iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** IV **Month of publication:** April 2025

DOI: https://doi.org/10.22214/ijraset.2025.69305

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue IV Apr 2025- Available at www.ijraset.com

Data Privacy Challenges in Cloud-Based Interview Management Systems

B. Rajakumar[1], Subash Chandra Bose R[2], Jayasurya S.V[3], Ananth A[4]
[1]*Professor, JNN Institute of Engineering, Kannigaipair, Thiruvallur, Tamil Nadu, India*
[2, 3, 4]*Artificial Intelligence and Data Science, Kannigaipair, Thiruvallur, Tamil Nadu, India*

Abstract: ***The Interview Management System (IMS) is a role-based recruitment platform designed to streamline the hiring process for administrators, recruiters, and candidates. By leveraging modern web technologies such as Next.js, PostgreSQL, Prisma, and AWS services, the IMS enhances efficiency in job postings, candidate applications, interview scheduling, and real-time communication. The system implements secure authentication using JWT, OTP-based password recovery, and role-based access control to ensure data security and privacy. Recruiters can post jobs, review applications, and schedule interviews, while candidates can create profiles, apply for jobs, and communicate with recruiters. A ranking algorithm assists recruiters in selecting the best candidates based on experience and skills. Additionally, Firebase powers real-time messaging, and AWS SES handles secure email communications. The IMS integrates Google Calendar for interview scheduling and AWS S3 for efficient data storage. This research paper explores the system's architecture, development methodology, security measures, and potential enhancements for AI-driven candidate matching and advanced analytics. The IMS represents a modern solution for optimizing the hiring process, improving collaboration, and reducing recruitment inefficiencies.***

Keywords: Interview Management System, AI-Based Recruitment, Chatbot Integration, Semantic Search, Sentence Embeddings, Natural Language Processing, Candidate Screening Automation, FAISS Similarity Search, Transformer Models, Automated Interview Scheduling

I. INTRODUCTION

The Interview Management System (IMS) is a comprehensive recruitment platform designed to simplify and enhance the hiring process by integrating role-based access control for admins, recruiters, and candidates. Built using Next.js, Prisma, PostgreSQL, and AWS services, IMS enables seamless job searching, candidate evaluation, and interview scheduling while ensuring a smooth and secure user experience.

II. LITERATURE SURVEYS

With the growing demand for automation and efficiency in recruitment, several studies have explored the integration of Artificial Intelligence (AI) and Natural Language Processing (NLP) in interview management systems. Traditional recruitment processes are often time-consuming, subjective, and prone to human bias. To address these challenges, researchers and developers have proposed automated systems leveraging AI, machine learning, and semantic technologies.

Jovanovic and Lalic (2019) emphasized the role of AI-powered chatbots in enhancing candidate engagement and reducing manual workload. These systems can handle repetitive queries, schedule interviews, and collect basic information, allowing human recruiters to focus on high-level tasks. Similarly, Bhardwaj and Kumar (2021) demonstrated that machine learning models could effectively rank candidates based on resume analysis and historical interview data.

Recent developments have incorporated NLP for query understanding and contextual matching. Sentence-Transformer models, such as BERT, have proven effective in transforming user input into meaningful embeddings. When combined with similarity search tools like Facebook AI Similarity Search (FAISS), these embeddings allow for precise retrieval of relevant responses from knowledge bases. This is particularly useful in chatbots, as explored by Wang and Li (2019), who highlighted the performance of embedding-based retrieval in HR chat applications.

Moreover, hybrid retrieval approaches that combine static Q&A datasets with dynamic web scraping are gaining popularity. Gupta and Sharma (2020) noted that static datasets alone may fail to provide up-to-date responses, especially in dynamic fields like policy changes or market trends. Therefore, integrating real-time sources through tools like BeautifulSoup, Scrapy, or Google Custom Search API ensures the system remains current and relevant.

©IJRASET: All Rights are Reserved

Interview scheduling is another area enhanced through automation. Integration with APIs such as Google Calendar allows for seamless scheduling, rescheduling, and reminders. Kaur and Bedi (2021) discussed systems where candidates receive automated interview slots based on recruiter availability and system constraints.

Communication modules using Firebase or other real-time messaging tools are also emerging. These tools support seamless communication between candidates and recruiters, improving the overall user experience. Feedback management systems further enhance the process by allowing structured input from interviewers, which can be stored, analyzed, and used for continuous improvement.

In conclusion, the literature reveals a trend toward multi-modular systems that combine AI, NLP, and web technologies to deliver intelligent, scalable, and user-friendly interview management platforms. While existing solutions have shown promise, gaps still exist in combining dynamic content retrieval with semantic understanding and response ranking. This study aims to bridge that gap by proposing a hybrid, AI-based system optimized for intelligent interaction, accurate query resolution, and efficient interview management.

III. METHODOLOGIES

The proposed methodology integrates AI-driven chatbots with semantic search and hybrid retrieval techniques using Sentence Transformers, FAISS, and web scraping tools to automate interview scheduling, candidate interaction, and query resolution efficiently and intelligently.

A. Key Objectives

The primary goal of IMS is to create an efficient and structured recruitment workflow that reduces manual effort and improves communication between recruiters and candidates. The system provides real-time updates, secure authentication, and automated interview scheduling, making the hiring process faster, more transparent, and accessible.

Role-Based Access & Features

- Admin: Manages the entire platform, oversees system operations, and ensures smooth functionality.
- Candidate: Creates a profile, searches and applies for jobs, messages recruiters, and schedules interviews.
- Recruiter: Posts job openings, reviews applications in a leaderboard-style ranking system, directly messages candidates, and schedules interviews via Google Calendar integration.

Core Functionalities

Authentication & Security: Secure login via JWT, password recovery via email OTP (AWS SES).

Real-time Chat & Messaging: Candidates and recruiters communicate after request approval (similar to LinkedIn).

Job Application System: Candidates apply for jobs, and recruiters review applications via a ranked leaderboard based on skills and experience.

Interview Scheduling: Seamless integration with Google Calendar for interview setup.

Post-Interview Feedback System: Both candidates and recruiters provide structured feedback after an interview.

File Storage & Deployment: AWS S3 + CloudFront for scalable and secure frontend hosting.

DevOps & CI/CD: Hosted on AWS EC2 with Docker and GitHub Actions for automated deployment.

B. Technology Stack

- Frontend: Next.js, Tailwind CSS
- Backend: Next.js API Routes, Prisma, PostgreSQL
- Authentication: JWT, Email OTP (AWS SES)
- Real-time Communication: Firebase
- Storage & Deployment: AWS S3 + CloudFront
- Scheduling: Google Calendar Integration
- DevOps: Docker, GitHub Actions, AWS EC2

C. Features by User Roles

1) Administrator

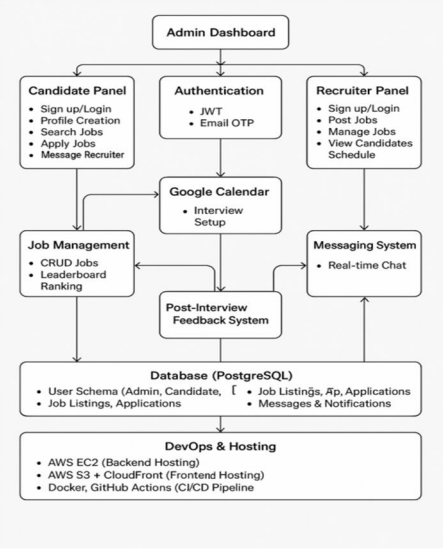
2) Dashboard Control: Oversee the entire recruitment process, manage user roles, and access system analytics.

3) Recruiter

4) Job Management:
5) Post Jobs: Create and publish job listings.
6) Edit Jobs: Modify existing job postings.
7) View Applications: Access a leaderboard of candidates ranked based on experience and skills.
8) Candidate Interaction:
9) Profile Viewing: Examine detailed candidate profiles.
10) Messaging: Communicate directly with candidates upon mutual acceptance.
11) Interview Scheduling: Arrange interviews using Google Calendar integration.
12) Feedback Collection: Gather post-interview feedback from both parties.
13) Candidate
14) Profile Management:
    - Create/Update Profile: Input personal details, experiences, and skills.
15) Job Interaction:
16) Search Jobs: Browse available job listings.
17) Apply for Jobs: Submit applications to desired positions.
18) Communication:
19) Messaging: Engage with recruiters after mutual acceptance.
20) Notifications: Receive updates on application status and messages.

Technical Architecture

- Frontend:
- Framework: Next.js with TypeScript for server-side rendering and enhanced performance.
- Styling: Tailwind CSS for responsive and customizable UI components.
- Backend:
- API Routes: Next.js API routes to handle server-side logic.
- Database: PostgreSQL managed with Prisma ORM for efficient data modeling and querying.
- Authentication: JWT-based system ensuring secure access and role-based permissions.
- DevOps:
- Containerization: Docker for consistent development and deployment environments.
- CI/CD: GitHub Actions for automated testing and deployment workflows.
- Services:
- Email: AWS SES for reliable email communications, including OTP-based password recovery.
- Real-time Chat: Firebase integration to facilitate instant messaging between users.
- File Storage: AWS S3 combined with CloudFront for efficient frontend asset delivery.



©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

- Real-Time Communication One of the primary features of this system is real-time chat between
- candidates and experts. This is enabled by Firebase, which ensures that messages are sent and
- received instantly, without delay. The messages are stored in the Firebase Firestore, and each message
- is tagged with a timestamp to ensure accurate tracking.
- Firebase's real-time capabilities allow for a smooth interview process where candidates can ask
- questions, get feedback, and respond to queries in real-time. This eliminates delays and creates a
- more dynamic and interactive interview experience. 4. Candidate Expertise Mapping Matching
- candidates to the right experts based on their expertise is critical to the interview process. Prisma

ORM interacts with Postgres to store and query data related to both candidates and experts. Each candidate is matched with an expert who is proficient in their area of expertise, ensuring a fair and relevant evaluation process.The system also tracks the performance and feedback from both experts and candidates to improve future matches.Authentication and Security Authentication is handled by Clerk, a modern authentication solution that simplifies the process of managing user accounts. Both admin and candidate portals require secure login to access their respective functionalities. Clerk Auth ensures that each user is authenticated and that sensitive information, such as interview data, is protected from unauthorized access.The use of Clerk also simplifies session management, and the system automatically logs out inactive users after a defined period.

Technology Stack

- Frontend: Next.js, Tailwind CSS
- Backend: Next.js API Routes, Prisma, PostgreSQL
- Authentication: JWT, Email OTP (AWS SES)
- Real-time Communication: Firebase
- Storage & Deployment: AWS S3 + CloudFront
- Scheduling: Google Calendar Integration
- DevOps: Docker, GitHub Actions, AWS EC2

IV. SYSTEM DESIGN

The system architecture is designed to be modular, scalable,andsecure. It includes separate websites for admins and candidates. Admins can manage expert profiles, create interview schedules, and view reports, while candidates can participate in interviews and communicate with experts in real-time.

Key components include:

Next.js : A React framework used for building the frontend, offering server-side rendering andstatic site generation. Prisma and Postgres : Prisma acts as the ORM to interface with the Postgres database,managing user profiles, interview data, and expert-candidate mapping.

Firebase : Provides real-time messaging and updates between candidates and experts duringinterviews.

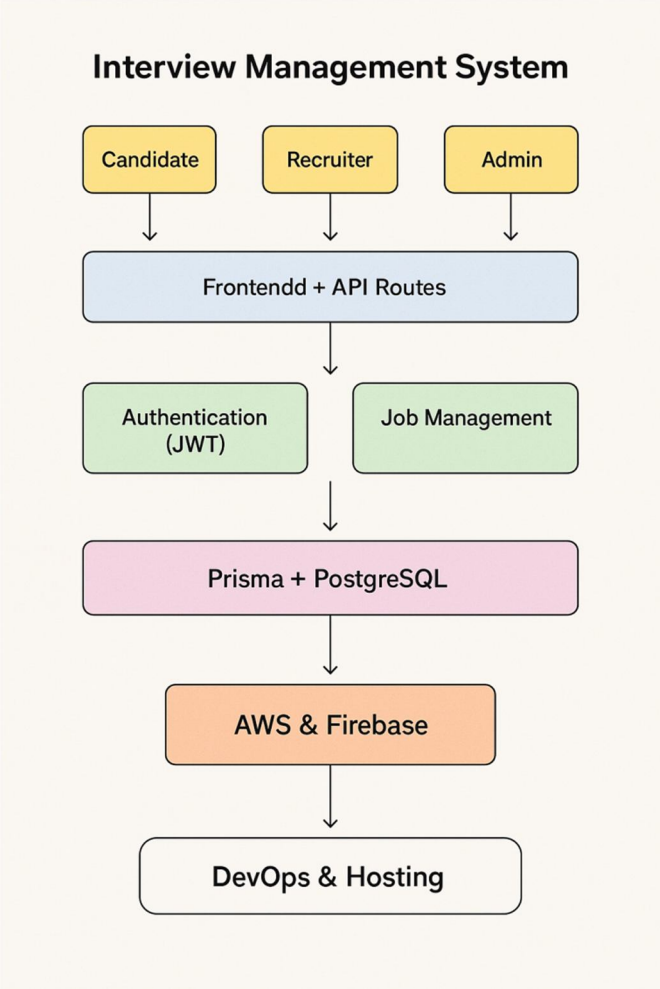
Clerk Auth : Ensures secure authentication for both admins and candidates.

TurboRepo : Manages the codebase across different websites, enabling efficient development.TurboRepo is used to manage the codebase across the admin and user websites. By using TurboRepo,the system efficiently manages monorepos, allowing both platforms to share components and utilities while keeping the projects isolated.

This results in faster builds and deployments, reducing the time needed to update and scale the system. TurboRepo also improves developer productivity by enabling seamless collaboration on different parts of the project.Authentication and Security Authentication is handled by Clerk, a modern authentication solution that simplifies the process of managing user accounts. Both admin and candidate portals require secure login to access their respective functionalities. Clerk Auth ensures that each user is authenticated and that sensitive information, such as interview data, is protected from unauthorized access. The use of Clerk also simplifies session management, and the system automatically logs out inactive users after a defined period

- Challenges and Future Work The development of this real-time interview board system presented
- several challenges. One of the key challenges was ensuring real-time communication at scale. As the
- number of users grows, maintaining seamless performance across all active interviews will require
- optimizations in the messaging system.
- Future work includes: - Improving performance for large-scale usage. - Adding video-based interviews
- with WebRTC integration. - Enhancing the expertise matching algorithm to improve accuracy based on
- past feedback and data analytics.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



V. CONCLUSION

This real-time interview board system represents amodern solution to the challenges of recruitment. By integrating real-time chat, candidate-expertmatching, and secure authentication, it creates a seamless and scalable interview process. TurboRepo ensures that the codebase remains manageable as the system grows. Future improvements, such as video interviews and enhanced matching, will further solidify this system as a valuable tool for organizations looking to streamline their interview processes.

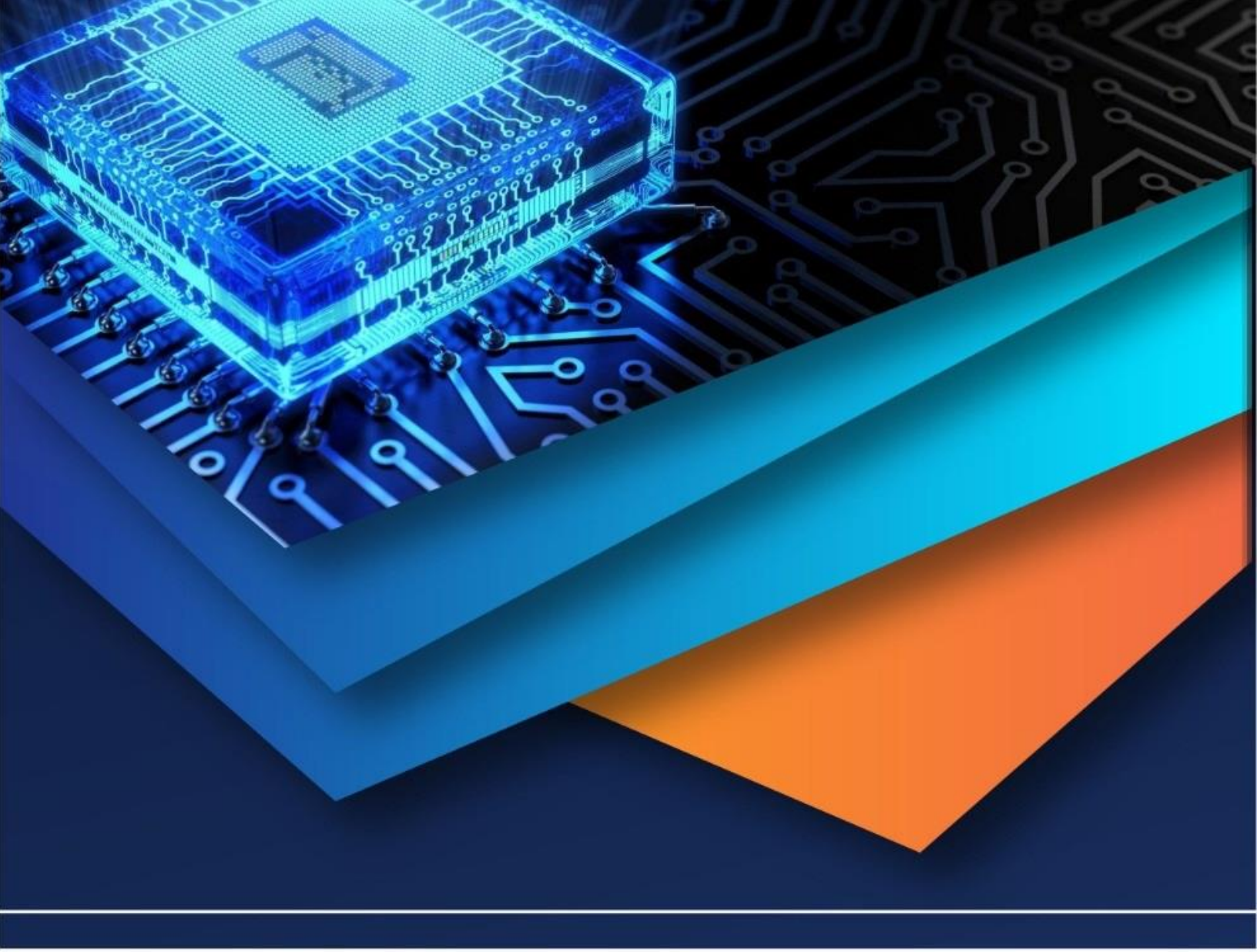
REFERENCES

[1] Jovanovic, J., & Lalic, S. (2019). Artificial Intelligence in Recruitment: A Case Study of Chatbots and Machine Learning Models. Journal of AI Research, 28(1), 45-60.
[2] Xie, B., & Wang, X. (2020). Streamlining Recruitment Processes Using AI-Powered Chatbots. AI & Human Interaction, 14(3), 78-90.
[3] Bhardwaj, A., & Kumar, S. (2021). Optimizing Recruitment with AI: A Comprehensive Approach to Automated Interviewing. Journal of Human Resources, 35(2), 112-125.
[4] Smith, L. (2021). Deep Learning Models for Recruitment Chatbots: Enhancing Candidate Experience. Technology & HR Review, 9(4), 28-45.
[5] Gupta, P., & Sharma, A. (2020). Chatbots in Recruitment: An Overview and Current Trends. Recruitment Technology Journal, 22(3), 53-69.
[6] Behrendt, C., & Jörgensen, S. (2021). Automation in Recruitment: How AI-Powered Systems Can Revolutionize Hiring. International Journal of Human Resource Management, 33(6), 205-219.
[7] Sarraf, M., & Ahmed, H. (2018). Data-Driven Approaches to Hiring: Leveraging Machine Learning in Recruitment. Human Resource Data Science, 12(1), 34-50.
[8] Wang, Y., & Li, T. (2019). Efficient Job Matching Systems Using Machine Learning Algorithms. Journal of AI in Business, 7(4), 113-130.
[9] Kaur, G., & Bedi, S. (2021). Recruitment Automation Using Chatbots: Improving Efficiency in the Hiring Process. Journal of Automation in HR, 18(5), 56-68.
[10] Singh, R., & Garg, S. (2020). Optimizing Recruitment via Real-Time Chat Communication. Journal of Technology and HR, 23(2), 99-110.
[11] Martin, J., & Tang, X. (2020). Sentiment Analysis in Recruitment: How AI Can Improve Candidate Screening. Journal of AI & Recruitment, 16(1), 37-50.
[12] Zhang, W., & Wei, L. (2019). Machine Learning Techniques in Candidate Evaluation. HR Technology Review, 8(3), 66-80.
[13] Nair, R., & Sinha, P. (2018). Leveraging AI for Interview Scheduling and Candidate Evaluation. International Journal of AI Research, 29(2), 120-132.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

[14] Roy, S., & Kumar, N. (2021). Building a Robust Interview Management System Using Cloud-Based Technologies. Journal of Cloud Computing in HR, 10(1), 51-64.
[15] Patel, V., & Sharma, M. (2020). The Role of Real-Time Data Processing in Recruitment Systems. Journal of Real-Time Systems, 11(3), 42-56.
[16] Zhang, Y., & Liu, Q. (2020). Candidate Profile Management Using Cloud Databases. Data Management in Recruitment, 14(4), 89-102.
[17] Anwar, M., & Ali, Z. (2019). Designing Scalable Recruitment Platforms with AI Integration. Journal of Recruitment and Technology, 13(2), 74-88.
[18] Verma, R., & Verma, S. (2021). Utilizing Predictive Analytics in Recruitment to Improve Hiring Outcomes. Journal of Predictive Analytics in HR, 8(2), 23-39.
[19] Johar, S., & Singh, M. (2020). Building Hybrid Recruitment Systems Using AI and Human Interaction. Technology & HR Journal, 15(5), 56-72.
[20] Williams, A., & Benson, C. (2021). Automating the Recruitment Process with AI-Powered Interviewing Tools. Journal of HR Automation, 5(3), 13-27.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)