Hostel Management System

ACKNOWLEDGEMENT

We have made efforts in this project. However, it would not have been possible without many individuals' support and help. We want to extend our sincere thanks to all of them.

I am highly indebted to the Department of <>, Faculty of <>, the <> University of <> for their guidance and constant supervision, for providing necessary information regarding the project & also for their support in completing the project.

I would like to express my gratitude towards my parents & members of ICT for their kind co-operation and encouragement, which helped us complete this project.

I would like to express my special gratitude and thanks to our lecturer <> for giving us such a project and time.

My thanks and appreciation also go to my colleague in developing the project and the people who have willingly helped me with their abilities.

Table of Contents

List of Figure

List of Table

Abstract

Hostel Management System (HMS) is a process of planning, arranging, allocating, and managing the hostel in effective and efficient manner without any intervention. The Hostel Management (HS) is somewhat different and difficult task compared to any other management task. There are variety of problems has been identified from the hostel management issues, such as, delay in getting hostel accommodation, problem in hostel payment, different course students have been mixed in the hostel, lack of decision-making process and issues in cost. The project provides a solution by proving the web-based environment to monitor the student, course, and room service in on view. The system is built with the Bootstrap, HTML, CSS, and JavaScript as a front-end and PHP as a back end. The project contains admin and student role which has different sessions in their interface. The MySQL database has been utilized for constructing the system. It is simple for the administrator to gain access to and can remain unchanged for a significant time without being compromised. Without a shadow of a doubt, both the students and the admin benefit from using this system

Chapter 01

01. Introduction

Hostel Management System (HMS) is a process of planning, arranging, allocating, and managing the hostel in effective and efficient manner without any intervention. The Hostel Management (HS) is somewhat different and difficult task compared to any other management task. The specific students from all over the country involving in this activity for a certain period. The traditional university HMS involves with record book services. The student needs to get a hostel room based on random based which is done by the hostel warden and other related crews. Also, students do not aware about the fees and other procedure mostly. There are many more problems are facing by the students to get a room in each year. The record book is paper based system which is prone to error most of the time. The security, cost, availability, confidentiality, and reliability are the most important problems in paper-based systems.

In any university there are four years with eight semesters. There are many students from far distance need to utilize the hostel service to avoid the accommodation issue. Maintaining the thousand of students are not easier task and it need effective and optimal approach. Some university offer hostels in 1^{st} year and 4^{th} year. However, some universities offer hostel accommodations for all eight semesters, hence the complexity of maintenance is increased with the responsibility to the management. On the other hand, the academic staffs, and non-academic staff from outside also utilizing the hostel services. As a management crew of the hostel management need extra assistance and hand to overcome those complexity. Still government can not hire more staffs to manage the complexity, since universities funds allocated for some other stuffs as well. And the hostel related issues are nightmare since long ago. These management problems need a very powerful and sophisticated solutions to avoid up coming years a greater number of intakes in university.

After web 2.0 and the rapid development of ICT, most impossible tasks became possible. The are many more management tasks are managed easily by the solutions from ICT. The website development is one of the most important applications of ICT, which is used to manage by different sector from small to medium organizations to manage considerable amount of task. There are many more websites are doing authoritative management task which are more helpful for humans these days. The Facebook, YouTube, Twitter, and E-commerce websites are the some of the web-based application in day-to-day human life with management of trilliant of people.

The aim of this project is to develop a dynamic web-based application to manage the students with hostel problem in order to provide a feasible solution to the hostel management issue.

1.1 Motivation of the Project

Many universities in Sri Lanka still proceeding with the legacy system of Hostel Management to maintain hostel related information, also sunk into the paper-based and spreadsheet maintenance of the Student Information. Whenever a student requires information regarding hostel, it takes time to handle those requests since the paper-based and spreadsheet filters are used. The following are some of the other issues we have listed to get motivated in the project.

- Checking the available room for students
- Subject wise room allocation
- Checking the payment details on time
- Issues in Delete, Update, and View room, course, and student information.

1.2 Objective of the Project

The primary goal of this project is to create an effective and efficient Hostel Management System that will automatically provide all information and monitoring the hostel room related issues. The following goals are intended to be accomplished unless the suggested system is successfully implemented.

- Maintain the room database to ease the room allocation.
- Create a web-based system to handle all the problems we are currently facing.
- Begin tracking students' room allocation and payment details with the new system, which comprises students' hostel issues.

Chapter 02

2. Requirement Analysis

2.1 Functional Requirements

a. Admin

- **Student Registration.**
  - Student must register in the HMS website for hostel related operation.
  - In the student registration process, student must provide their name, email id, etc.
  - username and password needed for future login.

- **Admin Login.**
  - Registered admin can login.
  - The app must be able to display the message upon a successful user login and generate an error message in Invalid attempt.

- **Add and Delete Student**
  - Admin able to add a student with their authority.
  - Admin able to delete a student with their authority.

- **Add Room**
  - Admin can add a room based on the availability of room.
  - The admin can add room number, bed, and fees per month.

- **Mange Room**
  - When the room is added, admin can be managed the room based on edit and delete functions.
  - When the room is vacated, admin can edit the room details and add that for new students.

- **Mange students**
  - Admin can manage the student profile based on Serial Number, Name, Registration number, contact information, room number, bed, and day of staying.
  - The admin can monitor all the information of the students from the day they start to get the room.

- **User log**
  - Admin able to check login activity of a student based on Student Id, Email, and time.

- **Admin Dashboard**
  - Using the dashboard admin can check the student detail, room details, and total course in one view. This is easing the task of the administrator, and they can do the decision based on the data.

- **Add Course**
  - Admins can add a course for the students. Using that course, they can classify the student's room. The same course students can be grouped in one room according to the details.

- **Mange course**
  - Admin can manage the courses based on course code, and course name.
  - Using edit and delete button admin can manage the course details.
- **Student Registration**
  - The admin can register a students based on the student request. In addition, the admin can

```
A[Admin] --> B{Room Management}
B --> C{Add Room}
B --> D{Manage Room}
A --> E{Student Management}
E --> F{Add Student}
E --> G{Manage Student}
G --> H{Assign Room}
E --> I{Student Registration}
A --> J{Monitoring}
J --> K{User Log}
J --> L{Manage Courses}
L --> M{Add Course}
```

```
L --> N{Manage Course}
```

The table above shows the graph view of the add operation. The all the functions and sub-functions are described above.

b. Student

- **Change Password**
  - The student can change the password based on their wish.

- **Book Hostel**
  - Students can book a hostel room based on the availability.

- **Room Details**
  - Students can check the room related details. Students can get the printed data based on the room they have booked.

- **Check Profile**
  - Student can check and update their profile according to their wish.

```
A[Student] --> B{Registration}
B --> C{Profile Management}
C --> D{Check Profile}
C --> E{Update Profile}
A --> F{Room Information}
F --> G{Check Available Rooms}
F --> H{Filter and Search}
A --> I{Book Hostel}
I --> J{Select Room}
I --> K{Choose Dates}
I --> L{Add Roommates}
A --> M{Change Password}
M --> N{Enter Current Password}
M --> O{Enter New Password}
M --> P{Confirm New Password}
```

The table above shows the graph view of the add operation. The all the functions and sub-functions are described above.

2.2 Non-functional Requirements

- **Availability**
  - This website should be available every day in 24 hours' time.
  - User must provide a username and a password to for avoiding unauthorized access into this website.

    - The application will authenticate its user. Only for verified users.
- **Efficiency**
    - This web can manage this process consuming a minimum amount of time as well as a minimum amount of manpower.

- **Accuracy and reliability**
    - This web can produce accurate results using provided data which are highly reliable.

- **Usability**
    - This website provides interfaces to each student to fulfil their tasks in a well-managed and clear manner.

Chapter 03

3. System Analysis and Design

A comprehensive study of the many tasks involved in a system and their interactions within and beyond the system is known as system analysis. An excellent analytical approach should include the processes for comprehending the problem and the foundation for the solution.

3.1 Existing System

The critical issue is, "What all flaws exist in the current system?" What steps must be taken to resolve the issue? When a user or management initiates a study of the software utilizing an existing system, analysis begins.

The Existing system of the universities contains the following system,

- Paper-based hostel management system
- Random allocation of the room for the students.
- Improper decision-making system.

This manual system needs much more time to maintain student-based problems. The identified cons of the existing system are as below,

- No user-friendly environment
- Prone to error mostly
- Maintenance cost high
- Slow access
- Poor student monitoring
- Poor room Maintenance
- Too primitive and
- Insecure

The concerns described above were discovered through monitoring and must be resolved as soon as possible with a proper solution.

3.2 Proposed System

We have designed our web system to allow for the most efficient addition, deletion, amending, and monitoring of student, course, and room information. As a result, the overhead of sub-warden has decreased. Another feature of the system is the ease with which the information on decision support.

The following advantages can be found in our proposed system when compared with the existing system,

- Very less error
- Quick decision support
- Easily add, delete, update, and view student, course, and information
- Eliminate paper cost.
- Provides better technical-related support.
- Reliability
- Provide authenticity facility.
- It is secure compared with the paper-based system.

3.3 System Configuration

The following hardware and software requirements are needed to develop the proposed system.

1. Hardware Configuration

Table 1: hardware Requirements

Hardware	Capacity
RAM	1GB
Hard Disc	10GB
Processor	250 MHz

2. Software Configuration

Table 2: Software Requirements

Software	Type
Operating System	Windows 8 or more
Language	HTML, CSS, JavaScript, and PHP
Text Editor or IDE	vscode or Notepad++ or NetBeans IDE
Server	Apache, XAMPP
Browser	Chrome or any

3.4 Sub-system Analysis

1. Economic Analysis

The Proposed system is a web-based application; all students' information, course, and room will be stored in a database table which is available in the hostel database; hence there is no

need for a paper-based system. It will eliminate the paper-based print cost. In addition to that, room details will be displayed via needy input. Hence the system will overcome cost, time, and additional work. The economy for this project is relatively meagre compared with the existing system. Moreover, it is a good idea to migrate this system rather than depend on the primitive existing system.

2. Technical Analysis

The system will be built with PDO php programming. When we consider the feature of PHP programming, we can list out the following features,

- **Simple**
  - It is straightforward to learn and easy to use compared with other languages.
- **Platform Independent**
  - It supports any platform, such as Windows, Linux, and iOS. So, it is easy to migrate from one platform to another platform easily.
- **Object-Oriented**
  - It supports Encapsulation, Inheritance, Polymorphism, Abstraction, Interface, classes, and objects. The source codes we can create using this language are easily reusable with function calls.
- **Robust**
  - The PHP program easily handles an unexpected crash of the program and will help avoid the program's termination while running. In addition, it handles memory management more neatly than other programming languages. Hence, it is a more robust and powerful technical language.
- **Secure**
  - Compared to other languages, PHP language supports encryption and decryption techniques. Most gadgets connect to the internet via mobile data and WiFi cellular networks. As a result, it may be vulnerable to attacks. PHP provides high security and evades all those security-related issues. In addition, it automatically prone to attacks like SQL Injection.

3. Legal Analysis

When considering the HMS project, all the details of students, Courses, and Rooms are belonging to university. The admin or the system operator can access the all the information. There are no legal issues found in this project.

4. End-User Analysis

The system's end-user (Students) does not need as much programming or filter knowledge as al least needed in the spreadsheet system. The admin or operator must know how to insert, delete, update, and view student, course, and room details. The intended system will hide what is going on behind the curtain, and the system user can efficiently work in a user-friendly environment.

5. Input Analysis

The proposed system has the following functionalities and input for each functionality as follows,

Table 3:Input Analysis Table

Functionality	Input
Add Student	Room no, bed, duration, course, firstname, lastname, gender, etc
Manage Student	Student Id
Delete Student	Student Id
Update Student	Student Id,
Add Room	Room no, fees, bed
Delete Room	Room id
Update Room	Room id
Delete Course	Course id
Update Course	Course id
User log	Serial no
Search student	Student id
Search course	Course id
Search Room	Room id

3.5 Database

Table 4: Database table and attributes

Table	Attributes
Admin	id (INT, PRIMARY KEY): Unique identifier for each user.
	username (VARCHAR(255)): Unique username for login.
	Password (VARCHAR(255)): Hashed password for secure login
	role (ENUM): User's role (e.g., "student", "admin").
Students	id (INT, FOREIGN KEY REFERENCES Users(id)): Student's ID (linked to Users table).
	name (VARCHAR(255)): Student's full name
	contact_info (VARCHAR(255)): Student's phone number and/or email address
	course (VARCHAR(255)): Student's enrolled course.
	emergency_contact (VARCHAR(255)): Emergency contact details
Rooms	id (INT, PRIMARY KEY): Unique identifier for each room.
	type (ENUM): Room type (e.g., "single", "double").
	capacity (INT): Number of occupants allowed in the room.

	facilities (TEXT): JSON or serialized list of room amenities
	price (DECIMAL(10,2)): Room price per night.
Logs	id (INT, PRIMARY KEY): Unique identifier for each log entry.
	user_id (INT, FOREIGN KEY REFERENCES Users(id)): User who performed the action.
	action (VARCHAR(255)): Description of the user's action (e.g., "booked room", "updated profile").
	timestamp (DATETIME): Date and time of the action.

3.6 Entities

- **Student:**
  - Attributes: studentID (PK), name, contact_number, email, course, emergency_contact, profile_picture, password (hashed)
- **Room:**
  - Attributes: roomID (PK), room_type, capacity, facilities, price, is_available
- **Admin:**
  - Attributes: adminID (PK), username, password (hashed)
- **Course:**
  - Attributes: courseID (PK), name, department, duration
- **Relationships:**
  - One Student can have many Bookings:
    - Cardinality: 1:N
    - Foreign Key: studentID (Booking) references studentID (Student)
  - One Room can have many Bookings:
    - Cardinality: 1:N
    - Foreign Key: roomID (Booking) references roomID (Room)
  - One Admin can create many Bookings:
    - Cardinality: 1:N
    - Foreign Key: adminID (Booking) references adminID (Admin)
  - One Student belongs to one Course:
    - Cardinality: N:1
    - Foreign Key: course (Student) references courseID (Course)

3.7 SYSTEM DESIGN

Systems design defines system aspects such as modularity, infrastructure, sub-systems, interconnections, and data for a system based on the requirements stated. It is the process of identifying, creating, and designing systems to meet a firm's unique objectives and requirements.

3.8 Design Diagram

A diagram is a graphic representation of relationships between network elements, people, and activities. In order to build a Hostel Management System, it is essential to construct the design diagram to ensure the path is correct. There are several design diagrams available. In this project, we will design with the help of a Flowchart and a class diagram.

1. Flowchart

A flowchart is a graphical representation of a process, structure, or computer program. They are widely employed in various disciplines to track, examine, plan, enhance, and convey frequently critical procedures in clear, simple diagrams—the Terminator, Process, Decision, Documentation, Data, and Flow Arrow are some of the needed aspects in flowcharts. Flowcharts usually help us to give ideas to solve the problem step by step. In addition, it increases efficiency and helps create clear documentation. The following flowchart shows how the Hostel Management System works.

From the flowchart we are able to identify that admin can do variety of the operations including add, manage students, course, and rooms. The payment details can be checked by the students based on the list provided.

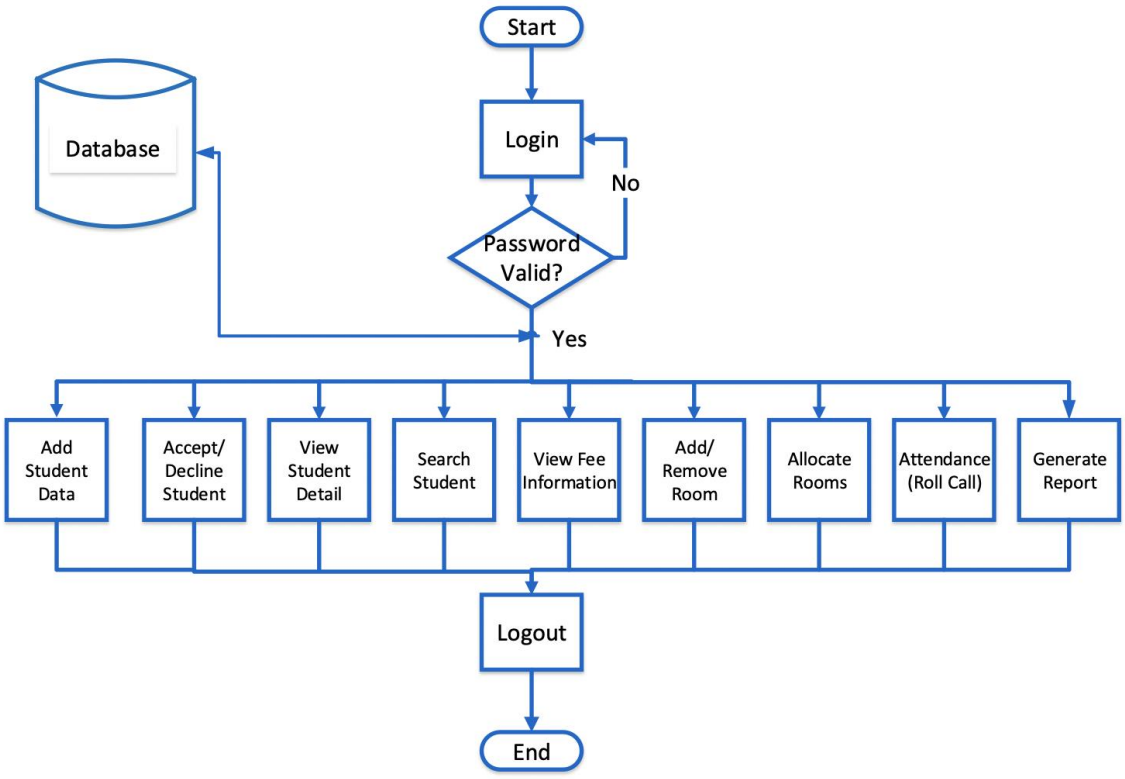


Figure 1: Admin Process

2. Activity Diagram

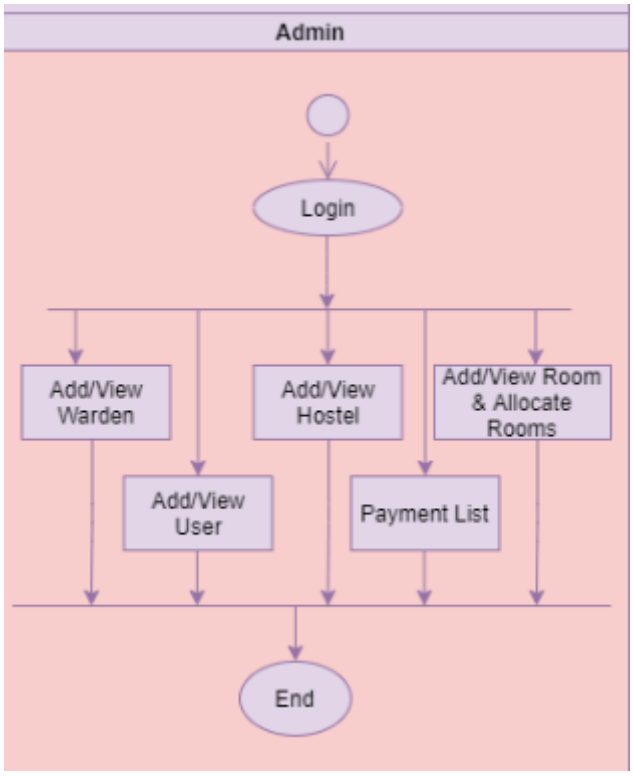


Figure 2: Activity diagram of Admin process

Activity diagrams in UML are a leading subset of behavioural diagrams. They display the functionalities of various activities and flows in management processes and software systems. The flow in the activity diagram can be sequential, branched, or concurrent. The following activity diagram shows how HMS is working,

According to the image, we can identify that admin can create another admin or warden. Also, they can do the create, read, update, and delete operations of student, rooms, hostel, and payment details.

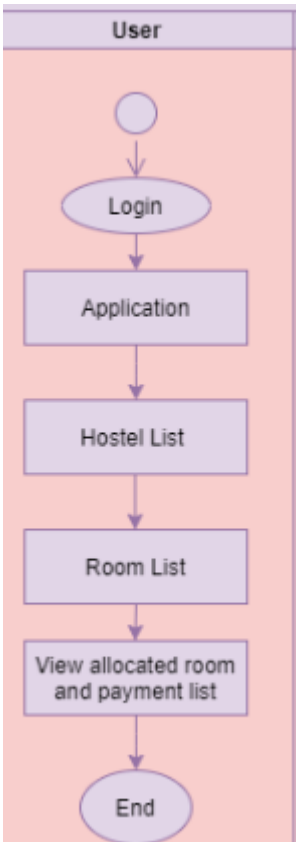


Figure 3: Activity Diagram of Student

The students can get the hostel list is booked, room list, payment list, and other details with the link provided on the website.

Chapter 04

4. Methodology

In this project as the methodology, we have decided to use the Agile methodology because of the flexibility, adaptability, faster delivery, transparency as well as it is flexible for changing requirements. Even though we have identified the requirements in the early stage, there are several possibilities of changes later in the process due to technological gap. Agile focuses on increasing client engagement by being both iterative and incremental and it will deliver a working software within short time. Because of that as our perspective agile would be the most suitable solution for this among of other methodologies.

Agile SDLC breaks down the product into small incremental builds and these are provided into iterations. Therefore, agile is a combination of a both iterative and incremental process models. It consists,

1. Planning
2. Requirement gathering and analysis
3. Design the requirements
4. Implementation/Construction/ iteration
5. Testing
6. Deployment
7. Maintenance

- **Planning** - Define the problem and scope of existing system. Overview the new system and determine its objectives. Confirm project feasibility and produce the project Schedule. A feasibility report for the entire project is created at the end of this phase.

- **Requirement gathering and analysis** – In this phase requirements will be identified, gathered, and analysed. Define the requirements and prototypes for new system. Evaluate the alternatives and prioritize the requirements.

- **Design the requirements** - After identifying the project, work with client to define requirements. Flow diagram or the high-level UML diagram diagrams are designed in this phase to show the work of new features and show how it can be applied into current existing system.

- **Implementation/Construction/ Iteration** – After defining the requirements in this phase work begins and members start working on their project. The aim of this phase is to deploy a working product within estimated time. The product will go into various stages of improvement and after each iteration there will be a new implemented feature/functionality. Priority will be given to the highest priority requirements and those are fulfilled in the early iterations. After every iteration requirement will be changed according to client's feedback and again iteration will continue until the all the features have been implement. This will move according to a detailed timeline. It allows for progress updates and checkpoints to

determine that the project requirements are being met. It ensures that the final product is launched in a timely manner.

- **Testing** - When the time client has already seen the project multiple times and additionally, integration testing and regression testing has been already done throughout the process and it will reduce the amount of testing that is required during this phase.

- **Deployment** – In this phase team issues a product for the user's environment.

- **Maintenance** – Now software is fully deployed and made available for customers. This action moves it into maintenance phase. In this phase software development team will provide ongoing support and keep the system running smoothly and resolve if there is a new bug.

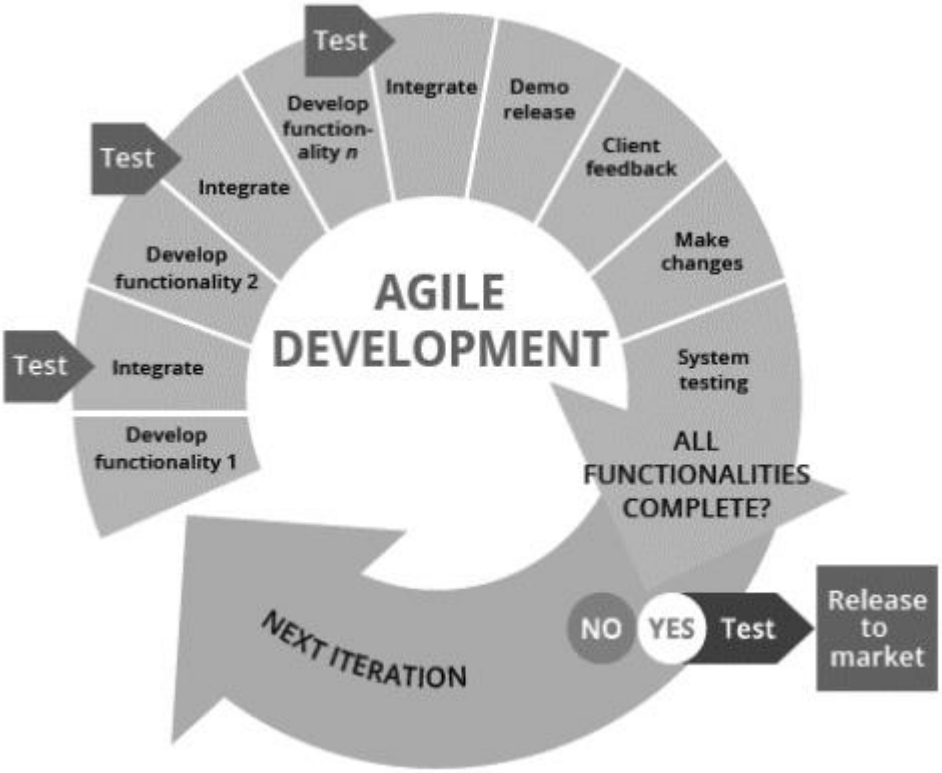


Figure 4: Agile Methodology

In Agile, after each iteration we can build a working software to deliver to the customer. Apart from that after every iteration, using customer's feedback can deliver a more productive and high-quality software. In this method priority will be given to the highest priority requirements and they will be implemented in the early iterations, and this will be more efficient. Requirements are accepted even later in the development and because of that it would be more user friendly.

The Agile development approach puts a strong emphasis on people and collaboration, which

renders the development team with an opportunity to work closely with their clients and can understand their real requirements and needs. Besides, the projects are delivered to the clients in the form of multiple sprints, which brings transparency to the process. It also enables the team to determine if both parties are on the same page and if not, allow them to make the necessary required changes before proceeding further of the project.

Apart from that we are going to use Object-Oriented Design of PHP which makes use of several different models to communicate the design of the system. Because they define the key components and requirements of the system, we can use these models to produce the software product. Also, we are going to give a priority to human interaction concepts when we are going to design interface design because it will help to website to more successful, safe, useful, functional, user-friendly, and more pleasurable for the user.

4.1 Software and Functionalities

The projects utilize the HTML, CSS, and JavaScript as a front-end and the PHP as a back end to build the system.

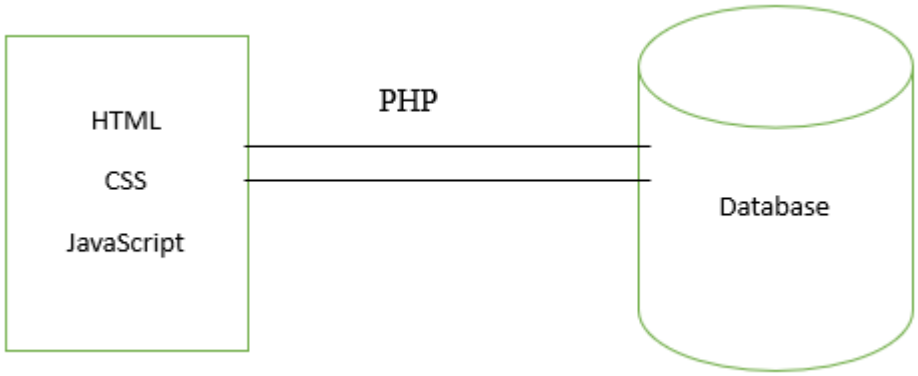


Figure 5: Working procedure of the system.

Here the PHP works as a mediator in between front-end and the backend database. All the connections, CRUD is done by PHP. All the request from the front-end pass to the database as "httpreq" and user get output as "httpres".

- HTML – Used to create a static web page.
- CSS – Design the client-side design.
- JavaScript – Client-side rendering and animation.
- PHP – Used to create a dynamic web page. It is a single thread programming language. The response will be served to the user in first come first serve basis.
- **Database**
  A database is a collection of coherent data which represents the mini world. The database is an essential part of any application development, which is used to maintain and handle the data very efficiently and effectively. Almost all applications, like

desktop, mobile, and web-based, use its database for data handling. The following advantages can be achieved using databases.

1. **High Security**
   Storing the information in a particular physical location instead of a logical location offers a very high level of protection for the data. The information can be safeguarded, and unauthorized parties cannot delete or remove it.

2. **Reduce data Redundancy**
   The database offers the capabilities necessary to prevent redundant data from being stored. Having the database in a normalized form will help accomplish this goal.

3. **Data backup and Recovery**
   It is possible to back up and restore the data in the database. Data backups can be stored on a variety of different media. Even if the data is lost, it can be retrieved in various ways.

4. **Data Sharing**
   The data inside the database can be easily shared with others rather than any other methods.

5. **Query Facilities**
   The data can be easily handled with SQL query facilities.

- **DBMS**
  The standard for Database Management System is used to manipulate the data in the database. The database uses Standard Query Language (SQL) to handle the database. The SQL handles the insert, delete, update, view, and search queries. The following tables show some sample queries of the DBMS.

Table 5: Operation and Query

Operation	Example Query
Insert	Insert into student values(1, "X," 23, "first," "ICT");
View	Select * from student;
Update	Update student set(name, year) where id = 1;
Delete	Delete student where id = 1;
Search	Select student name like %A;

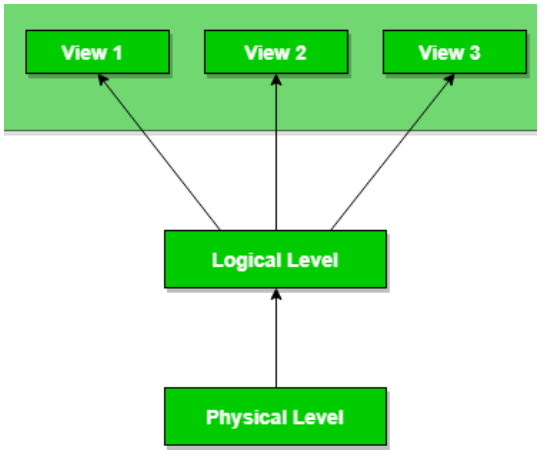


Figure 6: Database Views

- **Database connectivity**
  In order to work with the front-end and back-end, the database connectivity is vital part. Without having a proper database connection, we cannot work the HMS. To connect the database, the system utilize the following codes,

```
<?php
        $dbuser="root";
        $dbpass="";
        $host="localhost";
        $db="hostel";
        $mysqli =new mysqli($host,$dbuser, $dbpass, $db);
?>
```

  The object of mysqli is used for connecting the database and, the database name as “hostel”.

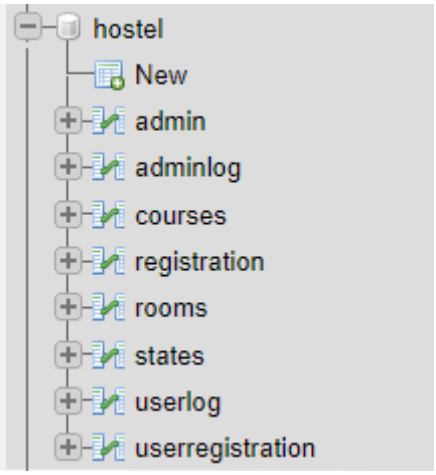


	Table	Action	Rows	Type	Collation	Size	Overhead
☐	admin	Browse Structure Search Insert Empty Drop	1	InnoDB	latin1_swedish_ci	16.0 KiB	-
☐	adminlog	Browse Structure Search Insert Empty Drop	0	InnoDB	latin1_swedish_ci	16.0 KiB	-
☐	courses	Browse Structure Search Insert Empty Drop	6	InnoDB	latin1_swedish_ci	16.0 KiB	-
☐	registration	Browse Structure Search Insert Empty Drop	1	InnoDB	latin1_swedish_ci	16.0 KiB	-
☐	rooms	Browse Structure Search Insert Empty Drop	5	InnoDB	latin1_swedish_ci	16.0 KiB	-
☐	states	Browse Structure Search Insert Empty Drop	9	MyISAM	latin1_swedish_ci	2.3 KiB	-
☐	userlog	Browse Structure Search Insert Empty Drop	6	InnoDB	latin1_swedish_ci	16.0 KiB	-
☐	userregistration	Browse Structure Search Insert Empty Drop	4	InnoDB	latin1_swedish_ci	16.0 KiB	-

Figure 7:Database and Table

The figure shows all eight tables of the hostel database as above.

Chapter 05

5. Results and Discussion

5.1 User Login / Default Home Page:

The User Login page is a crucial component of the hostel management system, providing safe access for authorized users. The website features streamlined navigation and a sidebar for simple access to relevant tasks, as well as user-friendly login and password entry fields.

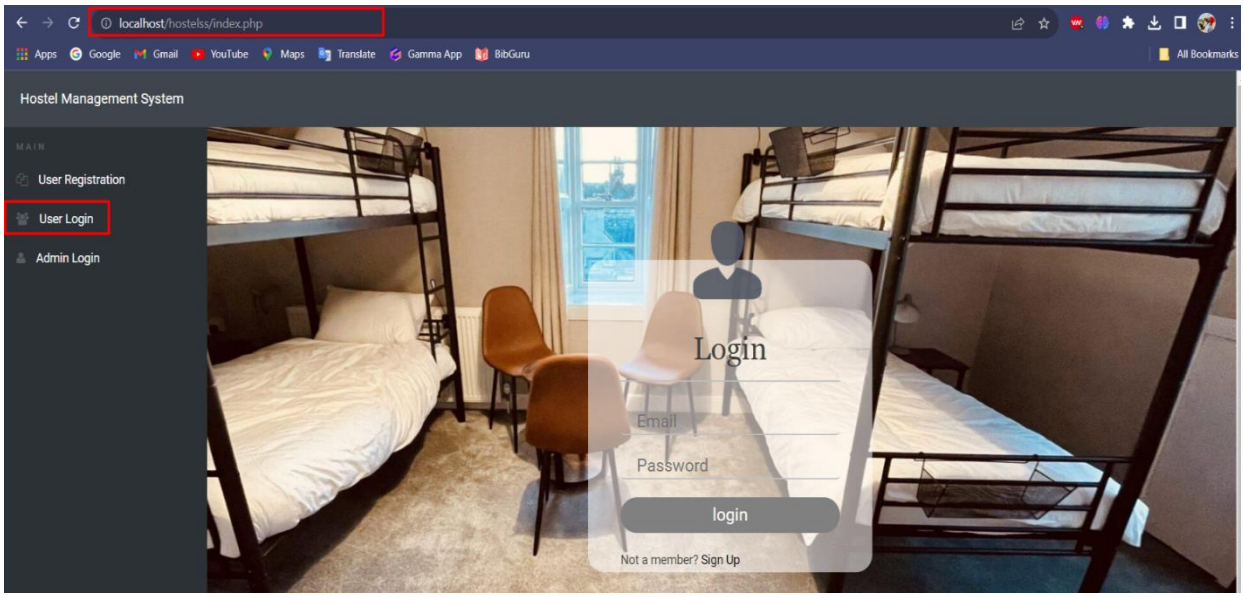


Figure 8: User Login

This is the default page to enter the project interface. When you type "**localhost/hostel"** in the browser it redirects to this **index.php**. In this page it has a Navbar consist of **"Hostel Management System".**

Its recognizable and dependable branding makes sure that consumers understand the system's background and goal. Users can utilize the navigation bar as a point of reference when navigating between the different areas of the website.

Also, a sidebar which consist of **"User Registration", "User Login" and "Admin Login".**

The sidebar improves the navigation bar with its quick access to important items. The three main options are as follows:

- **User Registration**: New users can create an account on the website by using this function. It facilitates the initial steps for anyone wishing to use services associated to hostels.
- **User Login**: This section allows current users to securely log in and acts as the heart of the website. By inputting their login and password, users can access information and services that are specific to them.

- **Admin Login**: This section is dedicated to administrators who oversee and oversee hostel operations. In addition to handling duties related to user management, hostel amenities, and other areas, administrators have access to additional functionality.

In here we can see a login page to enter the Hostel Management System. In this login form you need to type your **Email** and **Password.** To verify the Email and Password the system will check this or verify this using the data in the database called **"HostelSS".**

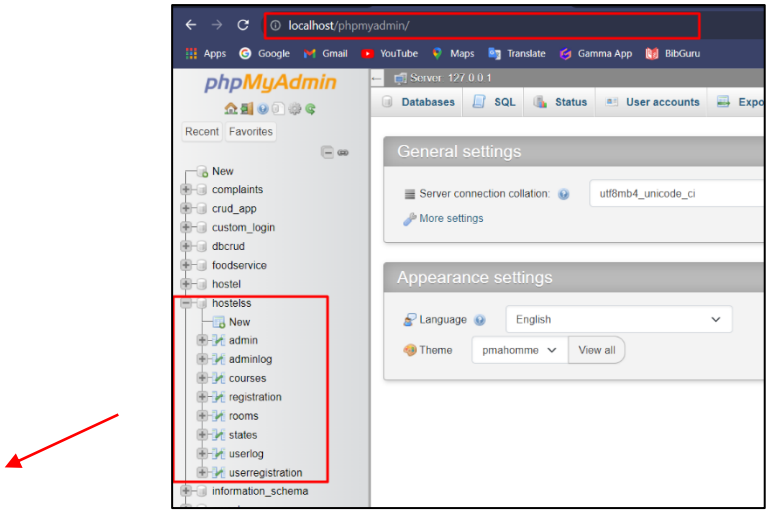


Figure 9: PhpMyAdmin

Here I have this database inside my localhost. So, from this table I have a table called **"userregistration".** From this table we have user details.

Figure 10: User profile

Here I have a dummy data to verify my user login. Using this I can verify and authenticate to user home page. If the **Email** and **Password** which you enter matches with the data in the table matches only it'll log in to the home page.

5.2 User Registration

When you don't have a user account to login this is the page that you'll needed to create an account to login. When you click on this User Registration, you'll redirected to User Registration page.

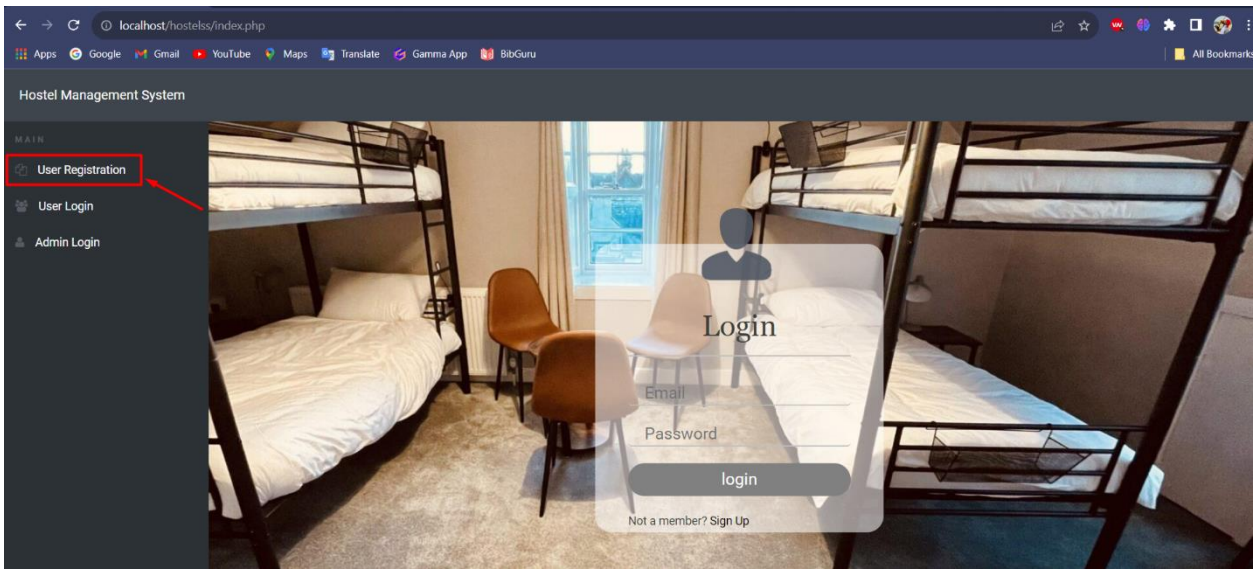


Figure 11: User Registration

The User Registration Page, which serves as a gateway for users to access and manage their accounts in the system, is a crucial component of the Hostel Management Website. With a focus on crucial data elements including the Registration Number, Email ID, Gender, First Name, Last Name, and Middle Name, in addition to the Contact Number, Password and Confirm Password.

The main goal of the User Registration Page is gathering the data needed from students of the hostel in order to create unique user accounts that will provide easy access to the Hostel Management System. This page guarantees reliable record-keeping inside the database and forms the basis for customized user experiences.

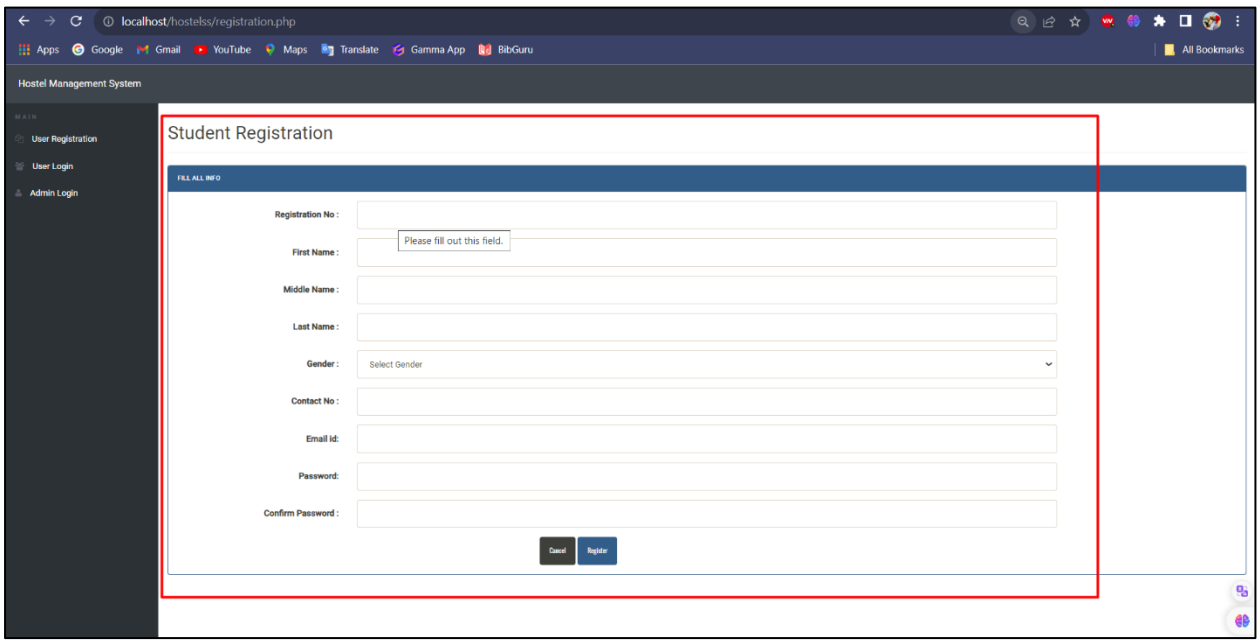


Figure 12: Student Registration

Every hostel resident is assigned a registration number, which functions as a distinct identity and facilitates tracking and administration.

Input fields of the User Registration Page

- **First Name, Last Name, Middle Name:** These are the fields used to properly document the resident's name so that their identity can be verified.
- **Gender:** Allows renters to specify their gender in order to receive personalized communications and be assigned to a hostel.
- **Contact Number:** Crucial for correspondence and emergency contact in the event of a crisis.
- **Email ID:** Unique email addresses serve as login credentials and a channel for interaction with the hostel management.
- **Password, Confirm Password:** By requiring a combination of alphanumeric, special, and alphabetic characters, it adds an additional layer of security to user accounts.

After successfully registering, users receive a confirmation message. And also, the data that he entered will be pass to the database table called userregistration. In the event that any issues or details are missing, helpful error messages assist users in making the required corrections.

5.3 User Dashboard

After login in the login page if the Email and Password are correct, you'll redirected to the below User Dashboard Page

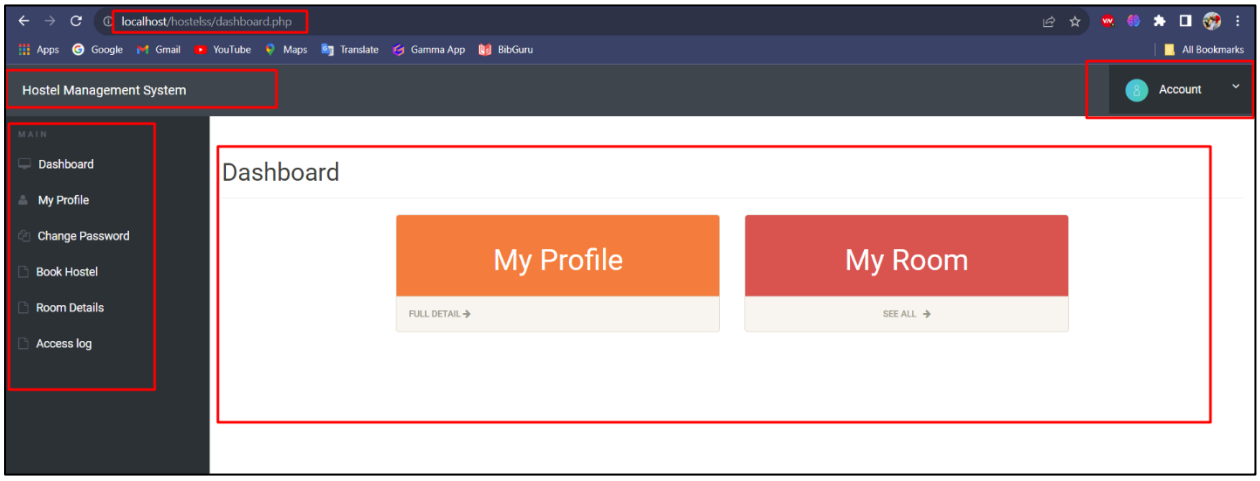


Figure 13: User Dashboard

If the login failed, you'll get notification like below.

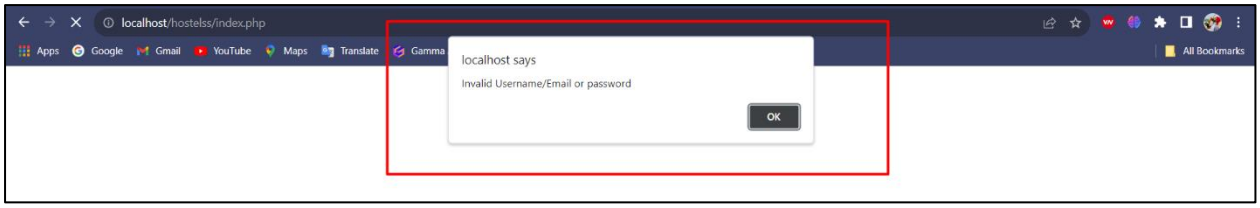
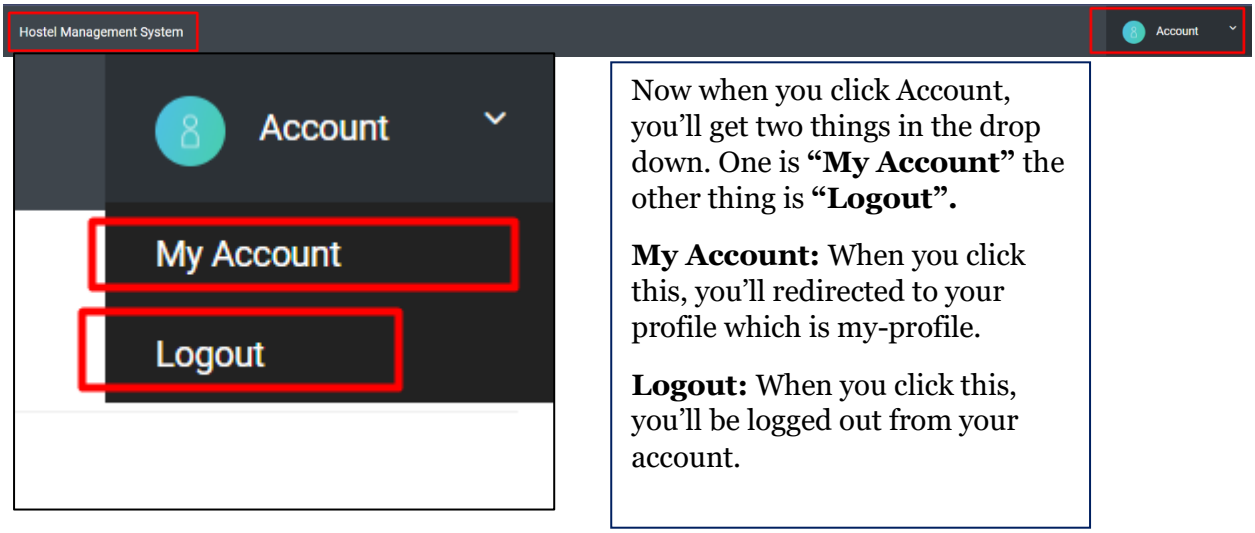


Figure 14: Authentication error

Now when you logged in and in the User Dashboard. You'll get a navbar with our Project Name and Side bar with the contents.

In the Navbar you'll see

Hostel Management System and one drop down type list with an icon. Here when you click on that Account

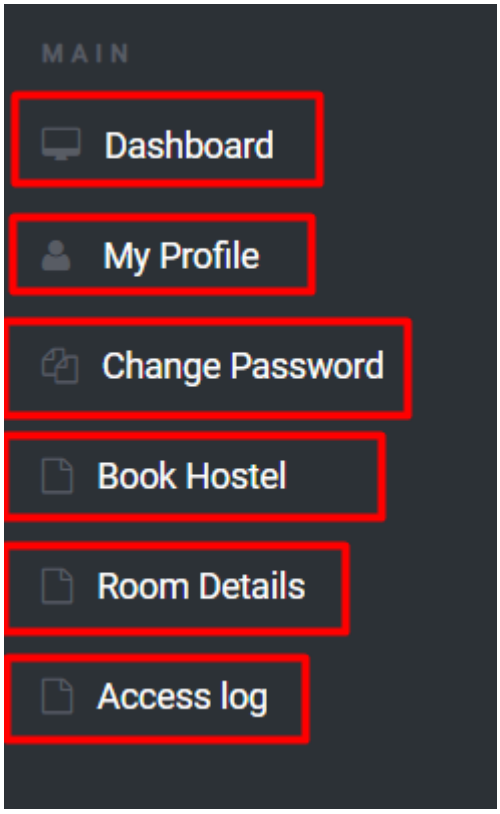


Now when you click Account, you'll get two things in the drop down. One is **"My Account"** the other thing is **"Logout".**

My Account: When you click this, you'll redirected to your profile which is my-profile.

Logout: When you click this, you'll be logged out from your account.

In the Side Bar you'll see



Dashboard: Dashboard is the default page after the login.

My Profile: My profile is the page which contains user details.

Change Password: This option gives you an option to change your password.

Book Hostel: Here this option gives an option to book hostel for the user.

Room Details: This page will give you the rooms details

Access log: This option will tell your login date time and entries.

In the body of User Dashboard, you'll see the below things.

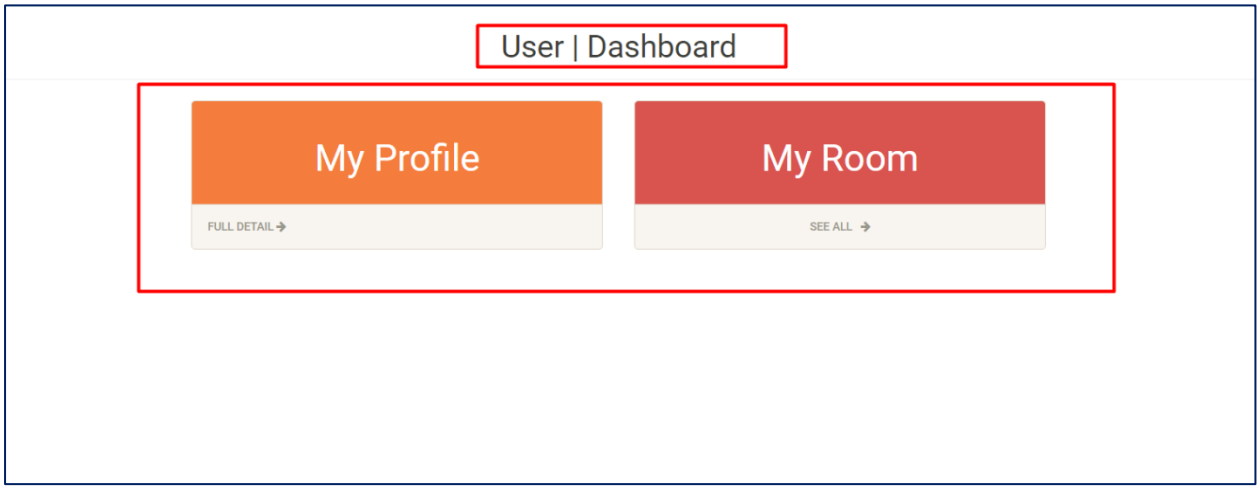


Figure 15: User Dashboard Interface

In here, when you click on My Profile it'll redirected to your profile page and when you click on My Room, you'll go to your room Details.

5.4 My Profile

When you click on **"My Profile"** in the sidebar of the dashboard

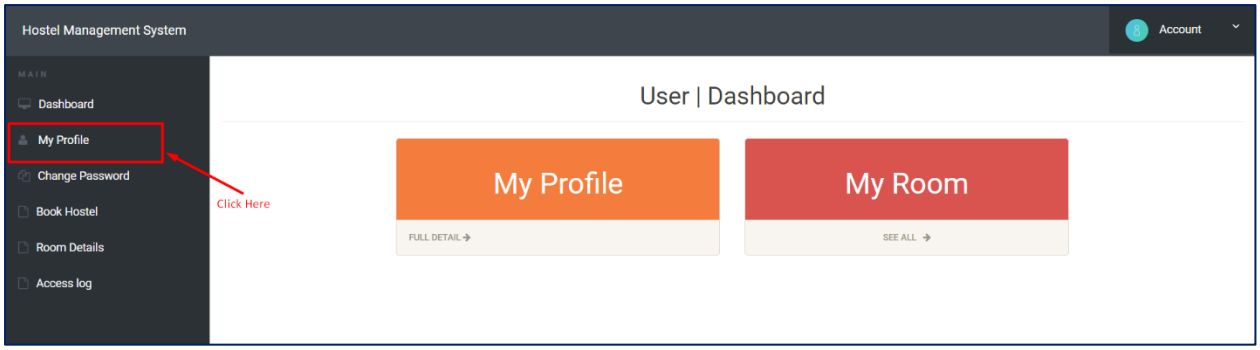


Figure 16: My Profile

You'll be redirected to your profile page which contains your personal information.

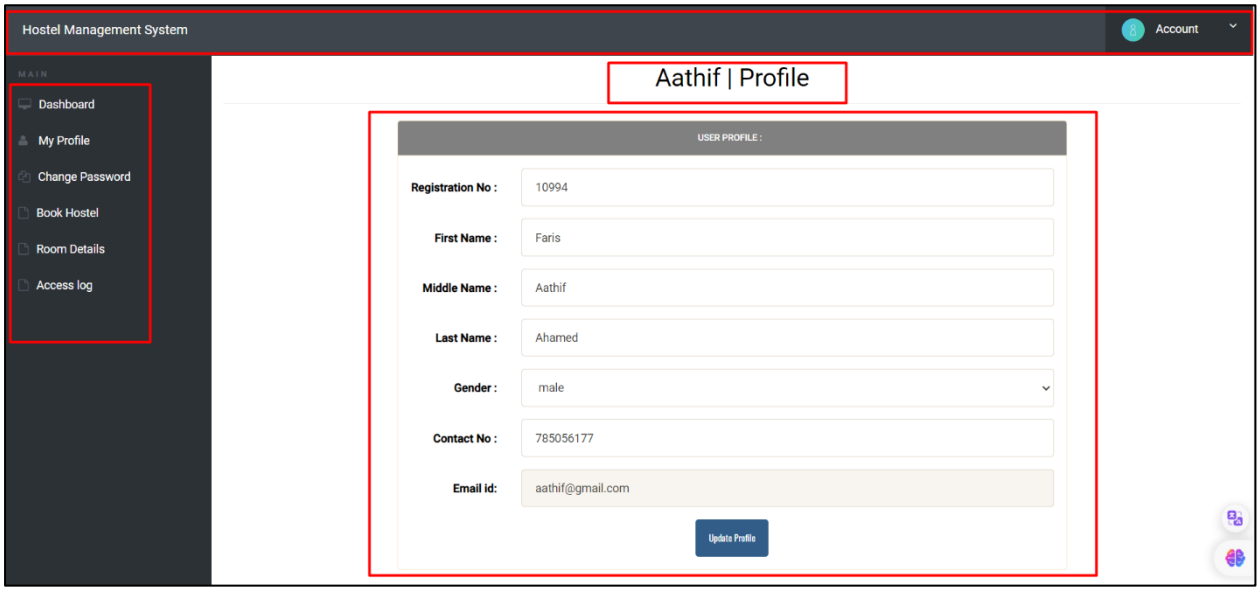


Figure 17: Profile Interface

Here also you'll get the same Navbar and Sidebar. But in the body of that My Profile you'll get your Personal Information in a form like structure with the specific field name.

Here you'll get your information from Database table. If you want to change or edit something, you can edit and click on the button **"Update Profile".** Once you click enter your changes will be altered in the Database **"userregistration"** table.

For Example,

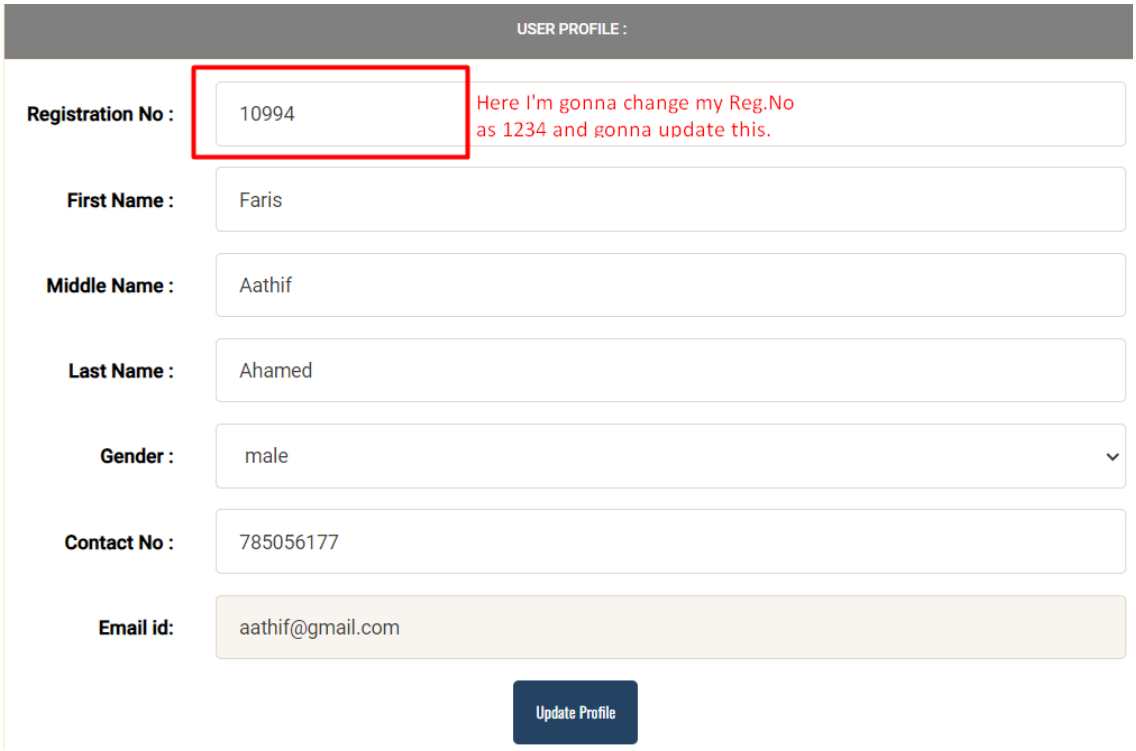


Figure 18: Update Profile

If I change and hit the update button, I'll get like below.

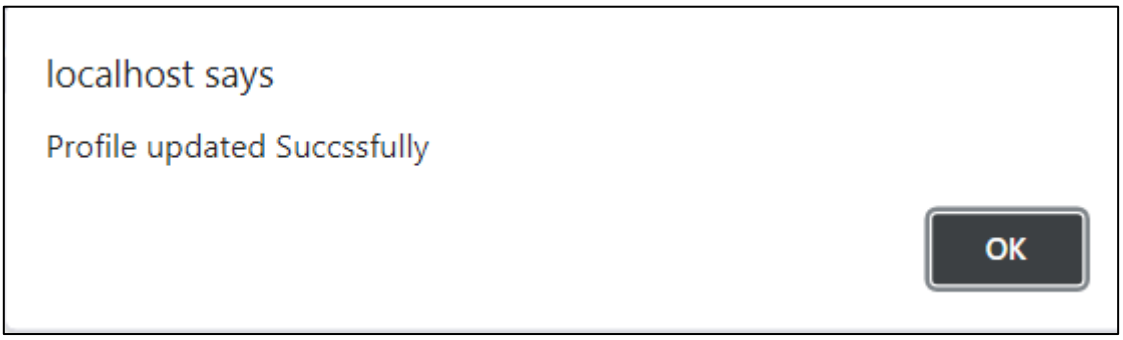


Figure 19: Update Profile

We'll get an alert saying that we have successfully updated our information. And also, it'll change in the database userregistration table

Figure 20: Database Update

5.5 Change Password

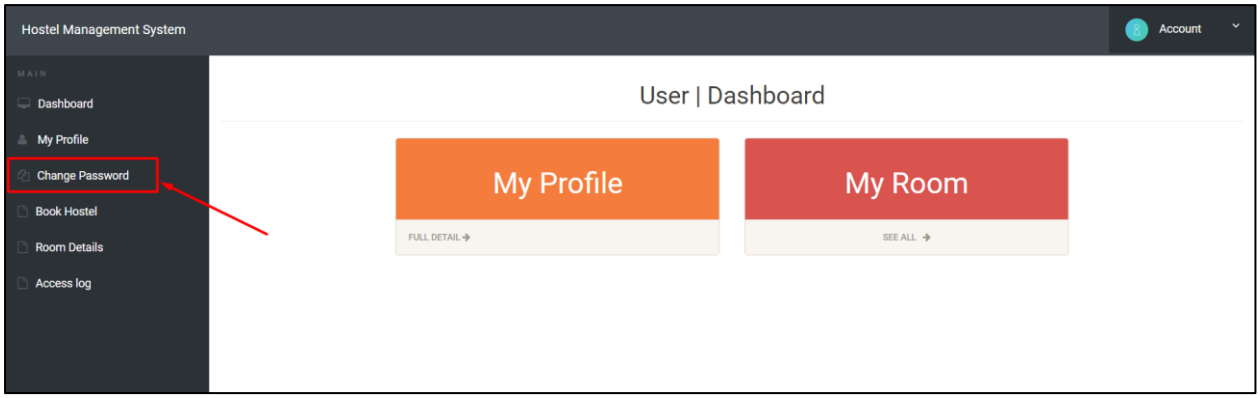


Figure 21: Change Password

When this Change Password is clicked it'll go to the change password page

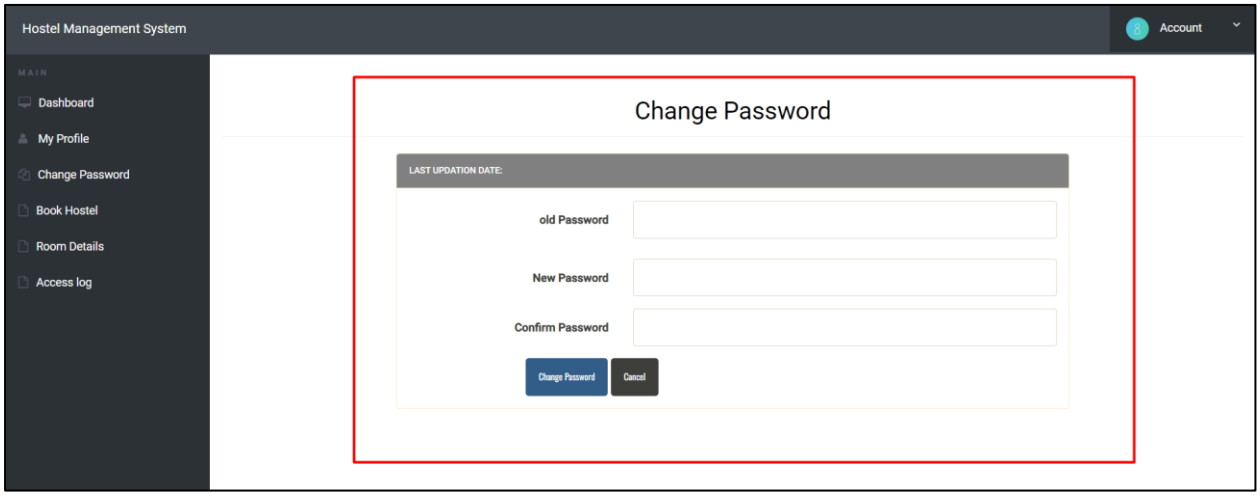


Figure 22: change Password Interface

In the Change Password Page, we'll get a form like structure to change our password. For that we need to give our **"old password", "New Password", and "Confirm Password".** If the Old password is correct only, we can change otherwise it'll show you the error.

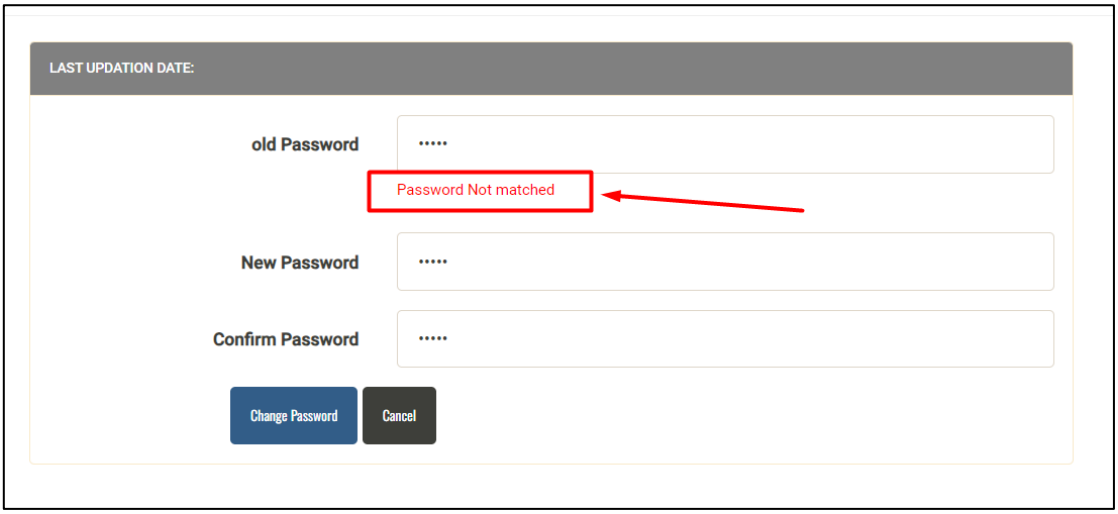


Figure 23: Password not matched.

When you give the old password wrongly, you'll get an alert saying that Password Not Matched. If the old password is correct, you'll get like below

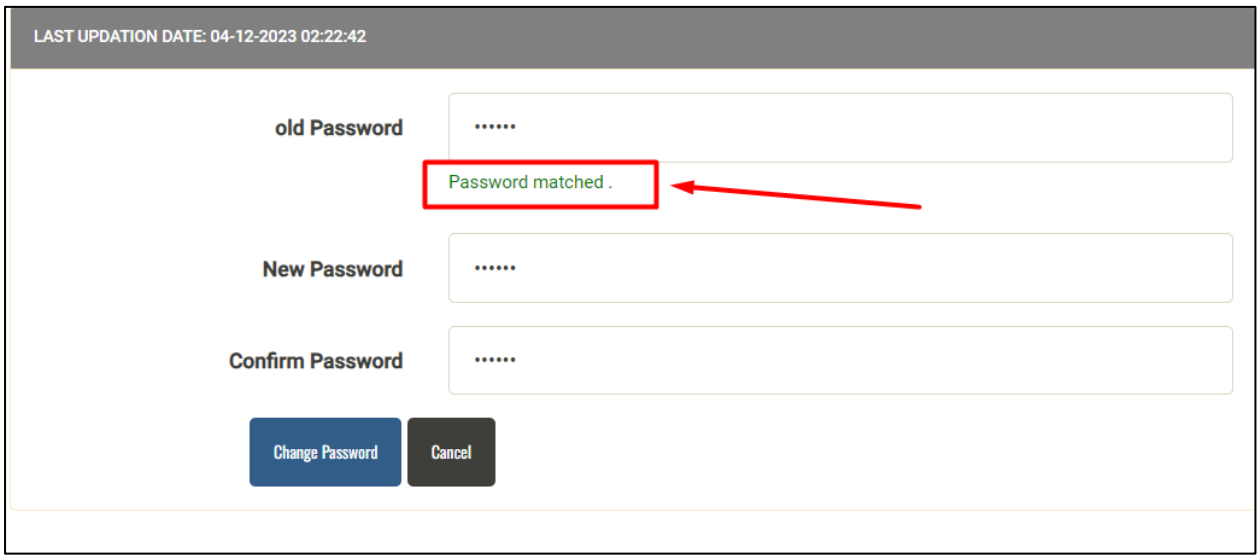


Figure 24: Password is Matched

If the old password that you've entered is correct, you'll get a green colour Password Matched message.

If you give everything correct and changes the password, you'll notify like this

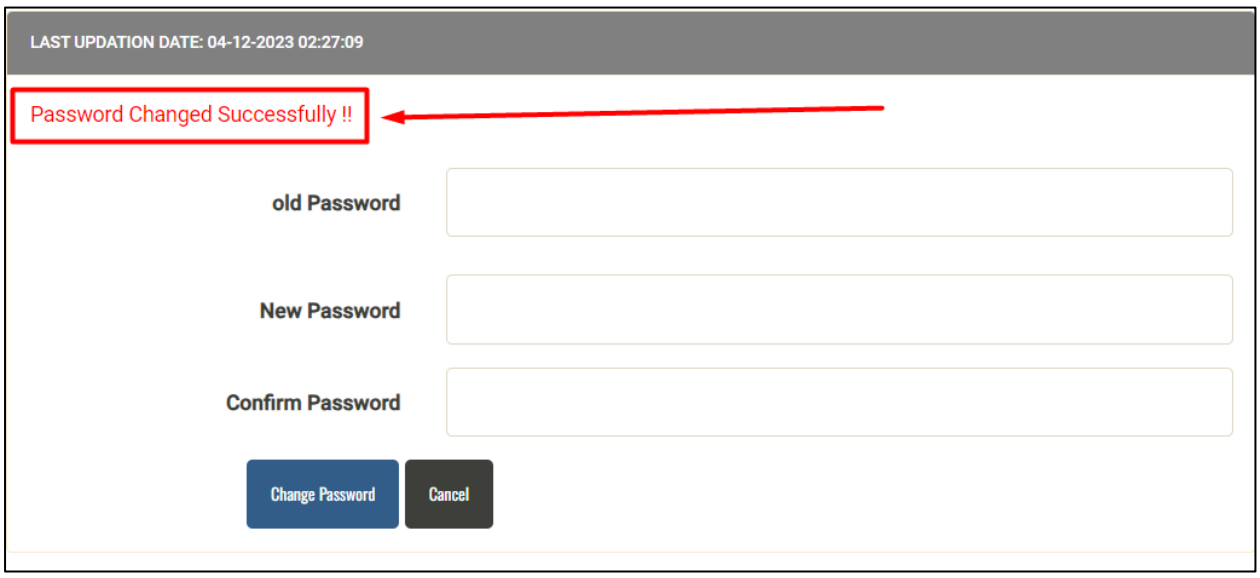


Figure 25: Password Change

And the changes will be updated in the database table also.

5.6 Book Hostel

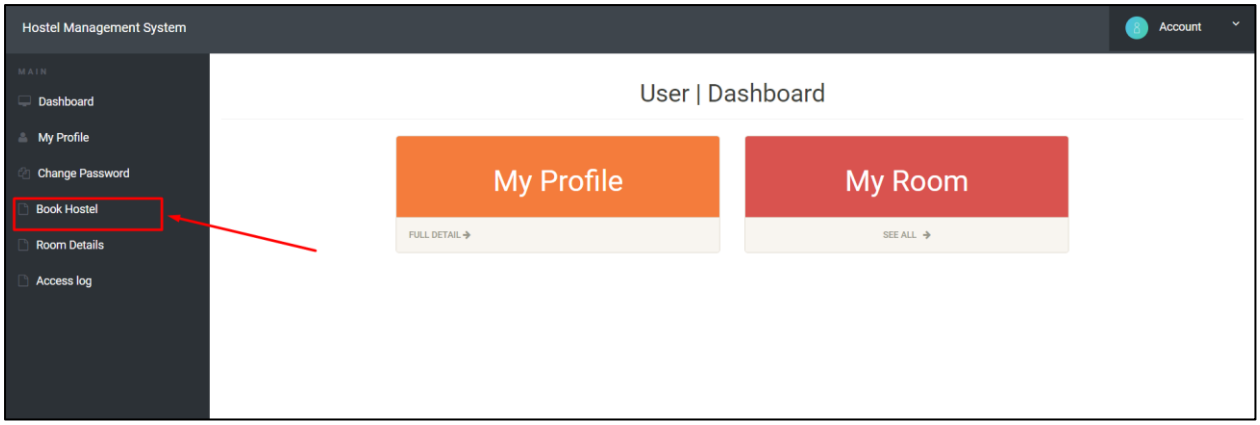


Figure 26: Book Hostel

When you click on the **"Book Hostel"** you'll go to the page to book a hostel room for you. If you didn't create anything it's fine but if you created already, you'll get a red color text warning message in the page itself.

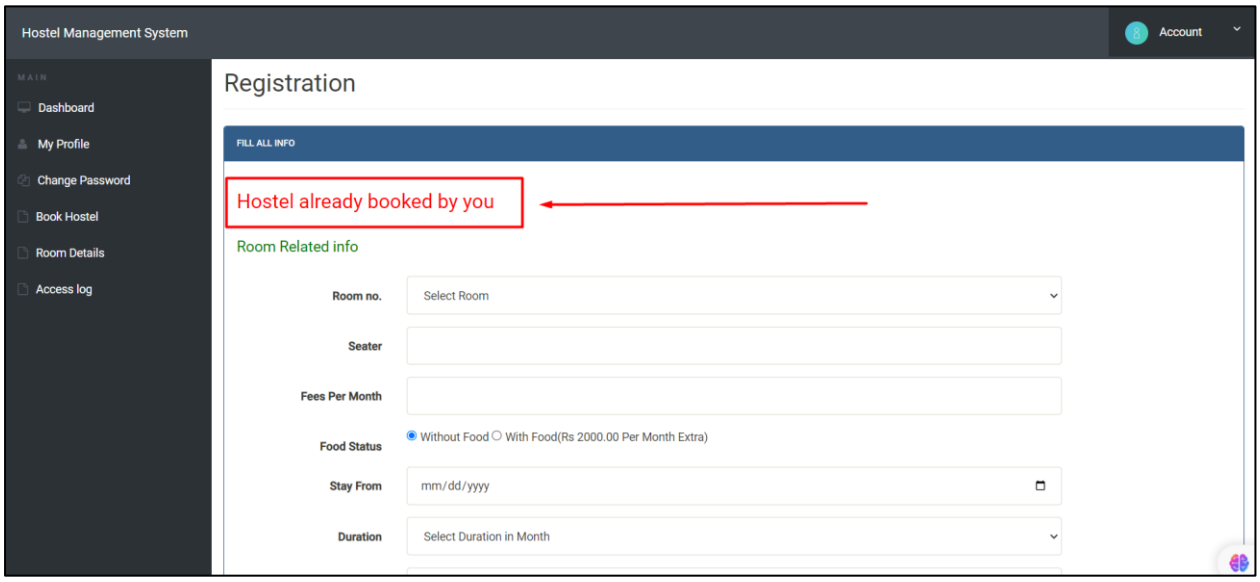


Figure 27: Already Booked

In this user case, I've already booked a room for them. If I take a new User, it'll show like below.

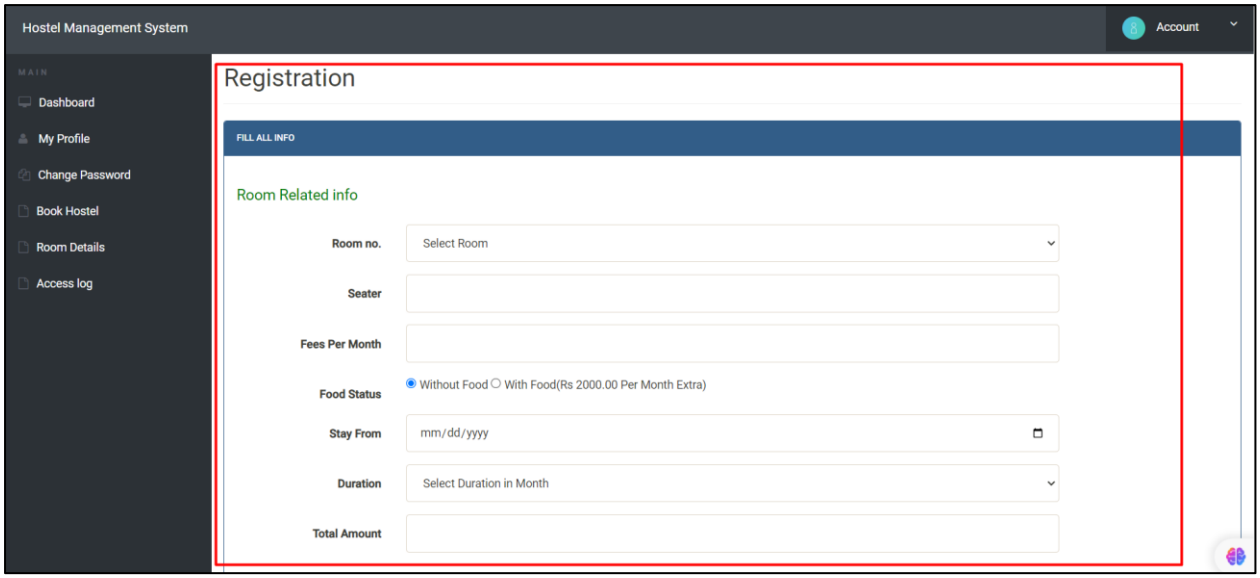


Figure 28: Room registration for new user

We won't see anything like **"Hostel Already Booked by You"** when the Hostel room is not booked. And you need to fill all the required information in this registration form and submit. It'll automatically updated in the database table called **"registration".**

In my case, I've applied for two users in the database registration table.

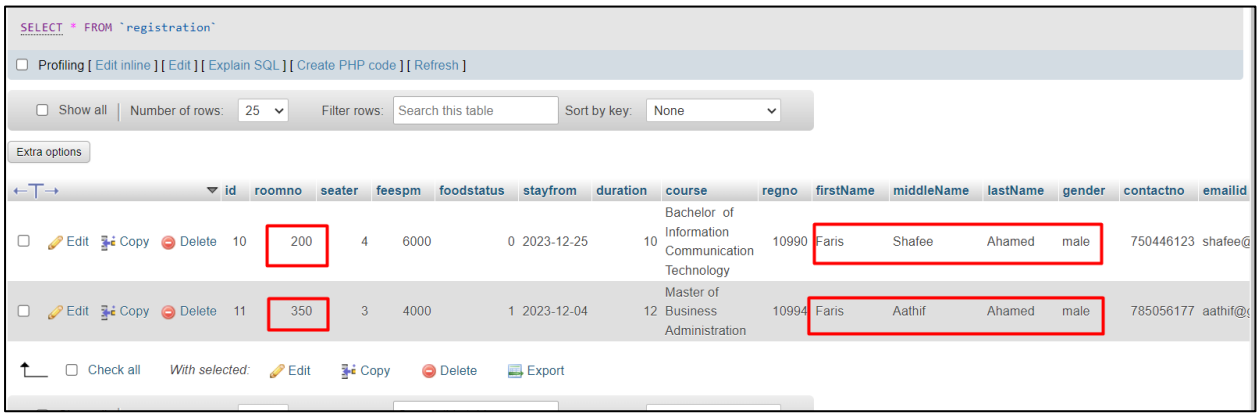


Figure 29: Database Change

5.7 Room Details

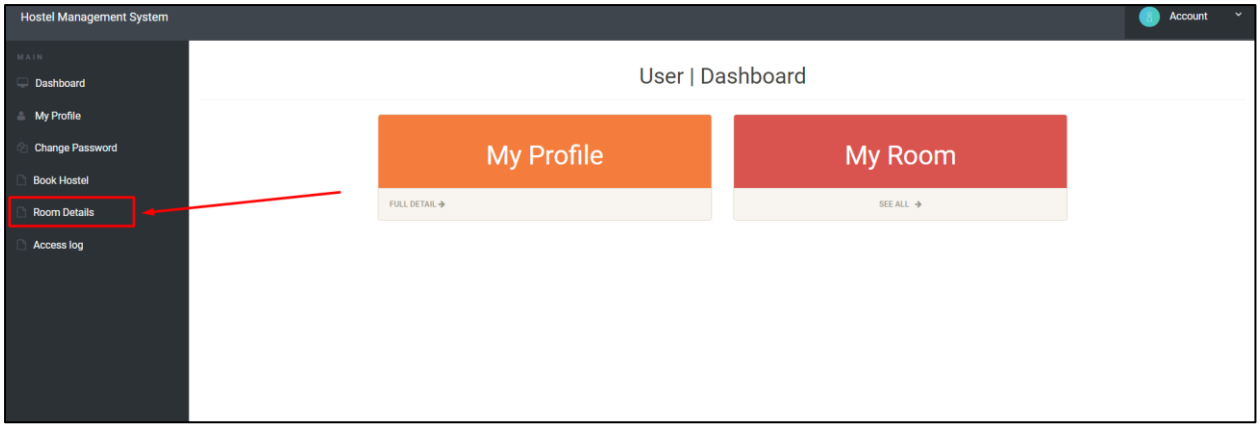


Figure 30: Room Details

When you click on Room Details, you'll be navigated to you room related info

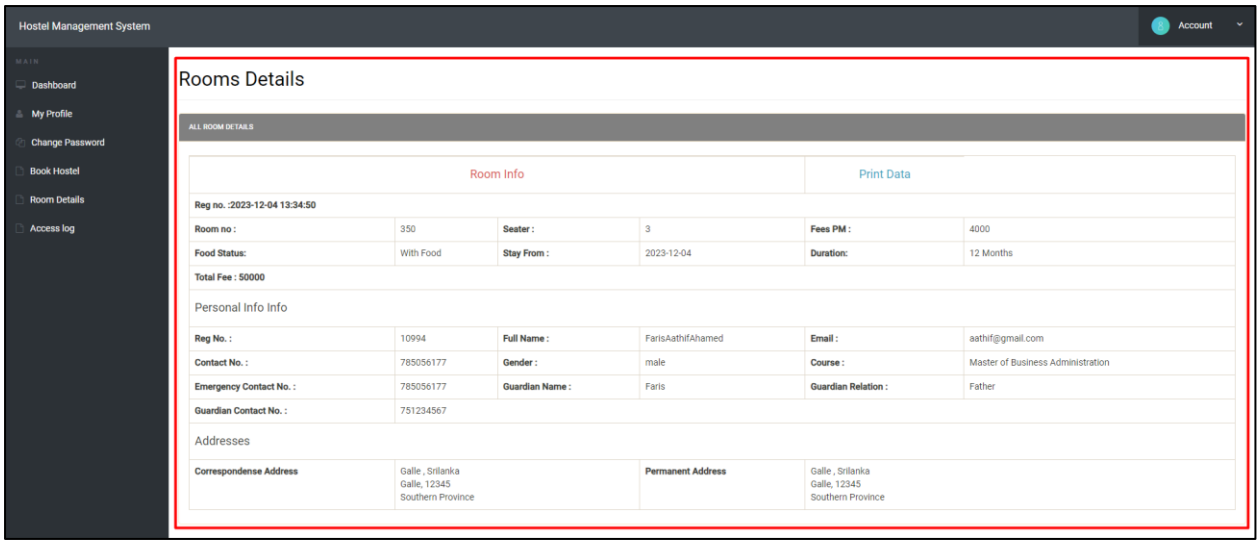


Figure 31: Room Detail interface

From this table you can view your information and if you want you can print the data using **"Print Data".**

5.8 Access Log

The Access Log Page is one of the most crucial tools for keeping an eye on user activity within a system or application. It is a record-keeping feature designed to capture crucial information such as times, dates, and user log-in details. Each user can see their own login details.

In the dashboard page user need to click on Access log.

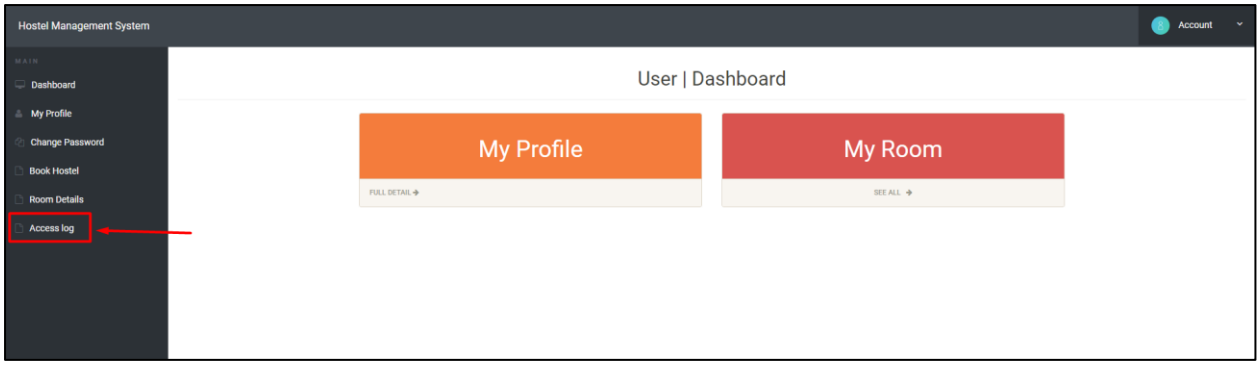


Figure 32: Access Log

When the user clicks on Activity log, it'll go to the Access Page

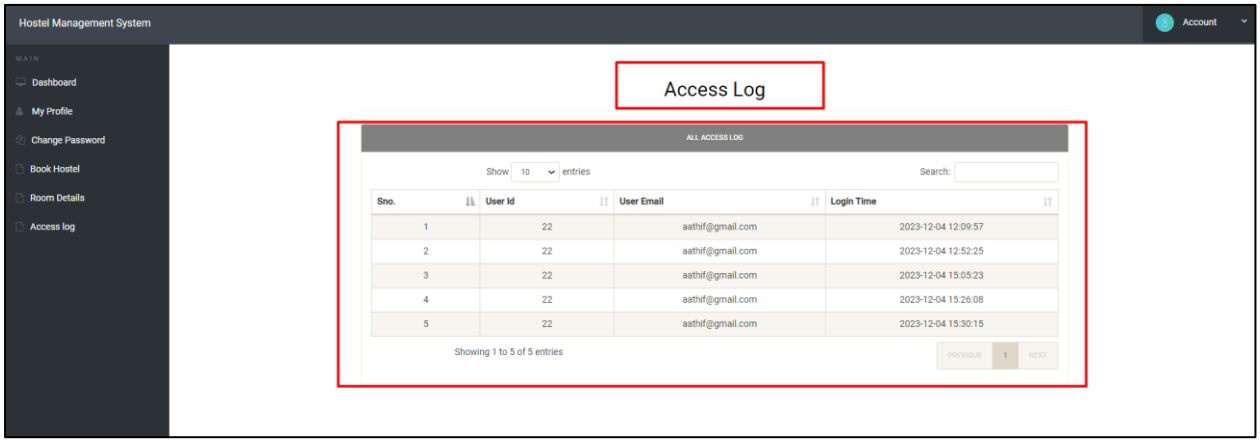


Figure 33: Access log Interface

By offering a clear record of user activity, an Access Log Page helps to promote user accountability. This functionality complies with industry regulations requiring compliance in a number of areas where user activity tracking and auditing is crucial for privacy and data security.

5.9 Admin Login

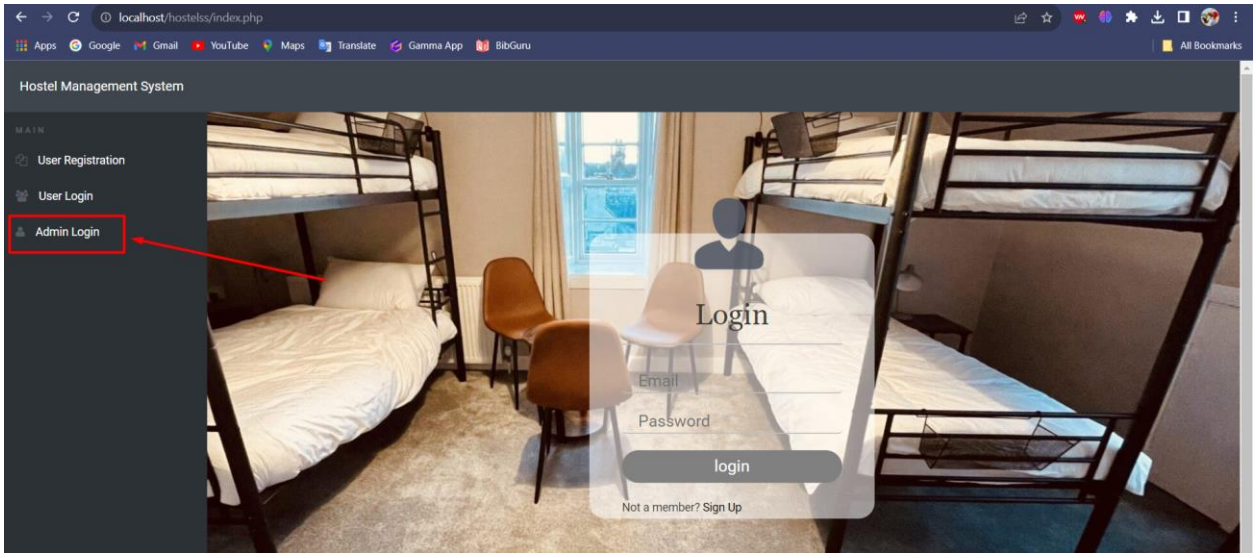
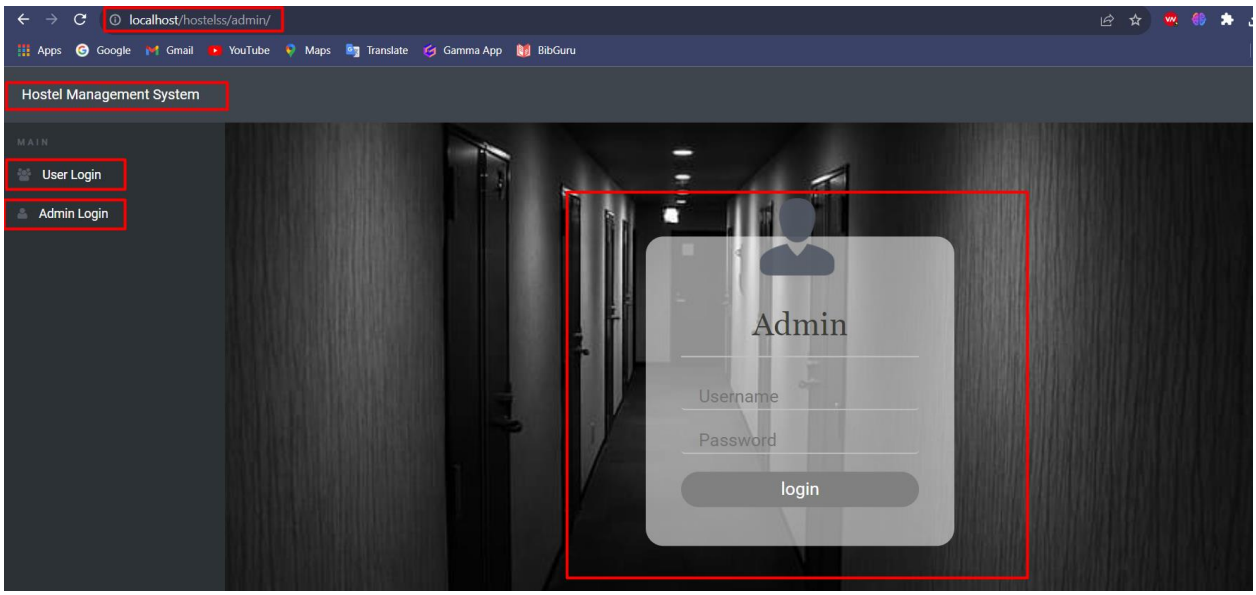


Figure 34: Admin login page

Here when you click this **"Admin Login"** you'll navigate to Admin login page. The Admin Login Page, which serves as the administrator's point of access and control for the many system functionalities, is a crucial component of the Hostel Management Website. This report provides an overview of the Admin Login Page's functionality and design, including its overall layout and user interface elements.



Here this is the user interface for the Admin Login. Here we have a navbar which consist of **"Hostel Management System"** and in the Side bar we have **"User Login" and "Admin Login".** Here it'll ask you to enter the username and password. If you enter the user and password it'll check in the admin table whether the password and mail you've entered is correct

or not. If that is correct, you'll redirected to Admin user interface if not you'll given an alert about the password/email error.

The verification is done using the Database table **"Admin".** It holds the authenticable data in it.

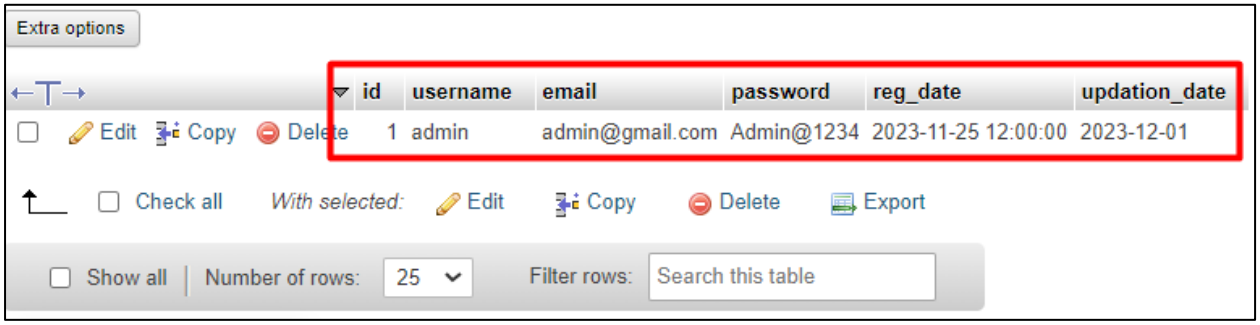


Figure 35: Admin Database

If you type the correct Username and password only it'll navigate to Admin UI(Admin Dashboard). If you don't give it'll give you an error

Figure 36: Invalid login

5.10 Admin/Dashboard Page

If the Admin login is correct, you'll navigate to Admin Dashboard

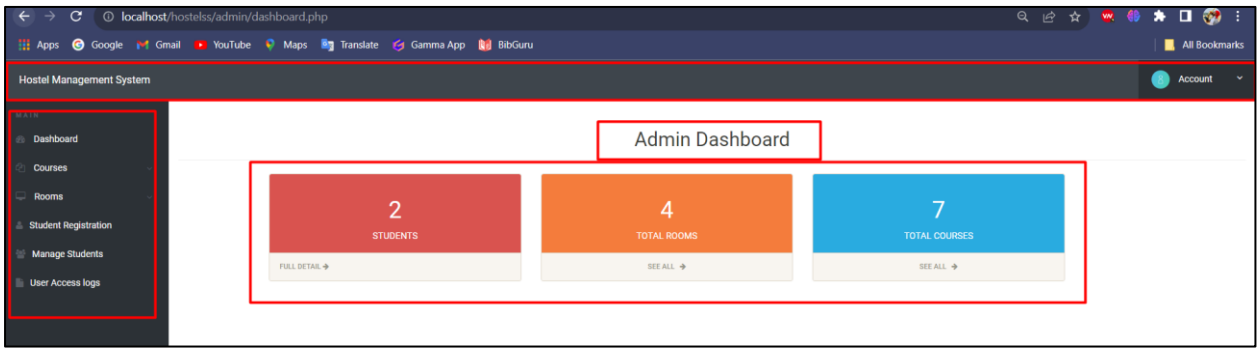


Figure 37: Admin Dashboard

Here we have Navbar, Sidebar and body. In the Navbar we have the topic name **"Hostel Management System" and a Dropdown type "Account".** When you click on that

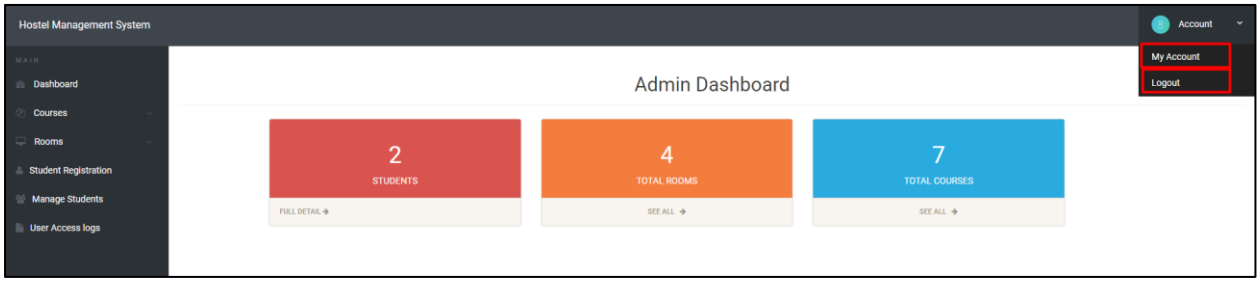


Figure 38: Admin Logout

We will have **"My Account"** and **"Logout"**. When we click on **"My Account"**

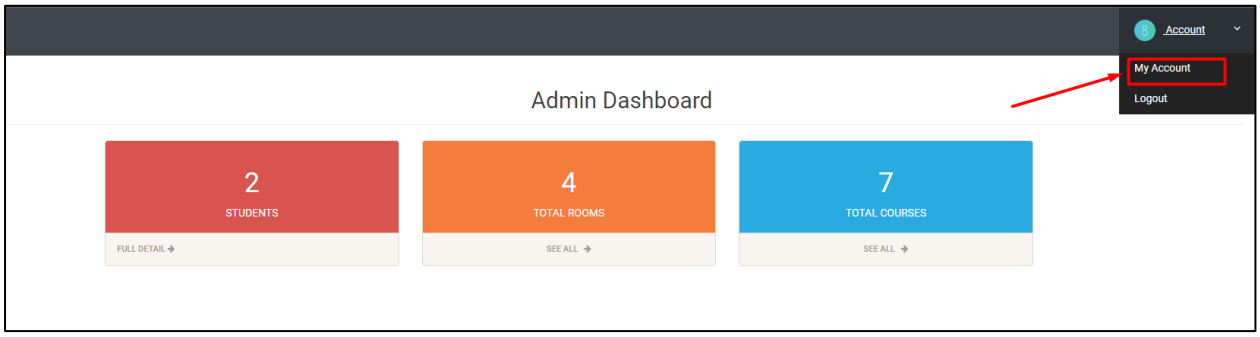


Figure 39: My Account

When you click on this it'll redirect to Admin Profile Page which Contains the information of Admin

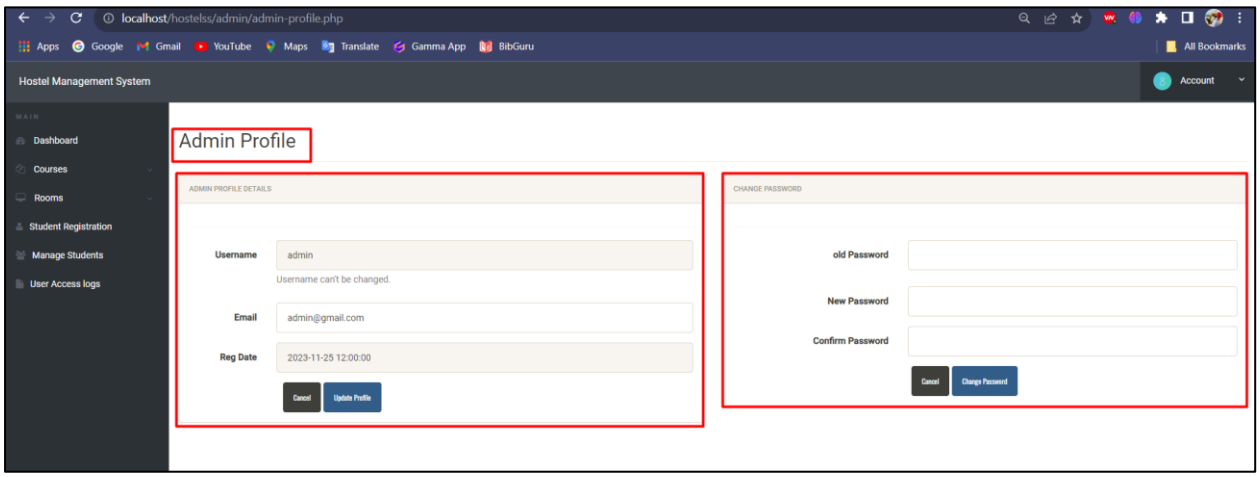


Figure 40: Admin Profile

Here we can see the admin details in one Side and Change Password in another side. Using this we can update the details if needed or else we can leave as it is.

When you click on **Logout**

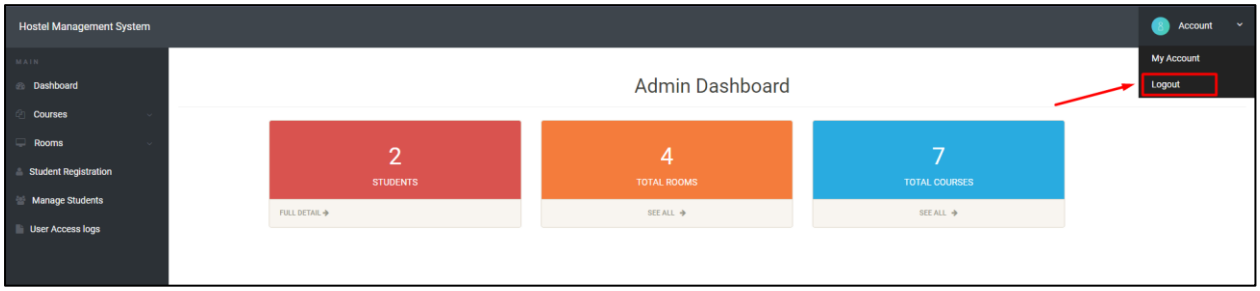


Figure 41: Admin Account

It'll move to the **admin login page.**

In the sidebar of the admin Dashboard, we have some menus

In the body of the Dashboard, we have

Dashboard: Dashboard is the default page after the admin login.

Courses: Courses is the page which contains course details. Here we can add new Courses.

Rooms: This option allows you to **add a new room** or **manage rooms**

Student Registration: Using this we can register room for a student.

Manage Students: This page will give you the rooms details

User Access log: This option will tell your users login date time and other related entries.

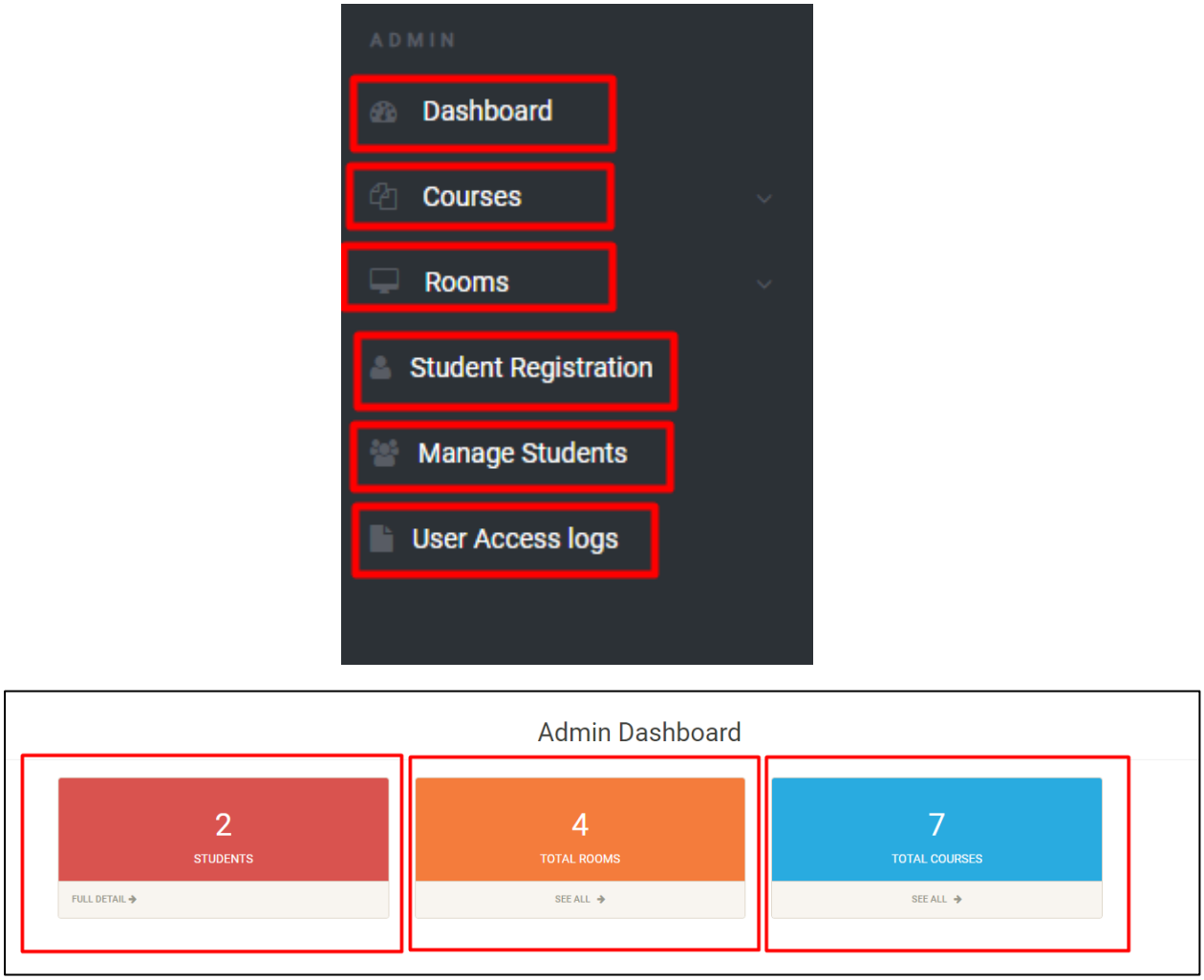
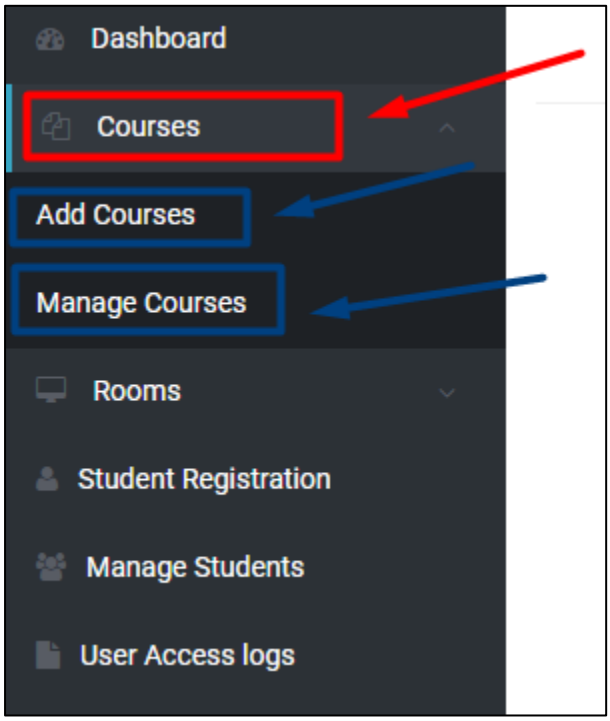


Figure 42: Admin Dashboard

When you click **STUDENTS**, you'll see the Students Details. When you click on **TOTAL ROOMS**, you'll see the available rooms. When you click **TOTAL COURSES**, you'll see Total Available Courses.

5.11 Admin / Courses Page



When you click on **Courses** you'll get

Add Courses: Here using this we can add the courses.

Manage Courses: Here using this we can manage those courses.

i. Add Courses

When you click on **"Add Courses",** we'll provide an Interface to add a new course

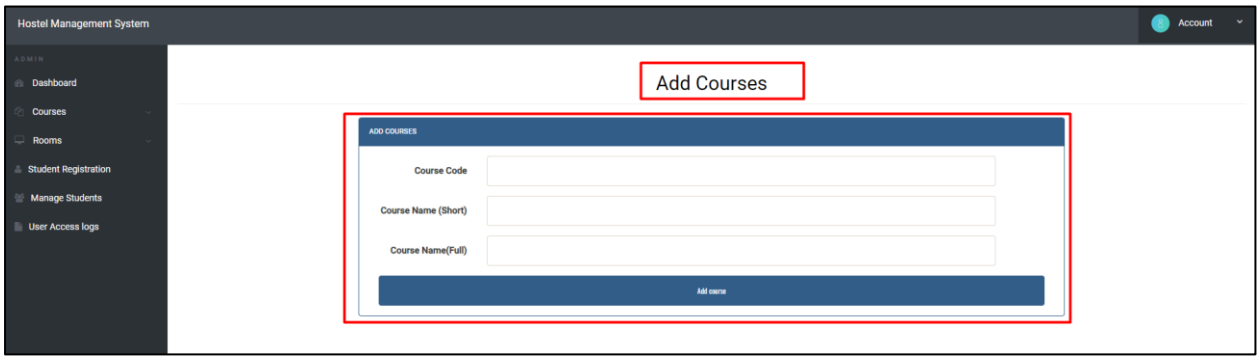


Figure 43: Add course form

Here you'll get a form like structure to add a new Course. Here you've to add Course Code, Course Name (Short and Full). Once you finish and press the add course button it'll add in the Database course table.

ii. Manage Course

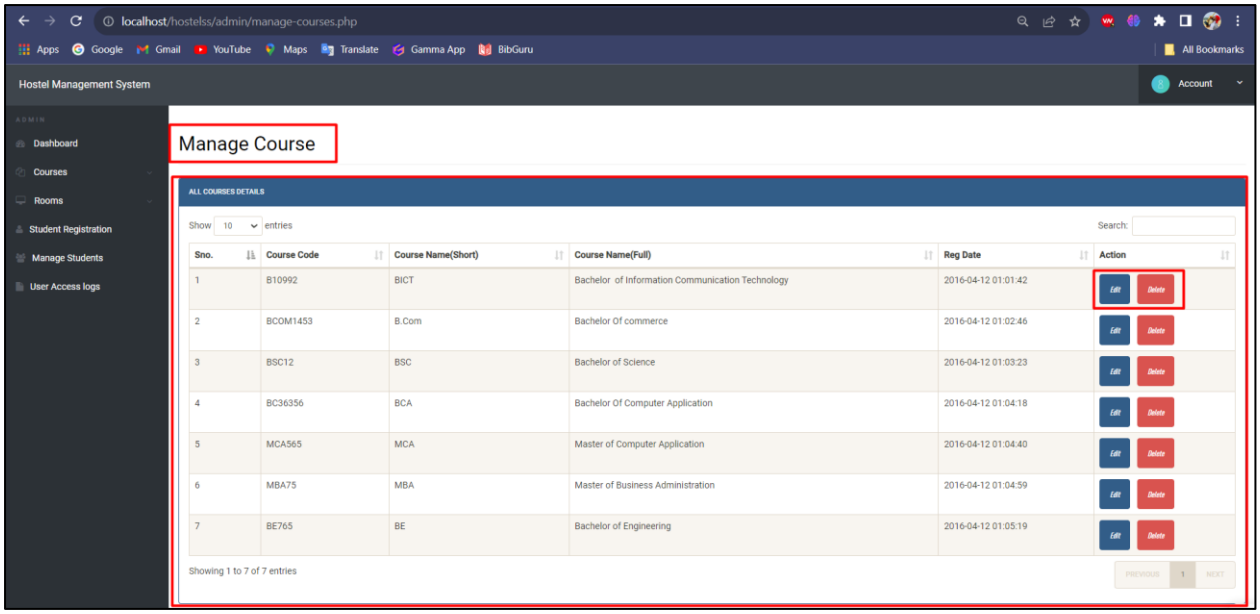


Figure 44: Manage course page

Here in the Manage Course, we have all the available courses in a table like Structure. Here in the Action Column, we have two buttons namely **Edit** and **Delete**. When you click on **Edit**

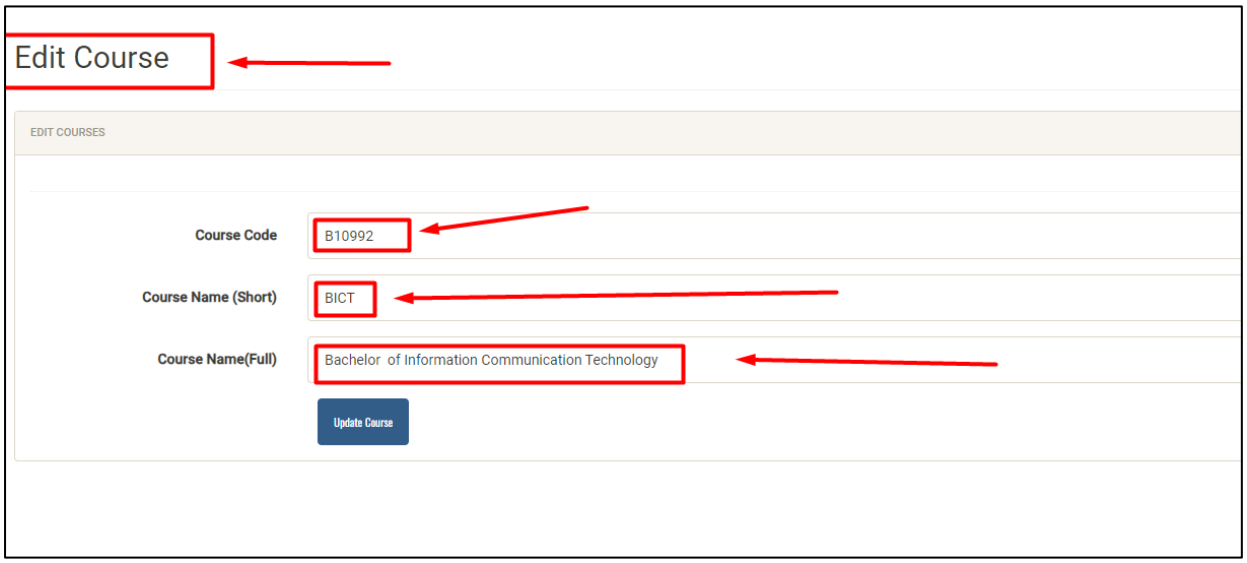


Figure 45: edit Course page

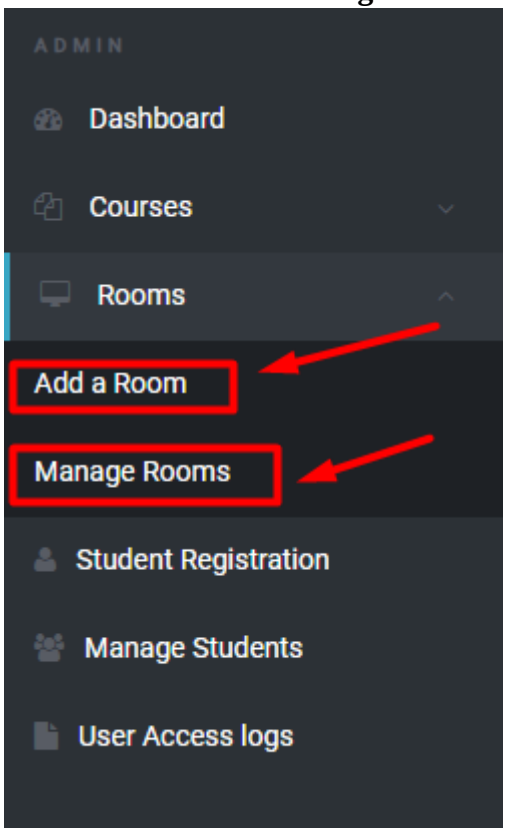
Through this we can edit our clicked Course. And if we change and click on Update, we can update the data. It'll change in the database table also.

When you click on the **Delete Button**

Figure 46: Delete Course Entry

The data will be deleted and give you an alert regarding the data deletion.

5.12 Admin / Rooms Page



When you click on **Rooms** you'll get

Add a Room: Here using this we can add room.

Manage Rooms: Here using this we can manage those rooms.

i. Add a Room

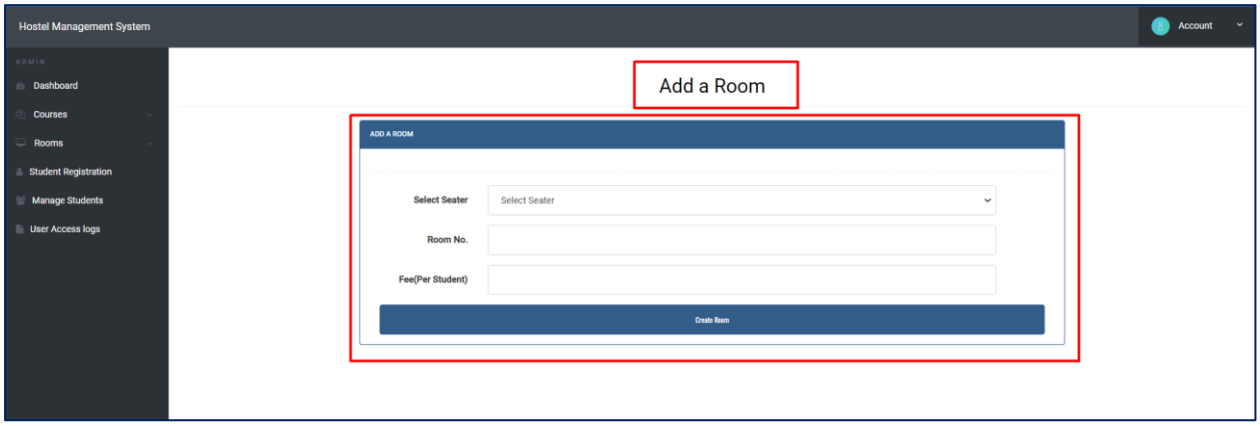


Figure 47: Add room form

Here we can add a room using this form. And the newly added room will be added to the database table also.

ii. Manage Rooms

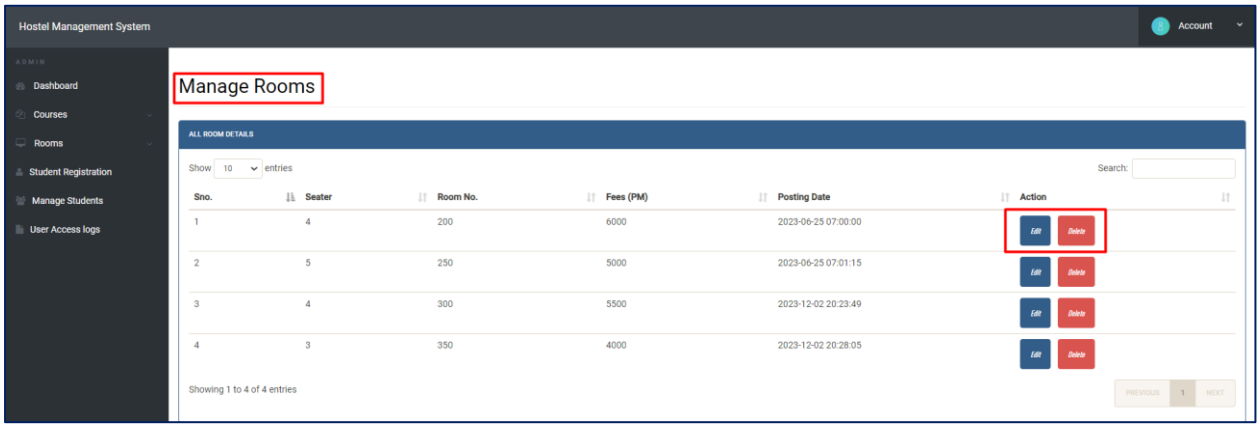


Figure 48: Manage room form

Here we can manage the Available rooms. Those rooms are displayed using the table. In the Action column, we have two actions. **Edit** and **delete** are those two.

5.13 Admin / Student Registration

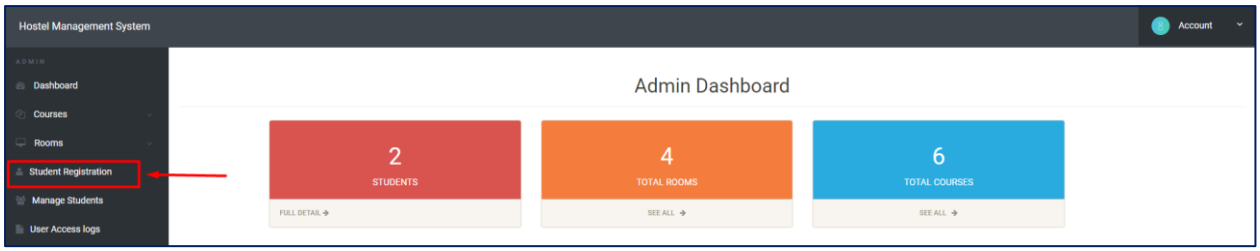


Figure 49: Student Registration

When you click on Student Registration, you'll redirected to Student Room Registration Page

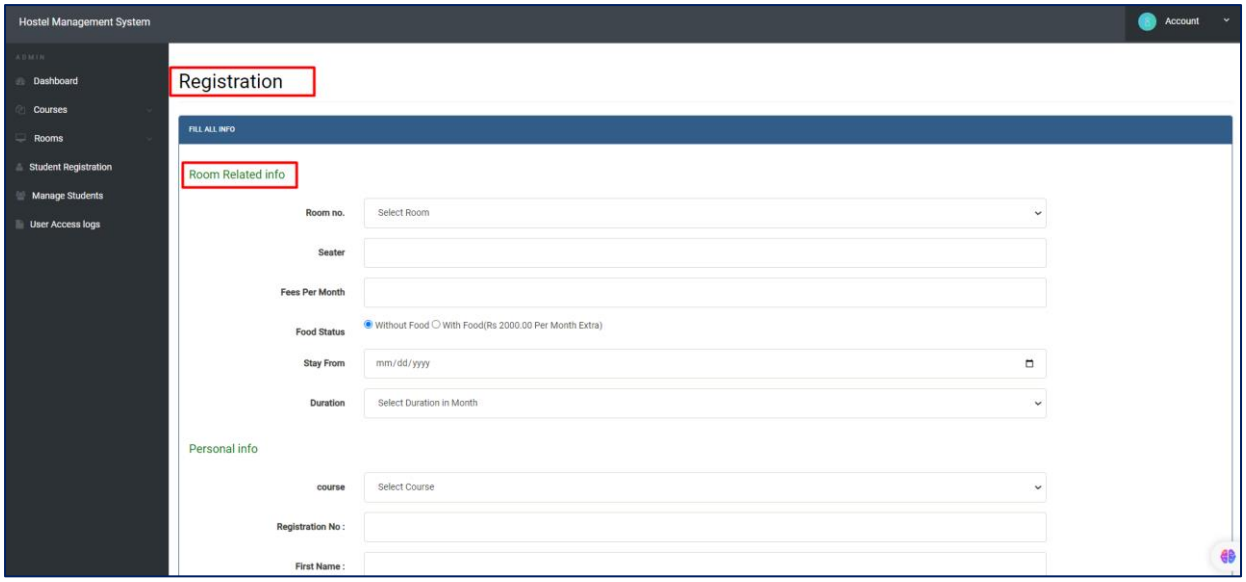


Figure 50: Student registration form

Here the admin can add/reserve room for student.

5.14 Admin / Manage Students

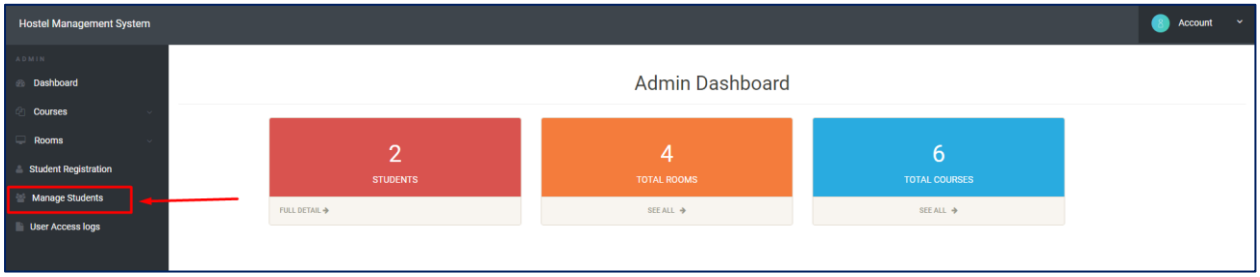


Figure 51: Manage Student

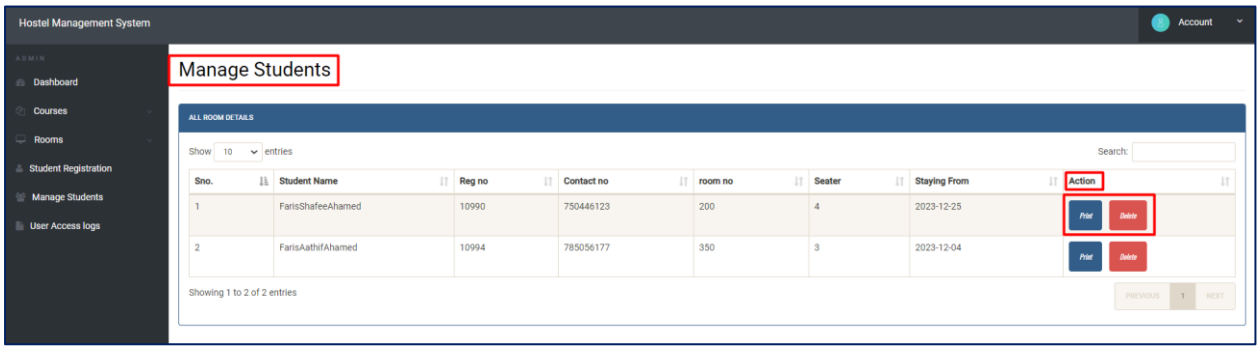


Figure 52: Manage student form

One of the most important parts of our hostel administration system is the Manage Students page, which is made to effectively manage and arrange records of the students that are staying there. All hostelers can maintain their login credentials and associated information in one place on this page. The major objective is to ensure a seamless and well-organized procedure by streamlining the administrative duties associated with student housing.

Here we can see all the student details in the table. In the Action part we have buttons called **Print** and **Delete**. When you click on **Print**

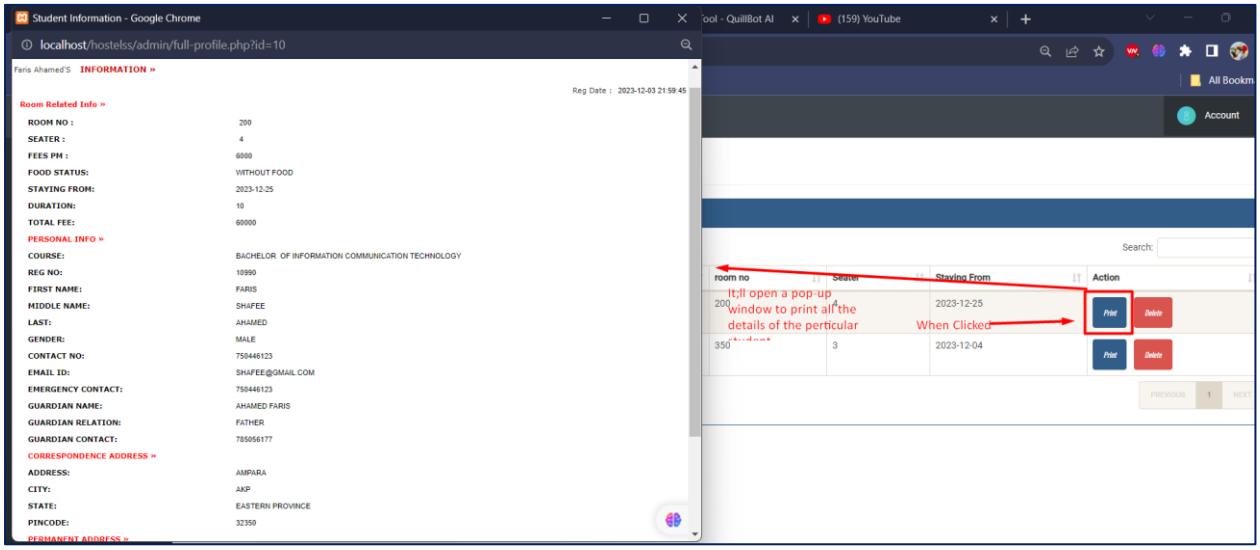


Figure 53: Print Student details

When you click on print, there'll be a pop-up window to print the details of students. It'll make the work of admin easier.

If we press delete, the entire details of that particular student will be deleted.

5.15 Admin / User Access Logs

The Access Log Page is one of the most crucial tools for keeping an eye on user activity within a system or application. It is a record-keeping feature designed to capture crucial information such as times, dates, and user log-in details. Here admin can see all the login details of all the students with time, date and related stuffs.

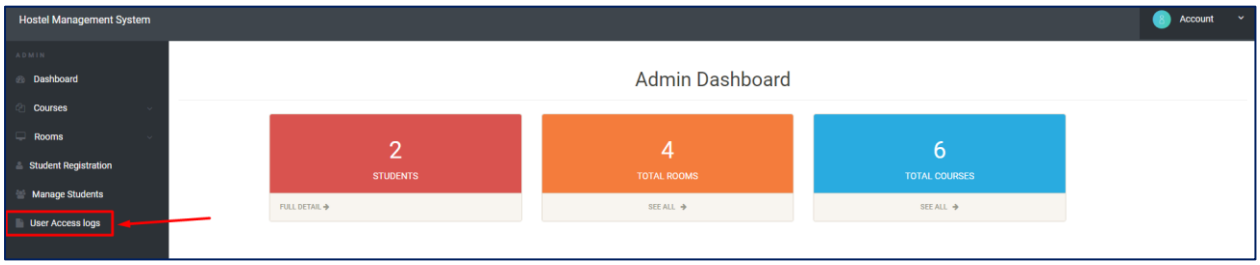


Figure 54: User access logs

When you click on User Access Logs, you'll able to see all the user's login stuffs.

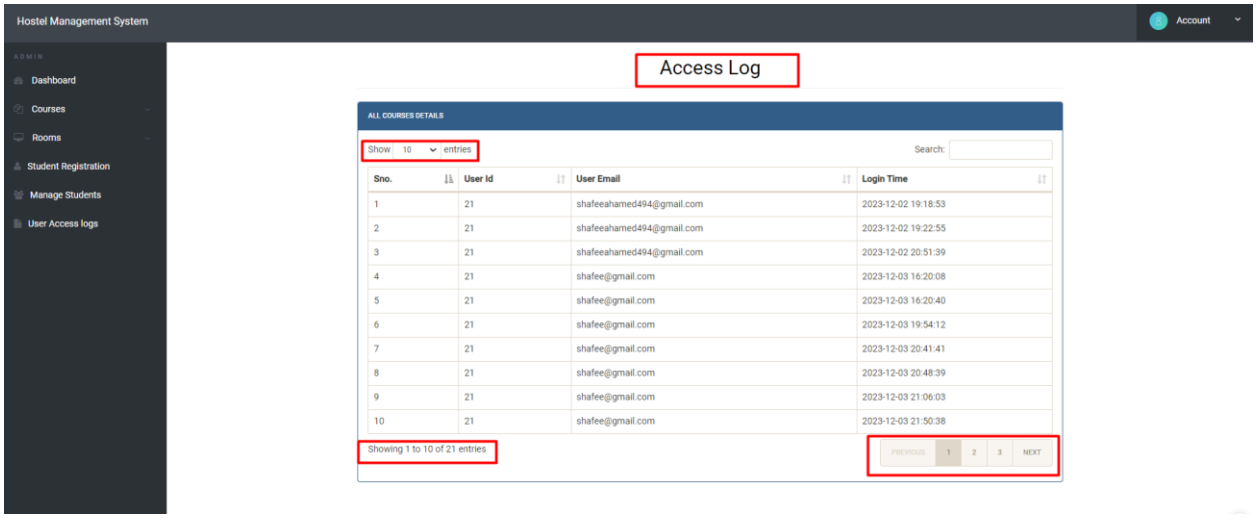


Figure 55: User access form

Here we can set the entries to show in the page using the show drop list in the table.

Chapter 06

6. Conclusion

When using the legacy system, it is challenging to accomplish all process because the information is dispersed, and some of the data may be repetitive. It could be extremely time-consuming to gather relevant information. The legacy system has some additional problems, such as maintenance of the system, not every employer being familiar with that system, and the cost of the system. This project will solve all these issues once and for all.

The "Hostel Management System" project was designed using an object-oriented program of PHP. This system added many user-friendly functions to make it a user-interactive web application. The website was designed so that even a user with a basic understanding of computers would have no trouble operating the system. The system is responsible for adhering to all the constraints that have been specified. It has been determined that the system offers significant advantages for the topic at hand. The system is both practical and trustworthy.

This system contributes to the universities' hostel staff's efforts to maintain the information that pertains to its user. It is simple for the administrator to gain access to and can remain unchanged for a significant time without being compromised. Without a shadow of a doubt, both the students and the admin benefit from using this system.

Chapter 07

7. References

[1] Build software better, together. (n.d.).

[2] Hostel management system in PHP with Source Code. (2022, April 5). CodeAstro. https://codeastro.com/hostel-management-system-in-php-with-source-code/

[3] Hostel management system project in PHP. (2018, September 26). Projectworlds | All Type Software Project Available Free or Paid; projectworlds | Free Projects and Free Learnings. https://projectworlds.in/free-projects/php-projects/hostel-management-system-project-in-php/

[4] Hunkanrin, S. (2018). HOSTEL MANAGEMENT SYSTEM(full project). https://www.academia.edu/37040228/HOSTEL_MANAGEMENT_SYSTEM_full_project_1

_

[5] (N.d.). Studocu.com. Retrieved December 4, 2023, from https://www.studocu.com/row/document/university-of-nairobi/information-communication-technology/hostel-management-system-full-project-1/31031685

[6] Otto, M., & Thornton, J. (n.d.). Bootstrap. Getbootstrap.com. Retrieved December 4, 2023, from https://getbootstrap.com/

[7] (N.d.). Openai.com. Retrieved December 4, 2023, from https://chat.openai.com/c/9c092b8f-be2d-4781-b363-2d18082d5831