Dupli Checker

PLAGIARISM SCAN REPORT

3% Plagiarised	97% Unique	Date	2024-10-20
		Words	989
		Characters	9989

Content Checked For Plagiarism

v

Abstract

BlockMeet is a decentralized video conferencing application designed to overcome the limitations of traditional centralized platforms. Using a multi-layer architecture that includes a front-end React application, server, WebSocket signaling, and MongoDB database, BlockMeet enables secure, privacy-focused, real-time communication. The hydrocarbon-like architecture connects super peers to normal peers in a structured manner, ensuring efficient information flow. While the current prototype supports core video conferencing features, future versions will incorporate blockchain for interaction logging, IPFS for decentralized storage, and AI for meeting summarization.

Github Repo: https://github.com/Omkar-Patil-2003/BlockMeet

1

1. Introduction

The increasing reliance on digital communication platforms has brought about the need for more secure and private solutions. Traditional video conferencing systems often rely on centralized servers, leading to data privacy concerns and the risk of single points of failure. BlockMeet addresses these challenges by implementing a decentralized video conferencing architecture using WebRTC and blockchain technologies, focusing on user privacy, data ownership, and censorship resistance.

2

2. Aim and Objectives

✓ To develop a decentralized video conferencing platform using a multi-layer architecture.
✓ To ensure high security and privacy through blockchain integration for logging interactions and decentralized storage using IPFS.
✓ To support scalable and reliable real-time communication with a hydrocarbon-like architecture.
✓ To enable future capabilities like AI-powered meeting summarization and storage of meeting recordings.

3

3. Literature Surveyed

• Existing video conferencing solutions like Zoom and Skype rely on centralized networks. These architectures suffer from limitations such as centralized data storage, potential data breaches, and censorship risks.
• Decentralized architectures offer advantages in terms of privacy and scalability but present challenges in consistency and network synchronization.
• WebRTC has emerged as a popular choice for peer-to-peer media transfer, while blockchain and IPFS provide secure and immutable data management solutions.

4

4. Problem Statement

The primary challenge with current video conferencing platforms is their centralized nature, which results in data privacy concerns, dependency on single points of failure, and potential content censorship. BlockMeet aims to build a decentralized solution to overcome these issues, providing a user-centric, secure, and censorship-resistant communication platform.

5

5. Scope

BlockMeet will initially focus on medium-sized meetings with basic conferencing features. Future versions will expand to include functionalities such as interaction logging on the blockchain, AI-generated meeting summaries, and decentralized storage for meeting recordings.

6

6. Proposed System

The BlockMeet system is built with a multi-layer architecture consisting of four main components: the front-end, server, WebSocket server, and MongoDB database. The proposed architecture follows a hydrocarbon-like structure, where super peers connect to two normal peers, facilitating decentralized data transfer. The system's architecture ensures fault tolerance, scalability, and privacy for medium-sized meetings.

• Front-End: A React-based application that provides a user-friendly interface for creating and joining meetings, enabling video/audio stream sharing.

• Server: Manages API requests, handles user authentication, and interacts with the database to store user information and meeting details.

• WebSocket Server: Acts as the signaling server for WebRTC, facilitating the exchange of **Session Description Protocol (SDP) and Interactive Connectivity Establishment (ICE)** candidates to establish peer-to-peer connections. It also maintains the mapping of super peers and normal peers in a meetings object.

• Database (MongoDB): Stores user-related data, such as user profiles and meeting metadata. The database does not store super-peer mappings, as they are managed in-memory within the WebSocket server.

The future system enhancements include incorporating blockchain technology to log meeting interactions, integrating IPFS for decentralized storage, and using AI models to convert meeting recordings into text summaries for easy access.

7

7. Methodology

The development process follows an iterative approach to build the decentralized video conferencing solution:

• Architecture Design: Initial discussions led to the selection of a hydrocarbon-like architecture, where super peers manage connections between normal peers and relay data across the network.
• Technology Selection: WebRTC was chosen for real-time media communication, Ethereum blockchain for future interaction logging, and IPFS for decentralized storage solutions.
• Implementation Phases:
1. Basic Features Development: Implemented core functionalities like meeting creation, joining, and decentralized audio/video streaming.
2. WebSocket Integration: Set up a signaling mechanism using a WebSocket server to manage peer connections.
3. Testing and Validation: Conducted extensive tests to ensure stable peer-to-peer communication and identify areas for improvement.
4. Future Development: Planned integration of blockchain for logging and IPFS for storing recordings and summaries.
The methodology emphasizes continuous refinement and validation through user feedback and iterative enhancements.

8

8. Analysis

8.1 Process Model Used for the Project
The project utilizes an Agile methodology, enabling the team to adapt to changes quickly and continuously improve the system based on iterative feedback.
8.2 Feasibility Study
• Technical Feasibility: The use of React, WebRTC, and WebSocket technologies makes the implementation feasible, while Ethereum and IPFS are viable for future features.
• Economic Feasibility: The project leverages open-source technologies, reducing development costs. Integration with blockchain and IPFS may introduce some cost considerations in future versions.
• Operational Feasibility: The system aims to be user-friendly with a simple **interface, making it easy for users to adopt.**

8.3 Cost Analysis
The project primarily incurs costs related to hosting the WebSocket server and server backend. Future costs may involve gas fees for blockchain transactions and storage costs for IPFS.

9

8.4 Timeline Chart

The following are git commits charts

10

11

12

9. Design and Architecture

13

14

15

16

10. Hardware and software Requirements

10.1 Hardware Requirements
Development Environment:
• Minimum: Intel i5 processor, 8 GB RAM, 256 GB SSD
• Recommended: Intel i7 or above, 16 GB RAM, 512 GB SSD
Deployment Environment:
• Server hosting with sufficient bandwidth to support signaling and data transfer for medium-sized meetings.

10.2 Software Requirements

• Front-End Development: React.js, Node.js, HTML, CSS
• Back-End Development: Express.js for the server, WebSocket library for signaling
• Database: MongoDB
• WebRTC: For peer-to-peer media communication

• Blockchain (Future Integration): Ethereum network, Solidity for smart contracts
• IPFS (Future Integration): For decentralized storage
• Development Tools: Visual Studio Code, Git, Postman (for API testing)

17

Matched Source

Similarity 4%

Title:Mixed reality human teleoperation with device-agnostic remote ...

Feb 1, 2024 · To establish connectivity initially, WebRTC ✔ uses Session Description Protocol (SDP) and Interactive Connectivity Establishment (ICE) to find an optimal connection between two peers over any network and through most router NAT (Network Address Translation) schemes or firewalls.

https://www.sciencedirect.com/science/article/pii/S0097849324000037

Similarity 3%

Title:16 Best Data Analytics Software in 2024 - Geekflare

Oct 10, 2024 · Microsoft Power BI ✔ is an advanced Excel version with a familiar interface, making it easy for users to adopt. Its tiered pricing structure is suitable for businesses of all sizes, offering a free Desktop version for content creation and Pro and Premium versions for sharing insights.

https://geekflare.com/software/best-data-analytics-software