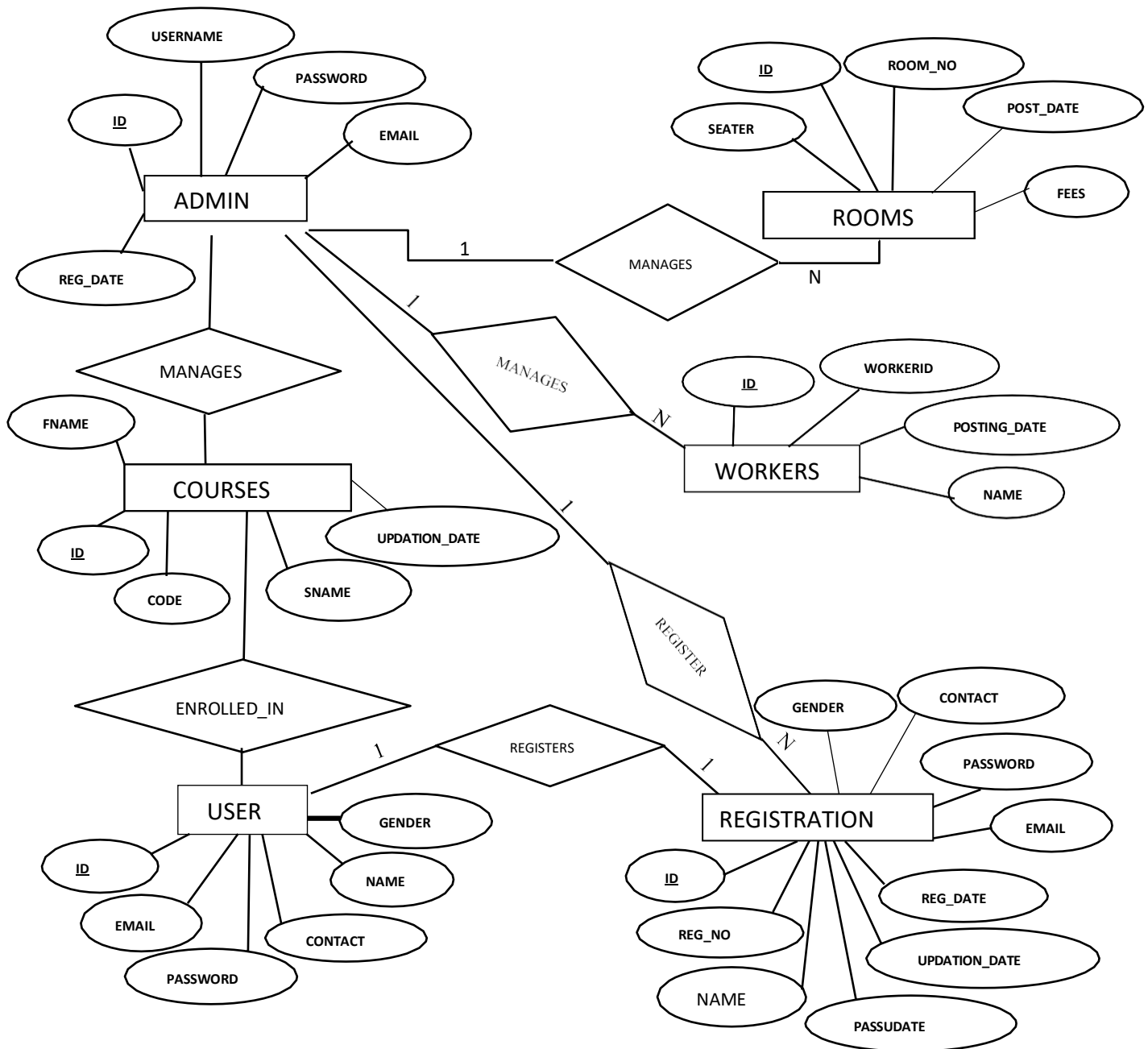


Fig 3.3 . ER Diagram Of Hostel Management System

HOSTEL MANAGEMENT SYSTEM

Schema Diagram

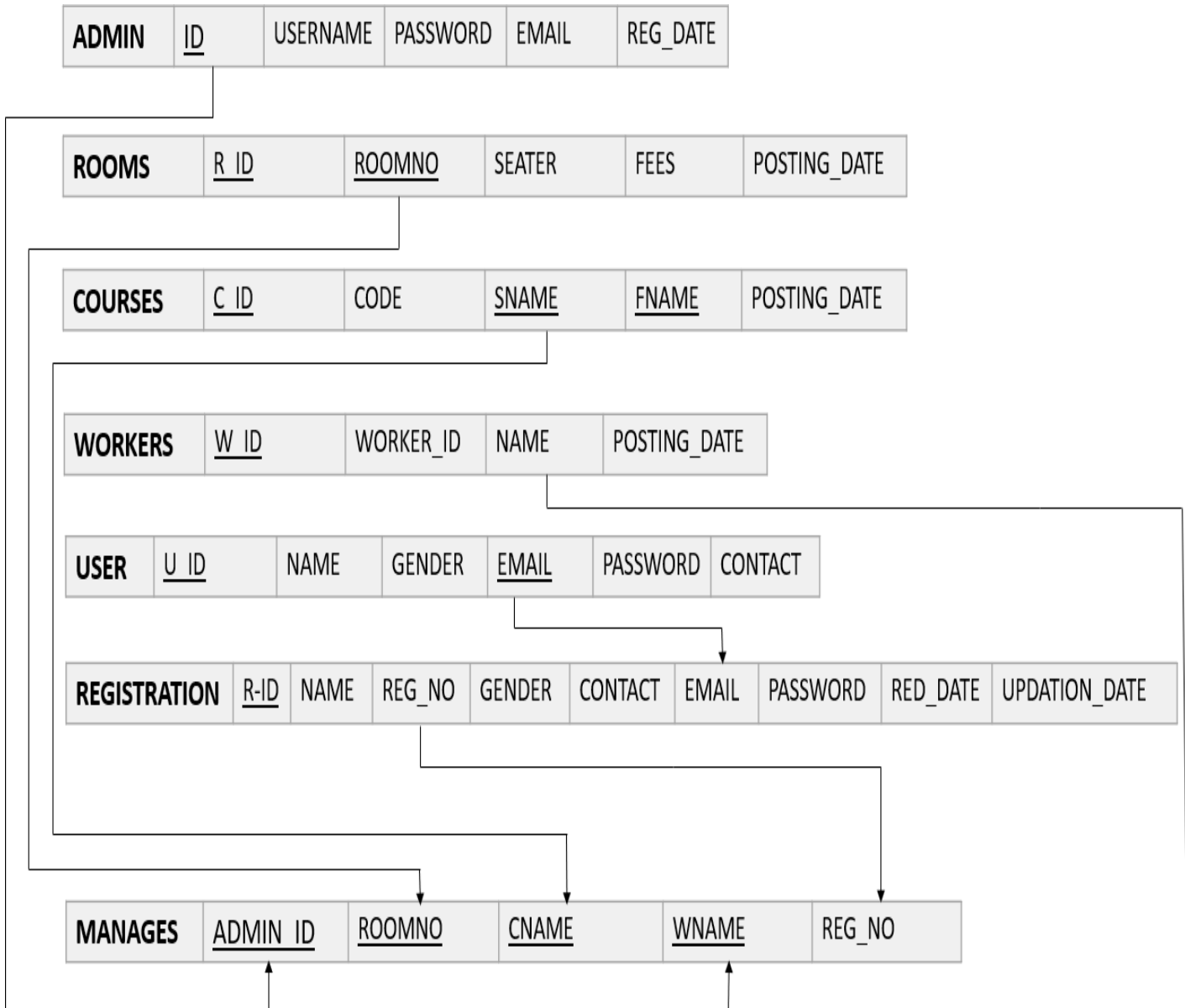


Fig 3.4 Schema Diagram for Hostel Management System

3.3 Requirements and Specifications

Hardware:

Processor: Intel Core i5 1155G7

RAM :8 GB DDR4 RAM

Monitor: 14’’ LED

Storage:512 GB SSD,1TB HDD

Keyboard: STANDARD 102 KEYS

Software:-

Operating System: Windows11

Tools: PHP, XAAMPP server 8.2.12

Backend Software: MySQL 8.3.0

3.4 Implementation

Part 1: System Architecture and Technologies

System Architecture:

- **Frontend:** Develop a responsive web application using HTML, CSS, and JavaScript for the user interface.
- **Backend:** Utilize PHP for server-side scripting to handle business logic and interaction with the database.
- **Database:** Implement MySQL as the relational database management system to store and manage data securely.

Part 2: Core Features and Modules

Core Features:

1. User Management:

- Implement user authentication and authorization mechanisms for administrators, staff, and students.
- Allow administrators to manage user roles and permissions.

2. Student Management:

- Develop functionalities for student registration, including capturing personal details and allocating rooms based on availability and preferences.
- Enable updating and tracking student information throughout their stay in the hostel.

3. Room Allocation and Management:

- Design modules to manage room inventory, including adding new rooms, updating room details, and handling room transfers or changes.

4. Fee Management:

- Implement a module for fee collection, allowing students to view their fee status, make payments online, and generate receipts.
- Include features for administrators to track fee payments and generate financial reports.

5. Maintenance Tracking:

- Develop a system for students to submit maintenance requests online, specifying the issue and urgency.
- Enable administrators to assign maintenance tasks, track their status, and generate reports on maintenance activities.

Part 3: User Interface and Experience

User Interface Design:

HOSTEL MANAGEMENT SYSTEM

- Focus on creating a clean and intuitive interface that is easy to navigate for both administrators and students.
- Implement responsive design principles to ensure compatibility and usability across devices (desktops, tablets, mobile phones).

Accessibility:

- Ensure accessibility features are integrated, such as screen reader compatibility and keyboard navigation, to accommodate users with disabilities.

Part 4: Testing and Deployment

Testing:

- Conduct comprehensive testing, including unit testing for individual modules and integration testing to ensure all components work together seamlessly.
- Perform usability testing with potential users to gather feedback and make necessary improvements.

Deployment:

- Deploy the HMS on a secure web hosting environment, ensuring data encryption and backup mechanisms are in place.
- Monitor performance and conduct regular maintenance to address any issues and ensure smooth operation.

Part 5: Objectives

Objectives:

1. **Automate Administrative Tasks:** Develop a system that automates hostel management processes to improve efficiency and reduce manual errors.
2. **Enhance Security:** Implement secure authentication and data encryption to protect sensitive student information.
3. **Improve User Experience:** Design an intuitive and user-friendly interface that enhances accessibility and usability for all users.
4. **Utilize Modern Technologies:** Employ PHP for backend development, MySQL for database management, and responsive web design principles to leverage current technologies effectively.

CHAPTER 4

CODING

Creation of Tables

Creation of Admin Table

```
CREATE TABLE IF NOT EXISTS admin (  
    id int(11) NOT NULL,  
    username varchar(255) NOT NULL,  
    email varchar(255) NOT NULL,  
    password varchar(300) NOT NULL,  
    reg_date timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,  
    updation_date date NOT NULL  
);
```

Creation of Courses Table

```
CREATE TABLE IF NOT EXISTS courses (  
    id int(11) NOT NULL,  
    course_code varchar(255) NOT NULL,  
    course_sn varchar(255) NOT NULL,  
    course_fn varchar(255) NOT NULL,  
    posting_date timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP  
);
```

Creation of Workers Table

```
CREATE TABLE IF NOT EXISTS workers (  
    id int(11) NOT NULL,  
    name varchar(40) NOT NULL,  
    workers_id int(4) NOT NULL,  
    posting_date timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP  
);
```

Creation of Rooms Table

```
CREATE TABLE IF NOT EXISTS rooms (  
    id int(11) NOT NULL,  
    seater int(11) NOT NULL,  
    room_no int(11) NOT NULL,  
    fees int(11) NOT NULL,  
    posting_date timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP  
);
```

Creation of UserRegistration Table

```
CREATE TABLE IF NOT EXISTS userregistration (  
    id int(11) NOT NULL,  
    regNo varchar(255) NOT NULL,  
    firstName varchar(255) NOT NULL,  
    middleName varchar(255) NOT NULL,  
    lastName varchar(255) NOT NULL,  
    gender varchar(255) NOT NULL,  
    contactNo bigint(20) NOT NULL,
```

```
email varchar(255) NOT NULL,  
password varchar(255) NOT NULL,  
regDate timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,  
updataDate varchar(45) NOT NULL,  
passUpdateDate varchar(45) NOT NULL  
);
```

Creation of Registration Table

```
CREATE TABLE IF NOT EXISTS registration (  
    id int(11) NOT NULL,  
    roomno int(11) NOT NULL,  
    seater int(11) NOT NULL,  
    feespm int(11) NOT NULL,  
    foodstatus int(11) NOT NULL,  
    stayfrom date NOT NULL,  
    duration int(11) NOT NULL,  
    course varchar(500) NOT NULL,  
    regno int(11) NOT NULL,  
    firstName varchar(500) NOT NULL,  
    middleName varchar(500) NOT NULL,  
    lastName varchar(500) NOT NULL,  
    gender varchar(250) NOT NULL,  
    contactno bigint(11) NOT NULL,  
    emailid varchar(500) NOT NULL,  
    egycontactno bigint(11) NOT NULL,  
    guardianName varchar(500) NOT NULL,  
    guardianRelation varchar(500) NOT NULL,  
    guardianContactno bigint(11) NOT NULL,
```

HOSTEL MANAGEMENT SYSTEM

```
corresAddress varchar(500) NOT NULL,  
corresCity varchar(500) NOT NULL,  
corresState varchar(500) NOT NULL,  
corresPincode int(11) NOT NULL,  
pmntAddress varchar(500) NOT NULL,  
pmntCity varchar(500) NOT NULL,  
pmntState varchar(500) NOT NULL,  
pmntPincode int(11) NOT NULL,  
postingDate timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,  
updateDate varchar(500) NOT NULL  
);
```

Insertions in the Table

Insertion to Admin table

```
INSERT INTO admin (id, username, email, password, reg_date, update_date) VALUES  
(1, 'admin', 'admin.ait@gmail.com', 'admin@1234', '2024-01-27 20:31:45', '2024-01-28');
```

Insertion to Courses table

```
INSERT INTO courses (`id`, `course_code`, `course_sn`, `course_fn`, `posting_date`) VALUES  
(1, 'B10992', 'B.Tech', 'Bachelor of Technology', '2024-06-27 19:31:42'),  
(2, 'BCOM1453', 'B.Com', 'Bachelor Of commerce ', '2024-06-27 19:32:46'),  
(3, 'BSC12', 'BSC', 'Bachelor of Science', '2024-06-27 19:33:23'),  
(4, 'BC36356', 'BCA', 'Bachelor Of Computer Application', '2024-06-27 19:34:18'),  
(5, 'MCA565', 'MCA', 'Master of Computer Application', '2024-06-27 19:34:40'),  
(6, 'MBA75', 'MBA', 'Master of Business Administration', '2024-06-27 19:34:59'),  
(7, 'BE765', 'BE', 'Bachelor of Engineering', '2024-06-27 19:35:19');
```

HOSTEL MANAGEMENT SYSTEM

Insertion to Registration table

```
INSERT INTO registration (`id`, `roomno`, `seater`, `feespm`, `foodstatus`, `stayfrom`,  
`duration`, `course`, `regno`, `firstName`, `middleName`, `lastName`, `gender`, `contactno`,  
`emailid`, `egycontactno`, `guardianName`, `guardianRelation`, `guardianContactno`,  
`corresAddress`, `corresCity`, `corresState`, `corresPincode`, `pmntAddress`, `pmntCity`,  
`pmnatetState`, `pmntPincode`, `postingDate`, `updatetime`) VALUES  
(5, 100, 3, 4000, 1, '2024-01-27', 2, 'Bachelor of Technology', 10806121, 'Rahul', '', 'yadav',  
'male', 8285703354, 'anuj.ait@gmail.com', 8285703354, 'Priya', 'Mother', 8285703354,  
'H no. 18/1 Bihari Puram Phase-1 Melrose Bye Pass', 'Aligarh', 'Uttar Pradesh', 202001,  
'H no. 18/1 Bihari Puram Phase-1 Melrose Bye Pass', 'Aligarh', 'Uttar Pradesh', 202001,  
'2024-01-27 06:32:43', '');  
(6, 100, 5, 8000, 0, '2022-01-22', 5, 'Bachelor of Technology', 10806121, 'Anuja', '', 'kumari',  
'female', 8285703354, 'anuja.ait@gmail.com', 0, 'Supriya', 'Mother', 8285703354,  
'H no. 18/1 Bihari Puram Phase-1 Melrose Bye Pass', 'Aligarh', 'Uttar Pradesh', 202001,  
'H no. 18/1 Bihari Puram Phase-1 Melrose Bye Pass', 'Aligarh', 'Uttar Pradesh', 202001,  
'2024-01-27 08:24:09', '');
```

Insertion to Rooms table

```
INSERT INTO rooms (`id`, `seater`, `room_no`, `fees`, `posting_date`) VALUES  
(1, 5, 100, 8000, '2024-06-27 01:25:43');  
(2, 2, 201, 6000, '2024-06-27 01:30:47');  
(3, 2, 200, 6000, '2024-06-27 01:30:58');  
(4, 3, 112, 4000, '2024-06-27 01:31:07');  
(5, 5, 132, 2000, '2024-06-27 01:31:15');
```

Insertion to User Registration table

```
INSERT INTO userregistration (`id`, `regNo`, `firstName`, `middleName`, `lastName`, `gender`,  
`contactNo`, `email`, `password`, `regDate`, `updateDate`, `passUpdateDate`) VALUES  
(10, '108061211', 'code', 'test', 'projects', 'male', 8467067344, 'test@gmail.com', 'Test@123',  
'2024-06-27 04:21:33', '2024-06-27 11:04:15', '');  
(19, '102355', 'Harry', 'projects', 'Singh', 'male', 6786786786, 'Harry@gmail.com', '6786786786',  
'2024-06-27 16:33:36', '', '');  
(20, '586952', 'Benjamin', '', 'projects', 'male', 8596185625, 'Benjamin@gmail.com',  
'8596185625', '2024-06-27 16:40:07', '', '');
```

Insertion to Workers table

```
INSERT INTO `workers` (`id`, `workers_id`, `name`, `posting_date`) VALUES  
(1, 11, 'Ratna ck', '2024-06-27 01:31:07');  
(2, 12, 'Ramu', '2024-06-27 03:27:10');  
(3, 13, 'Raghu', '2024-06-27 07:15:49');  
(4, 14, 'Shreya', '2024-06-27 10:38:16');  
(5, 15, 'Amma', '2024-06-27 19:35:19');
```

Connectivity to Database

A system to satisfy specified requirements. Systems design could see it as the application of systems theory to product development. There is some overlap with the disciplines of systems analysis, systems architecture and systems engineering. If the broader topic of product development “blends the perspective of marketing, design, and manufacturing into a single approach to product development,” then design is the act of taking the marketing information and creating the design of the product to be manufactured. Systems design is therefore the process of defining and developing systems to satisfy specified requirements of the user. Until the 1990s systems design had a crucial and respected role in the data processing industry. In the 1990s standardization of hardware and software resulted in the ability to build modular systems. The increasing importance of software running on generic platforms has enhanced the discipline of software engineering. Object-oriented analysis and design methods are

becoming the most widely used methods for computer systems design. The UML has become the standard language in object-oriented analysis and design. It is widely used for modelling software systems and is increasingly used for high designing non-software systems and organizations.[citation needed] System design is one of the most important phases of software development process. The purpose of the design is to plan the solution of a problem specified by the requirement documentation. In other words the first step in the solution to the problem is the design of the project.

Connecting from Front End to Backend PHP Code

- 1) mysqli connect () To connect to MySQL using the MySQL Improved extension, follow these steps:

- a) Use the following PHP code to connect to MySQL and select a database. Replace username with your username, password with your password, and dbname with the database name:

```
<?php
$mysqli = new mysqli("localhost", "username", "password", "dbname");
?>
```

- b) After the code connects to MySQL and selects the database, you can run SQL queries and perform other operations. The connectivity code used in this database is as follows:

```
<?php
$servername = "localhost";
$username = "root";
$password = "";
$db = "hostel";

// Create connection
$con = mysqli_connect($servername, $username, $password,$db);

// Check connection if (!$con) {
die("Connection failed: " . mysqli_connect_error());
}
?>
```


2) close() –

Closing a Database Connection. It is not always necessary to close a connection when you are finished, but it is advised. It is, however, necessary to close the connection to the database if you want to open up a new connection to a different database. To close a connection to a database, we use the `mysql_close()` function, as follows: `mysql_close();`

3) Error Handling - It is useful when debugging, and even when you just want to make sure that a database does not behave unexpectedly. Once a query has been created via the `mysql_query()` function, any error messages generated will be stored in the `mysql_error()` function. Here is a sample code snippet to display a error message. However, when there is no error messages, a blank string is returned. `print mysql_error();`

Front End page Code

```
<?php
session_start();
include('includes/config.php');
if(isset($_POST['login']))
{
$email=$_POST['email'];
$password=$_POST['password'];
$stmt=$mysqli->prepare("SELECT email,password,id FROM userregistration WHERE email=? and
password=? ");

$stmt->bind_param('ss',$email,$password);
$stmt->execute();
$stmt -> bind_result($email,$password,$id);
$rs=$stmt->fetch();
$stmt->close();
$_SESSION['id']=$id;
$_SESSION['login']=$email;
```

HOSTEL MANAGEMENT SYSTEM

```
$uip=$_SERVER['REMOTE_ADDR'];

$date=date('d/m/Y h:i:s', time());

if($rs)
{
    $uid=$_SESSION['id'];
    $uemail=$_SESSION['login'];
    $ip=$_SERVER['REMOTE_ADDR'];
    $log="insert into userLog(userId,userEmail,userIp,city,country) values(?,?,?,?,?)";
    $mysqli->query($log);
    if($log)
    {
        header("location:dashboard.php");
    }
}
else
{
    echo "<script>alert('Invalid Username/Email or password');</script>";
}
}

?>

<!doctype html>

<html lang="en" class="no-js">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1, minimum-scale=1, maximum-scale=1">

    <meta name="description" content="">
```

```
<meta name="author" content="">

<meta name="theme-color" content="#3e454c">

<title>Student Hostel Registration</title>

<link rel="stylesheet" href="css/font-awesome.min.css">

<link rel="stylesheet" href="css/bootstrap.min.css">

<link rel="stylesheet" href="css/dataTables.bootstrap.min.css">>

<link rel="stylesheet" href="css/bootstrap-social.css">

<link rel="stylesheet" href="css/bootstrap-select.css">

<link rel="stylesheet" href="css/fileinput.min.css">

<link rel="stylesheet" href="css/awesome-bootstrap-checkbox.css">

<link rel="stylesheet" href="css/style.css">

</head>

<body>

<div class="login-page bk-img" style="background-image: url(img/adminLogin.jpg);">

<div class="form-content">

<div class="container">

<div class="row">

<div class="col-md-6 col-md-offset-3">

<h1 class="text-center text-bold text-light mt-4x">Hostel Management System</h1>

<div class="well row pt-2x pb-3x bk-light">

<div class="col-md-8 col-md-offset-2">

<form action="" class="mt" method="post">

<label for="" class="text-uppercase text-sm">Your Username or Email</label>

<input type="text" placeholder="Username" name="username" class="form-control mb">

<label for="" class="text-uppercase text-sm">Password</label>

<input type="password" placeholder="Password" name="password" class="form-control mb">

<input type="submit" name="login" class="btn btn-primary btn-block" value="login" >

</form>
```

```
</div>
</div>
</div>
</div>
</div>
</div>
</div>
<script src="js/jquery.min.js"></script>
<script src="js/bootstrap-select.min.js"></script>
<script src="js/bootstrap.min.js"></script>
<script src="js/jquery.dataTables.min.js"></script>
<script src="js/dataTables.bootstrap.min.js"></script>
<script src="js/Chart.min.js"></script>
<script src="js/fileinput.js"></script>
<script src="js/chartData.js"></script>
<script src="js/main.js"></script>
</body>
</html>
```

4.5.1 Admin Login Page Code

```
<?php
session_start();

include('includes/config.php');

if(isset($_POST['login']))
{
$username=$_POST['username'];
$password=$_POST['password'];
```

HOSTEL MANAGEMENT SYSTEM

```
$stmt=$mysqli->prepare("SELECT    username,email,password,id    FROM    admin    WHERE
(userName=?|| email=?) and password=? ");

$stmt->bind_param('sss',$username,$username,$password);

$stmt->execute();

$stmt -> bind_result($username,$username,$password,$id);

$rs=$stmt->fetch();

$_SESSION['id']=$id;

$uip=$_SERVER['REMOTE_ADDR'];

$date=date('d/m/Y h:i:s', time());

if($rs)
{
// $insert="INSERT into admin(adminid,ip)VALUES(?,?)";
// $stmtins = $mysqli->prepare($insert);
// $stmtins->bind_param('sH',$id,$uip);
// $res=$stmtins->execute(); header("location:admin-profile.php");
}
else
{
echo "<script>alert('Invalid Username/Email or password');</script>";
}
}
?>

<!doctype html>

<html lang="en" class="no-js">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1, minimum-scale=1,
maximum-scale=1">
```

```
<meta name="description" content="">
<meta name="author" content="">
<title>Admin login</title>
<link rel="stylesheet" href="css/font-awesome.min.css">
<link rel="stylesheet" href="css/bootstrap.min.css">
<link rel="stylesheet" href="css/dataTables.bootstrap.min.css">
<link rel="stylesheet" href="css/bootstrap-social.css">
<link rel="stylesheet" href="css/bootstrap-select.css">
<link rel="stylesheet" href="css/fileinput.min.css">
<link rel="stylesheet" href="css/awesome-bootstrap-checkbox.css">
<link rel="stylesheet" href="css/style.css">
</head>
<body>
<div class="login-page bk-img" style="background-image: url(img/adminLogin.jpg);">
<div class="form-content">
<div class="container">
<div class="row">
<div class="col-md-6 col-md-offset-3">
<h1 class="text-center text-bold text-light mt-4x">Hostel Management System</h1>
<div class="well row pt-2x pb-3x bk-light">
<div class="col-md-8 col-md-offset-2">
<form action="" class="mt" method="post">
<label for="" class="text-uppercase text-sm">Your Username or Email</label>
<input type="text" placeholder="Username" name="username" class="form-control mb">
<label for="" class="text-uppercase text-sm">Password</label>
<input type="password" placeholder="Password" name="password" class="form-control
mb">
<input type="submit" name="login" class="btn btn-primary btn-block" value="login" >
```

```
</form>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

<script src="js/jquery.min.js"></script>
<script src="js/bootstrap-select.min.js"></script>
<script src="js/bootstrap.min.js"></script>
<script src="js/jquery.dataTables.min.js"></script>
<script src="js/dataTables.bootstrap.min.js"></script>
<script src="js/Chart.min.js"></script>
<script src="js/fileinput.js"></script>
<script src="js/chartData.js"></script>
<script src="js/main.js"></script>

</body>

<style> .foot{text-align: center; border: 1px solid black;}</style>

</html>
```

CHAPTER 5

RESULT

5.1 Result snaps

The screenshot shows the 'User Login' page of the 'Hostel Management System'. The browser address bar indicates the URL is 'localhost/hostel/index.php'. On the left, a dark sidebar menu contains 'User Registration', 'User Login', and 'Admin Login'. The main content area has a title 'User Login' and a form with two input fields: 'EMAIL / REGISTRATION NUMBER' (placeholder: 'Email / Registration Number') and 'PASSWORD' (placeholder: 'Password'). Below these fields is a blue 'login' button. A link 'Forgot password?' is located below the button. The Windows taskbar at the bottom shows the date as 11-07-2024 and time as 11:04.

Fig. 5.1.1.home page

The screenshot shows the 'Student Registration' page of the 'Hostel Management System'. The browser address bar indicates the URL is 'localhost/hostel/registration.php'. The sidebar menu is the same as in the previous screenshot. The main content area has a title 'Student Registration' and a form titled 'FILL ALL INFO'. The form contains the following fields: 'Registration No:', 'First Name:', 'Middle Name:', 'Last Name:', 'Gender:' (a dropdown menu with 'Select Gender' selected), 'Contact No:', 'Email id:', 'Password:', and 'Confirm Password:'. The Windows taskbar at the bottom shows the date as 11-07-2024 and time as 11:05.

Fig. 5.1.2.User registration page

Admin's Side

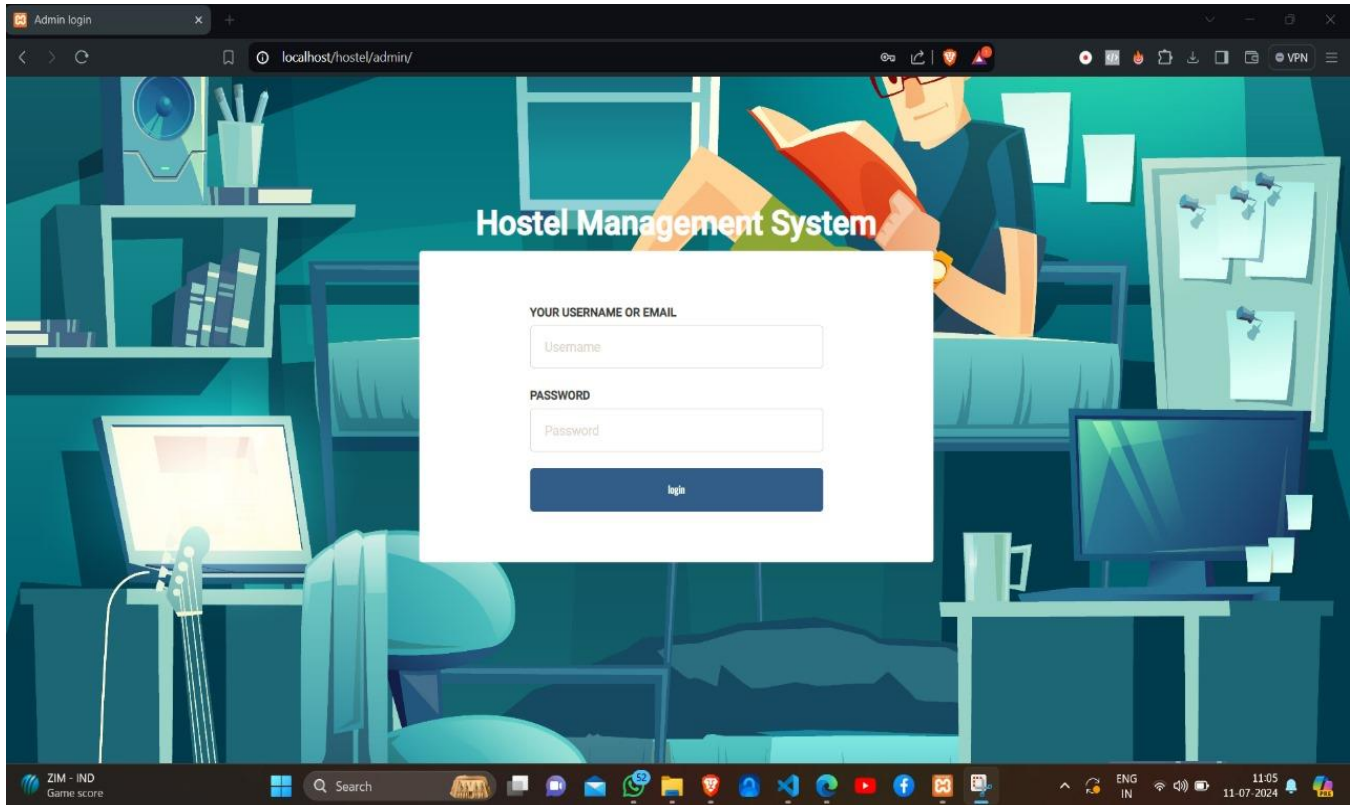


Fig.5.2.1. Admin Login Page

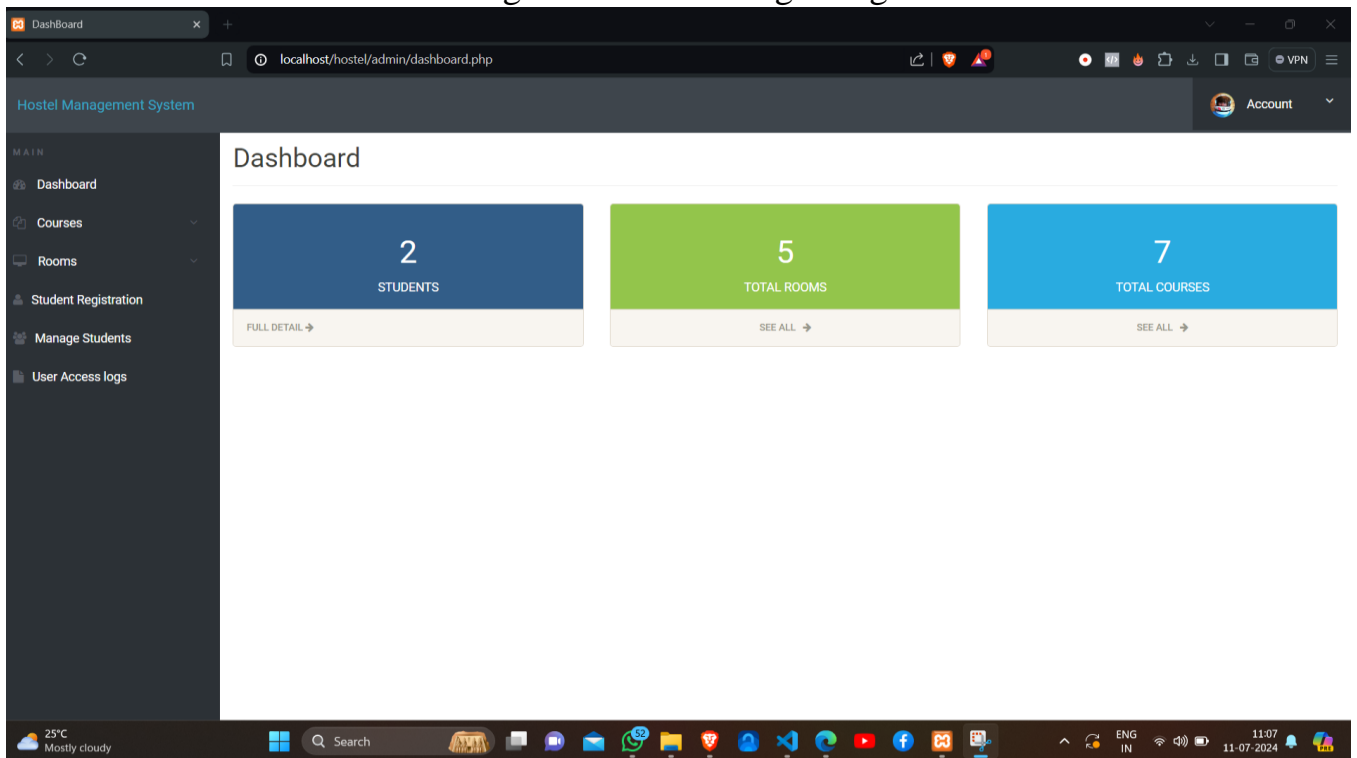


Fig.5.2.2. Admin's profile page

HOSTEL MANAGEMENT SYSTEM

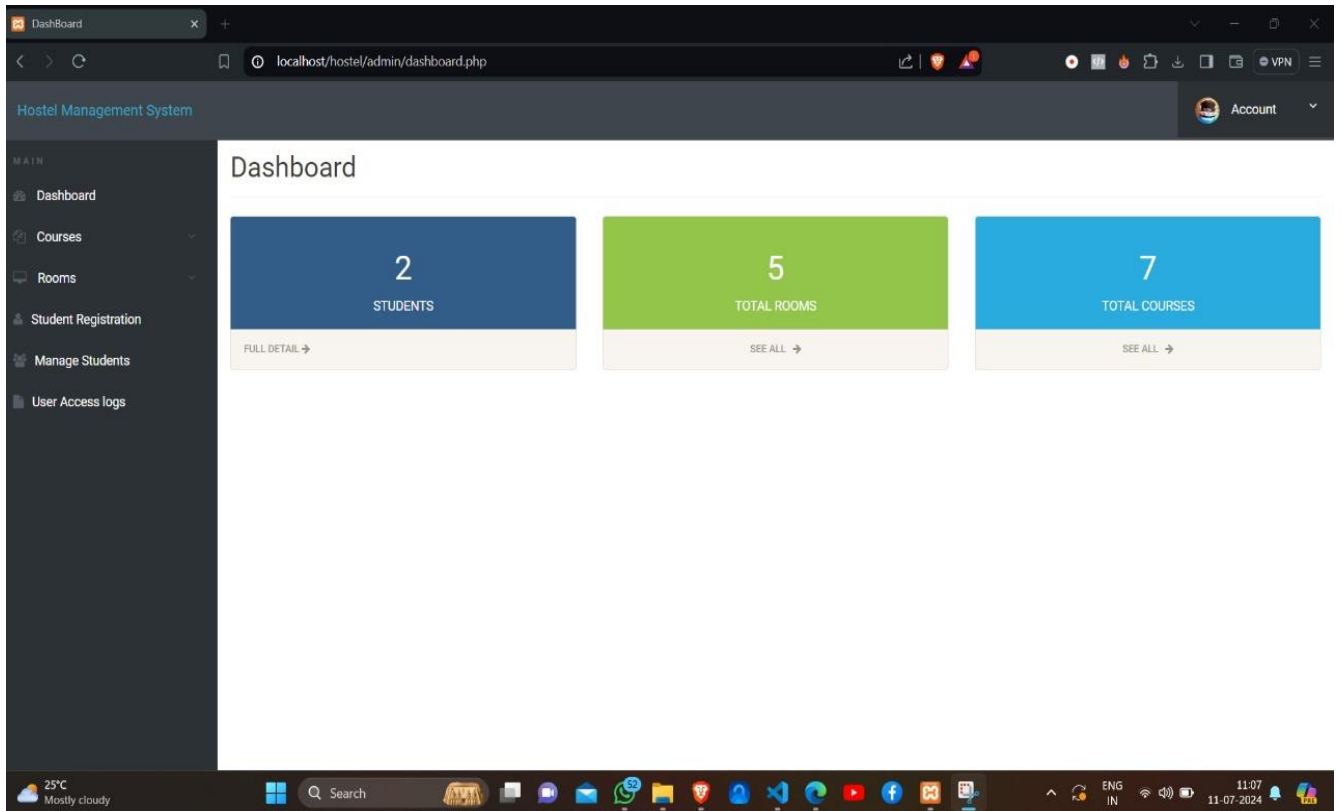


Fig.5.2.3 Admin's dashboard

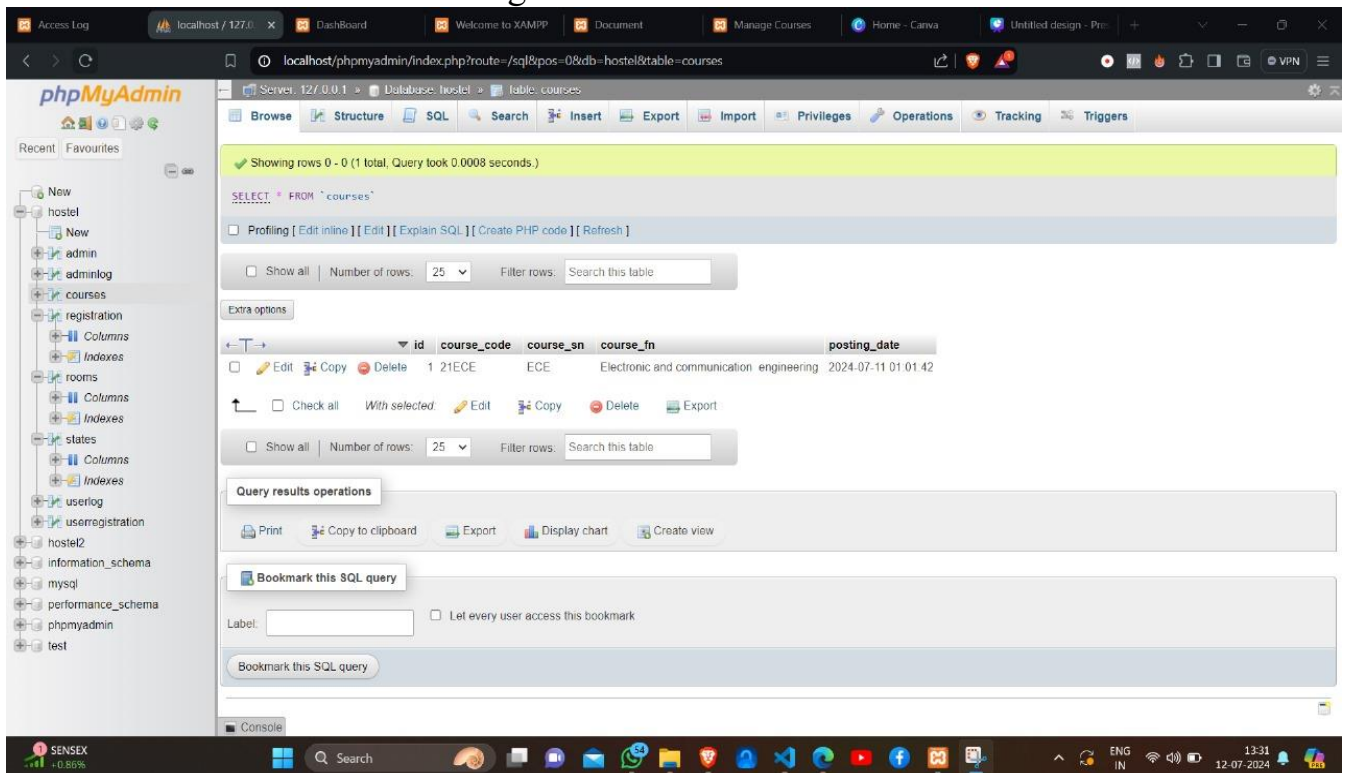


Fig.5.2.4 Add courses page

HOSTEL MANAGEMENT SYSTEM

The screenshot displays the 'Add a Room' interface of the Hostel Management System. The browser address bar shows the URL `localhost/hostel/admin/create-room.php`. The system's header includes the title 'Hostel Management System' and a user account dropdown. A dark sidebar on the left lists navigation items: Dashboard, Courses, Rooms, Student Registration, Manage Students, and User Access logs. The main content area, titled 'Add a Room', features a form with the following elements:

- A dropdown menu labeled 'Select Seater' with the placeholder text 'Select Seater'.
- A text input field labeled 'Room No.'.
- A text input field labeled 'Fee(Per Student)'.
- A blue button labeled 'Create Room'.

The Windows taskbar at the bottom shows the system clock as 11:08 on 11-07-2024, along with various application icons and system status indicators.

Fig.5.2.5 Manage rooms page

HOSTEL MANAGEMENT SYSTEM

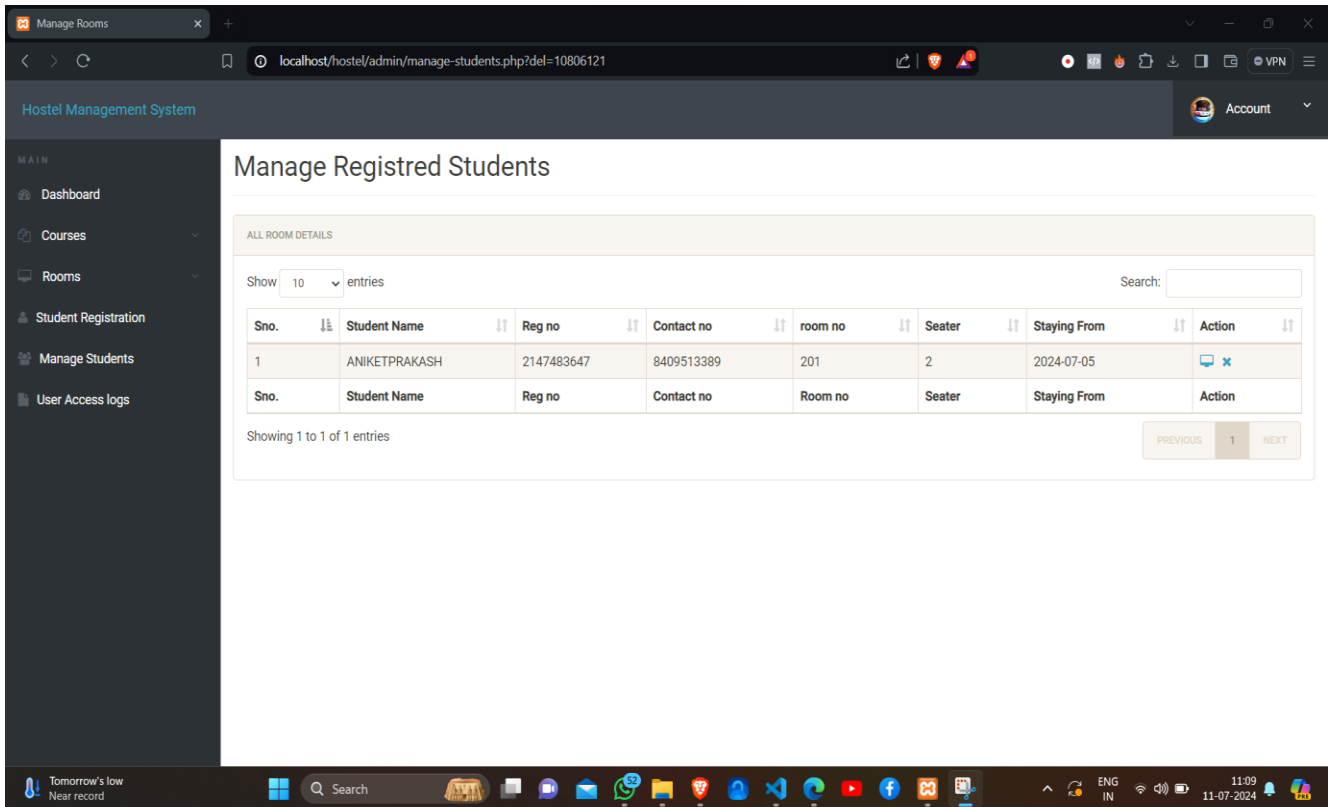


Fig.5.2.6 Manage Students page

User's Side

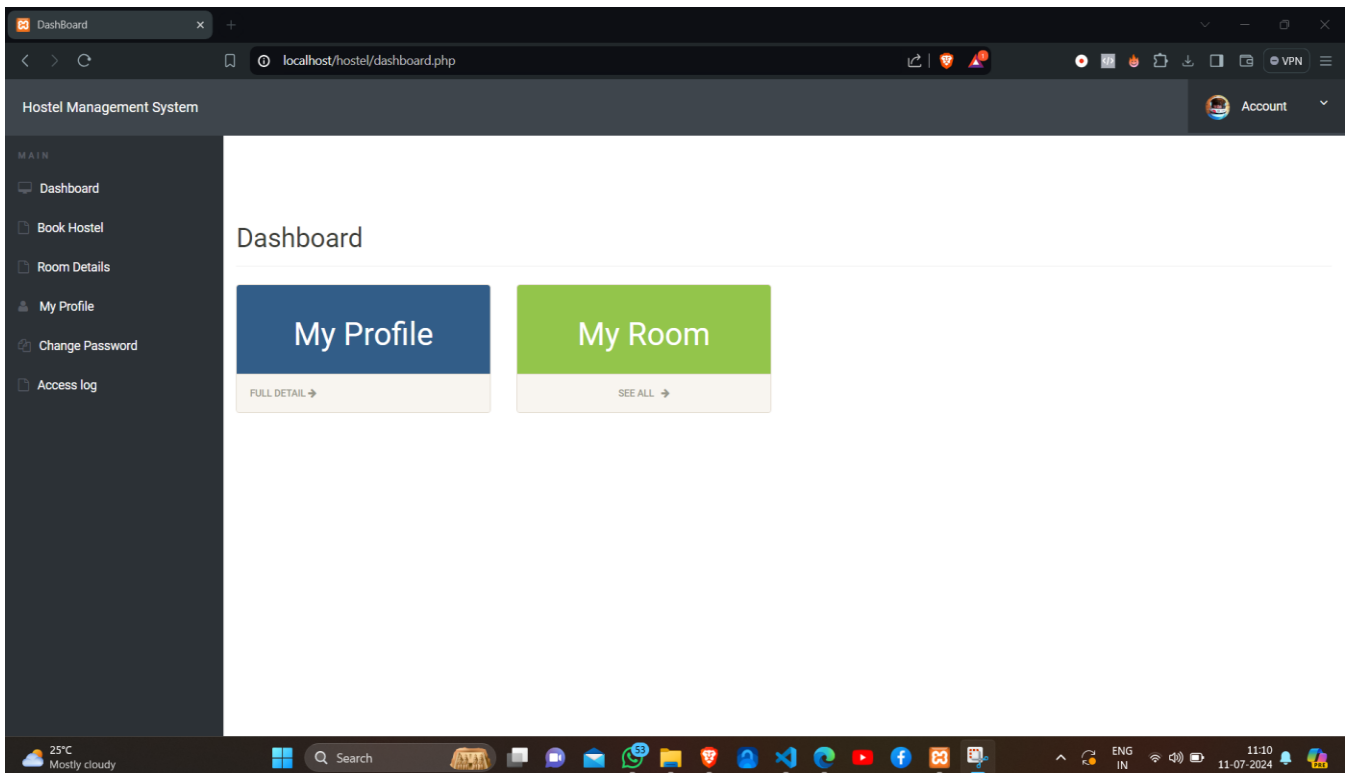


Fig.5.3.1 User's dashboard page

HOSTEL MANAGEMENT SYSTEM

The screenshot shows a web browser window with the URL `localhost/hostel/my-profile.php`. The page title is "Hostel Management System". On the left, there is a sidebar menu with the following items: Dashboard, Book Hostel, Room Details, My Profile (highlighted), Change Password, and Access log. The main content area is titled "ANIKET's Profile". It features a "LAST UPDATE DATE:" header. Below this, there are several input fields for profile information: Registration No (8409513389), First Name (ANIKET), Middle Name (empty), Last Name (PRAKASH), Gender (male), Contact No (8409513389), and Email id (ani.prakashcho123@gmail.com). An "Update Profile" button is located at the bottom right of the form.

Fig.5.3.2 User's profile page

The screenshot shows a web browser window with the URL `localhost/hostel/book-hostel.php`. The page title is "Hostel Management System". On the left, there is a sidebar menu with the following items: Dashboard, Book Hostel (highlighted), Room Details, My Profile, Change Password, and Access log. The main content area is titled "Registration". It features a "FILL ALL INFO" header. Below this, there is a message "Hostel already booked by you" in red text. A green button labeled "My Room" is displayed, with a "SEE ALL" link below it.

Fig.5.3.3 User's book hostel page

HOSTEL MANAGEMENT SYSTEM

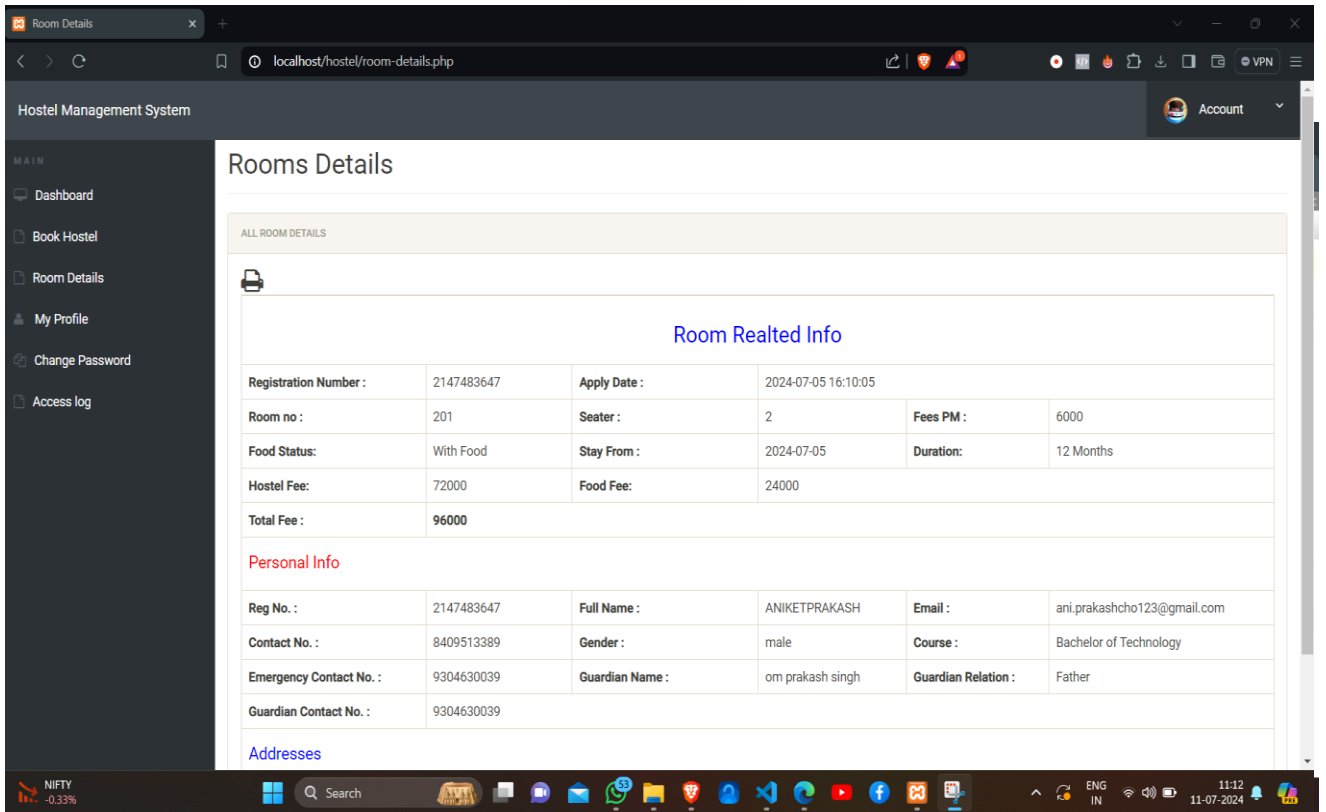


Fig.5.3.4. User's room details page

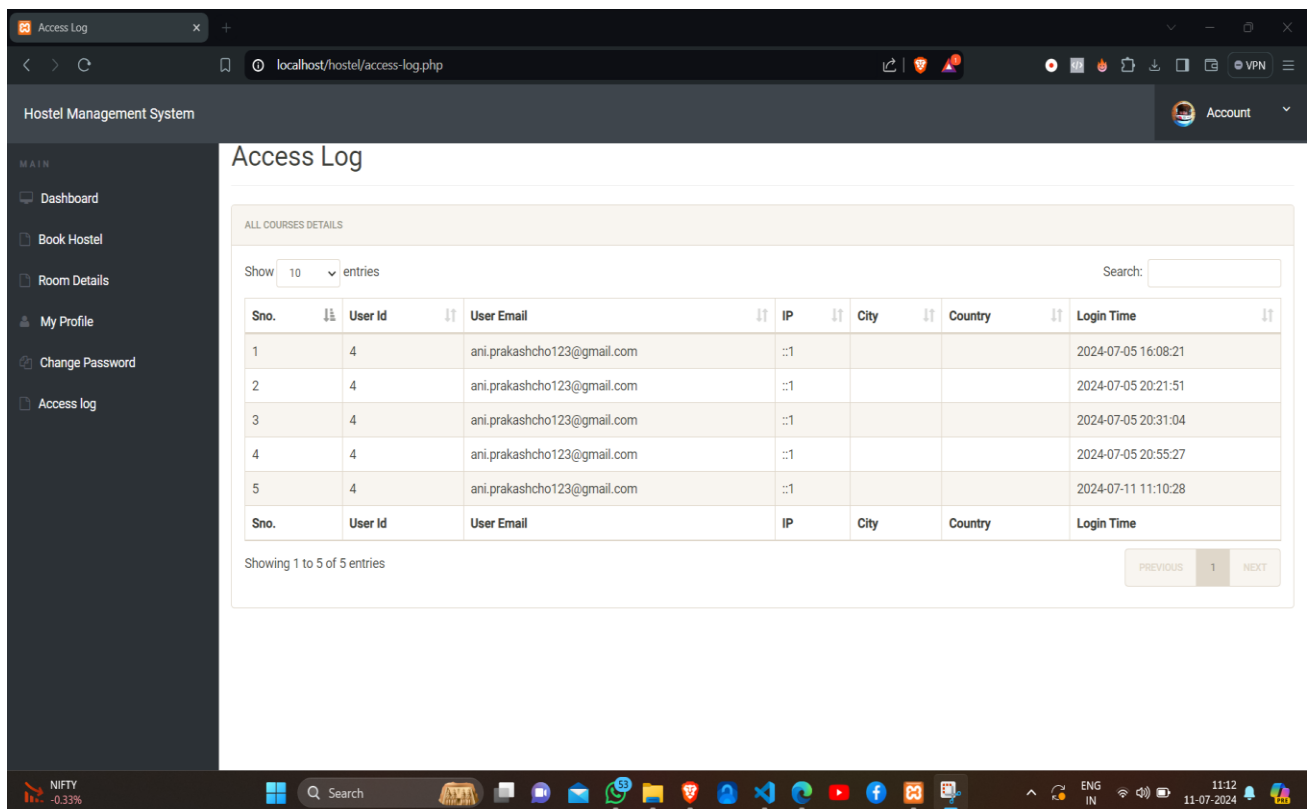


Fig.5.3.5. Access log page of user

Backend Tables Snapshots

Admin Table

The screenshot shows the phpMyAdmin interface for the 'admin' table in the 'hostel' database. The table structure is as follows:

id	username	email	password	reg_date	update_date
1	abhishek	abhishek.lpu1@gmail.com	Test@1234	2024-07-11 02:01:45	2024-04-11

The interface also shows a recent SQL query: `UPDATE `admin` SET `username` = 'abhishek', `email` = 'abhishek.lpu1@gmail.com' WHERE `admin`.`id` = 1;` and a message indicating '1 row affected'.

Courses Table

The screenshot shows the phpMyAdmin interface for the 'courses' table in the 'hostel' database. The table structure is as follows:

id	course_code	course_sn	course_fn	posting_date
1	21ECE	ECE	Electronic and communication engineering	2024-07-11 01:01:42

The interface also shows a recent SQL query: `SELECT * FROM `courses`` and a message indicating 'Showing rows 0 - 0 (1 total, Query took 0.0008 seconds)'.

HOSTEL MANAGEMENT SYSTEM

Registration Table

1 row affected.

```
UPDATE `registration` SET `stayfrom` = '2024-07-11', `course` = 'ECE', `firstName` = 'Abhishek', `lastName` = 'Anand', `emailid` = 'abhishek@gmail.com', `corresAddress` = 'Bengaluru', `corresCity` = 'Bengaluru', `corresState` = 'Karnataka', `pmntAddress` = 'Bengaluru', `pmntCity` = 'Bengaluru', `pmntState` = 'Karnataka', `pmntPincode` = '560056', `postingDate` = '2024-07-11 20:28:26' WHERE `registration`.`id` = 2;
```

Showing rows 0 - 0 (1 total, Query took 0.0004 seconds)

```
SELECT * FROM `registration`
```

id	roomno	seater	feespm	foodstatus	stayfrom	duration	course	regno	firstName	middleName	lastName	gender	contactno	emailid
2	100	5	8000	1	2024-07-11	6	ECE	10806121	Abhishek		Anand	male	1234567890	abhishek@gmail.com

Rooms Table

1 row affected.

```
UPDATE `rooms` SET `posting_date` = '2024-07-11 07:00:58' WHERE `rooms`.`id` = 3;
```

Showing rows 0 - 4 (5 total, Query took 0.0005 seconds)

```
SELECT * FROM `rooms`
```

id	seater	room_no	fees	posting_date
1	5	100	8000	2024-07-11 04:15:43
2	2	201	6000	2024-07-11 07:00:47
3	2	200	6000	2024-07-11 07:00:58

HOSTEL MANAGEMENT SYSTEM

User Registration Table

The screenshot displays the phpMyAdmin interface for a database named 'hostel'. The 'userregistration' table is selected, showing its structure and data. The table has columns: id, regNo, firstName, middleName, lastName, gender, contactNo, email, password, regDate, updateDate, and passUpdateDate. Two rows of data are visible.

	id	regNo	firstName	middleName	lastName	gender	contactNo	email	password	regDate	updateDate	passUpdateDate
<input type="checkbox"/>	4	8409513389	ANIKET		PRAKASH	male	8409513389	ani.prakashcho123@gmail.com	12345678	2024-07-05 16:03:46	NULL	NULL
<input type="checkbox"/>	5	123	abhishek		anand	male	1234567890	avshs@gmail.com	abhishek	2024-07-11 15:54:55	NULL	NULL

Query results operations: Print, Copy to clipboard, Export, Display chart, Create view.

Bookmark this SQL query: ☐ Let every user access this bookmark.

5.2 Applications

1. Student Registration and Profile Management:

- Simplifies the process of registering new students entering the hostel.
- Manages and updates student profiles with personal information, contact details, and academic records.

2. Room Allocation and Management:

- Automates the allocation of rooms based on student preferences, availability, and specific requirements.
- Tracks room occupancy, facilitates room changes, and manages room maintenance schedules.

3. Fee Collection and Financial Management:

- Enables online fee payment options for students, ensuring timely collection and reducing administrative workload.
- Generates fee receipts, tracks payment statuses, and provides financial reports for administrators.

4. Maintenance Tracking and Management:

- Allows students to submit maintenance requests online, specifying issues and urgency.
- Assigns tasks to maintenance staff, tracks progress, and maintains a history of maintenance activities.

5. Communication and Notification:

- Sends automated notifications and reminders to students and staff about upcoming events, fee payment deadlines, maintenance updates, etc.
- Facilitates seamless communication between hostel administration, students, and parents/guardians.

6. Reporting and Analytics:

- Generates detailed reports on room occupancy rates, fee collection summaries, maintenance activities, and financial performance.
- Provides analytics to identify trends, optimize resource allocation, and make data-driven decisions.

7. Security and Access Control:

- Implements secure access control mechanisms, such as biometric authentication or RFID cards, to ensure the safety and security of residents.
- Monitors entry and exit logs to enhance hostel security and accountability.

8. Integration with Academic Systems:

- Integrates with academic management systems to synchronize student data, academic schedules, and hostel accommodations.

HOSTEL MANAGEMENT SYSTEM

- Provides a unified platform for seamless information exchange between academic and residential departments.

9. Guest and Visitor Management:

- Manages guest accommodations and visitor access permissions, ensuring compliance with hostel policies.
- Tracks visitor logs and manages guest stays efficiently.

10. Feedback and Survey Management:

- Collects feedback from students and staff regarding hostel facilities, services, and overall satisfaction.
- Analyzes feedback to improve services, address concerns, and enhance the overall hostel experience.

Benefits of Using a Hostel Management System

- **Efficiency:** Streamlines administrative processes, reduces manual workload, and improves operational efficiency.
- **Accuracy:** Minimizes errors in data management, fee collection, room allocation, and maintenance tracking.
- **Transparency:** Enhances transparency in hostel operations, financial transactions, and communication.
- **Enhanced User Experience:** Provides a user-friendly interface for students, staff, and administrators, improving overall satisfaction.
- **Cost-Effectiveness:** Optimizes resource allocation, reduces unnecessary expenses, and enhances financial management.

5.3 Advantages

1. Automation of Administrative Tasks:

- **Efficiency:** Automates routine tasks such as student registration, room allocation, fee collection, and maintenance requests, reducing manual effort and operational time.
- **Accuracy:** Minimizes errors in data entry and processing, ensuring reliable management of hostel operations.

2. Enhanced Security and Data Privacy:

HOSTEL MANAGEMENT SYSTEM

- **Secure Access:** Implements robust authentication mechanisms and access controls to protect sensitive student information and ensure compliance with data protection regulations.
- **Data Encryption:** Encrypts data both at rest and in transit to prevent unauthorized access and safeguard privacy.

3. Improved Communication and Transparency:

- **Real-Time Notifications:** Sends automated alerts and reminders to students and staff about important updates such as fee deadlines, room changes, and maintenance schedules.
- **Transparent Operations:** Provides visibility into hostel activities, financial transactions, and maintenance status for administrators and residents.

4. Optimized Resource Management:

- **Room Allocation:** Efficiently manages room inventory, allocates rooms based on preferences and availability, and tracks occupancy rates to optimize space utilization.
- **Financial Management:** Tracks fee payments, generates financial reports, and forecasts budget allocations to streamline financial operations.

5. Enhanced User Experience:

- **User-Friendly Interface:** Offers an intuitive and accessible interface for students, staff, and administrators, improving user satisfaction and adoption.
- **24/7 Accessibility:** Provides round-the-clock access to hostel information and services through web and mobile interfaces, enhancing convenience for users.

6. Comprehensive Reporting and Analytics:

- **Insights and Decision-Making:** Generates detailed reports and analytics on occupancy rates, financial performance, maintenance trends, and student demographics to support data-driven decision-making.
- **Predictive Analysis:** Uses historical data and trends to predict future requirements, optimize resource allocation, and improve operational efficiency.

7. Integration with Academic Systems:

HOSTEL MANAGEMENT SYSTEM

- **Seamless Information Flow:** Integrates with academic management systems to synchronize student data, academic schedules, and hostel accommodations, providing a unified platform for holistic student management.
- **Cross-Functional Collaboration:** Facilitates collaboration between academic and residential departments, enhancing coordination and service delivery.

8. Scalability and Flexibility:

- **Adaptability:** Scales to accommodate growing numbers of students, staff, and facilities without compromising performance or functionality.
- **Customization:** Offers flexibility to tailor modules and features to meet specific institutional requirements and operational workflows.

9. Cost-Effectiveness:

- **Resource Optimization:** Reduces operational costs by optimizing resource allocation, minimizing administrative overhead, and preventing revenue leakage through efficient fee management.
- **Long-Term Savings:** Improves financial planning and forecasting, leading to long-term savings and sustainability in hostel management.

10. Compliance and Accountability:

- **Regulatory Compliance:** Ensures compliance with legal and regulatory requirements related to data protection, financial transparency, and hostel operations.
- **Audit Trails:** Maintains audit trails of transactions, activities, and communications for accountability and transparency in governance.

5.4 Disadvantages

1. Initial Cost and Setup:

- **High Initial Investment:** Implementing a robust HMS involves costs related to software development, hardware infrastructure, and training.
- **Complex Implementation:** Integration with existing systems and data migration can be complex and time-consuming, requiring specialized expertise.

2. Dependency on Technology:

HOSTEL MANAGEMENT SYSTEM

- Technical Issues: System downtime, software bugs, or hardware failures can disrupt hostel operations and affect user experience.
 - Training Needs: Staff and users may require training to effectively use new technologies and features, adding to implementation costs.
3. Data Security Risks:
- Cybersecurity Threats: Hostel management systems store sensitive student information, making them potential targets for cyberattacks and data breaches.
 - Compliance Challenges: Ensuring compliance with data protection regulations (e.g., GDPR, HIPAA) requires ongoing monitoring and updates to security protocols.
4. User Resistance and Adoption:
- Resistance to Change: Some staff members and students may resist using new technologies or prefer traditional methods, impacting system adoption rates.
 - Usability Issues: Poorly designed interfaces or complex navigation can hinder user satisfaction and discourage regular usage.
5. Maintenance and Upgrades:
- Ongoing Maintenance: Regular updates, maintenance tasks, and troubleshooting require dedicated IT resources and support.
 - Upgrade Costs: Upgrading software versions or scaling the system to accommodate growth may incur additional costs and disruptions.
6. Integration Challenges:
- Compatibility Issues: Integrating with other institutional systems (e.g., academic management, financial software) may pose compatibility challenges, requiring custom solutions or middleware.
 - Data Synchronization: Ensuring seamless data synchronization and consistency across integrated systems can be complex and prone to errors.
7. Vendor Lock-In:
- Dependency on Vendor: Choosing a specific HMS provider may lead to vendor lock-in, limiting flexibility in switching to alternative solutions or technologies.
 - Service Reliability: Reliance on external vendors for support and updates can impact service reliability and responsiveness.
8. Over-Reliance on Technology:
- Human Error: Despite automation, human errors in data entry, system configuration, or decision-making can still occur and impact system effectiveness.
 - Lack of Personalization: Standardized features and modules may not fully address unique institutional requirements or user preferences.

9. Operational Disruptions:

- System Failures: Unexpected system failures or downtime can disrupt hostel operations, affecting service delivery and user satisfaction.
- Emergency Preparedness: Contingency plans for managing emergencies or system failures may not be adequately addressed, impacting hostel management during critical situations.

CONCLUSION

This Website is made for both Admin as well as Users in this case Students. The basic objective of “Hostel Management System” is that the management of all the facilities provided within the hostel and their working can be easily managed by the Admin. Also, It provides convenience to it’s users to manage their hostel profile.

This is an effective, fast, user-friendly system with the main goal of the project as the reduction of the man power to some extent and stores the data in a systematic manner.

Our website stores the details i.e. Admin details, user registration details, course details, workers details, room details. This website also updates the records as many users registered themselves and/or are registered by Admin or their details are deleted from the database by the Admin, saving in MySQL.

FUTURE SCOPE

1. Integration with Smart Technologies:

- Internet of Things (IoT): Incorporate IoT devices for smart room management, energy efficiency, and predictive maintenance. Sensors can monitor room occupancy, temperature, and utility usage.
- Smart Access Control: Implement biometric scanners, RFID systems, or facial recognition for secure and convenient access to hostel facilities.

2. Artificial Intelligence and Predictive Analytics:

- Predictive Maintenance: Use AI algorithms to predict maintenance needs based on historical data, reducing downtime and improving facility management.
- Machine Learning: Analyze data patterns to optimize room allocation, predict student preferences, and improve resource utilization.

3. Mobile Applications and Cloud Computing:

- Mobile Accessibility: Develop mobile apps for students to manage bookings, payments, and maintenance requests conveniently from their smartphones.
- Cloud Integration: Shift towards cloud-based solutions for scalability, flexibility, and enhanced data security. Cloud platforms can streamline data storage, backup, and accessibility.

4. Enhanced User Experience and Personalization:

- Customizable Interfaces: Offer customizable interfaces and preferences for administrators, staff, and students to tailor their HMS experience.
- Feedback Mechanisms: Integrate feedback mechanisms and analytics to continuously improve user satisfaction and system usability.

5. Blockchain Technology for Security and Transparency:

- Data Security: Use blockchain to enhance data security, transparency, and integrity in financial transactions, student records, and access control systems.
- Credential Management: Explore blockchain-based solutions for managing student credentials, certifications, and academic records securely.

6. Augmented Reality (AR) and Virtual Reality (VR):

- Virtual Tours: Provide virtual tours of hostel facilities for prospective students and parents using VR technology.
- AR Applications: Develop AR applications for real-time navigation within the hostel premises, locating facilities, and accessing information.

7. Integration with Academic and ERP Systems:

HOSTEL MANAGEMENT SYSTEM

- Seamless Integration: Enhance integration with academic management systems (e.g., student information systems) and enterprise resource planning (ERP) software for streamlined operations and data synchronization.
- Cross-Functional Collaboration: Foster collaboration between academic departments, hostel management, and administrative units for holistic student support and services.

8. Sustainability Initiatives:

- Green Technologies: Implement eco-friendly practices and technologies for energy conservation, waste management, and sustainable hostel operations.
- Carbon Footprint Reduction: Monitor and reduce the hostel's environmental impact through data-driven sustainability initiatives.

9. Global Accessibility and Multilingual Support:

- Multilingual Interfaces: Support multiple languages to accommodate international students and diverse user demographics.
- Global Reach: Expand HMS capabilities to manage hostels across different geographical locations and cultural contexts.

10. Regulatory Compliance and Governance:

- Adaptation to Regulations: Stay updated with evolving data protection regulations (e.g., GDPR, CCPA) and compliance standards relevant to hostel operations.
- Ethical Use of Data: Ensure ethical use of student data and privacy protection in accordance with legal requirements and institutional policies.

REFERENCE

Few of the book(s) and websites that were instrumental in helping us to complete this project are as mentioned below.

BOOKS

1. Fundamental of Database System by Elmasri and Navathe ,5th Edition, AddisonWesley,2007.
2. Database System Concepts by Avi Silberschatz, Henry F Korth, and S. Sudharshan,1996.
3. Concepts of Database Management by Philip J. Pratt,2008
4. Modern Database Management by Jeffery A Hoffer,2010

URL

1. <https://www.w3schools.com>
2. <https://www.youtube.com>
3. <https://www.google.co.in>
4. <https://www.wikipedia.org>