International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 10 | Oct - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

HireFlow – Enterprise Campus Placement Tracker

Shreya Lavhale, Yadarthi Pund, Shruti Lende, Tanushree Wagh, Kartik Tawalare

Department of Computer Science & Engineering
P. R. Pote Patil College of Engineering & Management, Amravati, Maharashtra, India
Session 2026–27

Abstract— HireFlow – Enterprise Campus Placement Tracker is a centralized web-based platform designed to streamline the campus placement lifecycle and reduce dependence on spreadsheets, paper records, and disconnected communication. It provides role-based access for TPO/Admin, DTPC/Faculty, and Students and integrates student profiles, companies, placement drives, eligibility verification, applications, interviews, results, notifications, analytics, and reports. The current architecture uses Java 21, Jakarta JSP/Servlets, HTML, CSS, JavaScript, JDBC, MySQL, Maven, and Apache Tomcat in a monolithic layered MVC design. The system centralizes placement data and applies predefined criteria such as branch, CGPA, backlogs, and graduation year to support eligibility verification. The overall objective is to reduce repetitive administrative work, improve transparency and communication, and provide a structured digital environment for campus recruitment management.

Keywords— Campus Placement, Placement Management, Java, JSP, Jakarta Servlets, MySQL, JDBC, Role-Based Access Control, Eligibility Verification, Application Tracking, Analytics.

I. INTRODUCTION

Campus placement connects students with employment opportunities and enables organizations to recruit graduates. The process involves registration, profile verification, company coordination, drive creation, eligibility checking, applications, interviews, result publication, and reporting. When these activities are distributed across spreadsheets, forms, email, and separate records, maintaining consistency and timely communication becomes difficult.

HireFlow addresses this operational gap through a centralized web application. It brings major placement activities into one platform and provides role-specific access to TPO/Admin, DTPC/Faculty, and Students. The design emphasizes a connected workflow from student registration to final placement and analytics.

II. MOTIVATION AND PROBLEM STATEMENT

The motivation for HireFlow arises from repetitive data entry, manual eligibility verification, scattered student records, delayed notifications, difficulty tracking applications, and time-consuming report preparation. Students need a reliable way to identify relevant opportunities and follow recruitment status, while placement officers need centralized information for coordination and decision-making.

The problem is the absence of a centralized, secure, and role-based system capable of managing student information, companies, placement drives, eligibility, applications, interviews, results, notifications, and reports throughout the campus placement lifecycle.

III. OBJECTIVES

- Develop a centralized platform for students, companies, placement drives, applications, and recruitment activities.
- Provide secure role-based access for TPO/Admin, DTPC/Faculty, and Students.
- Support eligibility verification using branch, CGPA, backlog, and graduation-year criteria.
- Reduce manual effort in verification, application monitoring, interview coordination, and result publication.
- Improve transparency through application, interview, and placement-status tracking.
- Provide analytics and reports for placement monitoring and administrative decision-making.
- Protect placement information using authentication, password hashing, authorization, and controlled access.

IV. LITERATURE REVIEW

Prior placement-management studies show that web-based systems can reduce paperwork, organize placement records, monitor recruitment progress, and improve communication. Some systems emphasize record management and tracking, while others focus on analysis, recruitment interaction, or intelligent job matching. HireFlow combines

centralized placement management with role-based access, eligibility verification, application and interview tracking, notifications, analytics, and reporting.

Ref.	Study / Year	Main Focus	Relevant Gap
[1]	Patil et al. (2023)	Web placement records, company details and communication.	Limited intelligent/automated matching.
[2]	Srivastava et al. (2024)	Placement analysis and tracking.	Mainly analysis and tracking.
[3]	Sowmya et al. (2026)	Web recruitment and placement management.	Limited advanced personalization.
[4]	Placement Interaction-Fedf	Placement interaction and information sharing.	Limited advanced tracking.
[5]	Kumar et al. (2025)	Intelligent job matching and guidance.	Needs reliable skill data.
[6]	John David et al. (2026)	Centralized placement tracking.	Limited intelligent analytics.

V. PROPOSED SYSTEM

HireFlow follows a centralized, role-based, workflow-driven design. TPO/Admin has institution-wide control over placement operations. DTPCs coordinate activities for their assigned departments and verify student information. Students maintain profiles, upload resumes, view eligible opportunities, apply for drives, and track recruitment progress.

A. TPO / Admin Module

- Manage departments, students, companies, and placement drives.
- Define eligibility criteria and required skills.
- Publish and monitor placement opportunities.
- Monitor applications, shortlist candidates, schedule interviews, and publish results.
- Access analytics and reports.

B. DTPC / Faculty Module

- Verify department-wise student profiles and documents.
- Approve or reject student information where applicable.
- Monitor departmental drives, applications, interviews, placements, notifications, and reports.

C. Student Module

- Register and authenticate.
- Maintain personal, academic, and placement information.
- Upload resume and manage skills.
- View eligible placement drives and apply.
- Track applications, interviews, and results.

VI. SYSTEM WORKFLOW AND ARCHITECTURE

The workflow connects student registration and profile verification with company and drive management, eligibility verification, applications, shortlisting, interviews, result publication, placement tracking, analytics, and reports. A central MySQL database stores information shared across these activities.

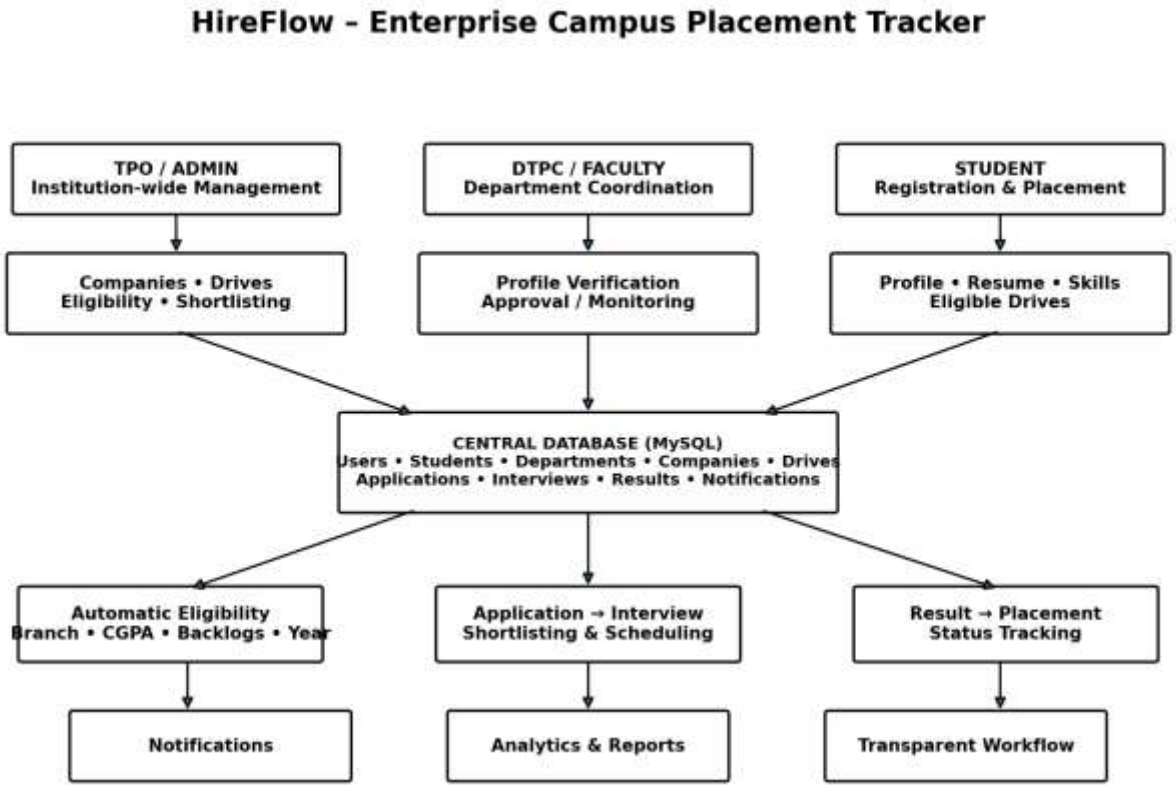


Fig. 1. HireFlow system workflow and interaction among major stakeholders.

A. Application Architecture

The current implementation follows a monolithic layered MVC architecture. The presentation layer uses JSP, HTML, CSS, and JavaScript. Jakarta Servlets handle requests in the controller layer. Service classes organize business logic, DAO classes perform database operations through JDBC, and utility/security components support database connectivity and password verification. Apache Tomcat provides the web runtime.

Presentation → Controller → Service → DAO → JDBC → MySQL

VII. DATABASE DESIGN

MySQL is used because the system contains strongly related entities and requires referential integrity. The database includes users, students, departments, tpo_profiles, dtpc_profiles, companies, placement_drives, applications, interviews, results, student_skills, drive_eligibility, drive_eligible_departments, drive_required_skills, notifications, and audit_logs.

The users table provides common authentication identity and role information. Student, TPO, and DTPC profiles are associated with user accounts. Drives reference companies, applications connect students with drives, interviews are associated with applications, and results record recruitment outcomes.

VIII. ELIGIBILITY AND RECRUITMENT PROCESS

A placement drive can define requirements such as eligible departments/branches, minimum CGPA, backlog constraints, graduation year, and required skills. HireFlow uses these criteria to determine whether a student can view or apply for an opportunity. After application, the workflow can progress through monitoring, shortlisting, interview scheduling, result declaration, and placement-status tracking.

IX. TECHNOLOGY STACK AND IMPLEMENTATION

Component	Technology
Programming Language	Java 21
Frontend	JSP, HTML5, CSS3, JavaScript, Bootstrap 5
Backend	Jakarta Servlets and JSP

Database	MySQL
Connectivity	JDBC
Build	Apache Maven
Web Server	Apache Tomcat
IDE	NetBeans IDE
Database Tool	MySQL Workbench
Security	BCrypt password hashing and role-based authorization
Processing/Reports	Apache POI and Apache PDFBox

X. SECURITY AND ACCESS CONTROL

Security is addressed through authenticated accounts, BCrypt-based password hashing, role-based access control, and server-side authorization. Account roles determine access to functional areas. Department-level restrictions are relevant for DTPC operations, while TPO operations are institution-wide. Session management maintains authenticated state during web requests.

XI. EXPECTED SYSTEM OUTCOMES

- Centralized management of placement information.
- Reduced repetitive administrative work and paperwork.
- Faster eligibility verification and better visibility of relevant drives.
- Improved coordination among TPO/Admin, DTPC/Faculty, and Students.
- Transparent application and placement-status tracking.
- Timely notifications for placement events.
- Analytics and reports for administrative monitoring.
- A structured foundation for future recruitment-management features.

XII. CHALLENGES AND LIMITATIONS

Key challenges include maintaining accurate student data, enforcing role and department-level authorization, applying eligibility rules consistently, protecting credentials and placement information, managing notifications reliably, and maintaining performance as the number of users and drives grows.

The current scope is focused on campus placement rather than general off-campus recruitment. Eligibility decisions depend on the accuracy of stored student and company information. External job-portal integration, video-interview platforms, and advanced AI-based recruitment capabilities are outside the initial scope.

XIII. CONCLUSION AND FUTURE SCOPE

HireFlow provides a centralized approach to campus placement management by connecting student profiles, company and drive management, eligibility, applications, interviews, results, notifications, and reporting within one workflow. The role-based design supports different responsibilities for TPO/Admin, DTPC/Faculty, and Students, while the layered Java architecture and MySQL database provide a structured foundation for development and maintenance.

Future work may extend resume intelligence, skill-based matching, external job-portal integration, richer communication services, video-interview integration, mobile support, and additional analytical capabilities.

REFERENCES

[1] R. Patil, R. Pawar, O. Salunke, and A. Gaigol, “Web Based Placement Management and Tracking System,” SSGM Journal of Science and Engineering, vol. 1, no. 1, pp. 72–76, 2023.

[2] J. Srivastava, A. K. Singh, and G. Kumar, “Placement Analysis and Tracking System,” TIJER – International Research Journal, vol. 11, no. 3, pp. 551–554, Mar. 2024.

[3] K. G. Sowmya et al., “A Web-Based Recruitment and Placement Management System,” IJRASET, vol. 14, no. 3, pp. 1329–1339, Mar. 2026.

[4] Placement Interaction-Fedf, online placement interaction document/platform.

[5] P. Kumar et al., "Intelligent Campus Placement System with Automated Job Matching and Guidance," IJPREMS, vol. 5, no. 11, pp. 625–629, Nov. 2025.

[6] S. John David et al., "Placement Tracking System," Iconic Research and Engineering Journals, vol. 9, no. 10, pp. 1611–1616, 2026.

AUTHOR CONTRIBUTIONS

Team Member	**Project Role**
Shreya Lavhale	Team Leader & Frontend Developer
Yadarthi Pund	Backend Developer
Shruti Lende	Backend Developer
Tanushree Wagh	Database Management & Frontend
Kartik Tawalare	Analysis & System Designing