WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

SKILL SYNC ANALYZER – AUTOMATED SKILL MATCHING AND SKILL ENHANCEMENT PLATFORM

Mrs. G.Hemaprabha[1], Vishwa B[2]

Department of Computer Science and Engineering, Bachelor of Engineering, Sri Shakthi Institute of Engineering and Technology, Coimbatore-64106

ABSTRACT—

Skill Sync Analyzer is a smart web-based application that can assess the skills of individuals based on the information available in their resumes and create a learning plan based on the skills that need improvement. It can analyze the skills mentioned in the resumes of individuals, and this data can be compared with the skills required by the company, thus ensuring a very good success rate. Based on the skills deficiency, a structured plan is formulated by giving the candidates suggestions and information on how they can improve. Furthermore, in cases where a good skill match is obtained, a technical test is also given, and a readiness level is determined. Since the application has been developed using a Flask-based backend, it is very efficient and scalable. As a result, it can be used as a developer-friendly application in the context of enhancing knowledge. The application encompasses a good number of characteristics that can be employed as an integrator of a wide variety of important activities.

Keywords— Resume Analysis, Skill Matching, Career Recommendation, Flask Web Application, Skill Gap Analysis, Technical Assessment, AI-based Learning Guidance.

I.INTRODUCTION

The modern job market is very competitive and makes it difficult for aspiring professionals to match their individual skills to the needs of employers in the industry. Many applicants submit resumes to leading companies without being fully aware of whether their technical competencies will meet the expectations of the jobs they are applying for, even though the resumes will contain a list of programming languages, tools, and frameworks. However, there is no organized way to evaluate how those skills fit in with the requirements of the specific companies that they are applying for. As a result, candidates may miss out on job opportunities and waste time preparing for jobs they will not qualify for.

To help solve this problem, we have created a web-based platform called the Skill Sync Analyzer. It is an intelligent platform that analyzes resumes to assess the skill compatibility between candidates and companies and their job requirements. The Skill Sync Analyzer takes the skills listed on an uploaded resume and compares them to established skills for the selected role custom study plan to help the candidate develop the missing skills. This structured methodology changes the way candidates assess their resumes from an unstructured and subjective process to a measurable and actionable one.

LITERATURE SURVEY

The tools for analyzing resumes have also become advanced with the development of artificial intelligence and web technologies. Various frameworks and methodologies have helped in the development of intelligent recruitment and career guidance tools.

Python's Flask framework [1] is a versatile and lightweight framework for web development. Because of its modular architecture and ease of use, it is frequently used for creating scalable backend applications. The backend processing engine of the suggested system, which includes resume parsing, skill extraction, score computation, and routing between application modules, is implemented using Flask. Large-scale deployment may necessitate additional extensions for advanced scalability and load balancing, despite Flask's great flexibility and ease of integration.

Significant usage of Natural Language Processing (NLP) techniques has been witnessed in the systems developed for skill extraction and resume parsing techniques [2]. Tokenization, removal of stop words, and keyword matching are examples of NLP techniques that enable the extraction of relevant information from unstructured documents

in a structured manner. The suggested system utilizes keyword normalization and rule-based text preprocessing techniques for the extraction of technical skills from resumes. However, semantic similarity between similar skill sets (e.g., "JS" and "JavaScript") is not possible with a purely rule-based approach.

Applicant Tracking Systems (ATS) [3] typically employ keyword filtering mechanisms to screen resumes. These systems compare keywords in a resume with job descriptions to find the suitability of candidates. The idea of set-based matching employed in the proposed system is inspired by the filtering logic of traditional ATS systems. However, unlike traditional systems, which provide feedback to recruiters, the proposed system extends the idea to provide skill gap analysis and study recommendations to candidates.

JSON (JavaScript Object Notation) [4] is a lightweight data interchange format that can be used for data storage in the form of structured information. In the proposed system, the skill requirements for the roles of the companies and the study plans will be stored using JSON-based data sets. Although JSON offers a simple data storage solution, handling huge amounts of data might require database integration for better performance.

Session management techniques for web applications [5] enable the temporary storage of user-specific information during multi-step processes. Session management is utilized for storing the extracted skills, match scores, missing skills, quiz responses, and readiness scores during the workflow in the Skill Sync Analyzer. However, improper management may cause memory inefficiencies.

Skill gap analysis models [6] are generally employed in research studies focusing on education and workforce development. The skill gap analysis is utilized for identifying the competency gap between the present skill level and the required skill level. The competency gap is generally calculated by finding the difference between the expected and actual competency levels. The proposed system utilizes the skill gap analysis concept by finding the difference between the required job role skills and the extracted skills.

Online technical assessment systems [7] are commonly used by companies to measure the readiness of candidates before the recruitment process. These online assessment systems have scoring and evaluation mechanisms. The quiz engine of the Skill Sync Analyzer is inspired by online technical assessment systems, which allow the evaluation of answers and scoring of readiness. Though this is effective for objective questions, subjective evaluation of technical depth using AI-based scoring is complex.

Similarity models using set theory [8] provide a theoretical framework to measure the similarity between two sets of objects. In this context, the skill matching algorithm uses set theory to find the intersection and difference of sets, which represent the matched and missing skills. This is a clear and efficient way to measure compatibility scores. However, this does not support the scoring of weights of skills, which is a complex task.

The principles of web usability, as discussed by Nielsen [9], focus on the importance of clarity, responsiveness, and simplicity of the user interface. In the case of the frontend of the Skill Sync Analyzer, the page flow and the use of a minimalist approach to dashboard visualization are aimed at improving the overall usability and user experience. Ineffective user interface can result in a significant decrease in the overall adoption rate, even if the backend processing is effective.

Shannon's research on the fundamental principles of information theory, as discussed in [10], highlights the importance of the use of structured information representation and the minimization of ambiguity. Although not related to the analysis of resumes, the use of the concept of structured information transformation can be applied to the overall approach to the analysis of unstructured resumes.

SYSTEM ARCHITECTURE AND DESIGN

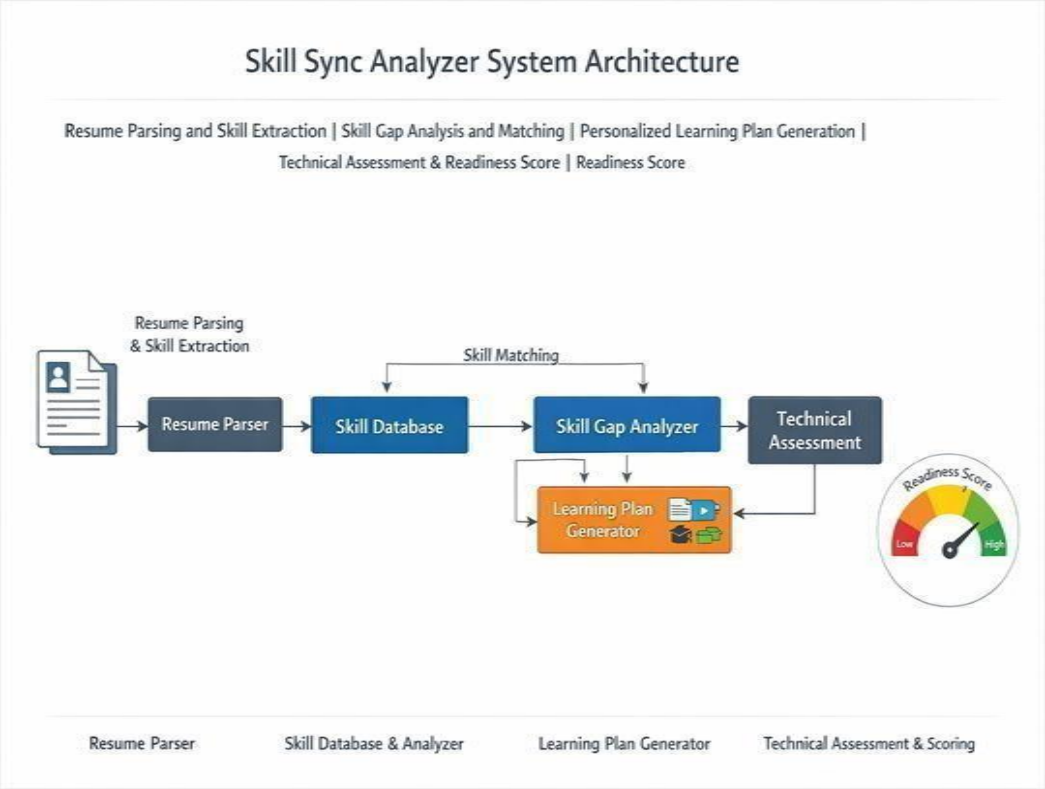
The Skill Sync Analyzer application employs a client-server architecture that clearly separates the two major components of the application: the user interface layer where user interaction takes place, and the back-end processing layer (including resume parsing,
extracting skills from resumes, comparing those skills to the company/job skill requirements, calculating matching scores, identifying skills that are missing from an applicant's skill set, generating a recommended study plan based on missing skills, and scoring quizzes taken by users). The front-end functionality handles various user interactions: uploading a resume, selecting a company and job position, displaying skill-matching results, and taking quizzes. The front end also communicates with the back end via HTTP requests, enabling secure and well-structured data exchange. This separation enhances modularity, allowing for easier maintenance, scalability and upgrades to the application codebase and any of its components.
The processing engine of the Skill Sync Analyzer is built with Flask (Python) and is responsible for most of the application's data processing. In addition to resume parsing and skill extraction, the back end module is responsible for comparing the user's skill set against the company/job skill requirements, calculating a match score between the two skill sets, identifying which skills are missing from the user's skill set, generating a recommended study plan based on the missing skills, and scoring user quizzes taken through the application. In addition, skill requirements and recommended study plans for specific companies/jobs are stored in JSON-based data files; session management, which allows for temporarily storing user-specific data such as downloaded skills, calculation of match scores, and user quiz scores, is used to provide a smooth multi-step user experience across various pages of the application.
The overall design of the Skill Sync Analyzer follows a modular structure so that each of the four primary modules (resume parser, skill matcher, study plan generator, and quiz engine) operates independently from the others.

ARCHITECTURE:

Skill Sync Analyzer follows a modular client–server architecture where the frontend manages user interaction and the Flask backend performs resume parsing, skill matching, and analysis. The system integrates a skill database, gap analyzer, study plan generator, and technical assessment module to produce a final career readiness score.

The system uses JSON-based datasets to store company-role skill requirements and predefined study plans. When a resume is uploaded, the backend extracts skills and compares them with the required skills using set-based matching logic. Based on this comparison, it calculates the match percentage and identifies missing skills. If gaps are found, the backend dynamically generates personalized study plans to guide the user in improving their competencies.

Session management is implemented to temporarily store user-specific data such as extracted skills, match score, missing skills, quiz answers, and final results. This allows seamless navigation between pages without losing analysis data. Additionally, the backend includes error handling mechanisms for invalid file uploads, missing inputs, and internal server errors, ensuring system reliability and stability.



IMPLEMENTATION

A. Frontend

The frontend of the Skill Sync Analyzer is designed to provide a simple, interactive, and user-friendly interface for resume analysis and career readiness evaluation. It is built using standard web technologies such as HTML, CSS, and JavaScript, ensuring compatibility across different browsers and devices. The interface allows users to upload resumes, select a target company and job role, and navigate seamlessly through results, study plans, and assessment sections. The design focuses on clarity and usability to make the analysis process intuitive even for first-time users.

The application follows a structured page flow where each stage of the analysis is visually separated. The homepage provides company and role selection along with a resume upload option. After submission, users are redirected to the results page, which displays the skill match percentage, missing skills, and personalized study plans. If all required skills are matched, the interface automatically redirects to the technical assessment page. This controlled navigation ensures a smooth multi-step workflow without overwhelming.

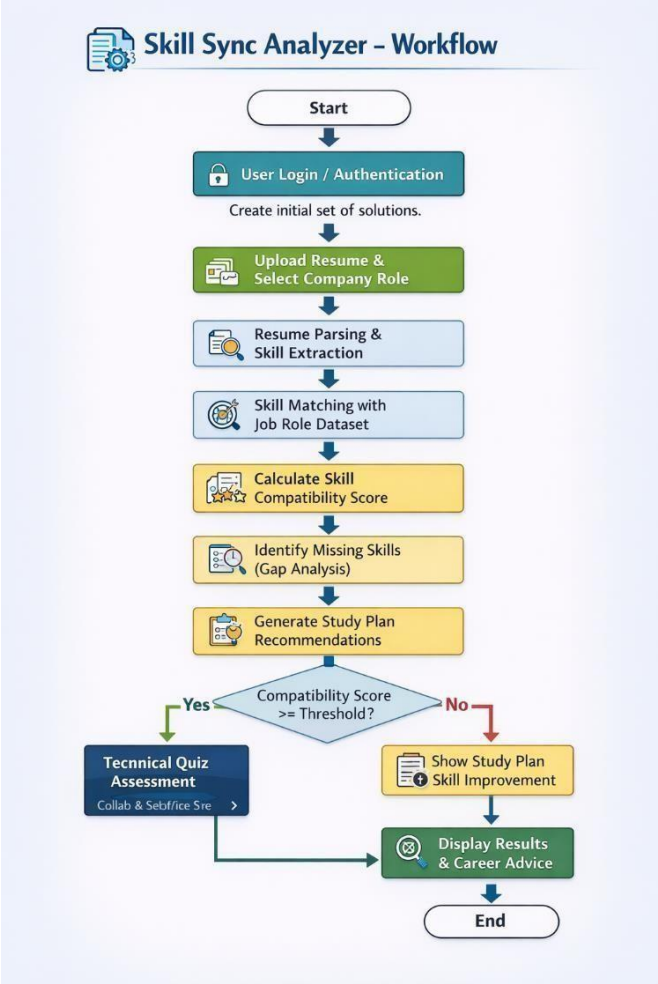
The frontend also dynamically displays analytical data such as skill comparison charts, readiness scores, and quiz results. It communicates with the Flask backend. Flash messages and error notifications are implemented to guide users in case of invalid uploads or missing selections. Overall, the frontend ensures responsiveness, clarity, and efficient interaction between user input and backend processing.

- User-friendly interface for resume upload and company-role selection.
- Dynamic dashboard displaying match score, missing skills, and study plans.
- Interactive quiz page with automatic scoring and readiness score visualization.

B. Backend

The backend of Skill Sync Analyzer is developed using Flask (Python) and serves as the core processing unit of the system. It handles resume parsing, skill extraction, skill comparison, and score calculation. The backend manages all business logic, ensuring that user inputs are processed securely and accurately. It also controls routing between different pages such as results, quiz, and final score, maintaining a smooth workflow throughout the application

The system uses JSON-based datasets to store company-role skill requirements and predefined study plans. When a resume is uploaded, the backend extracts skills and compares them with the required skills using set-based matching logic. Based on this comparison, it calculates the match percentage and identifies missing skills. If gaps are found, the backend dynamically generates personalized study plans to guide the user in improving their competencies.



Session management is implemented to temporarily store user-specific data such as extracted skills, match score, missing skills, quiz answers, and final results. This allows seamless navigation between pages without losing analysis data. Additionally, the backend includes error handling mechanisms for invalid file uploads, missing inputs, and internal server errors, ensuring system reliability and stability.

- Resume parsing and skill matching algorithm
- Study plan generation and company-specific quiz evaluation
- Session management with structured routing and error handling.

AIand Relevance Engine

The AI-Based Skill Matching and Readiness Engine is the intelligence core of the proposed Skill Sync Analyzer platform. It is responsible for analyzing resumes, identifying skill relevance, computing compatibility scores, detecting skill gaps, and generating readiness evaluations. The engine operates using a structured rule-based and weighted scoring mechanism to ensure transparent and measurable compatibility assessment between candidate skills and company-specific job requirements.

The system processes uploaded resumes, extracts structured skill data, compares it with predefined job-role skill datasets, and ranks compatibility

based on multiple weighted parameters. Only after computing the compatibility and readiness thresholds does the system proceed to generate study recommendations or technical assessments. The architecture of the engine is modular and can be extended with advanced machine learning techniques in future enhancements.

A. Skill Matching Mechanism

Skill matching is the primary function of the AI-based engine, which enables resume–job compatibility evaluation. Job-role requirements are defined using structured skill datasets categorized into domains such as programming languages, frameworks, databases, tools, and soft skills. When a resume is uploaded, the extracted skills are normalized and compared against the required skill set.

Each required skill is assigned a weight based on its importance to the role. Core skills receive higher weights compared to secondary skills. The matching score increases proportionally with the number and importance of matched skills.

In the enhanced version of the system, skill terms and resume content can be converted into vector representations using keyword embeddings or semantic similarity models. Cosine similarity can then be computed between the resume vector and job-role vector to improve contextual relevance detection.

The compatibility score is computed as:

This ensures transparent, interpretable, and structured compatibility scoring.

B. Skill Gap Identification and Knowledge Linking

The Skill Gap Identification module detects missing competencies by computing the difference between required and extracted skill sets. These missing skills are mapped to predefined learning resources stored in structured datasets.

Each missing skill is linked to recommended learning topics, practice exercises, and improvement suggestions. In an advanced version, a knowledge graph structure can be implemented to map dependencies between foundational and advanced skills. For example, understanding "Data Structures" may be considered a prerequisite for mastering "Algorithms."

This structured linking mechanism ensures that candidates receive logically sequenced learning guidance rather than isolated skill recommendations.

C. Multi-Dimensional Readiness Scoring Algorithm

The readiness scoring algorithm evaluates candidates using a multi-dimensional weighted approach to ensure comprehensive assessment. It considers:

- Skill compatibility score
- Technical quiz performance
- Core skill weight alignment
- Skill gap severity

The composite readiness score is calculated as:

Only candidates exceeding a predefined readiness threshold are considered highly compatible for the selected role. This ensures structured and reliable evaluation.

The module is designed to be modular and extensible. The current rule-based scoring system can be enhanced in future versions using machine learning models such as skill classification models, semantic similarity models, or predictive hiring compatibility systems.

D. Skill Prioritization Using Modified PERT-Based Learning Model

To provide structured improvement guidance, the system applies a modified PERT (Program Evaluation and Review Technique) model for prioritizing skill acquisition. Skills are categorized based on:

- Importance to job role
- Frequency of industry demand
- Dependency on foundational knowledge
- Estimated learning complexity

For each skill, three priority values are assigned:

- Optimistic priority (O)
- Most likely priority (M)
- Pessimistic priority (P)

The expected priority score is calculated as:

Dependencies between foundational and advanced skills ensure that prerequisite knowledge is prioritized before complex topics. The resulting priority network determines the optimal learning sequence for candidates, enabling structured and high-impact skill development.

E. Resume Content Scoring with Career Context To improve resume evaluation accuracy, the system applies contextual scoring at the skill and content level. Each extracted segment is evaluated based on relevance to job-role requirements.

The contextual skill scoring formula is defined as:

This ensures that redundant repetitions of similar skills do not artificially increase compatibility scores, maintaining fairness and accuracy in evaluation. Overall, the AI-Based Skill Matching and Readiness Engine transforms unstructured resume data into structured compatibility insights using weighted scoring, dependency modeling, redundancy control, and readiness evaluation. Its modular design enables seamless integration of future AI-driven enhancements, making Skill Sync Analyzer a scalable and intelligent career assessment platform.

VII. CORE FEATURES AND WORKFLOW

A. User Authentication and Personalization

User authentication and personalization form the foundational components of the Skill Sync Analyzer platform. The system is designed to provide individualized career readiness evaluation while ensuring data confidentiality and structured user management. Users are required to create an account and authenticate their credentials before accessing the full functionality of the platform. Upon successful authentication, the system verifies the user's identity and grants secure access to personalized features such as resume history, preferred job domains, previously attempted assessments, and stored readiness scores. This ensures that each user's data remains private and accessible only within their secured session environment.

Once authenticated, the platform retrieves the user profile, which includes essential information such as preferred job domain (e.g., Web Development, Data Science, Software Engineering), experience level (beginner, intermediate, advanced), and targeted company type (product-based, service-based, startup). These preferences serve as the foundation for personalization across the platform. The user can modify these preferences at any time through the profile management interface, allowing dynamic adaptation of recommendations.

Based on this profile information, the system personalizes the resume analysis process by aligning compatibility evaluation with the selected job domain and role. Skill matching, study plan generation, and technical assessments are tailored according to the user's chosen career objective. Additionally, the readiness scoring algorithm gives higher priority to core skills relevant to the selected domain. Overall, the authentication and personalization framework enhances security, user convenience, and analytical accuracy, ensuring that career recommendations remain aligned with individual professional goals.

B. Resume Analysis and Intelligent Skill Matching

The Resume Analysis and Intelligent Skill Matching module is a core backend service that processes uploaded resumes and evaluates job compatibility. When a user uploads a resume and selects a company-role combination, the backend extracts textual content and identifies technical skills using structured parsing and keyword normalization techniques.

The extracted skills are compared against predefined job-role skill datasets that are structured according to industry standards and company requirements. Each skill in the dataset is assigned a weight based on its importance to the role. Core competencies such as programming languages or domain-specific frameworks receive higher weights, while supplementary tools receive moderate weights. The system calculates a compatibility score based on weighted matching and identifies missing skills.

In addition to direct keyword matching, the engine can incorporate contextual similarity measures in advanced versions. Skill clusters and related competencies can be grouped to improve semantic understanding. For example, familiarity with "React" may partially contribute to "Frontend Development" skill alignment.

All computed metadata, including matched skills, missing skills, compatibility percentage, and core-skill alignment indicators, are structured into a result object and transmitted to the frontend for visualization. This ensures transparent evaluation and structured feedback for the user.

C. Personalized Study Recommendations and Assessment Workflow

To enhance career preparedness, the platform provides structured study recommendations based on identified skill gaps.

When missing skills are detected, the system dynamically generates a personalized learning roadmap. Each missing competency is linked to predefined improvement resources, including recommended topics, foundational prerequisites, and practice suggestions.

The study plan is organized logically, ensuring that prerequisite skills are learned before advanced concepts. For example, mastering "Data Structures" may be recommended before "Algorithm Optimization." A prioritization mechanism ensures that high-impact skills are addressed first, enabling efficient learning progression.

If the compatibility score exceeds a predefined threshold, the user is directed to a technical assessment module. This module consists of multiple-choice questions aligned with the selected job role. The quiz engine evaluates responses automatically and calculates a technical performance score. Upon completion, the system combines the compatibility score and quiz performance to generate a final readiness score, providing a comprehensive evaluation of both skill alignment and technical proficiency.

This structured workflow ensures that the platform not only identifies gaps but also guides candidates toward measurable improvement and readiness validation.

D. Accessibility and User-Centric Design

The Skill Sync Analyzer platform is developed with strong emphasis on accessibility, usability, and inclusivity. The user interface follows responsive design principles, enabling seamless access across desktops, tablets, and mobile devices. Clear typography, minimal visual clutter, and structured data presentation ensure that analytical results remain easily interpretable.

Interactive dashboards display compatibility metrics, missing skills, and readiness scores in a visually organized format. Immediate validation feedback prevents submission errors and improves interaction efficiency. The structured navigation flow ensures that users can move smoothly between resume upload, analysis dashboard, study plan recommendations, and technical assessment modules without confusion.

Additionally, the modular design of the system ensures adaptability to various learning styles and technical backgrounds. By combining structured analytics with clear visualization and guided workflows, the platform enhances user engagement and promotes informed career decision-making.

Overall, the Core Features and Workflow of Skill Sync Analyzer establish a secure, personalized, intelligent, and user-centric career readiness evaluation platform. The integration of authentication, skill matching, gap analysis, structured learning guidance, and readiness scoring creates a comprehensive ecosystem for measurable professional development.

VI.IMPLEMENTATION AND PRELIMINARY TESTING

The project has been developed and implemented as a fully integrated full-stack web application, meaning that all the key parts of the application, from the frontend interface, the backend processing engine, to the supporting data modules, are fully integrated to function in a seamless manner. In its current state, the system is fully functional and provides a good foundation for further development, optimization, and deployment in real-world career assessment contexts.

The frontend part of the application is implemented using standard web programming technologies, such as HTML, CSS, and JavaScript, and provides an interactive and responsive interface for users to upload their resumes and select their target company roles. Although the current implementation uses predefined data for skill requirements and study plans, the frontend part is already designed to communicate efficiently with the backend part using HTTP requests, ensuring full integration and smooth interaction between the frontend and backend parts.

The backend server is implemented using the Flask web framework in Python, which is used to implement the core analytical functionalities of the system. This includes resume parsing, skill detection, compatibility score calculation, gap detection, study plans, and technical quiz evaluation. When the user uploads the resume and selects the company role combination, the backend processes the resume text and detects the technical skills using various preprocessing techniques. It then matches the detected skills with the predefined skill sets for the selected company role combinations. The backend has been implemented and tested to ensure the accurate processing of user requests, compatibility analysis, and generation of learning recommendations.

The data storage is implemented using the JSON dataset format for storing the required skills for different company roles with their study plans. This is a lightweight data storage approach that can be easily modified and updated without changing the core logic of the system. Session management is used to store user-specific information during the multiple-step process of the system, including the detected skills, compatibility results, gaps identified, and quiz results. This is used to provide a seamless user experience throughout the analysis process.

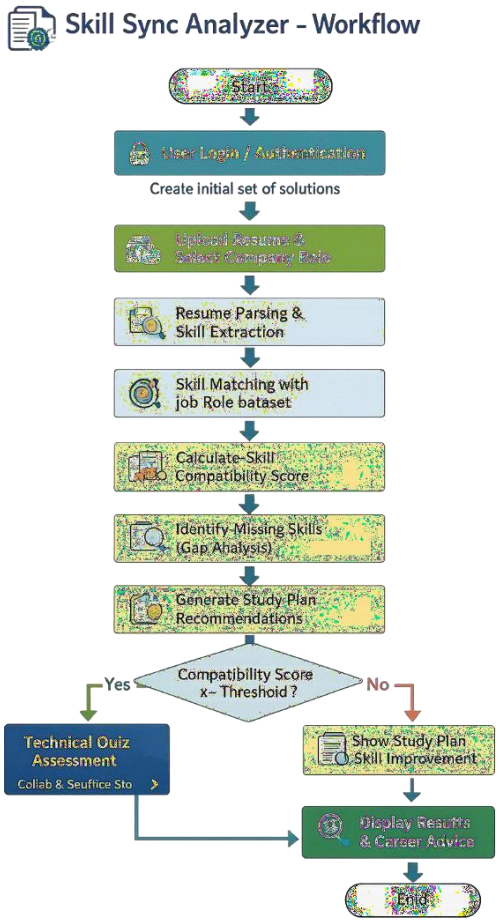
From the perspective of development and system design, the project has adhered to the principles of structured programming and modular design. Each module, such as the resume parser, skill matching engine, study plan generator, and quiz engine, functions independently while being part of the larger system workflow.

The project has been successfully developed as an integrated application, and the current version is complete and ready to be extended.

- **Frontend:** This part of the project has been designed to provide an intuitive user interface that can be used to upload the resume, select the company and role, and display the results such as the compatibility level, skills to be acquired, and study recommendations.
- **Backend:** This part has been implemented using the Flask framework to perform the resume parsing, skill matching, and technical quiz scoring.
- **Data Storage:** The job-role skill requirements and study recommendations have been implemented using structured JSON datasets.
- **System Modules:** The application has modules such as resume analysis, skill matching, gap detection, study plan generation, and readiness scoring.

Initial tests confirm that all primary system functionality, such as the upload of resumes, skill set extraction, compatibility score calculation, detection of missing skills, study plan generation, and technical assessment, are functioning as expected. The system is able to parse resumes, identify skill gaps in the user's profile, and provide the user with structured learning plans. The system's modular nature allows for the future extension of the system's individual components, such as the integration of more advanced artificial intelligence-based skill set matching algorithms, database systems, and cloud deployment technologies.

The diagram below presents the operational workflow of the Skill Sync Analyzer system. The operational workflow begins with user identification and the upload of resumes and company roles. The system then parses the resumes and compares the user's skill set with the predefined company role datasets and generates a compatibility score. It then determines the user's skill gaps and generates a study plan. If the compatibility score is higher than the threshold value, the user is then directed to the technical assessment module, where the user's performance in quizzes is assessed. The system then generates a readiness score and provides the user with structured feedback and career improvement recommendations.



CONCLUSION AND FUTURE WORK :

The Skill Sync Analyzer represents a structured and effective implementation of intelligent web technologies and rule-based analytical mechanisms to address a critical challenge in modern recruitment and career development. By transforming unstructured resume data into measurable compatibility insights, the platform reduces ambiguity in resume evaluation and provides candidates with clear, actionable feedback. The integration of weighted skill matching, gap identification, personalized study recommendations, and readiness assessment creates a comprehensive ecosystem for structured professional growth.

One of the key contributions of the platform is its ability to convert subjective resume screening into a transparent, data-driven evaluation process. Instead of merely indicating acceptance or rejection, the system identifies specific missing competencies and guides candidates toward improvement. The technical assessment module further enhances evaluation reliability by validating theoretical knowledge and practical readiness. Additionally, the responsive and user-centric design ensures accessibility across devices, promoting usability and engagement.

The proposed system also provides a strong foundation for future enhancements that can significantly increase its intelligence and scalability. Advanced Natural Language Processing (NLP) techniques can replace rule-based skill extraction with semantic analysis models capable of understanding contextual meaning and synonyms. Machine learning-based compatibility prediction models can be introduced to analyze historical hiring data and predict success probability for specific roles. Integration of transformer-based language models can further enhance resume summarization and skill inference capabilities.

Future work may also include the integration of real-time job portal APIs, enabling automatic comparison between resumes and live job descriptions. A cloud-based deployment architecture can be implemented to support large-scale real-world usage and concurrent users.

Additional features such as peer benchmarking dashboards, collaborative career planning tools, mentorship integration, and adaptive learning systems can further strengthen the platform. The introduction of AI-driven question generation for role-based assessments and performance analytics dashboards can enhance personalized career development strategies.

Overall, Skill Sync Analyzer establishes a robust framework for intelligent, adaptive, and personalized career readiness evaluation. The system contributes meaningfully to the advancement of AI-assisted recruitment support tools and lays a scalable foundation for future research and industrial deployment in the field of employability enhancement.

ACKNOWLEDGEMENT

The development of the Skill Sync Analyzer system has been facilitated by the availability of various open-source technologies and tools. The authors would like to thank the communities that have contributed to the development and availability of Python, Flask, HTML, CSS, and JavaScript, which are the essential tools required to complete this project.

REFERENCES

- Flask Documentation, "Flask Web Framework," 2023.[Online].
  Available: https://flask.palletsprojects.com/
- S. Bird, E. Klein, and E. Loper, Natural Language Processing with Python. Sebastopol, CA, USA: O'Reilly Media, 2009.
- Jobscan, "Applicant Tracking Systems (ATS): Overview and Functionality," 2023. [Online]. Available:
  https://www.jobscan.co/blog/applicant-tracking-systems/
- JSON.org, "Introducing JSON (JavaScript Object Notation)," 2023. [Online]. Available: https://www.json.org/
- Mozilla Developer Network (MDN), "Session Management in Web Applications," 2023. [Online].
  Available: https://developer.mozilla.org/
- World Economic Forum, "Skill Gap Analysis and Workforce Development Frameworks," 2022.
- HackerRank, "Technical Assessment Platforms and Skill Evaluation Systems Review," 2023. [Online]. Available:
  https://www.hackerrank.com/
- R. Diestel, Graph Theory, 5th ed. Berlin, Germany: Springer, 2017.
- J. Nielsen, Usability Engineering. San Francisco, CA, USA: Morgan Kaufmann, 1994.
- E. Shannon, "A Mathematical Theory of Communication," Bell System Technical Journal, vol. 27, no. 3, pp. 379–423, Jul. 1948.