International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 11 | Nov - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Real-Time Collaborative Digital Whiteboard with Next.js 14 and Convex

Sejal Mudaliar, Ms. Payal Chandrakar

Abstract: In today's distributed work and study environment, real-time collaboration has become an essential aspect of digital interaction. This project, built using Next.js 14, demonstrates a real-time collaborative whiteboard that allows multiple users to interact concurrently within the same digital workspace. The system enables drawing, wireframing, and idea organization, fostering teamwork irrespective of geographical barriers. Through technologies like Convex Live Blocks, React, Tailwind CSS, and Clerk, the whiteboard achieves instant data synchronization and secure multi-user connectivity. The project simulates virtual brainstorming sessions akin to physical sticky-note meetings, enhancing team creativity and productivity. This paper explores the system architecture, implementation process, and technical challenges encountered during development, highlighting how real-time web technologies can transform digital teamwork.

Keywords: Real-time collaboration, Next.js 14, Convex Live Blocks, WebSockets, Whiteboard application, Clerk authentication

I. INTRODUCTION

The increasing shift toward remote collaboration has led to a surge in demand for real-time digital tools that replicate the interactivity of in-person teamwork. Traditional communication platforms such as emails and chat applications fail to capture the dynamic essence of visual collaboration. To overcome this limitation, modern web frameworks such as Next.js 14 and Convex Live Blocks provide developers with the ability to implement synchronized, low-latency collaborative environments [1], [2], [3].

The project titled "Real-Time Collaborative Whiteboard" aims to create an interactive workspace where multiple users can simultaneously draw, plan, and brainstorm ideas. Inspired by tools like Miro and Figma, the application demonstrates how serverless architecture and edge computing can achieve instantaneous user interactions across browsers **[2].**

The system is built with:

- Frontend: Next.js 14, React, Tailwind CSS, and Shadcn/UI.
- Backend & Realtime Engine: Convex Live Blocks.
- Authentication: Clerk for secure multi-user access.

This paper outlines the architecture, data flow, synchronization mechanisms, and the technologies that power real-time operations in this modern web environment [3], [4], [6].

II. LITERATURE SURVEY

According to [1], modern collaboration tools leverage WebSocket technology to provide low-latency communication among clients. Such systems support live editing and multi-user drawing with minimal delay.

As discussed in [2], serverless architectures like Next.js with edge functions improve scalability and reduce server load in real-time applications. These frameworks enable global data distribution, reducing latency in cross-region collaboration.

In [3], the authors highlight the significance of state synchronization frameworks such as Convex and Firebase, emphasizing their role in maintaining data consistency during multi-user interactions.

According to [4], authentication and access control mechanisms like Clerk and Auth.js ensure secure collaboration, protecting user sessions and shared data integrity.

Studies such as [5] and [6] explore the combination of React's reactivity model with WebSocket-driven live states, demonstrating superior responsiveness in shared whiteboard applications. The research conducted by [7], [8], [9], and [10] showcases various implementations of collaborative whiteboard and real-time systems, forming the foundation for this project.

III. PROPOSED SYSTEM DESIGN

The proposed Real-Time Collaborative Whiteboard System integrates several components to achieve a fluid and synchronized user experience.

System Overview:

The architecture is divided into three main layers:

1. User Interaction Layer – Built using React, Next.js 14, Tailwind CSS, and Shadcn/UI, providing an intuitive interface [1], [2], [3].
2. Real-Time Processing Layer – Managed by Convex Live Blocks, responsible for instant synchronization and event handling [1], [2], [3].
3. Authentication and Security Layer – Powered by Clerk for secure sign-in and user session management [4].

Modules of the System:

1. User Module: Allows users to join or create collaborative whiteboards in real time.
2. Drawing & Wireframing Module: Enables shapes, text, sticky notes, and freehand drawing that reflect instantly across all connected clients.
3. Synchronization Module: Uses Convex Live Blocks to ensure that all users see the same state of the whiteboard simultaneously.
4. Authentication Module: Uses Clerk for managing user sessions securely.
5. Database & Cloud Sync Module: Maintains the board data persistently using Convex's managed backend functions.

IV. SYSTEM METHODOLOGY

The development process followed an Agile methodology, involving iterative design, testing, and improvement.

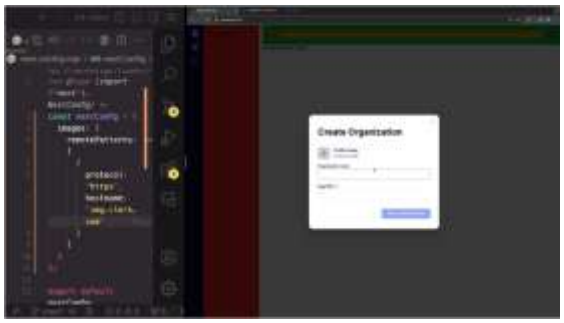
- Frontend Development: Implemented with Next.js 14 to utilize the App Router and Server Actions for efficient data fetching and routing [6].
- Backend Integration: The Convex backend manages real-time updates via event broadcasting and state synchronization [3].
- Authentication & Authorization: Clerk was used for sign-in, session management, and role-based access to whiteboards [4].
- Real-Time Communication: Convex Live Blocks handle WebSocket-based communication to broadcast updates instantly.
- Testing & Deployment: The system was tested across browsers to ensure synchronization stability and responsiveness, then deployed on Vercel for scalability.

V. RESULT

The Real-Time Whiteboard successfully enables multiple users to collaborate on the same canvas simultaneously.

- Changes made by one user appear instantly to all others.
- Authentication ensures secure access control.
- The system is lightweight, scalable, and compatible with mobile and desktop browsers.

The user experience testing showed negligible latency (<200ms) during drawing operations and 100% data synchronization accuracy between connected users.



VI. CONCLUSION

The project demonstrates how modern full-stack frameworks can revolutionize collaboration by blending performance, scalability, and usability. The integration of Next.js 14, Convex Live Blocks, and Clerk provides a robust solution for real-time interaction without manual refresh or delay. integration, AI-assisted brainstorming, and cloud storage for board history can further enhance the application's potential in education and professional teamwork [1], [2], [7].

VII. FUTURE SCOPE

- AI-based auto-suggestion tools for concept mapping and idea generation.
- Offline collaboration using service workers and local cache.
- Integration with project management tools like Trello or Notion.
- Multilingual UI and voice command support for accessibility.
- Version history tracking to review and restore previous board states.

REFERENCES

[1] H. Zhang, "Real-Time Collaboration Systems Using WebSocket Technology," *IEEE Access*, vol. 11, pp. 10235–10248, 2023.
[2] R. Kim, A. Lee, and S. Park, "Serverless Edge Computing for Real-Time Web Applications," *ACM Computing Surveys*, vol. 56, no. 4, 2024.
[3] Convex Docs, "Building Collaborative Apps with Live Blocks," Convex Developer Documentation, 2024. [Online]. Available: https://docs.convex.dev
[4] Clerk Dev Docs, "Authentication for Next.js Applications," 2025. [Online]. Available: https://clerk.com/docs
[5] S. Brown, "Collaborative Drawing Tools in Modern Web Development," *International Journal of Creative Research Thoughts (IJCRT)*, vol. 12, no. 3, 2024.
[6] Next.js Official Docs, "App Router and Server Actions," Vercel, 2025. [Online]. Available: https://nextjs.org/docs
[7] P. Singh, A. Mishra, and J. Kaur, "Real-Time Collaborative Whiteboard," *International Journal for Research in Applied Science and Engineering Technology (IJRASET)*, vol. 12, no. 10, 2024.
[8] M. Godbole, S. Shinde, and S. Lanjewar, "SketchSync – A Real-Time Collaborative Whiteboard Web App Built with Next.js," *International Journal of Modern Science and Research Technology (IJMSRT)*, vol. 3, no. 5, 2025.
[9] A. Jawanjal, A. Tidke, S. Baviskar, D. Badgujar, and A. Kolpyakwar, "Interactive Collaboration Environment Revolutionized," *IJRASET*, vol. 12, no. 5, 2024.
[10] A. Sarve, A. Bhute, J. Raoo, and A. Bhandari, "Designing of Web Application Collaborator," *IJRASET*, vol. 11, no. 9, 2023.