IJIRIS: International Journal of Innovative Research in Information Security
Volume 12, Issue 03, March 2026
E-ISSN: 2349-7017
P-ISSN: 2349-7009
https://www.ijiris.com/archives
https://doi.org/10.26562/ijiris.2025.v1203.01

LiveSyncCloud: A Real-Time File Sharing Web Application

Ruthendiran S.,Purushothaman K.,Ragul A.,Saravanan.V
UG Students, Department of Information Technology
Sengunthar Engineering College (Autonomous), Tiruchengode, India
ssruth2@gmail.com, purushothk70@gmail.com
aragulindhu@gmail.com, saravananvelmurugan26@gmail.com
SB Narmada.,
Assistant Professor, Department of Information Technology
Sengunthar Engineering College (Autonomous), Tiruchengode, India
sbnarmada.it@scteng.co.in

Publication History
Manuscript Reference No: IJIRIS/RS/Vol.12/Issue03/ISMR26.FBIS10081
Research Article Open Access| Double-Blind Peer-Reviewed| Article ID: IJIRIS/RS/Vol.12/Issue03/ISMR26.MRIS10080
Received: 31, January 2026, Revised: 14, February 2026, Accepted: 17, March 2026, Published Online: 25,March 2026.
https://www.ijiris.com/volumes/Vol12/iss-03/01.ISMR26.MRIS10080.pdf
Article Citation: Ruthendiran,Purushothaman,Ragul,Saravanan,Narmada(2026),LiveSyncCloud: A Real-Time File Sharing Web Application,IJIRIS: International Journal of Innovative Research in Information Security, Volume 12, Issue 03 of 2026 pages 83-87 **Doi:>** https://doi.org/10.26562/ijiris.2026.v1203.01
BibTeX Key: Ruthendiran@2026LiveSyncCloud
IJIRIS papers should be cited as IJIRIS (International Journal of Innovative Research in Information Security, AM Publications, India 2026, ISSN 2349-7017, https://doi.org/10.26562/ijiris.2026.v1203.01 The journal's official abbreviation is IJIRIS. Orcid: https://orcid.org/0009-0004-9398-7488

Abstract:: The rapid growth of internet-based communication has increased the demand for efficient real-time collaboration tools. Traditional file sharing systems depend heavily on centralized servers, which often lead to bandwidth congestion, high latency, and potential privacy risks. This research presents LiveSyncCloud, a web-based peer-to-peer communication platform designed to enable real-time file sharing and video communication using WebRTC technology. The system utilizes WebRTC Data Channels to transfer files directly between browsers without relying on centralized storage servers. In addition, WebRTC media streams enable real-time video and audio communication among multiple participants. ANode.js signaling server implemented using Socket.IO manages peer discovery, session management, and connection establishment. The application divides large files into smaller chunks to improve transmission reliability and network efficiency. Experimental evaluation shows that the proposed system can transfer files efficiently while maintaining stable video communication. By eliminating centralized data transfer, LiveSync Cloud reduces server load, improves privacy, and enhances communication performance. The proposed platform can be applied in collaborative environments such as remote learning, distributed teams, and enterprise communication systems.
Keywords: WebRTC, Peer-to-Peer Communication, Real-Time File Sharing, Socket.IO, Video Communication, Web Application.

1. INTRODUCTION

The increasing adoption of remote work, online learning, and distributed collaboration has created a need for efficient real-time communication systems. Organizations and individuals rely heavily on digital platforms for file sharing and communication. Traditional systems such as cloud storage platforms depend on centralized servers to managefiletransfersandcommunicationservices.Althoughcentralizedsystemsprovide reliable infrastructure, they introduce several challenges including server bottlenecks, higher operational costs, and potential privacy concerns. When multiple users attempt to access the same server simultaneously, network congestion can significantly reduce system performance. Web Real-Time Communication (WebRTC) is an emerging technology that enables direct peer-to-peer communication between web browsers. By eliminating the need for intermediate servers for data transfer, WebRTC reduces latency and improves network efficiency. The proposed system, LiveSync Cloud, leverages WebRTC technology to create a decentralized platform for real-time file sharing and video communication. The platform allows users to generate a unique session link through which multiple participants can join and exchange files and media streams.

PEER-TO-PEER COMMUNICATION

Peer-to-peer communication allows devices to communicate directly without relying on centralized servers. In a peer-to-peer architecture, each node acts as both a client and a server, enabling decentralized communication. This approach improves scalability and reduces dependency on centralized infrastructure. Peer-to-peer systems have been widely used in applications such as BitTorrent and distributed computing networks. By adopting a similar approach, LiveSync Cloud ensures efficient file sharing and reduced network latency.

E-ISSN: 2349-7017
P-ISSN: 2349-7009
https://www.ijiris.com/archives
https://doi.org/10.26562/ijiris.2025.v1203.01

WEB RTC TECHNOLOGY

WebRTC is an open-source project developed to enable real-time communication capabilities in web browsers. It supports data, audio, and video transmission through standardized APIs. Web RTC uses several protocols including:

- ICE (Interactive Connectivity Establishment)
- STUN (Session Traversal Utilities for NAT)
- TURN (Traversal Using Relay NAT)

These protocols help establish direct communication channels between devices even when they are located behind network firewalls or NAT configurations.

SIGNALING MECHANISM

WebRTC does not define a built-in signaling mechanism for exchanging connection information between peers. Therefore, an external signaling server is required to exchange metadata such as session descriptions and ICE candidates. In the LiveSync Cloud system, the signaling process is implemented using Node.js and Socket.IO. The signaling server manages room creation, peer discovery, and connection negotiation between participants.

REAL-TIME VIDEO COMMUNICATION

Video communication is an essential component of modern collaboration platforms. WebRTC enables real-time video streaming by capturing audio and video using the get User Media() API. Captured media streams are transmitted directly between peers through encrypted WebRTC connections. This enables low-latency communication suitable for online meetings, collaborative work environments, and remote learning systems.

OBJECTIVES OF THE SYSTEM

The main objectives of the proposed system include:

- To develop a decentralized platform for peer-to-peer file sharing.
- To enable real-time video communication between multiple participants.
- Reduce dependency on centralized servers.
- To provide a user-friendly interface for files haring and communication.
- To ensure secure communication using WebRTC encryption mechanisms.

2. LITERATURE REVIEW

Field inspection and attendance tracking play a vital role in ensuring compliance, efficiency, and accountability in various sectors, including government operations, corporate environments, and infrastructure projects. Existing research and technological advancements have made strides in improving these processes, but challenges remain in integrating real-time tracking, authentication, and role-based access. This section explores previous studies and technological implementations relevant to Trackify's objectives.

PEER-TO-PEER FILE SHARING SYSTEMS

Peer-to-peer file sharing systems have been widely studied in distributed computing environments. Early systems such as Napster and BitTorrent demonstrated the efficiency of decentralized file distribution networks. These systems allowed users to exchange files directly with each other without relying on centralized servers. However, many early P2P systems required dedicated software installations, which limited accessibility.

WEB RTC-BASED COMMUNICATION SYSTEMS

WebRTC technology has gained significant attention due to its ability to provide real-time communication directly within web browsers. Several research studies have explored WebRTC-based video conferencing and data sharing applications. WebRTC enables secure communication using encryption protocols such as DTLS and SRTP, ensuring data confidentiality during transmission.

SIGNALING PROTOCOLS

Signaling protocols play a crucial role in establishing peer-to-peer connections. Various signaling approaches including WebSockets, HTTP polling, and MQTT have been used to exchange connection metadata between peers. Socket.IO has emerged as a popular signaling solution due to its reliability and support for real-time event-based communication.

MULTI-RECEIVER FILE TRANSFER

Transferring files to multiple receivers simultaneously introduces several challenges including bandwidth management and synchronization. Efficient chunk-based file transfertechniquesarecommonlyusedtoimprovetransmissionperformance.Inthis approach, large files are divided into smaller segments which are transmitted sequentially to multiple receivers.

RESEARCH GAP

Although existing systems provide solutions for file sharing or video communication individually, few platforms integrate both functionalities within a single decentralized system. The proposed LiveSync Cloud platform addresses this gap by combining file sharing and video communication within a unified WebRTC-based framework.

3. EXISTINGSYSTEM

Most traditional file sharing platforms rely on centralized cloud servers to store and distribute files. Examples include Google Drive, Dropbox, and other cloud storage services. Although these systems provide reliable file storage and sharing capabilities, they introduce several limitations including increased server load, bandwidth costs, and potential privacy risks. Additionally, many video conferencing platforms rely on centralized media servers to manage video streams. This increases infrastructure complexity and operational costs.

E-ISSN: 2349-7017
P-ISSN: 2349-7009
https://www.ijiris.com/archives
https://doi.org/10.26562/ijiris.2025.v1203.01

4. PROPOSED SYSTEM

The proposed system introduces a decentralized communication architecture that enables direct interaction between users through peer-to-peer connections. Instead of transferring files through centralized servers, the system establishes direct WebRTC connections between users. Once a connection is established, files and media streams are transmitted directly between devices. The proposed system consists of three primary components:

- Frontend client application
- Signaling server
- Peer-to-peer communication network

The frontend interface allows users to select files, generate sharing links, and initiate video calls. The signaling server manages session creation and connection negotiation. Once the signaling phase is completed, direct peer-to-peer communication occurs between users.

FILE TRANSFER MODULE

The File Transfer Module is responsible for enabling users to share files directly between devices in a peer-to-peer manner. Users can select files from their local storage through the web interface. To ensure efficient transmission and prevent network congestion, the selected files are divided into smaller data chunks before being transmitted. These chunks are then sent sequentially through WebRTC Data Channels to connected peers. The receiver reconstructs the original file by combining the received chunks in the correct order. This approach improves reliability and allows large files to be transferred without interruption.

Additionally, the module includes progress indicators that allow users to monitor the status of file transfers in real time.

SIGNALING SERVER

The signaling server plays a crucial role in establishing communication between peers before a direct WebRTC connection is created. Since WebRTC does not include a built-in signaling mechanism, an external server is required to exchange connection metadata. In this system, the signaling server is implemented using Node.js and Socket.IO. It manages session creation, room management, and user identification. When a user joins a session, the signaling server facilitates the exchange of Session Description Protocol (SDP) offers and answers along with ICE candidates between peers.

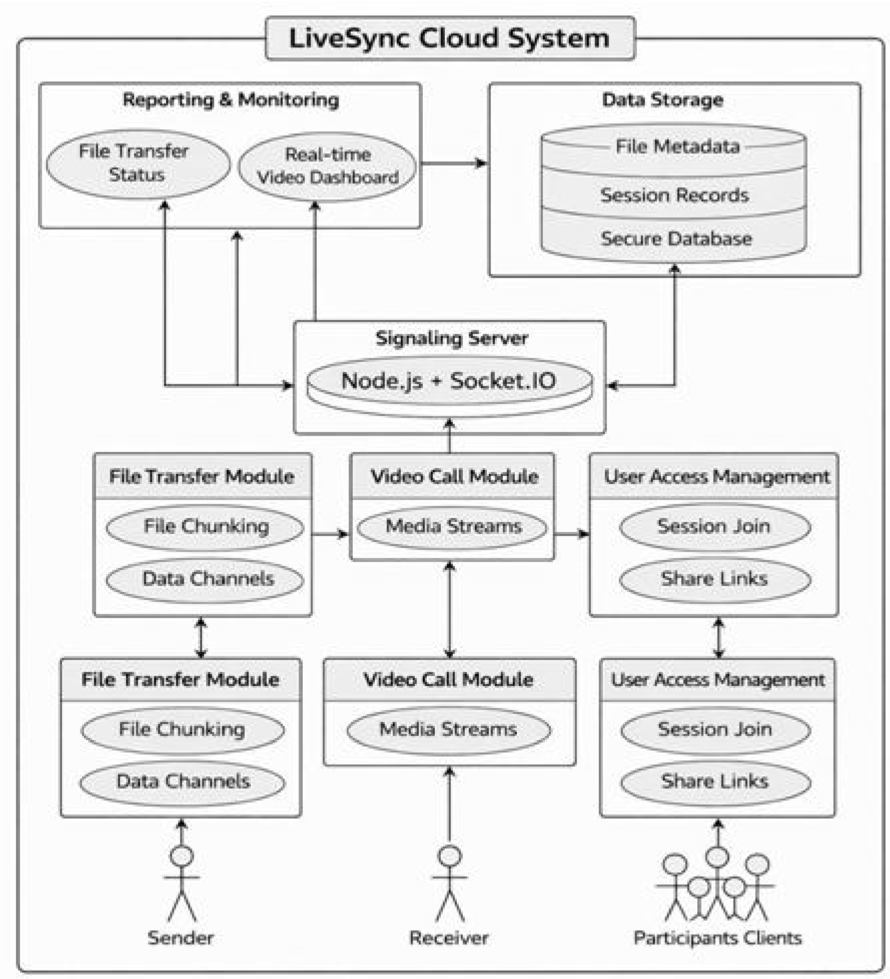

Figure 2. Block diagram

5. CONCLUSION

The LiveSync Cloud system successfully demonstrates the development of a decentralized communication platform that integrates real-time file sharing and video communication using WebRTC technology. The platform enables direct peer-to-peer interaction between users, eliminating the need for centralized data transfer servers. By implementing chunk-based file transfer and WebRTC Data Channels, the system achieves efficient and reliable file transmission. The integration of video communication further enhances collaboration between users. Experimental results indicate that the system provides stable communication and efficient data transfer while maintaining low latency. Overall, the proposed solution highlights the potential of WebRTC technology in building scalable and secure real-time communication systems.

6. FUTURE WORK

Although the current implementation demonstrates effective real-time communication capabilities, several improvements can be considered for future development. One potential enhancement is the integration of mobile applications for Android and iOS platforms to expand accessibility. Another improvement involves implementing advanced encryption mechanisms to provide additional layers of security for file transfers. The system can also incorporate Selective Forwarding Unit (SFU) architecture to support larger groups of participants without performance degradation. Additional features such as screen sharing, user authentication systems, and cloud backup integration can further improve the functionality and usability of the platform. Allow browsers to discover each other and determine the most efficient network path for communication.

PEER CONNECTION ESTABLISHMENT

Once the signaling phase is completed, direct peer-to-peer communication is established using the RTC Peer Connection API. This API enables two or more browsers to create secure communication channels without trelying on centralized data transfer. During this process, each peer verifies the connection parameters and establishes encrypted communication channels using DTLS encryption. Once the connection is successfully created, both peers can exchange data, video streams, and control messages. The peer connection mechanism ensures low latency and high performance because data is transferred directly between users rather than through a central server.

DATA CHANNEL COMMUNICATION

WebRTC DataChannels are used to transmit file data between peers efficiently and reliably. Data Channels provide a bidirectional communication path that supports both ordered and unordered data transmission. Similar to TCP communication, Data Channels ensure that data packets are delivered in the correct order and without loss. This feature is particularly useful for transferring files where data integrity is important. In the LiveSyncCloud system, Data Channels are used to send file chunks between participants while maintaining reliable delivery. The system also supports simultaneous data transmission to multiple receivers, enabling multi-receiver file sharing in collaborative environments.

VIDEO COMMUNICATION

The system also integrates real-time video communication to support collaborative interactions between users. Video and audio streams are captured using the getUserMedia() API, which allows browsers to access the user's camera and microphone with permission. These media streams are then attached to WebRTC peer connections and transmitted to other participants in the session. WebRTC automatically adjusts video resolution and bitrate based on available network bandwidth to ensure smooth communication. This feature enables users to conduct live discussions while simultaneously sharing files within the same platform.

USER INTERFACE DESIGN

The user interface is designed to provide a simple and intuitive experience for users interacting with the platform. The interface allows users to select files, create sharing sessions, and generate unique links for other participants to join. Visual elements such as progress bars, status notifications, and connection indicators help users understand the current state of file transfers and video communication. The interface is developed using modern web technologies including HTML, CSS, and JavaScript. A responsive layout ensures compatibility across various devices and screen sizes, improving usability and accessibility for different users.

SYSTEM WORK FLOW

The workflow of the proposed system begins when a user creates a new sharing session through the application interface. The system generates a unique session link that can be shared with other participants. When new users join the session, the signaling server coordinates the exchange of connection information between peers. After the WebRTC connection is established, users can start transferring files and initiating video calls. The file transfer module sends data chunks through Data Channels, while media streams enable real-time communication. Throughout the process, the system continuously updates the user interface with transfer progress and connection status information.

EXPECTED BENEFITS

The proposed system offers several advantages compared to traditional centralized communication platforms. By using peer-to-peer communication, the system significantly reduces dependency on centralized servers, which lowers infrastructure costs and network congestion. Direct data transmission between peers also improves transfer speed and reduces communication latency. Additionally, WebRTC's built-in encryption ensures secure data transmission and protects user privacy. The system's decentralized architecture improves scalability and reliability, making it suitable for collaborative applications such as remote education, distributed team communication, and secure file sharing environments.

E-ISSN: 2349-7017
P-ISSN: 2349-7009
https://www.ijiris.com/archives
https://doi.org/10.26562/ijiris.2025.v1203.01

REFERENCES

1. C.Nachiappan, “WebRTC and Web Sockets for Peer-to-Peer File Sharing,” IEEE Conference on Advanced Communication Systems, 2024.
2. Q.Duan,Y.Li,andH.Chen, “File Sharing Strategy Based on WebRTC Technology,” IEEE International Conference on Computer Communications, 2016.
3. C.Vogt, “Leveraging WebRTC for Peer-to-Peer Content Distribution in Web Browsers,” IEEE Communications Magazine, vol. 51, no. 4, pp. 34–40, 2013.
4. D.Smith and T.Brown, “Performance Evaluation of WebRTC-Based Video Conferencing Systems,” IEEE International Conference on Multimedia Computing and Networking, 2017.
5. T.Nguyen,“Peer-to-Peer Communication Using WebSockets and WebRTC,” International Journal of Advanced Computer Science and Applications,vol.10,no.2,pp.120–128, 2019.