



INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

INTERVIEWSYNC PRO-A COMPREHENSIVE INTERVIEW PLATFORM FOR STREAMLINED TECHNICAL RECRUITMENT

¹Ikshit Madan, ²Divyang Dheer, ³Mohit Uniyal, ⁴Pradyumn Ojha, ⁴Pratibha Sharma

¹Student, ²Student, ³Student, ⁴Student, ⁵Assistant Professor

¹Department of Computer Science,

¹Bhagwan Parshuram Institute of Technology, Rohini New Delhi-110089, India

Abstract: A Comprehensive Interview Platform for Streamlined Technical Recruitment

This paper presents a novel interview platform designed to enhance the technical recruitment process. Existing video conferencing tools used for interviews lack functionalities crucial for effective candidate evaluation. This platform addresses this gap by integrating a suite of features specifically tailored for software development interviews.

The core functionality includes a collaborative code editor supporting multiple programming languages. This allows candidates to showcase their coding skills in familiar environments and facilitates real-time interaction for code review, debugging, and problem-solving with interviewers.

Furthermore, the platform incorporates an interactive whiteboard powered by Socket.IO. This enables real-time collaboration where both parties can visually represent ideas and concepts, fostering clearer communication and a more engaging interview experience.

To provide additional flexibility in the interview structure, a chatbot functionality is implemented. This chatbot can be programmed to deliver pre-defined questions, handle basic queries, or even administer automated coding challenges. This frees up the interviewer's time for in-depth technical discussions and candidate assessment.

Finally, seamless face-to-face interaction is facilitated through WebRTC technology, enabling video and audio communication for a more human interview experience.

Index Terms – Interview, Technical Recruitment, Video Conferencing, MERN, webRTC, Chatbox, Facial Recognition, ReactJS, MongoDB, Node.js, Express.js.

1. Introduction

The conventional approach to conducting technical interviews is changing significantly in today's employment environment. Face-to-face interviews have long been the ritual of choice for assessing a candidate's technical skill, problem-solving aptitude, and cultural fit within an organization. However, the shortcomings of this conventional strategy are becoming more obvious as the world economy changes and the tech sector grows at an unprecedented rate.

Historically, technical interviews have been characterized by their in-person nature, requiring candidates to attend interviews at the premises of the hiring company physically. While effective in many respects, this process presents numerous challenges and inefficiencies. Geographical constraints often restrict the pool of available candidates, forcing companies to overlook talented individuals who cannot travel for interviews. Scheduling conflicts further compound this issue, as coordinating interviews between multiple stakeholders can be a time-consuming and logistically challenging endeavor.

Furthermore, conventional employment and hiring practices have been completely disrupted by the COVID-19 pandemic, calling for a thorough reassessment of accepted methods. The abrupt transition to remote work has made it more important than ever for businesses to find creative ways to conduct interviews virtually without sacrificing the process's quality.

In response to these challenges, the emergence of online interview platforms has offered a compelling alternative to traditional interviewing methods. These platforms leverage cutting-edge technologies to facilitate seamless communication between candidates and interviewers, transcending the constraints of time and space. By harnessing the power of video conferencing, collaborative coding environments, and real-time communication tools, these platforms enable companies to conduct interviews with candidates from around the world, thereby expanding their talent pool and driving greater inclusivity in the hiring process.

It is within this context of rapid technological innovation and evolving recruitment practices that InterviewSyncPro emerges as a pioneering solution. Built upon the principles of accessibility, efficiency, and user experience, InterviewSyncPro represents a paradigm shift in the way technical interviews are conducted. By integrating a comprehensive suite of features, including online collaborative coding environments, video conferencing capabilities, and facial recognition technology, InterviewSyncPro offers a holistic solution that addresses the multifaceted challenges of modern recruitment.

InterviewSyncPro represents the culmination of these technological advancements, offering a comprehensive platform tailored specifically for technical interviews. Built using the MERN (MongoDB, Express.js, React.js, Node.js) stack, InterviewSyncPro integrates a suite of innovative features designed to streamline every aspect of the interview process. Central to its functionality is the online collaborative text editor, which enables candidates to write and execute code in real-time, while simultaneously engaging in a video call with the interviewer.

To meet the various needs of businesses and startups conducting technical interviews online, InterviewSyncPro was created with a comprehensive approach. A variety of features designed to improve the interview experience for interviewers and candidates are included in its extensive feature set. The following are the main attributes that set InterviewSyncPro apart:

Online Collaborative Text Editor: Candidates can write and run code in real-time on a smooth platform made possible by InterviewSyncPro's integrated text editor. In contrast to conventional coding evaluations carried out through screen-sharing or offline integrated development environments, InterviewSyncPro's cooperative setting enables applicants to actively engage with coding tasks while getting prompt feedback from interviewers. This feature makes interviews more lively and helps interviewers evaluate candidates' coding expertise and problem-solving abilities in a dynamic environment.

Video Call using WebRTC: InterviewSyncPro enables candidates and interviewers to participate in high-quality video conferences by utilizing WebRTC (Web Real-Time Communication) technology. InterviewSyncPro guarantees a smooth and immersive interview experience by supporting real-time audio and video streaming, which facilitates natural communication and rapport-building between participants. Additionally, WebRTC technology reduces latency and guarantees dependable connectivity—even in low-bandwidth settings—improving the platform's overall dependability and accessibility.

Chat Functionality: Candidates and interviewers can communicate in real time thanks to InterviewSyncPro's integrated chat feature. During the interview process, this chat feature is a useful tool for sharing information, clearing up questions, and giving feedback. The chat feature improves teamwork and makes it easier for everyone participating in the interview to communicate effectively, whether they are talking about the requirements of a coding challenge, getting clarification on interview questions, or sharing crucial instructions.

Facial Recognition: The integration of facial recognition technology into InterviewSyncPro to confirm the interviewee's identity is one of its most notable features. InterviewSyncPro verifies that the person conducting the interview is, in fact, the approved candidate by examining facial biometrics. By adding a layer of security and authenticity to the interview process, this verification procedure reduces the possibility of impersonation or illegal access.

2. Related Work

In the field of technical interviewing, the major used applications are conventional video conferencing platforms like Teams, Google Meet, Skype, etc. They just provide the functionality of video calling and chatbox while for technical interviews we require more functionalities to comprehensively test the candidate.

A paper titled 'A Systematic Comparison of In-Person and Video-Based Online Interviewing' by Lobe, B., Morgan, D. L., & Hoffman, K. (Lobe et al., 2022) examined the conventional interviewing method of video conferencing and compared it with in-person interviews to recommend that though video-based online interviewing is going to replace in-person method but it calls for greater experimentation the wide range of available question strategies, to investigate what works best for generating online group discussions. Also, it raised the need for additional moderating techniques and to make the interview format more structured to prevent stiff interaction. It has pointed out the fallacy that online interviews produce smaller numbers of total codes and fewer words, this unavoidably generates a disadvantage with regard to code density.

A paper titled 'A modern online interview platform for recruitment system' published in Materials Today: Proceedings of ScienceDirect by Kathiravan, M. & Madhurani, M. & Kalyan, Sathya & Raj, Rahul & Jayan, Siddharth (Kathiravan et al., 2023) examined a similar problem. They created a web application for interviewing where they used a collaborative code editor, video/audio conferencing feature, and a collaborative whiteboard. Their goal is to make the interviewing process straightforward and foolproof, also cutting costs and time involved.

Deshmukh, Smita & Sarode, Dimple & Sonare, Shraddha & Yadav, Shweta produced a paper titled 'Online Platform for Coding Exams and Interviews' ((PDF) *Online Platform for Coding Exams and Interviews*, n.d.). This study explores a coding and interviewing web application making use of the Laravel framework of PHP and Docker to make a coding evaluation platform on the lines of HackerRank, InterviewBit, Code Wars, etc. They combined the coding round along with the interview such that the coding question will be prewritten by the interviewer and will be automatically tested against test cases which will eventually lead to a score where the interviewer can decide to take interviews with candidates meeting the cutoff. This project's primary objective is to create a web application that can be used in educational settings and supports many programming languages. It also aims to provide interviewers with the ability to create evaluations and evaluate exams (Rahul Kumar Mohata et al., 2021).

A WebRTC-based video conferencing system with an improved screen-sharing functionality has been presented by Abhay Kasetwar, Nikita Balani, Deepika Damwani, Alfiya Pandey, Aafreen Sheikh, and Apeksha Khadse (Kasetwar et al., 2022). The authors have put out a plan to enable cross-platform screen sharing using WebRTC technology within the Browser/Server architecture. The system architecture and all of its parts are thoroughly described in the article. The suggested WebRTC-based method guarantees high quality even in low-bitrate communication networks and offers a cross-platform, cross-device, and versatile user experience. The suggested approach is made to be simple to use and easy on the eyes. The writers have stressed how crucial privacy and security are to the system's architecture. Large-scale video conferences can be held using the technology, which is also scalable 1. The writers have carried out thorough testing on the system and found that it functions well even on networks with limited bandwidth. For cross-platform screen sharing and video conferencing, the suggested approach is a viable option.

According to Janghorban, Roksana, Latifnejad Roudsari, Robab, and Taghipour, Ali (Janghorban et al., 2014), the most commonly used method for data collection in qualitative research is interviewing. With technology improvements over the past few decades, the online interview has superseded onsite interviews in situations when time and money are limited, people are geographically dispersed, and physical mobility is restricted. The authors have suggested Skype as a synchronous online tool that enables researchers to hold small focus groups and individual interviews that are similar to those that take place on-site (Janghorban et al., 2014). The proposed system presents the characteristics of the Skype interview as an alternative or supplemental choice

for investigators who want to change their conventional approach to interviewing. The authors have emphasized the importance of security and privacy in the system design. The following are standout platforms that have gained recognition and widespread use in the narrower domain of technical interviewing, often employed by many tech companies internally (*The 8 Best Live Coding Interview Platforms in 2023* | by Daniel Borowski | *Tech x Talent* | *Medium*, n.d.):

- **Coderbyte:** Coderbyte stands out with its dynamic collaboration feature, offering a real-time collaborative code editor. This feature empowers both interviewers and candidates to collaboratively solve coding problems, making assessments interactive. The platform supports multiple programming languages, providing integrated compilers for seamless coding evaluations. Coderbyte is notable for its affordability, offering an industry-leading unlimited candidate and admin plan at \$199/month.
- **CoderPad:** CoderPad excels in its ease of use, providing a simple yet effective interview initiation process with built-in video/audio support. While it offers a straightforward coding environment, its features are limited compared to other platforms. It lacks support for Jupyter Notebooks, and back-end frameworks like Django or Ruby on Rails, and has primitive capabilities for multi-file, full-stack applications. \$100/month for 10 candidates, or \$70/month for unlimited candidates if billed annually.
- **HackerRank:** HackerRank shines in technical evaluations with a full-stack multi-file coding IDE integrated with video/audio support. Its breadth of roles is evident through features like embedded whiteboards and Jupyter Notebooks. The platform ensures a comprehensive coding environment for evaluating candidates across various technical domains. HackerRank offers a free starter plan with five interviews per month for a single interviewer, but additional features are strictly limited. As usage increases, pricing escalates, typically starting at \$10,000/year. The platform aims at larger enterprises, reflecting its pricing structure.
- **CodeSignal:** CodeSignal's proprietary IDE features powerful autocomplete and terminal access, providing a robust coding environment with integrated video/audio support. However, its focus is primarily on software development evaluations, excluding features for broader technical roles. CodeSignal offers a modern coding challenge library and grading systems. There is no free trial or transparent pricing, plans typically begin at \$10,000/year.
- **Byteboard:** Byteboard introduces a unique full-service approach, conducting end-to-end interviews with vetted engineers. While its technical evaluations are facilitated by the platform, the emphasis is on providing a consistent and candidate-friendly experience. Byteboard has limitations in terms of languages supported for back-end interviews and a predefined question set. Being a full-service solution, it comes with a higher price tag. Organizations can expect to spend at least \$25,000/year for a moderate amount of hiring.
- **Karat:** Karat distinguishes itself as a full-service solution with trained engineers conducting interviews in multiple languages. Its technical evaluations are facilitated by a staff of professionals, ensuring a standardized process. However, Karat's customization options are limited, and users must entrust the entire interview process to the platform. Organizations should anticipate spending at least \$25,000/year for a moderate amount of hiring.

3. Methodology

1. System Architecture

Start: This is the starting point of the user journey, where users are presented with options to either login or sign up if they are new users.

Login: Users who already have an account can choose to log in using their credentials.

Is new user?: This diamond-shaped decision point checks whether the user is new or returning. If they are new, they are directed to the signup process; otherwise, they proceed to the home screen.

Signup: New users are directed to the signup process where they can create a new account.

Home: After logging in or signing up, users are taken to the home screen. From here, they have access to various options such as entering a room code, creating a room code, leaving a room, or logging out.

Enter room code: Users can enter a specific room code to join an existing interview session.

Create room code: Users have the option to create a new room code, which likely initiates the setup of a new interview session.

Leave room: If users are currently in a room or interview session, they can choose to leave it.

Logout: Users can log out of their account, end their session, and return to the start point for subsequent login or signup actions.

Join room: Users who have entered a valid room code can choose to join the corresponding interview room.

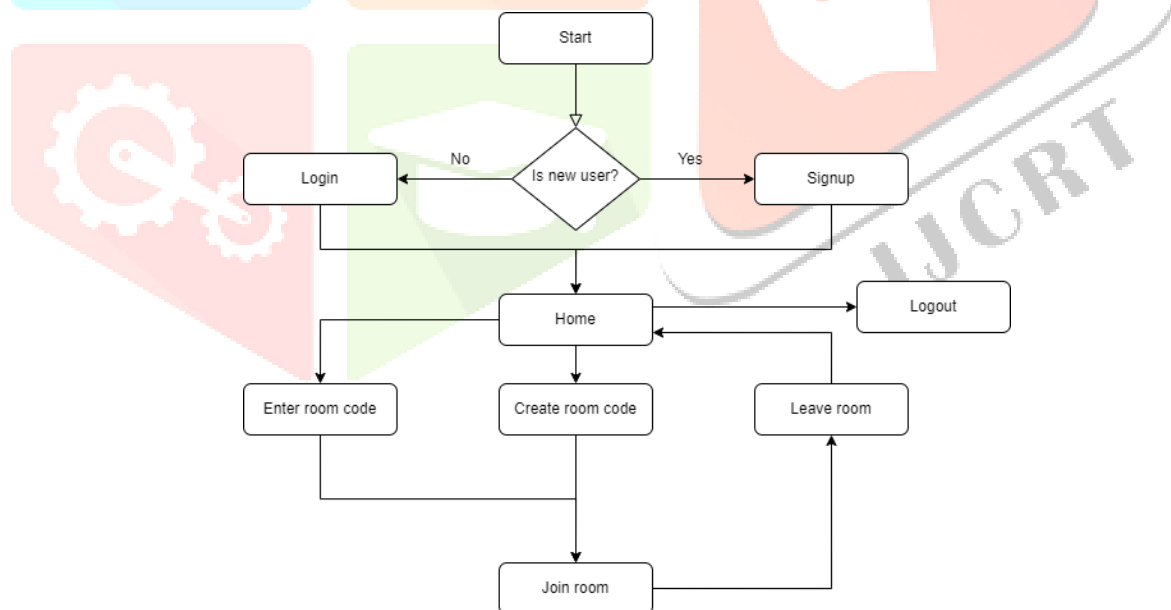


Figure 1 System Architecture

2. Technologies Used

InterviewSyncPro leverages a comprehensive suite of cutting-edge technologies to deliver a seamless and feature-rich platform for conducting technical interviews online. The selection of technologies was guided by considerations of scalability, performance, security, and user experience. Below is a detailed overview of the technologies employed in the development of InterviewSyncPro:

- **MERN Stack (MongoDB, Express.js, React.js, Node.js):**
 - MongoDB, a NoSQL database, is used to store interview data, user profiles, and other relevant information. Its flexible document-oriented model enables efficient data storage and retrieval, while its scalability features support the platform's growth.
 - Express.js serves as the backend framework for building RESTful APIs and handling server-side logic. Its lightweight and minimalist approach simplifies the development of robust and scalable web applications.
 - React.js powers the frontend of InterviewSyncPro, providing a component-based architecture for building interactive user interfaces. Its virtual DOM implementation enables efficient rendering and updates, resulting in a responsive and dynamic user experience.
 - Node.js facilitates server-side development, enabling JavaScript to be used across the entire application stack. Its event-driven, non-blocking I/O model ensures high concurrency and performance, making it well-suited for real-time applications like InterviewSyncPro.
- **WebRTC (Web Real-Time Communication):**
 - WebRTC technology is utilized to enable real-time audio and video communication between interviewers and candidates. By establishing peer-to-peer connections directly between clients, WebRTC ensures low-latency, high-quality communication without the need for intermediary servers. This technology is instrumental in providing a seamless and immersive interview experience, allowing participants to interact naturally and effectively.
- **Facial Recognition APIs/Libraries:**
 - InterviewSyncPro integrates facial recognition technology to verify the identity of interviewees and enhance the security of the interview process. Various facial recognition APIs and libraries are evaluated and incorporated into the platform, depending on factors such as accuracy, speed, and compatibility. These APIs/libraries provide functionalities such as face detection, landmark detection, and face matching, enabling InterviewSyncPro to authenticate users based on facial biometrics securely.
- **Authentication and Authorization Frameworks:**
 - Authentication and authorization in InterviewSyncPro are managed through JSON Web Tokens (JWT) and the bcrypt library. When users register, their credentials are securely stored using bcrypt for password hashing. Upon login, the server validates these credentials, issuing a JWT containing user identifiers and permissions, signed with a secret key. This JWT, included in subsequent requests' Authorization headers, grants access to authorized resources based on user roles and permissions. JWTs, incorporating expiration timestamps, enable secure token-based authentication, while bcrypt's robust hashing strengthens password security, ensuring a resilient authentication and authorization system within InterviewSyncPro.

3. Frontend Development

Frontend development in InterviewSyncPro plays a pivotal role in crafting a user-friendly, intuitive, and visually appealing interface that enhances the overall interview experience for both candidates and interviewers. Leveraging the React.js library, the frontend team focuses on creating modular, reusable components that facilitate seamless interaction and navigation within the platform. The following aspects highlight the key components and methodologies employed in frontend development:

- **Design Principles and User Interface (UI) Components:**

React.js components serve as the building blocks of InterviewSyncPro's UI, encapsulating logic, structure, and presentation within self-contained units. Components are designed with reusability in mind, allowing for efficient development and maintenance of the frontend codebase.

- **Client-Side Routing and State Management:**

- React Router is employed for client-side routing, enabling seamless navigation between different views and components within InterviewSyncPro. Route configurations are defined to map URL paths to corresponding React components, facilitating a single-page application (SPA) architecture.

- State management in InterviewSyncPro is handled using React's built-in state management capabilities using context API.

- **Integration of External Frameworks:**

Tailwind CSS is used in InterviewSyncPro to speed up website design. We can quickly style things by using pre-made classes. This maintains consistency in appearance and adherence to our design guidelines. Tailwind also makes it simple to alter the appearance of things. This facilitates the process of designing a visually appealing and functional website for our users. All things considered, we can improve and speed up the website by utilizing Tailwind CSS.

- **Optimization and Performance:**

Frontend optimization techniques are employed to improve the performance and loading times of InterviewSyncPro. Code splitting, lazy loading, and tree-shaking techniques are utilized to minimize the initial bundle size and load only the necessary code for each page or component.

4. Backend Development

Node.js Environment Setup: Setting up a Node.js environment is the first step in the backend development process. Because of its event-driven architecture and non-blocking I/O model, Node.js is selected to handle concurrent requests in an efficient manner.

Integration of Express.js Framework: Express.js, a minimalist web application framework for Node.js, is integrated into the backend. It facilitates the definition of API endpoints and the implementation of custom middleware functions by streamlining routing, middleware management, and request handling.

- **Designing RESTful APIs:** Backend architecture follows Representational State Transfer (REST) principles, with clear and intuitive API endpoints defined for client-server communication. CRUD operations are specified for interacting with data stored in the MongoDB database.
- **Integration with MongoDB Database:** The main data store is MongoDB, which provides an adaptable document-oriented database solution. Mongoose is an Object Data Modelling (ODM) library that makes integration easier and allows for smooth communication with Node.js applications.
- **Implementation of Authentication Middleware:** JSON Web Tokens (JWT) are used by middleware functions to handle authentication. A JWT with user data and permissions is created and signed with a secret key after a successful login, which makes it easier to authenticate and grant access to protected routes and resources.
- **Error Handling and Middleware:** Express.js middleware is used to implement robust error-handling mechanisms that catch and gracefully handle exceptions. To keep an application stable, middleware performs tasks like intercepting incoming requests, verifying user input, and handling errors.

5. Integration of Video Conferencing

Integrating video conferencing into our interview platform with WebRTC, Socket.IO, and Peer.js improved the interview experience by allowing seamless real-time communication. Using WebRTC's peer-to-peer communication capabilities, Socket.IO's event-driven architecture, and Peer.js's WebRTC simplification, our platform enabled high-quality video streaming between interviewer and candidate. This setup ensures efficient transmission of video and audio data, allowing for natural and uninterrupted conversations. Our platform optimized latency and reduced server load by supporting direct peer-to-peer connections via WebRTC, thereby improving the overall performance of the video conferencing feature. Socket.IO's real-time event-based communication ensured synchronized interaction between interviewers and candidates, allowing immediate feedback and collaboration. Peer.js simplified WebRTC implementation, making it easier to incorporate video conferencing functionality into our platform.

6. Implementation of Facial Recognition

The facial recognition system, which uses OpenCV's LBPH model and Haar cascade, streamlines the candidate identification process for interviews. A diverse array of candidate images are trained into the LBPH model using meticulous data collection and preprocessing, allowing it to recognize unique facial features. When a user approaches, their image is captured and subjected to face detection using the Haar cascade classifier, ensuring accurate localization of facial regions. The detected face is then passed through the trained LBPH model for recognition, which determines the corresponding candidate based on learned patterns.

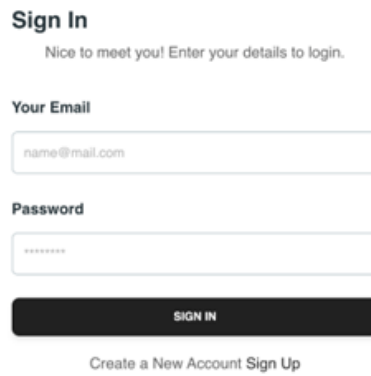
4. Literature Review

S no	Author	year	Focus of paper	Key points of coverage	Technique used	Parameter Analyzed	Research Gap
1	Roksana Janghorban Robab Latifnejad Roudsari Ali Taghipour	2014	Using skype platform for interviewing	Traditional interview challenges, online Skype interviews, real-time audio/video communication, convenience and accessibility, technological and ethical considerations.	Skype video calling	Skype interviews offer cost-effective, globally accessible alternatives to traditional interviews, with real-time communication, enhanced engagement, and important ethical considerations around privacy and consent.	No use of collaborative code editor, whiteboard, editor, facial recognition
2	Abhay Kasetwar , Nikita Balani , Deepika Damwani , Alfiya Pandey , Aafreen Sheikh , Apeksha Khadse	2022	WebRTC Based Video Conferencing System	WebRTC technology enables peer-to-peer communication over the web, facilitating seamless video conferencing with its components, offering advantages such as low latency,	WebRTC, HTML5, Node.js	Performance, Stability	Integration with Emerging Technologies For facial recognition Scalability

				high-quality audio/video, and ease of implementation, while also raising ethical and security concerns regarding user privacy and data protection.			
3	Aditya Kurniawan , Aditya Kurniawan , Christine Soesanto , Joe Erik Carla Wijaya4	2015	CodeR, real-time collaboration , programmers , coding, code editing, execution, teamwork.	CodeR: real-time code editor, collaborative programming, syntax checking, code execution, chat, project management	Web socket technology, real-time collaboration, operational transformation algorithms, consistency maintenance	CodeR performance evaluation, Apache Benchmark, HTTP requests handling, data transfer, requests per second, time per request, transfer rate, hardware specifications	Additional programming languages and tools support, Security measures, Cheating prevention
4	Prof.Smita Deshmukh , Dimple Sarode, Shraddha Sonare, Shweta Yadav	2022	Development of an online platform for coding exams and interviews	Providing an automated testing and evaluation environment	Docker containers, Laravel	Candidate authentication, Profile management, Coding examination , Test case evaluation, Interviewer authentication, Test case weightage, Candidate performance evaluation	No collaborative coding, No measures for cheating prevention And Performance Evaluation

5. Result

1. Login page



Sign In
Nice to meet you! Enter your details to login.

Your Email

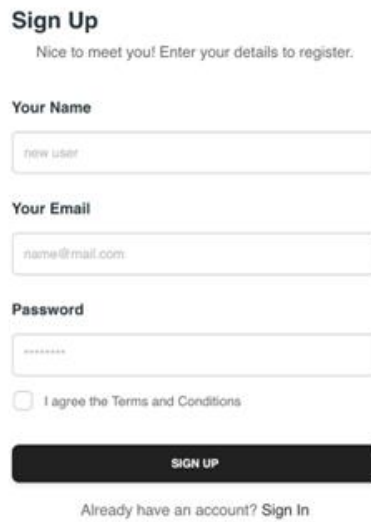
Password

SIGN IN

[Create a New Account Sign Up](#)

Figure 2 Login Page

2. Signup page



Sign Up
Nice to meet you! Enter your details to register.

Your Name

Your Email

Password

I agree the Terms and Conditions

SIGN UP

[Already have an account? Sign In](#)

Figure 3 Signup Page

3. Room Join Lobby

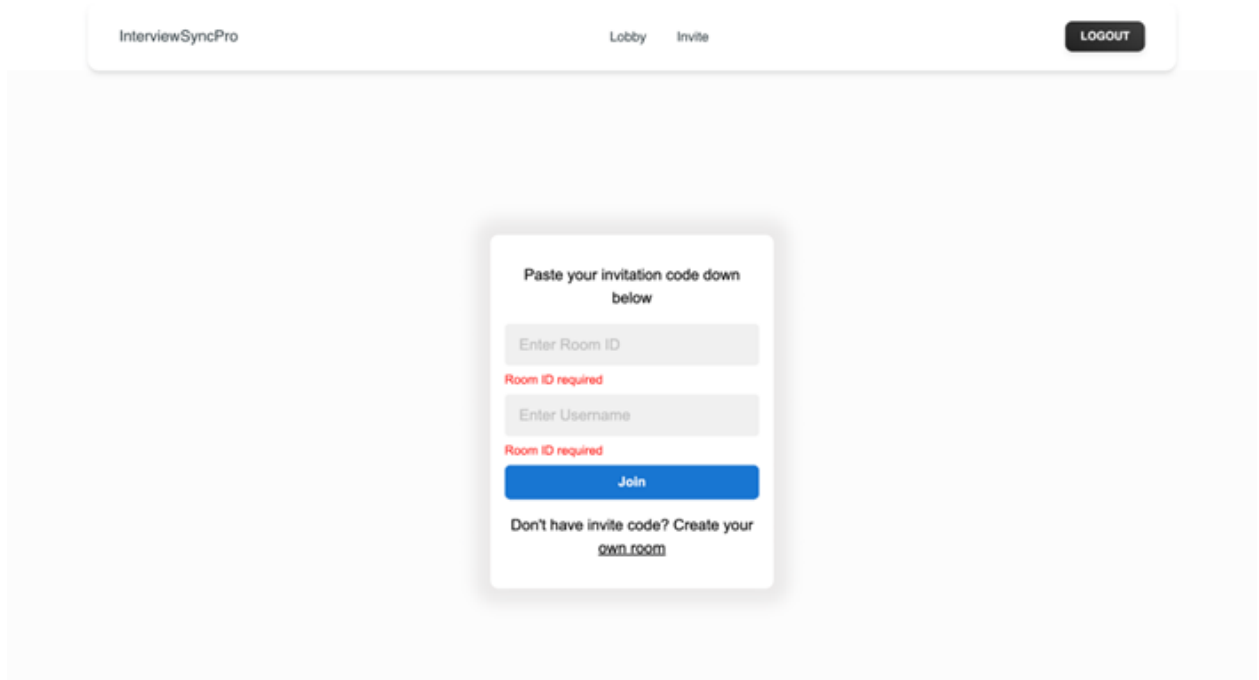


Figure 4 Join Room Lobby

4. Collaborative code editor

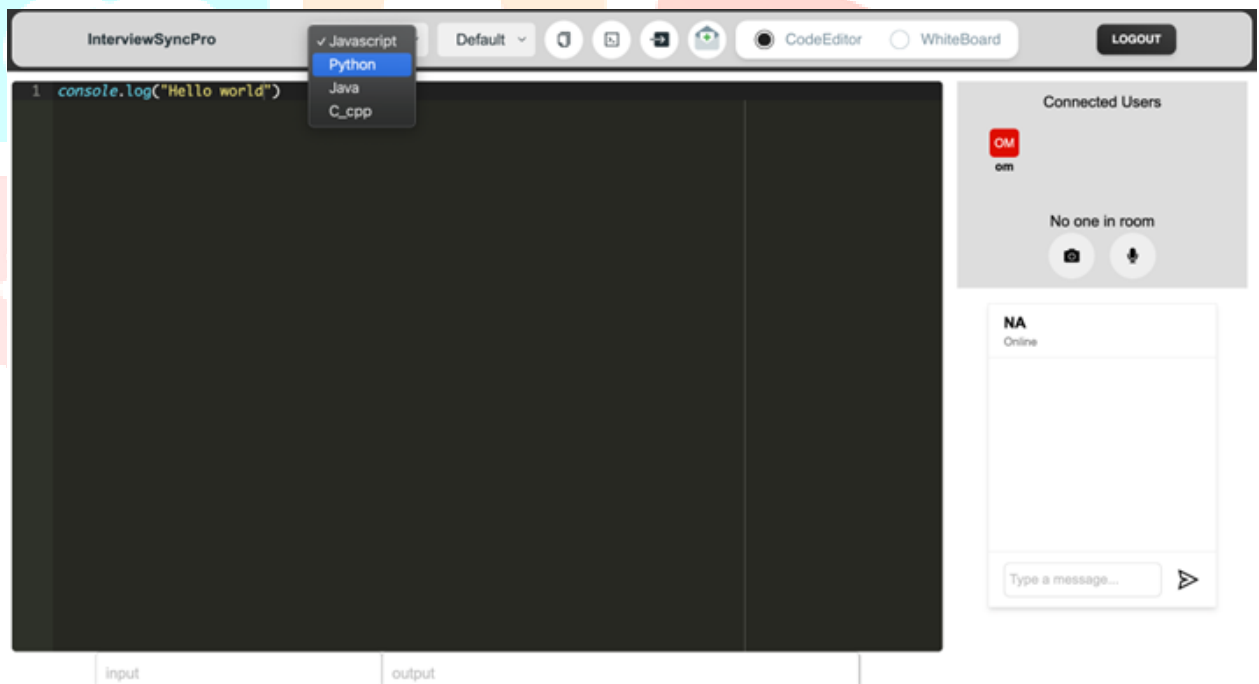


Figure 5 Collaborative Code Editor

5. Collaborative whiteboard

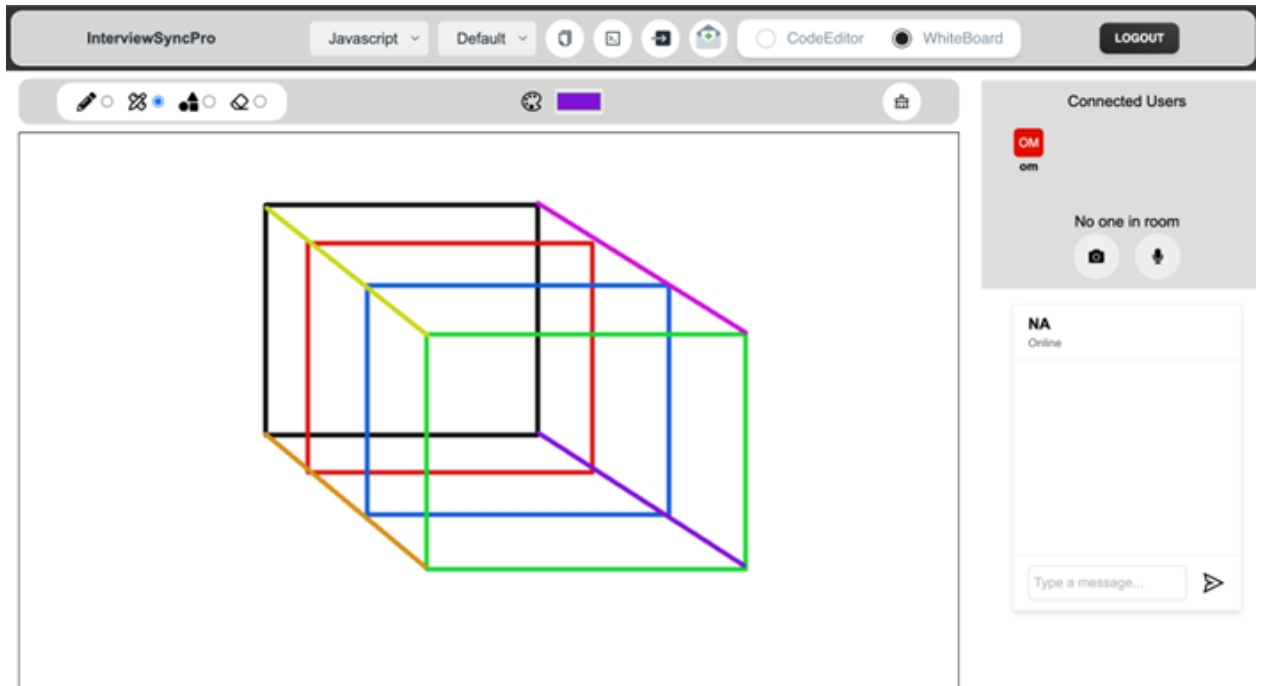


Figure 6 Collaborative Whiteboard

6. Webcam

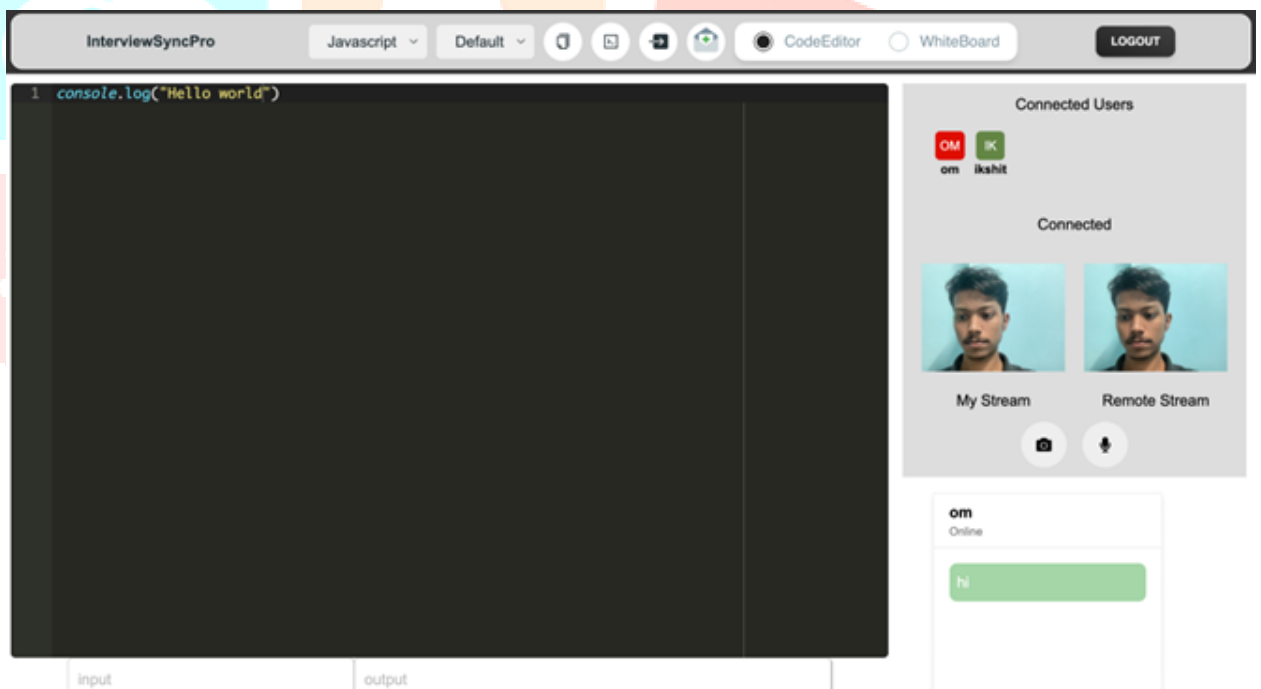


Figure 7 Collaborative Webcam

7. Email feature

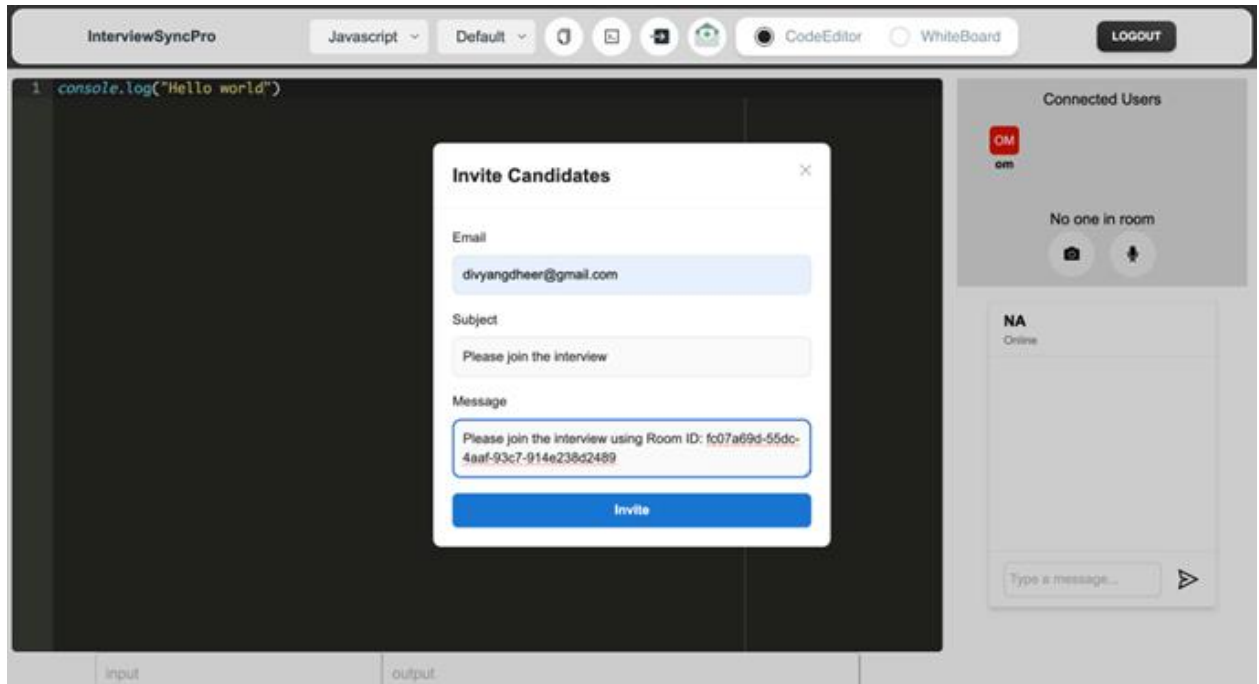


Figure 8 Email Feature

8. Logout feature

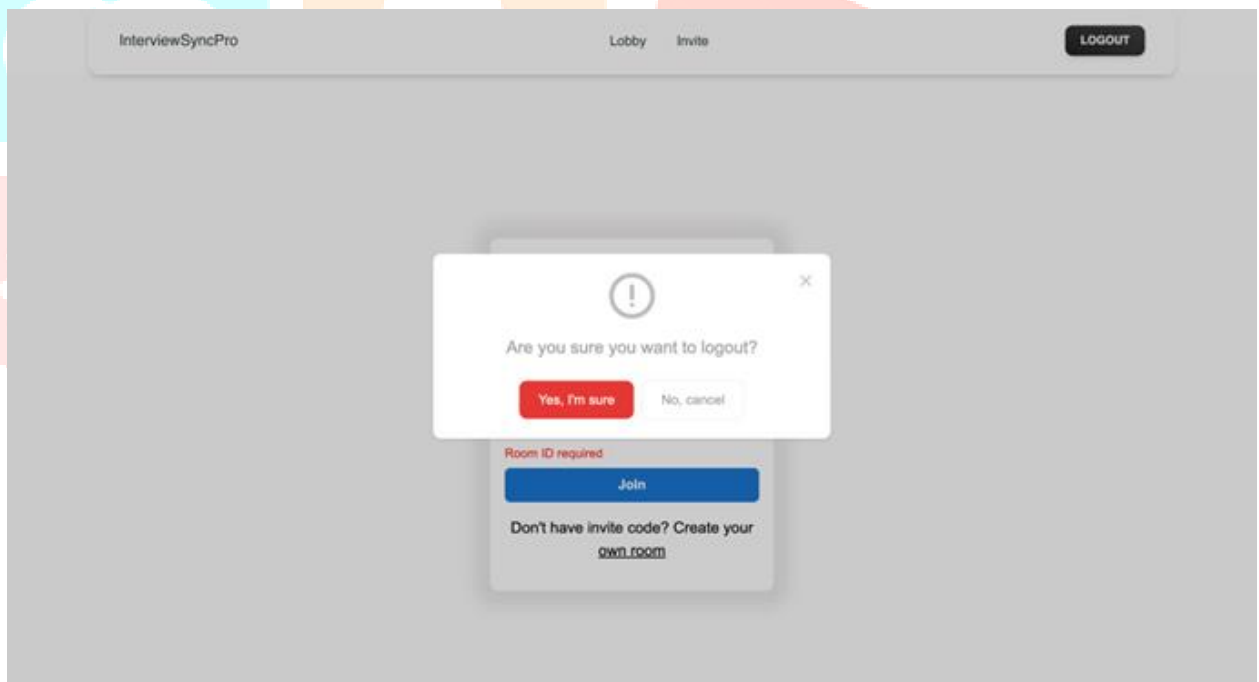


Figure 9 Logout Feature

Accuracy of Facial Recognition

The face recognition system underwent extensive testing with a dataset comprising multiple images, aiming to verify the accuracy of its identifications. Within this dataset, known individuals were expected to appear in 25 photos and unknown individuals in 25 photos. Upon analyzing the results through a confusion matrix given as Table 1, the system's performance in correctly identifying individuals can be observed. In the case of the known individuals, out of the 25 instances where they were expected to appear, the system accurately recognized their identity in 21 instances. A correct recognition implies that the system's identification matches the actual identity of the person in the photo. For the unknown individuals, the system correctly classified 22 out of the 25 instances.

This evaluation provides valuable insights into the performance of the face recognition system, we can calculate the accuracy of the system.

Table 1 Confusion matrix for image recognition

	Predicted Positive	Predicted Negative
Actual Positive	21	4
Actual Negative	3	22

The formulae for accuracy = $(TP + TN) / (TP + TN + FP + FN)$.

The overall accuracy that gets calculated from the confusion matrix is 86%.

6. Conclusion

In summary, the created interviewing platform offers a complete solution that expertly combines a number of characteristics necessary for carrying out effective and perceptive interviews. The interview process is improved for both interviewers and candidates when video streaming, collaborative code editing with multi-language support, a code compiler, and a collaborative whiteboard are included. The platform is made more convenient and secure by the incorporation of facial recognition technology, which also expedites the participant identification procedure.

In the future, there is much room for improvement and growth for the project. To assess applicant performance, more improvements can be achieved using AI-driven insights and sophisticated analytics. An even more immersive and interesting interviewing experience could be achieved by integration with virtual reality or augmented reality technologies. Furthermore, revisions to support industry trends and the new programming languages will guarantee that the platform stays current and flexible enough to meet changing demands in the labor market.

Our community-driven development methodology will make our open-source platform accessible to the general population. This indicates that the project's source code is available for everyone to see, utilize, edit, and share. The project is open to participation from people worldwide who can submit enhancements, new features, and bug fixes. Ours will be a free service model where businesses may easily utilize the interviewing platform.

REFERENCES

- [1] Dwi Prasetya, D., & Ashar, M. (2017). *Design of Interactive Whiteboard to Support E-Learning*. <https://doi.org/10.2991/ICOVET-17.2017.26>
- [2] Janghorban, R., Roudsari, R. L., & Taghipour, A. (2014a). Skype interviewing: The new generation of online synchronous interview in qualitative research. *International Journal of Qualitative Studies on Health and Well-Being*, 9(1). <https://doi.org/10.3402/qhw.v9.24152>
- [3] Kasetwar, A., Balani, N., Damwani, D., Pandey, A., Sheikh, A., & Khadse, A. (2022). A WebRTC Based Video Conferencing System with Screen Sharing. *International Journal for Research in Applied Science and Engineering Technology*, 10(5), 5061–5066. <https://doi.org/10.22214/ijraset.2022.43595>
- [4] Kathiravan, M., Madhurani, M., Kalyan, S., Raj, R., & Jayan, S. (2023). A modern online interview platform for recruitment system. *Materials Today: Proceedings*, 80, 3022–3027. <https://doi.org/10.1016/J.MATPR.2021.06.459>
- [5] Kumar, S., Pan, H., Wang, R., & Tseng, L. (2020). LiteDoc: Make Collaborative Editing Fast, Scalable, and Robust. *2020 IEEE International Conference on Pervasive Computing and Communications Workshops, PerCom Workshops 2020*. <https://doi.org/10.1109/PERCOMWORKSHOPS48775.2020.9156221>

- [6] Kurniawan, A., Soesanto, C., & Wijaya, J. E. C. (2015). CodeR: Real-time Code Editor Application for Collaborative Programming. *Procedia Computer Science*, 59, 510–519. <https://doi.org/10.1016/J.PROCS.2015.07.531>
- [7] Lobe, B., Morgan, D. L., & Hoffman, K. (2022). A Systematic Comparison of In-Person and Video-Based Online Interviewing. <https://doi.org/10.1177/16094069221127068>, 21. <https://doi.org/10.1177/16094069221127068>
- [8] (PDF) *Online Platform for Coding Exams and Interviews*. (n.d.). Retrieved March 3, 2024, from <https://www.researchgate.net/publication/360371440> *Online Platform for Coding Exams and Interviews*
- [9] (PDF) *VIDEO CONFERENCING AND ITS APPLICATION IN DISTANCE LEARNING*. (n.d.). Retrieved March 3, 2024, from <https://www.researchgate.net/publication/251237239> *VIDEO CONFERENCING AND ITS APPLICATION IN DISTANCE LEARNING*
- [10] Petrangeli, S., Pauwels, D., Van Der Hooft, J., Wauters, T., De Turck, F., & Slowack, J. (2018). Improving quality and scalability of WebRTC video collaboration applications. *Proceedings of the 9th ACM Multimedia Systems Conference, MMSys 2018*, 533–536. <https://doi.org/10.1145/3204949.3208109>
- [11] Rahul Kumar Mohata, Amita Goel, Vasudha Bahl, & Nidhi Sengar. (2021). Peer To Peer Real-Time Communication Using WebRTC. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, 178–183. <https://doi.org/10.32628/CSEIT217647>
- [12] Shae, Z. Y., Tseng, B., & Leung, W. H. (2001). Immersive Whiteboard Collaborative System. *Annals of Software Engineering*, 12(1), 193–212. <https://doi.org/10.1023/A:1013343611618>
- [13] *The 8 Best Live Coding Interview Platforms in 2023 | by Daniel Borowski | Tech x Talent | Medium*.(n.d.). Retrieved March 3, 2024, from <https://medium.com/coderbyte/the-8-best-live-coding-interview-platforms-in-2023-800ce2a3ee57>

