International Journal of Web of Multidisciplinary Studies
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
website: *https://ijwos.com* OPEN ACCESS E-ISSN : 3049-2424
Volume 03, Issue 10,October 2026 DOI: https://doi.org/10.71366/ijwos

Intelligent Recommendation System For Individual Development Plans

Prof. SHRADHA.HANABARATTI*[1], GANESH KARADIGUDA *[2], SACHIN KODIKURUBAR*[3], MADHUSHREE DANARADDI*[4], SOUMYA GUDDIMANI*[5]
[1,2,3,4,5] *Department of Artificial Intelligence and Data Science, Angadi Institute of Technology and Management, Belagavi, India*

Article Info

Article History:
Published: 30 Oct 2026

Publication Issue:
Volume 03, Issue 10
October 2026

Page Number:
524-538

Corresponding Author:
Prof. SHRADHA.HANABARATTI

Abstract:
Intelligent recommendation systems represent highly evolved frameworks based on artificial intelligence technologies. An intelligent recommendation system uses artificial intelligence and analyzes personal data to generate individual recommendations. The operation of an intelligent recommendation system occurs via a process involving data gathering, preprocessing, feature extraction, model selection, training, optimization, and performance analysis. Using different machine learning algorithms, including collaborative filtering, content-based filtering, and hybrids, an intelligent recommendation system detects significant associations between users and items to provide high-quality recommendations. Incorporation of keyword extraction, topic intelligence, and automatic summarization further increases the contextual awareness of the system to improve its recommendations. Performance analysis guarantees accuracy, reliability, scalability, and efficient processing of data. Adaptive mechanisms for continuous learning and optimization increase the efficiency of the recommendation process. In addition, the intelligent recommendation system analyzes multi-dimensional data related to users, including their profiles, skills, educational information, career goals, and feedback history. Preprocessing and feature engineering are used to improve the quality of data and make the model more efficient.

Keywords: Intelligent Recommendation System, Individual Development Plan, Machine Learning, Skill Gap Analysis, Personalized Learning, Career Development.

1. INTRODUCTION

In the present days of information economy, learning and development have become critical for students and working individuals since fast technological progress entails changes in the work environment and requires adaptation. That is why career planning and personal development have gained much importance recently.

Individual Development Plan (IDP) is an instrument that helps people to plan their personal and professional development. It allows individuals to define their strong and weak sides, career goals, and the necessary skills to achieve success. Generally, such plans are drawn up manually with the assistance of mentors, professors, or career coaches.

Today intelligent recommendation systems represent a promising technology to provide users with personalized recommendations based on their behavior and preferences. Such systems are applied in different fields such as commerce, e-learning, social networking, and entertainment. Therefore, implementing this technology into career development makes it possible to create intelligent recommendation systems that generate recommendations on learning path and development strategy of each individual.

The current project proposal is dedicated to creation of an Intelligent Recommendation System for Individual Development Plans which analyzes users' data (skills, interests, experience, and career goals) in order to generate personalized development plans using recommendation algorithms and data analytics.

Copyright © 2026 The Author(s) : This is an open access article under the CC BY license (http://creativecommons.org/licenses/by-nc/4.0/)

1.1 Problem Statement

Professional development and skill building are vital in career success. Nonetheless, there is a problem among many people in identifying skills to be acquired and courses to undertake. Current career advisory systems give advice that is generic and does not take into account one's abilities and interests.

1.2 Aim and Objectives

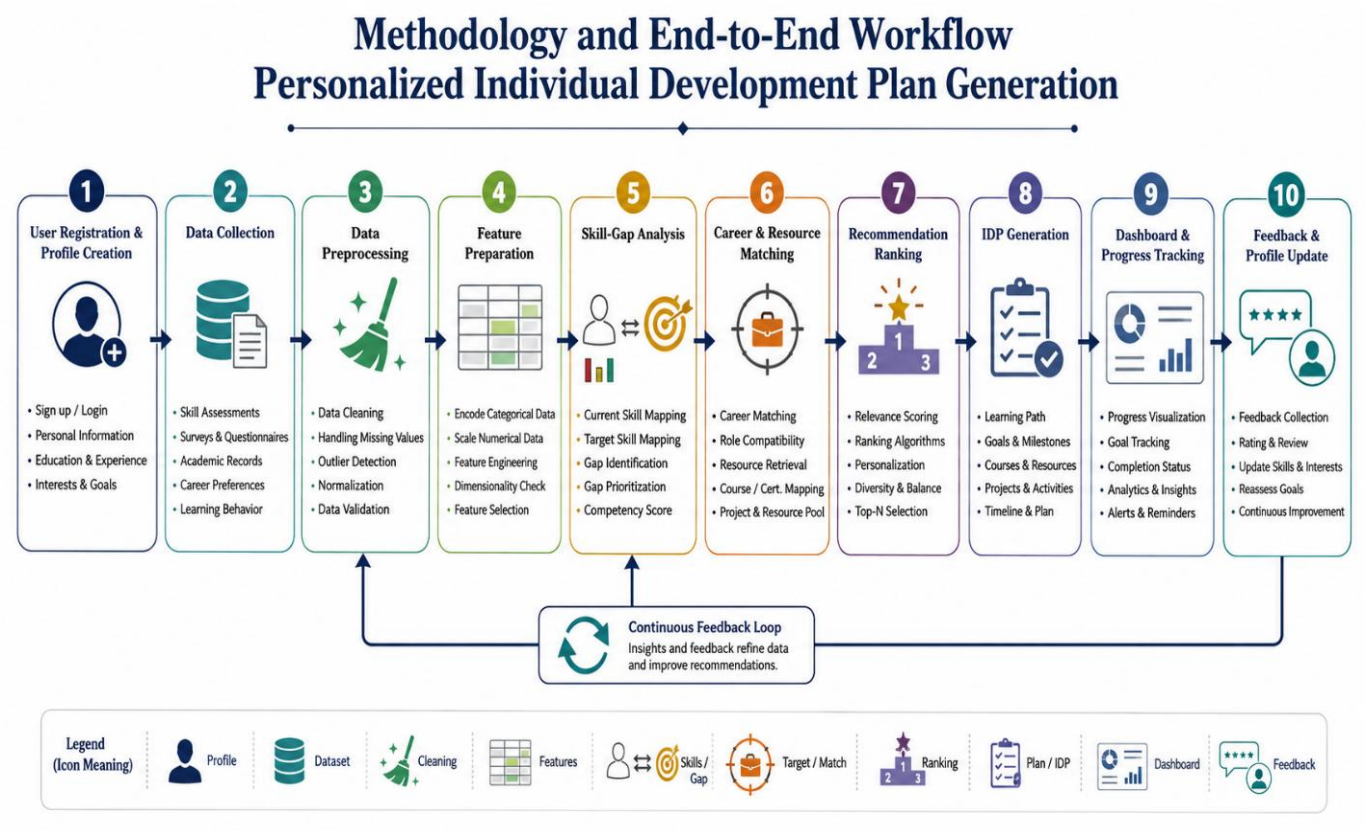
This objective is to design an intelligent system that can analyze the skill set, interest, education, experience, and career aspirations of users and come up with customized Individual Development Plans for them.

- Analyze user skill set, interests, and objectives.
- For coming up with customized development plans.
- Recommend relevant courses, certificates, and skills.
- Track progress of the user and revise recommendations accordingly.
- Enhance decision-making capability in career and personal development.
- Performance monitoring of the user through analytical dashboards.

1.3 Scope of the Project

The scope includes profile management, skill assessment, career advice, skill gap identification, resource recommendations, IDP creation, tracking, and administration management. The design has web access in mind and will eventually facilitate mobile application, multi-language support, and more sophisticated modeling and data sets.

1.4 Proposed Methodology



The process starts with the registration and profiling stage. User data is gathered and processed for further analysis. Data cleansing and transformation of the career recommendation dataset is performed. Gaps between the current skill set and the one needed by the target career are found. After that, candidate resources are matched against gaps and user preferences. Finally, ranked resources are structured in an Individual Development Plan.

Fig. 1. Proposed methodology for personalized Individual Development Plan generation.

1.5 Technology Overview

Technology	**Purpose**
HTML/CSS/JavaScript	Web structure, styling and interaction
React	Component-based interface
Python	Backend and recommendation logic
Flask/Django	Application/API layer
MySQL	Structured data storage
MongoDB	Flexible document storage
Pandas/NumPy	Data processing
Scikit-learn	ML and similarity operations
Matplotlib	Visualization
Localhost	Development and testing

2. RELATED WORK

2.1 Recommendation Systems

The recommendation system is one that selects items which are relevant to a particular set of users. This can be achieved using content based, collaborative filtering or a combination of the two. These concepts will be used in this project to develop personalized learning and career recommendation.

2.2 Personalized Learning Recommendation

Recommendation learning algorithms have been employed through learner profiles, past experience, interests, courses metadata, and skills. The current system has applied this strategy for career development through target career and current skill data.

2.3 Skill-Gap Analysis

The skill gap analysis method involves comparing existing learner abilities against those required for the target role. Using the above model, the gaps in skills will act as inputs in the recommendation process.

2.4 Career Recommendation

Career recommendation identifies traits of the learner and recommends appropriate careers. This project uses the career recommendation data set with 200 observations and eight feature sets.

2.5 Research Gap

Approach	**Personalization**	**Skill Gap**	**IDP**	**Progress**
Manual mentoring	Manual	Possible	Possible	Separate
Online learning	Partial	Limited	Limited	Course-level
Career portals	Partial	Partial	Limited	Limited
Skill assessment	Skill-focused	Yes	Limited	Limited

Proposed system	Integrated	Yes	Yes	Yes

2.6 Literature Basis

Current research demonstrates the significance of utilizing artificial intelligence and machine learning in career advising, personalized learning, and skill-gap assessment. Lahoud et al. (2023) examined recommender systems for university major and career advising. Nabizadeh et al. (2020) researched personalized learning trajectories considering learners' knowledge, interests, and goals. Trujillo et al. (2025) surveyed machine learning approaches for student career forecasting, with special attention to the assessment of models, fairness, and privacy.

Thus, based on the literature review above, the Intelligent Recommendation System for Individual Development Plans (IDP) suggested by this research involves profile analysis, career matching, skill-gap assessment, personalized learning trajectories, and monitoring the development process.

3. SYSTEM ARCHITECTURE AND METHODOLOGY

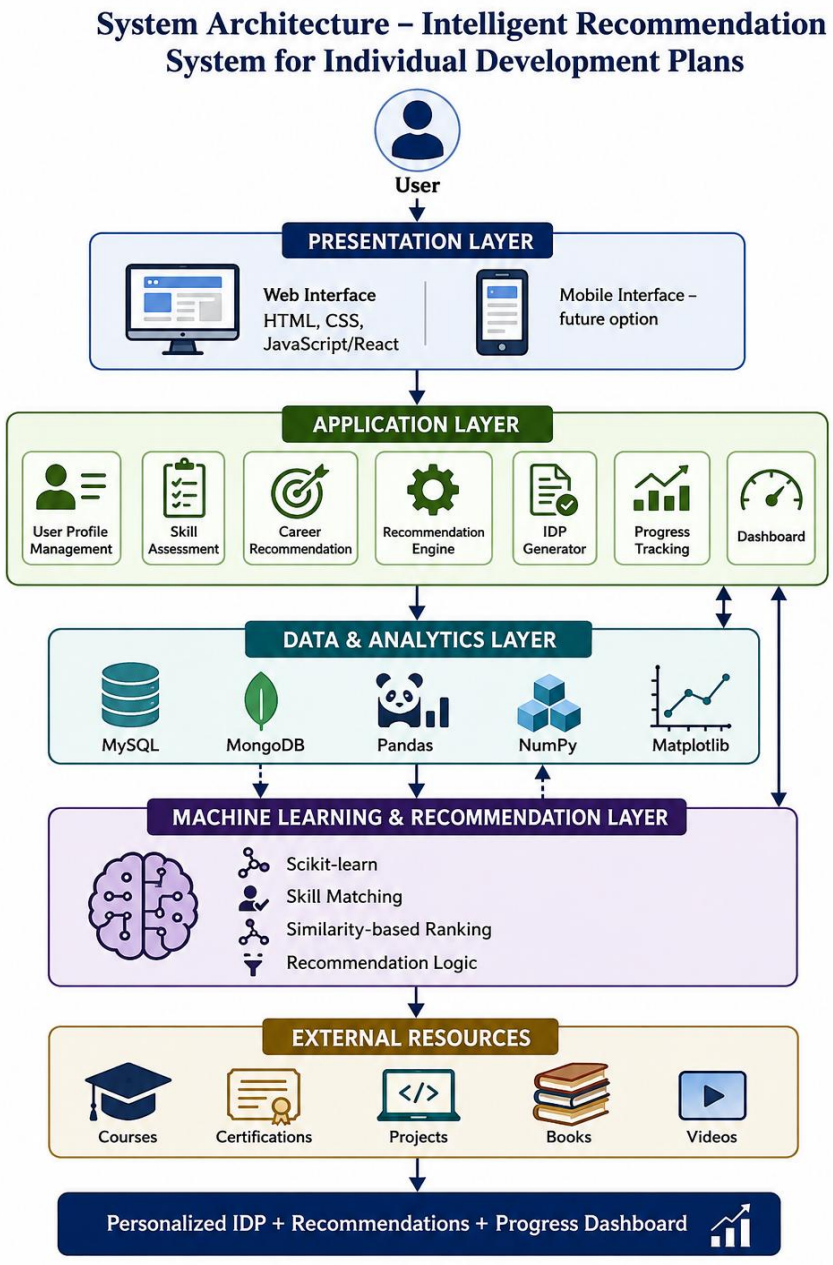


Figure 2. System Architecture of the Intelligent Recommendation System for Individual Development Plans.

The proposed architecture separates the user interface, application processing, data storage and recommendation functions. This separation supports maintainability and permits the recommendation module to evolve independently.

3.1 Main Components

- User Interface – registration, profile entry, skills, career choice, recommendations and progress.
- Application Layer – request processing, validation, recommendation management.
- Data Layer – user data, skill data, career data, resource data, recommendation data.
- ML and Analytics – data preprocessing, feature transformation, matching.
- Recommendation Engine – ranking of learning resources.
- IDP Generator – generation of structured recommendations to IDP.
- Dashboard – goals, gaps, recommendations and progress visualization.

3.2 Data Flow

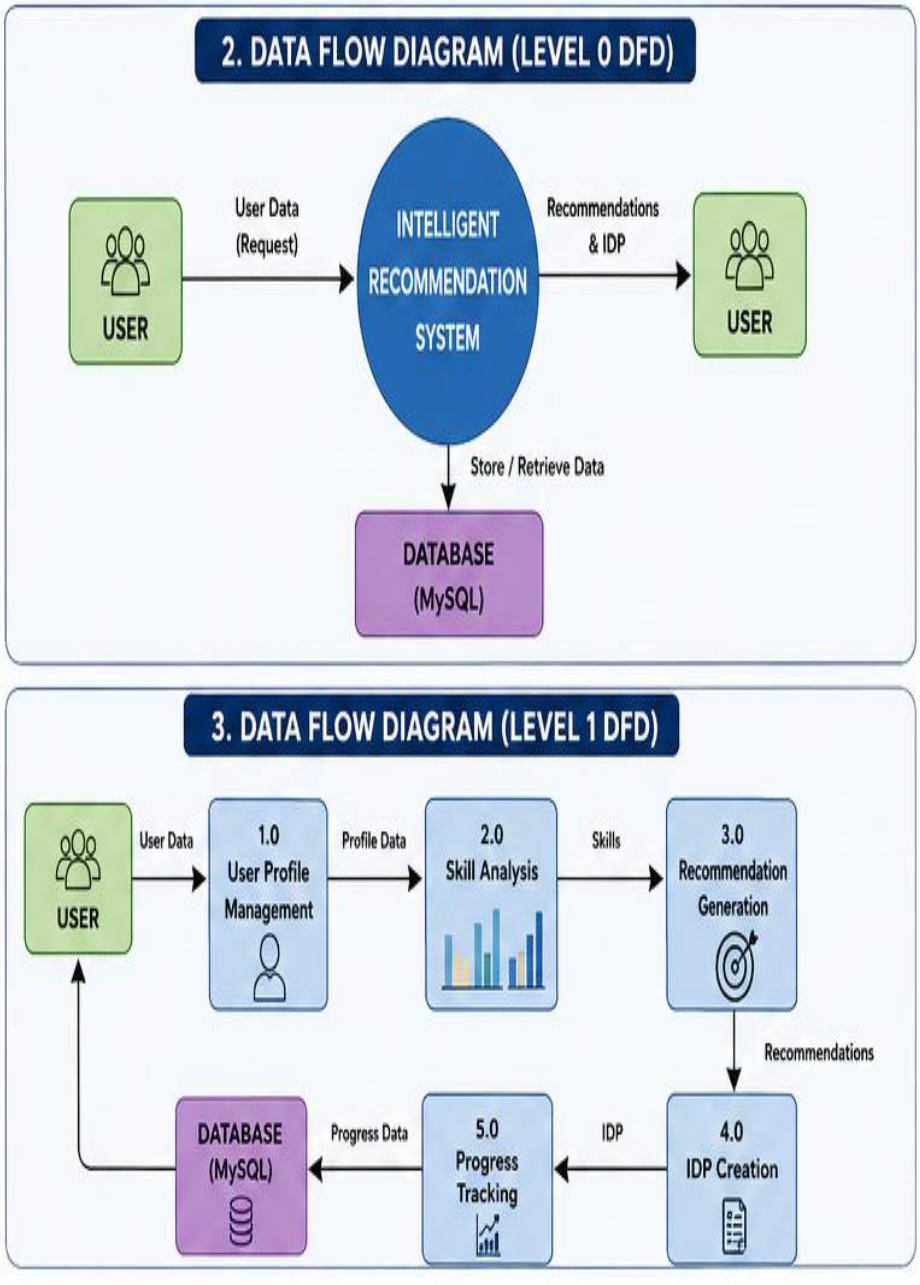


Fig. 3. Data flow from user profile input to recommendation and IDP output.

3.3 Data Collection and Dataset

For the project, we have selected a career recommendation data set that has 200 records and 8 feature sets. These feature sets can be described as follows:

Career Goal, Education, Experience, Skill Level, Interests, Learning Preference, Target Role, and Resource Type.

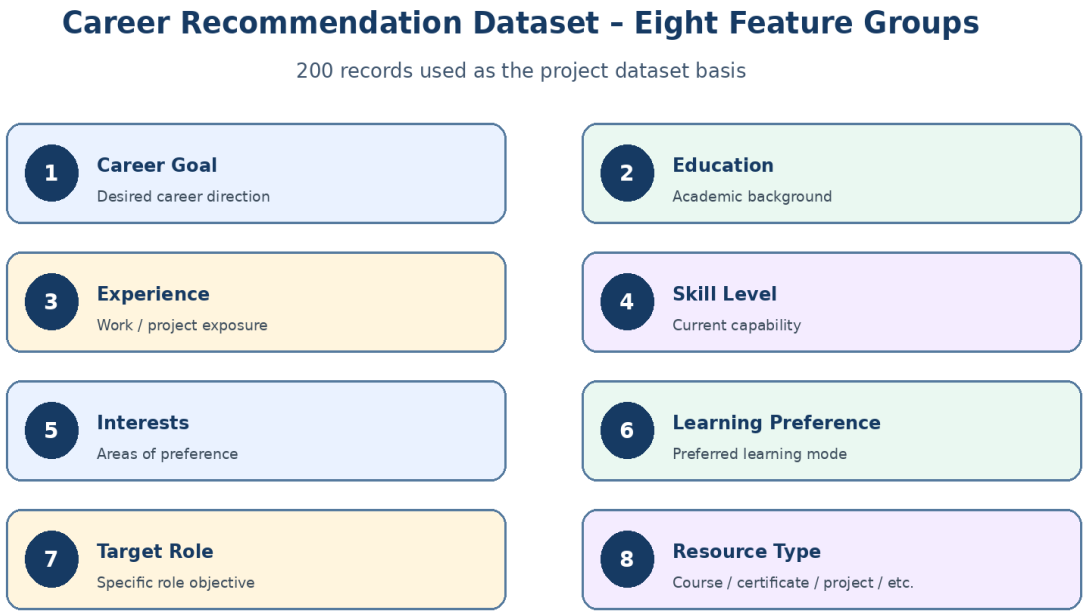


Figure 4. Eight feature groups considered in the career recommendation dataset.

Fig. 4. Eight feature groups considered in the career recommendation dataset.

3.4 Data Preprocessing

- Look for missing values or inconsistent records.
- Delete duplicate records or unneeded records as needed.
- Standardize or normalize numerical attributes as needed.
- Encode categorical attributes.
- Generate similar representations for user and dataset skills.
- Ready data for recommendation.

3.5 Feature Preparation

Career goal and desired position define the desired direction. Education and experience define the background. Skill set and interests define current ability and preference. Preferred learning style and resource type can define resource selection.

3.6 Skill-Gap Analysis

Conceptual Representation

Skill Gap = Required Skills – Existing Skills

This is an indication of skills which are lacking in the existing profile. The skills gaps are first prioritized and then recommendations are made.

3.7 Recommendation Generation

The recommendation component takes into account user profile, skills requirements, and resource attributes. A scoring mechanism based on similarity can rank the available candidate resources. The Phase-1 algorithm describes comparisons of existing skills with required skills, determination of skill gaps, and finally making recommendations.

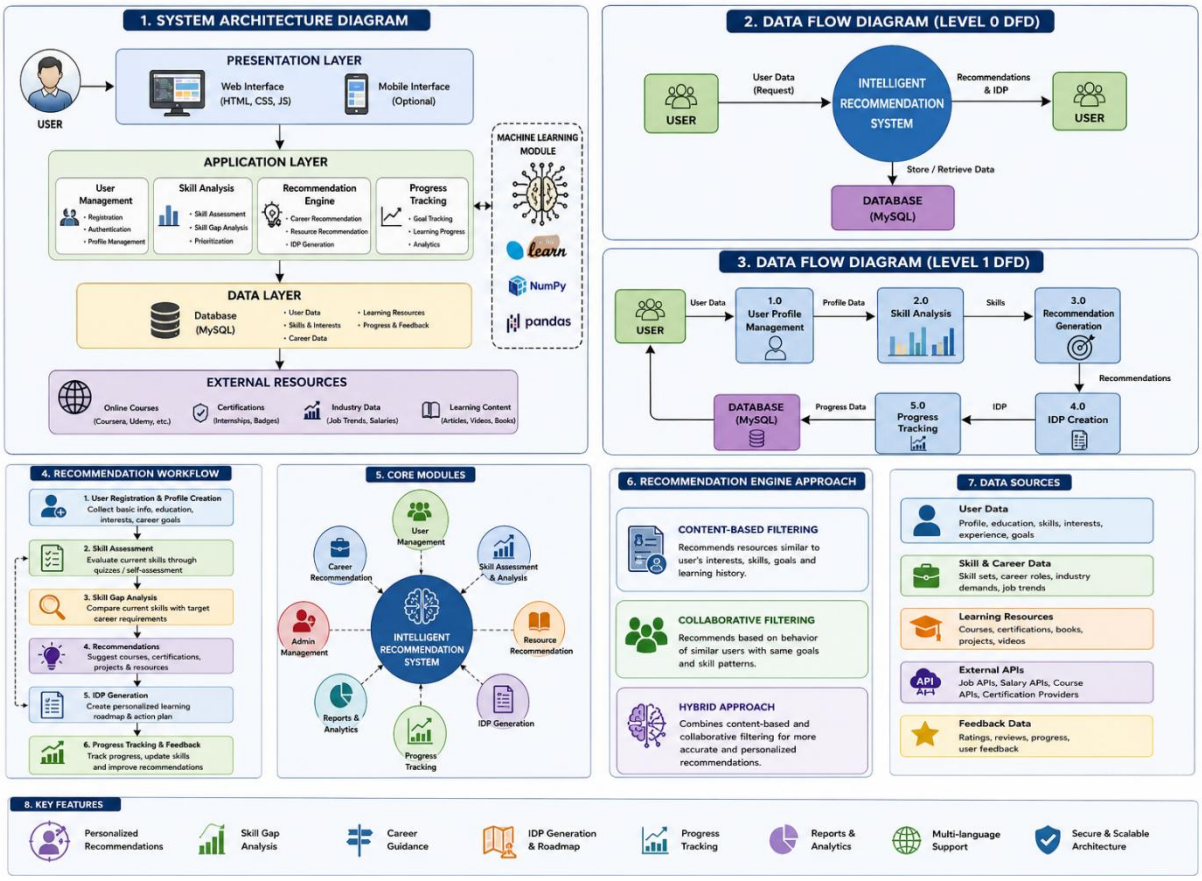


Fig. 5. Skill-gap and recommendation logic used to construct the IDP.

3.8 Individual Development Plan Generation

The IDP generator converts ranked recommendations into an ordered development plan. A plan can contain target career, skill gaps, learning activities, milestones and progress status. This gives users a structured sequence rather than an unorganized list of resources.

3.9 Recommendation Processing

After a user confirms a career direction, the system reads the profile and skill information and compares it with target-career requirements. Gaps are derived and candidate resources associated with the gaps are evaluated. The highest-ranked resources are selected for the development plan. Progress updates can later modify the user state.

3.10 System Workflow

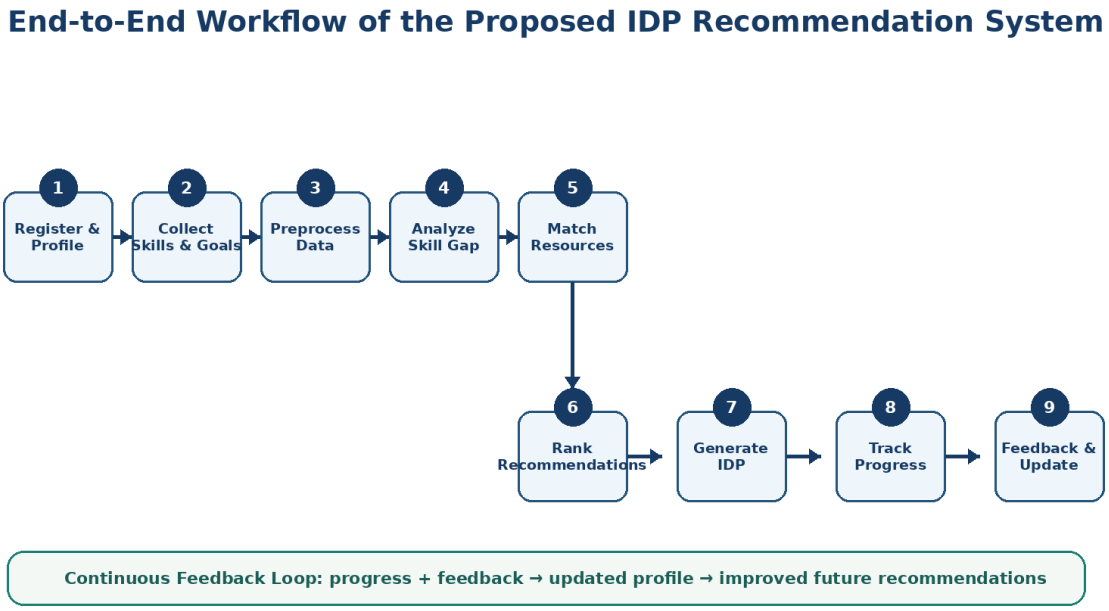


Figure 6. End-to-end workflow of the proposed system.

Fig. 6. End-to-end workflow of the proposed system.

3.11 Security and Data Management

Authentication and Role-Based Access can differentiate between the user and administrator roles. Profile and recommendation information must be secured from any unauthorized access. The Database separates the entities like users, skills, careers, recommendations and progress.

3.12 Scalability

The layered architecture enables the inclusion of further recommendation algorithms and sources without rewriting the entire program. Structured data is supported by MySQL and flexible documents are supported by MongoDB. The recommendation module can later become a Web service API.

3.13 Design Principles

- Personalization: recommendations should be made based on personal goals and skills.
- Modularity: components should be independent of each other when maintaining them.
- Interpretability: skill gaps should give an explanation for recommendations.
- Extensibility: new careers, new resources and algorithms should be added to the system.
- Feedback: future progress information should be used for recommendation refinement.

4. MLOPS INFRASTRUCTURE AND TECHNOLOGY JUSTIFICATION

4.1 MLOps Infrastructure

The suggested workflow comes after data preparation, feature creation, recommendation generation, validation, deployment, and updates. The environment for the current project is localhost, while future versions may incorporate continuous monitoring and updating as soon as large amounts of data become available.

• Data Collection – retain career, skill, and resources data.

• Data Preprocessing – cleansing, encoding, and transformation of data.

• Feature Creation – creating similar skill and career features.

• Recommendation Generation – applying matching algorithms with the help of Python and Scikit-learn.

• Validation – testing the relevance and functionality of recommendations.

• Deployment – integrating recommendation component into web application.

• Monitoring and Updating – relying on feedback and progress for future improvements.

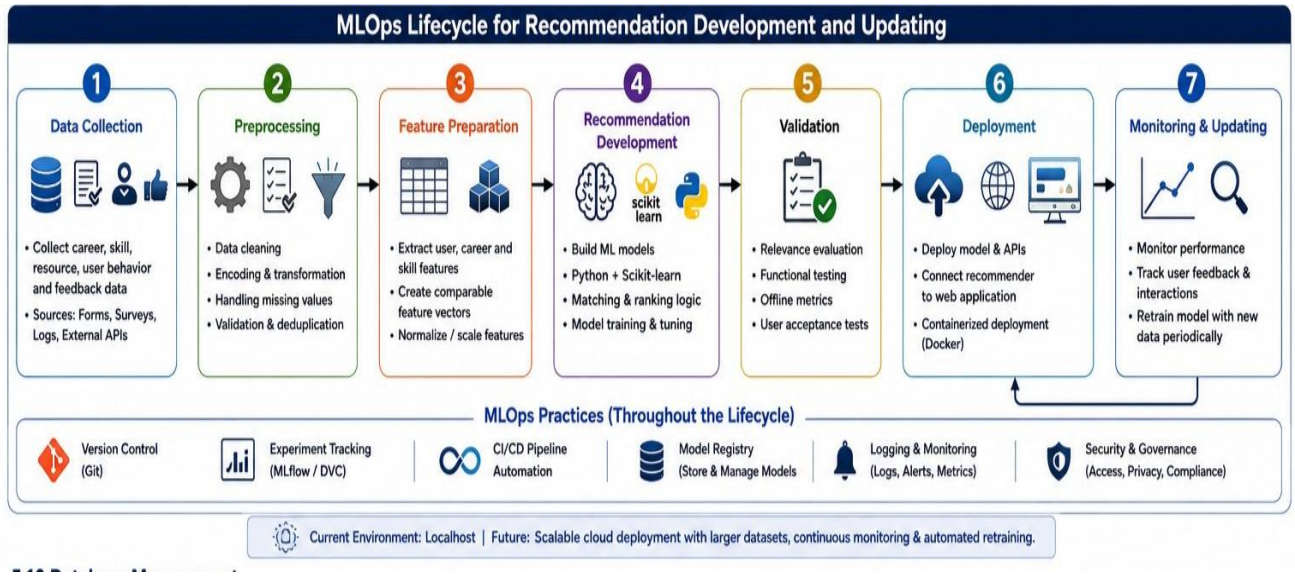


Fig. 8. MLOps lifecycle for recommendation development

4.2 Technology Justification

4.2.1 Python

Python supports backend processing, data preparation and recommendation logic.

4.2.2 Scikit-learn

Scikit-learn provides machine-learning and similarity-analysis functionality for recommendation development.

4.2.3 Pandas and NumPy

Pandas supports dataset manipulation and NumPy supports numerical computation.

4.2.4 Flask / Django

Flask or Django provides the application/API layer connecting interface, data and recommendation services.

4.2.5 HTML, CSS, JavaScript and React

These technologies support the presentation layer. React provides reusable interface components while HTML, CSS and JavaScript support structure, styling and interaction.

4.2.6 MySQL and MongoDB

MySQL provides relational storage for structured entities. MongoDB provides flexible document-oriented storage for semi-structured information.

4.2.7 Matplotlib

Matplotlib supports exploratory data analysis and visualization.

4.2.8 Localhost

Localhost provides the development and integration-testing environment for the current project.

5. IMPLEMENTATION DETAILS

5.1 User Authentication

The authentication module provides registration and login. Access to user and administrative functions can be controlled according to role.

5.2 User Profile Management

The profile module stores information required for personalization, including name, education, skills, interests and career goals.

5.3 Skill Assessment

The skill assessment module records current skills and skill levels. These values are used in skill-gap analysis.

5.4 Career Recommendation

The career recommendation module uses the dataset and user features to identify career directions relevant to the profile.

5.5 Skill-Gap Analyzer

The analyzer compares current skills with target-career skills and passes identified gaps to the recommendation engine.

5.6 Course and Resource Recommendation

The recommendation module maps identified skill gaps to relevant courses, certifications, projects, books, videos and other learning resources. Results can be ranked according to career alignment and profile compatibility.

5.7 Individual Development Plan Module

The IDP module converts recommendations into a structured development sequence containing goals, skills, activities, milestones and completion status.

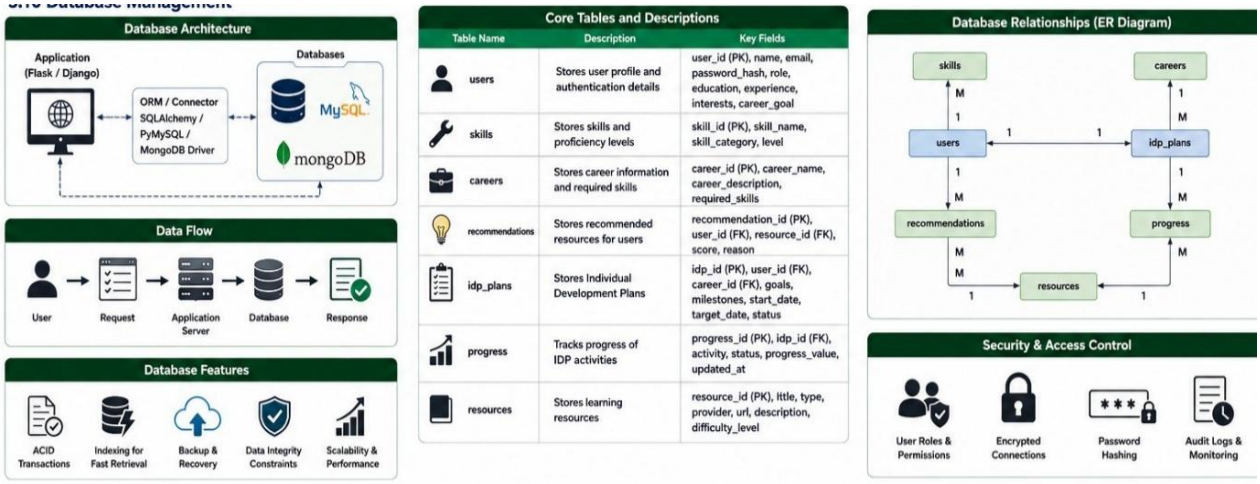
5.8 Progress Tracking

Progress tracking records completed activities and changes in skill status. A dashboard can summarize goals, completed activities and recommendations.

5.9 Dashboard

The dashboard provides a consolidated view of career direction, skill gaps, recommendations, development-plan activities, progress and achievements.

5.10 Database Management



Entities involved include User, Skill, Career, Recommendation, IDP, and Progress. The User entity is related to the Skill entity in terms of information as well as development activities.

Fig. 7. Conceptual entity relationship structure for the IDP system.

5.11 Backend API

The backend API bridges the presentation layer and the database and recommendation services. It receives profile and skills data, validates, requests for recommendations, saves the output and responds back to the user interface.

5.12 Recommendation Algorithm

• Activate the system.

• User registration and creation of a user profile.

• Collecting skills, interests, educational background, experience, and career objective.

• Storing user data.

• Analysis of the present skills.

• Comparing present skills with career skills.

• Identifying skill gap.

• Recommending suitable learning resources.

• Creating the Individual Development Plan.

• Visualizing the learning journey.

• Updating the profile based on development.

5.13 Recommendation Scoring Concept

The score of conceptual matching may include factors like skill similarity, career compatibility, and preference compatibility. The weights need to be determined experimentally since this paper does not provide performance numbers.

Score = w_1(Skill Match) + w_2(Career Match) + w_3(Preference Match)

5.14 User and Admin Functions

Actor	**Major Functions**
User	Register, login, update profile, enter skills, select career, view gaps, receive

	recommendations, manage IDP, track progress
Admin	Manage users, learning resources and career information; monitor system data

6. RESULTS AND DISCUSSION

This particular project has been presented as a recommendation framework based on functionality but not as an assertion of experimental proof for prediction accuracy. The given project data proves the existence of a career recommendation database containing 200 entries and eight sets of features within a localhost implementation environment. Hence, results emphasize functionality aspects rather than artificial numerical values.

6.1 Dataset Result

The 200 record dataset serves as the project-level ground on which the career recommendation process will be developed and tested. The eight feature sets include learner and career information for profile-to-career mapping.

6.2 Functional Result

• The system can obtain data regarding user profile and skills information.

• The process of recommendations will compare existing skills with the required career skills.

• Skill deficiencies can be obtained and used for recommendation.

• Recommendations can be structured into an IDP.

• The dashboard design enables tracking progress.

• Updated user data can be used for future recommendation cycles.

6.3 Recommendation Output

The main output is a personalized set of learning and development actions. Recommendations are linked to the user's target career and identified skill gaps rather than being treated as a generic list.

6.4 Skill-Gap Output

Skill-gap analysis provides an interpretable intermediate result by showing competencies associated with the target career that are absent or insufficient in the current profile.

6.5 IDP Output

The IDP combines target career, skill gaps, learning resources and progress status into a step-by-step development route.

6.6 Discussion

The primary benefit of the framework is the integration of career recommendations, skills assessment, gap assessment, resource recommendations, and development into a single process. The architecture design makes it layered to segregate the interface, backend, data, and recommendation components and improve incrementally.

However, due to the small scale of the existing dataset, we cannot draw any general recommendations about recommendation performance. A dataset of 200 records will work well for a project development phase, but not for evaluation purposes.

6.7 Evaluation Framework

In future empirical testing of the system, the precision, recall and F1 score can be employed in comparing recommendations with labeled items. For ranked recommendations, metrics like Hit Rate, MAP, NDCG can be utilized. For numerical preference predictions, MAE and RMSE may be applicable.

Area	**Validation**	**Evidence**
Data quality	Missing/duplicate checks	Counts before/after

Recommendation	Relevant-item comparison	Precision/Recall or ranking
Skill gap	Expert/labelled comparison	Agreement/error
Usability	User study	Questionnaire
Performance	Repeated requests	Response time
Scalability	Increasing records/users	Runtime/resource trends

6.8 Comparison with Existing Approaches

Capability	**Existing/Separate Tools**	**Proposed IDP System**
User profile	Often available	Integrated
Career recommendation	Some systems	Integrated
Skill-gap analysis	Separate/limited	Integrated
Resource recommendation	Available	Skill-linked
Structured IDP	Often manual	Generated
Progress tracking	Separate	Integrated

7. LIMITATIONS

• Dataset Dependency: Quality of recommendation depends on dataset coverage and consistency.

• Dataset Size: Current dataset consists of 200 records; generalization might be impossible.

• Feature Groups: Eight feature groups might not include all parameters influencing career progression.

• Manual Input: Correct user input is needed.

• Recommendation Coverage: Result depends on availability of careers and resources.

• Cold Start Problem: For new users with no profile data, recommendation might not be personalized enough.

• Evaluation Constraint: No numerical metrics such as accuracy, precision, recall and F1 score are reported without experimental tests.

• Localhost Deployment: Scaling on cloud environment is not done yet.

• Changing Requirements: Requirements for career progress change and need periodic dataset update.

• Human Factor: The application helps in making decisions but does not replace the importance of mentoring.

7.1 Risk Management

Risk	Effect	Mitigation
Technical	Integration failure	Modular testing
Data	Poor recommendations	Cleaning and validation
Security	Unauthorized access	Authentication/access control

Time	Delay	Phased development
Adoption	Low engagement	Simple interface and feedback
Maintenance	Outdated resources	Periodic updates

7.2 Ethical and Privacy Considerations

The system analyzes data related to education, skills, and career development. The collection of data should be made for a definite purpose and kept away from any sort of unauthorized access. Suggestions should be interpretable enough that the user knows why particular skills and sources are recommended.

8. CONCLUSION AND FUTURE SCOPE

Intelligent Recommendation System for Individual Development Plans (IDP) aims at assisting people in selecting proper learning paths that fit them best according to their skills, interests, and professional aspirations. In the constantly changing world of today, people face a problem of choosing right courses and training programs for developing their skills and achieving their future career goals. The presented project offers a solution in the form of the system that provides individualized recommendations for people on the matter of learning and developing new skills.

Users are offered to create an account in the system in order to get recommendations from it. Users should fill in the required information on their skills, learning interests, and development plans. When the system receives the user's information, it processes it through the use of modern data analysis and machine learning techniques. After analyzing the provided information, the system gives recommendations regarding proper courses and learning materials for developing new skills of the user.

As far as the implementation of the system is concerned, it involves the use of several modern technologies. The frontend of the application is implemented in HTML, CSS, and JavaScript and provides users with a user-friendly interface. As for the backend,

8.1 Expected Outcomes

- Develop personal course recommendations according to the user's skills and preferences.
- Help users detect skill shortages and recommend courses that can develop them.
- Make the process of education more effective through directing users to appropriate educational sources.
- Offer an intuitive and interactive environment for exploration of learning possibilities.
- Assist in career development through recommending courses corresponding to industry skill requirements.
- Enable users to monitor their progress and control their skill development plan.
- Showcase how machine learning is implemented into educational technology systems.
- Develop a scalable architecture that can later be expanded using new datasets and recommendations technologies.

ACKNOWLEDGEMENTS

The authors would like to express their gratitude to the faculty and project coordinators of the Department of Artificial Intelligence and Data Science, Angadi Institute of Technology and Management, Belagavi, for their guidance, technical support and encouragement during the development of this project. The authors also acknowledge the support of their institution and peers in preparing the system and documentation.

References

1. Burke, R. (2002). "Hybrid Recommender Systems: Survey and Experiments." User Modeling and Adapted Interaction, 12(4), 331–370.
2. Adomavicius, G., & Tuzhilin, A. (2005). "Toward the Next Generation of Recommender Systems." IEEE Transactions on Knowledge and Data Engineering, 17(6), 734–749.
3. Bobadilla, J., Ortega, F., Hernando, A., & Gutiérrez, A. (2013). "Recommender Systems Survey." Knowledge-Based Systems, 46, 109–132.

4. Koren, Y., Bell, R., & Volinsky, C. (2009). "Matrix Factorization Techniques for Recommender Systems." Computer, 42(8), 30–37.
5. Lops, P., De Gemmis, M., & Semeraro, G. (2011). "Content-based Recommender Systems: State of the Art and Trends." In Recommender Systems Handbook, Springer.
6. Salakhutdinov, R., Mnih, A., & Hinton, G. (2007). "Restricted Boltzmann Machines for Collaborative Filtering." Proceedings of the 24th International Conference on Machine Learning.
7. Schafer, J. B., Konstan, J., & Riedl, J. (2001). "E-Commerce Recommendation Applications." Data Mining and Knowledge Discovery, 5(1–2), 115–153.
8. Isinkaye, F. O., Folajimi, Y. O., & Ojokoh, B. A. (2015). "Recommendation Systems: Principles, Methods and Evaluation." Egyptian Informatics Journal, 16(3), 261–273.
9. Zhang, S., Yao, L., Sun, A., & Tay, Y. (2019). "Deep Learning Based Recommender System: A Survey and New Perspectives." ACM Computing Surveys, 52(1), 1–38.
10. Ricci, F., Rokach, L., & Shapira, B. (2022). Introduction to Recommender Systems Handbook. Springer.