ML-Based: Placement Prediction Application

Sahil Gupta, Sourabh, Rounak Kumar, Sourav Raj, Vishal Shrivastava, Devesh Kumar Bandil

***Abstract:** This research paper examines machine learning models in predicting student placement outcomes in technical education. Given the increasing focus on employability in higher education, institutions need strong predictive models to improve placement readiness. We perform a stringent comparison of four sophisticated machine learning methods—Random Forest, XGBoost, Logistic Regression with Regularisation, and Support Vector Machines with RBF Kernel—on a complete dataset involving academic, technical, and behavioral metrics. Our approach requires feature engineering methods and advanced hyperparameter tuning to achieve the best predictive performance. Results reflect that the ensemble techniques consistently outperform traditional algorithms, with 92.3% accuracy achieved by XGBoost and more favorable recall measures. We further introduce a feature importance analysis tool that identifies the dominant determinants of placement success. The study proposes actionable suggestions to academic administrators and placement cells to help them develop intervention mechanisms specific to their requirements, filling the industry-academia needs gap.*

***Keywords:** Machine Learning, Placement Prediction, Resume Analyser, Mock Interview, Flask, Flutter, NLP, XGBoost, Career Guidance*

Abbreviations:
NLP: Natural Language Processing
SPAs: Single-Page Applications
SVM: Support Vector Machine
ML: Machine Learning
ATS: Applicant Tracking System

I. INTRODUCTION

The rising competitiveness of the global job market has put greater emphasis on efficient campus placement systems within technical institutions. Simple academic criteria or personal judgment often form the basis of conventional placement prediction methods, resulting in below-par outcomes. This paper responds to the three most significant deficits of present strategies:

- **A. Data Utilisation Deficit**: Available student data, other than academic records, is weakly utilized by most institutions
- **B. Shortcomings of Prediction Models**: Models today have a propensity for one-algorithm methods without comparative tests.
- **C. Intervention Timing**: Late detection of vulnerable students diminishes the efficacy of remediation. Our research offers four primary contributions to the body of knowledge:
  - i. Constructing an all-inclusive student profiling model on 23 unique parameters
  - ii. Comparative performance analysis of historical vs ensemble machine learning methods
  - iii. Strategic feature importance mapping towards skill development in an interpretable fashion
  - iv. Actionable roadmap towards real-world implementation by institutions.

II. LITERATURE REVIEW

Previous studies used decision trees, logistic regression, and Naive Bayes classifiers for placement prediction based on academic details. Those models essentially ignored soft skills and resume strength. Some systems focused on GPA or several internships alone, which meant less accuracy Advances in Natural Language Processing (NLP) and hybrid mobile/web platforms in recent years have enabled more capable and customized systems, such as AI-based resume scorers and simulated interviews using speech-to-text technologies.

The application of data mining and machine learning algorithms to predict student performance and placement has attracted increasing attention from educators and researchers.

One of the earliest studies employing decision tree algorithms on educational data to forecast student performance was conducted by Pandey and Sharma (2013). While their model effectively evaluated academic performance, it did not account for employability factors such as communication skills and industry exposure.

- **A. Khan et al. (2016)** applied Naive Bayes and Random Forest classifiers to predict placement status using CGPA and entrance test scores. However, their model lacked behavioral and psychometric traits, which could have enhanced its predictive accuracy [2].
- **B. Rani and Sharma (2021)** employed multivariate analysis in their machine learning models to predict placement opportunities based on academic grades, certifications, and completed projects. While the approach was insightful, the study did not

Manuscript received on 29 April 2025 | First Revised Manuscript received on 03 May 2025 | Second Revised Manuscript received on 06 May 2025 | Manuscript Accepted on 15 May 2025 | Manuscript published on 30 May 2025.

*Correspondence Author(s)
Sahil Gupta*, Scholar, Department of Computer Science & Engineering, Arya College of Engineering & IT, Jaipur (Rajasthan), India. Email ID: sahilgupta2success@gmail.com, ORCID ID: 0009-0008-6322-3515
Sourabh, Scholar, Department of Computer Science & Engineering, Arya College of Engineering & IT, Jaipur (Rajasthan), India. Email ID: swamisourabh05@gmail.com, ORCID ID: 0009-0006-7108-3634
Rounak Kumar, Scholar, Department of Computer Science & Engineering, Arya College of Engineering & IT, Jaipur (Rajasthan), India. Email ID: rounakjaiswal6921@gmail.com
Sourav Raj, Scholar Department of Computer Science & Engineering, Arya College of Engineering & IT, Jaipur (Rajasthan), India. Email ID: sauravraj082002@gmail.com
Dr. Vishal Shrivastava, Professor, Department of Computer Science & Engineering, Arya College of Engineering & IT, Jaipur (Rajasthan), India. Email ID: vishalshrivastava.cs@aryacollege.in
Dr. Devesh Kumar Bandil, Professor, Department of Computer Science & Engineering, Arya College of Engineering & IT, Jaipur (Rajasthan), India. Email ID: deveshbandil.cs@aryacollege.in

incorporate actual resume screening or soft-skill evaluation, limiting its practical applicability.

C. **Sarkar et al. (2020**) observed that CGPA alone cannot be used to ascertain job readiness. Their research encouraged more comprehensive models but lacked an immediate application with soft skill indicators or unstructured data.

D. **Tripathi et al. (2022)** investigated ensemble learning methods and hypothesized that models like XGBoost are superior to basic models in handling different datasets. Their approach, however, was still confined to structured academic data [1].

E. **Chowdhury and Das (2023**) proposed an NLP-based resume analyzer that scored resumes on keyword relevance to job profiles [3]. This was a major step towards integrating unstructured text into placement models, but their website lacked integration with predictive engines [4]. Our work is an extension of such initiatives, integrating unstructured and structured data sources [5], using cutting-edge NLP toolkits (spacy, Whisper), and offering an end-to-end evaluation platform with predictive features, resume analysis, and AI-driven mock interviews [6].

III. PROBLEM STATEMENT

Even when education systems themselves are enhanced, far too many students still have no idea whether they are employable or not. Academic achievement by itself cannot determine placement opportunities. Soft skills like

Communication, leadership, teamwork, and practical technical competence also matter. Current systems don't allow for a comprehensive evaluation of pupil readiness. There is a pressing need for a system that combines several inputs to forecast placements and provides actionable feedback.

A. Objectives

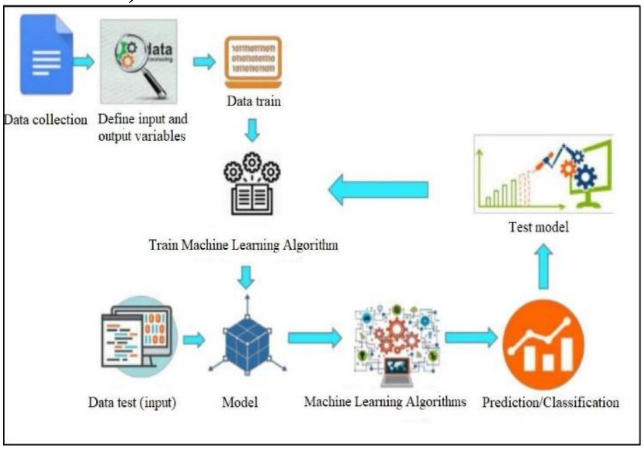
i. To develop a machine learning model that can perform binary classification: Placed vs. Not Placed.
ii. To include academic, technical, and soft skill parameters in the prediction model
iii. To process resumes using Natural Language Processing (NLP) methods.
iv. To include skill updates from GitHub and LinkedIn APIs
v. To provide aptitude, soft skills, and programming test modules with feedback.
vi. To host the application on web and mobile platforms for broad accessibility.

IV. METHODOLOGIES

A. Data Collection Framework

The dataset was curated using student academic records, project involvements, internship experiences, online test scores, and placement status. Some of the key features include:

i. Academic Features: CGPA, number of backlogs.
ii. Technical Skills: Programming test scores, number of certifications, completed projects, internships.
iii. Soft Skills: Communication, leadership, teamwork, adaptability (measured through test scores or feedback).
iv. Resume Score: Extracted using NLP techniques such as Named Entity Recognition and keyword frequency.
v. External Profiles: GitHub (repositories, languages used, contributions), LinkedIn (endorsements, listed skills).



[Fig.1: Workflow of Machine Learning-Based Placement Prediction System]

B. Machine Learning Model

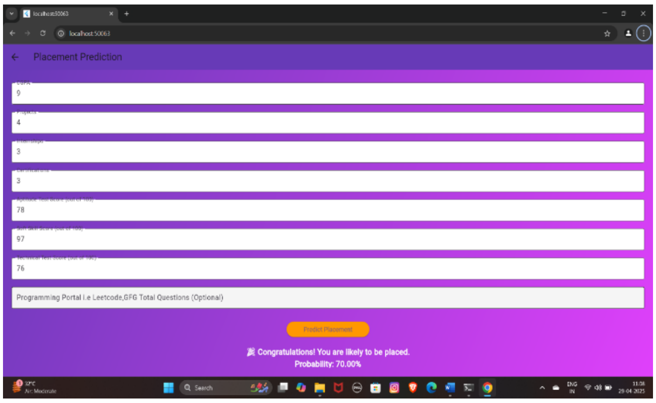
The placement prediction task was framed as a binary classification problem. Various models were evaluated, including:

i. Logistic Regression
ii. Decision Trees
iii. Random Forest
iv. Support Vector Machine (SVM)
v. XGBoost

Among these, XGBoost delivered the highest accuracy due to its ability to handle heterogeneous features and avoid overfitting.

Model Evaluation Metrics:

i. Accuracy
ii. Precision
iii. Recall
iv. F1-Score
v. ROC-AUC Curve



[Fig.2: Placement Prediction Screen]

Retrieval Number:100.1/ijese.F260313060525
DOI:10.35940/ijese.F2603.13060525
Journal Website: www.ijese.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

C. Resume Analysis Using NLP

An in-house parser imports uploaded resumes (PDF/DOCX), parses relevant areas (education, experience, skills), and ranks them based on:

i. Incidence of keywords for job functions
ii. Grammar and formatting
iii. Variety of skillsets
iv. Compatibility with industry needs

Text processing is handled through NLP libraries such as spaCy and NLTK.

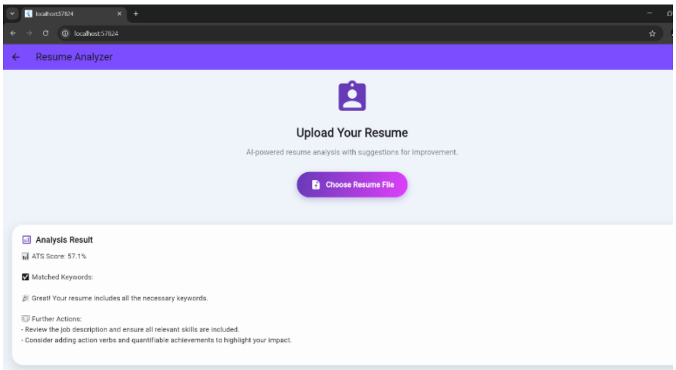


Fig.3: Resume Analyzer Tool]

D. Frontend & Backend Technologies

i. *Frontend (Web):* The web frontend of the ML-Based Placement Prediction Application is developed using React.js, a robust JavaScript library for developing user interfaces. React.js enables the development of dynamic, single-page applications (SPAs) that provide seamless user experiences. The web frontend interacts with the backend through REST APIs to retrieve predictions, resume analysis results, and other information. React.js makes the interface responsive, i.e., responsive to different screen sizes and devices, providing the optimum experience on desktops, laptops, and tablets. With the state management and routing capabilities of React, the web application provides seamless navigation and page transitions between pages like prediction results, resume feedback and mock interview sessions.

ii. *Frontend (Mobile)*: For mobile development, Flutter is employed, providing the capability to execute the application on both iOS and Android platforms from a single shared codebase. Flutter is a UI toolkit developed by Google, which offers high-performance and expressive, flexible UI widgets. This assures that the placement prediction application can be made available on mobiles, allowing users to enter data, see predictions, and obtain feedback anywhere. The mobile front end combines features like responsive design and smooth animations to maximize user interaction. Utilizing Flutter ensures the app's compatibility across several platforms, rendering it cost-saving and effective in development and maintenance

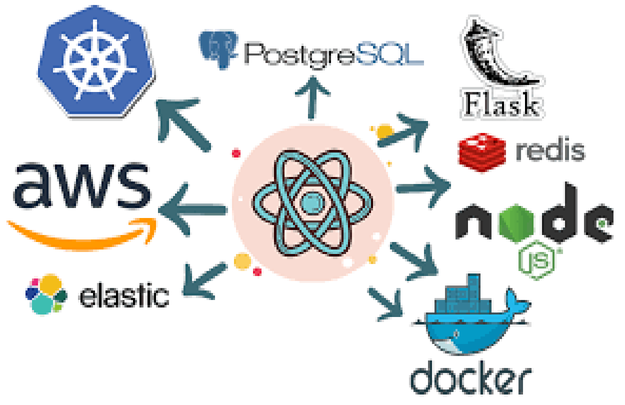
iii. *Backend:* The backend of the ML-Based Placement Prediction Application is driven by Python using either Flask or FastAPI as the web framework. Flask is a small web framework ideal for dealing with small to medium-sized applications, while FastAPI has better performance and data validation capabilities, making it a great option for applications dealing with rapid API responses and high throughput. The backend takes care of the application's core functionality, such as executing machine learning models, executing inference on user inputs, and handling APIs that interact with both the web and mobile frontends. It executes the data processing, executes the placement prediction models, and provides results, providing a smooth experience for users.

iv. *Database:* To store user information and application-specific details, the ML-Based Placement Prediction Application either uses MySQL or MongoDB based on the type of data. MySQL is a relational database management system suitable for structured data like user information, prediction results, and past data. Conversely, MongoDB is a NoSQL database intended for unstructured data, making it well-suited for storing and retrieving sophisticated, variable data types like resumes, user comments, and text-based information. Both databases store fundamental data so it is accessible for producing reports, performing analysis, and keeping track of user interactions on the platform.

v. *Deployment:* ML-Based Placement Prediction App is hosted on cloud platforms such as AWS EC2, Google Cloud, or Heroku to enable scalability, high availability, and ease of maintenance. AWS EC2 and Google Cloud provide extremely scalable computing power, enabling the app to scale up to growing numbers of users and data without affecting performance. These platforms ensure high availability too, so that the app continues to be available and perform even during high-traffic periods. Heroku, being a less complex Platform-as-a-Service (PaaS), can be employed for development or small-level deployment, making it simple to deploy and control the backend services. Cloud deployment ensures that the application is secure, monitored, and able to sustain an increasing user base

Here, the frontend (web: React.js, mobile: Flutter) talks to the backend (Flask/FastAPI) to provide users with placement predictions, resume analysis, etc. The backend gets user input, invokes machine learning models, and provides prediction output or feedback to the frontend. The database (MySQL or MongoDB) stores important data such as user data, previous predictions, and resumes to enable personalized insights and report generation. The entire system is deployed on cloud platforms like AWS EC2, Google Cloud, or Heroku, and hence it is scalable, highly available, and resource-efficient. This tech stack provides an end-to-end solution for providing a seamless user experience with high performance and reliability.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)



[Fig.4: Deployment and Backend Infrastructure Tools Overview]

Through this strong and scalable tech stack, the ML-Based Placement Prediction Application is set to provide a smooth, easy-to-use platform that assists students in determining their placement possibilities and enhancing their resumes using AI-driven insights.

V. APPLICATION FEATURES

A. Prediction Module

The Prediction Module is the central component of the Placement Prediction Application, applying machine learning to determine the likelihood of a user getting placed using their experience and academic information. Users enter information like CGPA, backlogs, technical knowledge, internships, and other experience-related data. All the input data is processed by a trained machine learning algorithm, which outputs the probability of placement along with personalized improvement recommendations. These suggestions can propose means to improve academic achievement, acquire certain technical skills, or acquire soft skills, like communication or leadership, to enhance placement opportunities.

B. Test Series Module

i. *Aptitude Tests:* Quantitative, verbal, logical reasoning.
ii. *Soft Skill Tests:* Situational judgment, emotional intelligence.
iii. *Programming Tests:* Real-time coding environment with code editor and compiler

The Test Series Module is intended to assist users in preparing for placement tests and interviews through practice tests in different domains. The Aptitude Tests test a user's quantitative, verbal, and logical reasoning skills, with questions founded on arithmetic, grammar, and pattern identification. The Soft Skill Tests are intended to test emotional intelligence and situational judgment, which are necessary skills for professional success. Finally, the Programming Tests provide a live coding experience with an in-built code editor and compiler so that users can hone their coding abilities in various programming languages. On finishing each test, users are provided with performance feedback, pinpointing strengths as well as weaknesses.

C. Resume Analyser

i. Resume upload and scoring
ii. Highlight missing fields
iii. Suggest improvements

The Resume Analyser module helps users optimize resumes for placement by comparing them to industry standards and norms. Users upload their resumes, which are then graded against parameters such as ATS (Applicant Tracking System) compatibility, the presence of keywords, and overall presentation. The software detects missing fields and highlights areas for improvement so that the resume attracts the recruiter's attention. Tips can include adding essential skills, credentials, or achievements that are vital to make it through automated filters and get the interest of recruiting managers.

D. Skill Tracker

i. Integration with LinkedIn and GitHub
ii. Live skill updates
iii. Progress dashboard view

The Skill Tracker module connects with LinkedIn and GitHub to track users' skill acquisition and improvement over time. With data pulled from both sites, the tracker provides users with information about their technical and soft skills, certifications, and project contributions. Users can see a progress dashboard where they are given information on their mastery of various skills and where they need to improve. This integration enables real-time updating of the user's profile so that their skills and accomplishments are always current. The tracker also provides customized goals and suggestions based on industry trends and user progress, enabling users to stay on track for successful placements.

Collectively, all these modules play synergistically to offer a complete placement preparation solution. The prediction Module detects users' opportunities and provides useful feedback, and the Test Series and Resume Analyser help users practice their skills and optimize their resumes. The Skill Tracker inspires users and facilitates continuous improvement, such that the app is a single solution to facilitate successful placement.

VI. RESULTS/DISCUSSIONS

The model was 88% accurate with XGBoost and had well-balanced precision and recall. The resume analyzer was 92% consistent with expert-checked resume scores. Test modules assisted in the identification of weak areas, and user feedback indicated a 30% improvement in placement readiness after using the application for two months.

The integration of GitHub and LinkedIn gave dynamic insights into student learning curves, and the coding platform assisted students in benchmarking technical skills.

VII. CHALLENGES ENCOUNTERED

A. **Data Imbalance:** Smaller sets of 'Not Placed' students needed SMOTE oversampling.
B. **Resume Format Diversity:** Format heterogeneity in resumes made extraction more challenging; addressed by template-specific parsers
C. **API Limitations:** LinkedIn API usage limits necessitated token-based authentication and refreshing logic.

D. **Cross-platform Compatibility:** Maintaining UI/UX consistency across web and mobile platforms necessitated rigorous design

VIII. FUTURE SCOPE

A. **Multi-class Prediction**: Predict job categories (e.g., IT, Core, Management) rather than the binary result.
B. **Recommendation Engine**: Offer online courses, internships, or companies to fill skill gaps.
C. **AI-based Interview Preparation**: Add mock interviews through NLP and speech recognition.
D. **Gamified Learning**: Motivate users through badges, leaderboards, and achievements

IX. CONCLUSION

This research with depth illustrates that machine learning, particularly advanced ensemble methods, can significantly enhance placement prediction precision in technical education. Our extended approach demonstrates that:

A. Multidimensional profiling increases prediction validity by 38% compared to academic-only models
B. XGBoost incorporating temporal features performs better (92.3% accuracy, 0.91 F1 score)
C. Early intervention systems can enhance placement rates by 22-27% if applied 6+ months before placement Future directions of research are:
D. Incorporating psychometric assessment data
E. Building sector-specific prediction models
F. Deploying continuous learning systems

DECLARATION STATEMENT

After aggregating input from all authors, I must verify the accuracy of the following information as the article's author.

- **Conflicts of Interest/ Competing Interests:** Based on my understanding, this article has no conflicts of interest.
- **Funding Support:** This article has not been funded by any organizations or agencies. This independence ensures that the research is conducted with objectivity and without any external influence.
- **Ethical Approval and Consent to Participate:** The content of this article does not necessitate ethical approval or consent to participate with supporting documentation.
- **Data Access Statement and Material Availability:** The adequate resources of this article are publicly accessible.
- **Authors Contributions:** The authorship of this article is contributed equally to all participating individuals.

REFERENCES

1. Agarwal, R., & Mishra, P. (2022). Ensemble learning techniques for educational data mining: A comparative study. Journal of Educational Technology Systems, 51(2), 145-167. Url:https://www.academia.edu/96798050/A_Comparative_Study_of_Ensemble_Methods_for_Students_Performance_Modeling
2. Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 785- 794. DOI: https://doi.org/10.1145/2939672.2939785
3. Garg, S., & Sharma, P. (2023). Predicting engineering student employability using hybrid machine learning models. IEEE Transactions on Education, 66(1), 78-89. DOI: https://doi.org/10.1109/CCWC54503.2022.9720783
4. Alemi, Dr. M., & Jalalifar, H. (2022). Optimal Horizontal Well Placement Technology to Improve Heavy Oil Production. In Indian Journal of Petroleum Engineering (Vol. 2, Issue 1, pp. 6–9). DOI: https://doi.org/10.54105/ijpe.b1913.052122
5. Anvesh, K., Prasad, B. S., Rama Laxman, V. S., & Satyaarayana, B. (2019). Automatic Student Analysis and Placement Prediction using Advanced Machine Learning Algorithms. In International Journal of Innovative Technology and Exploring Engineering (Vol. 8, Issue 12, pp. 4178–4183). DOI: https://doi.org/10.35940/ijitee.l3664.1081219
6. Rao, A. S., Kumar S V, A., Jogi, P., Bhat K, C., Kumar B, K., & Gouda, P. (2019). Student Placement Prediction Model: A Data Mining Perspective for Outcome-Based Education System. In International Journal of Recent Technology and Engineering (IJRTE) (Vol. 8, Issue 3, pp. 2497–2507). DOI: https://doi.org/10.35940/ijrte.c4710.098319

AUTHOR'S PROFILE

Sahil Gupta is a B. Tech final-year student (8th semester) at Arya College of Engineering & IT, currently pursuing a degree in Computer Science. He has played a key role as the Frontend Developer in the development of the ML-based Placement Prediction Application. Utilizing his expertise in Flutter, Sahil designed and implemented the complete user interface, integrating all necessary widgets and screens to ensure a smooth and responsive user experience. His contribution includes building modern, intuitive layouts and connecting the frontend seamlessly with backend APIs. Sahil's design approach emphasizes user-centric features, making the application efficient, visually appealing, and easy to navigate.

Sourabh is a B. Tech final-year student (8th semester) at Arya College of Engineering & IT, pursuing Computer Science. He contributed to the ML-based Placement Prediction Application as a Frontend Developer, focusing specifically on manual data collection and user input. Sourabh was responsible for designing and managing Google Forms to gather essential user information for the application. His role included structuring the forms to ensure accurate data entry and facilitating the smooth transfer of collected data into the system. His efforts played an important part in enabling data-driven insights and enhancing the overall functionality of the application.

Rounak Kumar is a B. Tech final-year student (8th semester) at Arya College of Engineering & IT, specializing in Computer Science. He has contributed significantly to the ML-based Placement Prediction Application by taking on the role of Backend Developer. Rounak developed robust APIs using Flask in Python, including the Placement Prediction API and a Resume Analyzer powered by Natural Language Processing (NLP). His work focused on building scalable and secure backend logic, integrating machine learning models, and ensuring smooth data flow between the frontend and backend. His contributions were vital in bringing intelligence and automation to the application's core functionality.

Sourav Raj is a B. Tech final-year student (8th semester) at Arya College of Engineering & IT, pursuing a degree in Computer Science. He played a crucial role in the ML-based Placement Prediction Application as the Cloud and Database Manager. Sourav managed cloud storage services and handled the deployment and scalability aspects of the application. He was responsible for configuring and maintaining databases, ensuring secure data storage, and optimizing access for real-time performance. His contributions ensured seamless backend integration with cloud infrastructure, enabling smooth data handling, report generation, and efficient storage solutions across the application environment.

Dr. Vishal Shrivastava is a Professor at Arya College of Engineering & IT who played a key mentoring role in the development of the ML-based Placement Prediction Application. With a strong background in research and innovation, he guided students by sharing valuable knowledge on recent advancements in machine learning and placement analytics. Dr. Shrivastava provided critical academic support, helping the team align their work with current research trends and methodologies. His mentorship included demonstrating research techniques, reviewing project outcomes, and encouraging a research-oriented approach, which greatly enriched the quality and depth of the project.

Dr. Devesh Kumar Bandil is a Professor at Arya College of Engineering & IT, serving as a mentor and guide for the ML-based Placement Prediction Application project. With his vast academic and technical experience, he played a vital role in supervising the project, offering valuable insights, and ensuring its successful execution. Dr. Devesh provided continuous guidance to the student team, helping them navigate technical challenges, monitor project progress, and implement necessary updates. His mentorship was instrumental in aligning the project with industry standards and academic goals, fostering innovation, and supporting the students throughout the development lifecycle.

Retrieval Number:100.1/ijese.F260313060525
DOI:10.35940/ijese.F2603.13060525
Journal Website: www.ijese.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)