Management System

CS-207 FINAL PROJECT

GROUP-1

Ujjawal Khadanga (B20139)
Gokul Goyal (B20100)
Dev Prajapat (B20093)
Sachin Mahawar (B20129)
Aditya Sood (B20077)
Niharika Batra (A20006)
Karan Baraik (B20110)

INDEX

ACKNOWLEDGMENT

We would like to thank our professor, Dr. Varun Dutt for providing us with the opportunity to take up this project on Hostel Management System and for teaching us the basics of database management and web development technologies through his lectures.

We would also like to express our gratitude to our head TA, Shashank Uttrani for his guidance and support. Tutorial sessions by Shanshank sir helped us a lot. And our group TA, Samarth Reddy for guiding us through our project work and for promptly addressing our queries and problems.

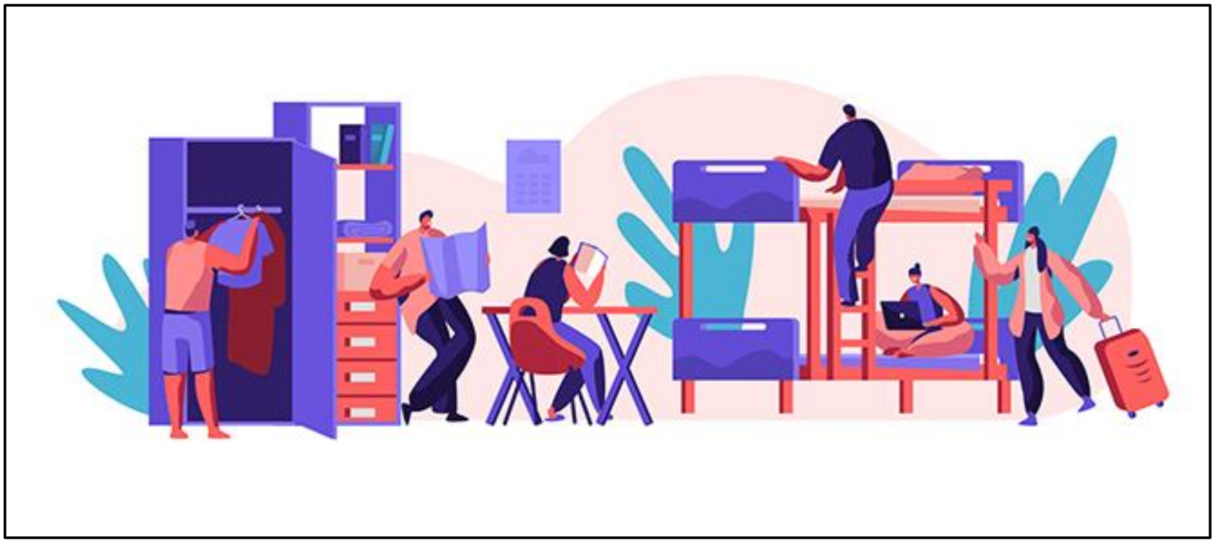
We wouldn't have been able to complete our project without the guidance and support of our professor and TAs. We thank them all.

ABSTRACT

"Hostel Management System" is a website developed for managing various activities in the hostel. As we know there are many students studying in the institute, there is a lot of strain on the people who are running the hostel and usually software is not used in this context. This project deals with the problems of managing a hostel and avoids the problems which occur when carried manually.

Hostel Management System helps students and the university to assign students available rooms according to their convenience without any hassle.

We have developed our website in such a way that students can book their rooms, or swap or vacate them in case they need to. This website allows us to help students to book their rooms from the comfort of being in their homes without standing in queues to get residence in a hostel.



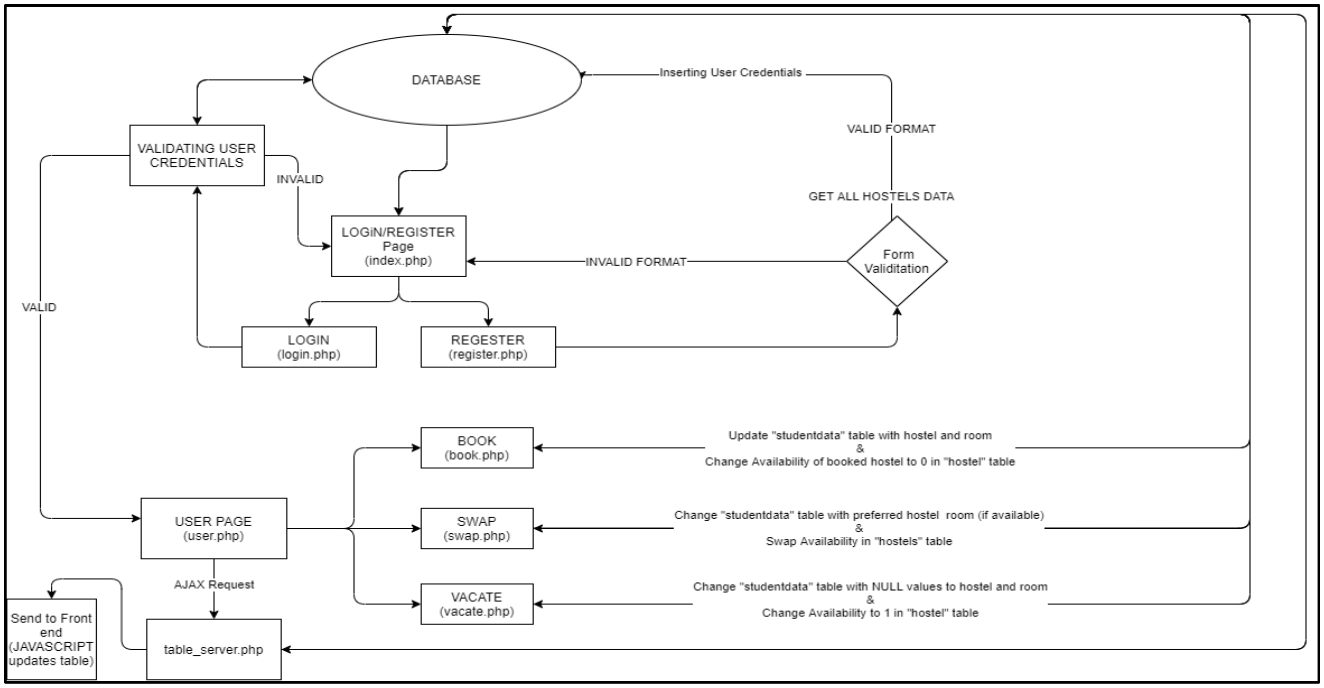
INTRODUCTION

Hostel management system is a web application that allows users to register/login to their accounts and reserve hostel seats. The application provides these 5 features to the user:

1) Login / Register an Account
2) Book Hostel seats that are available
3) Check the availability of Hostel Rooms
4) Swap their *Current Hostel* with another Available one
5) Vacate their current Hostel

While Booking/ Swapping/ Vacating the hostel, all their details would be safely stored in the database. The user can logout from their account and end the session.

The website is hosted Here


DATABASE
Inserting User Credentials
VALID FORMAT
GET ALL HOSTELS DATA
VALIDATING USER CREDENTIALS
INVALID
LOGiN/REGISTER Page (index.php)
INVALID FORMAT
Form Validitation
VALID
LOGIN (login.php)
REGESTER (register.php)
BOOK (book.php)
Update "studentdata" table with hostel and room & Change Availability of booked hostel to 0 in "hostel" table
USER PAGE (user.php)
SWAP (swap.php)
Change "studentdata" table with preferred hostel room (if available) & Swap Availability in "hostels" table
AJAX Request
VACATE (vacate.php)
Change "studentdata" table with NULL values to hostel and room & Change Availability to 1 in "hostel" table
Send to Front end (JAVASCRIPT updates table)
table_server.php

METHODOLOGY

The Web Application uses **PHP** for backend and **MySQL** for database. The front end is made with **HTML** and **vanilla CSS** and **Javascript** (vanilla and **jQuery**).

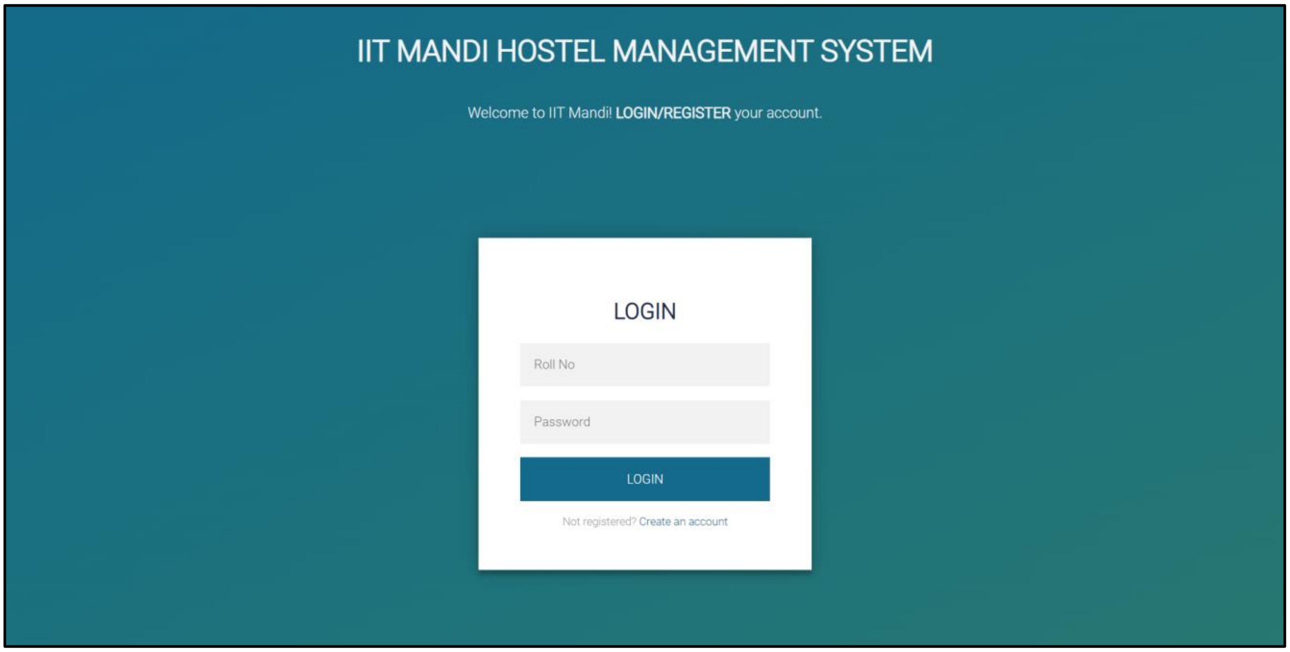
The application has 2 endpoints:

1) **Sign In Page** (index.php) - It is the entry point for our application. Users can login/register their accounts here using the form provided.
2) **User page**(user.php) - When logged in, a user will be directed to this endpoint. Here all the main functionalities like **book**, **swap**, **vacate**, **check available rooms** are present.

Front End

The HTML markup is displayed with the help of PHP on both the Sign In and User page.

1) Sign In Page - The Sign In page has 2 forms on the same page, the **login form** submits the data to *login.php* while the **register form** submits the data to *register.php*, where the backend connects to the database to execute the required sql queries. The javascript library, **JQuery**, is used in the sign in page for animating the toggle between the login and register forms.



IIT MANDI HOSTEL MANAGEMENT SYSTEM

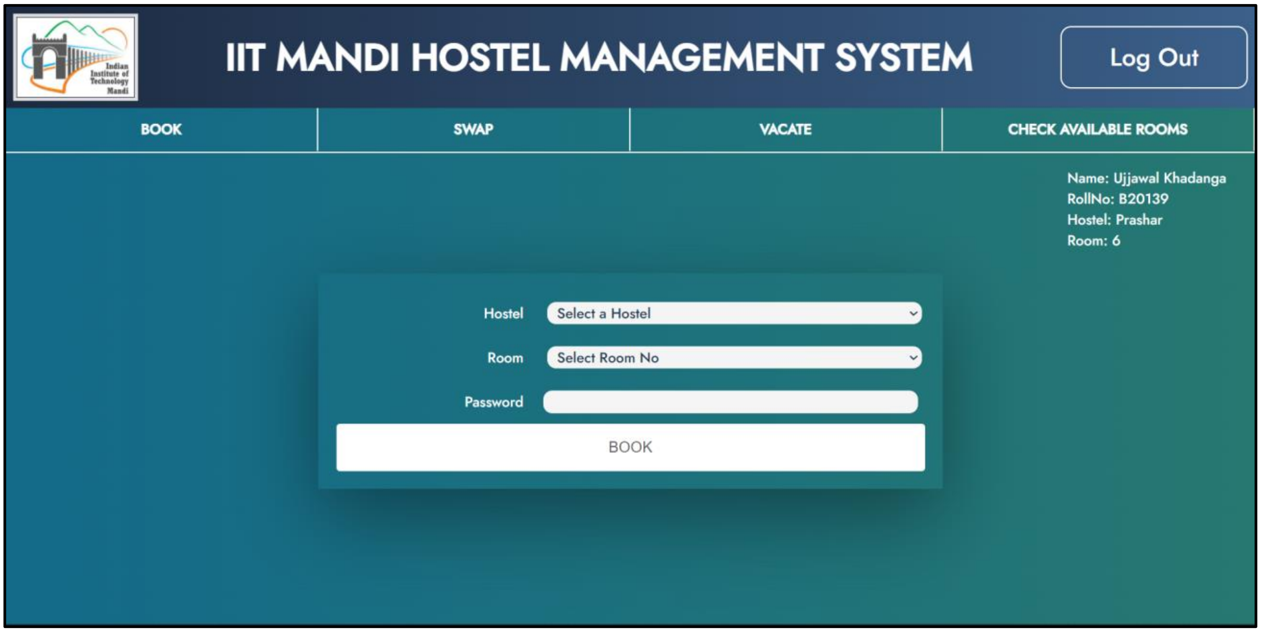
Welcome to IIT Mandi! **LOGIN/REGISTER** your account.

2) User Page - The User page has a **Header**, **Navigation Bar**, a **Main-Container** for forms and tables.

a) **Header** - The header has the logo title and Logout button. The header is styled using css “flexbox” which gives flexibility in positioning. The Logout Button is an anchor tag which on click, redirects to logout.php.

b) **Navigation Bar** - The Navigation Bar is an unordered list(<*ul*>) with 4 list items(<*li*>). Book , Swap, Vacate, Check Available Rooms. It is also styled using CSS flexbox. When clicked, the javascript toggles between the respective forms.

c) **Main Container** - The main container displays all the forms and tables corresponding to the selected menu on the navigation bar.

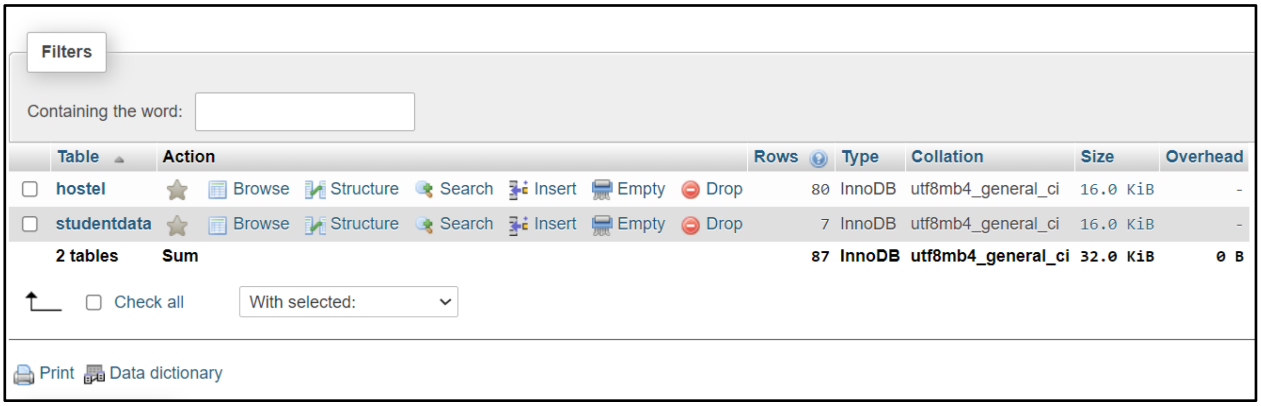


Name: Ujjawal Khadanga
RollNo: B20139
Hostel: Prashar
Room: 6

Hostel: Select a Hostel
Room: Select Room No
Password:

NOTE : All the elements are styled responsive, to ensure that users with small screened devices don't face any issues.

Database



Filters

Containing the word:

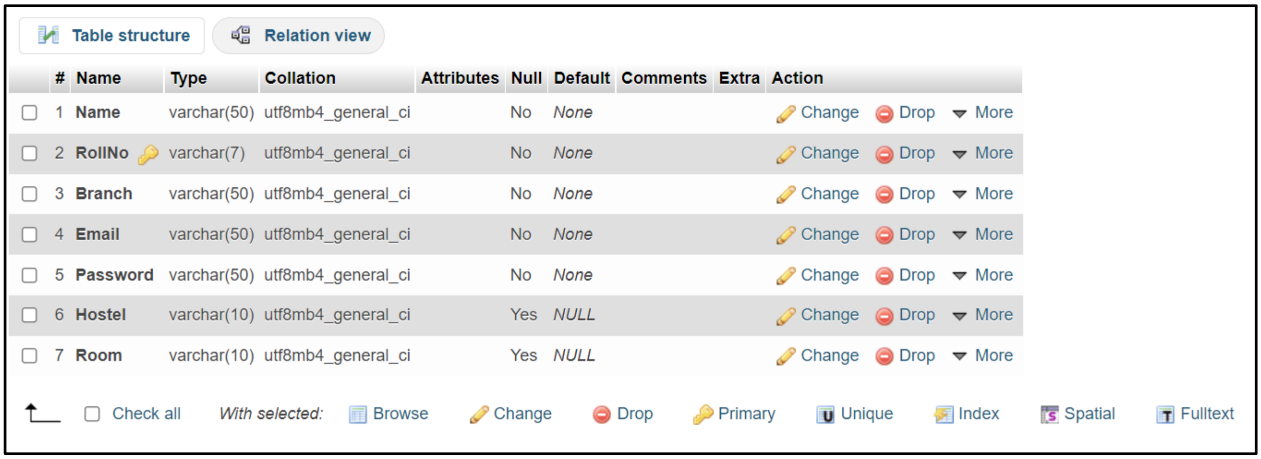
	Table	Action	Rows	Type	Collation	Size	Overhead
□	hostel	Browse Structure Search Insert Empty Drop	80	InnoDB	utf8mb4_general_ci	16.0 KiB	-
□	studentdata	Browse Structure Search Insert Empty Drop	7	InnoDB	utf8mb4_general_ci	16.0 KiB	-
	2 tables	Sum	87	InnoDB	utf8mb4_general_ci	32.0 KiB	0 B

□ Check all With selected:

Print Data dictionary

The database named, ***hms_database*** has the following tables:

1) *studentdata* - It stores all the information entered by the user at the time of registration. It also has columns "hostel" and "*room*" which store the assigned hostel name and room number respectively . At the time of registration, these 2 columns are assigned "*NULL" values*.



	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	Name	varchar(50)	utf8mb4_general_ci		No	*None*			Change Drop More
□	2	RollNo	varchar(7)	utf8mb4_general_ci		No	*None*			Change Drop More
□	3	Branch	varchar(50)	utf8mb4_general_ci		No	*None*			Change Drop More
□	4	Email	varchar(50)	utf8mb4_general_ci		No	*None*			Change Drop More
□	5	Password	varchar(50)	utf8mb4_general_ci		No	*None*			Change Drop More
□	6	Hostel	varchar(10)	utf8mb4_general_ci		Yes	*NULL*			Change Drop More
□	7	Room	varchar(10)	utf8mb4_general_ci		Yes	*NULL*			Change Drop More

□ Check all *With selected:* Browse Change Drop Primary Unique Index Spatial Fulltext

2) *hostel* - It stores all the information about all the hostels and rooms and their availability. The

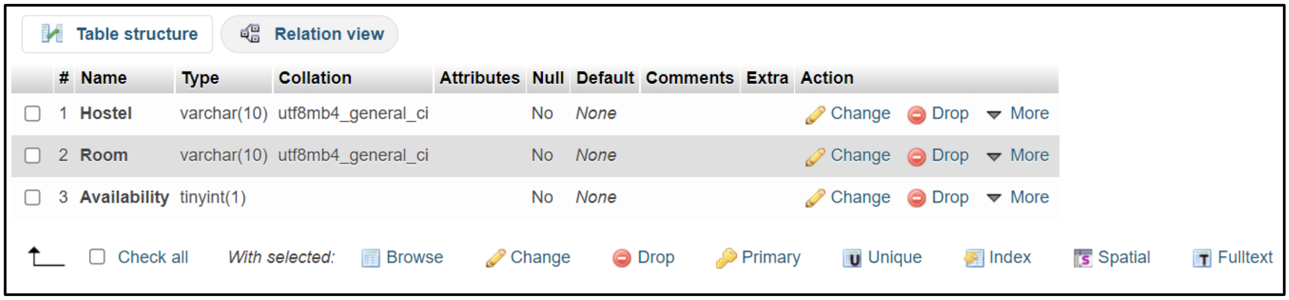


Table structure
Relation view

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	Hostel	varchar(10)	utf8mb4_general_ci		No	*None*			Change Drop More
□	2	Room	varchar(10)	utf8mb4_general_ci		No	*None*			Change Drop More
□	3	Availability	tinyint(1)			No	*None*			Change Drop More

□ Check all *With selected:* Browse Change Drop Primary Unique Index Spatial Fulltext

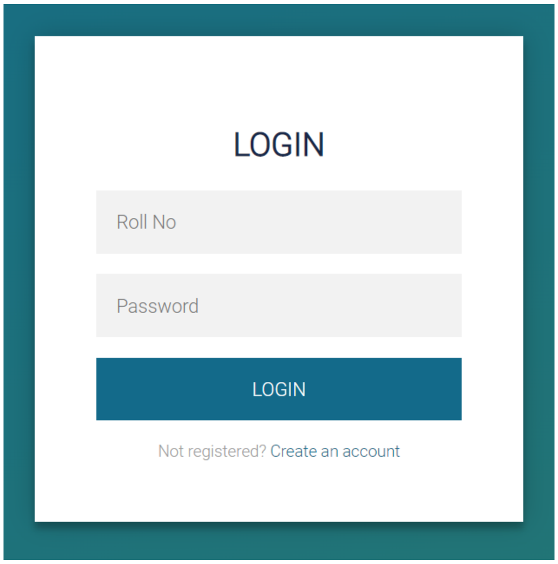
Backend

The PHP backend implements the following functions and features:

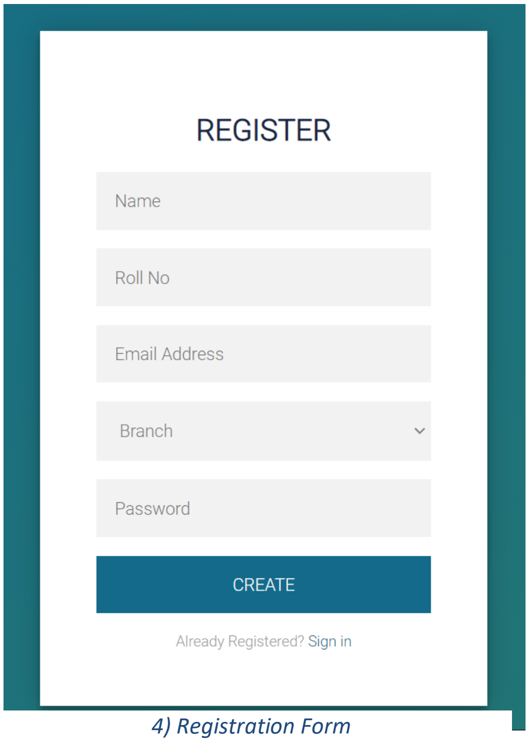
1) **Database connection** - The file “*dbconnection.php”* connects to the database using appropriate database-user details. This file is then included wherever database connection is required.
2) **Use of session variables** - The session variables store user-information like name(*user_name*), roll no(*user_RollNo*), hostel(*user_Hostel*) and room(*user_Room*). Session is initiated in “*login.php”* where all the information is retrieved from the database and stored as session variables. It also gets and stored as “*hostels_array*” as a session variable which stores available hostel rooms in json format.
3) **Book** - Submitting the booking form on the user page submits the data to “*book.php*”. It authenticates the user using the password, then updates the hostel name and room for the user if they are set to “*NULL”* or else redirects back to user.php. The availability in the hostel table is also changed to *“False(bool)”*. Also, “*user_hostel*” and “*user_room”* session variables are updated.
4) **Swap** - Submitting the swap form on the user page submits the data to “*swap.php”*. Firstly, it authenticates the user using password. If the “*currentHostel”* and “*currentRoom”* are not set to “*NULL”* (i.e. The user has not already booked a room), then it changes the hostel and room in the database to the given “*preferredHostel”* and “*preferredRoom*”. The hostel table is also updated accordingly. Also, session variables are updated.
5) **Vacate** - Submitting the swap form on the user page submits the data to “*vacate.php”*. It only accepts the user password which is used for user authentication, then updates the hostel and room to “*NULL”* in studentdata table. Availability is set to “*True(bool)”* for the vacated room. Also, session variables are updated.
6) **Check Available Rooms** - The form takes input as the Hostel names and displays all the rooms in that hostel with their availability status. The *“table_server.php”* takes the post data and filters a query for all the available rows in the hostels table corresponding to the hostel name which is posted. It then echos the result in json format. Since posting the form data in the regular way would refresh the page, which will be a problem, we have used **AJAX** and **jquery** for this purpose. **Ajax** can request form servers after the page has loaded. After getting the required data from “*table_server.php” and* processing it, the table is displayed in the DOM.

RESULTS/OUTPUTS

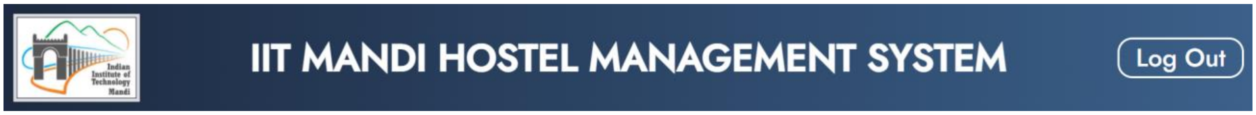
The website is intuitive to use, and it gives users the ability to book, swap, vacate or check availability of rooms fairly quickly and conveniently. The following are the independent components that were developed within the project.



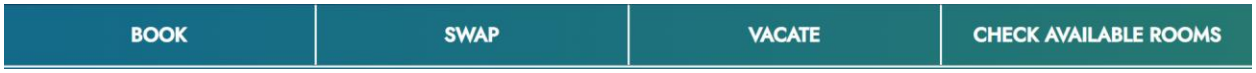
1) Login Form



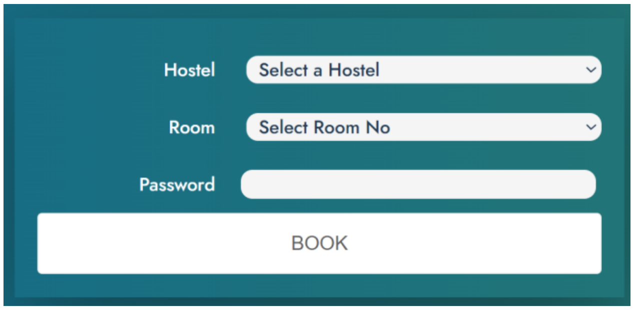
4) Registration Form


Indian Institute of Technology Mandi
IIT MANDI HOSTEL MANAGEMENT SYSTEM
Log Out

3) Header



2) Navbar

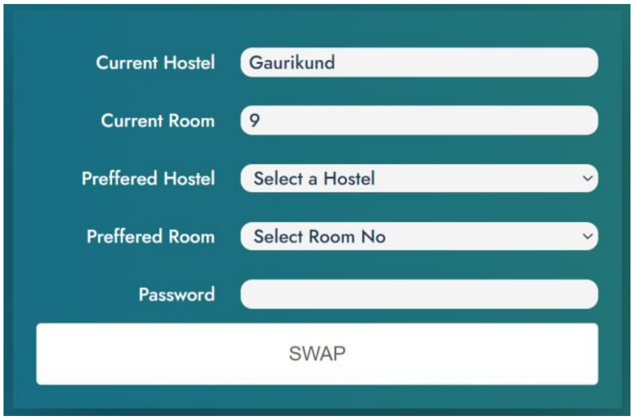


Hostel Select a Hostel

Room Select Room No

Password

5) Booking



Current Hostel Gaurikund

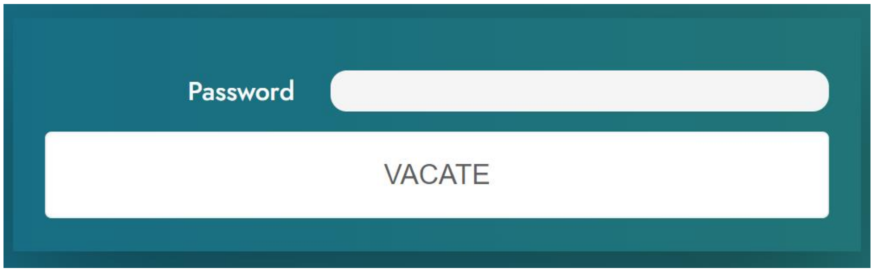
Current Room 9

Preffered Hostel Select a Hostel

Preffered Room Select Room No

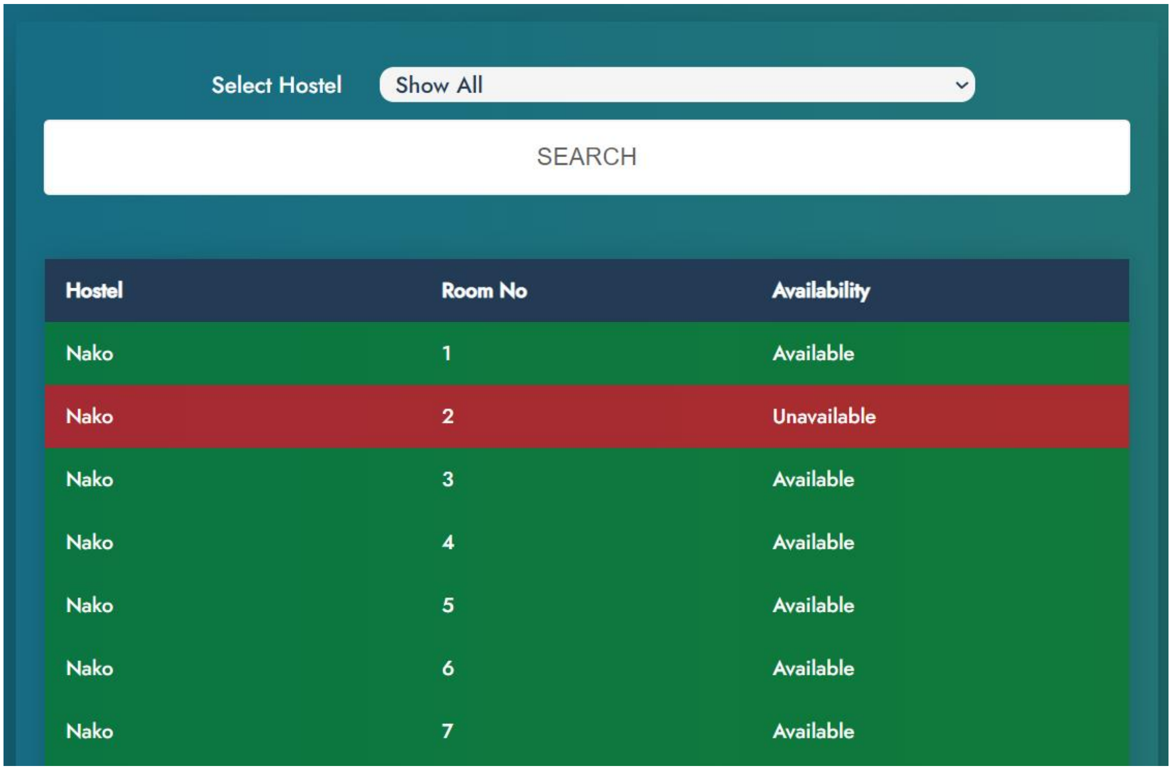
Password

6) Swap Room



Password

8) Vacate Room



Select Hostel Show All

Hostel	Room No	Availability
Nako	1	Available
Nako	2	Unavailable
Nako	3	Available
Nako	4	Available
Nako	5	Available
Nako	6	Available
Nako	7	Available

7) Availability Checking

CONCLUSION/SUMMARY

This project was aimed at learning Frontend, Backend, Databases, and their interaction with each other to build a complete web-application. We have learnt about various front-end technologies and languages like HTML, CSS, JavaScript and JQuery. We have learnt how relational databases like MySQL are designed and how they work. We have also learnt about backend technologies like PHP, and their integration with MySQL for building a full-stack web-application.

- This project helps students to conveniently manage their hostel rooms.
- Students can view which room is available/unavailable and then can accordingly book/swap their choice.
- The database helps college management to keep proper records of available/unavailable rooms.