CodeArena: An Interactive Online Coding Practice Platform with Real-Time Execution, Debugging, and Competitive Modules

Km Neeraj Yadav
Department of MCA
GLBITM
Greater Noida, India
neerajydv1113@gmail.com

Dr. Sanjeev Kumar
Department of MCA
GLBITM
Greater Noida
sanjeev.kumar@glbitm.ac.in

Abstract: With the rapid advancements in the current age, coding websites have become crucial tools in fostering algorithmic thinking, problem-solving skills, and overall programming abilities. Such coding websites provide learners with an opportunity to train themselves on coding problems, receive feedback on their solutions and keep improving their abilities. In this study, we introduce CodeArena—an online coding platform that allows learners to gain knowledge about programming and develop their coding abilities. This project involves creating CodeArena—a relatively simple and efficient full-stack web application. CodeArena enables secure user authentication, sorting coding problems based on different levels of complexity, and executing code submitted by users together with giving instant feedback on such codes. The application is also equipped with debugging capabilities that enable a better understanding of program execution and identification of common logical mistakes. Another unique feature of CodeArena is that it provides users with coding competitions that foster competencies in competitive programming and help users increase their coding speed. The system utilizes modern full-stack technologies in its implementation, including React.js, Node.js, and Express.js in the front end and back end parts of the website and MongoDB as a database management system. CodeArena is also connected to an external API called Judge0 to facilitate real-time code execution. Furthermore, CodeArena provides performance tracking capabilities for users' convenience.

Keywords-: Online Coding Platform; CodeArena; Problem-Solving; Debugging; Competitive Programming; Coding Contests; Full-Stack Development; Real-Time Code Execution; Performance Tracking.

1. Introduction

Programming skills have become one of the core competencies in the modern digital era, as they play a significant role in fields such as software engineering, artificial intelligence, data science, and information technology. With the rapid growth of technology, there is an increasing demand for individuals who possess strong programming knowledge and the ability to design efficient algorithms. To meet this demand, structured coding practice has become essential for learners at all levels.

There are numerous platforms available that focus on developing programming skills through coding practice. These platforms allow users to solve a wide range of problems, receive real-time feedback, and monitor their progress over time. They contribute significantly to improving logical thinking, coding efficiency, and problem-solving abilities. Moreover, such platforms play a vital role in preparing individuals for technical interviews and competitive programming .

Despite their usefulness, many existing coding platforms are complex, resource-intensive, and not always suitable for beginners. They often lack clear learning paths and sufficient debugging tools, making it difficult for users to identify errors and improve their coding skills effectively. Additionally, users may need to rely on multiple platforms to access different features, which can reduce learning efficiency and user convenience.

To address these challenges, CodeArena is proposed as an online coding platform designed to provide a simple, scalable, and user-friendly environment for coding practice . The platform enables secure user registration, allows users to solve problems categorized by different levels of difficulty, and supports code submission with real-time execution feedback . A key feature of CodeArena is its step-by-step debugging functionality, which helps users understand program execution and identify logical errors more effectively . Furthermore, the inclusion of coding contests enhances competitive programming skills and improves coding speed and accuracy

Table No 1: Comparison of CodeArena With Existing Coding Platform

Platform	Difficulty Levels	Contests	Debugging
CodeArena	Yes	Yes	Yes
Leetcode	Yes	Yes	Yes
HackerRank	Yes	Limited	No
CodeForces	Yes	Yes	Limited

Table 1 illustrates the comparative analysis of CodeArena with respect to other coding platforms. Features such as levels of difficulty, contests, and debugging have been considered in the comparison. It can be seen from Table 1 that CodeArena offers all these features.

The system is developed using modern full-stack technologies, including React.js for building a responsive user interface, Node.js for backend processing, and MongoDB for efficient data storage and management . In addition, the platform integrates external APIs such as Judge0 for real-time code execution and evaluation . This modular and flexible architecture ensures scalability, efficient performance, and ease of future enhancements .

2. Related Work

In recent years, online coding platforms have emerged as powerful tools for enhancing programming skills, algorithmic thinking, and problem-solving abilities [1]. These platforms provide structured environments where users can practice coding problems, receive instant feedback, and monitor their performance over time [2]. They are widely used by students, professionals, and competitive programmers for skill development, interview preparation, and participation in coding contests [3].

Several well-known platforms such as LeetCode, HackerRank, and Codeforces have significantly contributed to the growth of online coding practice [4]. LeetCode is primarily focused on technical interview preparation and provides a vast collection of problems categorized into easy, medium, and hard levels [5]. It also includes discussion forums, solution explanations, and company-specific question sets that help users prepare for real-world interview scenarios [6]. HackerRank offers a diverse range of domains including algorithms, data structures, databases, artificial intelligence, and mathematics [7]. It also provides certification programs and structured learning tracks that guide users in developing specific skills [8]. Codeforces, on the other hand, is widely recognized for its competitive programming contests, ranking system, and global participation, which helps users improve their coding speed, efficiency, and competitive skills [9].

These platforms incorporate various features such as real-time code submission, automated evaluation using multiple test cases, performance tracking, and leaderboards [10]. They also support multiple programming languages and provide detailed feedback on code execution, including time complexity and memory usage [11]. Such features contribute significantly to improving coding proficiency and enable users to solve complex problems effectively[12].

From a research perspective, studies have emphasized the importance of interactive and problem-based learning approaches in programming education [13]. Systems that provide hands-on practice and immediate feedback have been shown to enhance logical reasoning and computational thinking [14]. Debugging tools are also considered a crucial component, as they allow users to understand the step-by-step execution of their code and identify errors more effectively [15]. However, many existing platforms provide limited debugging support, which can be a major drawback for beginners who need detailed guidance [16].
Another important aspect observed in existing systems is the lack of personalization [17]. Most platforms provide the same set of problems to all users without considering their individual skill levels, learning pace, or weak areas [18]. This often leads to inefficient learning and reduced user engagement. Additionally, some platforms do not provide detailed analytics or insights into user performance, which are essential for tracking improvement over time .

Despite their numerous advantages, existing coding platforms have certain limitations [19]. Many of these systems are complex and not beginner-friendly, making it difficult for new users to navigate and utilize all available features [20]. They often lack a well-defined learning path, which can lead to confusion and inefficiency in learning [21]. Additionally, some platforms focus heavily on competitive programming and ranking systems, which may not be suitable for users who are primarily interested in learning and understanding concepts [22].

Another limitation is that these platforms can be resource-intensive and may require high system performance or stable internet connectivity, making them less accessible in certain environments [23]. Furthermore, users often need to switch between multiple platforms to access different functionalities such as learning materials, coding practice, debugging tools, and competitions [24].This fragmentation reduces overall efficiency and creates inconvenience for users [25].

To overcome these challenges, CodeArena is proposed as a simplified, scalable, and user-friendly online coding platform. It aims to integrate essential features such as categorized problem sets, real-time code execution, performance tracking, and coding contests into a single system. One of the key highlights of CodeArena is its step-by-step debugging feature, which provides detailed insights into code execution and helps users understand errors more effectively.

In addition, CodeArena is developed using modern full-stack technologies including React.js for frontend development, Node.js for backend processing, and MongoDB for efficient data storage. The platform also integrates external APIs such as Judge0 to support real-time code execution and evaluation . This modular architecture ensures scalability, flexibility, and ease of maintenance .

Furthermore, CodeArena focuses on providing a balanced combination of learning, practice, and competition in a single platform. It is designed to be accessible for both beginners and advanced users, offering a smooth and intuitive user experience .

3. Methodology

The proposed methodology for CodeArena focuses on the design and development of an efficient and scalable online coding platform that supports learning, practice, and competition. A full-stack development approach is adopted to ensure smooth integration of frontend, backend, database, and external APIs, resulting in a seamless and interactive user experience.

The system architecture consists of a frontend developed using React.js, which provides a dynamic and responsive user interface. The backend is implemented using Node.js and Express.js to handle server-side logic, user requests, and application functionality. MongoDB is used as the database to store user information, coding problems, submissions, and contest data. Additionally, external APIs such as Judge0 are integrated to enable real-time code execution and evaluation.

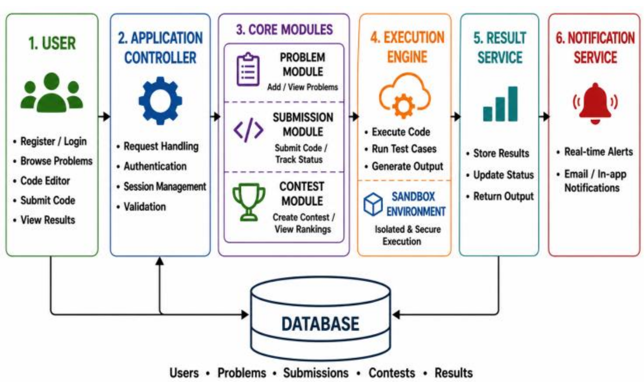

Fig. 1. System Architecture of CodeArena

The overall system operation starts with authentication, wherein the users need to provide their credentials in order to authenticate themselves on the website. Once a user logs in successfully, he or she is directed to the dashboard. In the dashboard, coding questions are sorted out depending upon their respective difficulty levels, ranging from easy to medium to hard. There is a code editor provided in the system through which users can create, edit, and run their code within the system. When a user finishes with his or her code, he or she will submit the code for checking purposes. The submitted code goes to the backend server for execution, where it is tested using pre-defined test cases. The system analyzes the output and compares the actual output with the expected output. Once the process is completed, the results will be instantly shown to the user with the output execution details.

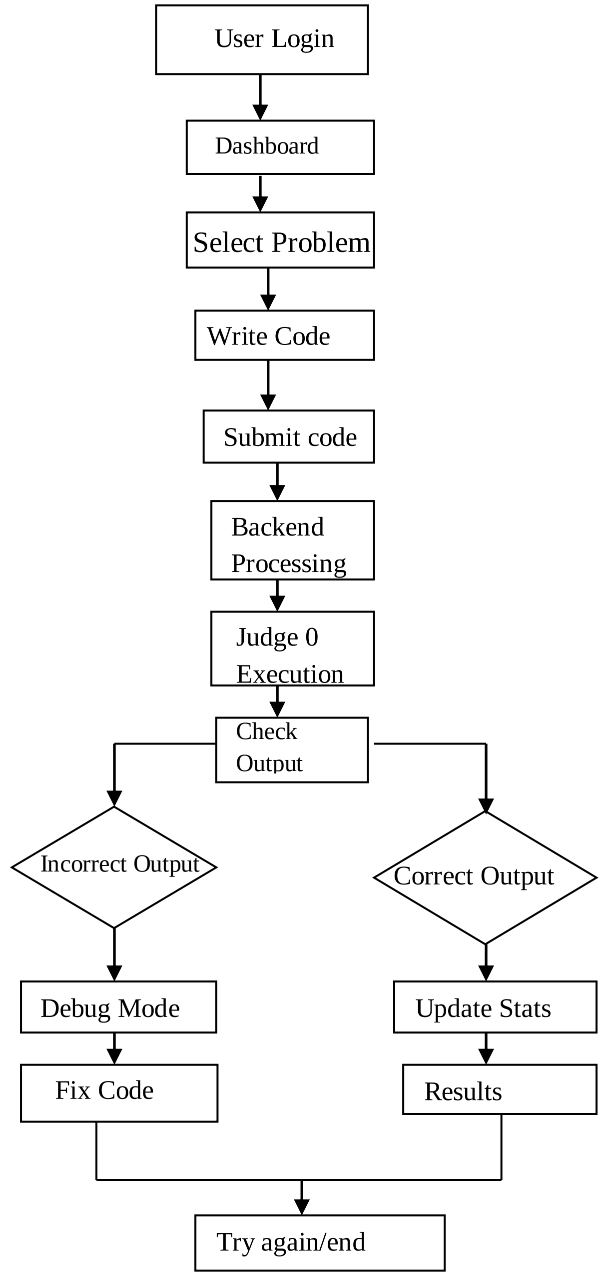

Fig. 2. Code Execution Workflow of CodeArena

The Figure 2 below highlights the operational flow of executing code in the CodeArena environment. The figure shows how the user works with the system when solving a coding challenge. First, a user will be required to log into the application from the dashboard page. On this page, all challenges available in the platform are listed according to their difficulty levels. After picking a problem, the user will solve the problem by writing the program in the code editor.

When done, the user submits the code to the backend system, which will process it further to check its validity. The solution is verified by executing the code in the Judge0 environment with the help of predefined test cases. Finally, the system compares the generated output with the expected output. In case the solution is correct, the system displays the output. Otherwise, debugging tools are provided to help the user locate mistakes.

4. Results

The use of CodeArena represents an example of the successful creation of a scalable and interactive platform for conducting coding contests. It was tested for several operations, such as user authentication, problem administration, code evaluation, debugging, and competition participation. The findings show that the system operates effectively in various conditions and provides valid output.

The analysis of the statistics available on the platform reveals that participants are involved in solving coding tasks actively enough to make several attempts at their solution. Such an approach is associated with efficient learning because the participants can refine their answers and develop relevant skills by repeatedly solving the same problems. The percentage of correct responses proves that users can attain desirable outcomes during the third attempt.

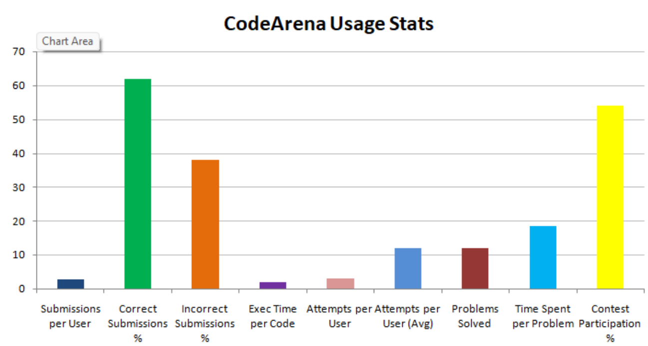

Fig No. 3: Statistical Data of CodeArena Usage

The analysis from graph depicts the involvement of users in the application to improve their ability to solve problems by making multiple attempts. Execution time represents how quickly the system works with respect to processing code written by users and giving results. Furthermore, the involvement of the users in coding competitions proves that the system motivates users to participate in competitions.

Output Summary:

The output results of the implemented CodeArena system demonstrate that all major functionalities operate effectively and provide accurate outcomes. The user dashboard successfully displays progress tracking and problem-solving statistics, enabling users to monitor their performance. The problem list interface presents coding challenges categorized by difficulty levels, allowing users to select problems based on their skill level. Furthermore, the code editor interface enables users to write, execute, and debug their code in real time, producing correct outputs based on the provided test cases. The contest module also functions efficiently by allowing users to participate in coding competitions and evaluate their performance. Overall, the system ensures smooth interaction, accurate code execution, and reliable output generation, thereby enhancing the user learning experience.

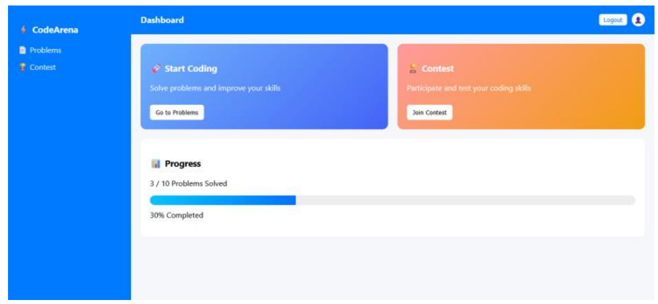

Fig No. 4: Code Editor Interface of CodeArena

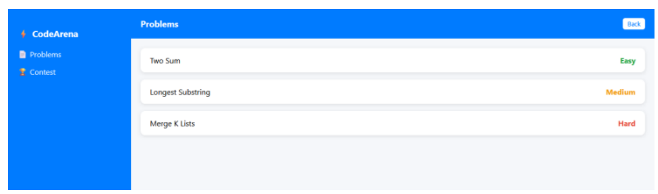

Fig No. 5: Code Editor Interface of CodeArena

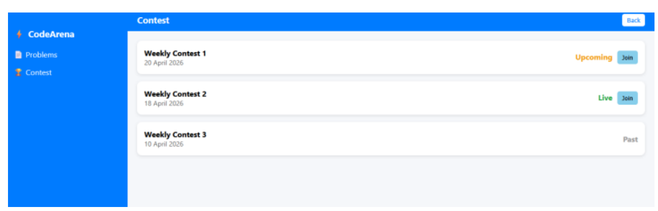

Fig No. 6: Contest Interface of CodeArena

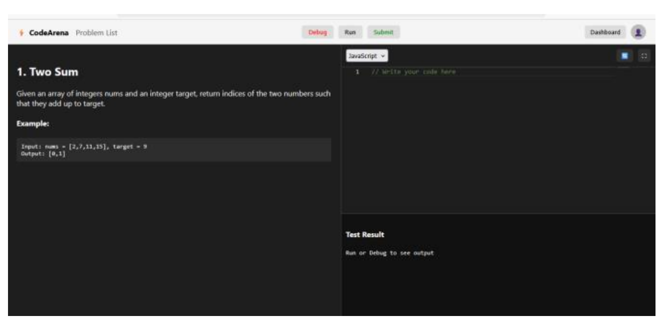

Fig No. 7: Code Editor Interface of CodeArena

5. Discussion

The study focuses on evaluating the performance and effectiveness of the CodeArena platform in enhancing programming and problem-solving skills. The analysis is based on user interactions with the system, including code submissions, execution results, debugging usage, and participation in coding contests. These indicators help in understanding user engagement, learning behavior, and overall system effectiveness.

The collected data shows that users actively interact with the platform by attempting multiple submissions for each problem. This behavior reflects an iterative learning approach, where users improve their solutions based on feedback received from the system. The availability of real-time results plays a crucial role in helping users identify mistakes and refine their coding logic.

5.1 Performance Based on User Activity

Users were analyzed based on their problem-solving performance and engagement level on the platform. Based on submission accuracy and participation, users can be broadly categorized into different groups such as high-performing users, moderate learners, and beginners. High-performing users tend to solve problems efficiently with fewer attempts, while beginners require multiple attempts but show gradual improvement over time.

The debugging feature significantly contributes to user performance by enabling step-by-step analysis of code execution. Users who actively use debugging tools demonstrate better understanding of program flow and improved accuracy in their submissions. This highlights the importance of integrating debugging functionality in coding platforms.

5.2 Impact of System Features on Learning

The results indicate that the integration of multiple features such as real-time code execution, structured problem sets, and performance tracking enhances the overall learning experience. The categorization of problems into different difficulty levels allows users to follow a systematic learning path. This structured approach helps users build confidence and improve their coding proficiency gradually.

The contest module also plays a vital role in increasing user engagement. Time-based coding challenges encourage users to improve their speed and accuracy, thereby enhancing their competitive programming skills. Users participating in contests show higher engagement levels compared to those who only practice problems.

5.3 System Efficiency and Usability

From a system perspective, CodeArena demonstrates efficient performance in handling user requests and executing code. The use of modern technologies such as React.js, Node.js, and MongoDB ensures fast response time and smooth system operation. The integration of the Judge0 API enables reliable and accurate code execution.

Compared to existing platforms, CodeArena provides a simpler and more user-friendly interface, making it suitable for beginners while still supporting advanced users. The integration of learning, debugging, and competition within a single platform reduces complexity and improves user convenience.

5.4 User Engagement and Learning Outcomes

Based on the analysis of user actions, it can be concluded that the proposed platform encourages constant involvement and learning of users. Such options as immediate response, several submissions, and monitoring the success rate motivate users to become involved in solving different problems on the website. The time users spend on the platform tends to increase due to the desire to solve problems more efficiently.

The presence of classified problems and learning paths allows for gradual development of user skills. Beginners have an opportunity to start with less complicated problems and then move to more difficult challenges. Thus, users get access to the personalized adaptation of the platform's capabilities.

In addition, users are motivated through competition and ranking. This factor contributes to increased user participation in the process of studying. Therefore, the suggested solution is able to maintain balance between learning and engagement.

5.5 Conclusion

In general, the evaluation demonstrates that CodeArena is a valuable web application for learning programming because it provides a user-friendly coding environment. In addition, the application involves learners in problem solving, debugging, and taking part in code competitions, thus contributing to the development of logical thinking and analytical skills. It should also be noted that the application gives users an opportunity to assess their performance and understand their weak spots. Finally, it should be noted that feedback is one of the major advantages of this platform. Furthermore, the platform ensures accurate code execution and reliable output generation for various coding problems. The integration of real-time execution and debugging tools enhances user understanding of program behavior. The system also promotes continuous learning through repeated practice and performance tracking.

References

[1] Zinovieva, I. S., et al. "The use of online coding platforms as additional distance tools in programming education." *Journal of physics: Conference series*. Vol. 1840. No. 1. IOP Publishing, 2021.

[2] Krancher, Oliver, Pascal Luther, and Marc Jost. "Key affordances of platform-as-a-service: Self-organization and continuous feedback." *Journal of Management Information Systems* 35.3 (2018): 776-812.

[3] Yuen, Kevin KF, Dennis YW Liu, and Hong Va Leong. "Competitive programming in computational thinking and problem solving education." *Computer Applications in Engineering Education* 31.4 (2023): 850-866.

[4] Billah, Md Mustakim, et al. "Are large language models a threat to programming platforms? an exploratory study." *Proceedings of the 18th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*. 2024.

[5] Cui, Jialin, et al. "How much effort do you need to expend on a technical interview? a study of leetcode problem solving statistics." *2024 36th International Conference on Software Engineering Education and Training (CSEE&T)*. IEEE, 2024.

[6] MR, Vishnu Priya, et al. "VoxMentor: A Voice-Driven AI Platform for Interview Simulation, Company-Specific Preparation, and Student Learning." *2025 International Conference on Intelligent Computing, Information and Control Systems (ICOIICS)*. IEEE, 2025.

[7] Huang, Cheng, et al. "HackerRank: Identifying key hackers in underground forums." *International Journal of Distributed Sensor Networks* 17.5 (2021): 15501477211015145.

[8] Randall, Michael H., and Christopher J. Zirkle. "Information technology student-based certification in formal education settings: Who benefits and what is needed." *Journal of Information Technology Education: Research* 4.1 (2005): 287-306.

[9] Codeforces, on the other hand, is widely recognized for its competitive programming contests, ranking system, and global participation, which helps users improve their coding speed, efficiency, and competitive skills

[10] Afrihyia, Erica, et al. "Enhancing software reliability through automated testing strategies and frameworks in cross-platform digital application environments." *Journal of Frontiers in Multidisciplinary Research* 3.2 (2022): 517-531.

[11] Kochhar, Pavneet Singh, Dinusha Wijedasa, and David Lo. "A large scale study of multiple programming languages and code quality." *2016 IEEE 23Rd international conference on software analysis, evolution, and reengineering (SANER)*. Vol. 1. IEEE, 2016.

[12] Beldar, Pankaj, et al. "Innovative Coding Teaching Methodologies: A Comprehensive Approach for Diverse Learners." *Journal of Engineering Education Transformations* (2025): 52-65.

[13] Chang, Chiung-Sui, Chih-Hung Chung, and Julio Areck Chang. "Influence of problem-based learning games on effective computer programming learning in higher education." *Educational technology research and development* 68.5 (2020): 2615-2634.

[14] Lee, Hsin-Yu, et al. "Integrating computational thinking into scaffolding learning: An innovative approach to enhance science, technology, engineering, and mathematics hands-on learning." *Journal of Educational Computing Research* 62.2 (2024): 211-247.

[15] Shynkarenko, Viktor, and Oleksandr Zhevaho. "Development of a toolkit for analyzing software debugging processes using the constructive approach." *Восточно-Европейский журнал передовых технологий* 5.2-107 (2020): 29-38.

[16] Zeller, Andreas. *Why programs fail: a guide to systematic debugging*. Morgan Kaufmann, 2009.

[17] Fan, Haiyan, and Marshall Scott Poole. "What is personalization? Perspectives on the design and implementation of personalization in information systems." *Journal of Organizational Computing and Electronic Commerce* 16.3-4 (2006): 179-202.

[18] Colace, Francesco, Massimo De Santo, and Mario Vento. "Evaluating on-line learning platforms: a case study." *36th Annual Hawaii International Conference on System Sciences, 2003. Proceedings of the*. IEEE, 2003.

[19] Campbell, Sharon, and Katrina Giadresco. "Computer-assisted clinical coding: A narrative review of the literature on its benefits, limitations, implementation and impact on clinical coding professionals." *Health Information Management Journal* 49.1 (2020): 5-18.

[20] Khider Ismail, Bakhtiar. "Designing and developing an online platform for learning react and vite for beginners: cases: user-centered data collection: integrating surveys and user stories in interviews." (2025).

[21] Guri-Rosenblit, Sarah, and Begoña Gros. "E-learning: Confusing terminology, research gaps and inherent challenges." *International Journal of E-Learning & Distance Education/Revue internationale du e-learning et la formation à distance* 25.1 (2011).

[22] Verdú, Elena, et al. "A distributed system for learning programming on-line." *Computers & Education* 58.1 (2012): 1-10.

[23] Liu, Fangming, et al. "Gearing resource-poor mobile devices with powerful clouds: architectures, challenges, and applications." *IEEE Wireless communications* 20.3 (2013): 14-22.

[24] Zinovieva, I. S., et al. "The use of online coding platforms as additional distance tools in programming education." *Journal of physics: Conference series*. Vol. 1840. No. 1. IOP Publishing, 2021.

[25] Kent, Christopher A., and Jeffrey C. Mogul. *Fragmentation considered harmful*. Vol. 17. No. 5. 1987.