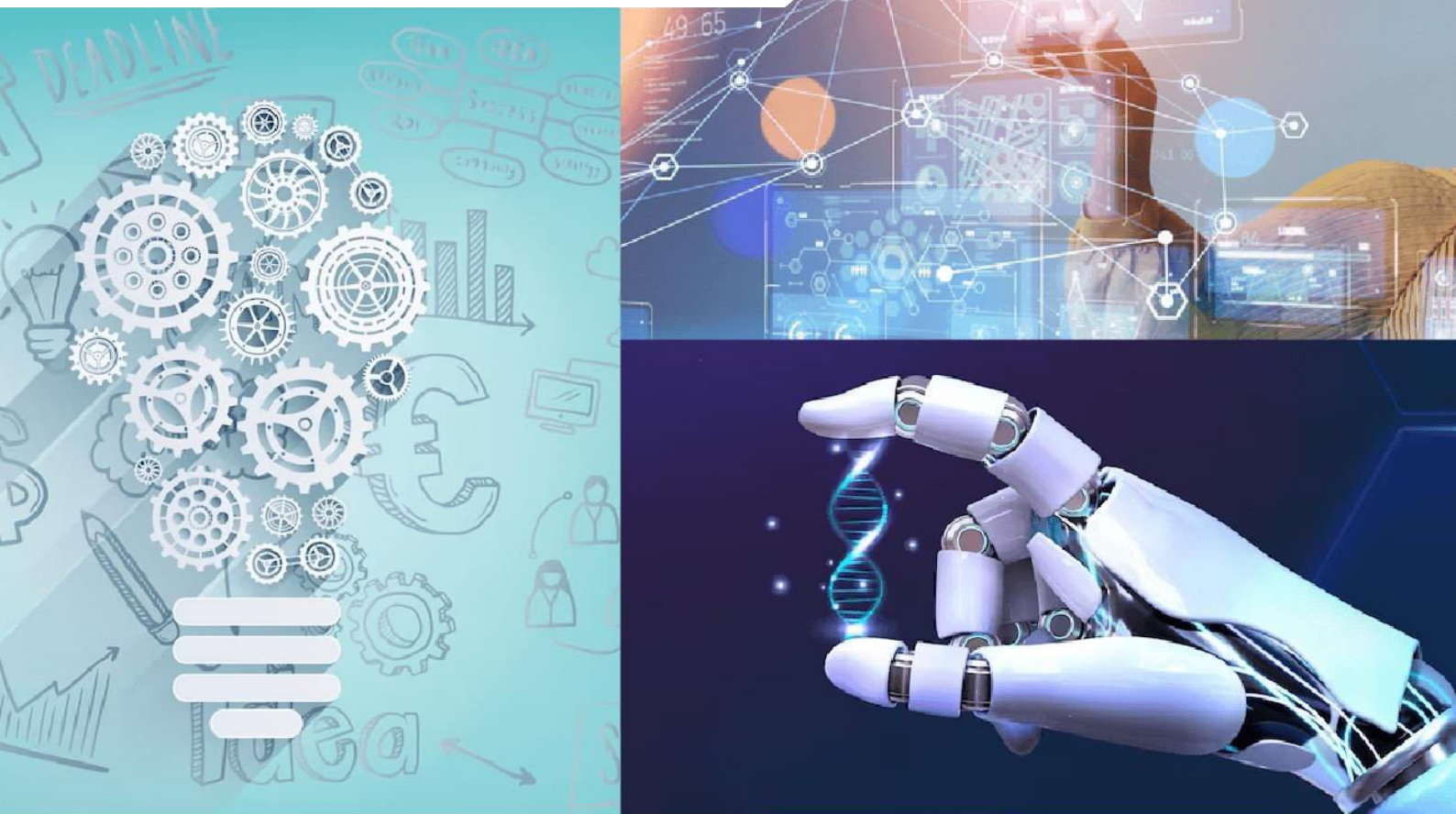




International Journal of Advanced Research in Education and Technology (IJARETY)

Volume 13, Issue 1, January - February 2026

Impact Factor: 8.152



INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



Codeify – Code Review and Execution Platform

Rittika Shaw

Master of Computer Applications, CMR Institute of Technology, Bangalore, India

ABSTRACT: Learning to code often presents challenges, particularly in identifying and fixing errors while following standard programming practices. Beginners frequently struggle with debugging, and feedback is often delayed due to manual code review, slowing down the learning process. To address this issue, this paper presents *Codeify*, a web-based coding assistant that enables users to detect, analyze, and correct errors directly within the browser without requiring installation. The platform supports multiple programming languages and provides real-time analysis of syntax errors, logical flaws, formatting issues, and coding standard violations. It also offers automatic correction suggestions while preserving the original program logic. Additionally, Codeify enables users to execute code within the same environment and view instant outputs along with runtime errors. The system includes a code complexity analysis module and supports voice-to-code functionality and report generation. Overall, the platform improves coding efficiency, enhances learning, and simplifies debugging through an integrated and accessible environment.

KEYWORDS— Codeify, Code Analysis, Debugging Tool, Web-Based IDE, Programming Education, Error Detection, Code Execution, Complexity Analysis, Voice-to-Code, Natural Language Processing, Real-Time Feedback, Browser-Based Application.]

I. INTRODUCTION

The rapid growth of digital technology and increasing demand for programming skills across education and industry have highlighted significant challenges faced by beginners in learning to code effectively. Many students and novice developers struggle with identifying syntax errors, understanding logical flaws, and maintaining proper coding standards. Unlike visible mistakes, logical errors and structural inefficiencies are often difficult to detect, making real-time feedback essential for improving programming skills [1], [2], [3].

Traditional programming education relies heavily on manual code evaluation by instructors, which often leads to delayed feedback and inconsistent assessment. While some automated tools exist, they are typically limited to syntax checking and lack the ability to provide meaningful explanations or real-time interaction. Additionally, many platforms require complex setup and configuration, creating accessibility barriers for beginners [4], [5].

Recent advancements in web-based development environments and intelligent code analysis systems have enabled the creation of more interactive coding platforms. Browser-based tools eliminate installation barriers, while integrated execution environments and automated feedback systems improve learning efficiency. Features such as real-time execution, complexity analysis, and voice-based interaction further enhance accessibility [6], [7].

To address these limitations, this paper presents *Codeify*, a web-based code review and execution platform that integrates writing, analysis, execution, and feedback into a unified environment, improving learning and coding efficiency.

II. LITERATURE REVIEW

A. Automated Code Review Systems

Automated tools provide quick syntax checking but lack detailed explanations for logical errors, limiting their usefulness for beginners [3].

B. Intelligent Tutoring Systems

ITS platforms provide guided learning but are restricted to predefined problems and lack flexibility for real-world coding tasks [4].

C. Static Code Analysis Tools

Linters detect formatting and syntax issues but often produce technical feedback that beginners find difficult to understand [5].

D. Online Code Execution Platforms

These platforms enable easy execution but do not provide insights into code quality or improvement suggestions [6].

E. Proposed System Advantage

Codeify integrates analysis, execution, feedback, and complexity evaluation into a single system, providing real-time and beginner-friendly guidance.

III. METHODOLOGY

The proposed system follows a modular, multi-layered architecture consisting of four layers: Input Layer, Processing Layer, Execution Layer, and Output & Support Utilities. The Input Layer allows users to write or paste code and select a programming language. The Processing Layer performs code analysis to detect syntax errors, logical flaws, and formatting issues. The Execution Layer runs the code securely in a sandbox environment, while the Output Layer displays results, feedback, and reports.

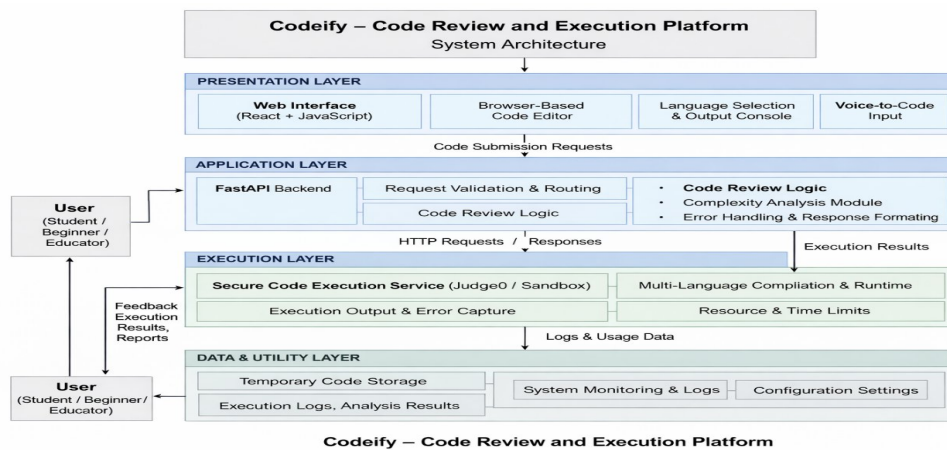


Figure 1: System architecture of Codeify showing Input, Processing, Execution, and Output layers.

The system includes key modules such as a code editor, analysis module, execution engine, and feedback system. The editor handles user input, the analysis module checks errors, the execution engine runs the code, and the feedback module provides suggestions for improvement.

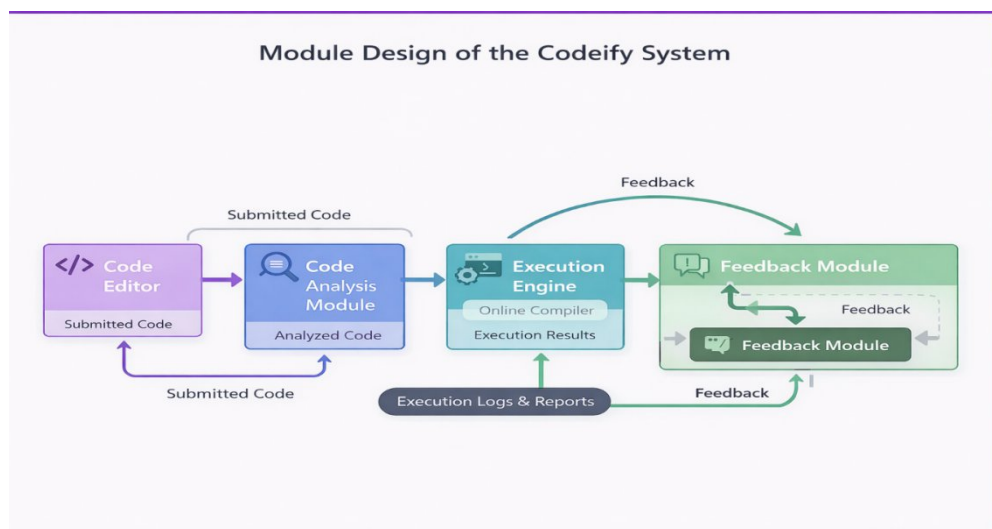


Figure 2: Module design of the Codeify system.

The workflow begins when the user submits code. The system analyzes and executes the code, then generates output and feedback. All results are displayed to the user, ensuring a smooth and interactive coding experience.

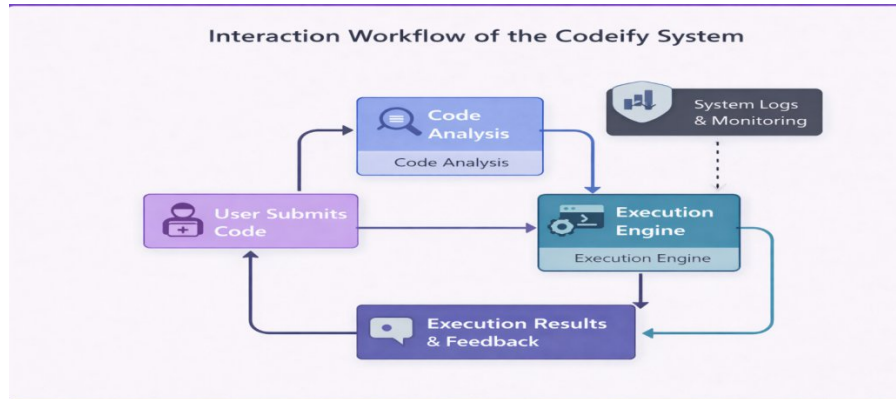


Figure 3: Interaction workflow of the Codeify system.

IV. RESULT AND OUTCOME

The Codeify system was evaluated using multiple programming scenarios to verify its functionality, usability, and performance. Figure 3 shows the home interface of the platform in dark mode. The interface provides a coding editor, language selection, and action buttons such as *Fix Code*, *Review*, and *Run*. The output panel is displayed alongside the editor, creating a clean and efficient coding environment.

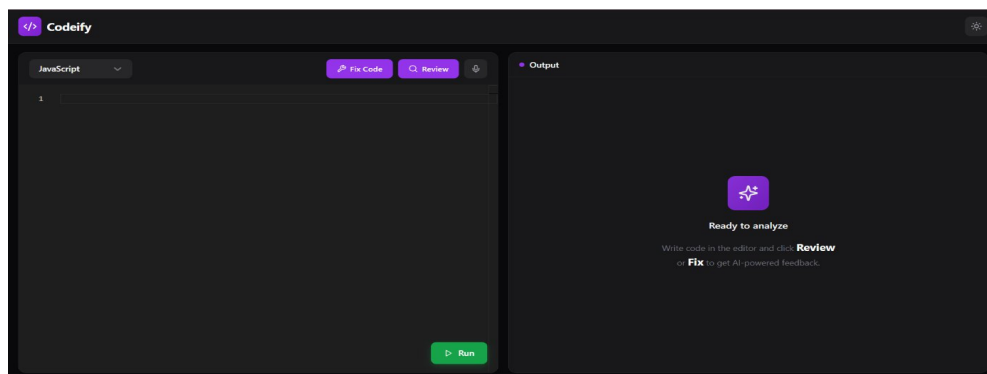


Figure 4: Home Page UI – Dark Mode.

Figure 5 illustrates the home interface in light mode. The system provides the same functionality with a different visual theme to improve accessibility and user comfort. Both modes ensure consistent performance and user experience..

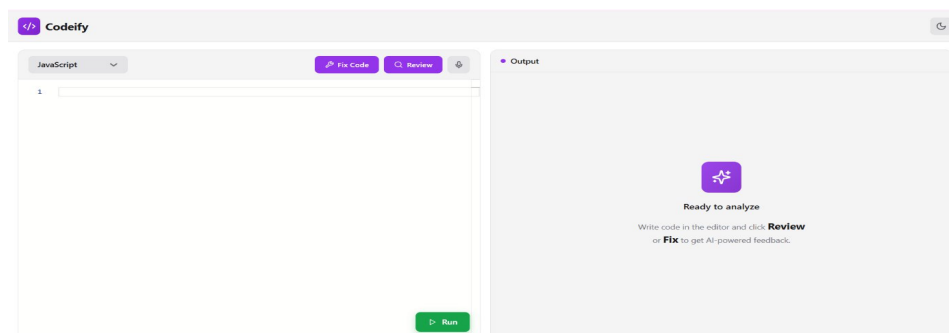


Figure 5: Home Page UI – Light Mode.

Figure 6 demonstrates the *Fix Code* feature. The system automatically detects issues such as incorrect indentation and spacing and provides corrected code along with explanations. This helps users understand mistakes and improve coding practices.

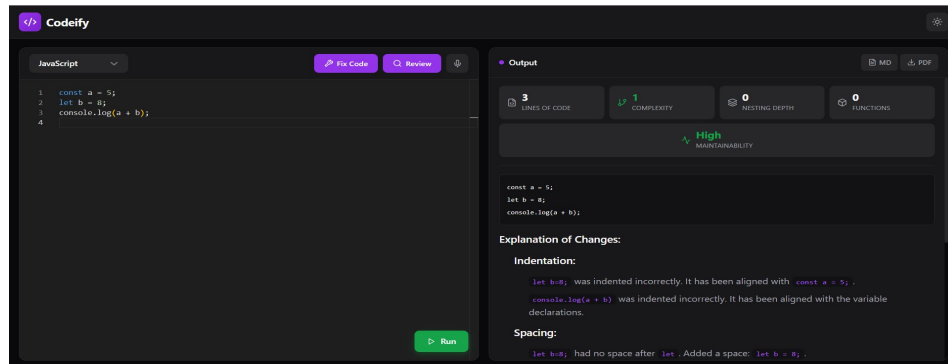


Figure 6: Fix Code feature showing automatic correction and explanation.

Figure 7 presents the *Code Review* functionality. The system evaluates the code and provides a quality rating along with reasoning. This helps users understand the structure, readability, and efficiency of their code.

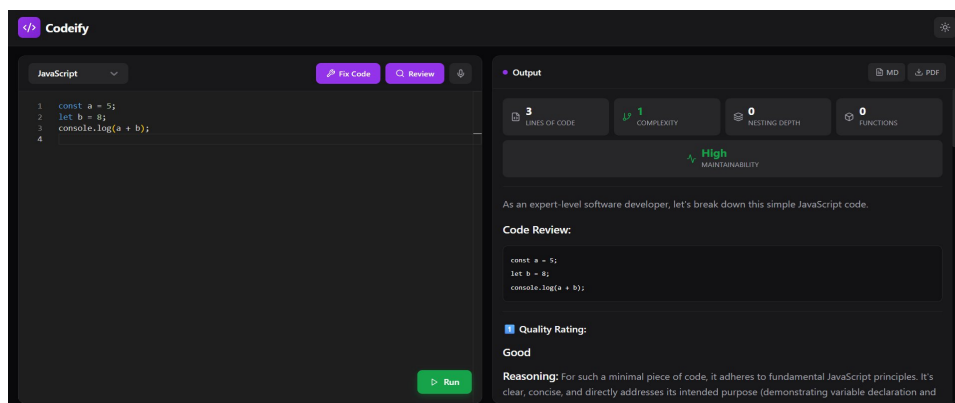


Figure 7: Code Review feature showing quality analysis and feedback.

Figure 8 shows the *Run Code* feature, where the system executes the program and displays the output instantly. This enables users to verify correctness and understand runtime behavior effectively.

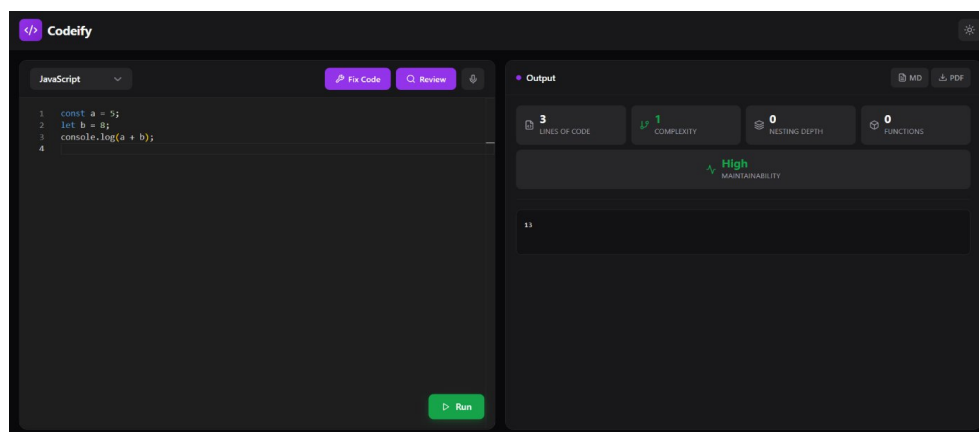


Figure 8: Run Code feature showing execution output.

V. CONCLUSION

This paper presented *Codeify*, a web-based code review and execution platform designed to simplify programming learning and improve coding practices. The system integrates code editing, error detection, execution, and feedback into a single unified environment, eliminating the need for multiple tools and complex setup. By providing real-time analysis of syntax errors, logical flaws, and formatting issues, Codeify enables users to understand and correct their mistakes efficiently [1], [2].

A key strength of the platform lies in its ability to deliver immediate and structured feedback. Features such as automatic code correction, code review with quality assessment, and complexity analysis help users gain deeper insights into their code. The inclusion of multiple interaction modes, including manual coding and voice-to-code functionality, enhances accessibility and supports diverse learning styles [3], [4].

Experimental evaluation demonstrated that the system performs reliably across different programming languages and user scenarios. The platform provides fast execution, accurate feedback, and a consistent user experience across both dark and light interface modes. Error handling mechanisms ensure stability even in edge cases, making the system suitable for educational and practical coding environments [5].

From an application perspective, Codeify serves as an effective learning support tool rather than a replacement for traditional programming education. It bridges the gap between theoretical concepts and practical implementation by enabling hands-on learning with instant guidance. This makes it particularly useful for students, beginners, and self-learners [6].

In conclusion, Codeify demonstrates how modern web technologies and intelligent analysis systems can be combined to create an accessible and efficient coding platform. Future enhancements may include support for additional programming languages, advanced AI-based code analysis, collaborative coding features, and integration with educational platforms to further improve learning outcomes [7].

REFERENCES

- [1] M. Robins, "Learning edge momentum: A new account of outcomes in CS1," *Computer Science Education*, vol. 20, no. 1, pp. 37–71, 2010.
- [2] J. Bennedsen and M. E. Caspersen, "Failure rates in introductory programming," *ACM SIGCSE Bulletin*, vol. 39, no. 2, pp. 32–36, 2007.
- [3] A. Lahtinen, E. Ala-Mutka, and H.-M. Järvinen, "A study of the difficulties of novice programmers," in *Proc. ITiCSE*, 2005, pp. 14–18.
- [4] C. Douce, D. Livingstone, and J. Orwell, "Automatic test-based assessment of programming: A review," *Journal of Educational Resources in Computing*, vol. 5, no. 3, 2005.
- [5] M. Ihanola et al., "Review of recent systems for automatic assessment of programming assignments," in *Proc. Koli Calling*, 2010, pp. 86–93.
- [6] J. Sorva, "Notional machines and introductory programming education," *ACM Transactions on Computing Education*, vol. 13, no. 2, 2013.
- [7] S. H. Edwards, "Improving student performance by evaluating how well students test their own programs," *Journal on Educational Resources in Computing*, vol. 3, no. 3, 2003.
- [8] Judge0, "Online code execution system," 2024. [Online]. Available: <https://judge0.com>
- [9] Monaco Editor, "Code editor for the web," 2024. [Online]. Available: <https://microsoft.github.io/monaco-editor/>
- [10] FastAPI, "Modern, fast web framework for building APIs," 2024. [Online]. Available: <https://fastapi.tiangolo.com>
- [11] React, "A JavaScript library for building user interfaces," 2024. [Online]. Available: <https://react.dev>
- [12] Python Software Foundation, "Python programming language," 2024. [Online]. Available: <https://www.python.org>

International Journal of Advanced Research in Education and Technology

ISSN: 2394-2975

Impact Factor: 8.152