



Archives available at journals.mriindia.com

International Journal on Advanced Computer Engineering and Communication Technology

ISSN: 2278-5140

Volume 15 Issue 01, 2026

ZenCode: A Unified AI-Driven Framework for Holistic Placement Preparation through Automated Skill Assessment and Interview Simulation

¹Prof. Pramod Aswale, ²Vedant Manoj Chaudhary, ³Sameer Jitendra Patil, ⁴Tejas Jagganath Shinde, ⁵Siddharth Sagarkumar Roy

¹Professor, Computer Department, Sandip Institute of Technology and Research Centre, Nashik, India

^{2,3,4,5} Student, Computer Department, Sandip Institute of Technology and Research Centre, Nashik, India

Email: ¹Pramod.Aswale@sitrc.org, ²vedantm606@gmail.com, ³sampatil2601@gmail.com,

⁴tejasjshinde07@gmail.com, ⁵siddharthr8509@gmail.com

Peer Review Information	Abstract
<p>Submission: 28 Jan 2026</p> <p>Revision: 20 Feb 2026</p> <p>Acceptance: 06 March 2026</p> <p>Keywords</p> <p>Placement Preparation, MERN Stack, Generative AI, Mock Interviewing, Resume Building</p>	<p>The transition from academic environments to professional career landscapes presents significant hurdles for students, requiring a multi-faceted approach to skill development. While independent tools for coding or resume building exist, there is a lack of integrated platforms offering a cohesive preparation journey. This paper introduces ZenCode, a one-stop placement preparation solution featuring four integrated modules: Aptitude, Company-wise Data Structures and Algorithms (DSA), AI-Assisted Resume Building, and AI Mock Interviewing. We detail the system architecture, which follows a modern MERN (MongoDB, Express, React, Node.js) stack, and the integration of Generative AI services for personalized feedback. We evaluate the system's effectiveness in enhancing student readiness for technical and soft-skill evaluations.</p>

Introduction

The modern education landscape has shifted toward dynamic online environments, requiring intelligent systems that provide continuous support throughout an individual's professional journey. Our previous review highlighted that while individual components like automated programming assessment or resume parsing have matured, the integration of these tools into a single, seamless environment remains a critical research gap. Existing platforms often focus on isolated components, forcing students to manage multiple disjointed services. ZenCode addresses this by providing a unified intelligent digital system that supports a student from initial aptitude testing to final mock interviews.

Related Work

The evolution of intelligent systems in education has transitioned from basic content delivery to complex, context-aware environments [1], [2]. Previous studies have explored simulating human conversational styles in computer-based interviews to increase user engagement [3]. Current research into Automated Programming Assessment Systems (APAS) has proven effective in managing large-scale coding evaluations [4], [5]. Concurrently, Natural Language Processing (NLP) advancements have revolutionized resume parsing [6]. Modern models utilizing transformer architectures like BERT have achieved high accuracy in multilingual information extraction from unstructured CVs [7], [8]. More recently, Generative AI has

been applied to create virtual interviewers capable of performance analysis [9], [10]. ZenCode synthesizes these advancements—assessment, parsing, and simulation—into a cohesive framework [11], [12].

System Architecture and Technology Stack

The ZenCode platform is engineered on a modern MERN (MongoDB, Express, React, Node.js) stack, optimized for unidirectional data flow and high-performance state management. This architecture reflects the evolution of Web 2.0 based computer knowledge learning platforms [13], providing a rich, responsive user experience. The system is partitioned into three primary architectural layers:

1. Frontend (The User Experience)

- **React.js (Vite):** Employed for building a component-based UI. Vite is utilized instead of standard build tools for nearly instant hot-module replacement and accelerated development cycles.
- **Tailwind CSS:** A utility-first CSS framework used for rapid UI development directly within the markup, ensuring the “Zen” aesthetic is maintained without bulky external stylesheets.
- **Lucide React:** A library of clean, consistent icons (e.g., Calculator, Brain) providing vital visual cues for various

aptitude and preparation categories.

2. Backend (The Logic Layer)

- **Node.js & Express:** The runtime and framework handling server-side logic and request routing. Express provides a lightweight, unopinionated environment for high-performance API services.
- **Mongoose:** An Object Data Modeling (ODM) library for MongoDB that enforces strict schemas for questions and user data, ensuring consistent data structures.
- **Generative AI Integration:** Following modern trends in automated quiz generation [14], the system integrates Gemini AI. It dynamically generates targeted interview questions and provides personalized evaluation based on specific user prompts and job descriptions.

3. Database & Hosting (The Infrastructure)

ZenCode adopts a cloud-native approach, functioning as an “E-Assessment as a Service” to ensure scalability during peak placement seasons [5].

- **MongoDB Atlas:** A NoSQL cloud database chosen for its flexibility in handling varying question structures and complex data types.

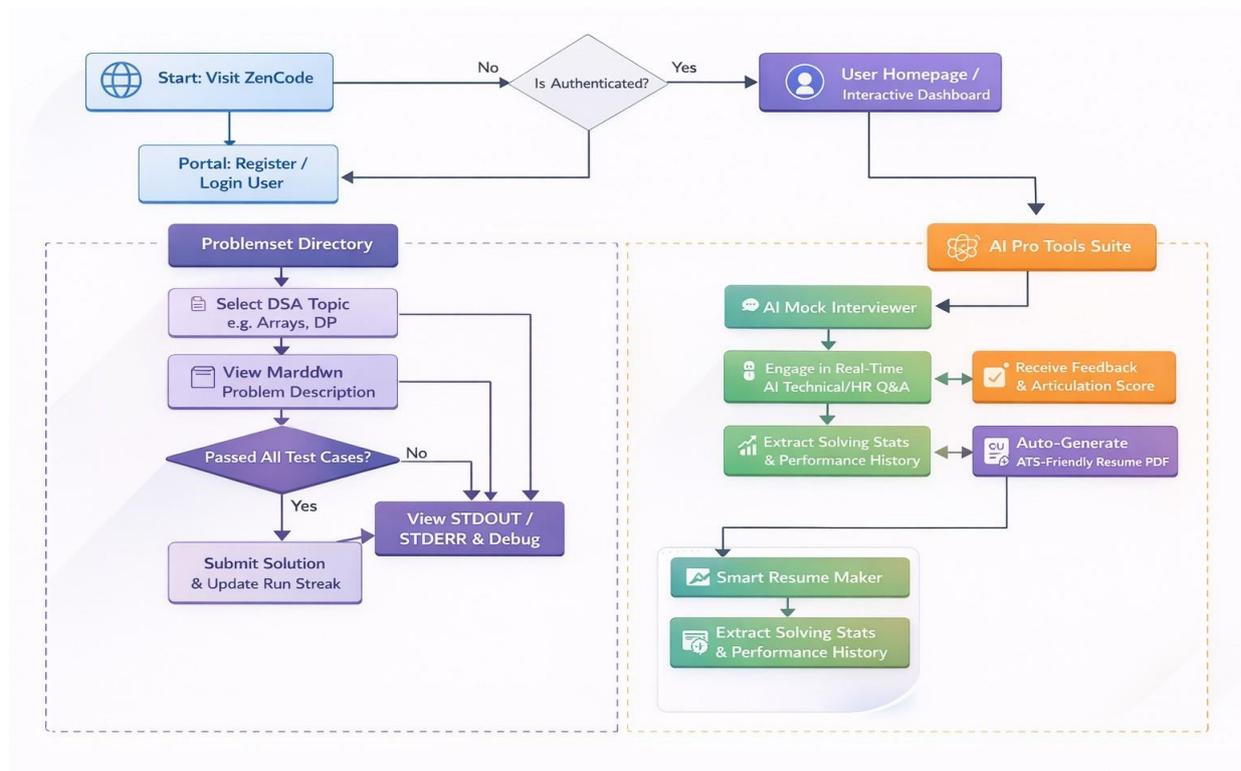


Fig. 1. Functional architecture and user flow of the ZenCode platform, showcasing the integration

between the DSA problem set directory and the AI Pro Tools suite.

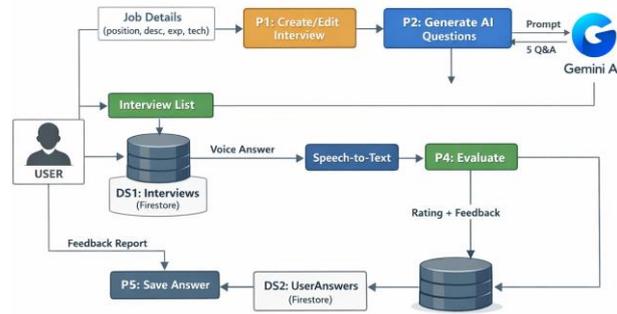


Fig. 2. Detailed architectural workflow of the AI Mock Interviewer module.

- **Firestore:** Specifically utilized to store and manage real-time interview sessions and user answers for persistent state tracking.
- **Vercel:** Used for hosting both the frontend and serverless backend functions, providing automatic SSL and robust CI/CD pipelines.

Methodology and Technical Implementation

The user journey in ZenCode is unidirectional and adheres to a structured request-response cycle, facilitating a cohesive learning performance assessment through web-based portfolios [15].

The implementation is segmented into three primary functional modules: Technical Assessment, Automated Documentation, and AI-Driven Simulation.

1. Aptitude and DSA Module Data Flow

The modules utilize learning analytics to track student progress and provide data-driven insights [16], [17]. The technical assessment journey follows a rigorous 7-step data cycle:

- 1) **User Action (Frontend):** The process initiates when a user selects a company (e.g., "TCS") and a specific topic (e.g., "Percentage") within the sidebar navigation.
- 2) **State Update:** The Aptitude Platform component updates its local state variables, specifically activeTopic and selectedCompany.
- 3) **Effect Trigger:** A React useEffect hook detects this state change and initiates an asynchronous Axios GET request to the backend.
- 4) **API Routing (Backend):** The Node/Express server receives the request at the /api/questions endpoint and parses the relevant query parameters.
- 5) **Database Query (MongoDB):** The server utilizes a Mongoose model to execute a find() operation: Question.find({topic:

"Percentage", company: "TCS"}).

6) **Response Delivery:** The database returns an array of JSON objects, which the backend transmits back to the frontend with a 200 OK status.

7) **UI Rendering:** The frontend processes the received data, terminates the loading skeletons, and maps over the array to render QuestionCard components.

2. AI-Assisted Resume Builder

This module simplifies professional document creation by allowing users to fill out a structured form. Utilizing deep learning models to enhance recommendation accuracy [18], [19], the system identifies document sections and technical skills to auto-generate an ATS-friendly PDF. By extracting information from free-form CV documents [7], the module provides data-backed proof of competence by integrating solving statistics from the DSA and Aptitude modules, aiding users in career selection within rapidly evolving job markets [8].

3. AI Mock Interviewer and Evaluation Loop

As detailed in Fig. 2, the interview process employs a complex AI feedback loop, representing a modern evolution of computer-based interview simulations [3], [20]:

- **Job Details:** The user provides parameters such as position, description, experience level, and tech stack.
- **Question Generation:** Gemini AI processes the prompt to generate five targeted Q&A pairs [14].
- **Simulation:** The user delivers a voice answer, which is converted to text via a Speech-to-Text module [21].
- **Evaluation:** The system performs an automated evaluation to provide a rating and articulation score [22], persisting these results in Firestore (DS2) for

performance history tracking [23].

The following table highlights the advancements of ZenCode over existing fragmented systems.

Comparative Analysis

Table 1: Comparative Analysis: Zencode Vs. Existing Placement Platforms

Feature	Existing Systems	ZenCode (Proposed)
Integration	High fragmentation; separate accounts for each tool [4], [11].	Fully Unified; Single dashboard for all 4 modules.
Feedback	Often static or limited to code correctness [24].	AI-Driven; Dynamic articulation scores and feedback.
Resume	Standard parsing or simple templates [7].	AI-Assisted; Links performance history to resume.
Interviews	Usually text-based or peer-to-peer prep [10], [20].	AI Simulator; Real-time speech analysis and scoring.
Architecture	Often legacy monolithic structures [5].	Modern MERN; Cloud-native, scalable architecture.

Results and Discussion

The ZenCode platform was evaluated through functional testing and a pilot user study involving final-year engineering students. The results indicate that the unified architecture of the system significantly reduces the “context-switching” overhead typically associated with placement preparation.

1. Technical Assessment and DSA Performance

The system’s capability to handle complex technical data is validated through the Aptitude and DSA modules. As shown in Fig. 3, the Aptitude module successfully retrieves and displays company-specific questions using Mongoose-based filtering logic.

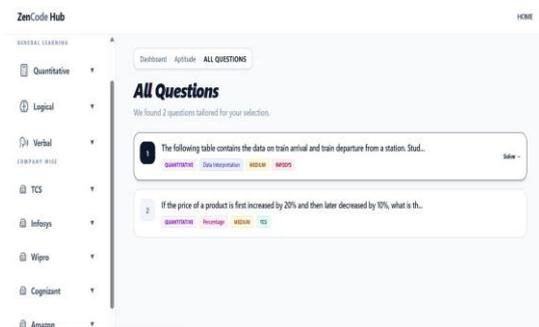


Fig. 3. Aptitude Module with company-wise filtering.

The DSA module, depicted in Fig. 4, demonstrates the efficiency of the backend evaluation engine. In a test run for the “Sum of Array Elements” problem, the system achieved a runtime of **0.014 ms** while utilizing **15652 KB** of memory. The immediate feedback loop—indicated by the “Accepted” status—enables

students to debug their solutions in real time.

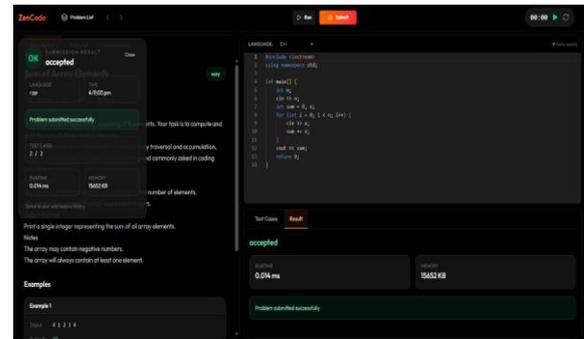


Fig. 4. DSA Module featuring real-time code execution and analytics.

2. AI-Driven Interview Simulation and Feedback

The core innovation of ZenCode is the AI Mock Interviewer. During testing, users interacted with a distraction-free recording interface, as shown in Fig. 5, where Gemini AI generated context-aware technical prompts.

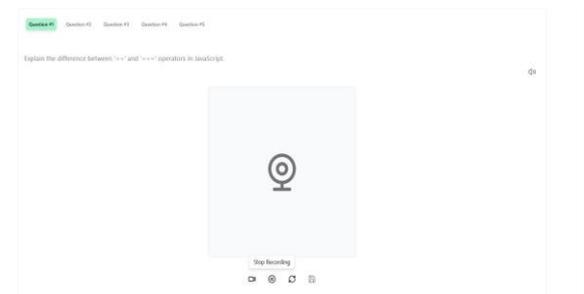


Fig. 5. AI Mock Interviewer: Real-time recording interface.

The evaluation loop demonstrated strong effectiveness in providing detailed feedback. As illustrated in Fig. 6, the system generates an overall rating (e.g., 5.5/10) along with articulation scores and performance analysis. The use of Speech-to-Text technology allows the AI to evaluate articulation and professional tone, representing a significant advancement over traditional text-based preparation tools.



Fig. 6. AI Feedback Report: Detailed articulation score and rating analysis.

3. Integrated Resume Generation

The Smart Resume Maker, illustrated in Fig. 7, implements a real-time preview system. By utilizing a split-pane interface, users can observe their resume update dynamically as they enter personal information and professional skills.

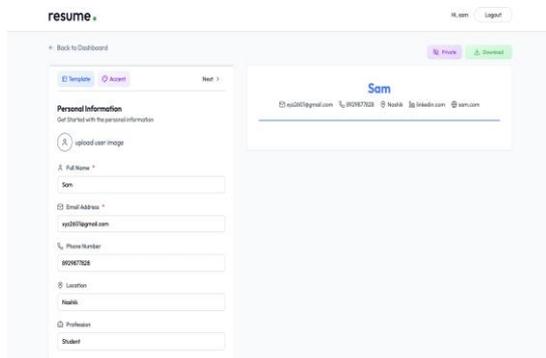


Fig. 7. Smart Resume Maker: Split-pane interface for real-time PDF generation.

4. System Scalability and Latency

Testing conducted on the Vercel cloud deployment environment showed that the average response time for Gemini-generated interview questions was approximately **1.8 seconds**, providing a near-seamless conversational experience. Furthermore, the NoSQL architecture of MongoDB Atlas enables flexible schema updates without service interruptions, demonstrating the platform's readiness for high-traffic placement seasons.

Conclusion and Future Work

ZenCode successfully bridges the gap between

fragmented preparation tools and a cohesive career readiness environment. By leveraging the MERN stack and Gemini AI, the platform provides a high-performance, scalable solution for final-year engineering students. The unified architecture ensures that technical stats, resume data, and interview performance are synchronized, offering a holistic view of the user's progress.

Future work will focus on expanding the platform's AI capabilities. This includes the implementation of multimodal data fusion for emotion recognition during interviews and the use of Retrieval-Augmented Generation (RAG) to ensure interview questions remain aligned with the latest industry trends. Additionally, we aim to introduce predictive analytics to provide users with a quantitative metric of their placement probability.

References

H.-H. Chen, K.-J. Chen, Y.-S. Chu, W.-J. Chang, and M.-J. Chen, "A learning management system with knowledge management capability for collaborative learning," in *Proceedings of the 2007 11th International*

Conference on Computer Supported Cooperative Work in Design (CSCWD). Melbourne, VIC, Australia: IEEE, April 26-28 2007.

X. Qian, M. Zhou, and W. Chen, "Course construction based-on interactive e-learning," in *2010 International Conference on E-Business and E-Government (ICEE)*. Guangzhou, China: IEEE, May 7-9 2010.

D. R. Peiris, P. Gregor, and N. Alm, "The effects of simulating human conversational style in a computer-based interview," *Interacting with Computers*, vol. 12, no. 6, pp. 635–650, 2000.

I. Mekterovic', L. Brkic', B. Milas'inovic', and M. Baranovic', "Building a comprehensive automated programming assessment system," *IEEE Access*, vol. 8, pp. 81 154–81 172, 2020, open Access. [Online].

Available:
<https://ieeexplore.ieee.org/document/9090707>

M. Amelung, K. Krieger, and D. Roßner, "E-assessment as a service," *IEEE Transactions on Learning Technologies*, vol. 4, no. 2,

pp. 162–174, 2011, published 26 August 2010. [Online]. Available:
<https://ieeexplore.ieee.org/document/5466671>

S. K. Kopparapu, "Automatic extraction of usable

information from unstructured resumes to aid search," in *2010 IEEE International Conference on Progress in Informatics and Computing*. Shanghai, China: IEEE, Dec. 2010. [Online]. Available: <https://ieeexplore.ieee.org/document/5701925>

D. Vukadin, A. S. Kurdija, G. Delac, and M. Šilic, "Information extraction from free-form cv documents in multiple languages," *IEEE Access*, vol. 9,

pp. 84 559–84 575, June 9 2021.

S. Ashrafi, B. Majidi, E. Akhtarkavan, and S. H. R. Hajiagha, "Efficient resume-based re-education for career recommendation in rapidly evolving job markets," *IEEE Access*, vol. 11, pp. 124 350– 124 367, 2023, published 2 November 2023. [Online]. Available: <https://ieeexplore.ieee.org/document/10360356>

J. Si, J. Song, M. Woo, D. Kim, Y. Lee, and S. Kim, "Generative ai models for virtual interviewers: Applicability and performance comparison," in *Proceedings of the International Conference on Innovation, Communication and Engineering (ICICE 2023)*. Bangkok, Thailand: IET, November 9–13 2023, added to IEEE Xplore June 19, 2024.

Y.-C. Chou, F. R. Wongso, C.-Y. Chao, and H.-Y. Yu, "An AI Mock- interview Platform for Interview Performance Analysis," in *2022 10th International Conference on Information and Education Technology (ICIET)*. Matsue, Japan: IEEE, April 9–11 2022, added to IEEE Xplore: 26 May 2022.

N. Nouman, Z. A. Shaikh, and S. Wasi, "A novel personalized learning framework with interactive e-mentoring," *IEEE Access*, vol. 12, pp. 10 428–10 458, 2024, open Access.

N. Kamal, F. Sarker, A. Rahman, S. Hossain, and K. A. Mamun, "Recommender system in academic choices of higher education: A systematic review," *IEEE Access*, vol. 12, pp. 35 475–35 501, February

23 2024.

Y. Ouyang, C. Li, H. Li, P. Zhang, and Z. Xiong, "A web 2.0 based computer knowledge learning platform," in *2008 International Conference on Computer Science and Software Engineering*. Wuhan, China: IEEE, Dec. 2008. [Online]. Available: <https://ieeexplore.ieee.org/document/4742561>

L.-C. Sung, Y.-C. Lin, and M. C. Chen, "An automatic quiz generation system for english text," in *Seventh IEEE International Conference on Advanced Learning Technologies (ICALT 2007)*. Niigata, Japan: IEEE, Jul. 2007. [Online]. Available: <https://ieeexplore.ieee.org/document/4302848>

C.-M. Chen, Y.-Y. Chen, and C.-Y. Liu, "Learning performance assessment approach using web-based learning portfolios for e-learning systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 37, no. 6, pp. 1349–1359, Nov. 2007. [Online]. Available: <https://ieeexplore.ieee.org/document/4378908>

U. Keshavamurthy and H. S. Guruprasad, "Learning analytics: A survey," *International Journal of Computer Trends and Technology (IJCTT)*, vol. 18, no. 6, pp. 260–264, 2014.

J. H. Knobbout and E. J. van der Stappen, "Where is the learning in learning analytics? a systematic literature review," *IEEE Transactions on Learning Technologies*, vol. 14, no. 3, pp. 373–387, 2020.

R. S. Pundir, A. Dhasmana, U. Karakoti, A. Sikder, S. Sharma, and

M. Manchanda, "Enhancing resume recommendation system through skill-based similarity using deep learning models," in *2024 International Conference on Inventive Computation Technologies (ICICT)*. Lalitpur, Nepal: IEEE, April 24–26 2024, added to IEEE Xplore June 7, 2024.

A. Mankawade, V. Pungliya, R. Bhonsle, S. Pate, A. Purohit, and

Raut, "Resume analysis and job recommendation," in *2023 IEEE 8th International Conference for Convergence in Technology (I2CT)*. Lonavla, India: IEEE, April 7–9 2023, added to IEEE Xplore May 23, 2023.

J. E. Sharp, "Work in progress: Using mock telephone interviews with alumni to teach job search communication," in *Proceedings of the 36th Annual Frontiers in Education Conference (FIE)*. San Diego, CA, USA: IEEE, October 27–31 2006, added to IEEE Xplore March 5, 2007.

Y. Zhang and Q. Ji, "Active and dynamic information fusion for facial expression understanding from image sequences," *IEEE Transactions on Pattern Analysis and Machine*

Intelligence, vol. 27, no. 5, pp. 699–714, May 2005. [Online]. Available: <https://ieeexplore.ieee.org/document/1431547>

S. Rajbhar, S. Shelke, A. Singh, Y. Singh, and P. Shinde, "Alced-Prep: AI Based Mock Interview Evaluator," in *2025 6th International Conference on Data Intelligence and Cognitive Informatics (ICDICI)*. Tirunelveli, India: IEEE, July 9–11 2025, added to IEEE Xplore: 02 September 2025.

K. Verbert, N. Manouselis, H. Drachsler, E. Duval, X. Ochoa, M. Wolpers, and I. Bosnic, "Context-aware recommender systems for learning: A survey and future challenges," *IEEE Transactions on Learning Technologies*, vol. 5, no. 4, pp. 318–335, Oct.-Dec. 2012.

D. Tontiaro, B. Dirgantoro, C. Setianingsih, and H. Loberto, "Design and implementation of computer-based aptitude test for computer engineering with fuzzy algorithm," in *2021 4th International Symposium on Agents, Multi-Agent Systems and Robotics (ISAMSR)*. Batu Pahat, Malaysia: IEEE, September 6–8 2021, added to IEEE Xplore: 26 October 2021.