

Smarthire: AI-Powered Placement Management System

Prof. Anita S. Talmale¹, Ayan Aalam², Daksh Koli³, Nilesh Amnerkar⁴, Arshan Sheikh⁵, Piyush Sontakke⁶

^{1,2,3,4,5,6}Department of Computer Science and Engineering, P.B.C.O.E Nagpur, India

Abstract -In modern educational institutions, placement activities are still managed using spreadsheets, manual verification, and disconnected communication methods. These traditional processes create delays, increase administrative workload, and reduce placement transparency for students and placement officers. The proposed system, “SmartHire: AI-Powered Placement Management System,” is an intelligent platform designed to automate and simplify institutional placement operations through a centralized digital interface. The system acts as a complete placement assistant that helps students manage profiles, upload resumes, track applications, and prepare for interviews using Artificial Intelligence (AI). SmartHire is developed using Next.js for the frontend interface, FastAPI for backend processing, and Firebase Firestore for cloud database management. The platform also integrates Artificial Intelligence (AI), resume analysis, and applicant ranking systems to improve placement efficiency and student readiness. Students can interact with the system through role-based dashboards that provide job recommendations, analytics, notifications, and AI interview preparation. This intelligent platform minimizes manual placement activities, improves communication efficiency, and supports a smart digital campus environment for modern educational institutions.

Key Words: Placement Management System, Artificial Intelligence, Resume Analysis, FastAPI, Firebase, Next.js, Cloud Computing, SaaS, Smart Campus, Automation

I. INTRODUCTION

The rapid growth of digital technologies has transformed traditional educational workflows into modern automated systems. Students today expect fast access to placement opportunities, application tracking, interview schedules, and career guidance. However, many institutions still depend on manual placement management methods such as spreadsheets, emails, and offline verification

processes. These approaches often create delays, increase administrative workload, and reduce communication transparency. To overcome these challenges, SmartHire provides a centralized AI-powered solution for institutional placement management. The platform is specially designed to automate placement workflows and improve operational efficiency within educational institutions. SmartHire allows students to create profiles, upload resumes, receive AI-generated resume analysis, apply for jobs, and prepare for interviews through intelligent assistance modules. The system integrates a Next.js frontend interface, FastAPI backend services, Firebase Firestore database, and AI-powered processing modules. Placement administrators can manage jobs, applications, student verification, placement drives, and analytics through dedicated dashboards. The platform also generates real-time reports and notifications that improve communication between students and placement officers.

II. Body of Paper

2.1 System Architecture

The SmartHire platform is designed using a cloudbased multi-layer architecture to improve scalability and system performance.

- **Front-end:** Developed using Next.js, providing responsive dashboards and user-friendly interfaces.
- **Back-end:** FastAPI handles business logic, authentication, analytics, and AI processing operations.
- **Database:** Firebase Firestore stores placement data such as jobs, applications, profiles, and reports.
- **Storage:** Cloudinary securely stores resumes, profile photos, and verification documents.

2.2 Methodology and NLP Implementation

The SmartHire system automates placement operations using Artificial Intelligence (AI), secure authentication,

and cloud technologies. When students upload resumes, the AI engine extracts resume content and analyzes technical skills, educational details, and project information. The system generates ATS scores, improvement suggestions, and candidate-job matching results. AI-based interview preparation modules also generate mock interview questions according to selected job categories.

2.3 Proposed Algorithm

- Input student profile and resume data.
- Validate authentication token and user role.
- Extract and preprocess uploaded resume content.
- Analyze resume using AI models.
- Store application data in Firebase Firestore.
- Generate notifications and placement analytics.

2.4 Technology Used

1. Frontend Stack

- Framework: Next.js 14
- Language: TypeScript
- Styling: Tailwind CSS
- Animations: Framer Motion
- Icons: Lucide React
- API Communication: Axios
- Charts & Analytics: Recharts

2. AI & Machine Learning

- Models: OpenAI GPT-4o-mini and Mistral AI
- Architecture: LangChain and LangGraph based AI workflow system
- Resume Analysis: AI-powered ATS scoring and recommendation engine
- OCR Processing: AI-based marksheet and document extraction

3. Backend & Database

- Backend Framework
- Framework: FastAPI
- Language: Python 3
- Validation: Pydantic schemas

Database & Cloud Services

- Database: Firebase Firestore (NoSQL document database)

- Authentication: Firebase Authentication
- Storage: Cloudinary cloud storage
- File Processing: Resume parsing and OCR extraction
- Development & Configuration Tools: dotenv and async middleware

4. **Security:** API keys managed through environment variables, Firebase JWT authentication, Role-Based Access Control (RBAC), request validation, and secure cloud storage mechanisms.



2.4.1 Data Flow Flowchart

2.5 Data Analysis and Results

To evaluate the efficiency of SmartHire, a comparison was conducted between traditional placement workflows and the SmartHire automated system.

- Group 1 (Manual): Placement activities managed using spreadsheets and offline communication.
- Group 2 (SmartHire): Students and administrators using the AI-powered placement platform.

2.6 Advantages of the System

- AI-powered resume analysis and interview preparation.
- Centralized placement management platform.
- Real-time notifications and application tracking.
- Reduced manual placement workload.
- Secure cloud-based architecture.

2.7 Admin Module and Data Management

To ensure efficient placement workflow management, SmartHire includes a dedicated Admin Dashboard. This module allows authorized administrators to manage jobs, applications, placement drives, verification workflows, and reports without direct database access.

cement drives, verification workflows, and reports without direct database access.

2.7.1 AI-Powered Resume Analysis

The platform includes an AI Resume Analyzer that converts uploaded resume data into structured information for ATS evaluation and candidate-job matching.

The module performs:

- Resume extraction
- Skill analysis
- ATS score generation
- Resume improvement suggestions

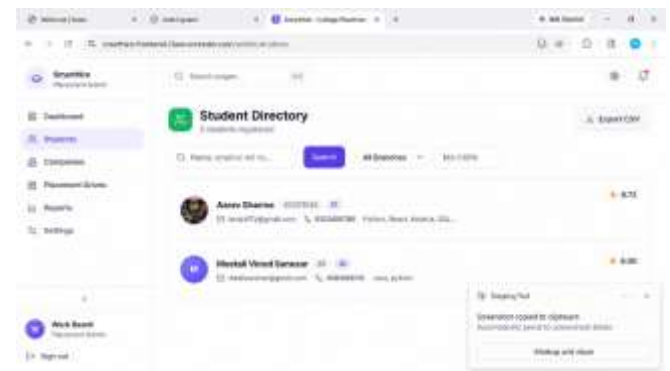
As shown in Fig. 2, the dashboard includes an AI-Powered Resume Analyzer that evaluates uploaded resumes using Artificial Intelligence (AI) models. The module extracts technical skills, programming languages, frameworks, and project-related information from unstructured resume documents and converts them into structured analytical data. It also generates ATS scores, identifies resume strengths, and suggests improvement areas to enhance placement readiness. This intelligent analysis system helps students optimize their resumes while reducing manual screening effort for placement administrators.



2.7. Placement Data Management (CRUD Operations)

As shown in Fig. 3, the SmartHire Admin Dashboard provides a centralized interface for managing student placement data and recruitment workflows. The system supports complete Create, Read, Update, and Delete (CRUD) operations for student profiles, applications, placement drives, and company records. Administrators can search students using filters such as branch and CGPA, monitor placement performance, and export placement data for reporting purposes. This centralized management system improves operational efficiency, simplifies data handling, and reduces manual administrative effort

- Job posting and management



- Student application tracking
- Placement drive scheduling
- Verification and approval management
- Analytics report generation

III. CONCLUSIONS

The development of SmartHire successfully addresses placement management challenges in educational institutions through automation, analytics, and Artificial Intelligence (AI). By integrating Next.js, FastAPI, Firebase Firestore, and AI-powered resume analysis, the platform provides an efficient solution for modern placement operations.

The implementation of AI-based resume analysis and interview preparation improves student placement readiness and reduces manual administrative workload. Furthermore, the analytical reports generated by the system improve institutional visibility and operational transparency.

In conclusion, SmartHire promotes a modern digital placement environment that improves communication, workflow management, and placement efficiency. Future work will focus on multi-tenant SaaS deployment, predictive placement analytics, and mobile application integration.

ACKNOWLEDGEMENT

The authors would like to express sincere gratitude to the Department of Computer Science and Engineering for providing guidance, technical support, and facilities during the development of SmartHire.

We are thankful to our project guide and faculty members for their valuable suggestions, encouragement, and continuous support throughout the implementation of this project. Their expertise and mentorship played an important role in the successful completion of the research work.

REFERENCES

1. Sommerville, I.: *Software Engineering*. 10th edn. Pearson Education, London (2015)
2. Fowler, M.: *Patterns of Enterprise Application Architecture*. Addison-Wesley Professional, Boston (2002)
3. Google Firebase Team: *Firebase Documentation and Cloud Firestore Guide*. Google Developers Documentation. Available at: <https://firebase.google.com/> (2025)
4. Ramírez, J., Patel, S., Kumar, A.: AI-Based Resume Screening and Applicant Ranking System. In: *International Conference on Artificial Intelligence and Data Engineering*. IEEE (2021) 112–118
5. OpenAI: *OpenAI API Documentation and GPT Models*. Available at: <https://platform.openai.com/docs/> (2025)
6. Tiangolo, S.: *FastAPI Framework Documentation*. Available at: <https://fastapi.tiangolo.com/> (2025)
7. Vercel Team: *Next.js Documentation for Full Stack Web Applications*. Available at: <https://nextjs.org/docs> (2025)
8. Pressman, R.S., Maxim, B.R.: *Software Engineering: A Practitioner's Approach*. 8th edn. McGraw-Hill Education, New York (2019)
9. Cloudinary Ltd.: *Cloudinary Media Storage and Management Documentation*. Available at: <https://cloudinary.com/documentation> (2025)
10. Goodfellow, I., Bengio, Y., Courville, A.: *Deep Learning*. MIT Press, Cambridge (2016)
11. Mistral AI Team: *Mistral AI Large Language Model Documentation*. Available at: <https://docs.mistral.ai/> (2025)
12. Brown, T., Mann, B., Ryder, N., et al.: Language Models are Few-Shot Learners. In: *Advances in Neural Information Processing Systems (NeurIPS)*, Vol. 33 (2020) 1877–1901
13. LangChain Developers: *LangChain Framework Documentation for AI Applications*. Available at: <https://python.langchain.com/> (2025)
14. Google Cloud Team: *Cloud Firestore Scalable NoSQL Database Documentation*. Google Cloud Platform (2025)
15. SmartHire System Architecture and Module Analysis Report