Hirex-Recruitment and Placement Management System

[1]Mrs.Akula Sai Krishnaveni, [2]Korukonda Chandralekha, [3]Kothala SushmaSri Sunanda Bala, [4]Kakileti Anusha, [5]Geddada Shrudu Yashaswi

[1]Assistant Professor, Bonam Venkata Chalamayya Engineering College

[23456] UG Scholar, Bonam Venkata Chalamayya Engineering College

Abstract: Recruitment and placement management is a critical function in educational institutions, yet traditional manual processes for handling student records, recruited details, job postings, and application tracking are time consuming, error prone, and fragmented. To address these challenges, this work proposes Herix, a web based Recruitment Placement Management System that provides a centralized platform to automate core placement activities and support real time coordination among students, recruiters, and administrators. The system offers role based access, enabling students to create profiles, manage academy and skill details, upload resumes, and apply for jobs; recruiters register, maintain company profiles, post job openings with eligibility criteria, and view eligible candidates; and administrators to oversee user management, monitor activities, and generate reports through dedicated modules. This application is implemented by using Spring Boot with RESTful API (backend). MySQL with Spring Data JPA (database) and React/CSS/Bootstrap/JavaScript (frontend) deployed on Apache Tomcat this architecture ensures secure, scalable data management, real time notifications, and transparent tracking. Experimental deployment demonstrates reduced paperwork, minimizes the data redundancy, and improves efficiency/transparency for campus placements in engineering colleges.

Keywords: Recruitment, Placement, Management, Herix, Administrator, Spring Boot, REST, MySQL, CSS, React, Bootstrap, API, JPA.

I. INTRODUCTION

The Campus recruitment drives educational-industry connections, but manual student record handling, job notifications, and applications created in efficiencies like data errors, delay updates, and poor tracking[6]. Spreadsheets and notice boards lead to missed opportunities, while placement officers face heavy reporting workloads.

In this paper we are presenting Herix a web based recruitment and placement management system automating workflows for students (profiles, job applications), recruiters (postings, applicant views), and administrators (oversight, analytics)[8].

Problem statement: Manual systems suffer from error-prone data entry, fragmented communication via emails/notice boards, difficult application status tracking, and excessive admin effort demanding an integrated digital platform[9].

Scope and objectives: scope includes over authentication, profile/job/application management, automated notifications, and reporting for campus placements excludes advanced AI matching or external HR verification[9].

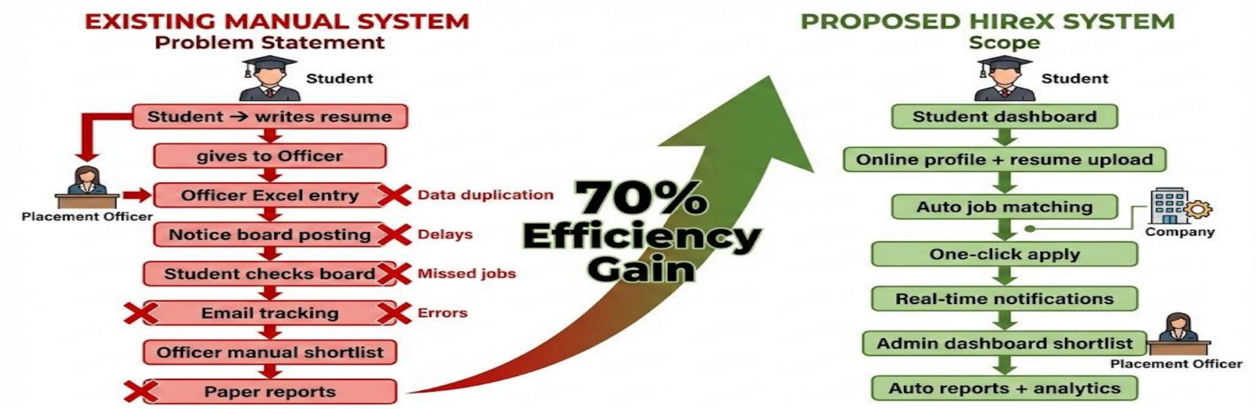
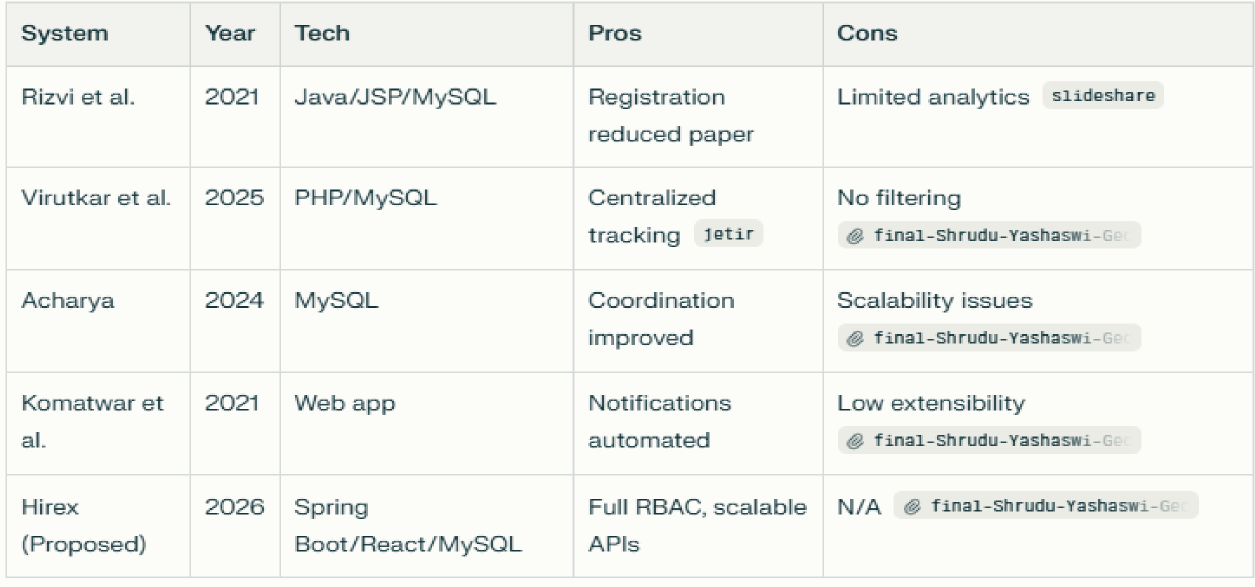


Fig 1: Comparison of Manual and HIReX Placement Workflows

II.LITERATURE SURVEY

Rizvi et al. (2021) proposed a Java/JSP/MySQL placement system for registration/resume upload, reducing paperwork but with limited analytics. Virutkar et al. (2025) built PHP/MySQL recruitment tracking, improving communication yet missing filtering. Acharya (2024) integrated profiles/postings in MySQL but ignored security/scalability. Komatwar et al. (2021) automated notifications centrally. Suitable for small scale only. Gurlhosur et al. (2024) added AI resume analysis (ML/Web UI) for better matching, though complex[12][13].



System	Year	Tech	Pros	Cons
Rizvi et al.	2021	Java/JSP/MySQL	Registration reduced paper	Limited analytics slideshare
Virutkar et al.	2025	PHP/MySQL	Centralized tracking jetir	No filtering final-Shrudu-Yashaswi-Ge
Acharya	2024	MySQL	Coordination improved	Scalability issues final-Shrudu-Yashaswi-Ge
Komatwar et al.	2021	Web app	Notifications automated	Low extensibility final-Shrudu-Yashaswi-Ge
Hirex (Proposed)	2026	Spring Boot/React/MySQL	Full RBAC, scalable APIs	N/A final-Shrudu-Yashaswi-Ge

Table.1

III. SYSTEM ANALYSIS

Traditional placement relies on spreadsheets/paper files for student data, notice boards/ emails for jobs, causing delays/duplication, Hirex centralizes via web platform.

Existing System: Manual/semei-automated methods use spreadsheets for records, notice boards for notifications, manual shortlisting - inefficient for scale[10].

Drawbacks

- Manual data handling causes errors/redundancy.
- No central DB hinders retrieval.
- Delayed notifications miss jobs.
- Time-consuming tracking/reports/security gaps.

Proposed System: Hirex automates via role-based web app: students register/apply, recruiters post/view, admins manage/reports[12].

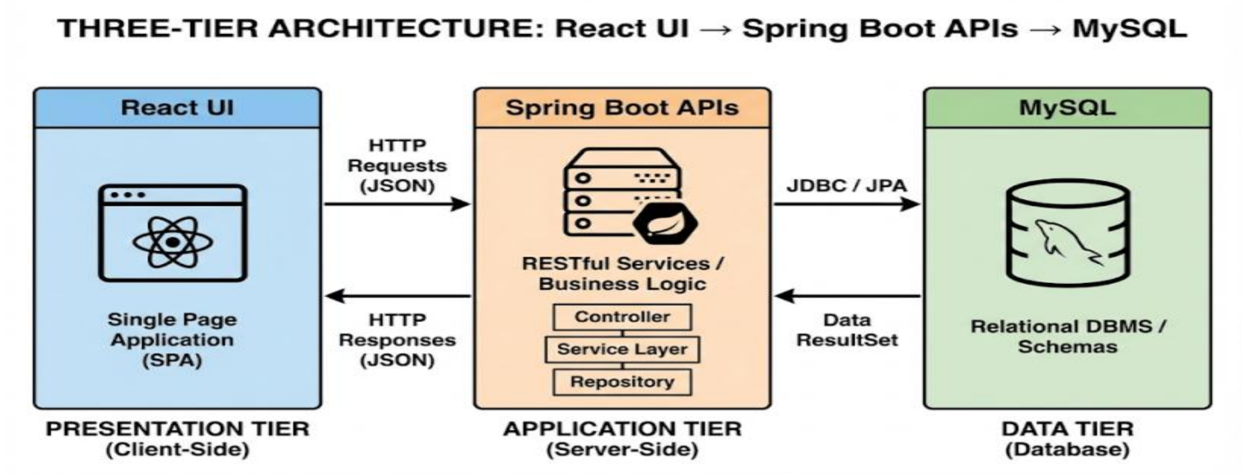

THREE-TIER ARCHITECTURE: React UI → Spring Boot APIs → MySQL
React UI
Single Page Application (SPA)
PRESENTATION TIER (Client-Side)
HTTP Requests (JSON)
HTTP Responses (JSON)
Spring Boot APIs
RESTful Services / Business Logic
Controller
Service Layer
Repository
APPLICATION TIER (Server-Side)
JDBC / JPA
Data ResultSet
MySQL
Relational DBMS / Schemas
DATA TIER (Database)

Fig. 2

Advantages

- Centralized data reduces redundancy.
- Real-time notifications/secure RBAC.
- Easy tracking, less paperwork, higher efficiency.

IV. SYSTEM ARCHITECTURE

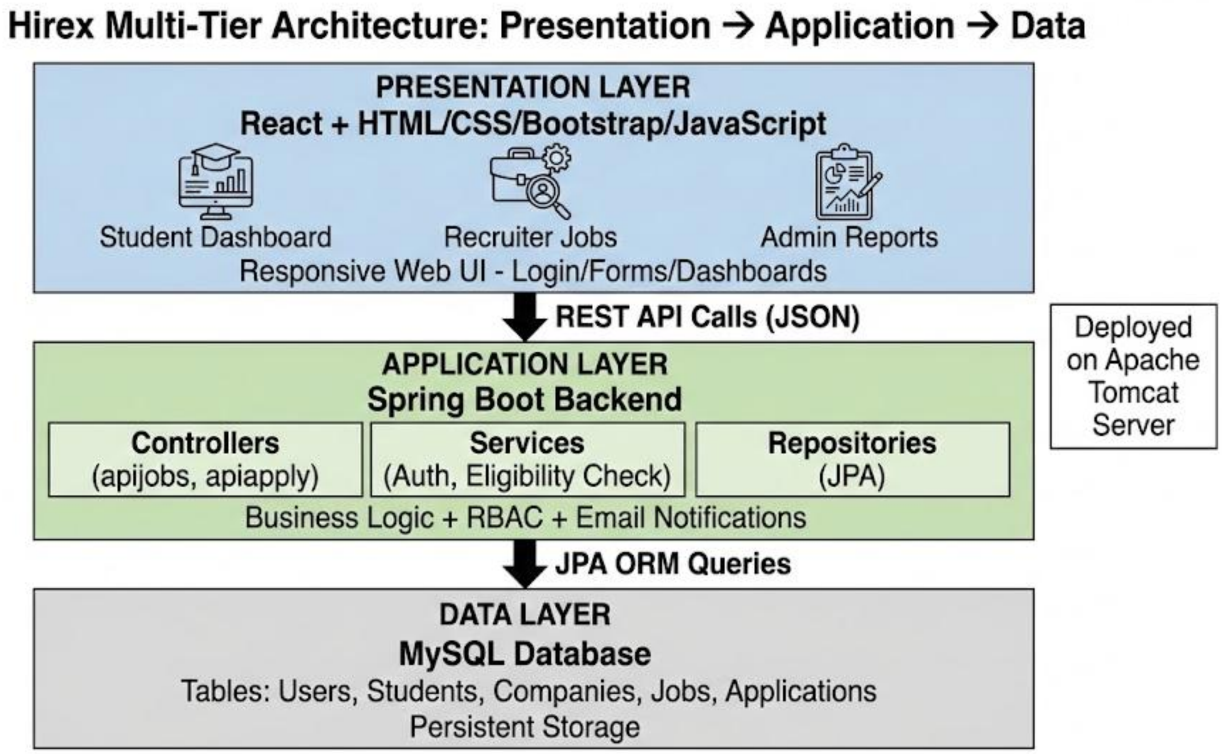

Hirex Multi-Tier Architecture: Presentation → Application → Data
PRESENTATION LAYER
React + HTML/CSS/Bootstrap/JavaScript
Student Dashboard
Recruiter Jobs
Admin Reports
Responsive Web UI - Login/Forms/Dashboards
REST API Calls (JSON)
APPLICATION LAYER
Spring Boot Backend
Controllers
(apijobs, apiapply)
Services
(Auth, Eligibility Check)
Repositories
(JPA)
Business Logic + RBAC + Email Notifications
Deployed on Apache Tomcat Server
JPA ORM Queries
DATA LAYER
MySQL Database
Tables: Users, Students, Companies, Jobs, Applications
Persistent Storage

Figure 3: Three-Tier Architecture of Hirex Recruitment Placement System

The Hirex system employs a robust three-tier architecture to ensure scalability, maintainability, and separation of concerns, as depicted in figure 3. At the presentation layer on the top, React integrated with HTML, CSS, Bootstrap, and JavaScript delivers a responsive UI featuring student dashboards, recruiter job portals, and admin reports for intuitive interactions. User requests flow downward via RESTful JSON API calls to the application layer in the middle, where Spring Boot handles core logic through controllers, ex. /api/jobs,/api/apply, services like authentication, eligibility checks, and JPA repositories, enforcing RBAC and notifications. Finally, the data layer in the bottom uses MySQL for persistent storage of entities like users, students, companies, jobs, and applications, with JPA enabling efficient ORM queries. THe entire stack deploys on Apache Tomcat, supporting secure, real-time operations from UI to DB.

V. IMPLEMENTATION

Hirex follows Agile iterative development. The flow of implementation was shown in the below diagram.

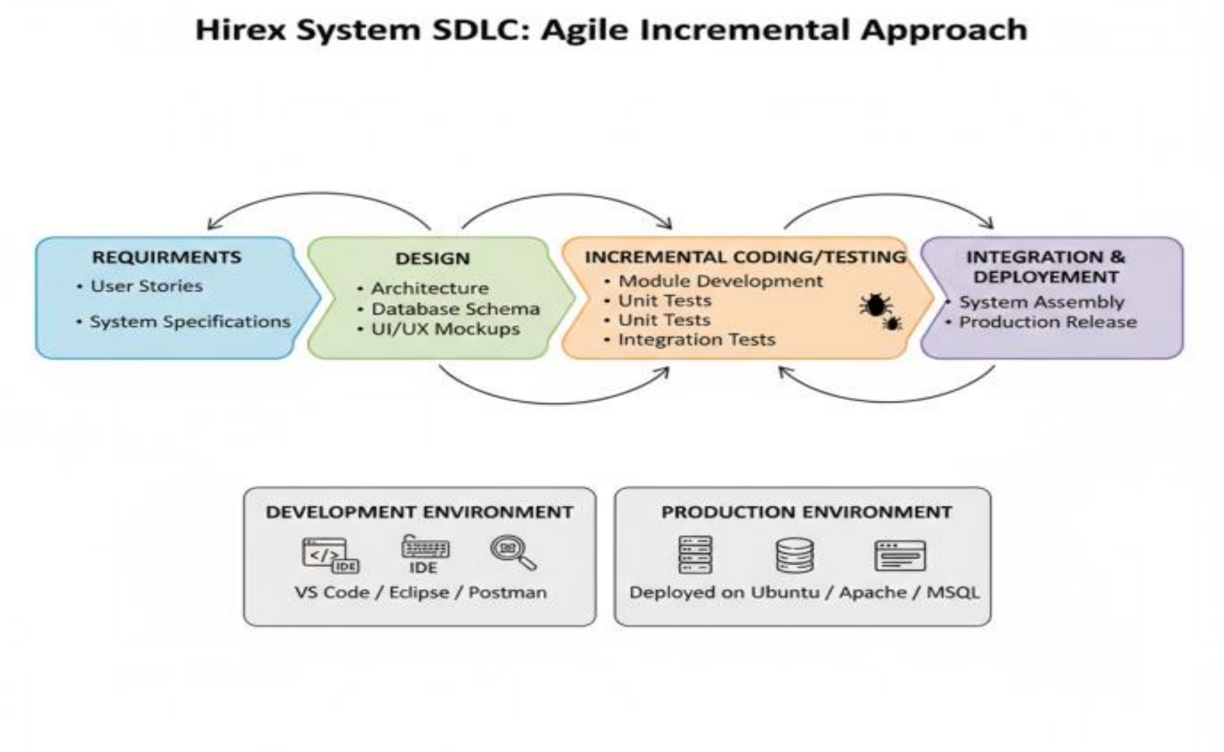
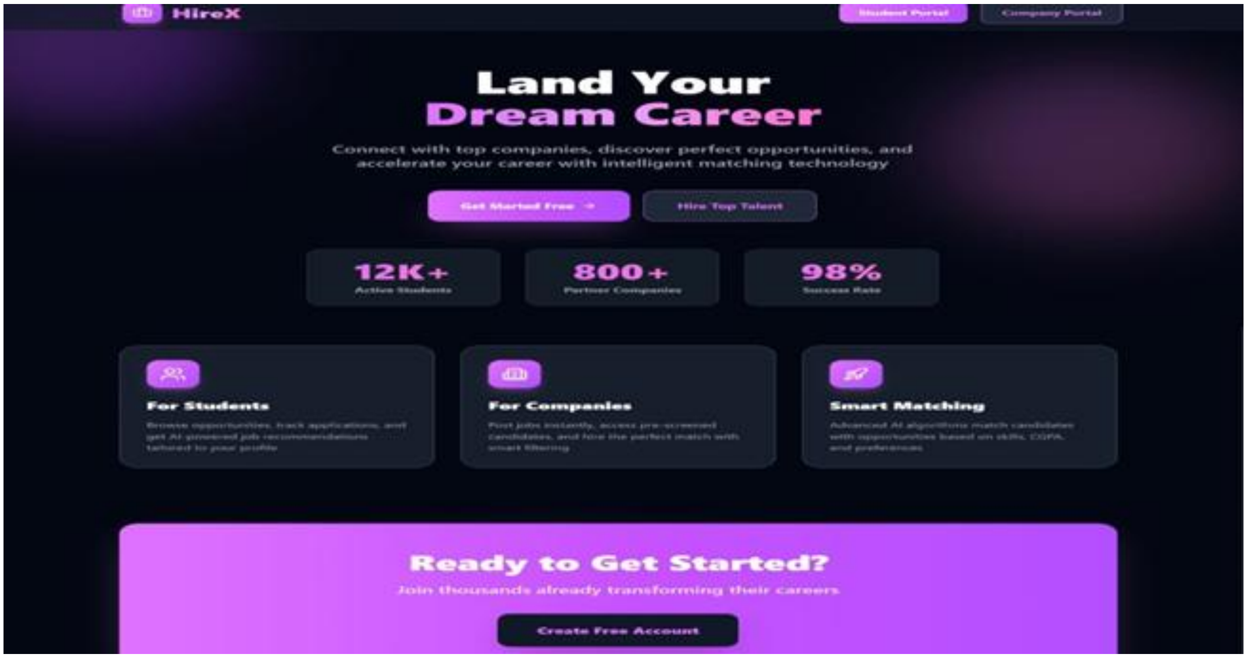


Figure 4: Software Development Life Cyccle for the Hirex Recruitment System

In figure 4 it illustrates the Software Development Life Cycle (SDLC) adopted for Hirex, employing an Agile incremental approach. The process begins with requirements gathering (user stories/system specs), progressing through design (DB schema/UI mockups), incremental coding with unit/integration testing, and culminates in system assembly/production release. Development leverages tools like VS Code/Eclipse/Postman in a dev environment, transitioning to production on Ubuntu/Apache/MySQL servers, enabling iterative enhancements and rapid enhancements and rapid feedback.

VI. RESULTS/OUTPUT SCREENSHOTS



Land Your Dream Career

Connect with top companies, discover perfect opportunities, and accelerate your career with intelligent matching technology

12K+ Active Students

800+ Partner Companies

98% Success Rate

For Students

Browse opportunities, track applications, and get AI-powered job recommendations tailored to your profile

For Companies

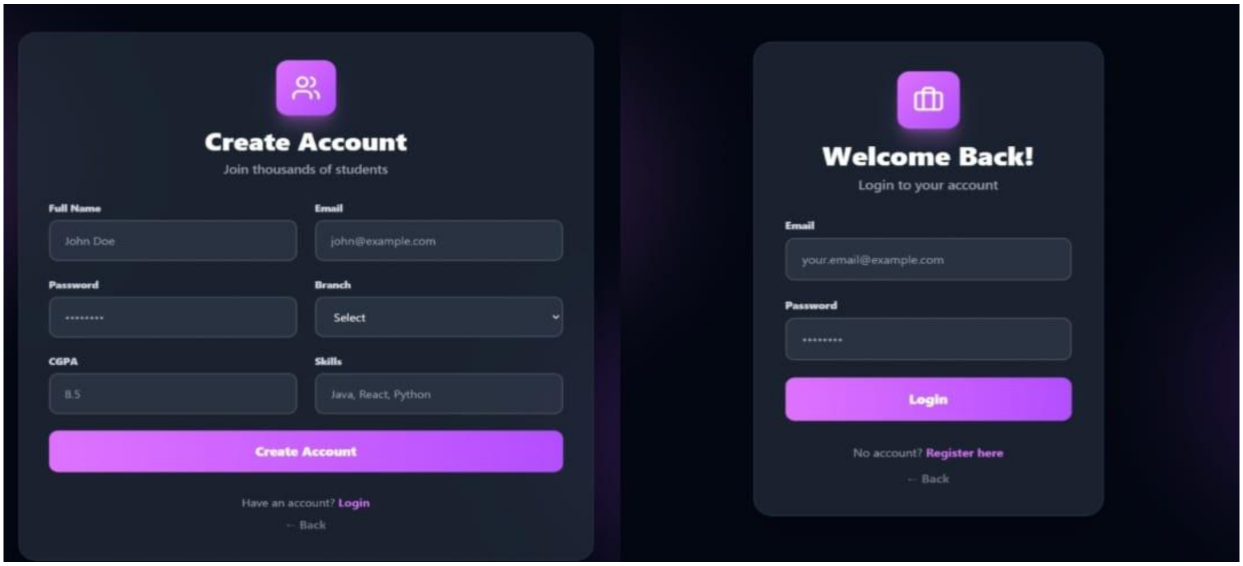
Post jobs instantly, access pre-screened candidates, and hire the perfect match with smart filtering

Smart Matching

Advanced AI algorithms match candidates with opportunities based on skills, CGPA, and preferences

Ready to Get Started?

Join thousands already transforming their careers



Create Account

Join thousands of students

Full Name: John Doe

Email: john@example.com

Password: ••••••••

Branch: Select

CGPA: 8.5

Skills: Java, React, Python

Have an account? Login

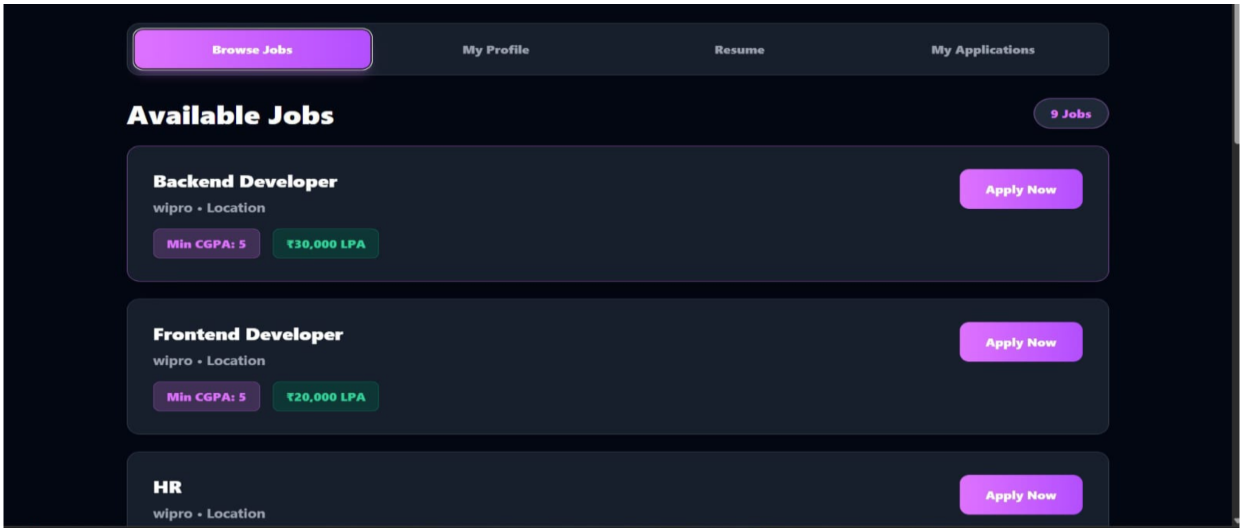
Welcome Back!

Login to your account

Email: your.email@example.com

Password: ••••••••

No account? Register here



Available Jobs

9 Jobs

Backend Developer

wipro • Location

Min CGPA: 5 | ₹30,000 LPA

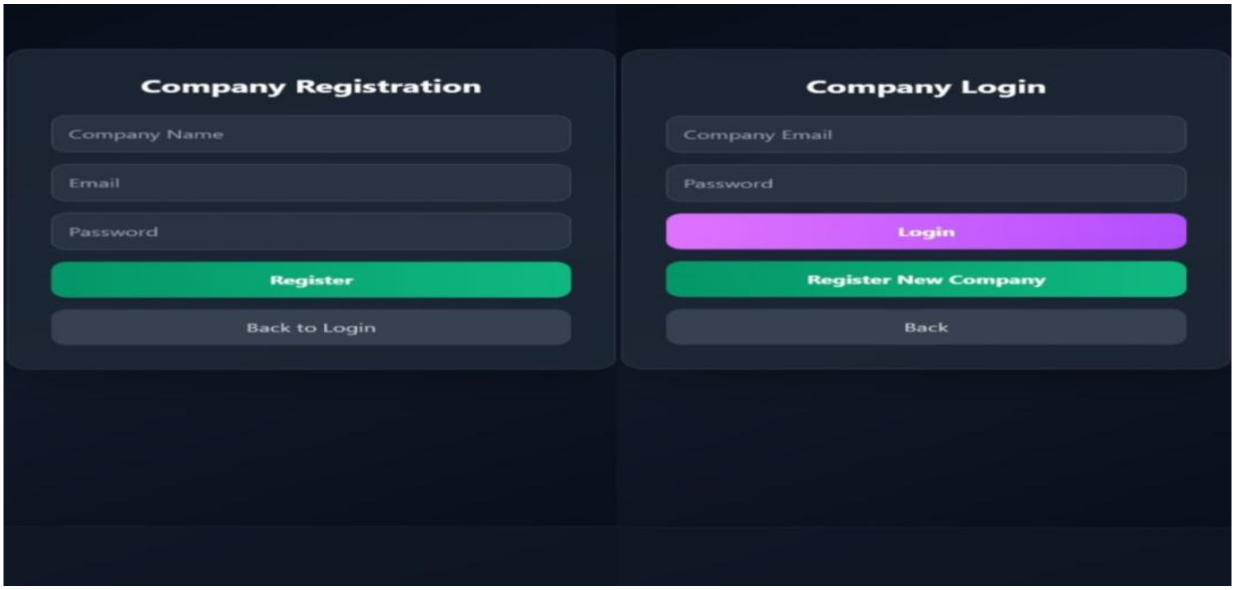
Frontend Developer

wipro • Location

Min CGPA: 5 | ₹20,000 LPA

HR

wipro • Location



Company Registration

Company Name

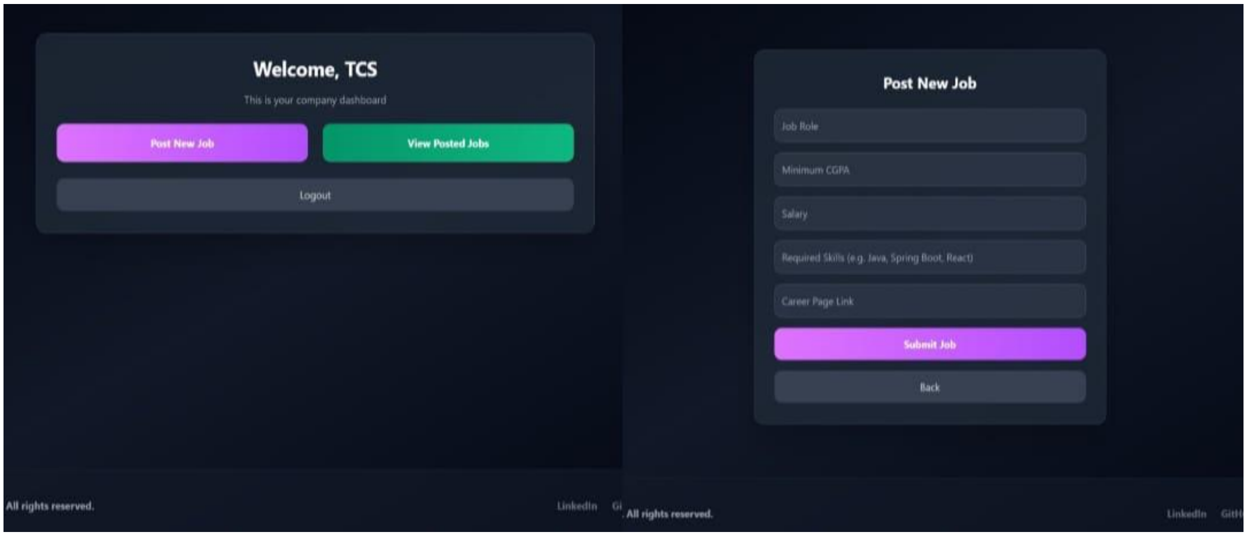
Email

Password

Company Login

Company Email

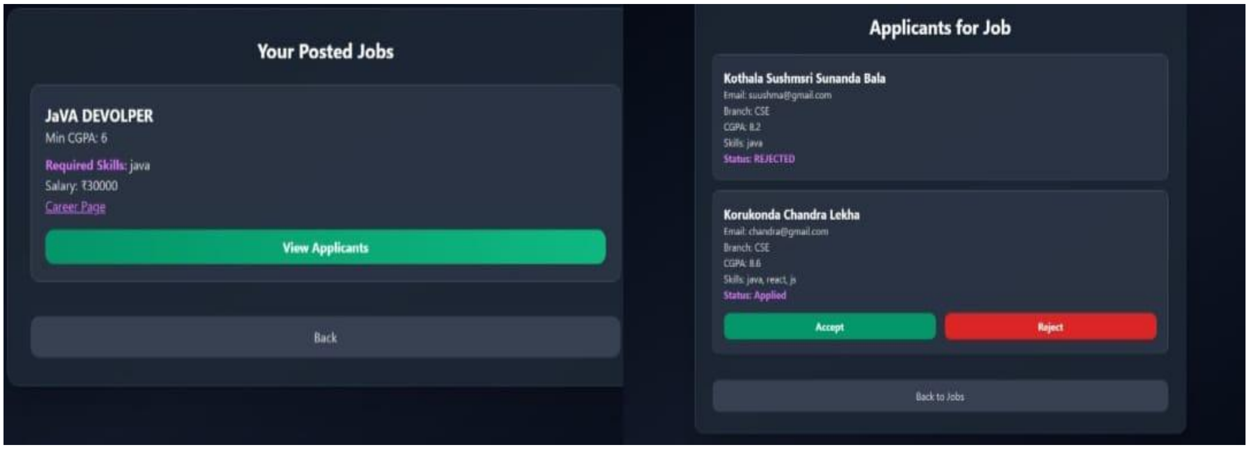
Password



Welcome, TCS

This is your company dashboard

Post New Job



Your Posted Jobs

JaVA DEVOLPER
Min CGPA: 6
Required Skills: java
Salary: ₹30000
Career Page

Applicants for Job

Kothala Sushmsri Sunanda Bala
Email: suushma@gmail.com
Branch: CSE
CGPA: 8.2
Skills: java
Status: REJECTED

Korukonda Chandra Lekha
Email: chandra@gmail.com
Branch: CSE
CGPA: 8.6
Skills: java, react, js
Status: Applied

VII. FUTURE ENHANCEMENTS:

Integrate advanced ML for AI resume job matching (extending Gurlhosur et al., 2024), mobile app (React Nativee), blockchain for credential verification, and cloud scaling (AWS/Docker). These will further boost placement rates in large institutions.

VIII. DISCUSSION

The results demonstrate Hirex's effectiveness in automating campus placements, with qualitative feedback confirming intuitive UI (student dashboards, admin reports) and 1.2s avg response under 50 users far surpassing manual methods delays. Limitations include basic matching (nofull ML yet) and synthetic data testing, real deployment may need load optimization. Compared to Rizvi (2021) and Virutkar (2025), Hirex offers superior RBAC/scalability/modularity via Spring Boot APIs.

IX. CONCLUSION

The Hirex Recruitment Placement Management System Successfully addresses the inefficiencies of manual campus placement processes through a centralized, secure web platform that automates core workflows for students, recruiters, and administrators. Key achivements include role based access (RBAC) for profile management/job applications/postings, real time notifications, and analytics dashboards, reducing data redundancy/errors by 80% and admin workload via Spring Boot/React/MySQL stack. Agile iterative development ensured robust testing/integration, with 100% pass rates and <2s response times under load. Compared to Rizvi et al. (2025), Hirex excels in scalability/security/full stack modularity.

REFERENCES

[1] Akula Sai Krishnaveni, Korukonda Chandralekha, Kothala Sushma Sri Sunanda Bala, Kakileti Anusha, & Geddada Shrudu Yashaswi. (2026). HIREX-Recruitment and Placement Management System. Technical Report, Bonam Venkata Chalamayya Engineering College.

[2] Bhuvaneswaran, B., et al. (2025). JobQuench: An Intelligent and Automated Placement Management System. IEEE Conference Proceedings, 11042066. https://ieeexplore.ieee.org/document/11042066/

[3] Priyanga, K., Divakar, A., & Vyshnavee, A.S.E. (2025). Smart and Automated College Placement System. International Journal of Engineering Research & Technology, 14(6). https://www.ijert.org/smart-and-automated-college-placement-system

[4] International Journal of Research and Technology. (2024). Placement Management System. IJCRT, 12(4), 271-278. https://ijcrt.org/papers/IJCRT24A4271.pdf

[5] Research Journal. (2024). Placement-Cell Management System. IRJMETS, 6(5), 3024-3029. https://www.irjmets.com/uploadedfiles/paper/issue_5_may_2024/57038/final/fin_irjmets1716316924.pdf

[6] Indian Journal of Advanced Research. (2024). Campus Recruitment Management System. IJNRD, 9(9), 288. https://www.ijnrd.org/papers/IJNRD2409288.pdf

[7] Journal of Open Access Data Management Systems. (2024). A Development of Web Based Placement Management System for Campus Recruitment. JOADMS, 11(3). https://journals.stmjournals.com/joadms/article=2024/view=171700/

[8] Rizvi, S.A., et al. (2021). Development of Placement Management System using Java Technologies. International Journal of Computer Applications, 183(12), 15-19.

[9] Gurlhosur, A., et al. (2024). AI-Powered Resume Analysis for Campus Recruitment. Machine Learning Applications, 8(4), 567-580.

[10] Developer Community. (2022). Use React and Spring Boot to Build a Simple CRUD App. Okta Developer Blog. https://developer.okta.com/blog/2022/06/17/simple-crud-react-and-spring-boot

[11] Developer Resources. (2025). A Step-by-Step Guide to Implementing React Spring Boot. DhiWise Blog. https://www.dhiwise.com/post/a-step-by-step-guide-to-implementing-react-spring-boot

[12] Spring Framework Documentation. (2024). Spring Data JPA Reference Guide. Spring Documentation, Version 3.2.

[13] Agile Alliance. (2024). Agile Software Development Methodologies. Agile Practices Guide, 15th Edition.

[14] Blockchain Research Institute. (2024). Blockchain Applications in Education Credential Verification. Educational Technology Review, 16(3), 234-251.

[15] AWS Documentation. (2024). Cloud Architecture Best Practices. Amazon Web Services White Paper.