

Collaborative Mock Interview Platform

Mohd Mawan Ahmad

Computer Science and Engineering
Graphic Era Hill University
Dehradun, Uttarakhand, India
maawan18@gmail.com

Prateek Srivastava

Computer Science and Engineering
Graphic Era Hill University
Dehradun, Uttarakhand, India
psrivastava@gehu.ac.in

Vaibhav Bharti

Computer Science and Engineering
Graphic Era Hill University
Dehradun, Uttarakhand, India
vaibhavbharti1g@gmail.com

Mohd Faraz Shamsi

Computer Science and Engineering
Graphic Era Hill University
Dehradun, Uttarakhand, India
farazshamsi3@gmail.com

Dr. Amit Jain

Computer Science and Engineering
O P Jindal University
Raigarh, India
amitscjain@gmail.com

Ankit Vishnoi

Computer Science and Engineering
Graphic Era Hill University
Dehradun, Uttarakhand, India
ankitvishnoi.cse@geu.ac.in

Abstract—The collaborative mock interview platform is a pioneering solution addressing the imperative need for streamlined preparation and evaluation in data structures and algorithms (DSA) interviews within the contemporary technological landscape. Recognizing the pivotal role DSA interviews play in shaping the careers of aspiring computer students, the platform offers a vibrant, interactive space with features like real-time video streaming, shared coding environments, and automated code testing using cutting-edge technologies such as WebRTC. Driven by key principles, including the educational significance of DSA interviews and the multifaceted nature of preparation, the project aims to bridge the gap between theoretical understanding and practical application, providing users with an authentic DSA interview experience to foster problem-solving skills, effective communication, and real-time interaction. Beyond interview readiness, the platform promotes continuous improvement in technical skills, collaborative learning, and self-confidence. Prioritizing accessibility and inclusivity, it caters to users from diverse backgrounds, and its innovation in assessment, versatility for educators, learners, job seekers, and professionals, and commitment to adaptability ensure its relevance in the evolving landscape of technical interviews. In essence, the Collaborative Mock Interview Platform serves as a cornerstone for individuals aspiring to succeed in the technology sector, offering a unique, accessible, and dynamic arena for DSA interview preparation and fostering the growth of the next generation of technology professionals.

Index Terms—WebRTC, DSA, Sockets, Peer-to-Peer connection, Protocols, Web Sockets, HTTP, Interview

I. INTRODUCTION

With digital technology always changing, the Collaborative Mock Interview Platform is a revolutionary tool that helps people navigate the complex route of interviews for Data Structures and Algorithms (DSA) [1]. These interviews, which are recognized as entry points to lucrative job prospects in the technology sector, require a dynamic, engaging, and practical approach. Technical proficiency and problem-solving skills are integrated into DSA interviews, which go beyond textbook knowledge. The goal of this project is to offer a comprehensive solution that virtually duplicates the complexities of these interviews.

It provides a forum where people can fully experience the dynamics of technical interviews, from expressing their ideas clearly to coding under duress [2]. This initiative is motivated by a number of important factors:

- A. Importance: If you want to choose career in IT sector then you must have a good knowledge of Data Structures and Algorithms (DSA) along with practical knowledge also of how things work.
- B. Interview Preparation: For cracking the interviews requires a good knowledge of Data Structures and Algorithms (DSA) and code writing along with that problem solving skills and impressive communication is also required.
- C. Practical Goals: The overall goal of our project is to close the gap between practical implementation and theoretical understanding to provide the best real-time interaction between the participants on a virtual environment [3,4].
- D. Development of Skills: Our platform offers development of skills as the user will giving interviews on our platform which helps in their coding skills, communication skills and problem-solving skills.
- E. Accessibility: The platform offers communication in different origins and make it more and more easy to use platform to wide range of people.
- F. Innovation: Our platform offers the same experience but in a new way which is more efficient and effective by using technologies like WebRTC, WebSockets and many more [5,6].
- G. Practical Usage: This is useful in not only for students but any job seekers who want to switch their job and want to prepare for IT sectors job and also want to improve their Data Structures and Algorithms (DSA) abilities.

II. LITERATURE SURVEY

Now the time has evolved for the hiring procedure in jobs using technology. For using these types of technology, we need qualified panel of experts. There are some coding platforms which are popular in this segment like Leet Code, Hacker Rank and Code Forces are some of the platforms where you can do coding. Now a days the applicants have

access to all most all the resources as they are all available on the internet through which they can practice all types of coding questions and can increase their skills. Data Structures and Algorithms are the main focus of all the applicants while increasing their skills and for them there are online courses available such as Coursera, edX, and Udemy.

With the base knowledge of Data Structures and Algorithms by the courses applicants can establish a good foundation in these subjects. The format of interview has evolved too much and the gap between theoretical knowledge and practical knowledge is huge but like the coding platforms we also have some platforms like Pramp and Interviewing.io for these kind of services[8,9]. The services provide the students all kinds of knowledge before going to a real Interview. These services have experts which helps the students a lot. Apart from internet resources for Data Structures and Algorithms we also have Books like Gayle Laakmann McDowell's "Cracking the Coding Interview" and websites like Geeks-for-Geeks which provide a vast information on Data Structures and Algorithms. The candidates also share their interview experiences, types of questions asked, company specific topics on platforms like Glassdoor and Blind. There are some applications which uses collaborative interactions and are good for practicing like Visual Studio Code Live Share and Google Docs. There are many innovations online which the Organization uses more frequently for distant interviews, integrating features like video conferencing.

Interviews are always changing and adapting along with the tech industry keeping up with the newest technology and new things in data structures, algorithms, and problem-solving strategies. In conclusion, the need for experienced people for conducting these kinds of high-end interviews are causing a evolution in preparation for technical interview. In order to with stand all the issues and help all the users we came up with Collaborative Mock Interview Platform project to offer a realistic and adaptive solution to keep up with the evolving nature of DSA interviews.

III. METHODOLOGY

The Collaborative Mock Interview Platform is a new and interactive process build for user-specific, dynamic and efficient platform for DSA Interviews preparation.

- A. Start-Up: The platform starts evaluating all the requirements which is needed for the platform to run. In this phase it gathered all the information from which the issues with the current interview process are noted.
- B. Platform Development: Design and Architecture: This choice of selecting the frameworks and tech stack that will make the platform to achieve its goals. User Registration and Profiles: Features like user registration and profile management are developed in place, giving users the option to make, edit, and manage their profiles.
- C. Booking Interview Slots: Users are given a booking system where they need to choose date and time of their interviews. Question Submission: A page is developed that allows users to submit test cases and interview questions. This contains systems for validating questions to guarantee their quality.

- D. Streaming Videos and Collaborative Coding: WebRTC is used for real-time video streaming and giving the feel of

in- person interviews. WebRTC offers us to connect to different people without involving server [10].

It is developed to allow several people to work together on a coding problem in a collaborative environment. Automated Code Evaluation: An integrated system evaluates user-submitted code for accuracy and efficiency by comparing it to test cases that are provided. Containerization: To maintain stability and security, the code execution environment is containerized. Analytics and User Feedback: These tools are designed to give users a better understanding of their performance, areas of strength, and areas in need of development. Scalability and Performance Optimization: Optimizations makes sure the system stays effective and responsive even as the number of users rises rapidly.

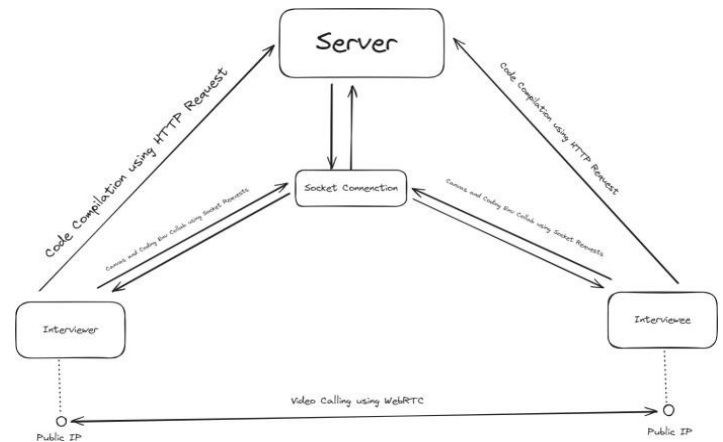


Fig 1. Connecting to server

- A. User Input and Comments: Our platform offers feedback mechanism which is very helpful for the future progress of this platform. The feedback creates a positive impact on the learning atmosphere, both the interviewers and interviewee can give their feedback. The feedback mechanism gives us valuable information of our platform's working and how it is handling things.
- B. Iterative development and continuous improvement: The platform is regularly updated with new features in response to user feedback and shifting industry expectations. Performance data: To evaluate the platform's influence, important performance data are continuously tracked, such as user happiness, interview success rates, and platform utilization.
- C. Training and Documentation: The documentation is for helping all the people who want to see how our platform is developed and understand the architecture of the platform.

IV. PROPOSED MODEL

This approach covers every stage of the project lifecycle, from planning to ongoing enhancement, and it demonstrates a dedication to responsibly, approachably, and scalably handling the special requirements and difficulties associated with DSA interview preparation and evaluation.

- A. Customization and Onboarding of Users: - Exclusive Privilege: Early adopters get the special chance to book simulated interviews on the platform. - Tailored

Experience: By entering individualized test cases and questions during the scheduling process, customers can personalize their interview experience.

- B. Safekeeping of Data - Server Security: The platform's server securely stores all customized questions and test cases, guaranteeing the privacy and accuracy of user data [11].
- C. Process of Scheduling: Individuals have the option to select a favorite time slot, such as the dynamic 10:00–10:30 hour. - Active Scanning: Every 30 minutes, the server actively searches for people who are online and have compatible time slots.
- D. Establishing Dynamic Connections: - Socket Connection: A complex socket connection is made after matching pairs are found to be compatible. - Integration with WebRTC: The power Using WebRTC makes it possible to share videos continuously, guaranteeing a smooth and excellent video stream [12].
- E. Instantaneous Communication: - Fluid Communication: Real-time communication between the interviewer and interviewee is made possible by the established socket connections.
- F. Continuous Data Exchanges: By exchanging data at exact 500-millisecond intervals, a discussion appears to be flowing and active. F. Environment for Coding: - Synchronization: Constant data exchanges via the established socket connections maintain perfect synchronization between the canvas and the coding environment.
- G. Integration of Microservices: - Dedicated Microservice: A microservice is a specialized backend server that is neatly incorporated into the system. - Purpose-Driven Backend: The platform's modular structure and general efficiency are greatly enhanced by the microservice's small and purpose-driven design.

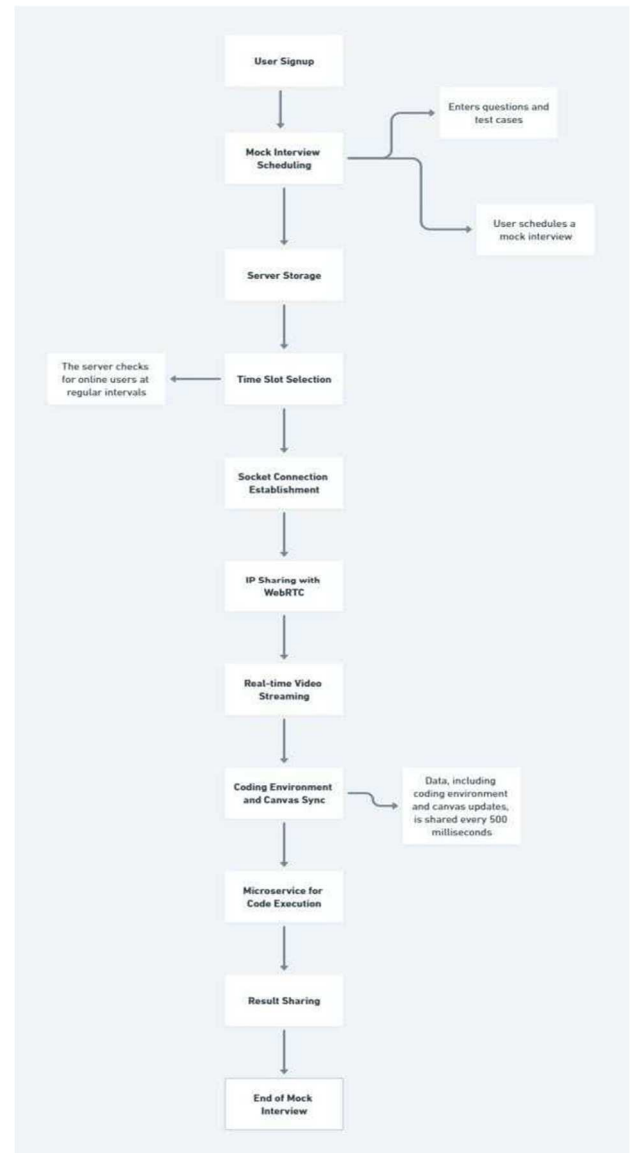


Fig 2. Working Flow of Model

- H. Code Execution and Assessment: - Microservice Functions: The microservice receives code, runs it, and evaluates it in accordance with a predetermined set of test cases. - Fast Results: The applicants will get their result in a proper and timely manner guaranteeing a smooth process.
- I. Effective Workflow: The services help us to improve the platform's overall efficiency by backend operations. This enables us to maintain this and we also can implement any future problems in it. All the components work together to give us a more professional and modern experience towards it.

V. COMPARING WITH EXISTING MODEL

- A. Prearranged Interviews: Our Structured Platform: This provides the user to schedule his interview according to his own wish and so that the user can be fully prepared before the interview and this is a more structured and effective way of getting things done quickly.

Customization: Allows users to input their own set of interview questions and test cases during the scheduling process, tailoring the interview experience to individual needs.

- Enhanced Relevance: Ensures that the interview content is relevant to the user's goals and objectives. Normal WebRTC System: - Primarily Audio-Video Focus: Primarily designed for audio video communication and may not inherently support features for integrating specific interview related content such as questions and test cases [13]. - Generic Communication Platform: Offers a more generic communication platform without specialized features for interview scenarios.

C. Server-Side Storage: Our System: - Data Retention: Stores interview-related data (questions, test cases) on the server for easy retrieval during scheduled sessions. - Persistent Information: Enables persistent storage, allowing for the continuity of information across multiple sessions and enhancing the overall user experience. Normal WebRTC System: - Peer- to-Peer Emphasis: Primarily emphasizes peer-to-peer connections, which may not include server-side storage of specific session-related data. - Transient Sessions: Sessions are often more transient, relying on Realtime communication without necessarily retaining detailed session data.

D. Socket Connection for IP Exchange: Our System: - Dynamic IP Exchange: Utilizes socket connections to dynamically exchange public IP addresses, facilitating seamless video sharing between participants. - Enhanced Connectivity: Ensures robust connectivity by actively managing IP exchange through dedicated socket connections. Normal WebRTC System: - Peer-to-Peer Focus: Primarily relies on peer-to-peer connections for audio-video communication and may not involve socket connections specifically for IP exchange. - Simplified Connectivity: Connectivity is typically established directly between peers without the need for intermediary socket connections for IP exchange [14].

E. Synced Coding Environment and Canvas: Our System: - Continuous Synchronization: Ensures the synchronization of the coding environment and canvas through continuous socket-based data exchanges. - Collaborative Coding: Provides a collaborative space where coding activities are seamlessly shared and synchronized in real time. Normal WebRTC System: - Audio-Video Synchronization: Primarily focuses on synchronizing audio and video streams, with limited or no built in features for syncing additional collaborative tools like coding environments. - Generic Collaboration: Offers general collaboration features but may not have the depth of integration required for coding scenarios.

F. Microservice for Code Execution: Our System: - Dedicated Microservice: Implements a specialized backend microservice for receiving and executing code, handling test cases, and promptly delivering results. - Efficient Code Processing: Enhances efficiency by offloading code execution tasks to a dedicated microservice. Normal WebRTC System: - Peer-to-Peer Communication: Primarily used for peer-to peer communication and may lack dedicated microservices for specific tasks such as code execution.

Communication- Centric: Designed more for communication rather than the execution and assessment of code [15]. In summary, your proposed system stands out by offering a tailored and structured platform for scheduled mock interviews, incorporating user customization, server-side storage, dynamic IP exchange, synchronized coding environments, and a dedicated microservice for efficient code execution — a comprehensive solution that goes beyond the typical capabilities of a generic WebRTC system.

VI. OUTPUT

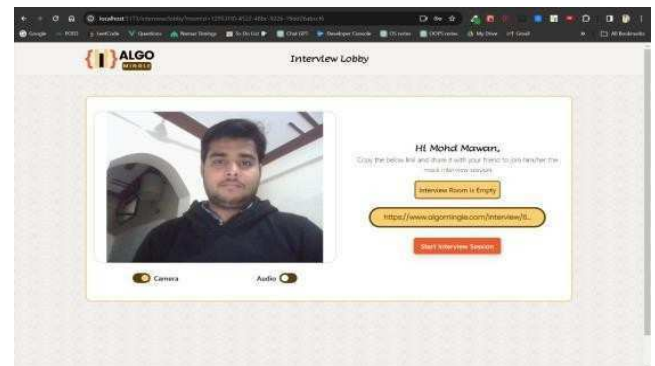


Fig 3. Interviewer Lobby Screen



Fig 4. Interviewee Lobby Screen

In Fig 3 and Fig 4 Lobby section for Interviewer and Interviewee are shown. The produced Mock Interview Platform is evidence of the effective fusion of Web-Sockets, WebRTC, and Docker, which together provide a complete and very productive virtual interview environment. By using the Web-Sockets, our platform offers a smooth and hassle-free communication between interviewers and candidates [16]. The main job of the WebRTC is that it has an excellent audio and video quality increasing the overall experience better. Apart from this WebRTC also offer screen sharing which enhance the productivity of the platform and an interactive process of interviewing.

Our Collaborative Mock Interview Platform uses Docker containerization which is portable and has impact on its scalability because each and every interview is inside a Docker container and the deployment is easy. This approach streamlines the managing of all the interview sessions. The combination of WebRTC, WebSockets and Docker offers us a seamless environment to work upon [17]. Our Platform mainly focus on those students, employee who needs a job in the field

of DSA and want to shape their skills according to the latest technology being used in today's time.

CONCLUSION

In conclusion our Collaborative Mock Interview platform is integrated creation of WebRTC, Sockets and Canvas. WebRTC is for the server-less video calls, sockets are for the real time coding environment and the canvas is for the visual interactions. These technologies combining together gives us power to take interviews to another level. The work of WebRTC is that a smooth and hassle-free video conversation without sacrificing the performance. The instant changes in the coding environment is done using the socket connections. Now apart from this we also have a collaborative canvas which can be used to express any ideas regarding the questions which has been asked in the interview. Our platform offers a flexible and efficient way of communication which improves the overall user experience. Our platform provides a solution for distance interviewing or remote interviewing giving them a whole new experience. Now a days remote interviewing is getting more and more popular and our platform is almost the best fit for these interviews. For the future planning it will require improvements which are based on the user experience and the upcoming new technologies and with its server-less architecture, interactive features, and real-time collaboration capabilities, it's positioned as a useful tool for remote interviews and provides an insight into the immersive and collaborative virtual interactions of the future.

REFERENCES

- [1] A study on the usage of data structures in information retrieval -V. R. Kanagavalli, G. Maheja - 1602.07799.pdf (arxiv.org)
- [2] Jennings, Cullen, Ted Hardie, and Magnus Westerlund. "Real-time communications for the web." *Communications Magazine*, IEEE 51.4 (2013): 20-26.
- [3] A Hybrid Approach for WebRTC Video Streaming on Resource-Constrained Devices – Bakary Dialli – <https://www.mdpi.com/2079-9292/12/18/3775>
- [4] Peer to Peer Multimedia Real-Time Communication System based on WebRTC Technology by Zina Nayyef, Sarah Faris Amer and Zena Hussain <https://www.researchgate.net/publication/331344763> Peer to Peer Multimedia RealTime Communication System based on WebRTC Technology
- [5] WebRTC Peer to Peer learning – Fateh Ali Khan – <https://www.ijert.org/research/webrtc-peer-to-peer-learning-IJERTV7IS030135.pdf>
- [6] Mohan, L, J Pant, P Suyal, and A Kumar, 'Support Vector Machine Accuracy Improvement with Classification', in *Proceedings - 2020 12th International Conference on Computational Intelligence and Communication Networks, CICN 2020*, 2020, pp. 477–81
<https://doi.org/10.1109/CICN49253.2020.9242572>
- [7] A Survey on Real-Time Communication for Web - Md. Habibur Rahaman – <https://www.scirj.org/papers-0715/scirj-P0715273.pdf>
- [8] Performance Analysis of webrtc and SIP for Video Conferencing - Navrattan Parmar – <https://www.ijitee.org/wp-content/uploads/papers/v8i9S/I11090789S19.pdf>
- [9] WebRTC: Real-Time Communication in Browsers - Cullen Jennings (Cisco) – <https://www.w3.org/TR/webrtc/>
- [10] Alvestrand, H., and Burnett, D. (Eds.). (2019). "webrtc 1.0: Real-time Communication Between Browsers." *World Wide Web Consortium (W3C) Recommendation*. Retrieved from <https://www.w3.org/TR/webrtc/>
- [11] Rescorla, E. (2014). "WebRTC Security Architecture." *Internet Engineering Task Force (IETF) Request for Comments: 5763*. Retrieved from <https://datatracker.ietf.org/doc/rfc5763/>
- [12] Fette, Ian, and Alexey Melnikov. "The websocket protocol." (2011).
- [13] Rosenberg, Jonathan. *Interactive connectivity establishment (ICE): A protocol for network address translator (NAT) traversal for offer/answer protocols*. No. RFC 5245. 2010.
- [14] Cherita L. Corbett, Raheem A. Beyah, John A. Copeland, Using Active Scanning to Identify Wireless NICs, in: *Proceedings of the 7th IEEE Workshop on Information Assurance*, U.S. Military Academy, West Point, NY, 21-23 June 2006.
- [15] WebRTC role in real-time communication and video conferencing – George Suciu – <https://ieeexplore.ieee.org/abstract/document/9119656>
- [16] An Overview of Web Sockets: The future of Real-Time Communication by Bhumi Gupta and Dr. M.P. Vani - <https://www.irjet.net/archives/V5/i12/IRJET-V5I1283.pdf>
- [17] Kapil, D, P Tyagi, S Kumar, and V P Tamta, 'Cloud Computing: Overview and Research Issues', in *Proceedings - 2017 International Conference on Green Informatics, ICGI 2017*, 2017, pp. 71–76 <https://doi.org/10.1109/ICGI.2017.>