Mukesh Kumar Kasireddy, 2026, 14:2
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology
An Open Access Journal

AI-Driven Project Collaboration and Team Management Platform

Mukesh Kumar Kasireddy[1], N Nikhil Kumar Reddy[2], N Srinivas[3], Dr. Sasikumar Gurumoorthy[4]

[1,2,3]UG Student, Department of Computer Science and Engineering, School of Engineering and Technology, Dhanalakshmi Srinivasan University, Trichy-621112-Tamilnadu.

[4]Professor, Department of Computer Science and Engineering, Dhanalakshmi Srinivasan Institute of Technology, Trichy-621112- Tamil Nadu

Abstract- Modern student projects require collaboration among members with different technical skills such as frontend development, backend development, database management, artificial intelligence, and UI/UX design. However, students often face challenges in forming effective teams, coordinating tasks, managing communication, and integrating project components using multiple disconnected tools. This paper proposes an AI-Driven Project Collaboration and Team Management Platform designed to provide a unified environment for student teams to collaborate efficiently. The platform integrates team formation, real-time communication, task management, document sharing, and project monitoring into a single system. Artificial intelligence is incorporated to recommend compatible teammates based on skills and interests, extract tasks from discussions, summarize conversations, and provide code suggestions. The proposed system utilizes a modern web architecture consisting of a React.js frontend, Spring Boot backend, PostgreSQL database, and WebSocket-based real-time communication. AI services powered by large language models are used to analyze chat data and assist in project management. The platform improves productivity, reduces coordination overhead, and provides students with an industry-like collaborative development environment.

Keywords: AI Collaboration Platform, Team Formation, Project Management, Natural Language Processing, Student Collaboration, Real-Time Systems

I. INTRODUCTION

Collaborative software development has become an essential part of modern computer science education. Many academic projects require students to work in teams where each member contributes specialized skills such as programming, system design, documentation, or testing. Despite the importance of teamwork, students often rely on separate tools such as messaging applications, cloud storage services, and version control platforms to coordinate their work. This fragmented approach leads to inefficiencies, communication gaps, and integration challenges.

Research has shown that communication and collaboration tools significantly influence productivity in software development environments [2]. Another common problem in student projects is the difficulty in forming balanced teams. Students may struggle to identify teammates who possess complementary technical skills or share similar interests. Without proper coordination and task management, project responsibilities may be unevenly distributed, resulting in reduced productivity and delayed completion.

Recent advances in artificial intelligence and natural language processing offer new opportunities to improve collaborative development environments. AI systems can analyse user profiles, chat discussions, and project activities to generate useful insights such as task recommendations, discussion summaries, and workload predictions [1].

This research proposes an AI-Driven Project Collaboration and Team Management Platform that integrates collaboration tools with intelligent

assistance features. The system enables students to create profiles, form teams, manage projects, communicate in real time, and track project progress within a single unified environment. By combining collaboration tools with AI- based analytics, the proposed system aims to improve efficiency and provide a more structured development experience for student teams.

II. RELATED WORK AND EXISTING SYSTEMS

Several digital platforms support collaboration and project management in software development environments. Tools such as GitHub, Jira, Trello, and Slack are widely used for managing tasks, communication, and version control.

GitHub provides powerful version control and collaboration capabilities for software development teams. However, it primarily focuses on code management and does not provide built-in mechanisms for intelligent team formation or task extraction from discussions.

Project management platforms such as Trello and Jira provide task tracking and workflow management features. These systems allow teams to create tasks, assign responsibilities, and track progress. While effective for structured project management, they do not provide advanced AI-based features for analyzing team interactions or recommending tasks automatically.

Communication platforms such as Slack and Microsoft Teams enable real-time messaging and collaboration. These platforms support file sharing and integration with external tools, but they typically lack built-in intelligence for extracting project insights from discussions. Studies indicate that such communication platforms play an important role in facilitating coordination among distributed development teams [2].

Recent research has explored the application of artificial intelligence in collaborative environments. Natural language processing techniques such as transformer- based language models have demonstrated strong capabilities in analyzing textual communication and generating summaries [7].

However, many existing solutions remain fragmented, requiring users to switch between multiple applications for communication, task management, and collaboration. There is a need for an integrated platform that combines these functionalities with intelligent AI assistance to improve team productivity.

III. PROPOSED SYSTEM

The proposed system is an AI-Driven Project Collaboration and Team Management Platform designed to support student teams throughout the project lifecycle. The platform integrates multiple collaboration tools into a single system while incorporating AI features to improve productivity and decision-making.

A. User Profile Management

Students begin by creating user profiles that include information such as technical skills, interests, preferred roles, past projects, and external links such as GitHub, LinkedIn, and portfolio websites. These profiles provide structured data that can be used by the AI system to recommend suitable teammates and project roles.

B. Ai-Based Team Formation

The platform includes an intelligent team formation module that analyses user profiles to identify compatible team members. AI algorithms evaluate factors such as skill compatibility, project interests, and availability to generate team recommendations. Students can accept AI suggestions or manually send collaboration requests to other users.

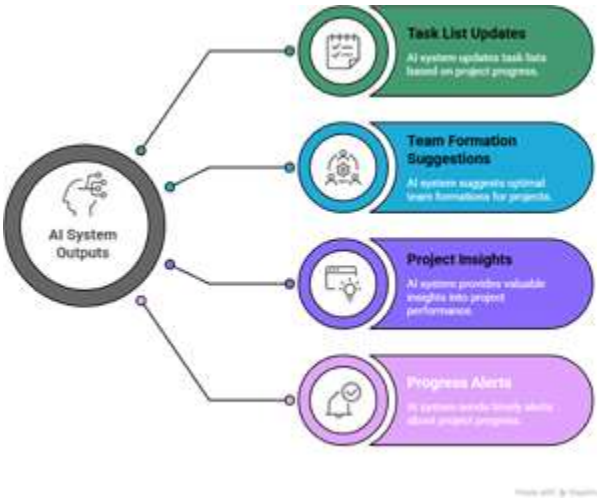
C. Project Workspace

Once a team is formed, a dedicated project workspace is created. This workspace provides modules for task management, document sharing, real-time communication, and project monitoring. Team members

can assign tasks, track progress, and collaborate on project activities within a single environment.

D. Ai Assistant

An AI assistant is integrated into the platform to support project collaboration. The AI system analyses chat discussions and project data to extract tasks, summarize conversations, and provide coding suggestions. It also generates insights about project progress and identifies potential delays or workload imbalances.



IV. COLLABORATION PLATFORM DEVELOPMENT FRAMEWORK

The development of the proposed platform follows a modular system framework that integrates collaboration tools with artificial intelligence services.

The system begins with a user profile management component that stores information about each user's skills, interests, and project experience. This structured profile data forms the foundation for AI-based team formation and collaboration recommendations.

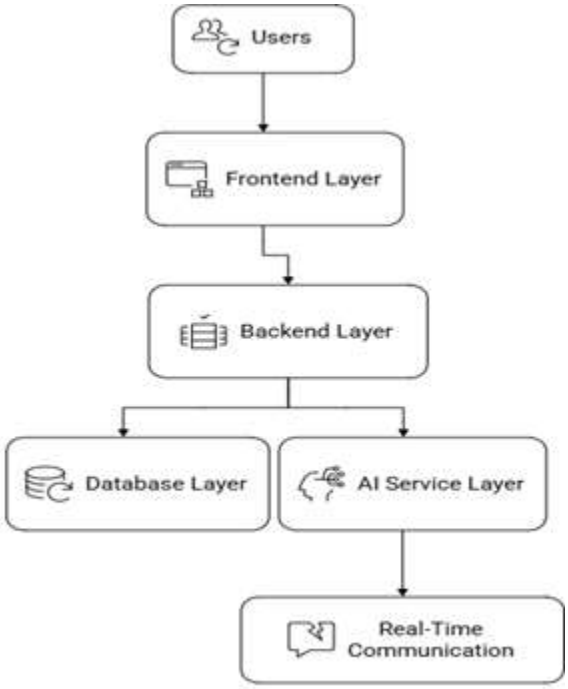
Once user profiles are established, the platform provides a team formation mechanism that allows students to form teams either manually or through AI-generated recommendations based on skill compatibility.

After teams are formed, a project workspace is created for each team. The workspace provides tools for task management, document sharing, communication, and project monitoring.

Artificial intelligence modules continuously analyse communication data and project activities to generate insights such as task suggestions and discussion summaries. This modular framework ensures that collaboration tools and AI services work together to support efficient project development.

V. SYSTEM ARCHITECTURE

The proposed platform follows a multi-layer architecture consisting of frontend, backend, database, and AI service layers.



The frontend layer is implemented using React.js, providing an interactive user interface that allows students to manage profiles, teams, tasks, and project workspaces. The dashboard interface displays project progress, task assignments, and team activity in real time.

The backend layer is developed using the Spring Boot framework, which handles business logic, user authentication, team formation, project management APIs, and communication services.

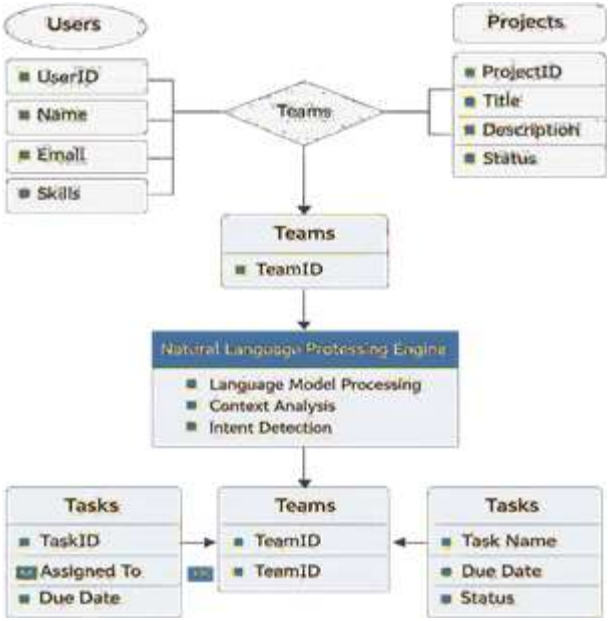
The database layer utilizes PostgreSQL to store structured data including user profiles, team information, project details, tasks, files, and chat messages.

The real-time communication layer uses WebSocket technology to enable instant messaging, live notifications, and collaborative updates between team members.

The AI service layer integrates large language model APIs such as Google Gemini to analyse chat messages and project data. This layer provides features including task extraction, discussion summarization, and intelligent recommendations.

VI. COLLABORATION PLATFORM MODULES

The proposed system is composed of several modules that work together to support project collaboration.



The User Management Module handles user registration, authentication, and profile management. It stores user skills and preferences in the database.

The Team Formation Module analyses user profiles and generates recommendations for forming balanced teams based on skill compatibility. The Project Management Module allows teams to create projects, define milestones, assign tasks, and track project progress.

The Communication Module provides real-time chat functionality, enabling team members to communicate and share files within the project workspace.

The AI Recommendation Module processes chat discussions and project data to extract tasks, generate summaries, and provide intelligent suggestions. The Analytics Dashboard Module visualizes project performance through charts and progress indicators, allowing teams to monitor productivity and identify potential issues.

VII. IMPLEMENTATION AND TECHNOLOGIES

The proposed system is implemented using a modern full-stack web development framework.

The frontend interface is developed using React.js, which provides a responsive and interactive user experience. The backend services are implemented using Java Spring Boot, which supports scalable API development and secure user authentication.

The system stores data in a PostgreSQL database, which ensures efficient storage and retrieval of structured project information. Real-time communication between users is enabled using WebSockets, allowing instant updates in chat messages and task changes.

Artificial intelligence capabilities are integrated through Google Gemini API, which enables natural language processing for analyzing chat messages, generating summaries, and extracting tasks.

Version control and collaborative development of the platform itself are managed using GitHub.

VIII. CURRENT DEVELOPMENT PRACTICES AND LIMITATIONS

In traditional student project environments, collaboration is often managed using a combination of independent tools such as messaging applications, cloud storage platforms, and project management software. These tools typically operate independently and require users to manually coordinate activities across multiple systems.

Such fragmented workflows create several limitations. Teams may experience difficulties in tracking project progress, managing tasks efficiently, and maintaining clear communication among members. Additionally, the absence of intelligent automation means that project discussions often contain valuable information that remains unstructured and unused.

Existing project management platforms also lack mechanisms for automatically analyzing communication data to extract tasks or summarize discussions. As a result, students must manually interpret conversations and convert them into actionable tasks.

These limitations highlight the need for a more integrated system that combines collaboration tools with intelligent AI-based assistance.

IX. FUTURE EVOLUTION OF THE COLLABORATION PLATFORM

The proposed collaboration platform represents an initial step toward intelligent project management systems that integrate artificial intelligence with collaborative workflows.

Future versions of the platform may incorporate more advanced AI capabilities such as automated workload balancing, predictive project risk analysis, and intelligent task prioritization. These features could enable the system to identify potential project delays and recommend adjustments to task assignments.

Another potential evolution involves integrating knowledge-graph representations of skills and technologies. Such representations would allow the system to understand relationships between different technical domains and recommend learning paths or alternative team roles for users.

The platform may also evolve to support larger collaborative environments involving multiple teams working on interconnected projects. In such environments, AI-based analytics could provide insights into collaboration patterns and knowledge sharing across teams.

X. FUTURE RESEARCH DIRECTIONS

Future enhancements to the system may include additional AI features such as advanced project risk prediction, automated code review assistance, and integration with external development platforms such as GitHub repositories.

The platform can also be extended with real-time collaborative coding environments and mobile

application support to improve accessibility for users.

Further research will focus on improving the accuracy of AI-based recommendations and expanding the system to support larger collaborative environments.

XI. CONCLUSION

This paper presented an AI-Driven Project Collaboration and Team Management Platform designed to improve teamwork in student software development projects. The proposed system integrates team formation, project management, real-time communication, and AI- based analytics within a unified environment.

By analyzing user profiles and project discussions, the platform provides intelligent assistance for team formation, task extraction, and project monitoring. The integration of modern web technologies and AI services enables efficient collaboration while providing students with an industry-like development experience.

REFERENCES

1. T. H. Davenport and R. Ronanki, "Artificial Intelligence for the Real World," Harvard Business Review, vol. 96, no. 1, pp. 108–116, Jan.–Feb. 2018.
2. M. A. Storey, A. Zagalsky, F. F. Filho, L. Singer, and D. M. German, "How Social and Communication Channels Shape and Challenge a Participatory Culture in Software Development," IEEE Transactions on Software Engineering, vol. 43, no. 2, pp. 185–204, Feb. 2017.
3. J. D. Herbsleb and R. E. Grinter, "Architectures, Coordination, and Distance: Conway's Law and Beyond," IEEE Software, vol. 16, no. 5, pp. 63–70, Sept.–Oct. 1999.
4. A. Vasilescu, V. Filkov, and A. Serebrenik, "StackOverflow and GitHub: Associations between Software Development and Crowdsourced Knowledge," in Proc. IEEE International Conference on Social Computing, Boston, MA, USA, 2013, pp. 188–195.
5. T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," in Proc. International Conference on Learning Representations (ICLR), Scottsdale, AZ, USA, 2013.
6. J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in Proc. 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT), Minneapolis, MN, USA, 2019, pp. 4171–4186.
7. T. Brown et al., "Language Models are Few-Shot Learners," in Advances in Neural Information Processing Systems (NeurIPS), vol. 33, 2020, pp. 1877–1901.
8. S. Bird, E. Klein, and E. Loper, Natural Language Processing with Python, 1st ed. Sebastopol, CA, USA: O'Reilly Media, 2009.
9. S. Chacon and B. Straub, Pro Git, 2nd ed. New York, NY, USA: Apress, 2014.
10. A. Begel and B. Simon, "Struggles of New College Graduates in Their First Software Development Job," in Proc. ACM SIGCSE Technical Symposium on Computer Science Education, 2008, pp. 226–230.
11. L. Singer, F. Figueira Filho, and M. A. Storey, "Software Engineering at the Speed of Social Media," in Proc. International Conference on Software Engineering (ICSE), 2014, pp. 1–10.