International Refereed Journal of Engineering and Science (IRJES)
ISSN (Online) 2319-183X, (Print) 2319-1821
Volume 14, Issue 2 (Mar.-Apr. 2025), PP. 186-195

Next-Generation Social Coding Platform: Merging Engagement Mechanisms with Technical Collaboration for Sustained Programming Enthusiasm

[1]Rahul Choudhary, [2]Pratham Patel, [3]Dhruv Pawar, [4]Pranav Sakpal,
[5]Deepa Agrawal
[1,2,3,4]UG Students, [5]Professor
[1,2,3,4,5]Department Of Artificial Intelligence and Data Science
Rajiv Gandhi Institute of Technology,
Juhu Versova Link Rd, behind HDFC Bank, Gharkul Society, Bharat Nagar, Versova, Andheri West, Mumbai,
Maharashtra 400053

Abstract: *It is widely recognized that consistency and practice are essential for mastering programming skills. Coding becomes more effective and engaging when learners can brainstorm ideas, solve problems, and work together in real time. However, existing platforms often lack features that encourage teamwork, interactive learning, and immediate feedback, leaving learners isolated and demotivated. To address this, we developed "Assemble A Social Coding Network", a web-based platform designed to make coding a team-driven, interactive, and enjoyable activity. The platform is suitable for learners of all skill levels, providing features such as real-time coding sessions, gamified challenges, and community-driven competitions. Users can work simultaneously on coding problems, share ideas, and track their progress through goal-setting tools and a progress dashboard. Additionally, the platform includes feedback systems and code review tools, enabling peer-to-peer interaction and continuous learning. Throughout development, the platform was tested with multiple users participating in coding challenges and collaborative tasks. The results showed increased engagement, improved problem-solving skills, and greater enthusiasm for coding. The Assemble platform successfully demonstrated how integrating social and collaborative elements can transform coding into an exciting, team-driven experience while fostering individual growth and creativity.*

Keywords: *Social Coding Network, Real-Time Coding, Gamified-Challenges, Community-Driven Competitions, Interactive Learning, Peer-to-Peer Interaction, Code Review Tools, Progress Tracking, Problem-Solving Skills, Coding Engagement, Skill Development*

Date of Submission: 14-04-2025 Date of acceptance: 28-04-2025

I. INTRODUCTION

In today's technology-driven world, programming has become a crucial skill, empowering individuals to innovate, create, and solve complex problems. The digital era has ushered in a growing demand for skilled programmers, making coding proficiency an invaluable asset in various industries, from software development and artificial intelligence to data science and cybersecurity. However, despite the increasing need for tech talent, many aspiring coders face significant challenges that hinder their progress and limit their ability to succeed in this dynamic field [9].

One of the key obstacles encountered by new learners is the lack of proper guidance and mentorship. Programming can be an intimidating discipline, especially for beginners who are just starting their journey. Without structured support and mentorship, many learners find themselves overwhelmed by the intricacies of syntax, logic, and problem-solving techniques[16]. The absence of a clear learning path often leads to frustration, making it difficult for beginners to navigate the complexities of programming languages and development frameworks. As a result, many individuals struggle to build a solid foundation, leading to stagnation or even abandonment of their coding aspirations. Early failures and setbacks, without adequate encouragement or direction, can be discouraging and detrimental to a learner's motivation, causing them to disengage from the learning process altogether [17].

Moreover, a lack of motivation and engagement is another critical factor contributing to the high dropout rate among coding enthusiasts. Traditional learning methods, such as self-paced online courses or static tutorials, often fail to provide an interactive and immersive learning experience. Many learners find these approaches monotonous and uninspiring, leading to a decline in enthusiasm over time [15]. The absence of a collaborative environment further exacerbates the issue, as learners feel isolated in their journey [3].

Without a strong support network, real-time feedback, or engaging challenges to test their skills, it becomes easy to lose interest in coding, leaving many aspiring developers unable to reach their full potential, [10], [18].

Recognizing these challenges, this paper introduces **"Asemble: A Social Coding Network,"** a web-based platform designed to address these issues by fostering a supportive, collaborative, and motivating space for learners. **Asemble** aims to bridge the gap between traditional learning and dynamic, community-driven engagement by offering features tailored to the needs of aspiring coders. The platform provides real-time coding environments where users can collaborate with peers, receive immediate feedback, and participate in interactive coding sessions [1], [2], [3]. By integrating gamified challenges, competitive coding events, and skill-based progression systems, Asemble keeps users engaged, motivated, and committed to continuous learning [4], [5], [6].

Furthermore, **Asemble** emphasizes the importance of peer-to-peer mentorship, allowing experienced developers to guide beginners through their coding journey. This mentorship model encourages knowledge-sharing and ensures that new learners receive the necessary support to overcome obstacles and build confidence in their abilities [11], [12]. The platform also promotes teamwork by enabling users to tackle coding challenges collectively, fostering a sense of camaraderie and shared learning. Through these collaborative features, Asemble creates a vibrant and inclusive ecosystem where coders of all skill levels can thrive, innovate, and grow together [8], [15].

In conclusion, the journey to becoming a proficient programmer is often fraught with challenges, but with the right support system, engagement strategies, and collaborative tools, learners can overcome obstacles and achieve success. **Asemble: A Social Coding Network** serves as a revolutionary platform that transforms the learning experience by providing a dynamic, interactive, and motivating environment for coders. By combining mentorship, real-time collaboration, and gamified learning, **Asemble** empowers individuals to stay committed to their coding journey, develop their skills with confidence, and ultimately, become part of a thriving community of programmers dedicated to continuous growth and innovation [16].

II. System Design

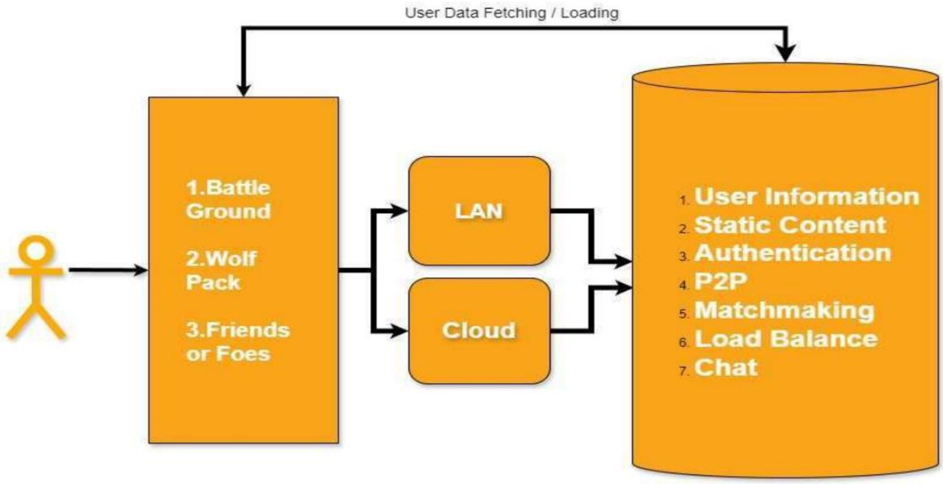


Figure 1. System Architecture of the proposed system

Asemble is designed to offer multiple interactive engagement modes that allow users to connect, compete, collaborate, and stay updated with industry trends. The platform provides various features to make coding more engaging, including social networking, solo competitions, collaborative problem-solving, structured learning paths, and expert insights.

Friends vs Foes – Social Coding Network:

- Introduces a social networking feature to help users connect, engage, and share their programming journey.
- Users can build a network of friends, post updates, and interact in a community-driven environment.
- A profile system allows users to showcase their coding progress, leaderboard rankings, and achievements.
- Users can post milestones, achievements, or queries, and others can like, comment, and flag posts.
- A suggested friends feature recommends programmers based on shared participation in challenges.
- This transforms coding from an isolated activity into a shared, interactive experience.

Battleground – Solo Competitive Coding:

- Designed for users who prefer individual coding challenges.
- Offers timed coding challenges focused on algorithmic problems, data structures, and optimization.
- A global leaderboard ranks users based on speed, accuracy, and efficiency.
- Problems are categorized based on skill level, ensuring fair competition for both beginners and experts.
- Ideal for users preparing for coding interviews and competitive programming.

Wolf Pack – Real-Time Collaborative Problem-Solving Interface:

- Unlike traditional coding problem discussions on forums or messaging apps, this feature allows real-time coding collaboration.
- Users can team up with friends or random coders to solve coding problems together.
- The interface supports live coding, simultaneous editing, code highlighting, and solution discussion.
- Encourages direct peer-to-peer problem-solving, reducing the need to search for answers on external platforms like Stack Overflow.
- Makes learning more effective by fostering interactive collaboration and debugging.

Daily Problems – A New Challenge Every 24 Hours:

- Introduces a new coding challenge daily, covering topics like algorithms, optimization, and system design.
- Users can discuss solutions, analyze different approaches, and receive peer feedback.
- Features like streaks and rewards encourage users to remain consistent with their learning.
- Helps users develop problem-solving habits and enhance coding efficiency through daily practice.

Campaign – Structured Learning Path from Beginner to Expert:

- Designed for users who prefer a step-by-step approach to mastering a programming language.
- Users can choose languages like Python, Java, C++, JavaScript, etc., and progress through multiple skill levels.
- Lessons include interactive exercises and challenges.
- Users unlock achievements, making the learning process more rewarding.
- At the expert level, users can showcase their expertise in the chosen language.
- Ideal for self-learners, students, and professionals looking to master a specific programming concept.

Blogs – Industry Insights and Learning Resources:

- A dedicated blog section provides insights into software development, competitive programming, and industry trends.
- Articles cover AI, cloud computing, cybersecurity, coding best practices, and more.
- Tech experts and mentors offer career advice, interview preparation tips, and resume-building strategies.
- Ensures that users stay updated with industry advancements while developing their technical skills.

III. Hardware and Software Setup

- **Software Setup**

  The development of **Asemble** involves multiple technologies and frameworks that collectively enhance the platform's performance, usability, and scalability.

- **Frontend Development**

  - **Asemble's** frontend is designed to offer a smooth, responsive, and interactive experience, allowing users to navigate coding challenges, connect with others, and collaborate effortlessly. The interface follows a structured layout to ensure ease of use, with well-defined sections for leaderboards, user profiles, and real-time coding activities.

  - The foundation of the frontend is built using HTML, CSS, and JavaScript. HTML (HyperText Markup Language) provides the structure of the web pages, defining elements like buttons, forms, navigation menus, and content sections. CSS (Cascading Style Sheets) enhances the visual appeal by adding styles, animations, and responsive designs, ensuring that the platform functions well on different screen sizes.

  - JavaScript plays a crucial role in enabling dynamic user interactions by handling real-time content updates, interactive elements, and event-driven functionalities. Features such as friend requests, leaderboard updates, and coding submissions are seamlessly integrated using

  - JavaScript. Additionally, AJAX (Asynchronous JavaScript and XML) and the Fetch API ensure that data is retrieved and updated without requiring full-page reloads. This real-time data synchronization is essential for a smooth and uninterrupted user experience, particularly in live collaborative coding and social interactions

By combining these technologies, **Asemble** delivers a modern and efficient coding environment where users can focus on problem-solving and learning without technical disruptions.

- **Backend Development**

    - The backend of **Asemble** is responsible for efficiently managing authentication, coding challenge execution, social interactions, and real-time collaboration, ensuring a seamless and responsive user experience. It serves as the backbone of the platform, handling all data processing, user requests, and live interactions.

    - The backend is powered by Node.js and Express.js, both of which are well-suited for building scalable, high-performance web applications. Node.js operates on a non-blocking, event-driven architecture, making it ideal for handling multiple simultaneous requests without compromising speed or efficiency. This ensures that coding challenges, leaderboard updates, and social interactions are processed quickly, even when a high number of users are active on the platform.

    - For real-time collaboration, **Asemble** integrates WebSocket (via Socket.IO), allowing users to engage in live coding sessions, interact with friends, and receive instant updates. This is particularly essential for features like Wolf Pack and Friends vs Foes, where multiple users need to see real-time changes without delays. By leveraging WebSocket connections, Asemble ensures that live coding interactions, chat messages, and coding updates occur instantaneously, enhancing the collaborative experience

Overall, the backend is designed to be robust, scalable, and efficient, ensuring that users can code, compete, and collaborate without interruptions.

- **Database Management**

    - **Asemble** relies on a flexible and scalable database to efficiently manage user data, coding activities, and real-time interactions. Given the dynamic nature of coding platforms, where data is frequently updated and retrieved, a robust database solution is essential for optimal performance. Database: MongoDB

    - **Asemble** utilizes MongoDB, a NoSQL database known for its ability to handle unstructured and semi-structured data. This makes it an ideal choice for storing user profiles, friend connections, chat messages, coding challenge submissions, and leaderboard rankings. Unlike traditional relational databases, MongoDB's document-based storage model provides greater flexibility, allowing seamless scalability as user activity grows.

    - To enhance efficiency and streamline data operations, Mongoose ORM is integrated. Mongoose provides a structured way to interact with MongoDB, ensuring smooth handling of database queries, data validation, and schema definitions. This improves data consistency while simplifying complex interactions, such as retrieving user progress, managing coding challenge results, and updating social interactions.

    - A critical aspect of **Asemble's** database architecture is real-time data synchronization, which ensures that leaderboards, friend requests, achievements, and coding updates are instantly reflected across the platform.

- **Docker – Containerization for Deployment Consistency**

- **Ensuring Environment Consistency with Docker**
  Asemble integrates Docker to streamline development, deployment, and scalability, ensuring that the platform remains stable and consistent across different environments. By leveraging containerization, Docker eliminates compatibility issues and simplifies team collaboration while optimizing resource management.

- **Standardized Development Environment for Consistency**
  One of Docker's key advantages is its ability to create a uniform development environment across all machines. By using Docker containers, **Asemble** ensures that every developer works with identical dependencies, configurations, and libraries. This prevents compatibility issues that arise when running the application on different operating systems or setups, leading to a smoother development workflow.

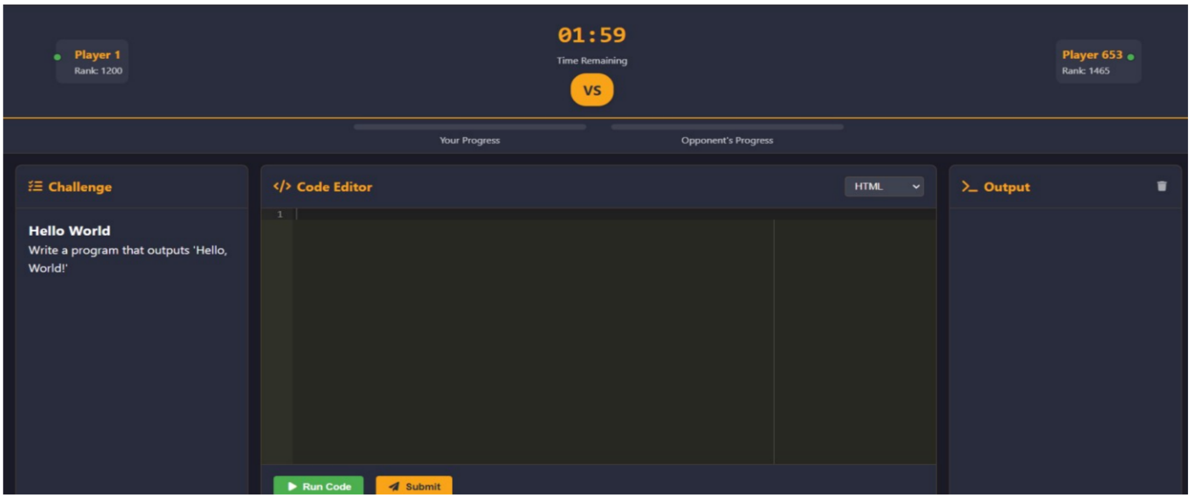
- **Containerization for Isolation and Scalability**
  Docker encapsulates each service (backend, database, WebSocket, etc.) into separate containers, ensuring they run independently without interfering with each other. This modular approach prevents dependency conflicts and allows each service to be scaled independently based on demand. For instance, if real-time collaboration usage increases, only the WebSocket container can be scaled without affecting other components.

By utilizing Docker's standardized environments, containerization, and easy deployment capabilities, **Asemble** ensures seamless collaboration, improved scalability, and reliable performance across different deployment setups.

- **Hardware Setup**

  - Processor (Intel Core i5 (8th generation or later) or equivalent AMD processor)
  - RAM (Minimum 4GB (8GB recommended for better performance))
  - ROM (Minimum 512GB SSD (Solid State Drive) for faster data access)
  - GPU (Integrated graphics (Intel HD Graphics or equivalent) is sufficient for basic use)

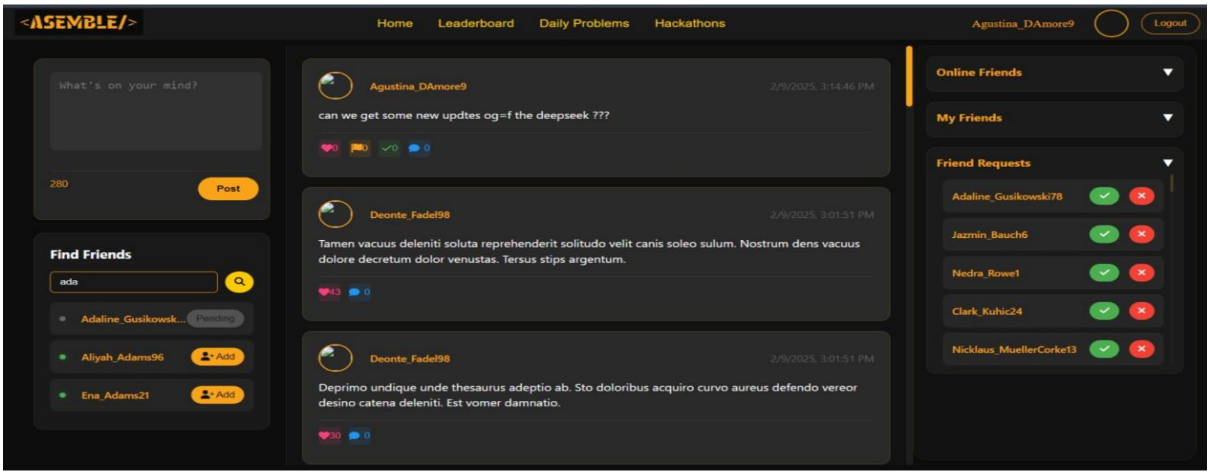
IV. Result



Hello World
Write a program that outputs 'Hello, World!'

Figure 2. Battleground

The Battleground mode presents a real-time competitive coding arena where two players face off to solve a given challenge within a time limit. The interface consists of a challenge panel, a code editor, and an output console, allowing users to write, test, and submit their solutions. A progress bar tracks each player's advancement, adding a dynamic and engaging element to the competition. With live ranking updates and a time-bound structure, this mode fosters a fast-paced coding battle, enhancing problem-solving skills under pressure.



Agustina_DAmore9 2/9/2025, 3:14:46 PM

can we get some new updtes og=f the deepseek ???

♥0 0 ✓0 0

Deonte_Fadel98 2/9/2025, 3:01:51 PM

Tamen vacuus deleniti soluta reprehenderit solitudo velit canis soleo sulum. Nostrum dens vacuus dolore decretum dolor venustas. Tersus stips argentum.

♥43 0

Deonte_Fadel98 2/9/2025, 3:01:51 PM

Deprimo undique unde thesaurus adeptio ab. Sto doloribus acquiro curvo aureus defendo vereor desino catena deleniti. Est vomer damnatio.

♥30 0

Figure 3.Friend vs Foes

The Friend vs Foe feature in **Asemble** serves as a dynamic social hub where users can connect, interact, and monitor their peers' activities. The interface displays a real-time social feed, allowing users to post updates, react to others' content, and engage in discussions. The friend management panel enables users to send, accept, or decline friend requests, while the online status indicators help track active connections. Additionally, the friend search and request system facilitates expanding one’s network. This feature fosters a collaborative and competitive environment, encouraging users to engage with their community through shared insights and coding challenges.

V. Performance metrics

Performance evaluation plays a crucial role in determining the efficiency, responsiveness, and scalability of **Asemble.** The platform was assessed based on response time, scalability, and user retention rate, ensuring a smooth and seamless coding experience for users. The results demonstrate that **Asemble** is capable of handling real-time collaboration, multiple concurrent users, and sustained user engagement effectively.

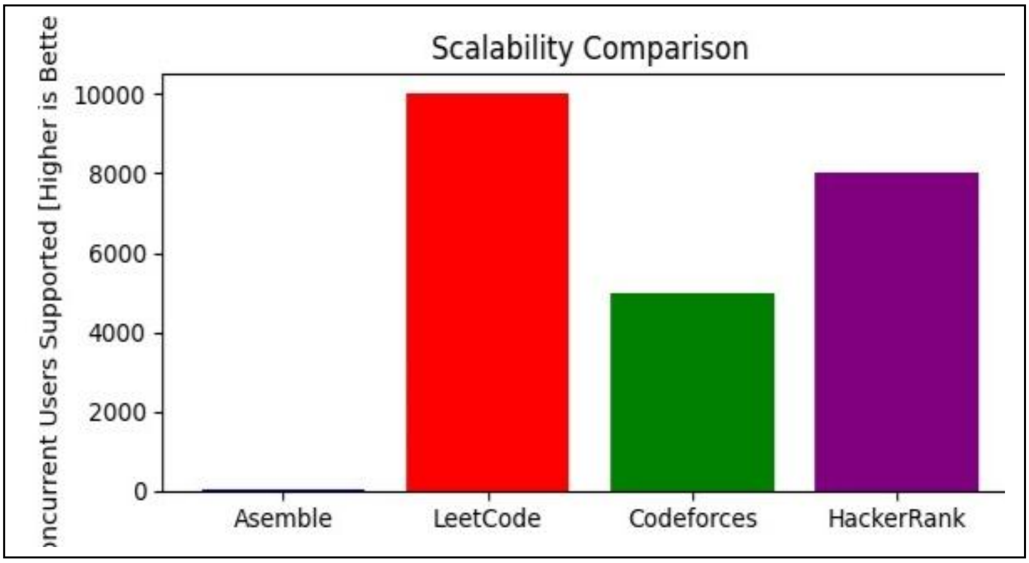


Fig 4. Scalability Comparison

Scalability: To assess how well **Asemble** handles multiple users interacting simultaneously, scalability testing was conducted.

- The system was tested with 15+ concurrent users participating in activities such as real-time coding, posting updates, chatting, and solving challenges.
- The server handled multiple API requests efficiently, preventing significant slowdowns or crashes.
- Database queries were optimized to avoid performance bottlenecks, ensuring smooth processing of friend requests, challenge updates, and leaderboard rankings.
- The platform demonstrated scalability for a growing user base, with future optimizations planned to support higher user loads as adoption increases.

Scalability Comparison with Other Platforms : This graph illustrates the scalability of platforms by measuring the number of concurrent users they can support:

- LeetCode leads in scalability, supporting the highest number of simultaneous users.
- HackerRank and Codeforces follow closely, managing large user volumes efficiently.
- **Asemble**, being a newer platform, has lower scalability in comparison but is improving.
- Further optimizations are necessary to handle large-scale traffic, especially during coding competitions and peak usage hours.

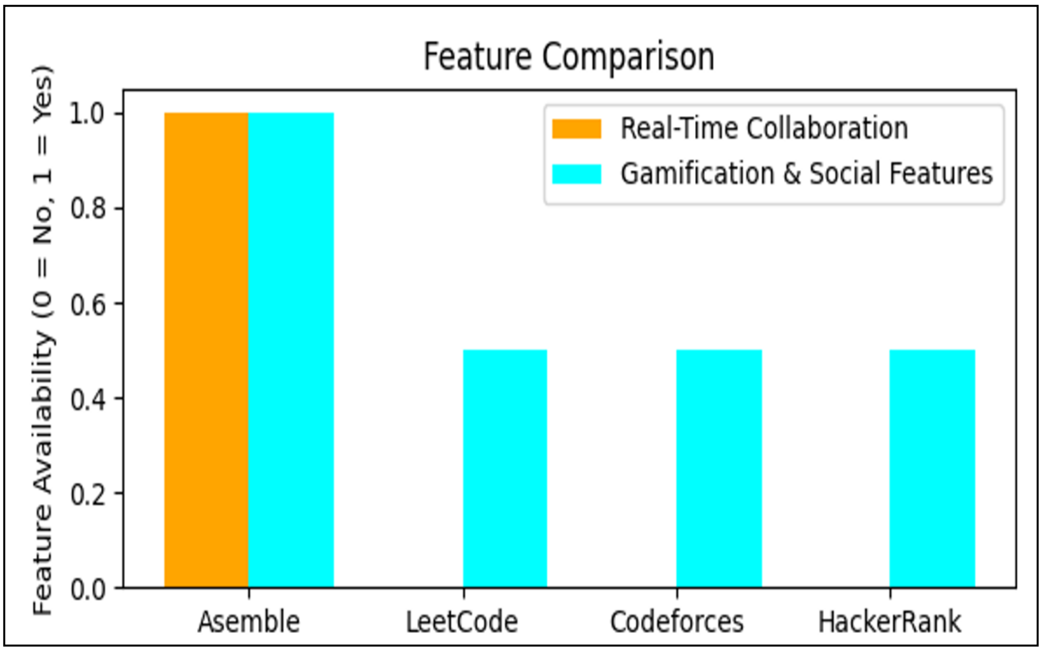


Fig 5. Feature Comparison

Feature Comparison : One of the key aspects of **Asemble's** performance evaluation was its feature collection, setting it apart from traditional coding platforms.

- **Asemble** was designed to provide a dynamic and interactive environment with features such as real-time collaboration, gamification, and social engagement tools.
- The platform supports real-time coding sessions, allowing multiple users to work on the same problem simultaneously with instant updates ensuring smooth synchronization.
- Gamification elements, including achievement badges, leaderboards, and competitive coding modes like "Wolf Pack" and "Friends vs Foes," enhance user engagement and motivation.
- These features transform **Asemble** from a coding platform into a community-driven space where users can learn, compete, and collaborate effortlessly.

Feature Availability Comparison :A graph highlights the feature availability across different platforms, focusing on real-time collaboration and gamification/social features:

- **Asemble** stands out as the only platform that fully supports both real-time collaboration and gamification elements.
- LeetCode, Codeforces, and HackerRank provide some social and gamification features but lack real-time collaborative coding.
- **Asemble's** unique integration of real-time collaboration makes it more interactive and engaging, catering to users who enjoy working on challenges together or participating in coding activities with friends.

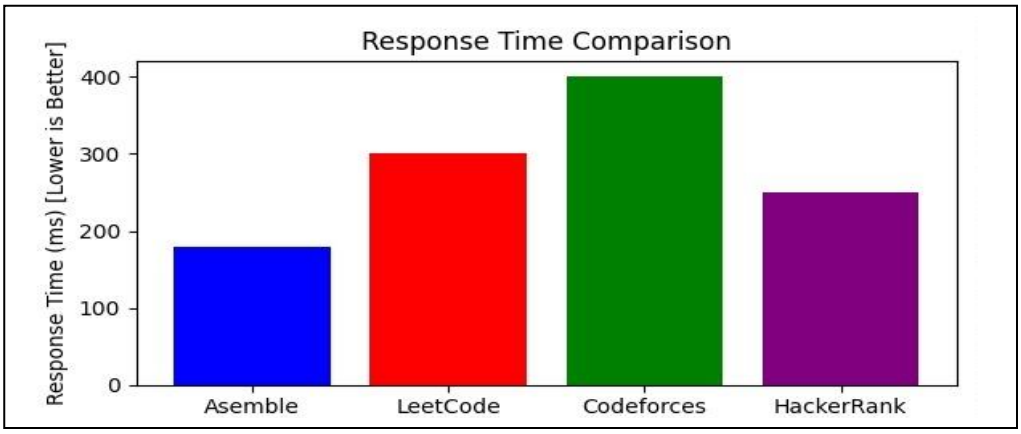


Fig 5. Response Time Comparison

Response Time : One of the primary performance metrics was the **response time**, which directly impacts the user experience in real-time collaborative coding sessions.

- **Asemble** was optimized to respond within <200ms for real-time operations such as coding interactions, leaderboard updates, and chat messages.
- WebSockets were implemented for live communication, ensuring instant updates and seamless synchronization between multiple users, especially in Wolf Pack mode and Friends vs Foes interactions.
- Fast response times helped eliminate lag in live coding, allowing users to collaborate without experiencing delays.

Response Time Comparison : This graph compares the **response times** of different coding platforms, including **Asemble**, LeetCode, Codeforces, and HackerRank.

- **Asemble** has the lowest response time, indicating faster execution compared to competitors.
- LeetCode and HackerRank have moderate response times.
- Codeforces has the highest response time, meaning users may experience longer feedback delays on code submissions.
- Lower response times contribute to a smoother user experience, making **Asemble** more efficient in handling real-time coding and execution.

VI. CONCLUDING REMARKS

We observed highly encouraging results from the implementation of **"Asemble"**. Users experienced increased motivation and engagement as the platform provided a collaborative and interactive environment that made learning programming more enjoyable and less intimidating. Beginners particularly benefited from real-time coding support and peer collaboration, which helped them overcome early challenges and build confidence. The inclusion of gamified challenges and goal-setting tools further enhanced user experience by promoting continuous learning and skill improvement. Looking ahead, **"Asemble"** has significant potential for expansion. Future enhancements could include matchmaking algorithms to pair users based on skill levels and interests, fostering balanced and productive collaborations. Features like friends vs foes feed can encourage healthy competition through coding duels, while hackathons can simulate real-world project-building experiences to inspire teamwork and innovation. Introducing prize-based bounties could further motivate users by offering rewards for solving complex coding challenges, making the platform even more engaging and impactful.

REFERENCES

[1]. H. Pathak, R. Pawar, H. Ritik, and Y. Pingle, "CodeFlow: Real-Time Collaborative Code Editor," *International Conference on Knowledge Engineering and Communication Systems (ICKECS)*, vol. 60, no. 5, pp. 5968–5975, 2024

[2]. K. P. M., M. P. S., K. Selvi, R. S. N., M. S., and M. G., "EscapeScript - Gamified Coding Platform for Empowering Future Coders," *International Conference on Sustainable Computing and Smart Systems (ICSCSS)*, vol. 60, no. 5, pp. 552–564, 2024

[3]. R. B. Hadiprakoso and R. P. P. Sihombing, "CodeGuardians: A Gamified Learning for Enhancing Secure Coding Practices with AI-Driven Feedback," *Ultima Infosys: JurnalIlmuSistemInformasi*, vol. 15, no. 2, pp. 105–120, Dec. 2024

[4]. J. Jiang, Y. Xie, B. Fang, M. Wang, and H. Fan, "Building Temporary Isolated Workspace in Real-Time Collaborative Programming Environment," *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, vol. 60, no. 5, pp. 4110–4120, Oct. 2023

[5]. K. Virdi, A. L. Yadav, A. A. Gadoo, and N. S. Talwandi, "Collaborative Code Editors - Enabling Real-Time Multi-User Coding and Knowledge Sharing," *International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)*, vol. 60, no. 5, pp. 614–626, 2023

[6]. W. N. Hidayat, E. P. M. Syahri, W. D. Prasetyo, M. M. Afdloly, A. N. Zahiroh, and R. K. Sari, "Development of Gamification and Live Coding-Based Programming Learning Platform to Foster Learning Motivation of Vocational Students," *International Conference on Electrical and Information Technology (IEIT)*, vol. 60, no. 5, pp. 299–310, 2023

[7]. B. Say, H. Altunel, M. Kosa, and M. Koca-Atabey, "Evaluation of an Industrial Case of Gamification in Software Quality Improvement," *International Journal of Serious Games*, vol. 10, no. 3, pp. 23–26, Sep. 2023

[8]. J. C. Paiva, R. Queirós, J. P. Leal, J. Swacha, and F. Miernik, "Managing Gamified Programming Courses with the FGPE Platform," *Information*, vol. 13, no. 45, pp. 1–16, Jan. 2022

[9]. H. M. Jawad and S. Tout, "Gamifying Computer Science Education for Z Generation," Information, vol. 12, no. 453, pp. 1–18, Nov. 2021

[10]. F. Y. W. Leung, M. Lau, K. Wan, L. Law, T. Kwong, and E. Y. W. Wong, "Promoting Students' Global Perspectives Through a Gamified e-Learning Platform," *Front. Educ.*, vol. 6, no. 617680, pp. 1–13, Sep. 2021

[11]. K. Tangtisanon and K. Tongprasert, "Computer Programming Classroom Platform," *7th International Conference on Engineering, Applied Sciences and Technology (ICEAST)*, vol. 2021, no. 52143, pp. 1–6, 2021

[12]. I. S. Zinovieva, V. O. Artemchuk, A. V. Iatsyshyn, O. O. Popov, V. O. Kovach, A. V. Iatsyshyn, Y. O. Romanenko, and O. V. Radchenko, "The use of online coding platforms as additional distance tools in programming education," *J. Phys. Conf. Ser.*, vol. 1840, no. 1, pp. 012029, Mar. 2021

[13]. Z. Y. Liu, Z. A. Shaikh, and F. Gazizova, "Using the Concept of Game-Based Learning in Education," *International Journal of Emerging Technologies in Learning (iJET)*, vol. 15, no. 14, pp. 53–70, 2020

[14]. J. C. Liang, J. Y. Wang, and S. H. Chou, "A Designed Platform for Programming Courses," *IIAI International Conference on Advanced Applied Informatics (IIAI-AAI)*, vol. 2020, no. 50415, pp. 1–8, 2020

[15]. Q. Mi, J. Keung, X. Mei, Y. Xiao, and W. K. Chan, "A Gamification Technique for Motivating Students to Learn Code Readability in Software Engineering," *International Symposium on Educational Technology (ISET)*, vol. 2018, no. 00062, pp. 1–8, 2018

[16]. D. Dicheva, K. Irwin, C. Dichev, and S. Talasila, "A Course Gamification Platform Supporting Student Motivation and Engagement," *ICWOAL 2014*, vol. 2014, no. 7009214, pp. 1–10, 2015

[17]. M. B. Ibáñez, A. Di-Serio, and C. Delgado-Kloos, "Gamification for Engaging Computer Science Students in Learning Activities: A Case Study," *IEEE Transactions on Learning Technologies*, vol. 2014, no. 2329293, pp. 1–10, Jul. 2014

[18]. Y. S. Wong and M. H. M. Yatim, "Computer Game As Learning and Teaching Tool For Object Oriented Programming in Higher Education Institution," *Procedia - Social and Behavioral Sciences*, vol. 123, pp. 215–224, Mar. 2014

[19]. M. Goldman, G. Little, and R. C. Miller, "Real-Time Collaborative Coding in a Web IDE," *UIST'11*, vol. 60, no. 5, pp. 4231–4241, Oct. 2011

[20]. K. Maillet and M. Porta, "Consequences of the Declining Interest in Computer Science Studies in Europe," *IEEE Transactions on Learning Technologies*, vol. 2010, no. 1, pp. 1–10, 2010