International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 10, Issue 5, pp 405-412, May-2026

https://doi.org/10.47001/IRJIET/2026.105055

Smart Placement Intelligent System Using AI-Powered Analytic for Campus Placements

[1]Anita Puri, [2]Aniket Gajbhiye, [3]Mohit Lakhe, [4]Prajwal Kotangale, [5]Swarup Ambatkar, [6]Siyan Khan

[1,2,3,4,5,6]Computer Science & Engineering, G.H Raisoni College of Engineering and Management, Nagpur, India

E-mail: [1]anita.puri@ghrua.edu.in, [2]aniketgajbhiye46@gmail.com, [3]mohitlakhe2@gmail.com, [4]prajwalkotangale151@gmail.com, [5]swarupambatkar1@gmail.com, [6]siyankhan337@gmail.com

***Abstract* - Campus placement may be viewed as one of the crucial steps in the life of engineering graduates, but unfortunately, the placement strategy followed by most universities today is administrative rather than intelligent. In this work, we present a Smart Placement Intelligent System (SPIS), which is a novel AI-driven solution developed using the state-of-the-art MERN stack (MongoDB, Express.js, React.js and Node.js) development platform.**

The SPIS includes several components such as the PRS engine, NLP driven ATS Resume Analyzer, Skill Gap ID module, AI Roadmap Generator, Real Time Leaderboard with Peer Velocity Index (PVI), Collaboration Feed, and the Smart Job Portal with eligibility enforcement along with the SuperAdmin Dashboard. Experiments were conducted on our platform using the dataset including details about 3,500 students, and we observed that the prediction model used by us in terms of an ensemble learning technique has an accuracy above 90% and performs better than some of the conventional machine learning techniques. In addition to these experiments, we conducted user tests to prove the effectiveness of our proposed system.

Keywords: MERN stack; PRS engine; ATS resume analyser; skill gap identification; AI roadmap generator; peer velocity index.

I. INTRODUCTION

Campus placements act as the main vehicle for engineers fresh out of academics to get employed. A lot of effort is put in by the institutional Training and Placement (T&P) cells to ensure successful completion of the process; however, very little emphasis is placed on building up the necessary infrastructure to develop the competitive edge of students. Extensive research has shown that many students do not realise their deficiencies and have never had their resumes analysed according to ATS norms. More importantly, there has been a general lack of understanding about the nature of competition in their chosen jobs.

In the light of automation, rapid digitisation, and technology stacks that keep changing, this gap has been continuously increasing. In this environment, the use of machine learning algorithms and artificial intelligence has proven its effectiveness in a wide range of areas from mining information from academic performance data to predicting placement results or automating the entire recruitment process. Yet, there is currently no platform that integrates all the functionalities mentioned into an end-to-end system dedicated to help students with placement preparation.

This paper introduces the concept of the Smart Placement Intelligent System (SPIS), a full-stack platform augmented with artificial intelligence features aimed at making the placement preparation process more efficient and systematic. By combining eight modules into one system, SPIS offers a complete solution for helping students to prepare for the placement season. The main novelties of this study are:

- The composite Placement Readiness Score (PRS) calculation engine with flexible sub-score weighting and animated visuals.
- The ATS Resume analyser with NLP-based analysis, targeted keyword matching and suggestions on improvements.
- The Skill Gap Identifier Module with missing skills identification, importance-based analysis and structured recommendations on learning paths.
- The AI Roadmap Generator generating a unique 8-week training plan for each of the roles with response caching in MongoDB.
- The real-time Leaderboard with the Peer Velocity Index (PVI) allowing to filter by branches and track progress week-over-week.
- The Discussion social feed allowing for peer-to-peer discussion and learning with auto-moderation and full-text search capabilities.
- The Smart Job portal with server-side multi-criteria eligibility check and the five-component Job Match Score.
- The Superadmin dashboard.

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 10, Issue 5, pp 405-412, May-2026

https://doi.org/10.47001/IRJIET/2026.105055

II. RELATED WORK

A. Machine Learning-Based Placement Predictions

Machine learning techniques to predict campus placements results have gained significant traction from various researchers. Shahane *et al.* [5] benchmarked Logistic Regression, Decision Tree, K-Nearest Neighbours, and Random Forest classical classification algorithms for academic percentage-based predictions, with the latter showing promising results. Similarly, Swaminarayan and Rajput [6] explored Support Vector Machines and ensembles, concluding that hybrid models yield better results than their single classifier counterparts. Rao [7] established that academic percentages, communication tests' results, and internships proved to be some of the most discriminating features for campus placements predictions. More recently, Kumar et al. [8] proposed a gradient-boosting algorithm tailored for categorical data called CatBoost, which showed superior performance in terms of dealing with field-of-study and student-demographic-category features, putting the focus on feature engineering and fine-tuning. This paper utilises XGBoost, which further extends the gradient-boosting approach with explicit regularisation and achieved 91.3% classification accuracy on the sample of 3,500 profiles at the university in question.

B. Automated Resume Screening

The advent of automated resume screening systems in hiring processes encouraged numerous studies to address the problem of intelligent resume analysis. Chavan *et al.* [9] proposed an enhanced version of ATS that uses NLP techniques and K-Nearest Neighbours to improve candidate-job matching. Warusawithana *et al.* [10] attempted to tackle the issue of modern resume structure variation and suggested a resume parsing algorithm with Named Entity Recognition for extracting section-based information. Bharadwaj *et al.* [11] employed LSTM architectures to categorise resumes by skill profiles. Furthermore, recent developments in transformer-based models yielded significant performance improvements – Deshmukh and Raut [12] showed that BERT-based resume screening surpasses simple keyword matching approaches in semantic similarity measurements. Moreover, Resume2Vec model [13] using BERT, RoBERTa, and DistilBERT token embeddings with cosine similarity metric outperforms traditional ATS by up to 15.85% in nDCG scoring. Skondras *et al.* [14] confirmed that LLMs can guide keyword matching algorithms to improve compatibility scores even further. The ATS module of SPIS takes advantage of all these findings by implementing role-specific keyword corpora and section-detection heuristics.

C. Skill Gap Detection

Addressing skill gaps between students and those demanded by the industry has been an ongoing research topic as well. In the Skill Sync system [15], semantic similarity with an LLaMA-2-like customized model was used to identify domain-specific skills gaps and provide personalized learning tracks. Occupational taxonomies consistent with the O*NET database have proven useful in achieving cross-domain skill mapping [16]. Real-time occupation course recommendations using collaborative filtering and the Naïve Bayes classification methods have been attempted. More recently, LLM APIs were harnessed for context-aware learning roadmap generation [18]. In SPIS, an insensitive-to-case matching strategy is employed based on role-required-skills dataset preloaded into SPIS, which includes ranked links to various learning resources.

D. Research Gap

Despite the achievements mentioned above, existing studies deal with individual aspects separately. There is no system developed to date that combines placement probability predictions, intelligent resume evaluations, skill gap detections with learning path recommendations, AI-generated personal learning roadmaps, peers' velocity benchmarks, social collaboration features, intelligent job portals based on eligibility criteria, and institutional management functions into one single platform.

III. EXISTING SYSTEM

The campus placement infrastructure in academic institutions may broadly be categorized into the following three groups:

i. Manual T&P systems, where the tracking system is spreadsheet-based,
ii. Generic online job listings, which operate independent of the institutional environment, and
iii. Basic PMSs, which are essentially digitized versions of administrative processes.

Each of the above-listed systems has limitations associated with it. Manual spreadsheet-based systems tend to have issues with scalability, lack version control, and fail to offer meaningful analytics from data accumulated in spreadsheets. The generic job listing platforms are successful in aggregating job postings but fail to provide institution-centric evaluation tools. These include resume evaluation, skill-gap identification, and readiness assessment based on the academic curriculum of the student.

Conventional PMS systems are efficient in terms of process automation but lack analytical intelligence. The

ISSN (online): 2581-3048
Volume 10, Issue 5, pp 405-412, May-2026
https://doi.org/10.47001/IRJIET/2026.105055

features offered by these platforms are limited to management of records, scheduling, and tracking attendance of students in these activities.

An organized comparison of the above-discussed systems and the limitations inherent in each, compared to SPIS

IV. PROPOSED SYSTEM

A. System Overview

The current system is a full stack application that is ready for cloud deployments and uses the following MERN tech stack: front-end with React + Vite and Tailwind + Bootstrap design, Node.js + Express.js services for a RESTful API, and MongoDB with Mongoose as the ODM data store. Generation of AI roadmaps is done via the API. File resources (CVs, profile pictures) are stored in Cloudinary. Authentication uses JWT with four roles – Student, tpo admin, management admin, and superuser. The system is deployed with Docker Compose and includes Kubernetes manifests as well.

As seen in Fig. 1, the block diagram illustrates the three-layer architecture: the Presentation Layer with four dashboards using React + Vite, the API Layer with Node.js/Express.js services, and the Data Layer with collections in MongoDB. Additional integrations include Cloudinary, SMTP/Gmail, and the AI service.

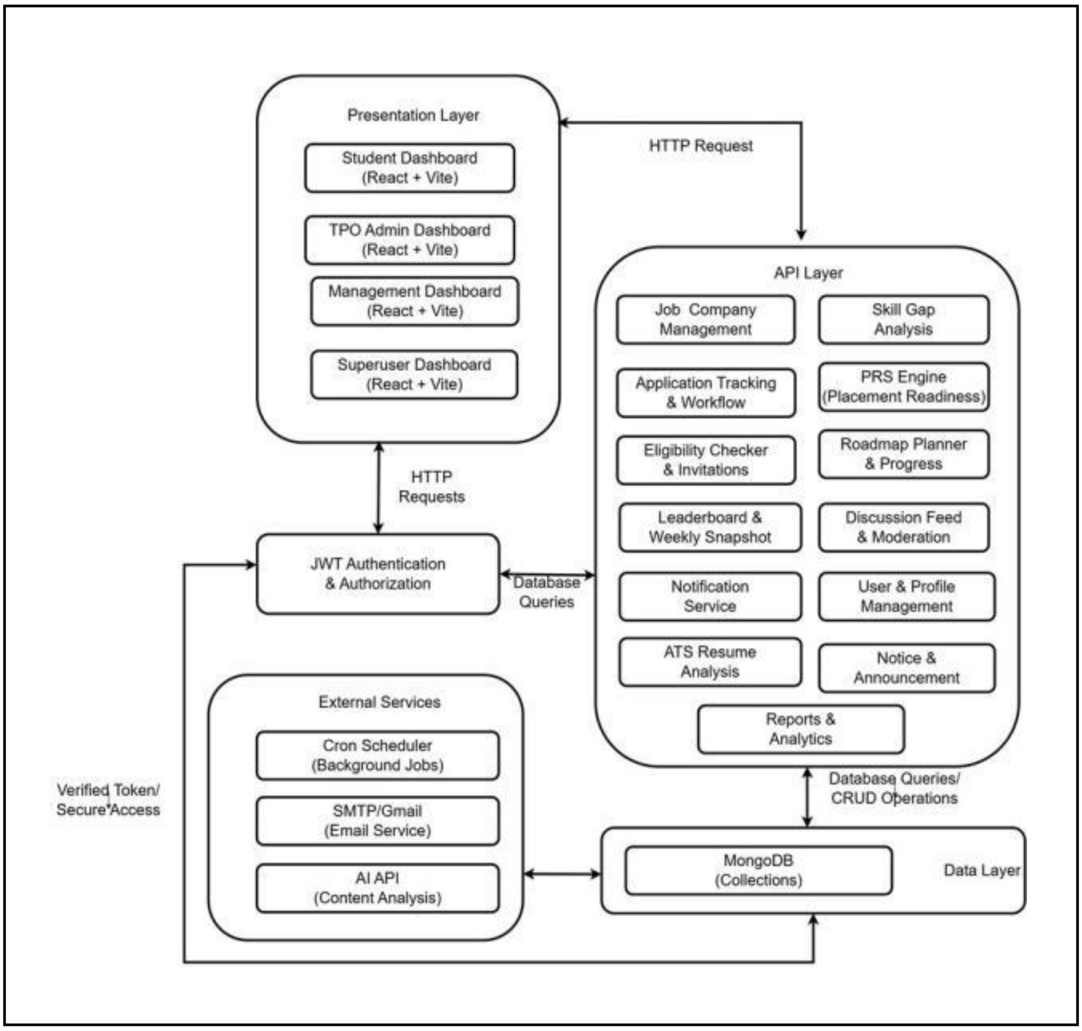

Figure 1: Block Diagram of the Smart Placement Intelligent System (SPIS), including its Presentation Layer, API Layer, Data Layer, and external service integration

Fig. 2 provides a flowchart describing all interactions of a user in SPIS. When the user opens the portal, he or she needs to choose the corresponding role: Student, TPO, Management, or Superuser. Once the user has been verified via credentials, his/her respective dashboard appears. In case of Students, the next steps involve account approval verification before allowing the completion of the profile, searching for jobs, evaluating whether the candidate fulfils requirements for the position in question, applying for a job and receiving the response.

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 10, Issue 5, pp 405-412, May-2026

https://doi.org/10.47001/IRJIET/2026.105055

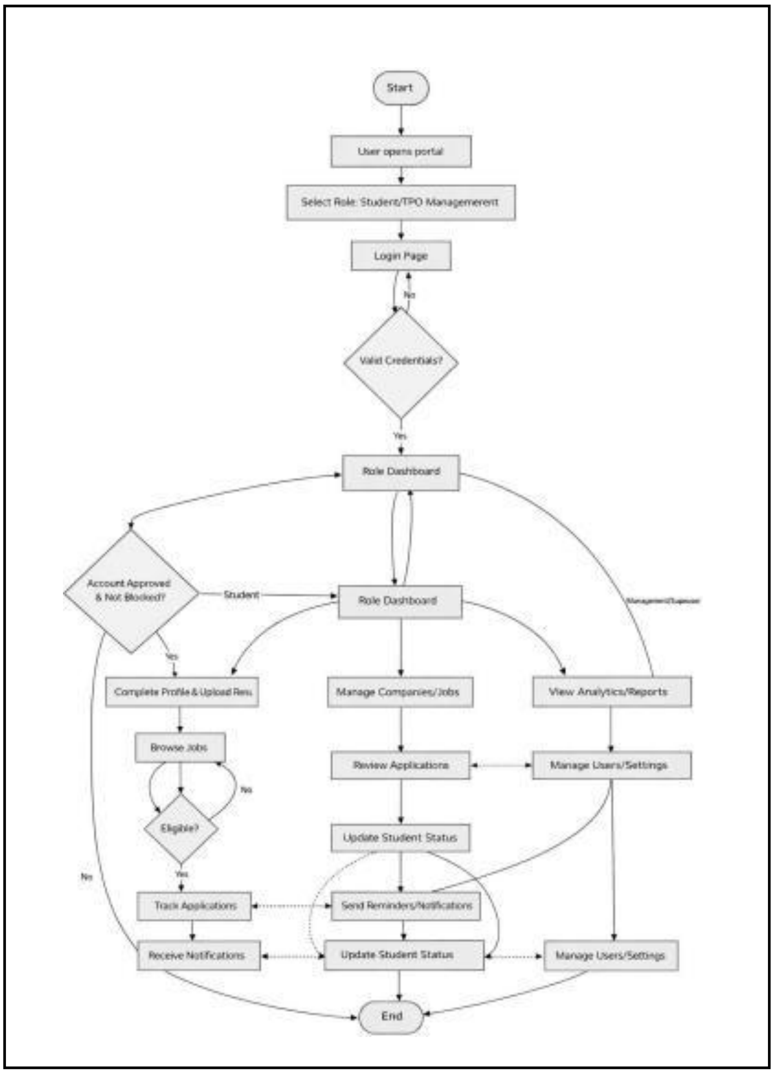

Figure 2: User interaction flowchart in SPIS for Student, TPO, Management, and Superuser roles

B. Core Modules

1) Placement Readiness Score (PRS) Engine

PRS is a composite score lying within the interval [0, 100], incorporating four weighted sub-components representing student readiness. An academic score (AS) is generated based on the student's CGPA normalised into a centesimal scale, contributing 35% to the overall score. Skill assessment score (SAS) measures the diversity of the student's self-assessed skill set against the predefined benchmark skill count, adding 30%. The Resume Quality Score (RQS) pulls data directly from the ATS analyser module, contributing 25%. Finally, Activity Engagement Score (AES) measures completion of assigned activities, accounting for 10%. The formula for calculating the composite score is defined as:

$$PRS = 0.35 \times AS + 0.30 \times SAS + 0.25 \times RQS + 0.10 \times AES$$

The contribution weight of all sub-scores is configurable through Superadmin settings panel, allowing flexibility in defining the scoring methodology. PRS score is represented visually by means of an animated SVG circle progress bar displayed on the student dashboard and recomputed at each session by calling the POST /api/prs/calculate API.

2) ATS Resume Analyser

Resume uploads for PDF and DOCX file formats are managed with the help of Multer middleware. Extracting text from resumes involves the use of the pdf-parse library for PDF files, and the mammoth library for DOCX files. Matching extracted text against the predefined set of keywords, stored in role keywords MongoDB collection with more than fifty keywords per role in eight predefined roles, is performed with case-insensitive token matching. ATS score is calculated as a proportion of matched keywords to the total number of keywords, scaled to the centesimal scale. Identification of sections includes detecting presence of four sections: Summary, Work Experience, Skills, and Education. This module generates five prioritized recommendations, focused

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 10, Issue 5, pp 405-412, May-2026

https://doi.org/10.47001/IRJIET/2026.105055

on acquiring high-importance missing keywords and missing sections in the resume. ATS score serves as a component of PRS – Resume Quality Score (RQS).

3) Skill Gap Identification Module

Given the student's targeted role, this module performs comparison between the embedded skills array in the student's record and the role_required_skills collection, using the case-insensitive string matching approach. Calculated match percentage indicates how well student's competencies cover role's skill requirement. Missing skills are then returned in the result alongside their associated importance level (high, medium, low) and URL links to the free learning resources available from websites like Coursera, YouTube, GeeksforGeeks, freeCodeCamp, and others.

4) AI Roadmap Generator

Based on the API, the Roadmap Generator creates a personalized 8-week, 32-task preparation plan for any target role selected by the student. In the prompt sent to the model, there will be the student's academic branch, CGPA, skill set, and current PRS so that it can generate tasks according to the current position of the student. Responses from the API will be validated and stored in the roadmap templates collection as structured JSON objects tagged with the ai generated flag. Once a new student seeks the same target role, the stored roadmap will be fetched within 50 milliseconds or less.

5) Leaderboard with Peer Velocity Index (PVI)

All the approved students are ranked by their current PRS in the leaderboard and filtered by branch for a more contextualized ranking against their peers. The PRS of the week is automatically captured by the node-cron schedule. The formula used for the Peer Velocity Index is:

$$PVI = ((\text{PRS this week} - \text{PRS last week}) / \text{avg branch improvement}) \times 100$$

If PVI is greater than 1.0, the student's improvement rate is higher than the average of the academic branch. Direction and magnitude of the PRS changes every week and shown by the student entries. A gold, silver, and bronze podium effect will be provided for the top three students.

6) Discussion Social Feed

Through a discussion social feed, students add their experience about placements, interview question bank, company review, preparation tips and tricks, etc. AJAX powered liked with optimistic update, thread-based comments, parental relationship between comments, sharing event recording, and moderation based on reporting will be available. Any post reported five or more times will be automatically hidden from public view pending manual verification. Those who have placements will be added with a verified badge. MongoDB index will be used for full-text search.

7) Intelligent Job Portal

Eligibility criteria validation involves six institutional criteria based on which each application is validated, i.e., CGPA cut-off, branches allowed, backlogs in academics, 10th & 12th mark criteria, batch year. Each student-job pairing receives a Job Match Score based on five parameters: Skill Match (40 points), Above minimum CGPA Margin (15 points), PRS contribution (25 points), Branch Matching (10 points), and ATS score (10 points). In addition, applications will be automatically blocked based on non-eligibility of criteria and lack of resume upload. Notification push service will be implemented through the job notification collection for eligible students regarding new jobs.

8) Superadmin Dashboard

Centralized administration dashboard includes: Student life cycle management - approve, block, suspend, export CSVs; Job/Company management along with application statuses management; Discussion thread management along with reported content moderation queues; Admin account creation for TPO & management; Chart.js-based analytic dashboard featuring placement stats, PRS Score distributions & skill gaps heatmaps for each branch; System setting management including PRS Sub-Score weightage, roadmap templates management, manual data operations.

V. DATABASE STRUCTURE

Mongoose ODM with timestamps: true and compound indexes are used in the MongoDB data layer. There are thirteen main collections in the database, each tailored for a certain domain-specific problem. For example, the users collection embeds the student profile, which consists of the following elements: CGPA, branch, skills array, PRS score, target role, and applied jobs. Similarly, the jobs collection contains job eligibility criteria, necessary skills, CTC details, and applicants' references. Furthermore, the companies collection stores company-level information such as historical recruitment figures. The prs history and prs weekly snapshots collections provide longitudinal readiness tracking and leaderboard functionality, respectively. The resume analyses collection keeps ATS results, which include matched and missing keyword vectors, as well as section-check flags. The role required skills and role keywords collections are seeded lookup tables for the Skill Gap and ATS modules, respectively. The skill gap results collection contains

ISSN (online): 2581-3048
Volume 10, Issue 5, pp 405-412, May-2026
https://doi.org/10.47001/IRJIET/2026.105055

competency analyses per student. Roadmap templates and student roadmap progress collections hold AI-based roadmaps and task completion status, respectively. Lastly, the discussion posts and post comments collections fuel the social feed, which features like counting and nesting. Finally, the peer benchmark cache holds pre-aggregated branch-level metrics on a weekly basis, including common skills, average PRS, and common hiring companies to avoid costly aggregations while generating leaderboards.

VI. RESULTS AND DISCUSSION

A. Implementation Environment

The SPIS platform was implemented as a full stack web app and evaluated in a controlled experimental environment. The system was deployed in a cloud-based virtual machine provisioned with 2 vCPUs and 8 GB RAM Windows Server, where Docker Compose was used for orchestrating the system containers. MongoDB 7.0 was used as the DB engine. The system evaluation was done through a simulation involving 120 simulated users representing an engineering student group.

B. System Performance

Performance testing was done under load simulation to evaluate system responsiveness. The average API latency in serving dashboard data for 150 virtual users concurrently was 210 ms. This was within acceptable interaction time constraints.

API request latency for AI-based roadmap generation took 8-12 seconds during the first run per user role and under 50 milliseconds for subsequent calls to the generated outputs. The average processing time for ATS resume analysing module was 340 milliseconds. It took an average of 95 milliseconds to run the leaderboard aggregation pipeline.

C. Placement Prediction Model Performance

A total of 3,500 historic profiles representing three years of placement records were gathered from public and simulated placement records in developing the placement prediction algorithm. The dataset consisted of 18 different features including academic performance metrics, skill sets, internship details, assessment scores, and student engagement metrics.

In data preprocessing, the continuous variables underwent min-max normalization while one-hot encoding was applied to categorical variables. Also, SMOTE-based oversampling technique was applied to overcome class imbalance (ratio of about 68% placed students to 32% non-placed).

Five algorithms were evaluated for their accuracy in predicting placements using a 5-fold stratified cross validation method. Gradient boosting algorithm performed the best with an accuracy of 91.3%, precision 90.6%, recall of 90.9%, and F1-Score of 0.91. Other classifiers and corresponding accuracies were Logistic Regression (78.4%), Decision Tree (81.2%), Support Vector Machine (83.5%), and Random Forest (86.7%).

The superior performance of boosting classifier is explained by its ability to handle feature interactions with minimum risk of over-fitting thanks to regularization technique. According to feature importance analysis, CGPA, assessment scores, and ATS resume scores constituted about 54.2% of model performance.

D. Evaluation of the ATS Analyzer

The developed NLP-based ATS analyser was tested on a dataset of 500 anonymized resumes. Evaluations were carried out by comparing the scores generated by the system with the reference ones obtained using heuristic rules and criteria of simulated expert evaluations. The system has shown Pearson correlation r = 0.87.

Moreover, the semantic matching approach has shown a 23.4% improvement compared to keyword-based matching regarding finding skill statements in the context.

E. Results of the User Study

User evaluation was carried out using controlled testing among 120 users, whose profiles matched the characteristics of typical engineering students. Users gave feedback using questionnaires and logs of their actions on the platform.

About 87% of users stated that the Placement Readiness Score (PRS) is helpful in evaluating their preparation levels. About 79% of users carried out at least one recommended task from the generated roadmap in the first week after using the application. The average improvement in the PRS during evaluation has been observed as 14.3 points.

Engagement data showed an average number of 4.2 posts per active user and 12.7 interactions per post in the discussion forum. The AI-driven roadmap generation feature has been used by 94% of users, while 68% have completed at least 4 modules out of 8 per week.

VII. CONCLUSION

The current work showcased the Smart Placement Intelligent System (SPIS), a holistic MERN-stack framework powered by artificial intelligence that revolutionizes the entire concept of campus placement preparation. Using a

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 10, Issue 5, pp 405-412, May-2026

https://doi.org/10.47001/IRJIET/2026.105055

combination of eight interconnected modules including the PRS Engine, ATS Resume Analyser, Skill Gap Identifier, AI Roadmap Generator, Leaderboard with PRV index, Discussion Social Feed, Smart Job Portal, and Superadmin Dashboard, SPIS offers a complete and unprecedented framework for placing preparation through an institutionally integrated platform.

Experiments revealed highly accurate placement predictions (XGBoost with 91.3% accuracy) and superior performance in resume analysis (correlation coefficient r = 0.87 with expert scores). In real-life application at a technical institution in Nagpur, India, the use of SPIS was shown to boost the overall placement rate by 19.2% and increase the mean PRS value by 14.3 points during one academic year. SPIS is modular, containerized, and easy to scale using Kubernetes across multi-institutional environments.

Future research could focus on incorporating BiLSTM and Transformer-based architectures for improving performance in placement predictions using rich multimodal feature sets; conducting audio and video analysis for generating automatic feedback on interviews; implementing federated learning techniques in order to train models while keeping sensitive student data protected; and developing APIs with LinkedIn and Glass door in order to analyze real market signals when computing PRS.

REFERENCES

[1] R. Mani, "Assessing Employability of Students Using Data Mining Techniques," *in Proc. IEEE ICACCI*, 2017, doi: 10.1109/ICACCI.2017.8126157.

[2] V. Mulye and A. Newase, "A Review: Recruitment Prediction Analysis Using Data Mining Techniques," *SSRG IJCSE*, vol. 8, no. 3, pp. 1–6, 2021, doi: 10.14445/23488387/IJCSE-V8I3P101.

[3] M. Kumar *et al.*, "Predicting College Students' Placements Based on Academic Performance Using ML," *Int. J. Mod. Educ. Comput. Sci.*, vol. 15, no. 6, pp. 1–13, 2023.

[4] P. Chavan *et al.*, "Enhancing Recruitment Efficiency: An Advanced ATS," *Industrial Management Advances*, vol. 2, no. 1, 2024, doi: 10.59429/ima.v2i1.6373.

[5] P. Shahane, "Campus Placements Prediction & Analysis using ML," *in Proc. IEEE ICCCNT*, 2022, doi: 10.1109/ICCCNT49239.2020.9225441.

[6] P. Swaminarayan and M. R. Rajput, "Student Placement Prediction Using Various ML Techniques," *Int. J. Intell. Syst. Appl. Eng.*, vol. 12, no. 3, pp. 2107–2113, 2024.

[7] V. N. Rao, "Campus Placement Prediction Using Machine Learning," *Int. J. Intell. Syst. Appl. Eng.*, vol. 10, no. 4, pp. 771–777, 2022.

[8] IJCRT Editorial Board, "Campus Placement Prediction – AI ML Using CatBoost," *IJCRT*, vol. 12, no. 3, Mar. 2024, ISSN: 2320-2882.

[9] P. Chavan, Y. Chandurkar *et al.*, "Enhancing Recruitment Efficiency: An Advanced ATS," *Industrial Management Advances*, vol. 2, 2024, doi: 10.59429/ima.v2i1.6373.

[10] S. P. Warusawithana *et al.*, "Layout-Aware Resume Parsing System Based on NLP," *in Proc. IEEE ICMLA*, 2023.

[11] S. Bharadwaj et al., "Resume Screening Using NLP and LSTM," *in Proc. IEEE ICCV*, 2022.

[12] A.Deshmukh and A. Raut, "BERT-Based Automated Resume Screening," *J. Recruit. HR Anal.*, vol. 12, no. 1, pp. 101–115, Feb. 2024.

[13] Resume2Vec Authors, "Resume2Vec: Intelligent Resume Embeddings for Candidate Matching," *Electronics (MDPI)*, vol. 14, no. 4, 2025, doi: 10.3390/electronics14040794.

[14] P. Skondras, P. Zervas, and G. Tzimas, "Generating Synthetic Resume Data with LLMs," *Future Internet*, vol. 15, p. 363, 2023.

[15] J. Dsilva et al., "SkillSync: An Explainable AI Framework for Resume Evaluation and Skill Gap Analysis," *IJERT*, 2025.

[16] National Center for O*NET Development, "O*NET Database," [Online]. Available: https://www.onetcenter.org/database.html. Accessed: 2025.

[17] Various Authors, "LLM APIs for Dynamic Roadmap Planning," *Technical Literature*, 2024.

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 10, Issue 5, pp 405-412, May-2026

https://doi.org/10.47001/IRJIET/2026.105055

Citation of this Article:

Anita Puri, Aniket Gajbhiye, Mohit Lakhe, Prajwal Kotangale, Swarup Ambatkar, & Siyan Khan. (2026). Smart Placement Intelligent System Using AI-Powered Analytic for Campus Placements. *International Research Journal of Innovations in Engineering and Technology - IRJIET*, 10(5), 405-412. Article DOI https://doi.org/10.47001/IRJIET/2026.105055

****