

# Web Chat using React Framework

Akhilesh Sarjit M S, Srivishak V, Shiddarth S, Saravana Kumar P, Preethi D

Department of Electronics and Communication Engineering,  
Bannari Amman Institute of Technology, Erode, Tamil Nadu, India

## ABSTRACT

With the development and enhancement in internet, more and more people have been choosing network chatting tools for communication. Applications such as these facilitates communication over great distances. Therefore, this application must both be real-time and multi-platform to be used by many users. The web-based real-time chatting application does not need any additional third-party client program, and the visual communication could be established conveniently. The programming tools used in building this application is React.js, Node.js with express framework and Mongo DB database. The text communication is transferred through and from servers and the data transmission is facilitated through point to point connection between servers. Due to the usage of react framework, virtual space concept is implemented which enhances the performance over existing applications developed using PHP by a factor of approximately 6 times.

**KEYWORDS:** real time chat, multi-perform, node js

## I. INTRODUCTION

With the growth and development in information technology, communication has become easier like never before. There are applications that help in the process of communication by relaying texts, images, files, etc. from one person to the other. Several such applications do exist that serve as a means to communicate to a large population.

Such applications are often aimed at the general public and serves the society as a whole. There are very few applications that facilitate communication within organizations such as institutes, industries, companies etc. that limit the number of users and keeps the content being transferred among the users of the organization private. Therefore, this project, the web chatting application, is aimed at to overcome this problem and to provide users with a much better platform that keeps the texts at bay and confined within a boundary. This paper proposes a chat driven online framework that empowers continuous cooperation between clients in a shared environment. The design of the framework presented is from other collective conditions by presenting novel calculations, conventions and techniques in the creation, coordination and correspondence of rich-content and other information [2]. Fundamental to the arrangement proposed is the utilization of the website page as a combined content producer and communication tool, equipped for both pushing and pulling the substance to numerous clients from numerous applications dynamics on one or many site pages being used. The solution provided in this paper is based on tying clients in a complex correspondence process upgraded with supposed "Smart Objects" which presents extra unique usefulness [6].

**How to cite this paper:** Akhilesh Sarjit M S | Srivishak V | Shiddarth S | Saravana Kumar P | Preethi D "Web Chat using React Framework" Published in International Journal of Trend in Scientific Research and Development (ijtsrd), ISSN: 2456-6470, Volume-4 | Issue-3, April 2020, pp.394-396, URL: [www.ijtsrd.com/papers/ijtsrd30416.pdf](http://www.ijtsrd.com/papers/ijtsrd30416.pdf)



Copyright © 2020 by author(s) and International Journal of Trend in Scientific Research and Development Journal. This is an Open Access article distributed under the terms of the Creative Commons Attribution License (CC BY 4.0) (<http://creativecommons.org/licenses/by/4.0>)



This chatting application will be built as a web-based and mobile application so as to provide the users with flexibility. This chatting application is aimed for the users in organizations and institutions with their very own servers so as to provide privacy for the users. But apart from the specifics usage of the application, this can also be put to generic usage so as to extend the services to the general public.

This application is developed and built using React.js, HTML, SASS, Node.js, Express and MongoDB [3].

Node.js is a software platform that is used in building server-side flexible applications in a network application.

HTML is the standard mark-up language for documents designed to be built in a web browser. SASS is a CSS pre-processor for styling. As this project is being aimed at for mobile applications, React.JS library is put to use. This is a JavaScript library for building user interfaces [5]. React can be used as a base in the development of single-page or mobile applications.

Based on the description above, the problem is how to design and build a dedicated mobile application and web-based chatting platform aimed for specific part or general public in real time to make the communication platform and the process of sharing a whole lot easier and faster [8].

The objective here can be defined as to design and build a dedicated mobile application and web-based chatting

platform aimed for specific part or general public in real time to make the communication platform and the process of sharing a whole lot easier and faster [4].

The scope of the project should be broken-down and the system should be declared before advancing further. The scope are as follows:

1. The design and construction of this application is aimed at building a web-based application and mobile application
2. This system is developed using React.js and Node.js
3. The database of this system implemented using Mongo DB

## II. LITERATURE SURVEY

Previous chat applications like Yahoo messenger were based on the usage of PHP and MySQL for server maintenance and for processing client requests. PHP is a server side, cross platform, HTML embedded tool used to design web pages [1]. MySQL is an open-source framework for the management of relational databases. It is the most common management system used with PHP database. MySQL is designed and provided by Oracle. However, the usage of PHP due to their synchronous working nature will be very time-consuming procedure for a large database retrieval [7]. The other similar technology used in hangouts is jQuery, which is a JavaScript based server design tool which uses DOM (Document Object Model) feature to navigate a document, select DOM elements, create animations, handle events, and develop Ajax applications [4]. jQuery also provides capabilities for developers to create plug-ins on top of the JavaScript library. This enables developers to create abstractions for low-level interaction and animation, advanced effects and high-level, theme-able widgets [9]. The modular approach to the jQuery library allows the creation of powerful dynamic web pages and Web applications. Since, the performance parameter can be improved by using the React framework, the concept of virtual space improves the performance rate by only altering the space created for that specified request instead of making a call for the altering of whole work area.

## III. SYSTEM DESIGN

The backend interface is implemented with two servers: client and Server. Server side is responsible for the database retrieval, database maintenance and their services to the client side, whereas the page maintenance and the application interface will be performed by the client side. Website is a set of pages provided through interconnected network that is available for everyone to access around the globe through network facility. First, the communication between client end and server end through Node.js which acts as intermediary by the usage of JSON. JSON (JavaScript Object Notation) is a lightweight data interchange format, readable and writable by humans, as well as easily translated and made (generated) by the computer.

The services are provided by MongoDB, Express.js, React and Node.js. Dual path communication between the client side and server side is possible through this channel. Here, every configuration is implemented only through JavaScript, because in MERN stack both server and client sides support the usage of common language. If the user starts using web chat application and perform any steps in the page, this will initiate a request to JSON, JSON would then initiate the

service, then after obtaining the required information from the server end, it will provide the results in the page.

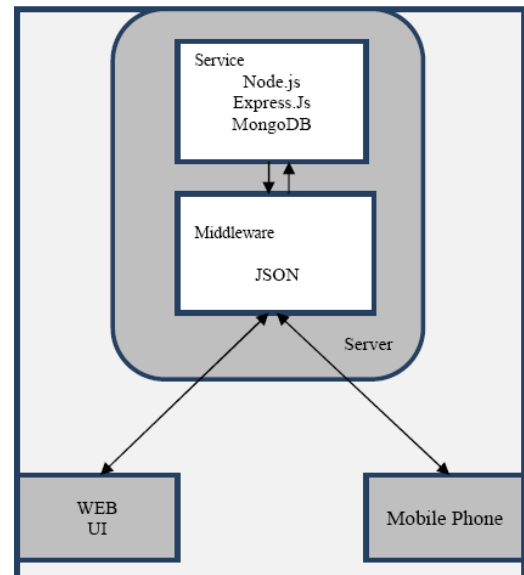


Fig. 1 Server Client Communication

The steps are as follows:

1. While accessing the site, main page will pop up first, with options for existing user login and new user signup. If User details are already present in the database, user details will be asked for logging in the login page. If the signup option is selected then it will redirect to a new user register page, where the user details are collected and then stored in database.
2. After logging in to the portal, user must have a contact before starting conversation. So, they can locate them directly from the app. User can then add friends to their friend list, and wait for confirmation of friend's request.
3. Moreover, the chat database must be checked for previous chat history and if present, chat history is displayed.
4. At last, chat will be possible between the users.

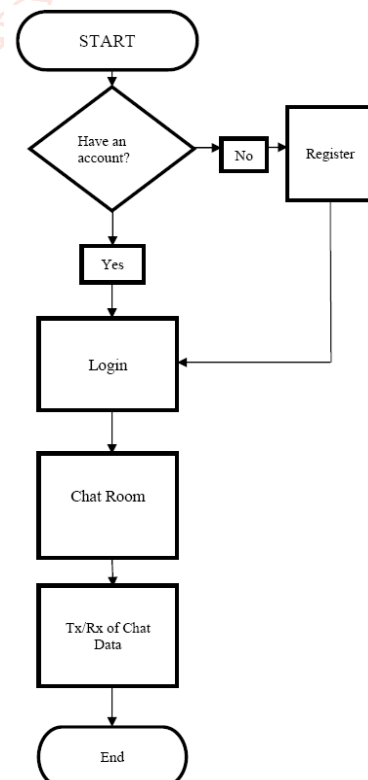


Fig. 2 User interface design

The user interface design is explained in the process below:

1. On opening the application, the user is first taken to the home page that requires user credentials to access the main page. This can be obtained by creating an account for the new users and the old users can continue with the previous accounts.
2. On successful login, the user will be directed to the home page or the chat room where the user is displayed with all the contacts. Chat room of specific contacts can be accessed by just clicking on the contact name being displayed.
3. The data received by the user through any of the user's contact will be displayed in the chat room. To send the data, the user will have to enter the chat room of a specific contact to send the data.

#### IV. IMPLEMENTATION



Fig 3 Home Page

The above picture depicts the home page that give an idea of the design aspect. It mainly contains two options i.e. login and signup

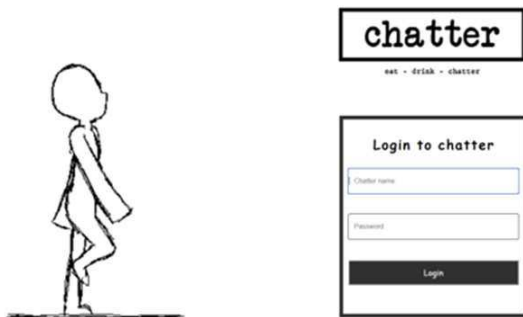


Fig 4 Login Page

Login page mainly checks for the credentials so as to provide a secure atmosphere for the users.



Fig 5 Signup Page

The data collected in signup page is used during the login phase.

#### V. TEST RESULT

Testing application build with Node.js, React and MongoDB with PHP and MySQL (speed parameter)

no.	Chat with PHP and MySQL (in seconds)	Chat with Node.js, MongoDB and React (in seconds)
1	1.56	0.13
2	1.44	0.13
3	1.44	0.15
4	2.03	0.18
5	1.34	0.13
6	1.34	0.19
7	2.00	0.18
8	1.53	0.17
9	1.55	0.16
10	1.39	0.22

Table I CPU Test (performance)

	CPU Time	System Time	RAM
PHP	102.69s	104.20	2497508KB
Node.js	2.64s	2.64s	92240KB

Table II

#### VI. CONCLUSION

On running the tests, it can be inferred that the chat application developed using Node.js, React and MongoDB is faster in real-time with a speed less than a second compared to the application developed using PHP and MySQL. Node.js is faster than PHP by more than 35 times (by system time) and is more efficient than PHP in terms of RAM usage.

#### VII. References:

- [1] Croucher, T.H., & Wilson, M. (2012). Node: Up and Running. United States
- [2] Keissling, Manuel. 2012. The Node Beginner Book. Lulu.com, United States
- [3] Teixeira, Pedro. 2012. Hands-on Node.js. Wrox.
- [4] Sidik, B. (2011). JavaScript. Bandung: Informatika
- [5] Purnomosidi, B. (2013). Pembangunan Sistem Informasi Penegelolaan Inventaris Barang Divisi Pustekin Berbasis Web. Bandung: Politeknik Telkom.
- [6] Tim A. Majchrzak University of Agder, Kristiansand, Norway Andreas Bjørn-Hansen Westerdals, Oslo, Norway Tor-Morten Grønli Westerdals, Oslo, Norway
- [7] Tim A. Majchrzak, Benjamin Ruland and Till Weber " Department of Information Systems, University of Munster, M " unster, Germany
- [8] A study of internet instant messaging and chat protocols published on 14 August 2006 by R. B. Jennings, E. M Nahum, D. P Olshefski, D Saha, Zon-yin Shae, Christopher J. Waters (<http://ieeexplore.ieee.org/document/1668399/>)
- [9] Robert W. Sebesta: Programming the World WideWeb, 8<sup>th</sup> Edition Pearson Education, 2015