DESIGN & IMPLEMENTATION OF REAL TIME CHAT APPLICATION

Rahul Khandelwal*1, Dishant Solanki*2, Teena Verma*3

*1,2 Student, B.Tech(Computer Science Engineering), Hmritm, Hamidpur, Delhi, India.
*3 Assistant Professor, Computer Science Engineering, Hmritm, Hamidpur, Delhi, India.

ABSTRACT

Chat application is an important tool for today’s world as it helps everyone to contact from anywhere and talk with them at real time without any problem and this tool resolve the issue of calling every time for any query related anything about that work. Now with the help of chat application user can just send the message to the other user and resolve the issue or query without calling. The best part about the chat application is that we can send the same message to many users at a same time without any problem as we can make a group of so many people or even we can broadcast the message to selected users to whom the sender wishes to send the message. Chat application also provided us the best feature in this as you can share your current location with anyone at any time and even you can share the images or pdf or any other type of file using the application.

Keywords: Real Time Chat, Multiple-Platform, Multiple-User, Node.Js, Socket.Io, Chat Application.

I. INTRODUCTION

As the time going on, the development of information and communication technology is growing so faster and playing very important role in the society. As with the help of the information and communication technologies we can facilitate the people to receive the information and communicate with each other from anywhere, anytime and faster than before. If we see the today’s scenario after the technology have been arrived most of the people don’t use newspaper for getting the information and even not much uses the news channel now a days for getting information, Majority is using the smart technology for receiving any type of information with the help of smart phones or other devices and if we talk about the chat application, in the chat application everyone is sharing many news there itself only about what's going on around them or in any place in the world, as with the help of the chat application information is spreading so faster than before. Currently, the chat application has been developed by many developers and it has its own advantages and disadvantages as every other thing have, and it's giving us more options to decide which thing is better for us and what features are really required in the application and what not, as the review of the users using the application tells a lot about the application and their experience towards the application. The Chat application is fully responsive as it means totally user friendly for every web-based or mobile or any other device users. It is built using Node.js, Socket.io, MongoDB, Passport.js, Auth2.0 and other JavaScript libraries.

Node.js is a software platform that is used to build server side more flexible for applications in a network application. Socket.IO is a JavaScript library and it is an implementation of web sockets protocol and various other required improvisation for real-time chat application. JavaScript is a scripting language that is used to create a program so the HTML document displayed in the browser becomes more interactive and impressive for users. MongoDB is an open source NOSQL database based on Document-Oriented Database program, which was originally created in C ++. Therefore, It's expected that this chat application can run easily on real time.

II. MAIN PROBLEM

Based on the title above, the problem is how to design and implement responsive web-based or mobile or other devices chat applications in real time to make communication becomes easier and faster.

III. OBJECTIVE

The objective can be defined from the main problem given above is that to build a proper real-time multiplatform chat application which can be easily used by people to make their life easy to share the information and communicate with each other easily and faster.

IV. SCOPE

The scope of application should be declared before moving towards the next step. Application scope are as follows:
1. Design and construction of this chat application is responsive for every web-based or mobile or other device users.
2. This application is developed using Node.js, socket.io and other JavaScript libraries.
3. Database of the application is stored using MongoDB NOSQL Database.
4. Used by less than 100 people (for testing purpose).

V. SYSTEM DESIGN

The application is built in two sides one is client and other is server. Client and Server connected bidirectional, when the client sends the message it goes to the server and when the server receives that message, server cannot do anything it could just print on terminal or do something else like dump it file into, but what we want to do is that make sure that everyone is connected to the chat room actually sees the message that this person has sent and at which time. We will bring other clients into mix and this time we send data from server to client.

Server will send the client1 message to other clients that client 1 has sent the message, server will send that across to client and client can render that message through browser. In short when any client send the message it goes to server and then after receiving the message server sends it to all the other clients present in the room at that time.

A. Block Diagram

Here is a block diagram of server – client communication

![Block Diagram]

B. Flow Chart

Chat application overall process is shown with the help of flow chart below for more clarity.
The explanation for each flow from picture above is as follow:

1. When the user enter into the chat application, user login credentials is required. If the user doesn't have an account, then the user should signup first, In both login and signup page user have option to sign up using Google or Facebook account.

2. After login, user have to enter the room name in which user can start a chat. Therefore, if user doesn’t have the room name, user can create its own room name and invite their friends to chat with them in that room. After entering the room name user will be redirected to chat page.

3. There are two sides in the chat application in left side sidebar is given where user can see the room name in which they have joined and they can also see the connected online user at that time in chat room. Also, there is an option to join other room and option to logout.

C. Data Flow Diagram

Here is a data flow diagram to describe overall process of chat application data.
D. Database Table Data Structure

Here is database table data structure for each table user and chatroom.

<table>
<thead>
<tr>
<th>Table 1 - User Data Structure</th>
</tr>
</thead>
<tbody>
<tr>
<td>No</td>
</tr>
<tr>
<td>----</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>6</td>
</tr>
<tr>
<td>7</td>
</tr>
<tr>
<td>8</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 2 - Join Room Data Structure</th>
</tr>
</thead>
<tbody>
<tr>
<td>No</td>
</tr>
<tr>
<td>----</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
</tbody>
</table>

VI. IMPLEMENTATION

E. Login Page(Home Page)

If the user have already account they can login using their username and password credentials and user can login their account via google or facebook account, for login using google account select that google account when it redirects to the selecting of google and same with facebook login.

![Login Page(Home Page)](image)

F. Sign up Page

User can create here the account with their user credentials e.g. email, name, password. Other option also given there to sign up using google or facebook account, and while signing up using anyone from these select the account from which user want to sign up into chat application.
G. Join Room Page
User can join any room which they like they just need to decide the room name so their friends also can join the same room.

H. Chat Page
Chat application is totally responsive and made for every device so every user can use the application easily.
If we talk about the sidebar you can see that the sidebar showing us information about the room name, showing us all the current active members online right now and there are two options available except that is you can join other join by just click on that, it will redirect you to join room page & you can logout your account whenever you want but if you don’t logout your account when you comeback you don't need to login again as express-session is added into this application which will store the data for you. In sidebar I have used icons also as you can see that when sidebar is open its showing the information but if I close the sidebar it will be shown like this see below.

Now Let’s talk about the chat message feature, When any user joins the chat they receive welcome message from server and when any other user joins the chat or leave the chat other user get notified.

Fig 8: Responsive Chat App

Fig 9: Closed Sidebar

Fig 10: When other user joining the chat
Fig 11: Other user leave the chat

Now let's talk about the chat message feature, when the user sends any message to the chat, it will be shown at the right side and for the other online users it will shown on the left side as what happens in every other chat application, from this way it's easy to understand that right side all messages are from user side and left one from other users. See below in the Screen Shot and yes one thing when the user sends the message, all the other users who are present in the chat at that time will get ding sound as the notification of the message, & we can share our current location also in the chat. See below

Fig 12 & 13: Users sends a broadcast message

I have added the feature of auto scrolling into the chat so the users can see the latest message easily without scrolling the chat and the user sends the message that automatically comes down with the help of auto scrolling. Now, will talk about last function of this chat app and the most important one for me. The function is if the both users are connected to same room chat then only they both can see other user message else they cannot, showed in below
As we can see that browser 1&2 have joined the same room and the 3rd browser joined the different room that's why when any user from browser 1&2 sending message that browser 3rd not receiving those and same with browser 3rd when that user sending message browser 1&2 cannot receive that.

VII. CONCLUSION

We've seen that making "Real Time Chat Application" is quite interesting thing and new thing to make other than making a website about a product or any portfolio. As, I can say that socket.io made chat application easy to build and develop our own chat application. I can say that because of this project I got to learn a lot more than I expected. In Today's world chat application is in much demand & needed around everywhere and its reducing our much problem and time, as I told above with the help of the chat application we can just send the same message to many users at a time and we don’t need to call every time to anyone for any type of queries or problem, we can just message them. At last I would like to say that making chat application using Node.js & Socket.io is best for mainly 2 thing, first is reducing time by just texting in some seconds and second one is you will learn a lot while making this app and till completing this application you have good hands on this technology.

VIII. REFERENCE

