

# Prepal AI: A Platform for End-to-End Technical Interview Preparation

Harshal Arya | Tauqeer Kondkari | Shaunak Sangle | Karan Mhatre | Archana Haral  
Department of Artificial Intelligence and Data Science  
A. C. Patil College of Engineering, Navi Mumbai, India

**ABSTRACT** This paper presents PrepPal AI, an integrated AI-driven platform designed to streamline technical interview preparation for engineering students. The increasing competition in the technology job market requires candidates to develop strong technical skills, effective problem-solving abilities, and professional communication. However, existing preparation resources are highly fragmented, forcing students to rely on multiple platforms for learning, coding practice, mock interviews, and resume building, resulting in inefficient preparation and limited progress tracking.

The proposed system integrates interactive learning modules, a coding practice environment, peer-to-peer mock interviews, and AI-based resume optimization within a unified framework. By leveraging real-time collaboration technologies and automated evaluation mechanisms, the platform provides personalized feedback and continuous performance tracking.

The implementation of PrepPal AI demonstrates improved usability and structured preparation by enabling users to identify skill gaps, enhance problem-solving capabilities, and simulate real-world interview scenarios. The results indicate that an integrated approach significantly improves overall interview readiness and helps bridge the gap between academic learning and industry expectations.

**KEYWORDS** Technical Interview Preparation, Artificial Intelligence, Resume Optimization, Coding Practice, Mock Interviews, EdTech Platform.

## I. INTRODUCTION

The rapid growth of the technology industry has significantly increased the competition among engineering graduates seeking technical roles. Employers now expect candidates to demonstrate strong proficiency in Data Structures and Algorithms, core computer science fundamentals, and effective problem-solving abilities. In addition to technical expertise, candidates are evaluated on their communication skills, interview performance, and the quality of their resumes, which are often screened using Applicant Tracking Systems (ATS).

Despite the availability of numerous online platforms for learning and practice, the current ecosystem for technical interview preparation remains highly fragmented. Students typically rely on separate tools for coding practice, conceptual learning, resume building, and interview preparation. This disjointed approach leads to inefficiencies, lack of structured guidance, and difficulty in tracking overall progress. As a result, many students struggle to identify their weaknesses and fail to prepare effectively for real-world recruitment processes.

Furthermore, existing platforms provide limited personalization and rarely offer integrated feedback across different aspects of

preparation. Mock interviews, which play a critical role in improving confidence and communication skills, are often underutilized due to the absence of accessible and structured peer-to-peer environments. Similarly, resume optimization tools focus primarily on keyword matching without considering a candidate's actual skill set or learning progress.

To address these challenges, this paper proposes **PrepPal AI**, an integrated AI-driven platform designed to support end-to-end technical interview preparation. The system combines interactive learning modules, a coding practice environment, peer-to-peer mock interviews, and AI-powered resume optimization within a unified dashboard. By leveraging real-time collaboration, personalized feedback, and performance tracking, PrepPal AI enables students to systematically improve their skills and readiness for technical interviews.

The proposed platform aims to bridge the gap between academic learning and industry expectations by providing a structured, intelligent, and user-centric approach to interview preparation.

## II. PROBLEM STATEMENT

Engineering students preparing for technical job roles are required to develop proficiency across multiple domains, including Data Structures and Algorithms, core computer science concepts, interview communication skills, and professional resume building. However, the current ecosystem of interview preparation tools is highly fragmented, with different platforms catering to isolated aspects such as coding practice, learning resources, resume creation, and mock interviews.

This lack of integration creates significant challenges for students. Firstly, learners are unable to maintain a unified view of their progress, as performance data is scattered across multiple platforms. This makes it difficult to identify skill gaps, track improvement over time, and design a structured preparation strategy. Secondly, most existing systems provide limited personalization, offering generic content and feedback that does not adapt to individual learning pace, strengths, or weaknesses.

Another major limitation is the absence of accessible and structured mock interview environments. While technical interviews require not only problem-solving skills but also communication and real-time thinking, many students lack opportunities for realistic interview practice. Existing platforms either do not support peer-to-peer interaction or fail to provide meaningful feedback, resulting in low confidence and inadequate preparation.

Additionally, current resume optimization tools primarily focus on keyword-based matching for Applicant Tracking Systems (ATS), without considering the candidate's actual technical skills, project experience, or learning progress. This often leads to misalignment between a candidate's resume and their true capabilities.

## III. LITERATURE REVIEW

Recent advancements in educational technology and artificial intelligence have led to the development of various tools aimed at improving technical interview preparation. Several platforms focus on coding practice and algorithmic problem-solving, enabling students to enhance their proficiency in Data Structures and Algorithms through structured problem sets and competitive programming environments. These systems provide curated questions, automated evaluation, and performance tracking; however, they primarily emphasize coding skills and do not address other critical aspects of

interview preparation such as communication and resume building.

Research studies have also explored the effectiveness of AI-driven resume optimization systems, particularly in improving compatibility with Applicant Tracking Systems (ATS). These tools utilize natural language processing techniques to analyze job descriptions, extract relevant keywords, and suggest modifications to resume content. While such approaches increase the likelihood of shortlisting, existing systems often generate generalized recommendations without considering the candidate's actual technical capabilities, project experience, or learning progression. This limitation reduces the practical effectiveness of resume optimization in real-world scenarios.

In addition, several studies highlight the importance of mock interviews in enhancing employability skills. Structured interview practice has been shown to improve candidate confidence, communication ability, and real-time problem-solving performance. However, current platforms provide limited access to realistic mock interview environments, and many lack peer-to-peer interaction or guided feedback mechanisms. As a result, students often enter recruitment processes without adequate exposure to actual interview conditions.

Furthermore, existing research indicates that the lack of integration among learning platforms is a major barrier to effective preparation. Students typically use multiple independent tools for learning concepts, solving coding problems, practicing interviews, and building resumes. This fragmented approach leads to inefficient learning, absence of centralized progress tracking, and difficulty in identifying overall performance trends.

These observations from existing literature highlight the need for a comprehensive and integrated system that combines intelligent learning, adaptive practice, realistic interview simulation, and personalized feedback within a unified framework.

## IV. RESEARCH GAP

The analysis of existing literature reveals that although numerous platforms and tools have been developed to support technical interview preparation, they largely operate in isolation and address only specific aspects of the preparation process. Coding platforms primarily focus on algorithmic problem-solving, while resume optimization tools emphasize keyword matching for Applicant Tracking Systems (ATS). Similarly,

learning platforms provide conceptual understanding, and a few systems offer limited mock interview capabilities. However, these functionalities are not integrated into a single cohesive environment.

A major gap identified in current systems is the absence of a unified platform that enables end-to-end interview preparation. The lack of integration results in fragmented learning experiences, where students are required to switch between multiple tools, leading to inefficiency and poor progress tracking. There is no centralized mechanism to monitor overall performance, identify skill gaps, or provide a holistic view of a candidate's readiness.

Furthermore, existing solutions provide minimal personalization and fail to deliver adaptive feedback based on individual performance. Resume optimization tools often generate generic suggestions without aligning with a candidate's technical abilities or learning progress. Similarly, mock interview platforms lack structured peer-to-peer interaction, real-time collaboration, and comprehensive feedback mechanisms, limiting their effectiveness in improving communication and interview skills.

Another significant gap is the lack of real-time, collaborative environments that simulate actual interview scenarios. Most platforms do not support integrated coding, communication, and evaluation within a single session, which is essential for replicating real-world technical interviews.

Moreover, the integration of AI-driven analytics can help identify individual strengths and weaknesses, enabling adaptive learning paths tailored to each user. The system can also foster peer-to-peer interaction, encouraging collaborative problem-solving and knowledge sharing among learners. By leveraging real-time performance insights, users can continuously refine their skills and improve interview readiness. Ultimately, such a comprehensive platform can bridge the gap between theoretical knowledge and practical application, ensuring a more structured and efficient preparation process.

In addition to these challenges, issues related to scalability, accessibility, and user engagement further limit the effectiveness of current platforms. Many systems are not designed to handle large-scale user interaction or provide seamless performance across diverse devices and network conditions. Furthermore, the lack of gamification elements and motivational features often leads to reduced user engagement over time. This would not only enhance user retention but also create a more engaging and immersive learning environment for continuous skill development.

## V. PROPOSED SYSTEM ARCHITECTURE

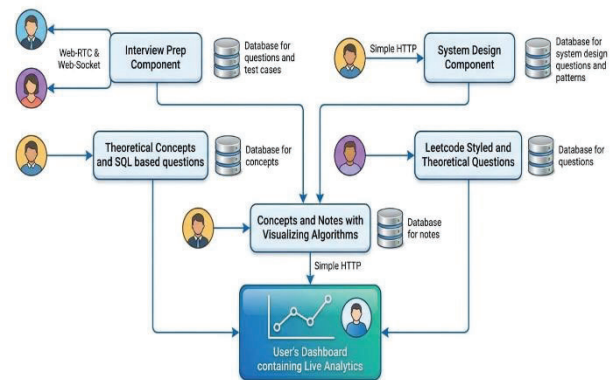


Fig. 1. System Architecture

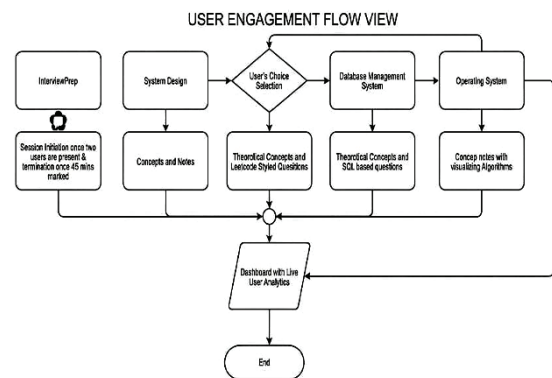


Fig. 2. User Flow Diagram

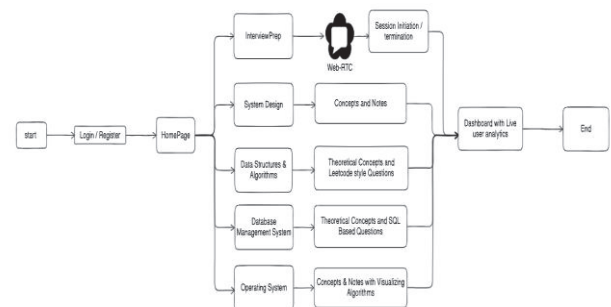


Fig. 3. Module Interaction Flowchart

To address the limitations identified in existing systems, this paper proposes **PrepPal AI**, an integrated and intelligent platform designed to support end-to-end technical interview preparation. The system architecture is designed to combine multiple functionalities, including learning, coding practice, mock interviews, and resume optimization, within a unified and scalable framework.

The overall architecture of PrepPal AI consists of a client-server model, where the frontend interface interacts with backend services through secure APIs. The system is structured into multiple interconnected modules that collaboratively deliver a seamless user experience. The key components of the proposed architecture are described as follows:

#### **A. User Interface Module**

The user interface module provides an interactive dashboard that allows users to access all features of the platform, including learning resources, coding problems, mock interviews, and resume tools. It is designed to offer a user-friendly experience with real-time updates, progress visualization, and personalized recommendations.

#### **B. Authentication and User Management Module**

This module handles secure user authentication and profile management. It stores user-specific data such as skill level, preferred topics, programming languages, and historical performance. This information is used to personalize the user experience and generate adaptive recommendations.

#### **C. Learning and Content Management Module**

This module provides structured learning materials for core computer science subjects, including Data Structures, Algorithms, and system design concepts. Content is organized based on difficulty levels and topics, enabling users to follow a guided learning path.

#### **D. Coding Practice and Evaluation Module**

The coding module allows users to solve problems in a real-time code editor environment. Submitted solutions are compiled and executed in a sandboxed runtime, and results are evaluated against predefined test cases. Performance metrics such as execution time, memory usage, and correctness are recorded for further analysis.

#### **E. Mock Interview and Collaboration Module**

This module enables peer-to-peer mock interviews using real-time communication technologies. It integrates video and audio interaction with a shared coding environment, allowing users to simulate real interview scenarios. Technologies such as WebRTC and WebSocket are utilized to ensure low-latency communication and synchronization between participants.

#### **F. Resume Optimization Module**

The resume module leverages AI-based techniques to analyze and improve resumes. It evaluates content based on job descriptions, identifies missing keywords, and suggests improvements to enhance ATS compatibility. Unlike

conventional tools, this module can be integrated with user performance data to provide more relevant and personalized recommendations.

#### **G. Performance Tracking and Recommendation Module**

This module aggregates user activity data across all components of the platform. It generates insights into strengths and weaknesses, tracks progress over time, and provides personalized recommendations to guide preparation strategies.

The interaction between these modules is coordinated through backend services and databases, ensuring efficient data flow and system scalability..

### **VI. METHODOLOGY AND ALGORITHM**

The working of PrepPal AI is based on a structured workflow that integrates user interaction, real-time collaboration, and automated evaluation to provide an efficient interview preparation environment. The methodology is designed as a sequence of interconnected steps, ensuring seamless data flow and system functionality.

#### **Step 1: User Authentication and Profile Initialization**

The process begins with secure user authentication using a login mechanism. Once authenticated, user-specific information such as skill level, preferred topics, and programming language is retrieved from the database. This data is used to personalize the platform experience.

#### **Step 2: User Matching and Session Creation**

For mock interview sessions, a participant matching algorithm is employed to pair users based on predefined criteria such as skill level and topic preference. A unique session identifier is generated to manage secure communication and interaction between participants.

#### **Step 3: Real-Time Communication Setup**

The system initializes real-time communication channels using WebRTC for peer-to-peer audio and video interaction, and WebSocket protocols for low-latency data exchange. This ensures seamless synchronization during mock interview sessions.

#### **Step 4: Problem Selection and Environment Setup**

A coding problem is dynamically selected from a categorized repository based on difficulty level and topic relevance. A shared code editor environment is then instantiated, allowing both participants to interact within the same workspace.

### Step 5: Collaborative Coding and Synchronization

Real-time collaborative coding is enabled through event-driven updates. Any changes made by one participant are instantly reflected on the other side, ensuring consistency and enabling effective interaction during problem-solving.

### Step 6: Code Execution and Evaluation

Submitted code is executed within a secure sandboxed runtime environment to prevent unauthorized access and ensure system safety. The solution is evaluated against a set of predefined test cases, including both visible and hidden cases, to verify correctness.

### Step 7: Performance Analysis

After execution, key performance metrics such as execution time, memory usage, and success rate are computed. These metrics provide quantitative insights into the user's coding efficiency and problem-solving ability.

### Step 8: Feedback Generation and Storage

A structured feedback report is generated based on performance metrics and interaction data. This report is stored in the database and visualized on the user dashboard, enabling users to track their progress and identify areas for improvement.

The above methodology ensures an integrated and systematic approach to interview preparation by combining authentication, collaboration, evaluation, and feedback mechanisms within a single workflow.

## VII. RESULTS AND DISCUSSION

The proposed system, PrepPal AI, was successfully developed and evaluated to analyze its effectiveness in improving technical interview preparation. The platform integrates multiple modules, including learning, coding practice, mock interviews, and resume optimization, within a unified interface. The results demonstrate the practical usability and functional efficiency of the system.

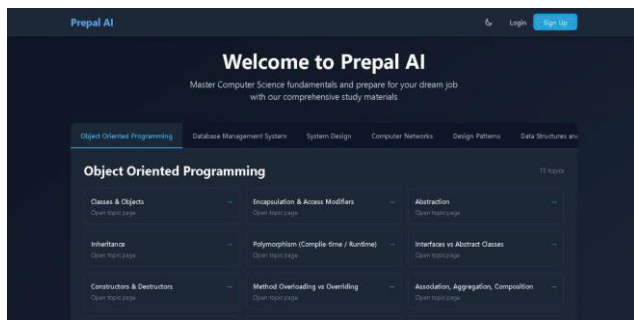


Fig. 1. PrepPal AI Homepage

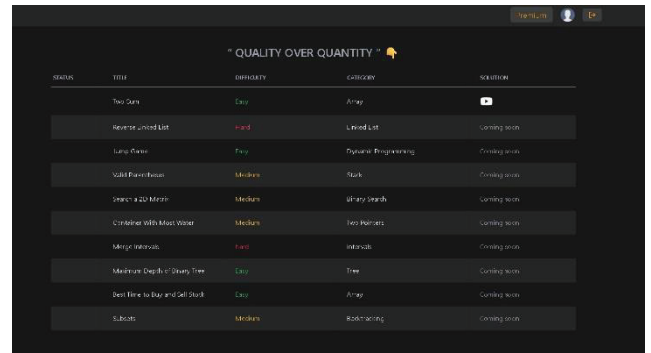


Fig. 2. Problem Solving and Code Editor Interface

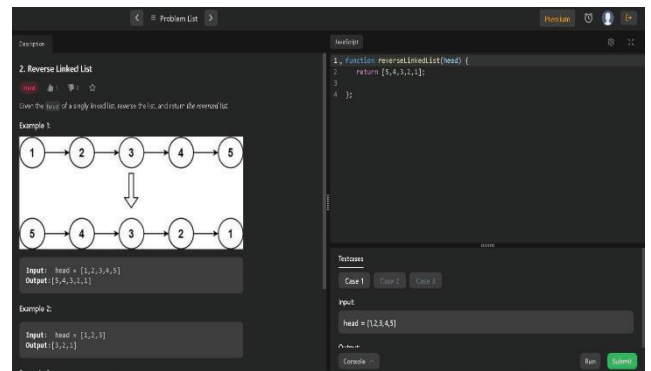


Fig. 3. Problem Interface

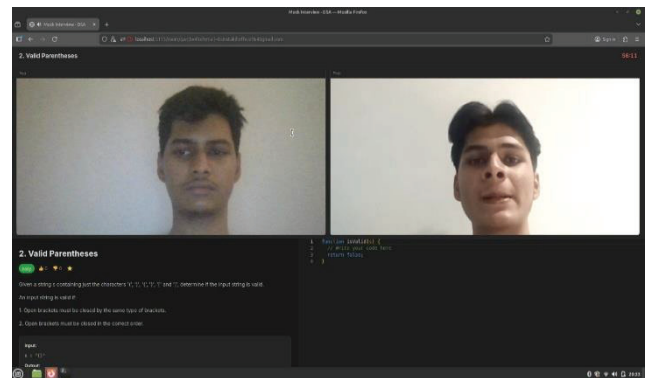


Fig. 4. Mock Interview Interface

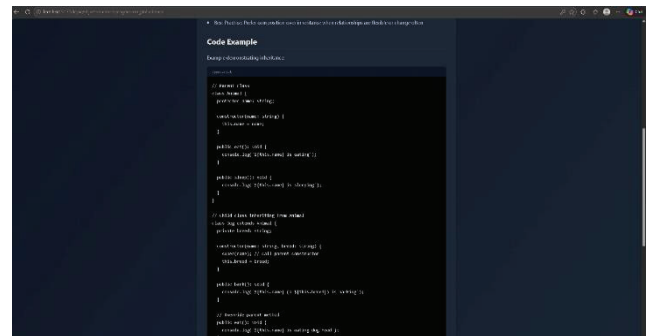


Fig. 5. Interview Solution

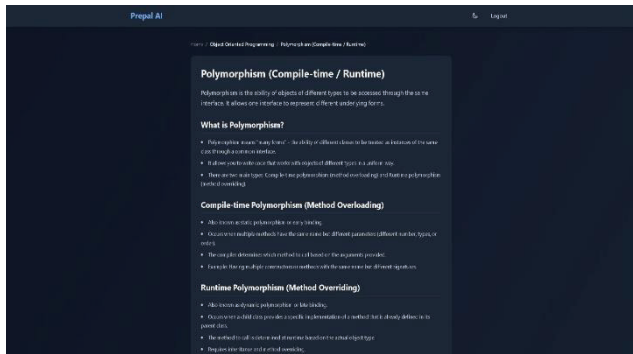


Fig. 6. Core concept learning module

### A. User Interface and System Functionality

The platform provides a centralized dashboard that allows users to access all features seamlessly. The homepage presents an overview of available modules and user progress, enabling easy navigation. The coding interface includes a real-time code editor with problem statements and execution capabilities, ensuring an interactive problem-solving environment.

During mock interview sessions, users can communicate through integrated audio and video channels while simultaneously solving coding problems in a shared workspace. This setup effectively simulates real-world technical interviews, improving both communication and problem-solving skills.

### B. Performance Evaluation

The system evaluates user submissions based on multiple parameters, including correctness, execution time, and memory usage. The integration of a sandboxed execution environment ensures secure and accurate code evaluation. Real-time synchronization enables collaborative coding without delays, demonstrating the efficiency of the underlying communication protocols such as WebRTC and WebSocket.

The performance tracking module aggregates user activity and generates insights into strengths and weaknesses. Users can monitor their progress over time and identify areas requiring improvement. This continuous feedback mechanism enhances learning efficiency and supports structured preparation.

### C. Resume Optimization Effectiveness

The AI-based resume module analyzes user resumes and provides suggestions for improvement based on job-specific

requirements. By identifying missing keywords and improving content structure, the system enhances ATS compatibility. Unlike conventional tools, the integration of performance data allows for more relevant and personalized recommendations.

### D. Discussion

The results indicate that the integration of multiple preparation components into a single platform significantly improves user experience and efficiency. Unlike traditional fragmented approaches, PrepPal AI provides a unified environment that reduces the need to switch between different tools.

The real-time mock interview feature is particularly effective in simulating actual interview conditions, helping users build confidence and improve communication skills. Additionally, the combination of coding evaluation and performance tracking provides a comprehensive assessment of a user's readiness.

However, the system currently relies on predefined problem sets and basic AI-based resume analysis, which can be further enhanced using advanced machine learning models. Future improvements can focus on incorporating adaptive learning algorithms, intelligent mentorship systems, and deeper integration with industry job platforms.

Overall, the results demonstrate that PrepPal AI effectively addresses the challenges of fragmented preparation systems and provides a structured, efficient, and user-centric solution for technical interview readiness.

## VIII. CONCLUSION AND FUTURE WORK

This paper presented PrepPal AI, an integrated AI-driven platform designed to streamline and enhance technical interview preparation for engineering students. The proposed system addresses the limitations of existing fragmented platforms by combining learning, coding practice, mock interviews, and resume optimization within a unified environment. Through real-time collaboration, automated evaluation, and personalized feedback, the platform enables users to systematically improve their technical and communication skills.

The implementation and evaluation of the system demonstrate that an integrated approach significantly improves the efficiency and effectiveness of interview preparation. By providing a centralized dashboard for performance tracking and skill assessment, PrepPal AI allows users to identify weaknesses, monitor progress, and adopt targeted improvement

strategies. The inclusion of peer-to-peer mock interviews further enhances practical exposure, helping users build confidence and perform better in real-world interview scenarios.

Despite its advantages, the current system can be further enhanced to increase its impact and scalability. Future work may include the integration of advanced artificial intelligence techniques for adaptive learning and intelligent mentoring, enabling more personalized and dynamic recommendations. Additionally, the incorporation of gamification elements can improve user engagement and motivation. Integration with job portals and recruitment platforms can also provide direct career opportunities, making the system more industry-oriented.

In conclusion, PrepPal AI offers a comprehensive, efficient, and user-centric solution for technical interview preparation, with the potential to bridge the gap between academic learning and industry requirements.

#### ACKNOWLEDGEMENT

We would like to express our sincere gratitude to the Department of Artificial Intelligence and Data Science, Jawahar Education Society's A. C. Patil College of Engineering, Kharghar, for providing the necessary facilities, guidance, and support to carry out this project. We would also like to thank our project guide Prof. Archana Haral and faculty members for their continuous encouragement, valuable feedback, and technical insights throughout the development of this work. Finally, We extend our appreciation to all students who participated in testing the system and provided constructive feedback that helped improve the quality and usability of the platform.

#### REFERENCES

- [1] [1] A. M. Akshayaa, F. Amber, and P. Arjun, "LeetCode clone," *International Research Journal of Education and Technology*, 2024.
- [2] [2] B. Bell, T. Thomas, S. W. Lee, and C. Brown, "How do software engineering candidates prepare for technical interviews?," *Association for Computing Machinery Conference Proceedings*, 2025.
- [3] [3] S. Anand and A. K. Giri, "Optimizing resume design for ATS compatibility," *International Journal on Smart & Sustainable Intelligent Computing*, 2025.
- [4] [4] V. Pangare, M. Pohekar, P. Potdar, S. Khapekar, V. Deotare, and C. D. Kokane, "Competitive programming contest lister," *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, 2025.
- [5] [5] M. Merkel and J. Dörpinghaus, "AI in software engineering on LeetCode and ChatGPT," *arXiv preprint arXiv:2501.03639*, 2025.
- [6] [6] B. A. Bell, "Understanding the phases of preparation of technical interviews," *ACM SIGSOFT International Symposium on the Foundations of Software Engineering*, 2023.

- [7] [7] A. Kapoor and C. Gardner-McCune, "Introducing a technical interview preparation activity in a data structures and algorithms course," *ACM Conference on Innovation and Technology in Computer Science Education*, 2024.
- [8] [8] D. Zingaro, L. Porter, C. Taylor, and C. Lee, "Identifying student difficulties with basic data structures," *International Computing Education Research Workshop*, 2018.