IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 9, April 2026
Impact Factor: 8.2

PrepWise - Intelligence Suite for Adaptive Conversational Interview Simulation

Prof. Snehal Rewatkar[1], Ms. Bhavika Kailas Ahire[2],
Ms. Vedangi Pankaj Ghode[3], Ms. Akansha Sharad Hatmode[4]
Professor, Department of Computer Science & Engineering[1]
Students, Department of Computer Science & Engineering[2-4]
Tulsiramji Gaikwad Patil College of Engineering and Technology, Nagpur, Maharashtra, India
snehal.cse@tgpcet.com[1], bhavikaahire024@gmail.com[2]
vedangighode47@gmail.com[3], akankshahatmode6@gmail.com[4]

Abstract: *PrepWise is an AI-powered web platform that changes how people prepare for mock interviews by making the experience more interactive, conversation-like, and based on real data. It uses a modern web technology setup where Next.js handles both the front-end and back-end parts, Firebase takes care of user login and data storage, Tailwind CSS and shadcn/ui create a clean, responsive design, Vapi AI manages voice interactions, Google Gemini creates interview questions and evaluates answers, and Zod ensures that all input is correct and secure. This combination of tools helps PrepWise offer something better than traditional interview prep tools, which usually rely on fixed sets of questions and general feedback. Instead, PrepWise adapts to what users choose, like the job role, experience level, tech stack, and type of interview, creating a more realistic and tailored practice environment. After the interview, the platform gives detailed analysis on communication skills, technical knowledge, problem-solving, how well the user fits the role, and confidence levels, helping users understand their strengths and areas to improve more clearly. Overall, PrepWise shows how AI and advanced web development can work together to create a powerful, scalable, and user-focused system for preparing for interviews and improving professional skills.*

Keywords: Artificial Intelligence, Mock Interview Simulation, Conversational AI, Next.js, Firebase, Google Gemini, Vapi AI, shadcn/ui, Tailwind CSS, Zod, Web Development, Interview Preparation, Adaptive Learning, Transcript Analysis

I. INTRODUCTION

Artificial intelligence is playing a bigger role in shaping educational tools, career growth resources, and the way people interact with computers. It helps create experiences that adjust to what users need, how they behave, and how they perform. However, in interview practice, many traditional platforms still use fixed sets of questions, manually created prompts, and general feedback that don't really mirror the stress and real-time nature of actual interviews. Because of this, learners often get used to the questions but don't really gain confidence, communication skills, or the ability to think on their feet.

PrepWise was created to address these issues by offering an intelligent conversation-based interview simulator.

Unlike platforms that only use text, PrepWise uses an AI voice agent to conduct interviews in a spoken, interactive way. The questions are created on the fly based on the user's chosen role, experience level, and the technologies they know. This lets the system tailor the interview experience to each person. After the conversation, the system creates a detailed transcript and provides structured feedback. This helps users not only understand what they said but also how well they conveyed their ideas.

The project brings together artificial intelligence and web development, combining powerful AI services with a modern, professional web application.

DOI: 10.48175/IJARSCT-33214

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 9, April 2026 Impact Factor: 8.2

Firebase is used for user authentication and secure management, while Next.js offers a single framework for both the front and back ends. Vapi agents manage the voice interaction, and the feedback is shown through clear and user-friendly interview and dashboard pages. This mix makes PrepWise more than just a simple interview practice tool; it's a fully built, scalable system that adapts to assess and improve skills.

II. LITERATURE SURVEY

The study of AI-powered mock interview tools has developed from basic rule-based conversation tools to more advanced, voice-activated, and multi-mode platforms.

Recent reviews show four main areas of focus: rule-based systems, AI-powered large language model (LLM) systems, avatar-based immersive systems, and gamified training tools. There's a growing trend towards combining both technical and behavioral assessments in one system. PrepWise is part of this newer generation because it combines question creation, voice interaction, transcript analysis, and feedback dashboards, rather than treating each as a separate feature.

2016–2019: Early Automated Assessment

Earlier research mainly focused on automated candidate evaluation and speech-based scoring, not on realistic interactive interviews.

Studies from this time examined AI in assessing candidates, speech patterns, automated scoring, and the use of machine learning models like XGBoost for analyzing spoken responses and oral skills. This period laid the groundwork for future interview simulators by showing that speech and language features could be used to gauge communication quality, confidence, and accuracy.

2020–2022: Moving Toward Adaptive and Multimodal Systems

Research during this time started shifting from static scoring to more dynamic interview training environments.

Work on voice analysis and facial expression detection showed that interview performance could be measured using both verbal and non-verbal clues. Research on adaptive questioning also showed that interview prompts could be tailored based on a candidate's performance or context. The 2022 systematic review also pointed out that many systems still provided delayed feedback, had limited realism, or used separate modules for technical and behavioral assessments, leaving room for more integrated solutions.

2023: Conversational and Context-Aware Interview Tools

By 2023, research began focusing more on realistic conversations and context-aware question generation.

Papers on adaptive question generation and AI-powered mock interviews showed that large language models could create more relevant questions and enhance user engagement compared to fixed question sets. However, many of these systems were still limited by offline processing, weak context retention, or a lack of genuine voice-based interaction, which reduced their realism when compared to live interviews.

2024: Personalized Feedback and Practical Usability

In 2024, the focus of research and industry publications shifted toward usability, personalized practice, and instant feedback.

Modern interview simulators now provide immediate feedback on clarity, tone, structure, confidence, and subject knowledge, often adjusting difficulty based on user performance. This period also emphasized the need for browser-accessible, lightweight systems that avoid the cost and hardware requirements of VR-only simulators.

2025: Integrated AI Voice Interview Agents

The most relevant work for PrepWise emerged in 2025, with systems that combined real-time voice interaction, adaptive question generation, and automated scoring.

One paper introduced an AI Voice Interview Agent using Vapi SDK, Open LLM for question generation, and XGBoost for speech, confidence, and organization scoring. The 2025 systematic review also stated that the field still suffered from fragmentation and called for unified frameworks that integrate technical and behavioral training, emotionally aware feedback, and standardized evaluation. PrepWise directly supports this direction by using Gemini for questions

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 9, April 2026 Impact Factor: 8.2

and feedback, Vapi for voice interaction, Firebase for storage and login, and TypeScript/Zod for a safer implementation.

Key Themes from the Literature

The literature highlights five main themes.

First, realism is crucial: systems are more effective when they simulate spoken, interactive interviews rather than just text-based conversations. Second, adaptability is important: users benefit when the difficulty and topic of questions change based on their responses. Third, feedback should be immediate and structured, covering both how well they communicate and the quality of their answers. Fourth, multimodal and immersive systems enhance user engagement, but hardware and accessibility challenges limit widespread use. Fifth, integrated systems are preferred because they reduce complexity and make it easier to maintain AI-based applications.

Relevance to PrepWise

PrepWise reflects the latest direction in the literature: a unified, browser-friendly, voice-driven interview platform with AI-powered adaptability and structured assessment.

Compared to earlier rule-based or text-only systems, it offers a more realistic interview experience and a clearer feedback cycle for users preparing for technical and behavioral roles. Its use of Gemini, Vapi, Firebase, Next.js, shadcn/ui, Tailwind CSS, and Zod fits the literature's emphasis on modularity, personalization, and scalable deployment. In a research paper, this positioning makes PrepWise a valuable contribution to the growing field of AI-driven interview training.

III. METHODOLOGY OF THE SYSTEM

The approach used by PrepWise is based on a modular, service-oriented structure. Instead of handling the whole interview as a single process, it breaks it into several controlled stages. This design makes the system easier to manage, allows for easier expansion, and enables each part of the interview to be improved separately. The full process includes user login, setting up the interview, creating questions with an AI, delivering the interview through voice, recording the conversation, evaluating the performance, and presenting the feedback.

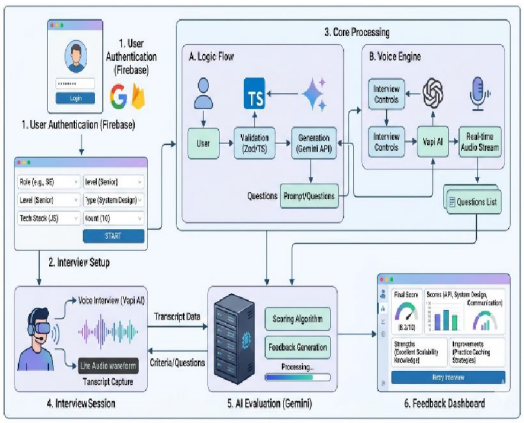

Fig- 1: Methodology

The process starts with user authentication and managing sessions. A user logs in using Firebase, which ensures secure access and links each interview attempt to a user profile. Once logged in, the user sets up an interview by providing details like the desired job role, experience level, type of interview, technology stack, and the number of questions. These details are important because they define the context of the practice session and help the system create questions tailored to the user's goals rather than using standard, generic questions.

After the interview setup is complete, the system creates a prompt for the AI to generate questions.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 9, April 2026
Impact Factor: 8.2

This prompt includes all the necessary information so the AI can create questions that match the selected role and difficulty. The questions are formatted carefully to ensure they can be spoken naturally by a voice assistant. This involves avoiding symbols, special characters, and punctuation that could cause confusion or issues when the questions are read aloud. The focus here is not just on making the questions relevant, but also on ensuring they work well with voice interaction and are easy to understand when spoken.

Once the questions are ready, Vapi AI serves as the conversational interface of the platform. It presents the questions through a voice system, making the experience more realistic because the user responds by speaking, similar to a real interview. This interactive flow helps create a natural rhythm of asking and answering questions, and follow-up interactions. As a result, the platform better simulates the actual conditions of a real interview.

During the interview, the conversation is recorded as a transcript. This transcript becomes the main source for analyzing the user's performance. Instead of just checking if the right answers were given, the system reviews the entire conversation to understand how well the user communicated their ideas, how well they managed the interview context, and how organized their responses were. This approach is important because real interview performance depends heavily on factors like clarity, structure, confidence, and consistency, not just the accuracy of the answers.

The next step is evaluation. The transcript is sent to a Gemini-based prompt that asks the AI to score the user across five categories: communication skills, technical knowledge, problem-solving ability, cultural or role fit, and confidence. These categories were selected because they align with common standards used in professional interviews. The AI is also asked to provide strengths, areas for improvement, and a summary of the overall assessment. This turns the transcript into structured feedback that is easier for users to understand and use to improve their performance.

The methodology also includes displaying and storing the final interview results.

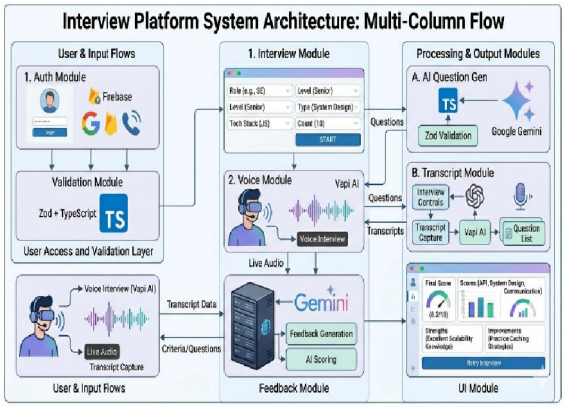

Fig- 2: System Architecture

The user can review their overall score, performance in each category, and the qualitative feedback on a dashboard and feedback page. This presentation is important because it turns the AI output into a useful learning tool. Feedback isn't just shown once; it becomes part of a continuous learning loop that helps users track their progress over multiple interviews. This allows for repeated practice and improvement with each attempt.

A key strength of the methodology is its focus on reliability and safe input handling.

The system uses Zod for input validation and TypeScript for strict type checking, which helps prevent incorrect input from reaching the AI pipeline. This is especially important in systems that rely on multiple external services, as errors can occur at various points, including user input, prompt creation, voice interaction, and response processing. Validation reduces the risk of errors like malformed prompts, invalid interview parameters, and inconsistent data. It also improves the predictability of the system when it's used in real-world situations.

From a software engineering perspective, the approach follows a clear separation of responsibilities. Each part of the system—like authentication, question generation, voice interaction, evaluation, and feedback—is treated as a separate module. This modularity makes the project easier to debug, test, and expand. For example, improving the question

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 9, April 2026 Impact Factor: 8.2

generation system doesn't affect the feedback engine, and changing the user interface doesn't require changes to the AI prompts. This separation is especially useful in AI-based systems, where ongoing improvements are often needed to enhance quality and usability.

Overall, the methodology of PrepWise combines AI coordination, voice interaction, structured validation, and effective feedback into a single learning workflow.

It is practical because it mirrors the real interview process while remaining flexible enough to adapt to different fields and user needs. The end result is a system that is not only built well from a technical standpoint but also has real educational value, helping users practice, reflect, and improve through repeated simulated interviews.

A. Workflow

The operational process of PrepWise follows a step-by-step cycle that uses AI technology.

The user starts by signing up or logging in using Firebase authentication.

Once logged in, the user sets up an interview by providing details like the job role, experience level, technical skills, type of interview, and the number of questions they expect.

The app then sends this information to a Next.js route that uses Google Gemini to create a prompt for generating interview questions.

The generated questions are formatted into a list that works with voice, and then sent to Vapi AI to manage the conversation during the mock interview.

The user takes part in the interview by speaking, and their conversation is recorded as a transcript for later review.

This transcript is then used to generate feedback.

Google Gemini analyzes the user's performance and provides scores in different categories, along with comments, strengths, areas to improve, a total score, and a final evaluation.

The app saves or retrieves the interview results using its Firebase-based data system and displays them on the dashboard and feedback pages.

B. Algorithm (Pseudocode)

Algorithm: PrepWise_Interview_Simulation

Input: userProfile, role, level, interviewType, techStack, questionCount

Output: structuredFeedback, scoreBreakdown, interviewRecord

text

```
Begin
   Authenticate user through Firebase
   If authentication fails then
      return access_error
   EndIf

   Validate input fields using schema rules
   If validation fails then
      return validation_error
   EndIf

   prompt_q <- buildQuestionPrompt(role, level, interviewType, techStack, questionCount)
   questions <- Gemini.generate(prompt_q)
   questions<- sanitizeForVoiceAgent(questions)

   session<- Vapi.startVoiceInterview(questions, userProfile)
```

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 9, April 2026
Impact Factor: 8.2

```
transcript <- captureTranscript(session)

prompt_f<- buildFeedbackPrompt(transcript)
structuredFeedback<- Gemini.evaluate(prompt_f)

scoreBreakdown<- extractCategoryScores(structuredFeedback)
interviewRecord<- persistInterview(userProfile, questions, transcript, structuredFeedback)

Display feedback dashboard and retake options
return structuredFeedback, scoreBreakdown, interviewRecord
End
```

C. Flowchart

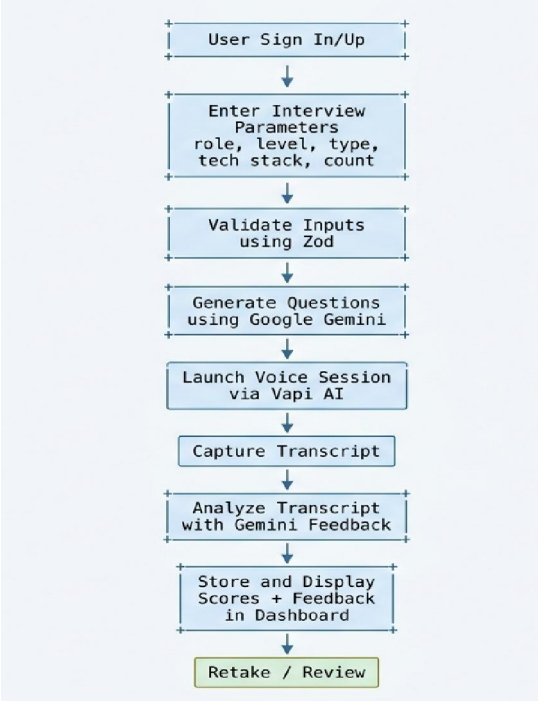

Fig-3 : Flowchart

IV. IMPLEMENTATION

PrepWise is built as a modern application using Next.js and React, with a focus on TypeScript. It supports both client-side and server-side functionality. The main technology stack includes Next.js 15.2.2, React 19, Firebase 11.4.0, Firebase Admin 13.2.0, the Vapi web SDK, Google AI SDK packages, Zod 3.24.2, and React Hook Form. These tools work together to provide authentication, AI coordination, validation, and form handling all in one place.

The user interface is created with Tailwind CSS and shadcn/ui.

The setup uses a New York style theme, includes Lucide icons, supports server components, and uses CSS variables for theming through a global stylesheet. This gives the project a clean and responsive look, which is especially important for a voice-based interview platform where clear controls, readable feedback, and mobile-friendly layouts are key to usability.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 9, April 2026 Impact Factor: 8.2

On the backend, the app uses Next.js route handlers and action-based logic to separate question generation from interview evaluation.

One prompt is used to create interview questions based on the selected role, experience level, technology stack, interview type, and number of questions. Another prompt is used to analyze the transcript and provide feedback grouped by category, along with strengths and areas for improvement. Keeping these prompts, separate is a smart design choice since generating questions and evaluating answers are different tasks that need different instructions.

The project also shows strong engineering practices through strict TypeScript settings, module resolution using a bundler, and path aliases that make modular development easier.

Zod handles schema validation, React Hook Form manages structured input, and the overall structure makes the system easier to maintain, debug, and expand. These choices are important because AI-based applications often face issues not just from model performance, but also from weak validation, unclear component boundaries, or inconsistent data flow.

V. RESULTS AND ANALYSIS

PrepWise shows that an adaptive conversational interview simulator can bring together authentication, AI-generated questions, voice interaction, transcript analysis, and structured feedback into a single, smooth process. According to the repository documentation, the platform successfully covers the main functions expected from such a system: secure user access, setting up interviews, voice-based questioning, capturing conversation transcripts, AI-driven scoring, and reviewing interviews after they're done. This means the project is not just an idea—it's a real, working example of an end-to-end intelligent interview prep environment.

One big outcome of the system is the detailed feedback it provides. Instead of just giving a simple pass or fail result, PrepWise gives an overall score, a breakdown by category, a final assessment, strengths, and areas for improvement. This is important because interview performance is not just one thing—it can include many different skills. For example, a candidate might know a lot about the subject but not be very good at communicating, or they might sound confident but not be good at solving problems in a structured way. By separating these different aspects, the platform helps users understand where they are doing well and where they need to focus their practice.

The results also highlight the benefits of combining several technologies into one system.

Next.js is used for the front and back ends, Firebase handles user authentication and data storage, Vapi allows for natural voice interaction, and Gemini is used both to generate questions and evaluate the transcripts. The strength of this integration is that each part plays a different role in the interview experience while all working together as a single workflow. This makes the platform feel more like a real conversation than a regular chatbot or quiz app.

From a practical standpoint, the system is valuable because it supports repeated practice.

After each interview, users can look at their feedback and retake the interview to work on specific weaknesses. This creates a learning cycle where feedback leads to practice, and practice leads to better performance. This cycle is especially useful in preparing for interviews, where improvement often comes from repeating, reflecting, and gradually building confidence.

However, there are some technical considerations. The scoring based on transcripts depends a lot on the quality of speech recognition and how complete the conversation is captured. If the transcript is inaccurate, the feedback might not be reliable. The quality of voice interaction can also vary depending on internet stability, microphone quality, and how clearly the user speaks. Additionally, AI evaluation is sensitive to the way prompts are designed, so small changes in instructions can affect the scoring style or the balance of feedback.

Another key discussion point is evaluation consistency.

Since the system uses generative AI for both creating questions and providing feedback, the results may not always be the same each time. This variability can be helpful for adaptive learning, but it also means that more research is needed to check the consistency, fairness, and repeatability of the scores. A stronger benchmarking system would help determine how reliably the system measures communication skills, technical knowledge, and problem-solving ability across different sessions.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 9, April 2026
Impact Factor: 8.2

Overall, the results show that PrepWise is both technically solid and educationally meaningful. It goes beyond just automation and creates a realistic practice environment that includes interaction, assessment, and reflection. The discussion also points out that the project could benefit from deeper validation, wider testing, and more objective performance benchmarks, which would make it even more valuable as both a product and a research contribution.

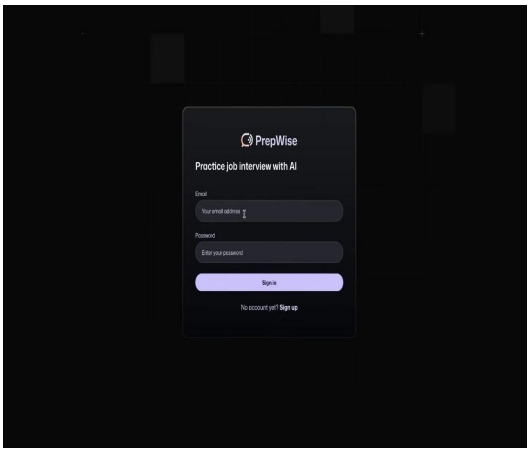

Fig-4 : Sign Up page

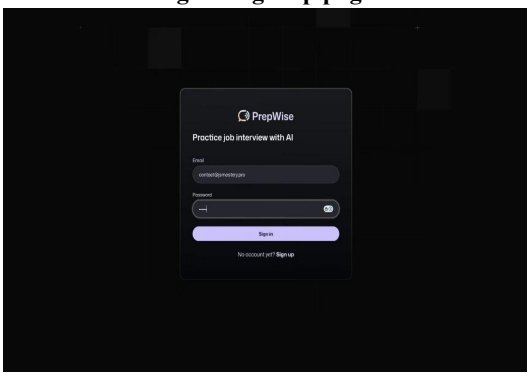

Fig- 5 : Sign In page

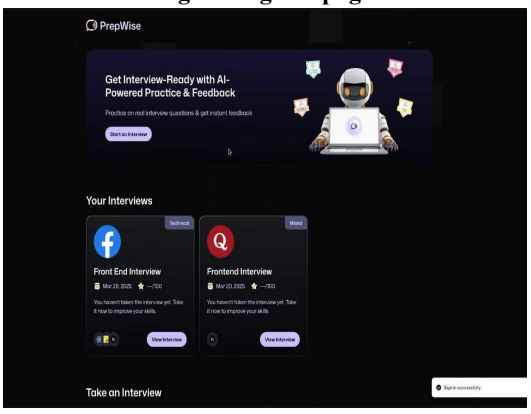

Fig- 6 : User page

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 9, April 2026
Impact Factor: 8.2

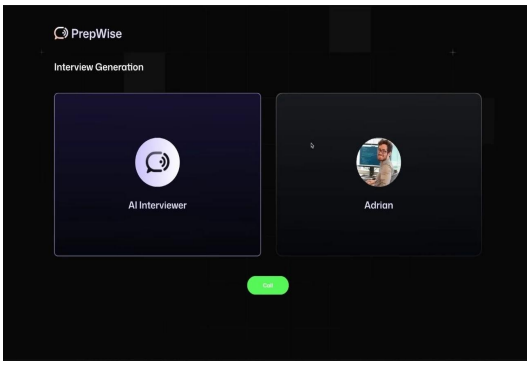

Fig- 7 : Interview Interface

VI. FUTURE SCOPE

PrepWise has a lot of potential to grow in terms of technical features and research importance.

One key improvement is adaptive difficulty control, where the system changes the level of questions based on how well the user does in previous responses. For example, if someone answers confidently and correctly,the next question can be more challenging or more situational. If the response is not strong, the system can simplify the next question or focus on basic concepts. This would make the interview experience feel smarter and more like a real conversation, because actual interviewers naturally adjust the difficulty based on how well the candidate is doing.

Another useful direction is multimodal scoring.

Right now, evaluation is mostly based on what is said, but in the future, it could also look at speech patterns like how long someone pauses, how fast they speak, how hesitant they are, their tone, and how confident they sound. These clues can show important interview behavior that text alone can't capture. By combining both what is said and how it's said, PrepWise could provide a more accurate and complete assessment, especially for skills like communication and confidence.

The platform can also be expanded by specializing in different fields.

Instead of using a single general interview model, separate interview packs could be made for specific roles like data science, cloud engineering, front-end development, back-end development, product management, and HR screening. Each pack could use unique prompts, scoring rules, and feedback templates. This would make the system more useful for people aiming for specific careers and increase its value for both students and working professionals. Another possible extension is using the platform in educational and enterprise settings.

PrepWise could be adapted for universities, training institutes, and placement cells by adding tools like recruiter dashboards, analysis of groups of students, and tracking performance over time. These institutions could use the platform to monitor how ready students are for interviews, find common areas where they struggle, and track improvements. Features like exporting rubrics, keeping performance records, and comparing groups would turn the tool from a personal prep system into a broader platform for analyzing employability.

From a product development angle, future versions could include interview replay, hint-based coaching, and real-time help.

Interview replay would let users review their own answers and understand where they hesitated or were unclear. Hint-based coaching could offer gentle tips before or after a response, while real-time help could support beginners by giving them gentle guidance during the interview. These features would make the platform more adaptable to different levels of experience.

Another important area for future work is testing and refining the AI models.

Since AI feedback can change depending on how the questions are phrased and how the model behaves, future research should check how consistent and reliable the scoring is over time. Comparing PrepWise against other tools like static

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 9, April 2026
Impact Factor: 8.2

mock interviews, text-only bots, and human evaluators would help show how effective it really is. This type of testing would strengthen the academic standing of the project and make its evaluation more scientifically sound.

Future work should also focus on studying how easy the platform is to use and how much it helps users learn.

Research could check if users feel more confident after using it, if it improves real interview performance, and if the feedback is seen as realistic and helpful. These studies would connect the system's technical design to actual educational results. In a research context, this is especially important because it shows that the project is not just innovative but also truly beneficial.

VII. CONCLUSIONS

PrepWise shows that modern AI services can be smoothly added into a full-stack web app to create an interview simulation platform that's both flexible and useful. The main strength of PrepWise is how it brings together dynamic question creation, real-time voice interaction, transcript-based evaluation, cloud-based login, and reusable parts of the user interface into one smooth system. This makes the experience feel more like a real interview than just a simple question-and-answer chatbot, because the user is placed in a conversation that closely mimics an actual interview setting.

The project also highlights the value of using AI not only to create content but also for structured evaluation and personalized feedback.

After the interview, PrepWise analyzes the conversation to give meaningful insights into communication skills, technical knowledge, problem-solving abilities, role fit, and confidence levels. This helps users practice repeatedly, identify specific areas where they need improvement, and track their progress through multiple attempts. This approach supports both learning and self-evaluation.

From a software engineering point of view, PrepWise is a great example of how breaking a system into modular parts can improve reliability, maintainability, and scalability.

Each part—like authentication, validation, interview generation, voice interaction, and feedback delivery—is handled in separate stages, which lowers complexity and makes the system easier to build on in the future. Using current web technologies and AI services makes the project highly relevant, especially in areas where intelligent automation and interactive learning are becoming more important.

Overall, PrepWise is more than just an interview practice tool. It's an intelligent assessment system that shows how AI, web development, and user-focused design can come together to help with career preparation in a meaningful way. Its structure and features make it not only a viable product idea but also a strong basis for academic research and future work in AI-assisted learning systems.

VIII. ACKNOWLEDGEMENT

This research paper is prepared on the basis of the design, implementation, and documentation evidence available in the PrepWise project materials. The repository files provided a clear understanding of the system architecture, technology stack, feature set, prompt design, and AI workflow, which made it possible to write a detailed academic-style description of the project. Special attention was given to the project README and configuration files, as they reveal how the platform combines authentication, voice interaction, question generation, and feedback analysis into a unified interview preparation system.

The documentation also indicates a tutorial-driven development context associated with JavaScript Mastery, which appears to have guided the project's conceptual framing and implementation direction. Such educational resources play an important role in helping developers understand how to integrate modern AI services into full-stack web applications and how to structure a project for practical use.

Appreciation is also extended to the open-source ecosystem that makes projects like PrepWise possible. Technologies such as Next.js, Firebase, Vapi AI, Google Gemini, Tailwind CSS, shadcn/ui, React, TypeScript, and Zod together

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 9, April 2026
Impact Factor: 8.2

provide the foundation for building scalable, interactive, and reliable AI-powered applications. Their combined support enables rapid development of intelligent web systems that are both technically robust and user-centered.

REFERENCES

[1]. Vercel. "Next.js Docs." https://nextjs.org/docs
[2]. Firebase. "Firebase Documentation / Web setup." https://firebase.google.com/docs/web/setup#access-firebase
[3]. Vapi. "Introduction to Vapi." https://docs.vapi.ai/quickstart/introduction
[4]. GitHub. "Vapi Platform Documentation Repository." https://github.com/VapiAI/docs/blob/main/README.md
[5]. React. "React Documentation." https://react.dev/
[6]. Tailwind Labs. "Tailwind CSS Documentation." https://tailwindcss.com/docs
[7]. shadcn/ui. "shadcn/ui." https://ui.shadcn.com/
[8]. Google. "Gemini API Documentation." https://ai.google.dev/gemini-api/docs
[9]. TypeScript. "TypeScript Handbook." https://www.typescriptlang.org/docs/
[10]. React Hook Form. "Documentation." https://react-hook-form.com/
[11]. Lucide. "Lucide React." https://lucide.dev/
[12]. Day.js. "Day.js Documentation." https://day.js.org/
[13]. Next.js India Community. "Next.js Documentation Mirror." https://en.nextjs.im/docs
[14]. LMU Computer Science Notes. "Firebase." https://cs.lmu.edu/~ray/notes/firebase/
[15]. Shen, X., et al. "MetIVA: XR-based interactive visualization platform for meteorological data." ScienceDirect, 2025. https://www.sciencedirect.com/science/article/pii/S1877050925003822