

EduCode: A Beginner-Friendly Educational Programming Environment

D. I. De Silva, S. Vidhanaarachchi, M. W. H. S. L. Ruwanpura

T. D. B. Wickramasinghe, G. T. S. T. Rathnayaka, T. K. Hewapathirana

*Department of Computer Science and Software Engineering
Sri Lanka Institute of Information Technology, Malabe, Sri Lanka*

Abstract: - Coding proficiency has emerged as a crucial skill in the rapidly evolving technological landscape. The objective of this study is to establish an inclusive and engaging learning environment catering to diverse learning styles, empowering novice coders in their educational journey. This research presents the methodology and outcomes of designing and constructing a novel code learning environment specifically tailored for beginners. The environment comprises four key components: a code editor, voice recognition system for coding, peer review system, and code learning tutorials. The code editor, the central interface of the system, offers a user-friendly platform for coding, encouraging experimentation, and fostering collaborative learning. The voice recognition system introduces a new dimension by enabling vocal coding, enhancing accessibility for individuals of all abilities. The peer review system fosters collaborative learning through the provision of constructive feedback, encouraging critical thinking, and elevating code quality. The code learning tutorials provide structured learning pathways that blend theoretical knowledge with hands-on practice, providing a comprehensive learning experience within an integrated code editor. This study not only contributes to the advancement of coding education but also has implications for fostering critical thinking, collaborative skills, and accessibility in the evolving digital landscape.

Keywords: *code learning, novice coders, collaborative learning, technology-enhanced education.*

1. Introduction

In addressing the growing need for coding skills, this study endeavors to simplify coding education and empower individuals from diverse backgrounds to embark on this journey. The research focuses on the creation of a beginner-friendly code-learning environment [1]. Recognizing that conventional one-size-fits-all approaches often fail to cater to the specific needs and learning trajectories of newcomers [2], the study aims to bridge this gap and enhance the teaching and learning of programming through innovative methods and technology [3].

Research conducted by [4] observing a group of undergraduate novice programmers indicated that they encounter numerous errors, spending a considerable amount of time struggling to fix them, with many unable to execute the written code. Many novice programmers often face difficulties in their basic courses [5], and this initial struggle can be particularly daunting, hindering the development of a growth mindset crucial for effective learning [6]. Therefore, one of the primary objectives of this research is to establish an environment that fosters exploration, collaboration, and self-assurance. This objective is manifested through the integration of a code editor and a voice recognition system, which enable learners to engage in coding through various sensory channels. This novel approach not only enhances accessibility for individuals with disabilities but also motivates learners to transcend the conventional boundaries of coding interactions.

Furthermore, collaborative learning plays a pivotal role in nurturing excellent coding skills. The introduction of a Peer Review System within the code learning environment aims to tap into the collective knowledge of a community of learners. This feature facilitates a dynamic exchange of ideas by allowing participants to review, assess, and contribute insights to one another's code. This collaborative mindset not only supports the development

of individual coding skills but also facilitates the creation of a shared knowledge repository, thereby simplifying and enhancing the learning process.

To guide learners systematically, the Code Learning Tutorials tool leads them through a carefully selected sequence of coding principles within a structured framework. This guided exploration strategy, supplemented with practical tasks and interactive assessments, ensures a comprehensive understanding of the world of coding, catering to various learning styles and paces by combining theory with hands-on experience. Consequently, the environment promotes an independent learning experience, enabling users to chart their own path to becoming proficient in coding.

The remainder of research paper is divided into four key sections. The literature review emphasizes the growing demand for coding skills and the value of programming education, highlighting the potential of innovative tools such as interactive code editors and voice-to-code converters. The methodology section delves into the design and development of a user- friendly code learning environment, introducing the main components of EduCode. The proposed system section provides an overview of the educational programming environment, emphasizing the uniqueness of the Voice Recognition System and the collaborative learning enabled by the Peer Review System. Finally, the conclusion summarizes the paper's key findings and discusses the positive impact of the proposed environment on coding education, fostering confidence and enthusiasm among learners.

2. Literature Review

The quick developments in businesses throughout the world in recent years have greatly increased the value of coding abilities. As a result of this change, there is an increasing demand for programmers across a variety of industries, which has renewed interest in programming education. This tendency is a result of the realization among businesses and industries that knowledge of programming languages offers a substantial competitive advantage in the quickly changing technical environment of today [7] [8]. As a result, the number of new programmers has increased. Many of these programmers, lacking prior coding expertise, are eager to acquire practical techniques for mastering programming languages [9]. The ability to easily include cutting-edge technologies into coding instruction websites, such as interactive code editors and voice-to-code converters, presents a chance to improve the learning experience as technology develops [10]. This in-depth study intends to evaluate the body of knowledge on websites for learning to code, with a focus on the addition of voice-to-code converters and code editors as a method of meeting the changing demands of beginning programmers.

The incorporation of powerful code editors and voice-to-code converters into websites that teach coding is still a relatively unexplored area in the fields of programming education and e-learning platforms [11]. Although there is a sizable body of research in this area, it has primarily concentrated on traditional text-based programming tutorials, obviously ignoring the potential of voice-driven programming and interactive code editors to support and enhance the learning experience for beginners [12].

Numerous studies have been conducted to underline the relevance of interactive and engaging learning experiences in effective programming teaching. The research of Xie and Salvendy [13] indicates the importance of visual and interactive components in multimedia-based programming lectures. Johnson and Miller's research [14] looks on the possibilities of voice interfaces in teaching, especially for beginning programmers. The study by Lee and Kao [15] focuses on the influence of real-time feedback and cooperation in code editors on the development of programming abilities. These studies highlight the importance of visual engagement and successful interaction in programming instruction.

Online programming programs have grown in popularity, transforming the digital learning scene. To offer a rich and interesting learning environment, these sessions use a variety of materials such as video lectures, interactive coding exercises, and quizzes. This method adapts to a variety of learning styles, making it appealing to a wide spectrum of novice programmers. Furthermore, granting certificate programs upon the completion of work serves as a significant motivation and provides formal validity for gained abilities. Individual progress tracking, achieved through various mechanisms, also personalizes the learning path, encouraging students' dedication and determination.

As novel innovations such as interactive code editors and voice-to-code converters gain popularity in the field of programming training, the importance of strong customer service, encompassing quick and effective assistance, becomes clear. These developments anticipate expected technological issues, emphasizing the significance of accessible and excellent customer support. Quick and effective assistance is critical for maintaining a pleasant learning experience, avoiding frustration, and encouraging user participation. As these technologies continue to alter programming education, they help to build an overall learning platform that not only improves programming abilities but also encourages long-term growth and a sense of success in learners.

The study aims to fill research gaps by examining voice-based programming, interactive code editing, and digital learning components. It aims to contribute knowledge to programming education, promoting language mastery and personal growth.

3. Methodology

This section discusses the design and development of a user-friendly code learning environment for novice programmers, encompassing four key functionalities: the Code Editor, Voice Recognition System, Peer Review System, and Code Learning Tutorials.

A. Code Editor

The proposed application utilizes *CodeMirrorEditor*, a JavaScript code editor that employs the *CodeMirror* library, to offer users various options such as text editing, syntax highlighting, and error checking. It offers operations for manipulating the code, including adding and deleting text, navigating the cursor, and selecting code segments. This editor incorporates features such as code execution, runtime output capture, and automatically generated formatted code. With a user-friendly design, the code editor enhances the overall coding experience. Fig. 1 & Fig. 2 illustrate the error highlighting and syntax highlighting capabilities of the code editor, while Fig. 3 presents its workflow.

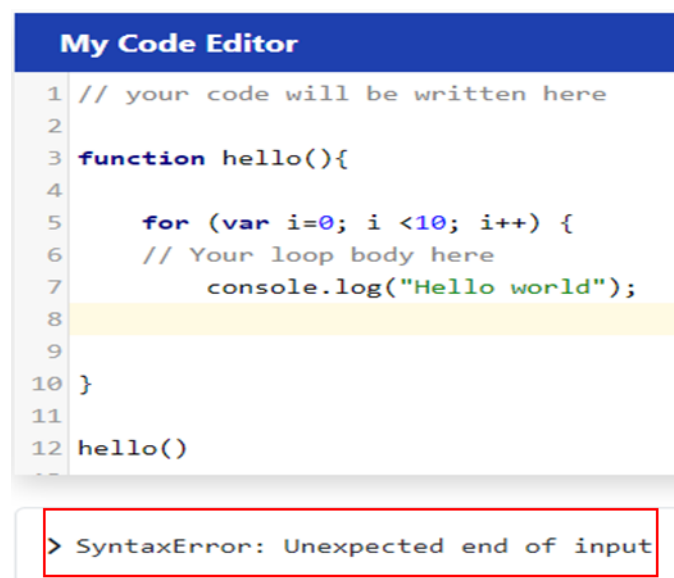


Fig. 1: Error highlighting option of code editor

B. Voice Recognition System

The voice recognition system created using the react- speech-recognition library, enables real-time transcription of voice commands. Users can select their preferred programming language, dynamically affecting code generation commands. The main technical component of the voice recognition system is the *languageCommandHandlers* object, which associates voice commands with specific JavaScript and Java code generation functions. The code generated by these functions is transferred to the editor, allowing users to make further modifications and execute it as needed.

```

1 // your code will be written here
2
3 function hello(){
4
5   for (var i=0; i <10; i++) {
6     // Your loop body here
7     console.log("Hello world");
8   }
9
10 }
11
12 hello()

```

Fig. 2: Syntax highlighting option of code editor.

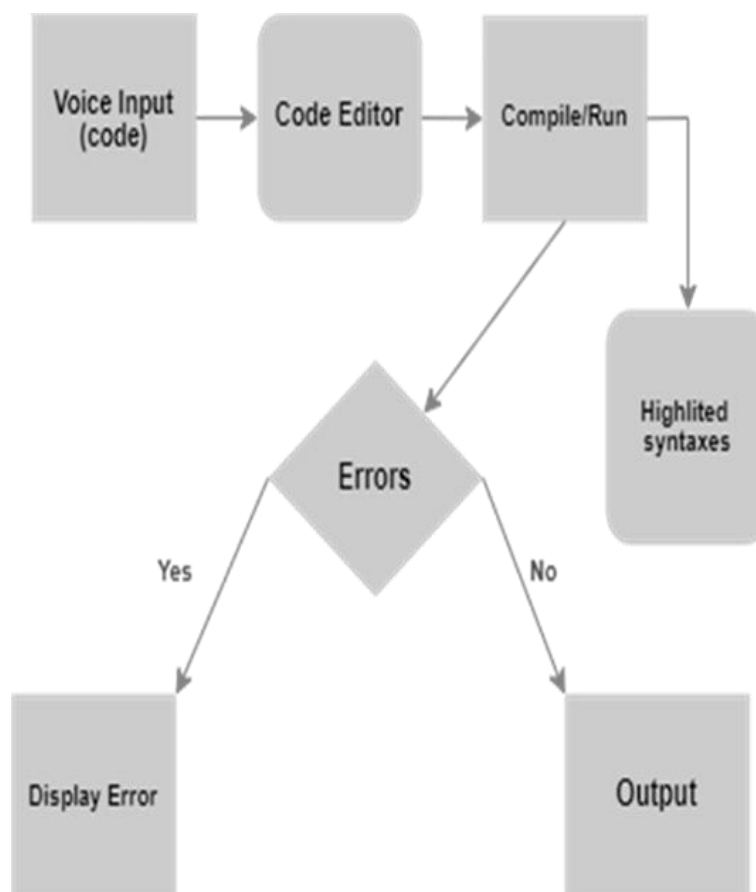


Fig. 3: Workflow of code editor

The proposed system is equipped with features enabling users to effortlessly copy and clear code through voice commands. Additionally, a dedicated guidance component enhances the user experience by providing well-structured instructions and examples for supported voice commands. The proposed system prioritizes readability,

achieved through meticulous documentation, organized code structure, and reusability. Furthermore, the system incorporates robust error handling mechanisms to validate user input. Collectively, these attributes contribute to establishing an innovative and technically advanced coding environment centered around audio interactions.

C. Peer Review System

Built on the MongoDB, Express.js, React, Node.js (MERN) stack, the peer review system follows a user-centric approach. Registered users have access to the system, which allows them to submit code, leave comments, and observe postings and comments. Each post and comments are saved in a MongoDB database, which manages interactions by using unique IDs to ensure data integrity.

JSON Web Tokens (JWT) are used for authentication, validating users' identities, and granting them relevant rights. Only the authors of articles and comments have the option to alter or remove their own inputs, giving them control and accountability.

Essentially, the peer review system establishes a safe and organized environment in which learners can actively participate in peer review, share their knowledge, and engage in collaborative learning. As users contribute constructive comments and explore other viewpoints and techniques, it fosters a culture of continuous growth, critical thinking, and problem-solving, ultimately improving their coding talents and comprehension.

D. Code Learning Tutorial

With each course organized as a collection, MongoDB serves as the storage for class materials, such as lessons and tutorials, in the online course system. The front end, implemented by React.js, presents learning materials and interactive tutorials. *Express* framework connects the backend and frontend. Additionally, MongoDB keeps track of user progress, updating as users finish tutorials and lectures. Furthermore, accessibility across a range of devices and screen sizes is ensured through responsive design concepts and CSS frameworks.

4. Proposed System

The system overview provides a concise insight into the essential components and functionalities of a beginner-friendly educational programming environment, offering a foundational understanding of its structure and capabilities. Fig. 4 provides a visual representation of the proposed system's overview.

A. Code Editor

The code editor is the learning environment's foundation, offering users with an easy interface for coding activities. Its user-friendly interface, syntax highlighting, and programming compilation capabilities seek to make coding easier for novices. It promotes hands-on learning by providing real-time error highlighting and a clean workspace, allowing learners to focus on understanding coding principles and experimenting with different programming languages.

B. Voice Recognition System

The voice recognition system within the code learning environment is an exciting changer in coding education. This one-of-a-kind feature allows users to interact in a creative way by allowing them to orally explain their code, going beyond standard keyboard-based input. The system seamlessly transforms spoken words into the syntax of several programming languages by utilizing cutting-edge voice recognition technology. This strategy promotes accessibility and participation in coding environments that mostly depend on textual input.

This feature removes the requirement for good keyboard skills, making programming activities more accessible to people with a wide range of learning styles and abilities, including those with impairments that make traditional typing difficult. The system understands spoken words and context, accurately maps them to the relevant code structures, providing accuracy and error-free translation. It is a simple learning curve that makes the move from typing to speaking easier, bridging the gap between voice communication and code creation.

C. Peer Review System

Within the coding learning environment, the integrated peer review system plays a critical role in facilitating collaborative and interactive learning. Through a knowledge-sharing mechanism, this technology allows a dynamic feedback loop that improves users' coding abilities and comprehension. Learners submit their code for peer evaluation, receive validation for their work, and are exposed to a variety of viewpoints and techniques.

This method encourages students to explore other solutions and expand their problem-solving talents by promoting critical thinking, analytical skills, and a culture of continuous development. This iterative method, founded on cooperation and reflection, installs a feeling of ownership in learners, encouraging them toward mastery of coding concepts and skill enhancement. Finally, the peer review system encourages a virtuous loop of learning in which individuals actively contribute to the growth of their peers while benefiting from a broad learning experience themselves.

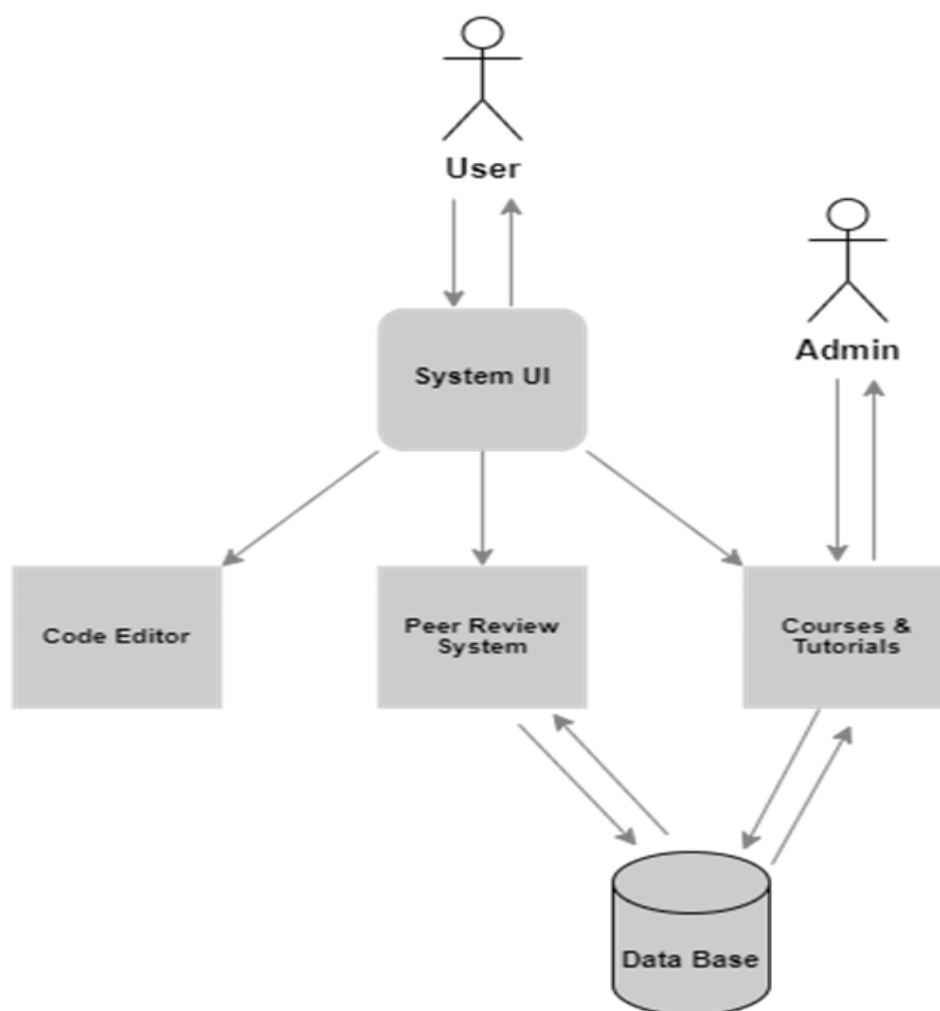


Fig. 4: System overview diagram

D. Code Learning Tutorial

The code learning tutorial is a systematic and thorough coding guidance tool for novices just starting out. It is a vital resource for users at the start of their coding journey, consisting of interactive courses. These courses are designed to accommodate a variety of learning styles and paces, covering coding ideas ranging from fundamental foundations to advanced subjects.

The tutorials provide a comprehensive learning experience by combining theoretical understanding with practical application, complete with real-world examples, hands-on exercises, and engaging activities to aid comprehension and recall. Notably, students have access to a dedicated code editor incorporated into the educational environment, allowing for real-time application of information, and facilitating experiential learning with immediate feedback.

E. Syntax Highlighting and Error Highlighting

The code editor of the proposed system comprises advanced error-highlighting mechanisms to help learners discover and correct syntax issues and mistakes. This feature is intended to improve code professionalism and readability, preparing students for real-world coding tasks. The code editor also employs syntax highlighting, using color and style differences to visibly differentiate various code components such as keywords, variables, strings, and comments. This not only improves code understanding but also promotes the use of proper coding techniques.

These interconnected components collaborate to create a comprehensive educational environment that promotes precision in coding and enhances readability. With these state-of-the-art features, the code learning environment aims to provide novices with a holistic and engaging experience, fostering the establishment of a robust foundation in coding concepts and the development of positive coding habits right from the beginning.

5. Conclusion

The research has unveiled a unique code learning environment designed to address challenges encountered by new coders. This innovative approach paves the way for an inclusive, engaging, and inspiring learning experience, incorporating sophisticated techniques and creative elements. The study proposes a comprehensive approach that accommodates various learning methods and skills of beginner coders. The code editor, voice recognition system, peer review system, and code learning tutorials synergize to provide an organized framework, enabling users to embark on their coding adventure at their own pace.

The voice recognition system offers an innovative interaction method distinct from traditional keyboard input. The peer review system, fostering cooperation and idea sharing among students, contributes to building a sense of community. Furthermore, the code learning tutorials provide a structured approach to assist novices in understanding coding basics.

To alleviate coding difficulty, this learning environment smoothly integrates technology, education, and creativity. It surpasses expectations by nurturing critical thinking, problem-solving abilities, and creativity—all of which are essential skills for success in today's digital world. The study advances education by encouraging students to embrace coding with confidence and enthusiasm, recognizing its critical role in shaping the future.

References

- [1] M. Prensky, "Programming is the new literacy," Edutopia.org, 2008. [Online]. Available: <http://www.edutopia.org/literacy-computerprogramming>. [Accessed: 02-Mar-2016].
- [2] M. Resnick and J. Silverman, "Some Reflections on Designing Construction Kits for Kids," in *Proc. 2005 Conf. on Interaction Design and Children*, New York, USA, 2005, pp. 117-122.
- [3] J. R. Anderson, *The architecture of cognition*, 1st ed., Harvard University Press, 1983.
- [4] S. J. Whittall, W. A. C. Prashandi, G. L. S. Himasha, D. I. De Silva and T. K. Suriyawansa, "CodeMage: Educational programming environment for beginners," *9th Int. Conf. on Knowledge and Smart Technology (KST)*, Chonburi, Thailand, 2017, pp. 311-316, doi: 10.1109/KST.2017.7886101.
- [5] M. Amaratunga, G. Wickramasinghe, M. Deepal, O. Perera, D. De Silva and S. Rajapakse, "An Interactive Programming Assistance tool (iPAT) for instructors and novice programmers," *8th Int. Conf. on Computer Science & Education*, Colombo, Sri Lanka, 2013, pp. 680-684, doi: 10.1109/ICCSE.2013.6553995.
- [6] C. S. Dweck, *Self-Theories: Their Role in Motivation, Personality, and Development*, 1st ed., Psychology Press, 1999.

- [7] J. Smith, "The Growing Importance of Coding Skills in the Digital Age," *J. Technol. Educ.*, vol. 32, no. 2, pp. 45-58, 2020.
- [8] J. B. Clark, and L. White, "Novice Programmers' Learning Needs and Challenges," *Comput. & Educ.*, vol. 145, no. 3, pp.14-18, 2019.
- [9] R. Brown and M. Lee, "Learning Programming from Scratch: Novice Challenges and Strategies," *Educ. Technol. Res. Dev.*, vol. 66, no. 2, pp. 541-560, 2018.
- [10] S. Garcia and L. Martinez, "Enhancing Programming Learning Through Voice-to-Code Converters," *J. Interact. Technol. Pedagogy*, vol. 10, no. 1, pp. 30-48, 2022.
- [11] Q. Wang and H. Chen, "Interactive Code Editors: A Review of Their Impact on Novice Programmers' Learning," *Comput. Hum. Behav.*, vol. 112, 2021.
- [12] L. Chen and H. Wang, "Enhancing Novice Programmers' Learning Experience through Interactive E-learning Platforms," *Interact. Learn. Environ.*, vol. 28, no. 5, pp. 667-683, 2020.
- [13] H. Xie and G. Salvendy, "Multimedia-based programming tutorials: Impact of modes and text-labeling formats," *Univ. Access Inf. Soc.*, vol. 17, no. 3, pp. 643-657, 2018.
- [14] Johnson and C. Miller, "Voice interfaces in education: The importance of dialogue in learning," *TechTrends*, vol. 63, no. 4, pp. 442-449, 2019.
- [15] T. C. Lee and G. Y. Kao, "Collaborative programming supported by real-time feedback," *IEEE Trans. Learn. Technol.*, vol. 11, no. 2, pp. 213-255, 2017.