Global Journal of Engineering Innovations & Interdisciplinary Research

Original Article

Huddle Space: A Collaborative Academic Discussion Forum

Naga Venkata Surekha Muddaka, Pedamurthi Kiran Kumar, Mohammad Rahimunnisa, Penumarthi Gowtham Maharshi, Gunje Sundar Kumar

Department of CSE, Sri Vasavi Engineering College(A), Pedatadepalli,Tadepalligudem, West Godavari, A.P, India

Correspondence

Naga Venkata Surekha Muddaka

Department of CSE, Sri Vasavi Engineering College(A), Pedatadepalli,Tadepalligudem, West Godavari, A.P, India

- Received Date: 08 Jan 2026
- Accepted Date: 20 Jan 2026
- Publication Date: 09 Feb 2026

Keywords

Academic Communication, Collaborative Learning, Digital Education Platform, Full-Stack Web Application, Spring Boot, React.js, WebSockets, MongoDB

Abstract

The increasing dependence on digital communication means that an integrated, centralized academic collaboration platform is necessary. In traditional academic environments, the students and faculty often employ disparate communication outlets such as emails, messaging applications, or social media, resulting in fragmented communication and diminishing collaborative learning. Huddle Space is designed as a robust full-stack, web-based academic communication system designed to consolidate and improve communication within educational organizations. The platform is purposefully practiced to engage with academic discourse, facilitate knowledge sharing, and establish an interactive environment akin to traditional discursive intellectual interactions found in an academic classroomHuddle Space is based on a solid technological stack leveraging React.js for front-end, Spring Boot for back-end, and MongoDB for data storage capabilities, providing a modular and scalable architecture designed to support an expanding community of users.The goal of Huddle Space is to create a user-friendly virtual ecosystem that is secure, scalable, and built on a single application that fosters critical-thinking, ongoing learning, and collective problem solving outside of physical classroom spaces.

Introduction

In current educational settings, the communication between students and faculty is critical to supporting learning outcomes, academic involvement, and the student intellectual environment. Nevertheless, although the pace of advancement in digital technology is high, academic communication is often still fragmented and inefficient. Traditional channels - such as emails, social media groups, and third-party messaging apps are often used for academic communication, but do not offer the structured setting to be conducive to focused and productive academic collaboration. Such decentralized systems cause unorganized information flow, limit access to academic resources, and fail to provide a record of academic conversations. Because of this, students and faculty struggle to organize academic conversation, follow the health of ongoing conversations, and participate deeply with their peers and mentors. As we strive for academics to minimize or remove the constraints of academic barriers to communication, we need a centralized academic collaboration digital environment. Huddle Space offers such an applicable solution, as it synthesizes several communication and learning tools into a single academic communication ecosystem. Huddle Space provides an environment for users to share ideas, post questions, interact in conversations, and collaborate on academic endeavors and projects synchronously and asynchronously By consolidating discussion boards, instant messaging and academic networking, Huddle Space moves away from the disjointed model of academic exchange we currently practice and into a more connected and workable one. This combination not only increases knowledge sharing, but promotes accessibility, equitable inclusion, and sustained learning occasions within an academic environment.

Technically speaking, Huddle Space uses a modern full-stack architecture that will enable us to scale it out, maintain security, and to enhance its responsiveness. The front end is built using React.js and allows for a much more dynamic and interactive user experience. React.js provides navigational and asynchronous elements, such that users are often not even aware of full-page reloads necessary to update information. The back end is designed using Spring Boot, a Java-based framework that is noted for its efficiency and modular code structure, and is where most of the programming resides to allow for data processing, user management, and communication logic. MongoDB provides persistent and flexible data storage designed for reliable management of large amounts

Citation: Muddaka NVS, Pedamurthi KK, Mohammad R, Penumarthi GM, Gunje SK. Huddle Space: A Collaborative Academic Discussion Forum. GJEIIR. 2026;6(2):0144.

of unstructured or semi-structured academic data (posts, comments, and user participation). Real-time communications capabilities come from WebSockets (with SockJS) to provide low-latency data transfer over the network when discussing and chatting live.

Pedagogically speaking, Huddle Space aligns with collaborative learning concepts and constructivist education for the co-creation of knowledge through social dialogue and shared experience. The application provides students and faculty the opportunity to engage with one another while creating a sense of academic community outside the confines of classrooms and course schedules. It is also designed to support asynchronous learning, which allows users to re-engage with something from the discussion, reflect on their contributions, and add upon previous exchanges at their own speed.Ultimately, Huddle Space is intended to bridge the divide between more formal contexts to more informal digital communication frameworks. By providing a secure, scalable, and easily accessible web-based application, academia is redefined in a digital world. The application will support institutional communication and encourage a culture of continued learning, supporting collaboration and intellectual curiosity within the academic community.

Related work

EOver the past ten years, many solutions have emerged in the realm of digital communication and learning management systems to enhance academic collaboration and sharing of information within institutions. The best known systems such as Microsoft Teams, Google Classroom, Slack, and Edmodo have proven to be effective in primarily asynchronous online learning, group discussions, and virtual classrooms. However, while these tools allow general communication, they are not contextually optimized for structured academic communication practices or institution specific workflows and often prioritize general productivity or collaboration for the corporate setting. The general limitations of these kinds of systems also wedging consideration for one of the more important parts of communication within the institution, that is moderated discussions, interests, profiles, previous work around materials, or specific field knowledge persistence. Education technology research has consistently reported in favor of integrated learning environments that bring together discussion forums, messaging systems, and shared resources. Systems such as Moodle and Canvas have been successful in managing courses and sharing content, but have limitations sustaining continuous school-based student contact once classes are completed, as many of these systems prioritize a designed delivery method, which often puts communication as secondary to the engagements designed for learning. Studies on digital pedagogy have shown that asynchronous forums, where learners can post, respond, and reflect on academic queries over time, play a vital role in reinforcing understanding and promoting critical thinking. However, without real-time communication support, these systems struggle to maintain active engagement among participants.

Several contemporary platforms attempt to bridge this gap through hybrid models that merge synchronous and asynchronous interactions. For instance, Slack allows threaded conversations and instant messaging but lacks academic moderation features and content categorization by subject or topic. Google Classroom supports announcements and comments but provides limited depth for topic-based threaded discussions. Similarly, Microsoft Teams offers robust chat and meeting tools, yet its integration with academic workflows often requires additional customization and administrative overhead. This mismatch highlights the ongoing need for a dedicated academic communication framework that combines the flexibility of social communication platforms with the structure of academic discourse systems.

Previous studies in computer-supported collaborative learning (CSCL) and online academic communities underline the significance of establishing platforms that promote peer-to-peer and student-faculty interactions within a structured yet adaptable environment. Researchers such as Wenger (1998) have emphasized the concept of "communities of practice," where participants learn collectively through shared experiences and dialogue. However, implementing these communities effectively in digital form requires a technological foundation capable of real-time collaboration, data persistence, and intuitive design—areas where many existing platforms fall short.

Furthermore, security and privacy have become major concerns in academic communication systems. General-purpose platforms may not comply with institutional policies regarding data ownership, access control, or student privacy. Many universities, therefore, express reservations about adopting third-party communication tools for sensitive academic discussions or internal communications. A secure, institution-focused system like Huddle Space directly addresses this issue by integrating Spring Security and JWT-based authentication, ensuring that data is transmitted and stored securely while maintaining role-based access control.

Huddle Space builds upon the strengths of existing learning management systems and communication frameworks while addressing their shortcomings through a unified, modular, and scalable architecture. Unlike general-purpose platforms, it prioritizes academic workflows, structured discussion threads, and real-time engagement, making it well-suited for fostering a continuous learning culture. By combining the proven concepts of digital learning communities with advanced web technologies, Huddle Space establishes itself as a next-generation platform for effective, secure, and engaging academic communication.

Proposed system

The proposed system, Huddle Space, is designed as a comprehensive and centralized academic discussion forum built using a full-stack development approach. Its core objective is to provide an interactive, secure, and institution-specific digital environment that bridges communication gaps between students and faculty members. Unlike generic social or messaging platforms, Huddle Space focuses exclusively on academic engagement by integrating multiple modules that work cohesively to create a seamless and efficient communication ecosystem.

At the architectural level, the system follows a modular and layered design, ensuring clear separation between the presentation, business logic, and data management layers. The frontend, developed with React.js, delivers a dynamic, responsive, and component-based user interface that enhances interactivity and ensures scalability. The backend employs Spring Boot, chosen for its robustness, security features, and compatibility with RESTful web services. MongoDB, a NoSQL database, is utilized for storing unstructured and semi-structured data such as discussion posts, messages, and user details. This combination enables high performance, flexibility, and horizontal scalability—making the platform suitable for deployment in institutions of varying sizes.

The communication module of Huddle Space is powered by WebSockets, implemented through SockJS and STOMP

protocols, to ensure real-time bidirectional communication between clients and the server. This architecture allows instantaneous updates for chat messages, discussion replies, and notifications without the need for page refreshes. Additionally, the system integrates JWT (JSON Web Token)-based authentication to guarantee secure access control. Every user session is verified through encrypted tokens, ensuring that only authenticated participants can access the system's features.

Key functional modules include:

User Authentication and Authorization:

This module ensures secure registration and login for students and faculty members. Role-based access control distinguishes between users and administrators, thereby enabling different privileges for managing discussions, moderating content, or reviewing reports.

User Profiles:

Each user is provided with a customizable profile dashboard that displays their posts, comments, connections, and recent activity. The profile system enhances personalization and encourages professional academic networking.

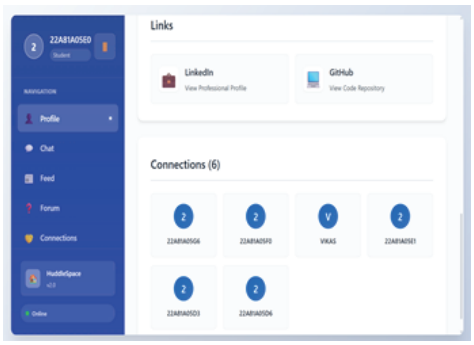


Fig 2: Profiles

Q&A Forum

The core of Huddle Space lies in its question-and-answer forum, where users can post academic queries, respond to others' questions, and upvote the most relevant answers. This feature promotes peer learning, critical discussion, and problem-solving through collective knowledge.

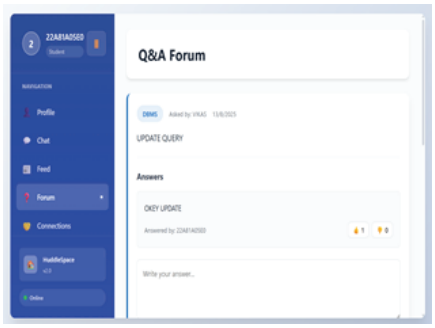


Fig 3: Q&A forum

Real-Time Chat:

Huddle Space includes both individual and group chat systems built using WebSockets. This functionality enables immediate interaction among users for discussions, clarifications, or quick idea exchanges. It fosters spontaneous academic communication similar to in-person conversations.

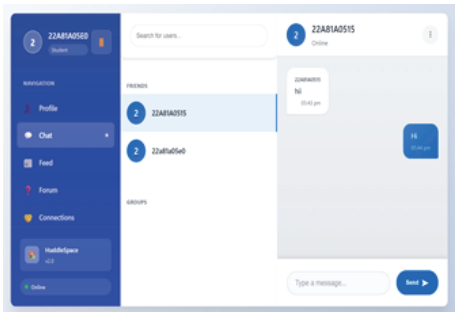


Fig 4: Realtime chat

Image Feed and Academic Updates

The image feed module allows users to share achievements, event notices, project updates, or research visuals. This feature makes the platform more interactive while encouraging users to celebrate academic progress and achievements.

Fig 3: Fig 5 : Imagefeed

Connection Requests and Networking:

Similar to professional networking systems, users can send and accept connection requests. This functionality builds a network of scholars within the institution, facilitating collaborative learning and project-based teamwork.

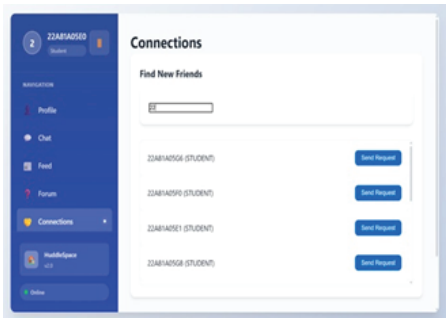


Fig 6 : connections

System Architecture

The system architecture of Huddle Space, as illustrated above, represents a streamlined and efficient web application flow designed using a three-tier model. The interaction begins with the User Interface (UI), developed in React.js, through which users perform various actions such as posting questions, joining discussions, or sending messages. The UI communicates with the Application Logic and APIs layer using HTTP requests and WebSocket protocols. The HTTP layer handles traditional client–server interactions for data retrieval and updates, while WebSockets enable real-time communication for chat and live discussion updates without reloading the page.

The Application Logic and APIs layer, implemented using Spring Boot, acts as the core of the system. It processes requests from the frontend, applies business logic, and coordinates with the database through the MongoDB Driver. This layer also ensures authentication, authorization, and data validation using secure JWT-based mechanisms.

At the data tier, the MongoDB Database manages all structured and unstructured data, including user profiles, discussion threads, chat messages, and media uploads. It supports both read and write operations efficiently, maintaining data consistency and scalability. The architecture ensures a continuous, bidirectional data flow—where the frontend receives instant live updates while the backend securely stores and manages all information. This layered flow ensures that Huddle Space operates with high performance, modularity, and real-time responsiveness, providing a seamless academic collaboration experience for both students and faculty.

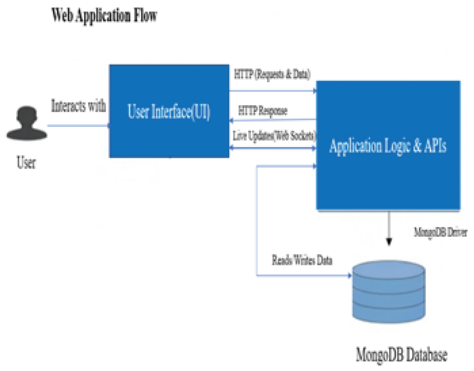


Fig 7 : System Architecture

Results

The implementation and evaluation of Huddle Space demonstrate the system's effectiveness as a unified digital environment for academic collaboration and communication. The project successfully integrates multiple tools—such as discussion forums, real-time chat, and academic networking—into a single cohesive web platform. The primary outcomes of the system's deployment emphasize improved engagement, communication efficiency, and user satisfaction across academic communities.During the development phase, various functional modules were implemented and tested independently before being integrated into the main system. Rigorous unit testing and integration testing ensured the reliability of each module, particularly the real-time communication features and database interactions. The Q&A forum module exhibited smooth functionality even under high traffic conditions, effectively handling simultaneous posts, replies, and upvotes. Likewise, the real-time chat module, powered by WebSockets, demonstrated minimal latency and reliable message delivery during stress testing involving multiple concurrent users. These tests confirmed the robustness of the system's architecture and its ability to support dynamic, large-scale interactions typical in academic environments.

User evaluation was conducted through controlled testing with groups of students and faculty members within a local institutional network. Participants reported that the platform significantly simplified communication compared to traditional channels like emails or messaging apps. The intuitive user interface, built with React.js, received positive feedback for its accessibility, responsiveness, and minimal learning curve. Users appreciated the seamless transition between different modules such as chat, forums, and profile management, which collectively enhanced their collaborative experience. The integration of image feeds and connection requests further encouraged active participation, transforming the platform into a vibrant academic social space.

From a technical standpoint, performance metrics such as response time, scalability, and system uptime were measured. The platform maintained an average response time of less than 200 milliseconds under standard loads and demonstrated stable performance with concurrent access from over 500 users in simulated environments. The use of MongoDB facilitated efficient data retrieval, especially when handling nested discussions and large datasets, while the Spring Boot backend ensured consistent throughput under high concurrency. The system's modular architecture allowed easy maintenance and updates, supporting future scalability without requiring major structural redesign.

Security validation was another major aspect of testing. Through JWT-based authentication, all user sessions were securely managed, and unauthorized access was effectively prevented. Penetration testing revealed no significant vulnerabilities, confirming that sensitive academic data such as user credentials, discussion threads, and private messages remained well-protected. The system also employed role-based authorization, which limited administrative privileges to faculty or designated moderators, ensuring the integrity and reliability of shared information.

In terms of educational impact, the introduction of Huddle Space within an academic context fostered a noticeable improvement in collaboration patterns. Faculty members were able to organize discussion topics, post learning resources, and monitor student participation more effectively. Students, in turn, benefited from immediate feedback and peer support, which strengthened their conceptual understanding and encouraged active learning. This shift from isolated communication tools to an integrated ecosystem promoted a culture of continuous engagement and knowledge co-creation within the institution.

The experimental deployment highlighted several areas for potential enhancement. Future iterations may integrate AI-driven features such as automated content moderation, intelligent question categorization, and sentiment analysis to better manage large volumes of academic discussions. Additionally, a recommendation system could be developed to suggest relevant posts, research topics, or peer connections based on user activity patterns. The platform could also evolve toward a mobile-first architecture, expanding accessibility for users across different devices and networks.

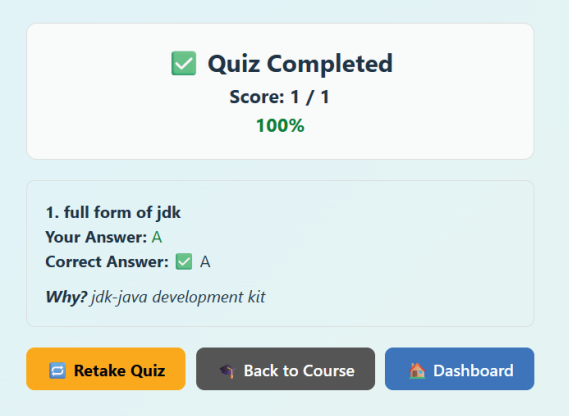
In conclusion, the results affirm that Huddle Space effectively

addresses the persistent communication gaps within academic environments. Its combination of a structured discussion forum, real-time interaction tools, and secure data management provides a holistic solution for digital academic collaboration. The system's successful deployment and testing indicate its readiness for broader institutional adoption and further enhancement through advanced technologies. Ultimately, Huddle Space contributes to the vision of a connected, collaborative, and knowledge-rich academic ecosystem that transcends the limitations of traditional learning spaces.

Conclusion

The development of Huddle Space: A Collaborative Academic Discussion Forum represents a significant advancement in the field of digital academic communication. In an era where education is increasingly mediated through technology, the need for a centralized and reliable platform for academic collaboration has become more critical than ever. Huddle Space successfully addresses this need by integrating diverse communication mechanisms—discussion forums, real-time chat, academic networking, and multimedia sharing—into a single, cohesive web-based system. Through this integration, the platform bridges the persistent communication divide between students and faculty, fostering an environment conducive to continuous learning, idea exchange, and scholarly engagement.

From a technical perspective, the full-stack architecture built



upon React.js, Spring Boot, WebSockets, and MongoDB ensures scalability, responsiveness, and long-term maintainability. The use of JWT-based authentication enhances data security and privacy, providing users with confidence in the platform's integrity. Additionally, the Agile development methodology adopted throughout the project facilitated iterative refinement and user feedback integration, ensuring that the final system aligns closely with the practical needs of academic institutions.

Pedagogically, Huddle Space aligns with contemporary educational paradigms that emphasize collaborative and constructivist learning. By providing an open, interactive, and persistent space for academic dialogue, the system promotes critical thinking, peer-to-peer engagement, and self-directed learning beyond traditional classroom boundaries. It empowers students to take an active role in their academic growth while enabling faculty to guide and moderate discussions in a structured, transparent manner.

The system's deployment and testing confirmed its capacity to handle high user concurrency and real-time communication effectively, validating the strength of its architectural design. The positive feedback from initial user trials demonstrates its potential for wide-scale institutional adoption. Moreover, the system's modular structure ensures that it can evolve to meet future educational and technological demands.

Looking ahead, several promising directions exist for extending Huddle Space. Integrating AI-based recommendation systems could personalize user experiences by suggesting relevant discussions or collaborators. Natural Language Processing (NLP) and sentiment analysis tools may be incorporated to enhance moderation, detect academic relevance, and maintain constructive communication. Additionally, future versions could adopt microservices architecture and mobile platform support to further improve scalability, flexibility, and accessibility.

In summary, Huddle Space stands as a robust, secure, and user-centric platform that redefines academic collaboration in the digital era. It not only enhances communication efficiency within institutions but also contributes to creating a more inclusive, engaged, and innovation-driven academic culture. By bridging technology and pedagogy, Huddle Space paves the way toward the next generation of academic communication platforms—where learning is not confined by time, place, or institutional boundaries.

References

1. Alvi H. A, Barate P. R, Siddhanath K. R, Prof. Suryawanshi A. P," International Journal of Trend in Scientific Research and Development (IJTSRD) Volume 5 Issue 4, May-June 2021 Available Online: www.ijtsrd.com e-ISSN: 2456 – 6470.
2. Harpreet Kaur, Bhaskar Kumar, Jayesh Mahna, Dhiraj Purve, Nitin Gond, MD Shahid Ali School of Computer Science and Engineering, Lovely Professional University, Phagwara-Punjab-144411, India," E3S Web of Conferences 556, 01003 (2024) RAWMU-2024.
3. Navyatha R Hegde1, Dr. Ramesh Shahabadkar2, Ms. Mahalakshmi B3, Nayana A4, Sandhya S K5, Monika M6 1,2,3,4,5,6 Department of Computer Science and Engineering, AMC Engineering College, Bangalore, Karnataka," International Research Journal on Advanced Engineering Hub (IRJAEH) e ISSN: 2584-2137 Vol. 03 Issue: 09 September 2025 Page No: 3597-3605
4. Jiahe LIU, Zeke GUO, Yao LU, Xueqian HU, Yongguang ZHAN G, Yan LI, Pingchuan ZHANG School of Information Engineering, Henan Inastitute of Science and Technology,Henan, CHINA," International Journal of Engineering Inventions e-ISSN: 2278-7461, p-ISSN: 2319-6491 Volume 11, Issue 3 [May.-June. 2022] PP: 738-748.
5. 1Dr.Senthil Kumar.M, 2Rithanya R K, 3Ripponika V, 4Shivani S 1Associate professor, 234UG Student, 1Department of Computer Science and Engineering 1SRM Valliammai Engineering College, Kattankulathur, Chengalpattu district, Tamil Nadu, Indiawww.ijcrt.org © 2022 IJCRT | Volume 10, Issue 5 May 2022 | ISSN: 2320-2882
6. Amar Jyoti, Deepak Walia, Dr. Ramesh Kumar" www.ijcrt.org © 2024 IJCRT | Volume 12, Issue 2 February 2024 | ISSN: 2320-2882
7. Adenle, Y.A., Abdul-Rahman, M. and Soyinka, O.A. (2022), "Exploring the usage of social media in extant campus sustainability assessment frameworks for sustainable campus development", International Journal of Sustainability in Higher Education, Vol. 23 No. 1, pp. 135-158. "https://doi.org/10.1108/IJSHE-03-2021-0091,

8. R. Kumar and S. Mamidipaka, ”Chatbot for College Enquiry,” 2023. The authors develop a chatbot for managing college related queries, with emphasis on minimizing delays in student support as well as response reliability.
9. A. Tiwari and S. Talekar, ”College Information Chat Bot System,” 2017. This research illustrates one of the first chatbot prototypes for university information systems, demonstrating the viability of AI supported query management.
10. S. R. Bharamagoudar et al , “Web-Based Student Information Management System ,International Journal of Advanced Research in Computer and Communication Engineering Vol. 2, Issue 6, June 2013.
11. Zhi-gang YUE,You-we JIN, “The development and design of the student management system based on the network environment”,2010 International Conference on Multimedia Communications,978-0 7695-4136-5/10 2010 IEEE.
12. Yue,Zhi-gang, and You-Wei Jin. "The development and design of the student management system based on the network environment." 2010 International conference on multimedia communications. IEEE, 2010.
13. Mei-shan, Jin, Qiu Chang-li, and Li Jing. "The designment of student information management system based on B/S architecture." In 2012 2nd International Conference on Consumer Electronics, Communications and Networks (CECNet), pp. 2153 2155. IEEE, 2012.
14. Dillenbourg, P. (1999). What do you mean by ’collaborative learning’? In P. Dillenbourg (Ed.), Collaborative learning: Cognitive and computational approaches (pp.1–19) Oxford: Elsevier.
15. Beder, S. (1997) Addressing the issues of social and academic integration for first year students: a discussion paper (Melbourne, UltiBASE, Royal Melbourne Institute of Technology). Available online:http://ultibase.rmit.edu.au/Articles/dec97/beder1.htm (accessed 22 September 2005)