Chat Application Using Mern Stack

Vinoth Kumar G and G. Dhivya
Department of Computer Application, Karpagam Academy of Higher Education, Coimbatore-641 021, Tamil Nadu, India

Keywords. MERN (MongoDB, Express.js, React.js, Node.js), Internet, Chat, Security.

Abstract: The latest evolution of the Internet has brought the entire world within our reach. Everything now revolves around the internet, from the exchange of information to experiencing reality. The internet has transformed the world into an interconnected global community, and this project is no exception, as it relies heavily on the internet. This document highlights the importance of reverse communication in our daily lives and its profound impact on the technological landscape. The objective here is to create a reverse communication system based on the MERN stack. This system enables people to send messages privately and publicly, as well as share various forms of content such as text, images, videos, and more. It's an online platform designed for people to interact and converse with each other over the internet. Moreover, it is not only more reliable and secure than traditional communication systems but also surpasses them in popularity. Reverse communication features are highly sought after by internet enthusiasts and smartphone users alike. Every year, hundreds of millions of smartphone owners engage in reverse communication activities. These communication tools offer free and convenient communication options, and most of them are available as free downloads, making them incredibly appealing to users. These platforms provide a wide range of services and built-in features to their users, although often, users tend to overlook the broader aspects of their interactions and conversations.

1 INTRODUCTION

The phenomenon of reverse communication wields a profound impact on our everyday lives. A plethora of reverse communication platforms have gained widespread popularity worldwide. Each of these platforms introduces novel features that distinguish them from their peers. These platform providers engage in spirited competition, continually rolling out competitive features with each new release. Their contributions hold significant importance for users and profoundly influence their daily routines. Users seek out superior internet-based communication platforms that they perceive as dependable, secure, and trustworthy. Currently, some of the prevalent reverse communication platforms include WhatsApp, Facebook, Instagram, Hike, and others. These platforms collectively boast billions of avid users across the globe, establishing the companies behind them as global leaders. These companies have demonstrated consistent revenue growthand employ vast teams dedicated to developing innovative features to stay ahead in the competitive landscape. These platforms offer diverse features and employ various strategies to safeguard the security of user data. In today's world, data theft constitutes a major criminal concern, with a multitude of cases arising. As a result, companies are compelled to implement stringent measures to protect against data breaches by external entities. The fundamental reverse communication system should encompass both message transmission and reception processes occurring simultaneously. Within this system, the concurrent sending and receiving of messages occur seamlessly, facilitated by the MERN technology stack. The rapid expansion and enhancement of the internet have prompted an increasing number of individuals to opt for web- based communication tools.

2 REVIEW OF LITERATURE SURVEY

Such tools have revolutionized communication across vast distances, necessitating real-time and multi-platform capabilities to cater to a broad user base. Notably, web-based real-time communication tools eliminate the need for additional third-party

G, V. and Dhivya, G.
Chat Application Using Mern Stack.
DOI: 10.5220/0012612100003739

In *Proceedings of the 1st International Conference on Artificial Intelligence for Internet of Things: Accelerating Innovation in Industry and Consumer Electronics (AI4IoT 2023)*, pages 240-244
ISBN: 978-989-758-661-3

software, making visual communication easily accessible. The technology stack utilized in building these tools includes React.js, Node.js with the Express framework, and MongoDB for database management. Communication relies on servers for message exchange, facilitated through point-to-point connections between servers. This innovative approach incorporates a response framework, resulting in a six-fold enhancement in performance compared to systems developed using PHP.

This paper focuses on the development of an Online College Management System, a significant asset for educational institutions. Named College ERP and built on the MERN stack, this system serves academic professionals and their diverse responsibilities. Its primary purpose is to streamline access to critical information within an educational organization. To achieve this, users must register with the system, which functions as an information hub for the institution. Both students and staff, both specialized and non-specialized, can access and upload data to the system's database. In today's rapidly evolving business landscape, understanding customer demand in an efficient and timely manner is paramount. The shift to online commerce has become integral to the lives of the younger generation. E-commerce platforms offering a wide array of products have transformed basic needs into accessible luxuries. This design introduces an interactive web interface for viewing diverse products and enables registered users to purchase items using various payment methods. Additionally, it provides an accessible avenue for business owners to manage incoming orders.To establish an e-commerce web platform, various technologies have been examined and incorporated, including React.js, MongoDB, Node.js, and Express.js. This project aims to simplify the presentation of diverse products and the creation of a web platform that seamlessly connects customers with their desired products. Conversational tools have become indispensable for smartphone users, offering the capability to exchange text messages, images, and videos securely and free of charge. This paper endeavors to propose a chat application with End-to-End encryption, ensuring secure communication while prioritizing data protection. A set of security conditions is presented, serving as the foundation for the development of this chat application, which is then compared with other popular alternatives to validate its End-to-End encryption capabilities.

Additionally, a website has been developed to allow users to sign up, log in, and engage in academic and social interactions, including sharing updates, liking, and commenting on posts, and following other users. This website also provides a valuable directory of race sponsors, addressing the challenge of locating sponsors during crucial times.

Moreover, users can create or join chat rooms, facilitating global conversations and fostering connections with individuals worldwide. These chat applications have gained immense popularity among internet users and smartphone owners, with millions of users engaging in chat conversations annually. They offer cost-free communication and are often available as free downloads, making them particularly attractive to users. While these applications offer various services and built-in features, they sometimes overshadow their core purpose – simple and effective communication.

In conclusion, the recent evolution of the internet has effectively brought the entire world within our reach, revolutionizing the way we exchange information and engage in online transactions. This paper underscores the significance of chat applications in modern daily life and their impact on the technological landscape. The project aims to develop a chat system based on Java multithreading and network design, enabling secure private and public communication, as well as seamless sharing of various media content. This online system is designed to foster interaction and collaboration on the internet, prioritizing reliability and security over traditional communication systems. The utilization of Java, multithreading, and client-server architecture ensures the robustness and scalability of the chat application, making it suitable for deployment in various private organizations, such as colleges and IT institutions.

Proposed System:

The envisioned chat application system, constructed using the MERN stack, is set to emerge as a comprehensive web application that empowers users in effortlessly establishing, overseeing, and erasing chat rooms. This system will comprise four fundamental elements: a MongoDB database, an Express.js backend, a React.js frontend, and a Node.js server.

The MongoDB database will serve as the repository for all chat rooms and user-related data, encompassing messages dispatched by users. Meanwhile, the Express.js backend will shoulder the responsibilities of user authentication and serve as the intermediary for communication between the server and the database. The React.js frontend will craft the user interface, enabling users to seamlessly create and administer their chat rooms. Finally, the Node.js server will facilitate communication between the frontend and backend components.

The system will feature a user-friendly interface that simplifies the process of chat room creation and management. Users will have the capability to exchange messages with each other and peruse past conversations. Additionally, a straightforward authentication system will permit users to log in and log out of their accounts securely.

In summary, the proposed chat application system utilizing the MERN stack promises to be an uncomplicated and user- friendly tool. It will empower users to effortlessly create, manage, and delete chat rooms while ensuring secure communication and data storage.

Existing System:
The current configuration of a chat application utilizing the MERN stack comprises essential components, including a MongoDB database, a Node.js server, an Express server, and a React front-end.

Within the MongoDB database, user data and conversations are stored. When a user registers, their email address, username, and password are recorded in the database. Additionally, when users engage in conversations, the database maintains records of the conversation's messages and participants.

The Node.js server assumes the role of managing user authentication. This encompasses tasks such as validating usernames and passwords, as well as issuing confirmation messages upon user registration or login.

The Express server primarily functions as a conduit for directing requests originating from the front-end. Furthermore, it handles user sessions and upholds the connection between the front-end and the database.

On the user-facing side, the React front-end serves as the interface through which users can access their conversations, transmit messages, and search for other users. It also takes charge of displaying notifications and messages to users upon receiving new messages.

3 METHODOLOGY

The present setup of a chat application employing the MERN stack is composed of fundamental components, including a MongoDB database, a Node.js server, an Express server, and a React front-end.
Inside the MongoDB database, user data and conversations are securely stored. When a user registers, their email address, username, and password are meticulously recorded in the database. Furthermore, during user interactions within conversations, the database dutifully maintains records of messages exchanged and the participants involved.
The Node.js server assumes the crucial role of overseeing user authentication procedures. This encompasses critical tasks like validating usernames and passwords, in addition to promptly dispatching confirmation messages following user registration or login.

The Express server primarily serves as a conduit, responsible for routing requests originating from the front-end. Additionally, it capably manages user sessions and ensures the seamless connection between the front-end and the database.

On the user-facing front, the React front-end functions as the user interface. Through this interface, users gain access to their ongoing conversations, send messages, and conveniently search for other users. Moreover, it efficiently handles the presentation of notifications and incoming messages, ensuring users are promptly informed when new messages arrive.
Technology Used:
HTML, CSS, and JavaScript:
MongoDB - A Cross-Platform Document-Oriented Database
Express.js - A Backend Framework
React.js - A Frontend Library
Node.js - A JavaScript Runtime Environment

HTML, known as the Hyper Text Markup Language, serves as the industry standard for creating documents intended for webdisplay. It is compatible with technologies such as Cascading Style Sheets (CSS) for styling and scripting languages like JavaScript for interactivity.

CSS, short for Cascading Style Sheets, is a styling language employed to define the presentation of documents written in markup languages like HTML. CSS's primary purpose is to enable the separation of document structure and styling, encompassing layout, color schemes, and font selections.

JavaScript, often referred to as Java's cousin, stands as a foundational technology of the World Wide Web. Over 97% of websites employ JavaScript on the client side to enhance web functionality, frequently incorporating third-party libraries. Major web browsers dedicate a JavaScript engine to execute code on users' devices.

In the MERN stack, React.js takes the lead as the declarative JavaScript framework for crafting dynamic client-side applications in HTML. React enables the creation of user interfaces using reusable components, linking them to backend data, and rendering them as HTML.

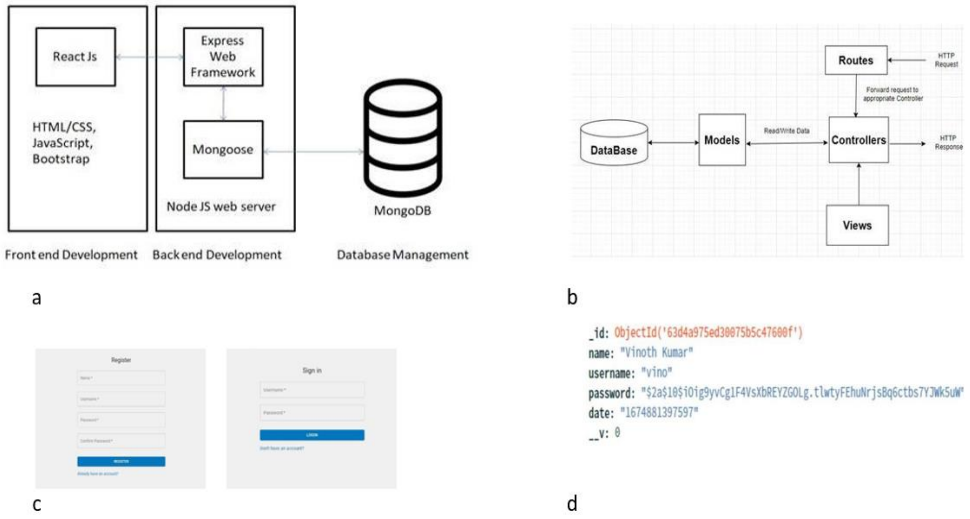

Figure 1: (a)MERN Architecture (b) Block Diagram (c) Register and Login page (d) JSON Token.

Node.js, positioned as the server-tier component, runs the Express.js server-side framework. Express.js is described as a rapid, unbiased, minimalist web framework for Node.js, and it lives up to this description. Express.js offers robust features for URL routing, facilitating the matching of incoming URLs with server functions, and efficiently handling HTTP requests and responses.

In the database tier, MongoDB comes into play. MongoDB is a versatile, document-oriented database suitable for storing various types of data such as user profiles, content, comments, uploads, and events. Its compatibility with React, Express, and Node makes it a seamless choice for web applications.

4 RESULTS AND DISCUSSION

A user's password holds greater sensitivity compared to their session ID due to three fundamental reasons:

1. Session IDs can be subject to access restrictions. For instance, despite having a session ID, a system may require the user to re-enter their password for specific actions, such as modifying their account email address.
2. Session IDs can be invalidated or revoked.
3. A user's password is frequently used in various other systems, such as banking websites. If it gets compromised, it could potentially be exploited in other services, posing a significant risk to the user's security and well-being.

Given these considerations, it is of paramount importance to NEVER store or reuse a user's password. We will delve into hashing techniques in subsequent discussions, which elucidate the proper methods for securely storing hashed data to authenticate user-provided passwords during the login process.

5 ALGORITHM

Entire books can be dedicated to these topics, and it is advisable to explore more comprehensive resources for a deeper understanding of these subjects, as suggested in the references at the end of this text.

Jwt Hash Possword Store in Database: The JWT Hashing Algorithm is a cryptographic technique employed to securely safeguard passwords in contemporary web applications. This algorithm is based on the JSON Web Token (JWT) standard and is designed to offer both simplicity and robust security. To grasp the JWT Hashing Algorithm's functionality, it's essential to first comprehend what a JSON Web Token (JWT) entails.

A JWT is an openly recognized standard (RFC 7519) that delineates a secure method for exchanging information between two parties in the form of a JSON object. It comprises three core components: a header, a payload, and a signature. The header and payload serve to identify the user, while the signature ensures data integrity and verifies that it hasn't been tampered with.

The JWT Hashing Algorithm is specifically employed to store passwords securely. It operates by initially combining the password with a salt—a random string of characters used to thwart attackers from easily deducing the password. This salt is

securely stored alongside the hashed password in a database. When a user attempts to log in, the password they provide is hashed in the same way with the salt and then compared to the stored hashed password in the database. Access is only granted if these two hashes perfectly match.

This algorithm is intentionally designed to be highly secure and is considered one of the most robust methods for password storage. The use of a unique and randomly generated salt for each user makes it exceedingly challenging for attackers to guess the password. Additionally, the hashed password is not stored in plain text, further enhancing security by preventing unauthorized access.

In summary, the JWT Hashing Algorithm is a secure method for safeguarding passwords in contemporary web applications. Built upon the JWT standard, it prioritizes both simplicity and security. By utilizing a unique salt and securely hashing passwords, it effectively guards against unauthorized access and ensures the user's data remains secure.

6 CONCLUSION

In conclusion, I can confidently state that this learning experience has been tremendously enriching. Thanks to this program, I have gained a deeper understanding of my specialized skills, and it has provided me with valuable interactions. Currently, the MERN stack is a widely adopted technology for web and chat applications, enjoying immense popularity among developers worldwide. When we explore the internet, we encounter millions of websites and applications built using MERN and MEAN stacks.

I find myself navigating a different terrain from what I'm accustomed to, and it's a journey of growth and independence. This experience has not only enhanced my professional skills but also my personal life, instilling the belief that I can achieve more and continue to acquire new skills independently.

For those considering a career in this field, there are vast opportunities available. Both private and public organizations are actively recruiting web developers and app developers for their online projects and development efforts. With the rapid growth of the online industry, the demand for web and app development professionals is steadily increasing, promising significant job prospects in the days to come.

Additionally, individuals well-versed in this field can explore freelance opportunities. There are numerous online platforms that offer freelance projects to individuals, providing an avenue for independent work and entrepreneurship.

REFERENCES

Shubham Patil, Saurav Daware, Ameya Bhagat, Prof. Jayant Sawarkar, Department of Computer Engineering, Datta Meghe college of Engineering, Airoli, Navi Mumbai, Maharashtra, India.

Sourabh Mahadev Malewade, Archana Ekbote, Information Technology Department VCET, Vasai, Palghar, India.

Akhilesh sarjit M S, Srivishak V, Shiddarth S, Sarvana Kumar P, Preethi D, Department of Electronics and Communication Engineering, Bannari Amman Institute of Technology, Erode, Tamil Nadu, India.

Tanya agarwal, Swapnil Upadhyay, Shreeansh Srivastava, Rohit Sharma, Poornima college of Engineering, India.

Nithin Katla, Gautham Kumar M, Pidugu Rohith Raj, Dr. S Shitharth, Dept. of CSE, Vardhaman college of Eng., Hyderabad.

Noor Sabah, Jamal M , Kadhim and Ban N. Dhannoon, Department of computer science , Al-Nahrain Univerversity.

Sumit Kumar Mishra, Ankit Kumar, Ankush Kaundal, Upasna Joshi, Department of Computer Science and Engineering, Delhi Technical Campus, New Delhi, India

Suman Kharel, Jukka Jauhiainen, Degree Programme in Information Technology, Oulu University of Applied Sciences

Bishal Kumar, Kumar Sagar, M Arvindhan, Ankit kr. Tiwari, Galgotias University Greater, Noida, India.

Lavish Mangal, Pushpendre Pratap Singh, Keshav Gupt, Jai Kumar, Saijal Gupta, Department of Information Technology, Dr. Akhilesh Das Gupta Institute of Technology and Management, New Delhi, India.