CODEPILOT: An AI-Powered Interactive Platform for Learning Data Structures and Algorithms”

Gaurav S. Men[1], Parth N. Kulkarni[2], Vedant P. Lohar[3], Atharva S. Mujumale[4], H. A. Shide[5]
[5]*Guide, Computer Engineering, AISSMS Polytechnic, Pune, India*
[1,2,3,,45]*Member, Computer Engineering, AISSMS Polytechnic, Pune, India*

Abstract - Learning Data Structures and Algorithms (DSA) is a fundamental requirement for students pursuing computer science and software development. However, many beginners face significant difficulties in understanding algorithmic logic, visualizing data structure operations, and applying theoretical concepts to practical coding problems. Traditional learning methods such as textbooks, static tutorials, and video lectures often lack interactivity and real-time guidance, which can result in reduced learner engagement and conceptual confusion.

This paper presents CodePilot, an AI-assisted interactive learning platform designed to simplify the study of Data Structures and Algorithms for beginners. The proposed system integrates artificial intelligence with visualization tools, interactive coding environments, and structured learning modules to create an engaging educational experience. The platform provides step-by-step explanations of algorithms, real-time code execution, algorithm visualization, and intelligent assistance to help learners understand complex concepts more effectively.

Index Terms — Artificial Intelligence, Coding Assistant, Generative AI, Programming Education, Natural Language Processing

I. INTRODUCTION

Data Structures and Algorithms (DSA) form the foundation of computer science and software engineering. They are essential for designing efficient programs, optimizing computational resources, and solving complex real-world problems. Mastery of data structures such as arrays, linked lists, stacks, queues, trees, and graphs enables programmers to organize and process data efficiently. Similarly, algorithms provide systematic procedures for solving computational tasks, making them a crucial component of modern software development. As a result, DSA knowledge is widely required in academic curricula, technical interviews, and professional programming environments.

Despite their importance, many beginners struggle to learn and understand data structures and algorithms. The primary challenges include difficulty in visualizing how algorithms operate internally, understanding the logical flow of data manipulation, and translating theoretical concepts into working code. Traditional learning resources such as textbooks, lecture notes, and pre-recorded tutorials often present static explanations that lack interactivity. Without hands-on experimentation and real-time guidance, students may find it difficult to grasp the underlying logic of algorithmic operations.

The objective of the proposed system is to create an engaging and beginner-friendly environment that simplifies complex programming concepts through visualization, experimentation, and intelligent assistance. By combining artificial intelligence with interactive educational tools, CodePilot aims to reduce learning barriers, improve conceptual understanding, and promote active learning in the field of data structures and algorithms.

II. LITERATURE REVIEW / RELATED WORK

The rapid growth of programming education and online learning platforms has encouraged researchers to explore innovative approaches for teaching Data Structures and Algorithms (DSA). Traditional teaching methods rely heavily on theoretical explanations and static examples, which often fail to provide the level of interactivity required for beginners to fully understand algorithmic behavior. As a result, several studies have focused on developing intelligent systems, visualization tools, and AI-powered platforms to improve programming education.

Early research on algorithm visualization systems demonstrated that graphical representations of algorithms significantly improve students' ability to understand complex computational processes. Visualization tools allow learners to observe step-by-step execution of algorithms, making abstract concepts more concrete and easier to comprehend. Systems such as algorithm animation frameworks provide dynamic demonstrations of sorting algorithms, tree operations, and graph traversal techniques. However, many of these systems focus only on visualization and do not provide personalized guidance or explanations for learners.

With the advancement of artificial intelligence and machine learning technologies, AI-assisted programming tools have gained considerable attention. Modern AI-based coding assistants are capable of generating code, providing syntax suggestions, and offering debugging support. These systems are widely used in professional software development environments to increase productivity and reduce development time. While such tools demonstrate strong capabilities in code generation and completion, they are generally designed for experienced developers and may not provide beginner-friendly explanations or educational guidance.

Recent studies have also examined the role of intelligent tutoring systems (ITS) in computer science education. Intelligent tutoring systems aim to replicate the behavior of a human tutor by providing adaptive feedback, personalized learning paths, and interactive problem-solving assistance. Research indicates that students using ITS-based platforms show improved engagement and better retention of programming concepts. These systems help learners identify mistakes, understand algorithmic logic, and gradually develop problem-solving skills through guided learning.

III. PROBLEM STATEMENT

Learning Data Structures and Algorithms (DSA) is an essential part of computer science education and software development training. These concepts form the basis for efficient problem solving, optimized programming, and the design of scalable systems. However, beginners often experience significant difficulties when attempting to understand and implement data structures and algorithms due to the abstract nature of the concepts and the lack of intuitive learning tools. Many students find it challenging to visualize how algorithms operate internally, how data moves within structures, and how different operations affect system performance.

IV. PROPOSED SYSTEM / METHODOLOGY

The proposed system, CodePilot, is an AI-assisted interactive learning platform designed to help beginners understand Data Structures and Algorithms more effectively. The system integrates visualization tools, an interactive coding environment, and AI-based analysis to provide a practical and engaging learning experience. The primary objective of the platform is to simplify complex algorithmic concepts and allow users to experiment with code while observing how algorithms work internally.

The system follows a modular architecture consisting of a frontend interface, backend processing layer, AI analysis module, and data management system. The user interface module provides a simple and intuitive dashboard where learners can explore different data structures and algorithms. The interactive coding playground allows users to write and execute code in real time, helping them test algorithm implementations and understand program logic.

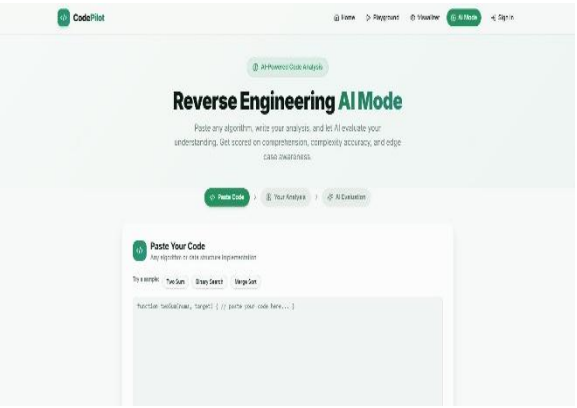
The algorithm visualization module graphically demonstrates how algorithms and data structures operate step by step. This helps users observe operations such as insertion, deletion, searching, and traversal in structures like arrays, linked lists, stacks, queues, binary search trees, and graphs. In addition, the platform includes an AI analysis and reverse engineering module that assists learners in understanding algorithm behavior and provides explanations for outputs generated by different algorithms.

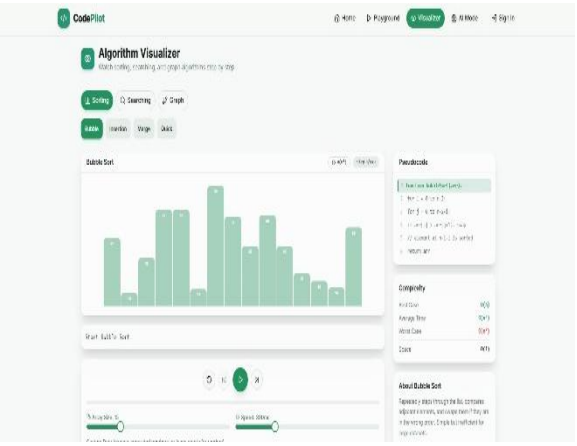
SYSTEM ARCHITECTURE

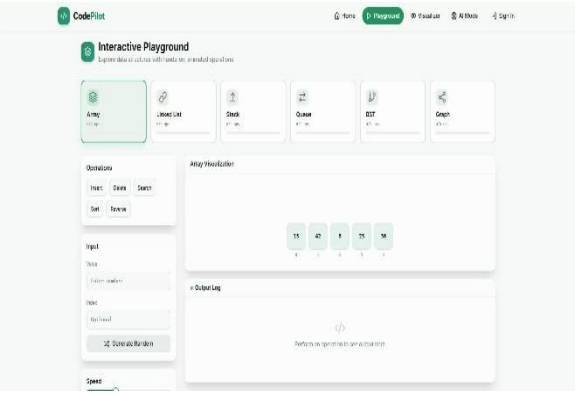
The system architecture of CodePilot follows a modular client–server design to ensure flexibility, scalability, and efficient performance. The architecture mainly consists of four major components: the Frontend Interface, Backend

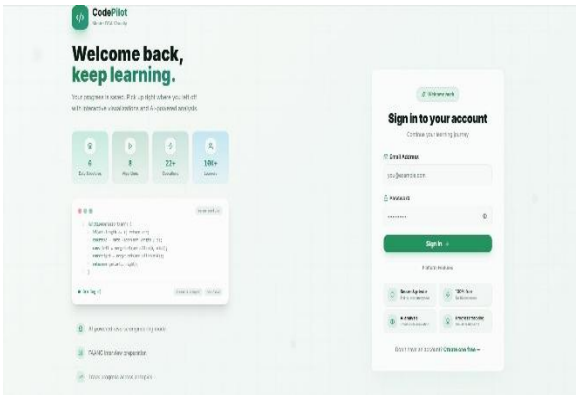
Processing Layer, AI Analysis Module, and Data Storage System.

The Frontend Interface provides a user-friendly platform where learners can interact with the system. Through this interface, users can access algorithm learning modules, visualize data structures, and experiment with code in the interactive playground. The interface is designed to be simple and intuitive so that beginners can easily navigate different features of the platform.









V. ADVANTAGES AND PROPOSED BENEFITS

The proposed system offers several advantages for beginners learning Data Structures and Algorithms. First, the platform provides an interactive learning environment that allows users to explore algorithms and data structures through visualization and hands-on experimentation. This improves conceptual understanding and helps learners grasp complex topics more easily.

Another important advantage is the integration of an AI-assisted analysis system that provides explanations and guidance during coding practice. This helps beginners understand errors, improve problem-solving skills, and learn algorithmic logic more effectively.

The system also includes an interactive coding playground, which allows users to write, modify, and execute code in real time. This feature encourages experimentation and helps learners observe how different algorithms behave under various conditions.

VI. CHALLENGES AND LIMITATIONS

Although the proposed system offers several benefits, it also faces certain challenges and limitations. One of the primary limitations is that the effectiveness of the system depends on the accuracy and capability of the AI analysis module. If the AI model provides incorrect or unclear explanations, it may affect the learning experience of the users.

Another challenge involves handling highly complex algorithms or advanced programming scenarios. While the system is designed primarily for beginners, more advanced topics may require additional explanations and deeper technical analysis that may not always be fully supported by the platform.

VII. CONCLUSION

This paper presented CodePilot, an AI-assisted interactive platform designed to improve the learning experience of Data Structures and Algorithms for beginner programmers. The system integrates algorithm visualization, an interactive coding playground, and AI-based analysis to create a comprehensive learning environment. By providing real-time interaction and graphical representation of algorithm operations, the platform helps learners better understand complex programming concepts.

The proposed system addresses common challenges faced by beginners, such as difficulty in visualizing algorithm behavior and understanding the logical flow of data structures. Through interactive experimentation and intelligent assistance, the platform promotes active learning and strengthens problem-solving abilities.

REFERENCES

[1] A. Chen, P. Jain, and K. Lee, "Evaluating Language Models Trained on Code," *arXiv preprint arXiv:2107.03374*, 2021.

[2] T. Nijkamp, B. Pang, and C. Zhang, "Small Language Models for Code Optimization," *arXiv preprint arXiv:2310.01333*, 2023.

[3] B. Vaithilingam, P. Francis, and S. Pradel, "Expectations, Outcomes, and Challenges of Modern Code Completion Tools," in *Proceedings of ACM ISSTA*, 2022.

[4] A. Vaswani et al., "Attention Is All You Need," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[5] T. Wolf et al., "Transformers: State-of-the-Art Natural Language Processing," in *Proceedings of EMNLP*, 2020.

[6] L. C. Page and H. Gehlbach, "How an Artificially Intelligent Virtual Assistant Helps Students Navigate the Road to College," *Journal of Educational Technology*, 2019.

[7] M. Mekni, "Smart Virtual Assistant for Students," St. Cloud State University, USA, 2020.

[8] Z. Baani and D. Sulieman, "Smart Virtual Assistant for Students," *International Journal of Computer Applications*, 2021.

[9] P. Chiranjeevi et al., "A Coding Chatbot Using Generative Artificial Intelligence," *International Journal for Interdisciplinary Sciences and Engineering Applications*, 2025.

[10] J. Kocielnik et al., "Guidelines for Human-AI Interaction," in *Proceedings of CHI Conference on Human Factors in Computing Systems*, 2019.