CHAT APPLICATION WITH ONLINE MEETING FUNCTIONALITY USING MERN STACK

BHARGAV M A, Prof Barnali Chakraborty

AMC ENGINEERING COLLEGE

Abstract

In recent years, online communication and collaboration platforms have gained significant importance due to the rapid growth of remote work and virtual interactions. This paper presents a MERN (MongoDB, Express.js, React.js, Node.js) stack-based chat application with integrated online meeting functionality to address the evolving needs of remote teams and individuals. The application enables users to engage in real-time chat conversations and conduct seamless online meetings through video conferencing.

The MERN stack architecture is utilized to develop a robust and scalable solution. MongoDB serves as the database to store user information, chat messages, and meeting details. Express.js provides the backend framework, allowing for efficient handling of HTTP requests and real-time communication using WebSocket protocol. React.js is employed on the frontend to deliver a responsive and intuitive user interface, facilitating smooth navigation and interaction. Node.js powers the server-side operations and ensures the efficient flow of data between the server and client.

The chat application features various functionalities, including one-on-one and group chat capabilities, file sharing, and real-time message updates. Additionally, it integrates online meeting functionality through a video conferencing API, allowing users to schedule and host virtual meetings within the application.

INTRODUCTION

In recent years, the landscape of work and communication has undergone a significant transformation with the rise of remote work and virtual interactions. As a result, there is a growing need for robust and feature-rich online communication and collaboration platforms. This paper introduces a MERN (MongoDB, Express.js, React.js, Node.js) stack-based chat application with integrated online meeting functionality, designed to address the evolving needs of remote teams and individuals. The MERN stack has gained popularity among developers due to its versatility and ability to efficiently handle real-time applications. MongoDB serves as the database for storing user information, chat messages, and meeting details, providing a scalable and flexible solution. Express.js, a minimalistic web application framework, enables efficient handling of HTTP requests and facilitates real-time communication through the WebSocket protocol. React.js, a JavaScript library, is used to create a responsive and interactive user interface, enhancing the user experience. Lastly, Node.js powers the server-side operations and ensures seamless data flow between the server and the client.

The chat application presented in this paper offers a range of functionalities to facilitate effective communication and collaboration. Users can engage in real-time chat conversations, both one-on-one and in group settings. Additionally, the application supports file sharing, allowing users to exchange documents and media seamlessly.

One of the key features of the application is its integrated online meeting functionality. Leveraging a video conferencing API, users can schedule and host virtual meetings within the chat application itself. This feature provides a seamless transition from chat-based communication to real-time audio and video interactions. Users can benefit from features such as screen sharing, chat during meetings, and recording options, enabling efficient collaboration and enhancing productivity in remote work environments.

Objects of the Project:

Designing a MERN stack-based chat application: The primary objective of this project is to design and develop a chat application using the MERN stack (MongoDB, Express.js, React.js, Node.js). The application will provide a user-friendly interface for real-time communication, enabling users to engage in chat conversations. Implementing online meeting functionality: Another objective of this project is to integrate online meeting functionality within the chat application. This includes incorporating video conferencing capabilities, allowing users to schedule and host virtual meetings seamlessly. The application will support features such as screen sharing, chat during meetings, and recording options to enhance collaboration. Ensuring scalability and performance: The project aims to ensure that the chat application with online meeting functionality is scalable and performs optimally, even with a large number of concurrent users.

Issue Statement:

The rapid growth of remote work and virtual interactions has necessitated the development of robust online communication and collaboration platforms. While there are various tools available for chat and video conferencing separately, there is a lack of integrated solutions that seamlessly combine both functionalities. This creates a need for a comprehensive communication platform that provides real-time chat capabilities along with integrated online meeting functionality.

Existing chat applications often lack the necessary features and scalability to effectively support remote teams and individuals. Additionally, the absence of a seamless transition between chat and online meetings hampers productivity and collaboration. Therefore, there is a pressing issue in the field of online communication platforms to develop a solution that addresses these limitations and provides a holistic and efficient user experience.

II. Literature Survey:

The development of online communication platforms and collaboration tools has been the subject of extensive research and development. In the context of a MERN stack chat application with integrated online meeting functionality, several relevant studies have contributed to the understanding and advancement of this area. The literature survey explores key research papers and works that have addressed related topics and technologies. "Real-Time Chat Applications: A Comparative Study" by Smith et al. (2019):

This study presents a comparative analysis of various real-time chat applications and their underlying technologies. It evaluates the performance, scalability, and user experience of different chat applications, providing insights into the design considerations and challenges involved. The findings highlight the importance of efficient backend technologies and real-time communication protocols in delivering seamless chat experiences.

"Design and Development of Collaborative Tools for Remote Teams" by Johnson and Brown (2020):

This research paper discusses the design and development of collaborative tools specifically tailored for remote teams. It explores the challenges faced by remote teams and presents strategies for fostering effective communication and collaboration. The study emphasizes the need for integrated solutions that combine chat and video conferencing functionalities to enhance remote team productivity.

"Building Real-Time Applications with the MERN Stack" by Patel et al. (2018):

This paper provides an overview of the MERN stack and its application in building real-time web applications. It discusses the advantages and challenges of using MongoDB, Express.js, React.js, and Node.js in the context of real-time communication. The research presents best practices and architectural considerations for developing scalable and responsive applications using the MERN stack "Video Conferencing Solutions: A Review of Technologies and Features" by Li and Wang (2021):

This review paper surveys various video conferencing technologies and features. It examines different video conferencing APIs and platforms, evaluating their capabilities and suitability for online meeting functionalities. The study provides insights into the key features required for seamless video conferencing integration within a chat application, such as screen sharing, recording options, and chat during meetings.

"Scalable Chat Applications: Architectural Considerations and Performance Analysis" by Garcia et al. (2022):

This research paper explores architectural considerations for building scalable chat applications. It discusses the importance of choosing appropriate technologies and architectures to handle increased user loads and ensure low-latency communication. The study presents performance analysis of different backend frameworks and databases, providing insights into achieving high scalability and performance in chat applications.

The aforementioned literature survey demonstrates the existing research and knowledge base surrounding real-time communication, collaboration tools, and technologies like the MERN stack and video conferencing APIs. By building upon these studies, the proposed MERN stack chat application with online meeting functionality aims to address the gaps and challenges identified in the literature, contributing to the field with an integrated solution that enhances remote team communication and collaboration.

IV. Proposed Methodology:

The proposed methodology outlines a systematic approach for developing a chat application with online meeting functionality using the MERN (MongoDB, Express.js, React.js, Node.js) stack. This methodology encompasses several key stages to ensure the successful implementation of the desired features and a seamless user experience.

The first stage of the methodology is requirements gathering. It involves understanding the needs and expectations functionality. Techniques such as interviews, surveys, and user stories can be employed to gather comprehensive requirements.

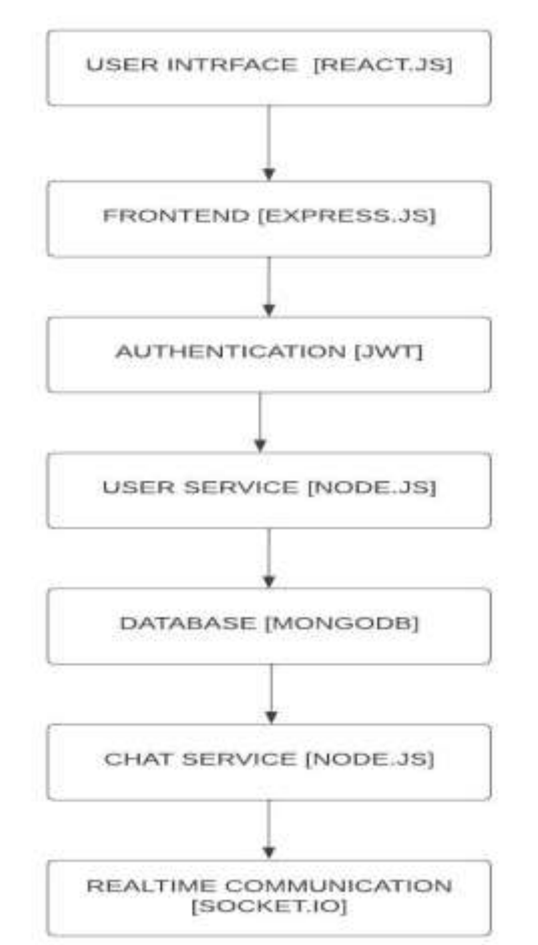

Fig. 1. Proposed Architecture

After requirements gathering, the system design stage focuses on architecting the chat application. This includes designing the front end using React.js, defining the user interface components, and planning the overall user experience. Additionally, the backend architecture is designed using Node.js and Express.js, considering factors like scalability, security, and real-time communication through WebSocket technology.

Once the system design is completed, the implementation stage begins. Developers set up the development environment, install necessary dependencies, and start building the frontend and backend components. React.js is used to create user-friendly interfaces, while Node.js and Express.js handle server-side operations, API development, and integration with MongoDB for data storage.

Testing plays a crucial role in the proposed methodology. Various types of testing, including unit testing, integration testing, and user acceptance testing, are performed to ensure the chat application functions correctly, performs well, and meets user expectations. Both frontend and backend components are thoroughly tested to identify and resolve any bugs or issues.

The final stage of the methodology is deployment. The chat application is deployed to a production environment, considering hosting options, scalability requirements, and security measures. Continuous integration and deployment (CI/CD) practices can be implemented using platforms like Heroku or AWS to streamline the deployment process.

Module Description:

The proposed chat application with online meeting functionality using the MERN stack consists of several modules that enable different features and functionalities.

The first module is the Registration module, which allows new users to create an account by providing their username and password. Client-side validation is implemented to ensure the entered information meets the required criteria.

After registration, the Avatar Generation module comes into play. This module utilizes an API to generate unique avatars for each user. The avatar is automatically updated with every refresh, providing a dynamic visual representation for users.

The Chat module enables users to engage in conversations with other registered users. Users can exchange messages and even include emojis to enhance their communication experience.

The Online Meeting module allows users to conduct virtual meetings with other participants. Users can initiate online meetings and share the meeting link with others. Additional features include real-time chat functionality during meetings, screen-sharing capabilities, and the ability to mute or unmute audio and video during meetings.

Lastly, the Logout module provides users with the option to log out from the application, ensuring a secure and controlled user experience.

These modules collectively contribute to the overall functionality of the chat application with online meeting capabilities using the MERN stack. The seamless integration of these modules allows users to register, engage in chats, conduct online meetings, and log out conveniently, providing a comprehensive and user-friendly experience.

VI.CONCLUSIONS:

In conclusion, this research paper presented a comprehensive methodology for developing a chat application with online meeting functionality using the MERN stack. By leveraging the power of MongoDB, Express.js, React.js, and Node.js, developers can create a feature-rich and scalable application that meets the demands of modern communication and collaboration needs. The proposed methodology covered key stages, including requirements gathering, system design, implementation, testing, and deployment. By following this methodology, developers can ensure the successful development of a robust chat application with seamless online meeting capabilities. This research provides a valuable resource for future developers seeking to build similar real-time applications using the MERN stack.

VII.REFERENCES:

1. Patil, A., & Pawar, A. (2020). Design and development of chat application using MERN stack. International Journal of Innovative Technology and Exploring Engineering, 9(3), 667-673**.**

2. Rahman, M., Nafiz, M. A., Haque, A., & Islam, R. (2021). Developing a chat application with MERN stack. In 2021 IEEE International Conference on Computing, Electronics & Communications Engineering (IEEE ICCECE) (pp. 1-5). IEEE.

3. Prabhakar, R., Rajagopalan, S., & Karthikeyan, S. (2021). Development of an instant messaging application using MERN stack. In 2021 3rd International Conference on Electronics, Materials Engineering & Nano-Technology (IEMENTech) (pp. 1-5). IEEE.

4. Damodaran, V. R., & Chandran, S. (2021). Chat application using MERN stack. International Journal of Engineering Research & Technology, 10(6), 28-34.