

INTELLIHIRE: AI-ENCHANCED PERSONAL DATA SHEET PARSING SYSTEM FOR PESO (PUBLIC EMPLOYMENT SERVICE OFFICE)

management.

Purpose and Description

The description serves as a comprehensive overview of IntelliHire, detailing its functionalities, objectives, and key features. It provides with a clear understanding of the system operates and benefits for users in the recruitment process. By defining the system's purpose, this section establishes the foundation for further discussions on its implementation. The IntelliHire project aims to develop an intelligent system, IntelliHire is designed to automate and improve how the Public Employment Service Office also called PESO handles job applications through Personal Data Sheets (PDS) for government positions. PESO is a government office mandated to facilitate employment services, particularly those seeking local job opportunities.

The purpose of IntelliHire is to provide a centralized and streamlined system, an AI-enhanced platform that integrates intelligent parsing tools to read, interpret, and an algorithm to rank Personal Data Sheets submitted by job applicants. Designed specifically for the Public Employment Service Office (PESO), the system automates the traditional manual process of profiling applicants, enabling faster and more accurate sorting of candidate information such as skills, experience, education, and eligibility.

IntelliHire proposes an AI-enhanced solution that uses advanced technologies such as Natural Language Processing (NLP) techniques and Machine Learning (ML) to automatically parse and rank informations that were extracted from a PDS pdf file.

The Natural Language Processing (NLP) tools and machine learning techniques to address common challenges faced by PESO, including data overload, inconsistency in PDS formats, and time-consuming manual screenings. The innovative features of the system empower users to prioritize tasks effectively. The Personal Data Sheets (PDS) submitted by the applicants are automatically rank based on the applicants' skills, experience, educational background, and other qualifications for the job. Beyond automation, IntelliHire's deeper purpose is to enhance the operational capability of the Public Employment Service Office (PESO) by enabling data-driven service delivery. With organized digital records of job applicants, PESO staff can do their jobs better specifically, they can match people to jobs more efficiently and accurately, because the system helps them consider what job seekers are qualified for.

Future developers are recommended to expand the system to cover more HRMO processes. This project will provide a comprehensive solutions that improves the efficiency of applicant short-listing at the Public Employment Service Office (PESO), and will serve as an invaluable reference for those pursuing similar projects in the future.

Objectives of the study

This project aims to develop a Job Portal integrated with an AI-enhanced parsing system for Personal Data Sheets (PDS), specifically designed to assist the Public Employment Service Office (PESO) under the City Government of Sto. Tomas. The system will automate the extraction and organization of critical applicant information, making the shortlisting and matching process faster and more efficient for PESO staff. At the same time, it will provide local job seekers with a more accessible and user-friendly platform to submit applications and connect with employment opportunities.

1. To create a web-base system with the following components:

1.1. Creation, modification and archiving of job postings;

1.2. Uploading of files such as:

1.2.1. Personal Data Sheets Form,

1.2.2. Curriculum Vitae and Resume.

1.3. Data extraction from Personal Data Sheets;

1.4. Matching applicant information with job requirements;

1.4.1. Education: ensures candidates meet educational requirements;

1.4.2. Experience: verifies whether candidates meet experience thresholds.

1.4.3. Eligibility: verifies that applicants possess required certifications and qualifications

1.4.4. Expertise and competency: matches specific skills and competencies..

1.5. Viewing relevant applicant information stored in the database;

1.6. Creating new user accounts; and

1.7. Exporting the list of applicants in a job posting.

2. To facilitate pooling and profiling of applicants based on:

2.1. Job position;

2.2. Relevant education or field of study;

2.3. Work experience; and

2.4. Eligibility.

3. Providing analytics such as:

3.1. Descriptive analytics:

3.1.1. Applicant Qualification Rankings;

3.1.2. (Admin) Dashboards such as:

3.1.2.1. Active and Inactive Job posts;

3.1.2.2. Job Category and Geographical Distribution;

3.1.2.3. Profiling of applicants; and

3.1.2.4. Key metrics of Users and Job applicants:

3.2. Prescriptive analytics;

3.2.1. Job recommendation based on applicants' skills, experience, and expertise.

3.2.2. Candidate Success Prediction; and

3.2.3. Career Path Prediction.

4. Conversion of Personal Data Sheet Template to ATS Resume Format.

4.1. Formatting Standardization; and

4.2. Converts PDS template into ATS-format.

5. Creation of reports

5.1. Automated Report Generation;

5.2. Types of Reports;

5.2.1. Applicant Summary Report; and

5.2.2. Job Application Success Report.

6. Integration with External Platforms;

6.1. Video Interviewing Platform: Integrate with video interviewing platforms like zoom, to facilitate remote interviews directly from the system; and

6.2. Email Marketing Tools: For engagement, reminders, or sending newsletters about open job positions.

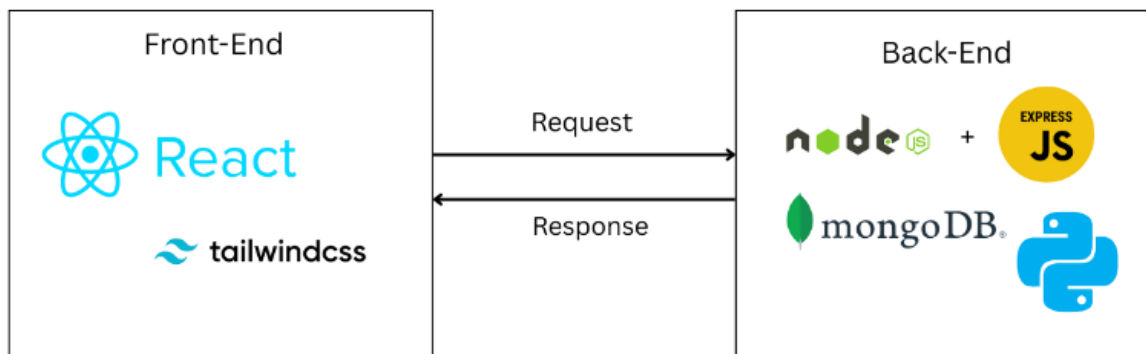
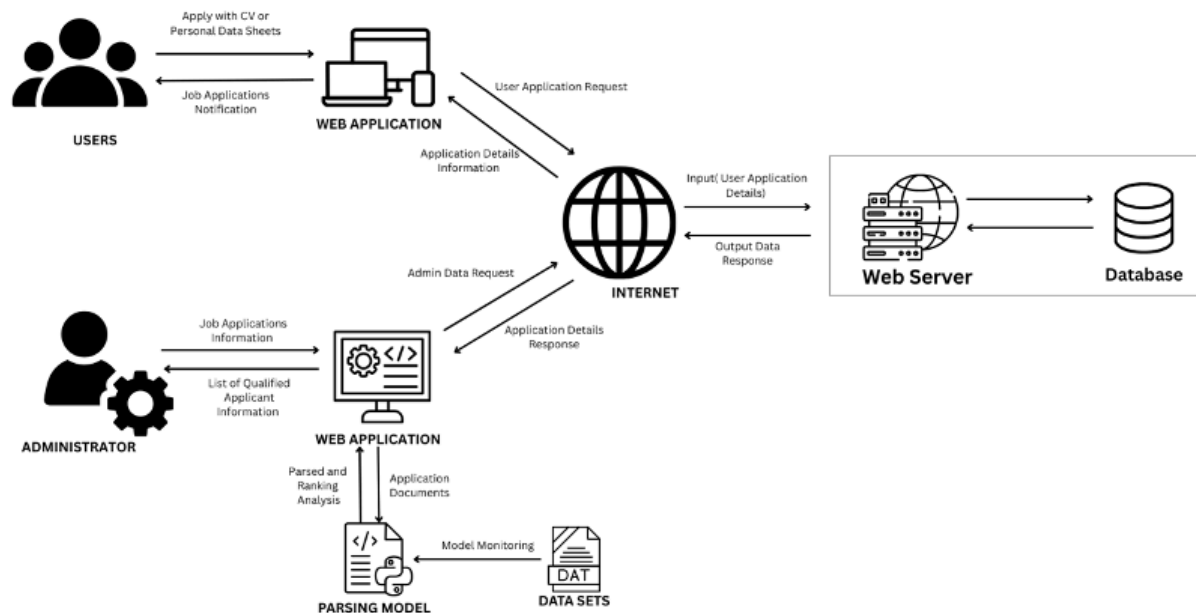


Figure 1. System Architecture Diagram of IntelliHire

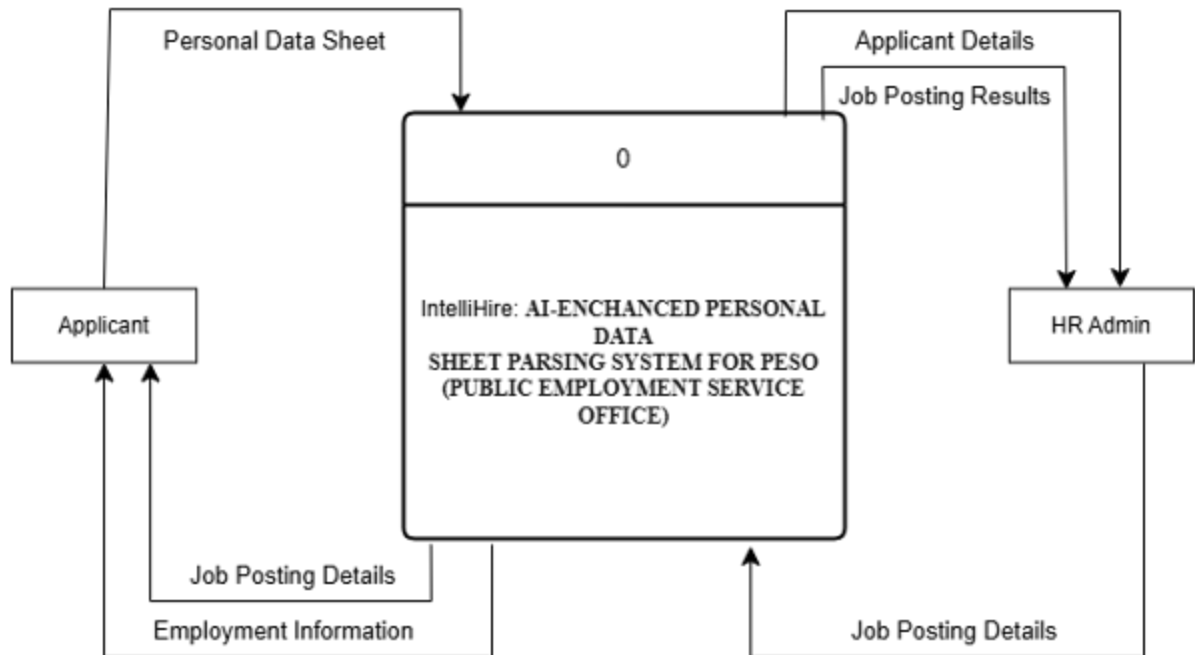


Figure 5. Context Flow Diagram of the Proposed System

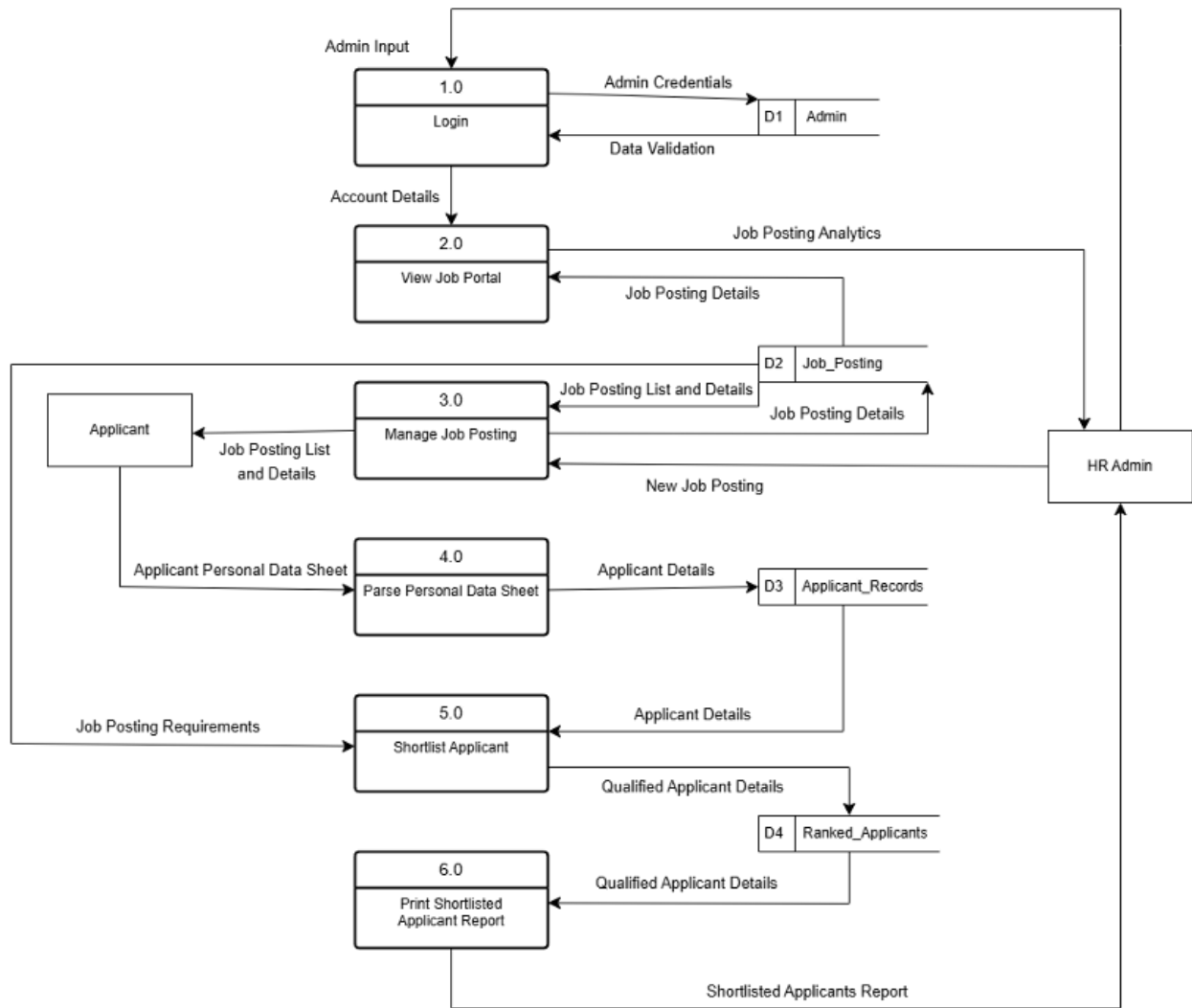


Figure 6. Data Flow Diagram of the Proposed System

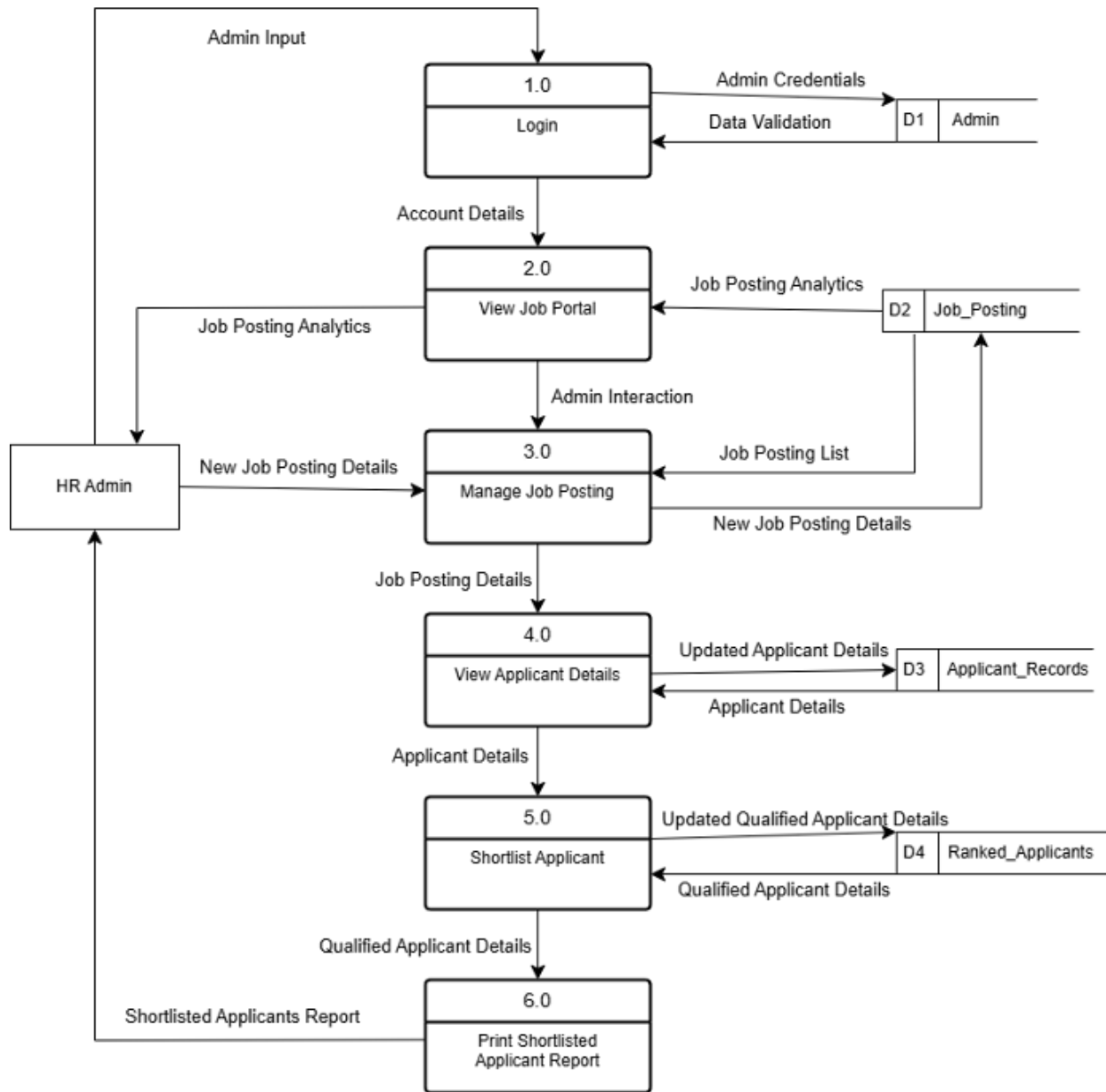


Figure 7. Data Flow Diagram for the HR Administrator

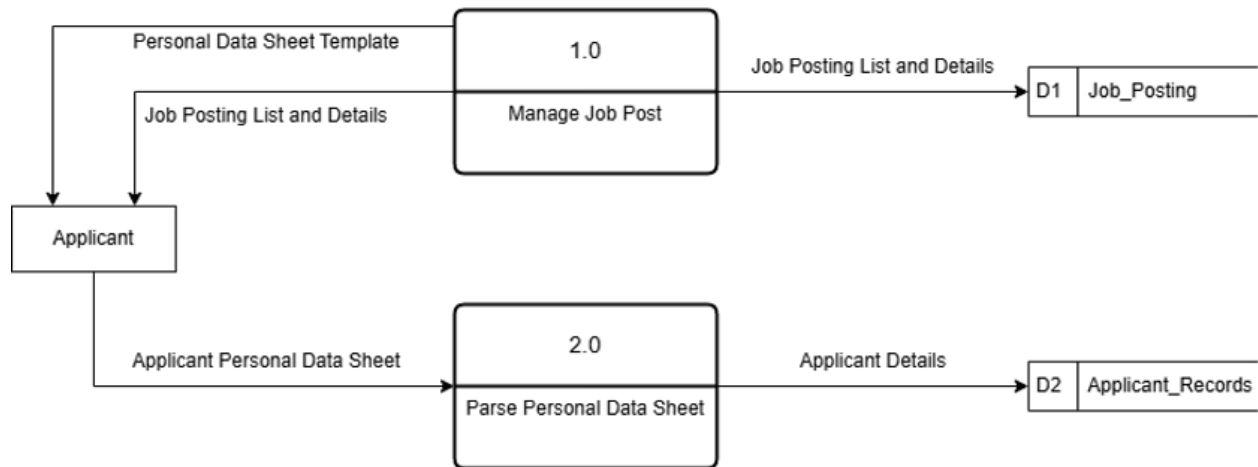


Figure 8. Data Flow Diagram for the Applicant

Table 1.

Functional Requirements

Module	Description
Login/Signup	The system shall be able to have login and sign up for users to input their respective credentials to access the web system.
Dashboard	The system shall be able to display key metrics for administrator account wherein data such as total number of registered users, job postings, users' locations, and top careers.
Job Portal	The system allows the administrator to access the job portal that displays and manages job postings and applicant information.
Job Posting	The system shall allow both administrator and the applicant to view the details of the job posting details such as job title, descriptions, qualifications and application deadlines.

Table 1

Functional Requirements (cont'd)

Download PDS Template	The system allows the applicant to download a template of the personal data sheet.
Upload PDS	The system allows the applicant to upload personal data sheets of applicants on a pdf file format as part of the application process.
Applicant Ranking	The system shall automatically rank applicants based on how well the qualifications, experience and skills match the job description.
Search and Filter	The system shall allow applicants and administrators to search and filter job listings based on the category.
AI Parsing	The system shall parse uploaded PDS PDFs using NLP techniques to extract structure data.
Reports	The system shall generate visual reports for administrators account to provide insights to support provide insights that support decision-making related to job postings, applicant trends, user engagement, and overall system performance.
Newsletter	The system shall allow the administrator to create and send newsletter to registered users.
Banner Management	The system shall allow administrators to upload and manage banner images to display on the homepage section. This feature is intended to highlight important announcements, featured job postings, upcoming events, or system updates, helping to improve user engagement and deliver timely information.
Job Recommendations	The system shall provide personalized job recommendations to applicants based on their profile information.

Table 5.

Software Requirements for the Developers

Equipment	Type/Specification
Operating System	Windows 10 (32/64-bit) or higher; at least 8 GB RAM, at least 2.25 GHz CPU, and at least 20 GB free disk space
Programming IDE	Visual Studio Code, Jupyter Notebook
Frontend Tool	React.js and Tailwind CSS
Backend Tool	Node.js , Express.js ,Python
Database Management	MongoDB
Browser	Google Chrome, Microsoft Edge, Mozilla Firefox, and Opera Web Browser
Parsing and NLP Tools	NLTK, TextBlob and SpaCy

Software Development Tools

To create a modern functional web-based job portal with a Personal Data Sheet parsing feature for PESO under the City Government of Sto. Tomas, the previous developers, utilized various software tools, which were instrumental in achieving the project's goals and ensuring its success for the client.

The MERN architecture allows you to easily construct a three-tier architecture (front end, back end, database) entirely using JavaScript and JSON. On the frontend, React.js is used to create a responsive, interactive user interface (UIs), while Tailwind CSS ensures rapid development and design consistency, minimizing the need for custom CSS.

For the backend, Node.js and Express.js will handle API routes, user authentication, file uploads and communication with the MongoDB database, which stores structured data such as user profiles, job postings, and parsed PDS information.

MongoDB is an ideal choice for our AI-enhanced Job Portal System database due to its flexibility, scalability, and seamless integration with the MERN stack. In contrast to traditional relational databases, MongoDB enables us to store dynamic and complicated data without a set schema, including user profiles, job descriptions, and parsed PDS information. This is extremely helpful when working with AI and Natural Language Processing (NLP) outputs.

To support the system's intelligent parsing feature, Natural Language Processing (NLP) techniques are used to parse and understand content of PDF-based applicants documents such as Personal Data Sheets and Resume/CV. Advanced models such as BERT are used for semantic understanding, enabling the system to accurately extract information like skills, experience, and educational background. This extracted data is then used to automatically rank applicants based on how well their profiles match the job descriptions.

To perform parsing and NLP tasks effectively, the system integrates powerful libraries including NLTK, TextBlob, and SpaCy, which handle tokenization, part-of-speech tagging, named entity recognition, and sentiment analysis. This setup ensures a scalable, intelligent, and maintainable job-matching platform that streamlines recruitment for HR administrators while enhancing the application experience for job seekers.

In our system, the algorithm cosine similarity is used to rank Personal Data Sheets (PDS) or Resume/CV based on how closely they match a given job description. The documents are transformed into numerical vectors. Cosine similarity is calculated between these vectors to determine the degree of similarity, with higher scores indicating a closer match. Based on these scores, the system ranks the documents in descending order, allowing the administrator to quickly identify the most suited candidates for a specific job role.

This allows the system to rank multiple applicants' documents based on how well they match the requirements of a particular job posting. By applying this algorithm, we can prioritize candidates whose qualifications, skills, and experience best fit the role, streamlining the recruitment process.

By leveraging these modern tools and frameworks, the developers created a robust and scalable system that meets the needs of the Human Resource Management Office (HRMO) and provides a solid foundation for future enhancements.