

# **Job Hiring and Applying Web-App**

**Prof. Malatesh Kamatar<sup>1</sup>, Prof. Indira<sup>2</sup>**

**Mr. Abhishek<sup>3</sup>, Mr. Abhishek S H<sup>4</sup>, Mr. Adarsha<sup>5</sup>, Mr. Ashok B<sup>6</sup>**

<sup>1,2</sup>Professor, Department of CS&E, Proudhavevaraya Institute of Technology, Hosapete, Karnataka, India

<sup>3,4,5,6</sup>Students, Department of CS & E, Proudhavevaraya Institute of Technology, Hosapete, Karnataka, India

**Abstract:** *This paper presents the development of "CareerFlow," a Job Hiring and Applying Web-App designed to bridge the gap between job seekers and recruiters. In the current digital landscape, traditional hiring processes are often inefficient, manual, and cluttered. This project introduces a streamlined web application that facilitates secure user authentication, dynamic job postings, and real-time application tracking. The system utilizes a Full Stack architecture to provide a user-friendly interface for candidates to apply for jobs and for administrators to manage vacancies effectively. This solution aims to democratize access to employment opportunities by simplifying the connection between talent and industry*

**Keywords:** Talent Discovery, Next.js, Firebase Authentication, Applicant Tracking System (ATS), Live Location Search, Dynamic UI/UX, Web Application, Recruitment System, Job Portal, Hiring Platform

## **I. INTRODUCTION**

UNEMPLOYMENT and the complexity of finding suitable candidates are significant challenges in the modern economy. While numerous job portals exist, many are difficult to navigate for non-professional or entry-level users. This document outlines the creation of a dedicated "Job Hiring and Applying Web-App" to address these issues. The proposed system simplifies the recruitment process by offering two distinct modules: one for Candidates and one for Recruiters. The goal is to create a centralized hub where talent meets opportunity without the noise of social networking features found in other platforms. The application ensures that the hiring process is transparent, fast, and accessible to users with varying levels of technical expertise.

## **II. LITERATURE SURVEY**

### **1. Traditional Corporate Portals (e.g., LinkedIn, Naukri)**

Observation: These platforms are highly effective for white-collar corporate roles but are often too complex for non-professional or blue-collar workers.

Gap: They lack "Market Transparency" regarding salary valuation for specific skills and often suffer from "Ghost Jobs" (inactive listings).

### **2. Local Classifieds (e.g., OLX Jobs, Local Ads)**

Observation: These sources are accessible but lack verification and organized tracking systems.

Gap: There is no "Applicant Tracking System (ATS)" for small business owners, leading to chaotic hiring processes via phone calls or WhatsApp.

### **3. The Proposed System**

Solution: It's a bridges these gaps by offering a "Talent Discovery" approach. It introduces Dynamic Salary Valuation to ensure wage transparency and utilizes Live Location Search to connect candidates with nearby opportunities instantly. The integration of an ATS Kanban Board democratizes enterprise-level hiring tools for all recruiters.

## **III. OBJECTIVES**

1. For Everyone (Not just Office Jobs) To build a website that helps both professional people (like software engineers) and non-professional workers (like drivers or electricians) find jobs easily in one place.



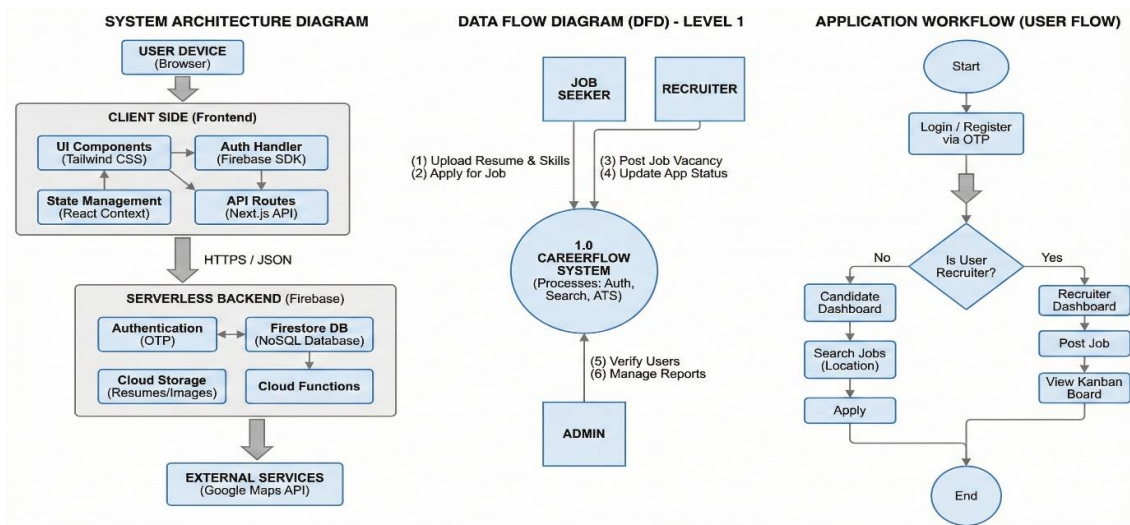
2. To Show Real Value (Salary) To help job seekers understand how much salary they should get based on their skills. We want to stop the confusion about money by showing a "Dynamic Salary" value.
3. To Find Jobs Nearby To make a "Live Location Search" feature. This helps people find jobs close to their homes so they don't have to travel far.
4. To Make Hiring Easy for Recruiters To give recruiters a simple board (Kanban Board) where they can drag and drop candidates. It helps them organize who applied, who is interviewing, and who is hired, without any mess.

#### IV. SYSTEM DESIGN WORK FLOW

Frontend Layer: Built with Next.js and Tailwind CSS. It handles the Dynamic UI/UX and interacts with the user.

Service Layer: Manages API calls for the Multi-Language Chatbot and Location Services.

Data Layer: Firebase acts as the Backend-as-a-Service (BaaS), handling Authentication, Database (Firestore), and Storage.

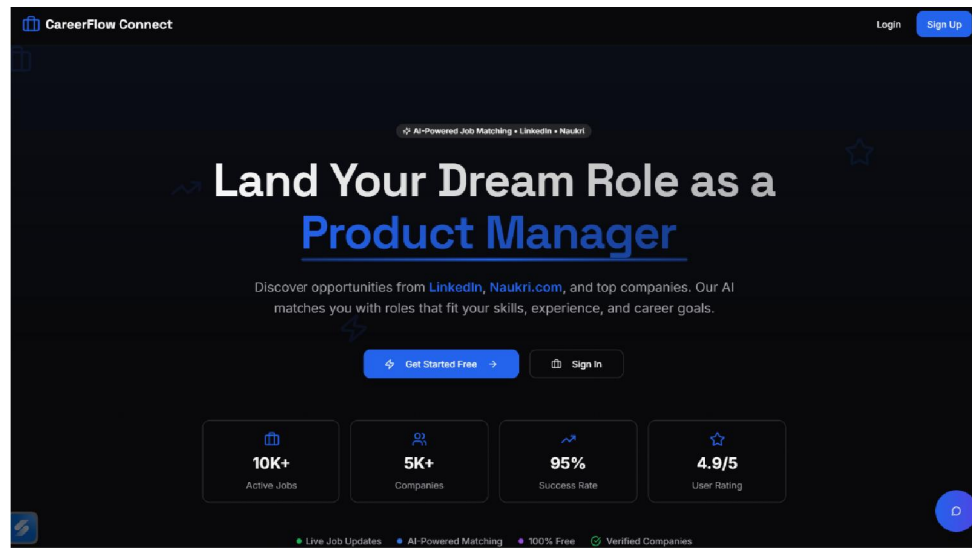


#### V. RESULTS AND DISCUSSION

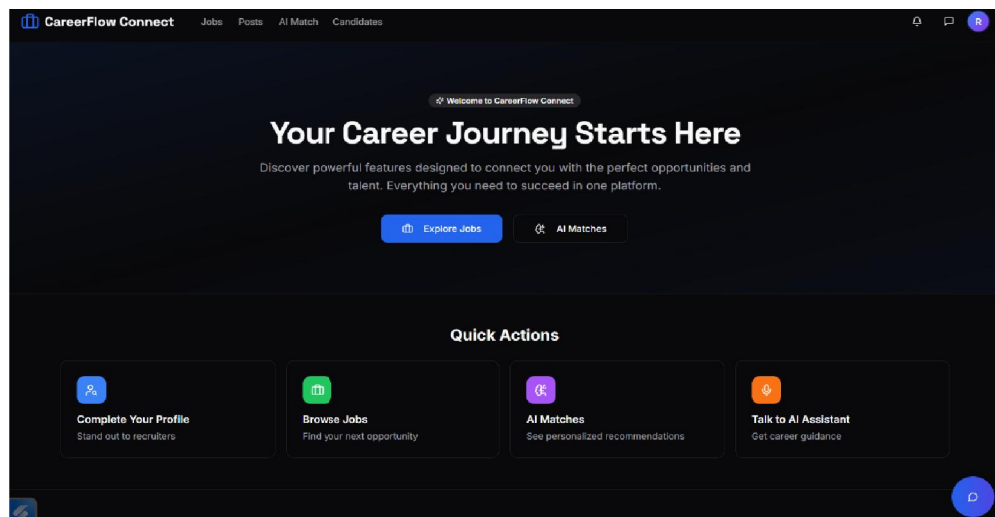
The application was successfully tested in a local development environment. The following key outcomes were observed:

- User Authentication: Secure login and registration functionalities were verified for both students and recruiters.
- Job Feed: Real-time updates of job postings were observed on the user dashboard immediately after submission by the admin.
- Data Integrity: Applicant details were accurately stored in the database and retrieved by the admin without data loss.





a) Landing Page



b) Home Page

## VI. CONCLUSION

The "Job Hiring and Applying Web-App" successfully demonstrates a functional, efficient, and scalable solution for online recruitment. By reducing manual intervention, the system accelerates the hiring process and reduces the administrative burden on recruiters. Future enhancements will focus on AI-driven candidate matching and mobile application support to further improve accessibility. its unique Public Profile Model and Dynamic Salary Valuation, providing crucial transparency for both professional and non-professional sectors. The inclusion of advanced features like Live Location Search and a Multi-Language Chatbot significantly improves accessibility and relevance for diverse user groups. Furthermore, the centralized ATS Kanban Board streamlines the hiring workflow, proving that recruitment can be both efficient and user-centric. This project sets a new benchmark for inclusive and technologically advanced hiring solutions.



**REFERENCES**

- [1]. Firebase Documentation. (Current Version). Official documentation for Firebase Authentication (especially OTP login), Firestore Database, and Storage services. [Cited for: Technical basis for secure passwordless access and scalable cloud architecture.]
- [2]. Next.js & React Documentation. (Current Version). Core documentation defining the high-performance rendering architecture and component model. [Cited for: Frontend architecture and performance standards.]
- [3]. W3C (World Wide Web Consortium). (Ongoing). Standards for Web Accessibility Initiative (WAI) and mobile-first design principles. [Cited for: Core structural and accessibility standards for the UI/UX.]
- [4]. Google Maps Platform / Geolocation API Documentation. (Current Version). Documentation pertaining to interactive mapping and proximity search services. [Cited for: Implementation of the Live Location-Based Search functionality.]
- [5]. Chaudhuri, P., & Ghosh, T. K. (2018). Salary prediction using machine learning techniques. International Journal of Computer Applications. [Cited for: Validation of using ML regression models for the dynamic base salary valuation feature.]

