“CHAT ROOM”

Mini- Project

(Third Year/ Sem V)

Submitted in fulfilment of the requirement of University of Mumbai
For the Degree of

Bachelor Of Engineering
(Computer Engineering)

By

1.	**Karan Dhiman**	**TE-Comps B-28**	**TU3F1819099**
2.	**Mayuresh Phansikar**	**TE-Comps B-31**	**TU3F1819102**
3.	**Amey Thakur**	**TE-Comps B-50**	**TU3F1819127**
4.	**Hasan Rizvi**	**TE-Comps B-51**	**TU3F1819130**

Under the Guidance of

Mrs Reshma Rohan Koli

Department of Computer Engineering
TERNA ENGINEERING COLLEGE
Plot no.12, Sector-22, Opp. Nerul Railway station,
Phase-11, Nerul (w), Navi Mumbai 400706
UNIVERSITY OF MUMBAI

2020-2021

Internal Approval Sheet

Terna Engineering College

NERUL, NAVI MUMBAI

CERTIFICATE

This is to certify that

1. **Karan Sukhvinder Dhiman**
2. **Mayuresh Deepak Phansikar**
3. **Amey Mahendra Thakur**
4. **Hasan Mehdi Rizvi**

*Has satisfactorily completed the requirements of the **WDL Mini Project (Third year/Sem V)***

entitled

"Chat Room"

*As prescribed by the **University of Mumbai** Under the guidance of*

Prof. Reshma Koli

Guide **APC** **HOD**

Approval Sheet

Project Report Approval

This Mini Project Report - I entitled

“Chat Room”

by the following students is approved for the degree of

Bachelor in "Computer Engineering(Sem-V)".

Submitted by:

KARAN DHIMAN	**TU3F1819099**
MAYURESH PHANSIKAR	**TU3F1819102**
AMEY THAKUR	**TU3F1819127**
HASAN RIZVI	**TUEF1819130**

Examiners Name & Signature:

1. ----------

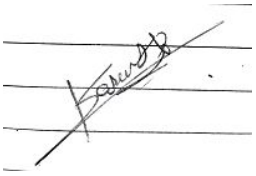
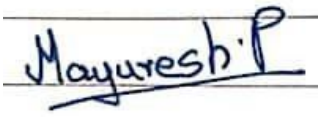
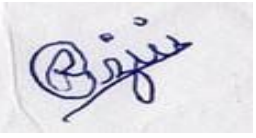
2. ----------

Date: 25-11-2020

Place: MUMBAI

DECLARATION

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

KARAN DHIMAN	**TU3F1819099**	
MAYURESH PHANSIKAR	**TU3F1819102**	
AMEY THAKUR	**TU3F1819127**	
HASAN RIZVI	**TU3F1819130**	

Date: 25-11-2020

Place: MUMBAI

ACKNOWLEDGEMENT

We would like to express my sincere gratitude towards my guide, **Prof. Reshma Koli** for the help, guidance and encouragement they provided during the Mini Project. This work would have not been possible without their valuable time, patience and motivation. I thank them for making my stint thoroughly pleasant and enriching. It was great learning and an honour being their student.

I am deeply indebted to **Prof. Archana Mire,** HOD of Computer Engineering Department, **Prof. Mohini Misale,** Academic Program Coordinator and the entire team in the Computer Engineering Department. They supported us with scientific guidance, advice and encouragement, they were always helpful and enthusiastic and this inspired me in my work.

I take the privilege to express my sincere thanks to **Dr Lakshmappa Ragha,** our Principal for providing encouragement and support.

ABSTRACT

Earlier there was no mode of online communication between users. In big or small organizations communication between users posed a challenge. There was a requirement to record these communications and store the data for further evaluation. The idea is to automate the existing Simple Chat Room system and make the users utilize the software so that their valuable information is stored digitally and can be retrieved for further management purposes. There was no online method of communicating to different users. There were many different interfaces available in the market but this method of using windows sockets to communicate between nodes would be fast and reliable.

The main objective of our Simple Chat Room project is to create a chat application which helps different users to communicate with each other through a server connected. This is a simple chat program with a server and can have many clients. The server needs to be started first and clients can be connected later. Simple Chat Room provides bidirectional communication between client and server. It enables users to seamlessly communicate with each other. The user can chat using this chat application. If the user at the other end is active then they can start a chat session. The chat is recorded in the application.

LIST OF FIGURES

Figure No.	Figure Name	Page No.
1.2	Flow Diagram	**11**
1.2.1	Zero Level DFD	**12**
1.2.2	First Level DFD	**13**
1.2.3	Second Level DFD	**14**
5.2	System Layout Architecture	**22**

LIST OF ABBREVIATIONS

Acronym	Abbreviation
PHP	Hypertext Preprocessor (Personal Home Page)
WS	Web Socket
HTTP	HyperText Transfer Protocol
CSS	Cascading Style Sheets Cascading Style Sheets
DFD	Data Flow Diagram

TABLE OF CONTENTS

Caption		Page No.
CERTIFICATE		**2**
APPROVAL SHEET		**3**
DECLARATION		**4**
ACKNOWLEDGEMENT		**5**
ABSTRACT		**6**
LIST OF FIGURES		**7**
LIST OF ABBREVIATIONS		**8**
Repository - *https://github.com/Amey-Thakur/Chat-Room*		
CHAPTER 1	***INTRODUCTION***	**11**
	1.1 Need for Chat Room	**11**
	1.2 Architecture of Chat Room System	**12**
	1.2.1 Zero Level DFD	**13**
	1.2.2 First level DFD	**14**
	1.2.3 Second level DFD	**15**
	1.3 Applications of your system	**16**
	1.4 Organization of Report	**16**
CHAPTER 2	***Literature Survey***	**17**
	2.1 Brief History of Of-Line System or Desktop Based System	**17**
	2.2 Comparison of Chat Room system with previous system	**18**
CHAPTER 3	***Problem Statement***	**19**
	3.1 Problem Statement	**19**
	3.2 Solution Statement	**19**

	3.3 Project Scope and Features	**19**
CHAPTER 4	***System Analysis***	**20**
	4.1 System Requirement	**20**
	4.1.1 Hardware Requirements	**20**
	4.1.2 Software Requirements	**21**
	4.1.3 Functional Requirement	**21**
	4.1.4 Non-Functional Requirements	**22**
CHAPTER 5	***Design***	**23**
	5.1 Design Process	**23**
	5.2 System Layout	**23**
	5.3 User Interface Design	**24**
	5.4 Database Design	**24**
Chapter 6	***Results and Discussions***	**25**
	6.1 Input/Program	**25**
	6.2 Snapshots	**32**
Chapter 7	***Conclusion***	**38**
REFERENCES		**39**

CHAPTER 1

INTRODUCTION

1.1 Need for Chat Room

We developed an application which facilitates the creation of a chat room with a live server for the users to enable sharing messages or chat on the go. Develop an instant messaging solution to enable users to seamlessly communicate with each other and also which can be used by any novice user. I.e. Live chat room on the fly (online).

Teleconferencing or Chatting is a method of using technology to bring people and ideas "together" despite geographical barriers. The technology has been available for years but the acceptance it was quite recent.

This mini project is an example of a chat server. The chat application is quite simple. It does not require a login, AJAX-style features, and will also offer support for multiple users. It is made up of two applications: the client application, which runs on the user's device and the server application, which is hosted to run the chat room live on the network. To start chatting clients should get connected to a server where they can practice two kinds of chatting, public one (message is broadcasted to all connected users) and a private one (between any 2 users only) and during the last one security measures were taken.

1.2 Architecture of Chat Room System

Online Chat Application Data flow diagram is often used as a preliminary step to create an overview of the Chat Application without going into great detail, which can later be elaborated. It normally consists of overall application dataflow and processes of the Chat Application process. It contains all of the user flow and their entities such all the flow of Chat, Chat History, Chat Profile, Smiley Chat, Chat User, Chat Group, Chat Delete. All of the below diagrams has been used for the visualization of data processing and structured design of the Chat Application process and working flow.

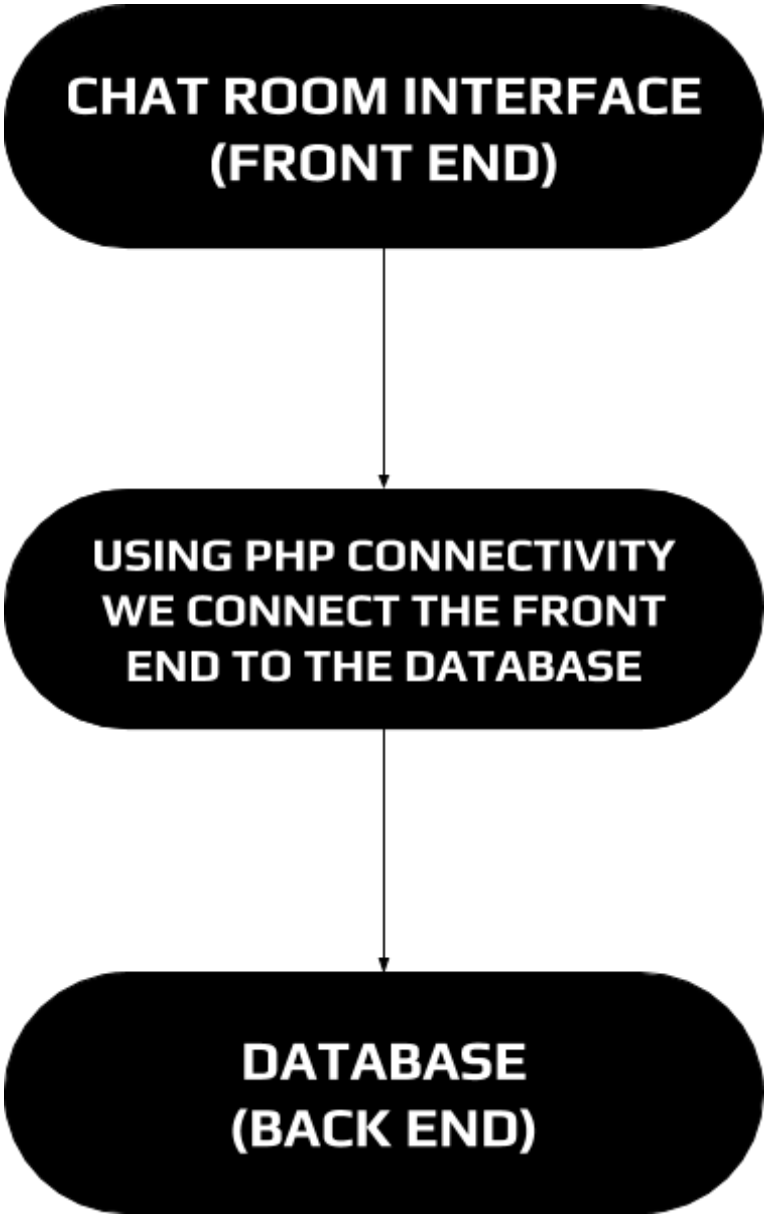


Fig 1.2 Flow Diagram

1.2.1 Zero Level Data Flow Diagram (0 Level DFD) OF Online Chat Application :

- ➔ This is the Zero Level DFD of Online Chat Application, where we have elaborated the high level process of Chat Application. It's a basic overview of the whole Online Chat Application or process being analyzed or modeled. It's designed to be an at-a-glance view Chat User,Chat Group and Chat Delete showing the system as a single high-level process, with its relationship to external entities of Chat Chat History and Chat Profile. It should be easily understood by a wide audience, including Chat Chat Profile and Chat User In zero level DFD of Online Chat Application, we have described the high level flow of the Chat Application system.
- ➔ High Level Entities and process flow of Online Chat Application:
    1. Managing all the Chat
    2. Managing all the Chat History
    3. Managing all the Smiley Chat
    4. Managing all the Chat User
    5. Managing all the Chat Group

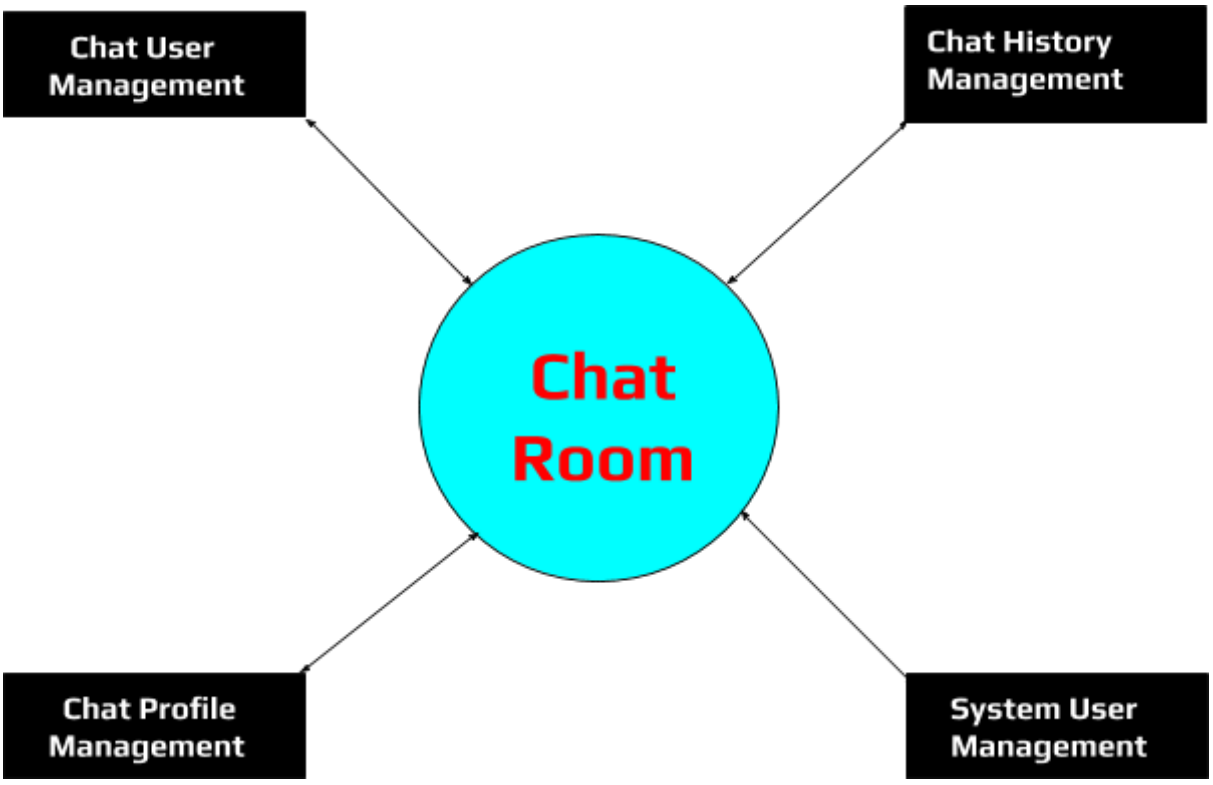


Fig 1.2.1 Zero Level DFD

1.2.2 First Level Data Flow Diagram (1st Level DFD) Of Online Chat Application :

- ➔ First Level DFD (1st Level) of Online Chat Application shows how the system is divided into subsystems (processes), each of which deals with one or more of the data flows to or from an external agent, and which together provide all of the functionality of the Online Chat Application system as a whole. It also identifies internal data stores of Chat User, Smiley Chat. DFD Level 1 provides a more detailed breakout of pieces of the 1st level DFD. You will highlight the main functionalities of Chat Application.
- ➔ Main entities and output of First Level DFD (1st Level DFD):
    1. Processing Chat records and generate report of all Chat
    2. Processing Chat History records and generate report of all Chat History
    3. Processing Smiley Chat records and generate report of all Smiley Chat
    4. Processing Chat User records and generate report of all Chat User
    5. Processing Chat Group records and generate report of all Chat Group

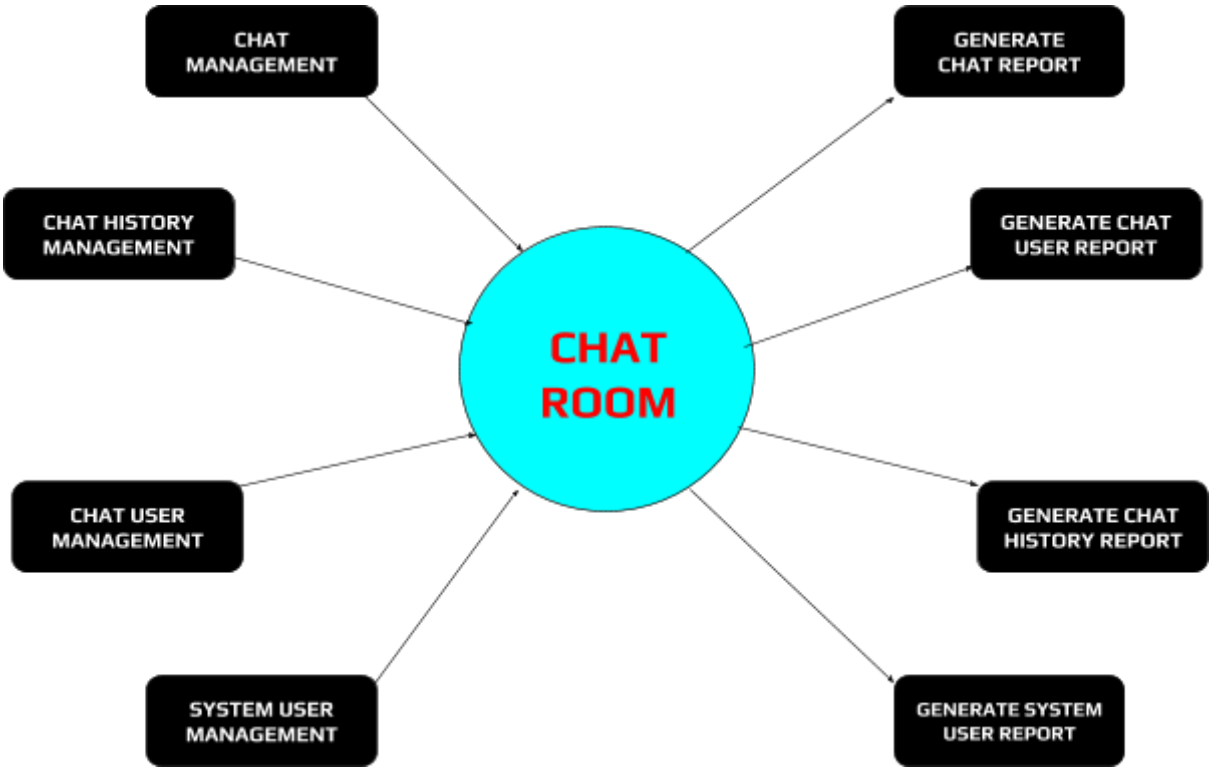


Fig 1.2.2 One Level DFD

1.2.3 Second Level Data Flow Diagram (2nd Level DFD) Of Online Chat Application :

- ➔ DFD Level 2 then goes one step deeper into parts of Level 1 of Chat Application. It may require more functionalities of Chat Application to reach the necessary level of detail about the Chat Application functioning. First Level DFD (1st Level) of Online Chat Application shows how the system is divided into subsystems (processes). The 2nd Level DFD contains more details of Chat Delete, Chat Group, Chat User, Smiley Chat, Chat Profile, Chat History, Chat.
- ➔ Low level functionalities of Online Chat Application
    1. Admin can add, edit, delete and view the records of Chat, Chat User, Chat Delete
    2. Admin can manage all the details of Chat History, Smiley Chat, Chat Group
    3. Admin can also generate reports of Chat, Chat History, Smiley Chat, Chat User, Chat Group
    4. Admin can search the details of Chat History, Chat User, Chat Group
    5. Admin can tracks the detailed information of Chat History, Smiley Chat, Chat User

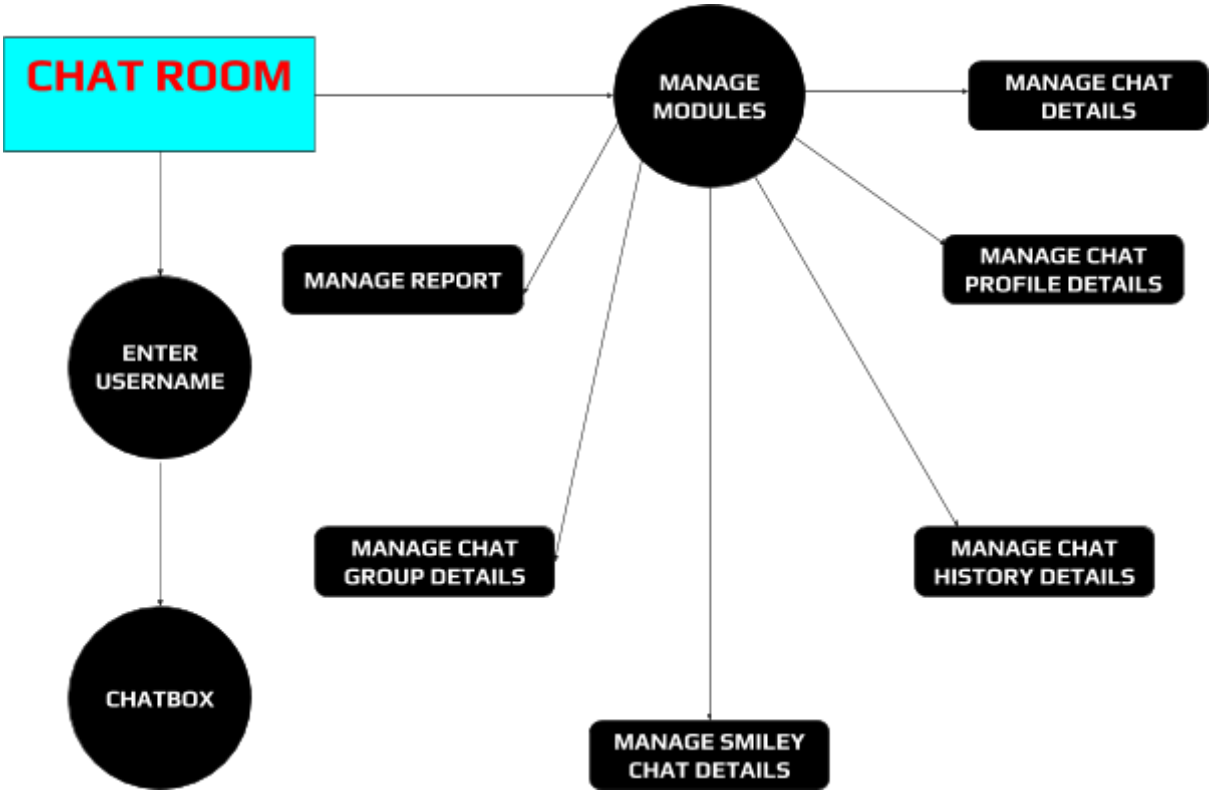


Fig 1.2.3 Two Level DFD

1.3 Application of Chat Room system

Here are some of the applications of chat room system :

1. Google Chat
2. Google Meet Chat Room
3. Google Hangouts
4. Microsoft Teams Chat Room
5. Zoom meeting Chat Room

1.4 Organization of Report

In this introduction chapter, basic concepts related to the proposed system are discussed. In Chapter 2 literature survey comparison between off-line system and web-based system are explained. Chapter 3 Problem Statement provides a detailed problem statement of the proposed system. In Chapter 4 System Analysis different types of requirements of the proposed system are documented. Chapter 5 Design consists of User Interface Design and Database Design of the proposed System. Chapter 6 Result and Discussion consists of various snapshots of the proposed system. Chapter 7 Conclusion consists of the conclusion of the proposed system.

CHAPTER 2

LITERATURE SURVEY

2.1 Brief History of Off-Line Computerized System or existing system similar to your chosen system

Communication saw new developments between ancient times and the 15th century. While it is interesting, we've all got things to do. So I'll keep it brief: For the most part, communication before the 15th century included verbal communication and limited written communication. People spoke to each other and wrote handwritten notes for an individual or a group.

This all changed after the printing press was invented for removing the need to hand write all paper messages.

In the 18th century, the public postal system was established. Letters were sent before this time period, but the system was flawed and inefficient. Delivery could take months, and even when the mail did reach its destination, it was left at a random public place

As the 19th century came around, the great minds working towards improving communication took a large step. Electricity was incorporated into communication channels, coining the term telecommunications.

Telecommunications provided people with a way to make long distance written and verbal communication personal. Distance became less of a problem for people who wanted to communicate verbally, increasing the flow of information.

Radio and television became a great source of information in the early 1900s. Not only did they provide musical entertainment, but they also delivered news, sports, and weather to listeners.

We all know that email and PCs were just the beginning of the most important innovations in communication. But these origins can't be forgotten. Every communication tool used today was somehow influenced by simple ones from the long and never-ending history of communication.

21st century communication. When thinking of modern day communication, keeping the idea of the Information Age in mind is important. The Information Age is characterized as the economic transition from industry to information technology. Simply put, everything is digital and knowledge has never been more powerful. People want information, and the best way to get it is through communication tools.

2.2 Comparison of Chat Room System with Previous system

Sr. No.	Basis	Chat Room	Google chat	Google Meet Chat Room	Microsoft Teams Chat Room	Zoom meeting chatroom
1	Number of participants	Limitless	150	250	250	100
2	Number of characters	255	160	4000	1000	128
3	Cost	Free	Paid	Free Trial	Free Trial	Paid
4	Anonymity	Yes	No	No	No	No
5	Launch Year	2020	2017	2020	2017	2013

Table 2.1 Comparison of Chat Room System with Previous system

CHAPTER 3

PROBLEM STATEMENT

3.1 Problem Statement -

Develop an application which facilitates the creation of a chat room with a live server for the users to enable sharing messages or chat on the go. Develop an instant messaging solution to enable users to seamlessly communicate with each other and also which can be used by any novice user. I.e. Live chat room on the fly (online).

3.2 Problem Solution -

Chat Room as a service is a model of communication deployment where the server hosts a live chat room as a service for users on the Internet. Users are admissible to enter the chat room and share messages or interact with each other.

This project is a real-time communication system provided in the form of a chat room. It is a simple Web-Based Chat Application using PHP, MySQL, javascript, ajax. The project has been created keeping in mind the fact that the anonymity of the users will not be compromised under any circumstances. Our goal is to create a way to use technology to bring people and ideas together despite geographical barriers. The technology has been available for years but it has only recently been acquired.

3.3 Project scope and features -

1. Chat Server Application is going to be a text communication software, it will be able to communicate between two computers using point to point communication.
2. Anonymity is the key feature of this project.
3. The limitation of our project is that it doesn't support audio conversations.
4. Companies would like to have a communication software wherein they can communicate instantly within their organization.
5. The fact that software uses an internal network within the organization makes it very secure from outside attacks.

CHAPTER 4

SYSTEM ANALYSIS

4.1 System Requirements

To be used efficiently, all computer software needs certain hardware components or other software resources to be present on a computer. These prerequisites are known as (computer) system requirements and are often used as a guideline as opposed to an absolute rule. Most software defines two sets of system requirements: minimum and recommended. With increasing demand for higher processing power and resources in newer versions of software, system requirements tend to increase over time.

Various categories are:

1. Hardware Requirements
2. Software Requirements
3. Functional Requirements
4. Non-functional Requirements

4.1.1 Hardware Requirements :

Input devices:

1. Keyboard
2. Mouse
3. Monitor
4. Wireless Peripherals
5. Smartphone

4.1.2 Software Requirements:

1. **Xampp:**
   - **Apache:**
     - (Application Server) Apache , often referred to as Server, is an open-source Java Servlet Container developed by the Apache Software Foundation.
   - **MySqlServer:**
     - It handles large databases much faster than existing solutions.
     - It consists of multi-threaded SQL server that supports different back ends, several different client programs and libraries, administrative tools, and application programming interfaces (APIs)
     - Its connectivity, speed, and security make MySQL Server highly suited for accessing databases on the Internet.
2. **Sublime Text:** Sublime Text is a sophisticated text editor for code, markup and prose. You'll love the slick user interface, extraordinary features and amazing performance.
3. **Web browsers:** Google Chrome,Mozilla Firefox, Opera and Internet Explorer.
4. **Github:** GitHub Inc. is a web-based hosting service for version control using Git. It is mostly used for computer code. It offers all of the distributed version control and source code management functionality of Git as well as adding its own features.

4.1.3 Functional Requirements:

Functional requirements are those requirements that are used to illustrate the internal working nature of the system, the description of the system, and explanation of each subsystem. It consists of what task the system should perform, the processes involved, which data should the system hold and the interfaces with the user.

1. **Sending message:** After the text is sent or received, the user will be notified with a notification tone that will be played.
2. **Broadcast message:** The user will be able to see the timestamp sent or received next to the message.

4.1.4 Non-Functional Requirements

It describes aspects of the system that are concerned with how the system provides the functional requirements. They are -

- ➔ **Security -** The subsystem should provide a high level of security and integrity of the data held by the system, only authorized personnel of the company can gain access to the database where all the chats and user id's will be saved and only the valid users can access the database.

- ➔ **Performance and Response time -** The system should have a high performance rate when executing users input and should be able to provide feedback or response within a short time span usually 30 seconds for highly complicated tasks and 15 to 25 seconds for less complicated tasks.

- ➔ **Error handling -** Error should be considerably minimized and an appropriate error message that guides the user to recover from an error should be provided. Validation of users input is highly essential. Also the standard time taken to recover from an error should be 15 to 20 seconds.

- ➔ **Availability -** This system should always be available for access at 24 hours, 7 days a week. Also in the occurrence of any major system malfunctioning, the system should be available in 1 to 2 working days, so that the communication process isn't affected by any delay.

- ➔ **Ease of use -** Considering the level of knowledge possessed by the users of this system, a simple but quality user interface should be developed to make it easy to understand and require less training,

CHAPTER 5

DESIGN

5.1 Design Process:

User interface (UI) design is the process designers use to build interfaces in software or computerized devices, focusing on looks or style. Designers aim to create interfaces which users find easy to use and pleasurable. UI design refers to graphical user interfaces and other forms.

5.2 System Layout:

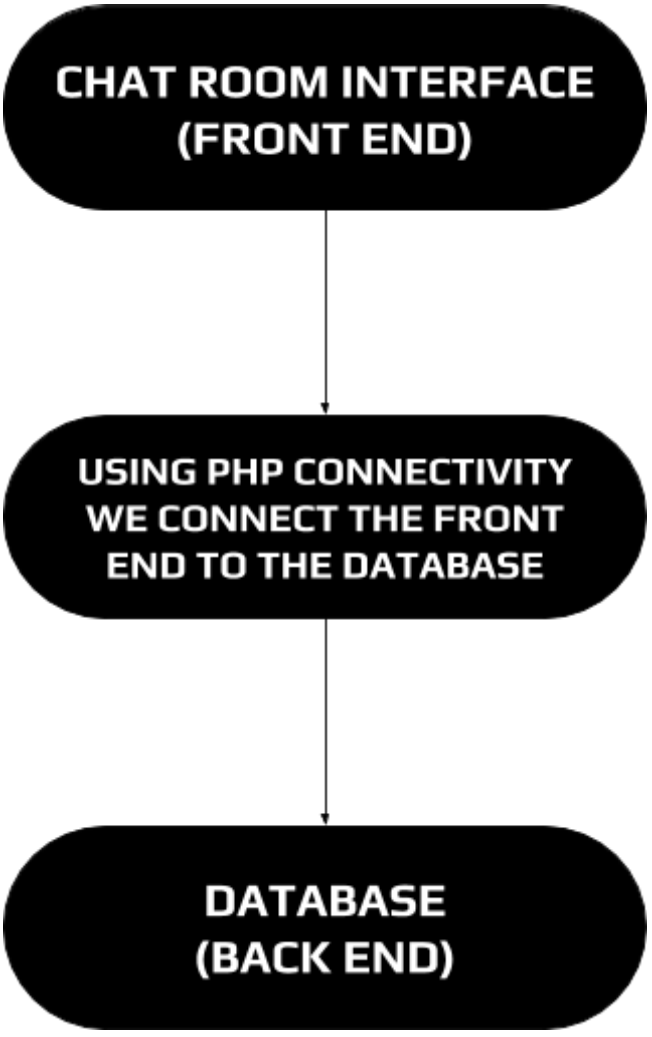
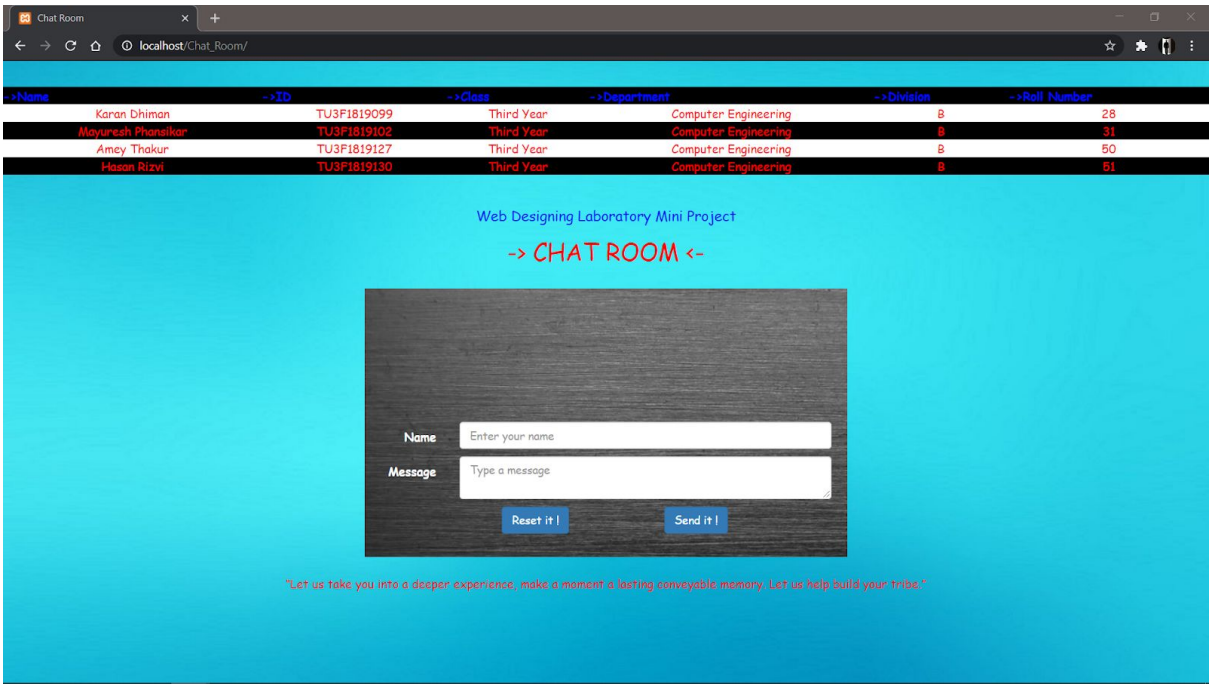


Fig 5.2 System Layout Architecture

5.3 User Interface Design:



5.4 Database Design:

You can think of a database as a sophisticated, digital filing cabinet. It's what will help us organize all of our app's data. We can add, update, and remove pieces of data from our database, at will.

Databases are tables, which you can think of as file folders. Tables are records, or rows, which you can think of as individual documents within one of the file folders, We can add new records to that table each time a bill is paid, just as we could have added new pieces of paper to our file folder. Our records can contain various pieces of information, such as the bill number, amount, and date paid.The database will give us the ability to collect, store, manipulate, and display data throughout our web app. have the ability to retrieve information that has been added to our database.

A query is a command that we can send to our database, which tells it to perform certain operations.

CHAPTER 6

RESULT AND DISCUSSIONS

6.1 Input/Program

➢ index.php

```
<?php
include 'db.php';
?>

<!DOCTYPE html>
<html>
        <head>
                <meta charset="utf-8">
                <meta name="viewport" content="width=device-width">
                <meta name="description" content="form">
                <meta name="keywords" content="Chat Room">
                <meta name="author" content="Amey Thakur">
                <title>Chat Room</title>

<link rel="stylesheet"
href="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">

<script
src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>

<script
src="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js">
</script>

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

        </head>

<body onload= "ajax(); ">
<br>
 <aside id="sidebar">
          <p>
          <table style="width:100%;text-align:center;border-collapse:collapse;">
          <tr style="background-color:#000000;color:#000dff;text-align:center;">
```

```
        <th>->Name</th>
        <th>->ID</th>
        <th>->Class</th>
        <th>->Department</th>
        <th>->Division</th>
        <th>->Roll Number</th>
</tr>

<tr style="background-color:#ffffff;color:#ff0000;text-align:center;">
        <td>Karan Dhiman</td>
        <td>TU3F1819099</td>
        <td>Third Year</td>
        <td>Computer Engineering</td>
        <td>B</td>
        <td>28</td>
</tr>

<tr style="background-color:#000000;color:#ff0000;text-align:center;">
        <td>Mayuresh Phansikar</td>
        <td>TU3F1819102</td>
        <td>Third Year</td>
        <td>Computer Engineering</td>
        <td>B</td>
        <td>31</td>
</tr>

<tr style="background-color:#ffffff;color:#ff0000;text-align:center;">
        <td>Amey Thakur</td>
        <td>TU3F1819127</td>
        <td>Third Year</td>
        <td>Computer Engineering</td>
        <td>B</td>
        <td>50</td>
</tr>

<tr style="background-color:#000000;color:#ff0000;text-align:center;">
        <td>Hasan Rizvi</td>
        <td>TU3F1819130</td>
        <td>Third Year</td>
        <td>Computer Engineering</td>
        <td>B</td>
        <td>51</td>
</tr>
```

```
        </table>
        </p>
</aside>
<br>
<center>
    <p> <h4 style="color:#000dff">Web Designing Laboratory Mini Project</h4></p>
    <p> <h2 style="color:#ff0000">-> CHAT ROOM <-</h2></p>
</center>

<div class="container" style="" >
        <div id="chat_box">
                <div id="chat">
                </div>

<form method = "post" action="index.php" class="form-horizontal"
style="margin-top:150px;">
        <div class="form-group">
        <label for="inputEmail3" class="col-sm-2 control-label">Name</label>
        <div class="col-sm-10">
        <input type="text" class="form-control" id="" placeholder="Enter your name"
        name ="username">
        </div>
</div>

<div class="form-group">
        <label for="comment" class="col-sm-2 control-label">Message</label>
        <div class = "col-sm-10">
                <textarea name = "message" class="form-control" rows="2"
                id="comment" placeholder="Type a message"></textarea>
        </div>
</div>

<div class="form-group">
        <div class="col-sm-offset-2 col-sm-10">
                <td colspan="2" align="center">
                           
                <button type="reset" name = "reset" class="btn btn-primary">
                Reset it !</button>
                                   
                     
                <button type="submit" name = "submit"
                class="btn btn-primary">Send it !</button></td>
        </div>
```

```
    </div>
</form>

<?php
if (isset($_POST['username']) && isset($_POST['message']))
{
    $name = $_POST['username'];
    $message = $_POST['message'];
    $query_1 = "INSERT INTO chat_info (name,msg) VALUES ('$name','$message')";
    $query_run = mysqli_query($con, $query_1);

    if ($query_run)
    {
        echo "<audio src = 'sound/134332-facebook-chat-sound.mp3' hidden = 'true'
autoplay = 'true' /></audio>";
    }
}
?>
        </div>
</div>
<br>

<footer>
        <center><p style="color:#f70000">
                "Let us take you into a deeper experience, make a moment a lasting
                conveyable memory. Let us help build your tribe."
        </p></center>
</footer>
</body>
        <script src="script.js"></script>
</html>
```

➢ db.php

```
<?php
        $host = "localhost";
        $user = "root";
        $pass = "";
        $db_name = "chat_info";
        $con = new mysqli($host,$user,$pass,$db_name);

        function formatDate($date)
        {
```

```
            return date('g:i a',strtotime($date));
        }
?>
```

➢ chat.php

```
<?php
        require 'db.php';

        $query = "SELECT * FROM chat_info ORDER BY id DESC";
        $query_run   = mysqli_query($con,$query);

        while($query_row = mysqli_fetch_assoc($query_run)):
?>

<div id ="chat_data">
</div>
        <span style="color:red;"><?php echo $query_row['name'].' : '; ?></span>
        <span style="font-family:cursive;"><?php echo $query_row['msg']; ?></span>
        <span style = "font-family:cursive;float:right;">
        <?php echo formatDate($query_row['date']); ?></span>
<?php endwhile; ?>
```

➢ script.js

```
function ajax()
{
        $.ajax
        (
                {
                        url: "chat.php", success: function(result)
                        {
                                $("#chat").html(result);
                        }
                }
        );
}
setInterval
(
        function()
        {
                ajax();
        },
```

```
        1000
);
```

➢ stylesheet.css

```
.container
{
        background: url('images/black.jpg');
        padding: 20px;
        width: 40%;
        margin-top: 2%;
}

span
{
   color: white;
}

#chat_box
{
        height:100%;
        margin-top: 4px;
}

.form-group
{
        margin-bottom: 10px;
        margin-top: 5px;
        color: white;
}


body
{
        background: url('images/blue.jpg');
        font: 15px/1.5 cursive;
}
```

➢ Database: chat_info >> Table: chat_info

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	int(11)			No	*None*		AUTO_INCREMENT
2	name	varchar(255)	latin1_swedish_ci		No	*None*		
3	msg	varchar(255)	latin1_swedish_ci		No	*None*		
4	date	timestamp			No	current_timestamp()		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	0	A	No	

Partitions

No partitioning defined!

Information

Data	48	B
Index	2.0	KiB
Overhead	48	B
Effective	2.0	KiB
Total	2.0	KiB

Space usage

Format	dynamic
Collation	latin1_swedish_ci
Rows	0
Next autoindex	32
Creation	Sep 13, 2020 at 05:23 PM
Last update	Sep 13, 2020 at 05:28 PM
Last check	Sep 13, 2020 at 07:56 PM

Row statistics

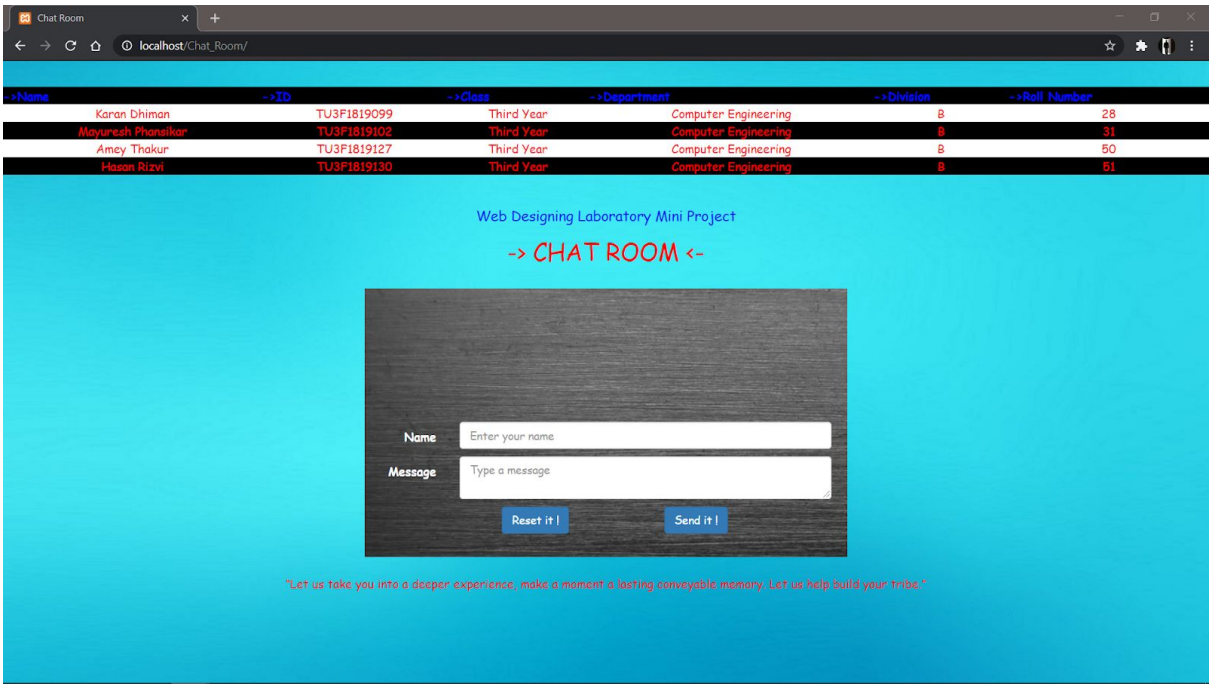
➢ Database > Table > Browse

```
SELECT * FROM `chat_info`
```

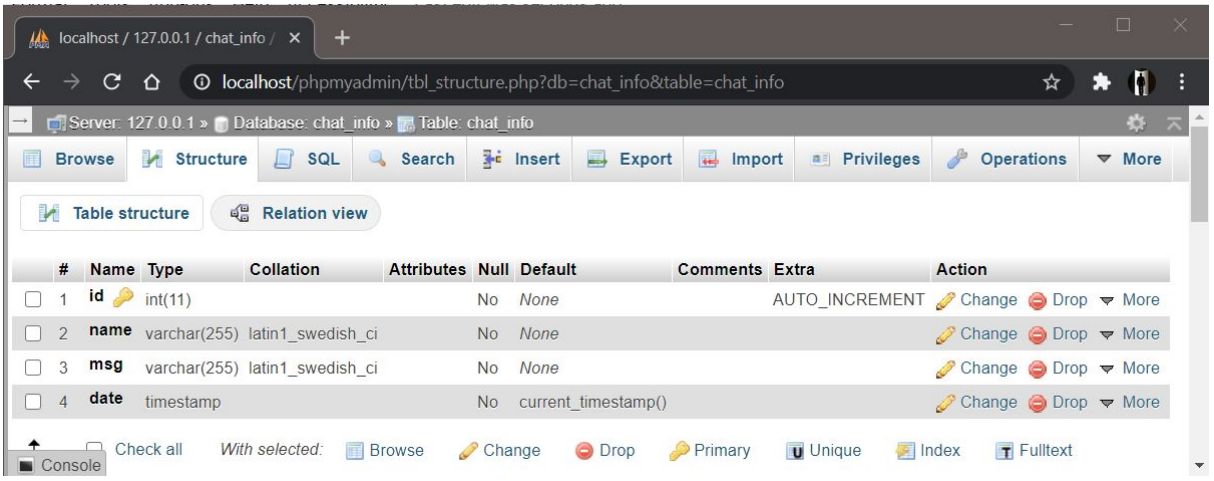
id	name	msg	date

6.2 Snapshots

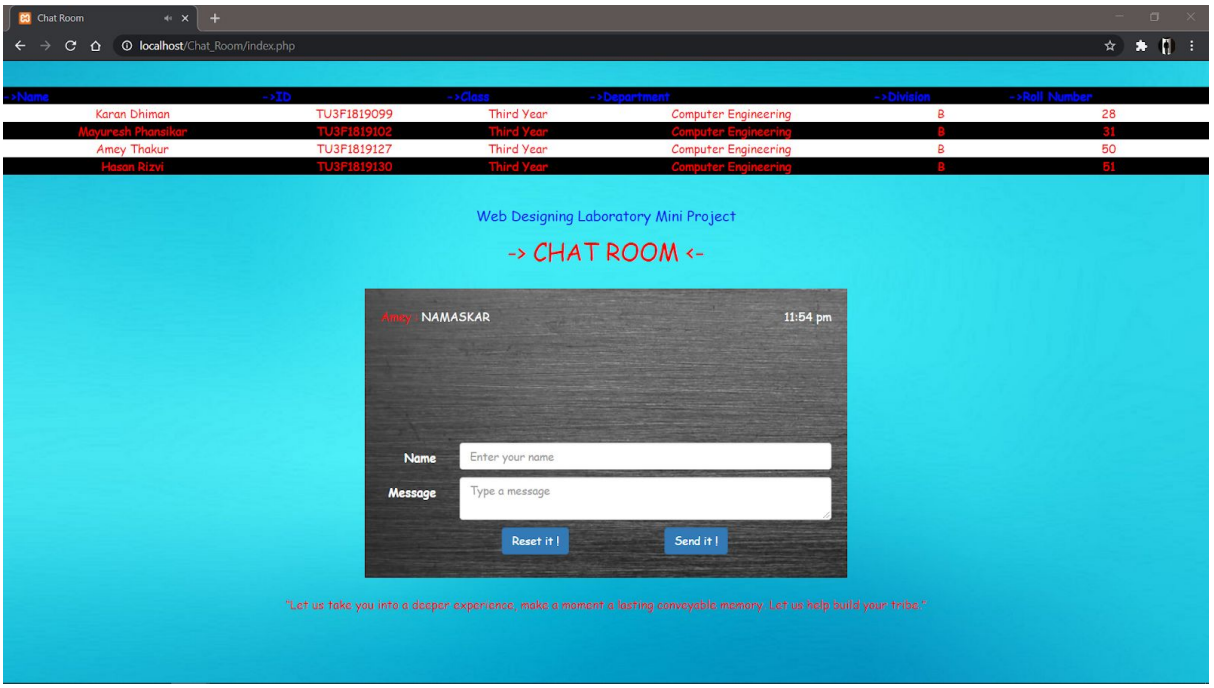
➢ Chat Room



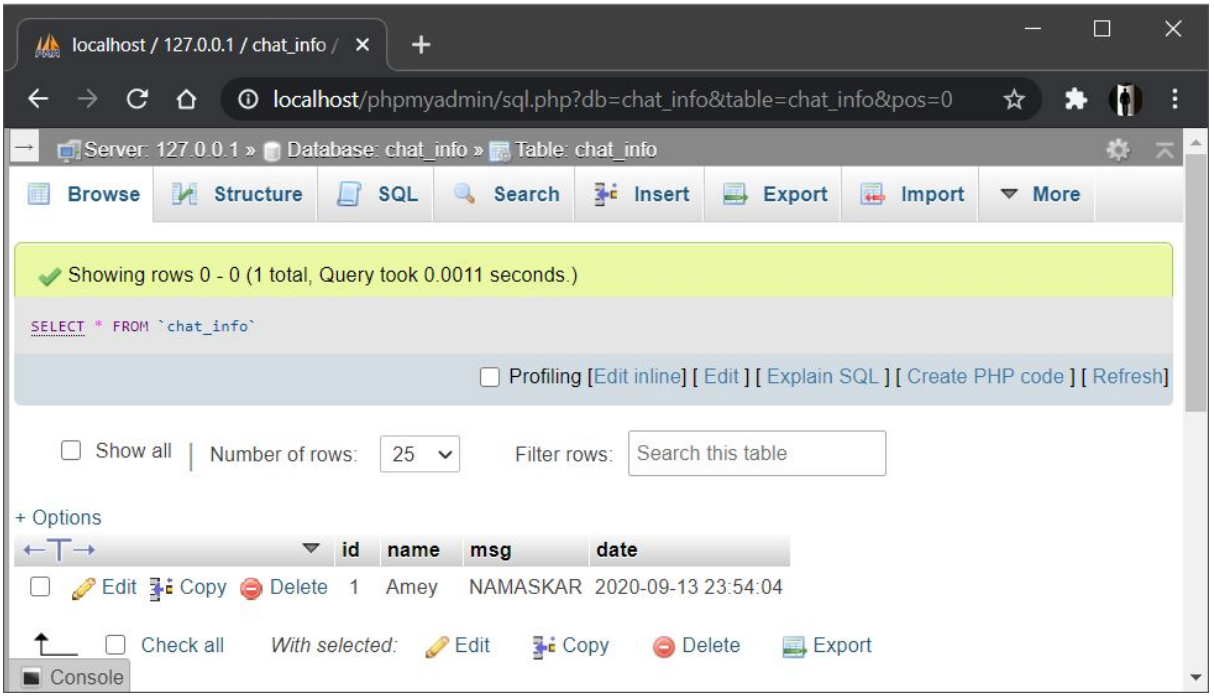
➢ Database > Table > Structure



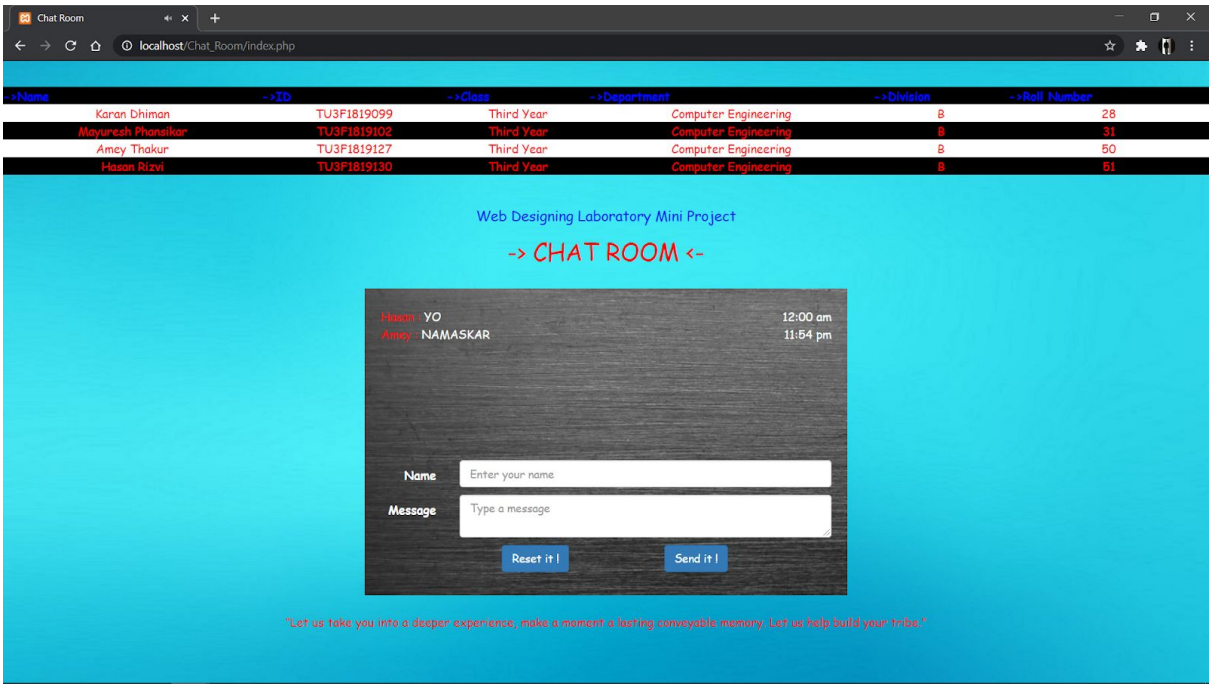
➢ User 1 > Amey > Message



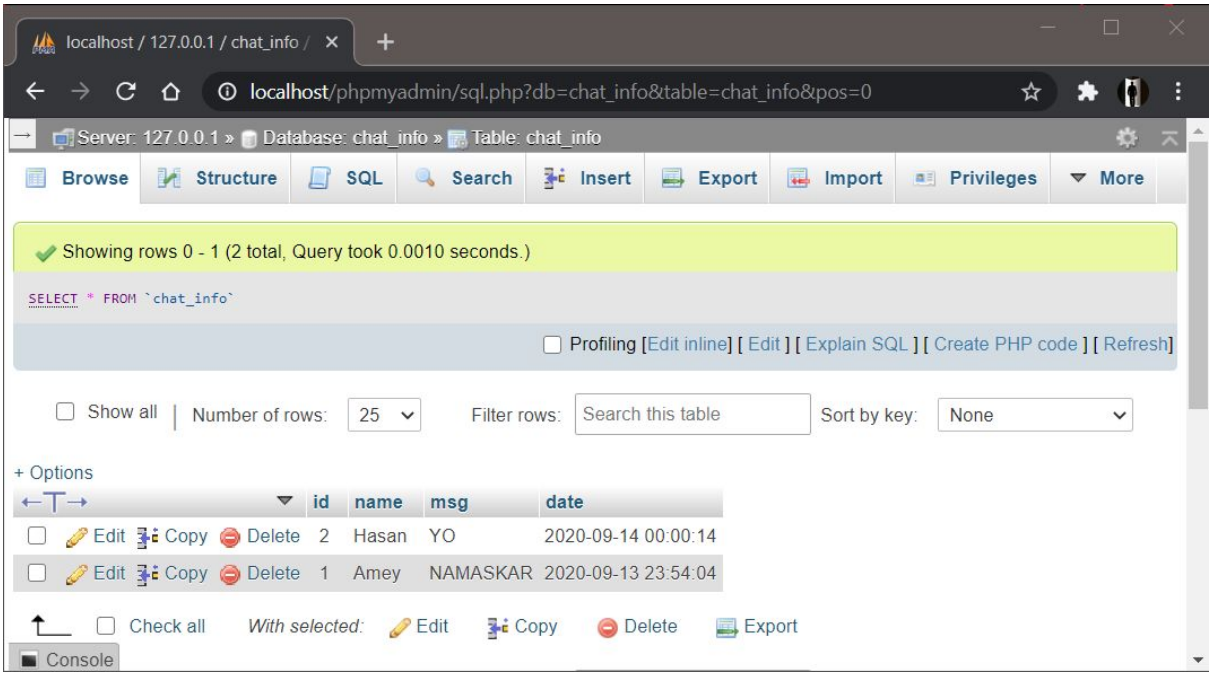
➢ User 1 > > Message >> Database



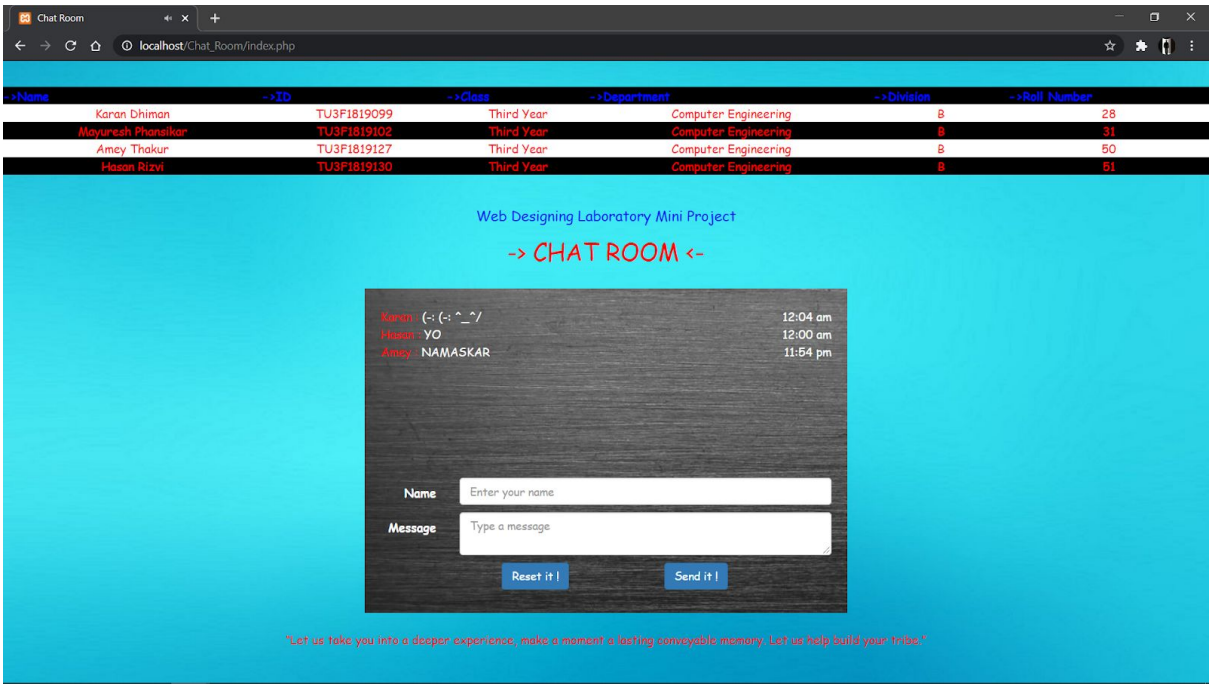
➢ User 2 > Hasan > Message



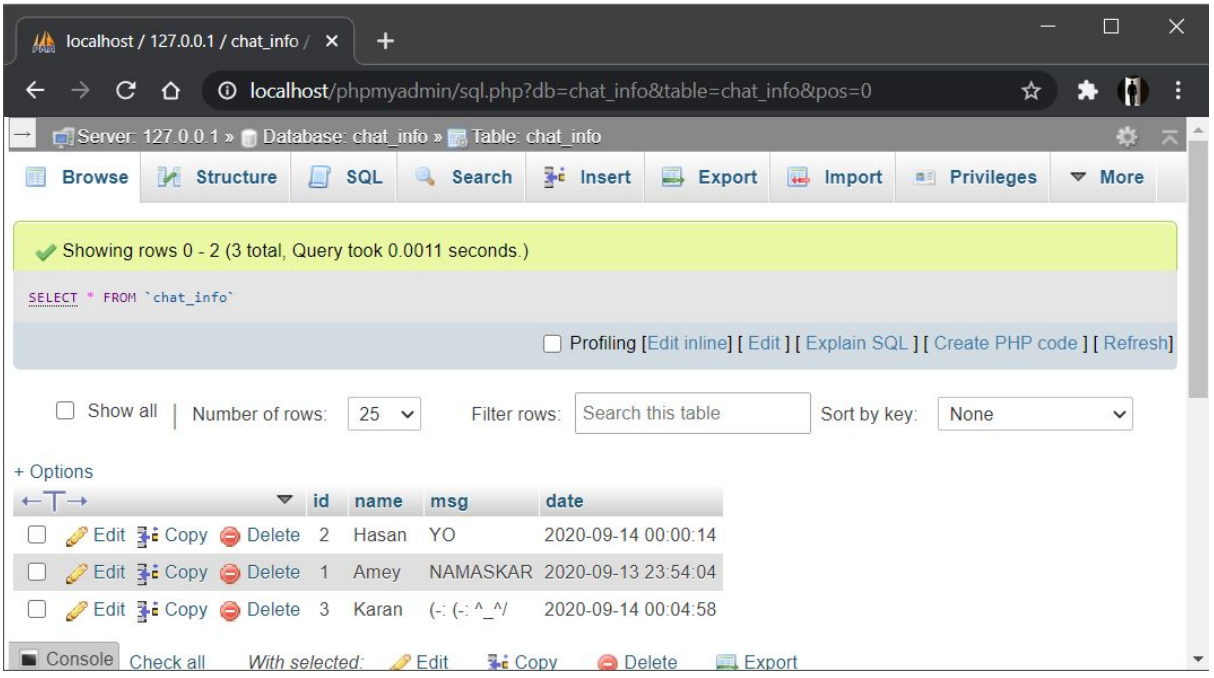
➢ User 2 > > Message >> Database



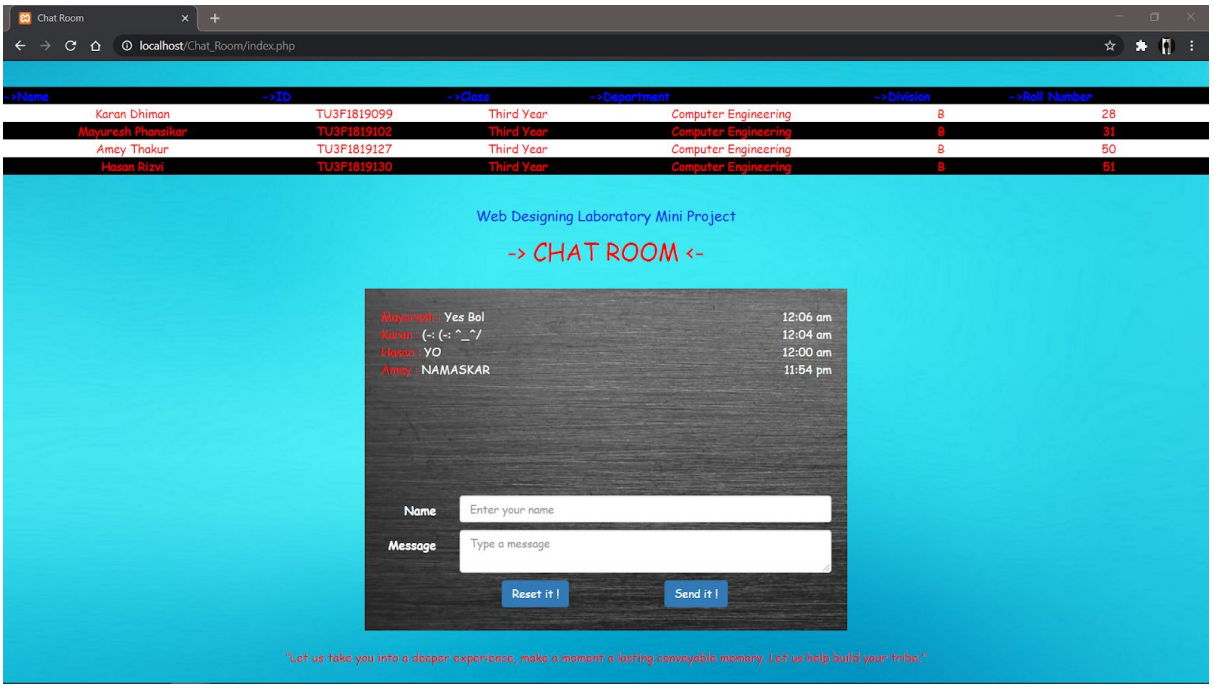
➢ User 3 > Karan > Message



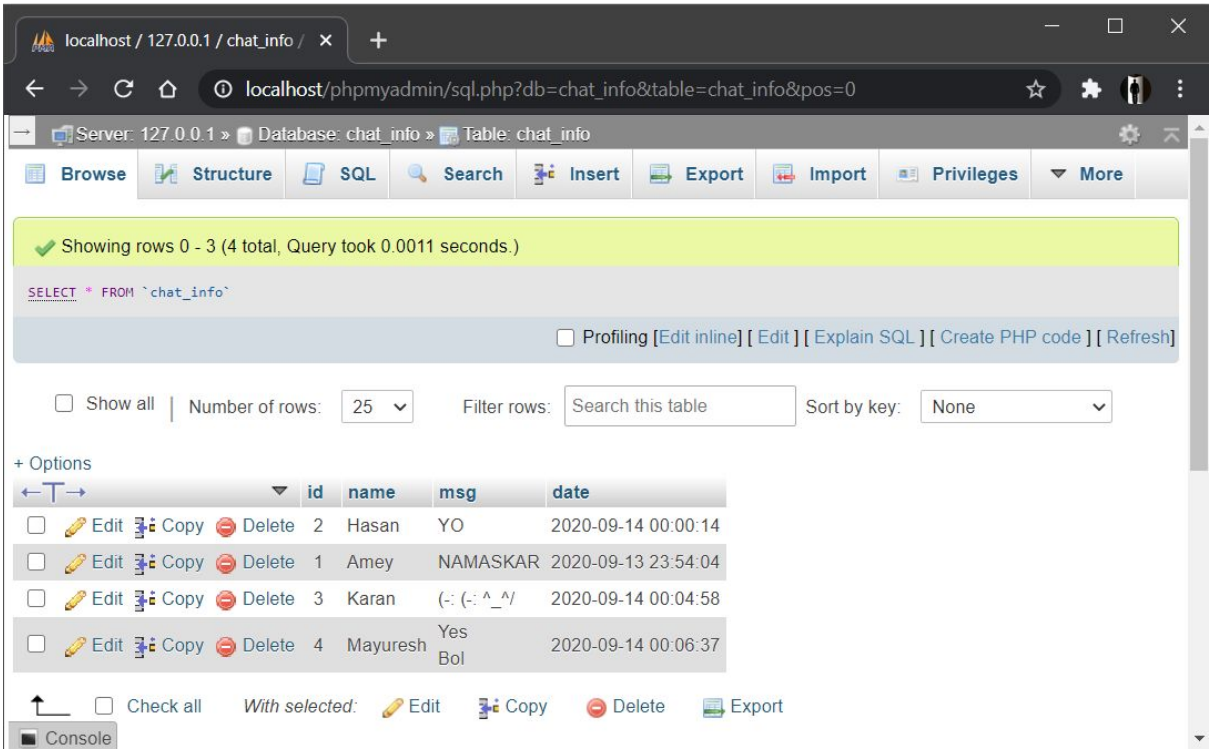
➢ User 3 > > Message >> Database



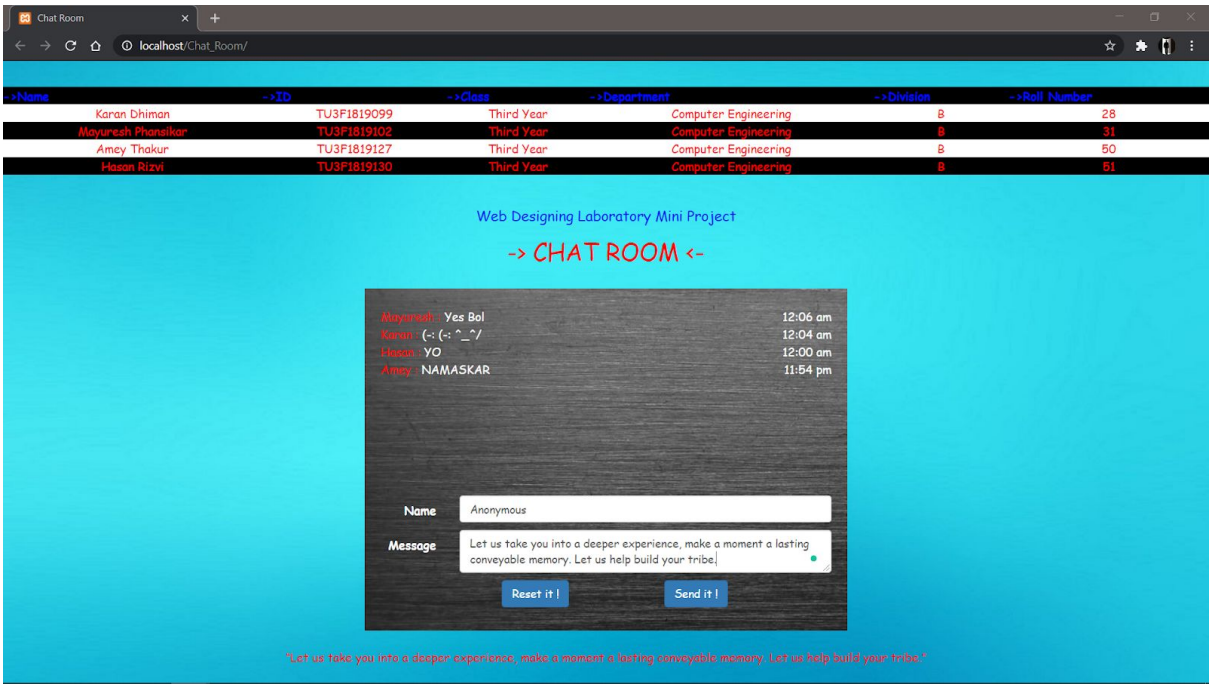
➢ User 4 > Mayuresh > Message

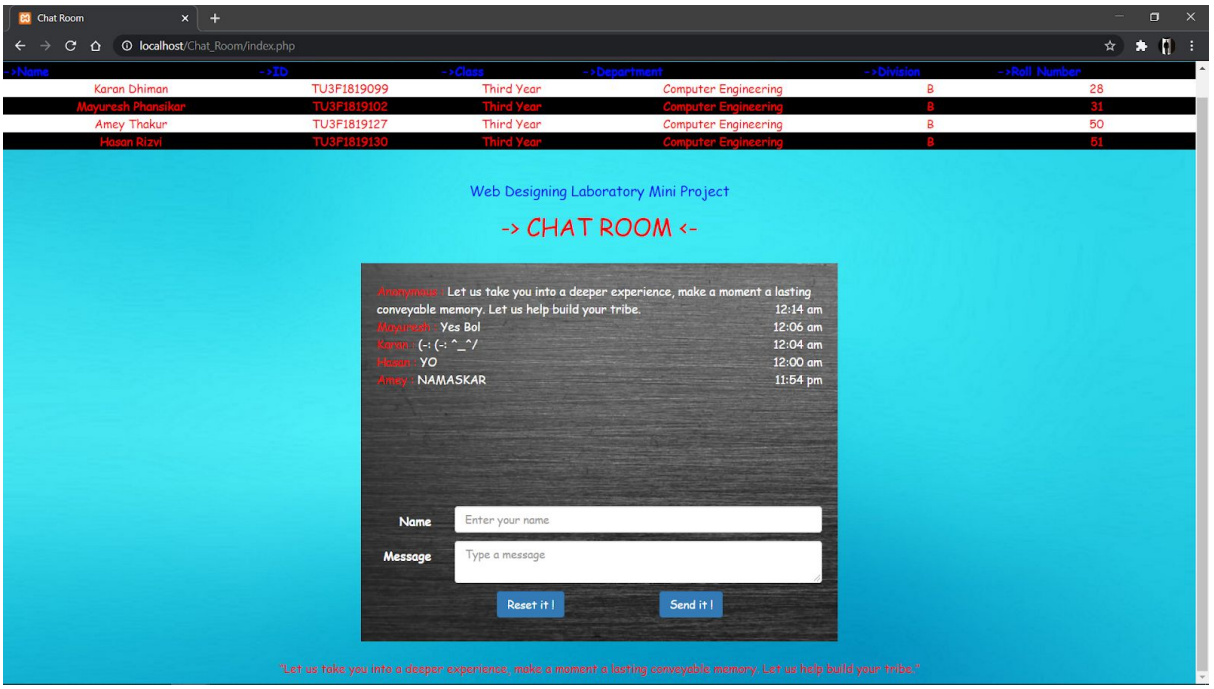


➢ User 4 > > Message >> Database



➢ User > Anonymous > Message





CHAPTER 8

CONCLUSION

The objective of Chat Room is achieved by providing an extremely rich communication experience. We've tried to make the UI crisp and clean with no rigorous and over the top additions. Flexibility in design makes users explore their imagination and thus, even novice users can design useful web pages.

Here, we have built Simple Web-Based Chat Application using php, mysql, javascript, ajax. There's always room for improvements in any product, and we have tried to adapt the design accordingly, while also keeping track of our limitations. We have encountered a bunch of problems and learnt how to research our way out of those, during the course of building this application.

We hope our idea was clear and well presented with the final output for it.

REFERENCES

Documentation:
https://getbootstrap.com/docs/4.1/getting-started/introduction/
https://api.jquery.com/category/ajax/

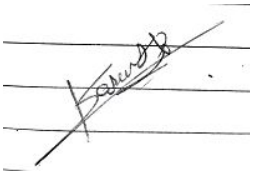
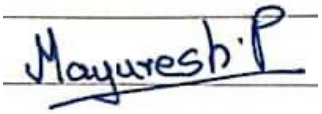
References:
https://www.w3schools.com/php/DEFAULT.asp
https://www.w3schools.com/js/js_ajax_intro.asp
https://www.w3schools.com/js/
https://www.w3schools.com/w3css/
https://www.w3schools.com/bootstrap/
https://www.w3schools.com/jquery/

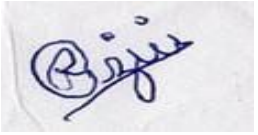
Software:
https://atom.io/
https://www.sublimetext.com/

Repository:
https://github.com/Amey-Thakur/Chat-Room

The project has done by -

Terna Engineering College
TE COMPS B - 28, 31, 50, 51



Karan Dhiman **Mayuresh Phansikar** **Amey Thakur** **Hasan Rizvi**