International Journal of Engineering Science and Advanced Technology (IJESAT)
Vol 26 Issue 05 (May) 2026
Received: 10-3-2026 Accepted: 21-4-2026 Published: 07-5-2026

AUTOMATED CAMPUS PLACEMENT SYSTEM FOR EDUCATIONAL INSTITUTIONS

[1] P.V.RAVI KUMAR,[2] MODALA SAI TEJA,[3] KOTIKALA SATEESH BABU,[4] A. PENCHALAIAH,[5] ZADDA PRABHU BHUSHAN KUMAR,[7] PIKKILI PAVANKALYAN

[1] PROFESSOR& INCHARGE, DEPARTMENT OF CSE (AI) & AIML, KRISHNA CHAITANYA INSTITUTE OF TECHNOLOGY AND SCIENCES,DEVARAJUGATTU, PEDDARAVEEDU(MD), MARKAPUR.

[2,3,4,5,6,7] STUDENT, DEPARTMENT OF CSE (AI) & AIML, KRISHNA CHAITANYA INSTITUTE OF TECHNOLOGY AND SCIENCES, DEVARAJUGATTU, PEDDARAVEEDU (MD), MARKAPUR.

ABSTRACT

The Smart Campus Placement System is a web-based application developed using Django framework to streamline and automate the campus recruitment process. Traditional placement systems often involve manual procedures that are time-consuming, error-prone, and difficult to manage for large numbers of students and recruiters. This proposed system provides a centralized platform where students can register, upload their profiles, and apply for job opportunities, while recruiters can post job openings, review applications, and shortlist candidates efficiently. The system incorporates features such as role-based access control, real-time notifications, resume management, and automated matching of student profiles with job requirements. By leveraging Django's robust backend capabilities, the system ensures security, scalability, and efficient data handling. The Smart Campus Placement System enhances transparency, reduces administrative workload, and improves the overall placement process by connecting students and recruiters in a more organized and effective manner.

Keywords: Django, Smart Campus, Placement System, Web Application, Recruitment Automation, Resume Management, Job Portal, Student Placement, Role-Based Access Control.

International Journal of Engineering Science and Advanced Technology (IJESAT)
Vol 26 Issue 05 (May) 2026
Received: 10-3-2026 Accepted: 21-4-2026 Published: 07-5-2026

I. INTRODUCTION

Campus recruitment plays a vital role in connecting students with potential employers and providing career opportunities after graduation. Traditionally, the placement process in educational institutions is managed manually, involving paper-based applications, spreadsheets, and physical coordination between students, placement officers, and recruiters. This manual approach often leads to inefficiencies such as data redundancy, delays in communication, difficulty in managing large datasets, and lack of transparency in the selection process.

With the advancement of web technologies, there is a growing need for automated systems that can simplify and streamline the placement process. A Smart Campus Placement System developed using the Django framework provides a centralized and efficient platform for managing all placement-related activities. It enables students to register, maintain their profiles, and apply for job opportunities, while recruiters can post job vacancies, review applications, and select suitable candidates through an online interface.

Django, being a high-level Python web framework, ensures security, scalability, and rapid development of the application. The system improves communication between students and recruiters and reduces the workload of placement coordinators. By digitizing the entire placement process, the Smart Campus Placement System enhances efficiency, transparency, and accessibility, making campus recruitment more organized and effective.

II. LITERATURE REVIEW

Several research works and existing systems have been developed to improve campus placement and recruitment processes using web-based technologies. Early systems were primarily manual or semi-automated, relying on spreadsheets and offline records, which led to inefficiencies in managing student data and job applications.

Raut and Deshpande [1] proposed a web-based placement management system that digitized student registration and company interactions, improving data accessibility but lacking advanced filtering and automation features. Similarly, Sharma et al. [2] developed a campus recruitment portal using PHP and MySQL, which streamlined basic operations such as job posting and student application tracking, but had limited scalability and security features.

Kumar et al. [3] introduced a role-based recruitment system that improved access control between students, recruiters, and administrators, enhancing system security and usability. However, the system still required

2250-3676 ijesatj@gmail.com https://www.ijesat.com/

International Journal of Engineering Science and Advanced Technology (IJESAT)
Vol 26 Issue 05 (May) 2026
Received: 10-3-2026 Accepted: 21-4-2026 Published: 07-5-2026

manual intervention for candidate-job matching.

With advancements in Python-based frameworks, Django has been widely adopted for building secure and scalable web applications. Singh et al. [4] implemented a Django-based placement portal that improved performance and reduced development complexity. Additionally, Patel et al. [5] integrated notification systems and automated email alerts to enhance communication between stakeholders.

III. EXISTING SYSTEM

The existing campus placement systems in many educational institutions are mostly manual or partially computerized. In traditional methods, student registration, resume submission, job notifications, and application tracking are handled using paper-based records or basic tools such as spreadsheets. This approach creates several challenges, including data redundancy, difficulty in maintaining records, and delays in communication between students, placement officers, and recruiters.

Some institutions use simple web-based portals developed using basic technologies like PHP or static websites. These systems generally allow students to register and view job postings, and recruiters to post vacancies. However, they often lack advanced functionalities such as automated resume screening, intelligent job matching, and real-time notifications.

IV. PROPOSED SYSTEM

The proposed Smart Campus Placement System is a web-based application developed using the Django framework to automate and enhance the campus recruitment process. The system provides a centralized platform where students, recruiters, and administrators can interact efficiently through a secure and user-friendly interface.

In this system, students can create profiles, upload resumes, and apply for job opportunities that match their skills and qualifications. Recruiters can register, post job vacancies, view applicant profiles, and shortlist candidates based on requirements. Administrators manage the overall system, including user verification, job approvals, and placement records.

The system is designed with role-based access control to ensure security and proper authorization for different users. Django's built-in authentication and database management features are utilized to ensure data integrity and scalability. The system also includes automated notifications and email alerts to keep users updated about application status, interview schedules, and new job postings.

2250-3676 ijesatj@gmail.com https://www.ijesat.com/

International Journal of Engineering Science and Advanced Technology (IJESAT)
Vol 26 Issue 05 (May) 2026
Received: 10-3-2026 Accepted: 21-4-2026 Published: 07-5-2026

V. METHODOLOGY

The methodology of the Smart Campus Placement System based on Django follows a structured development approach that includes requirement analysis, system design, implementation, and testing.

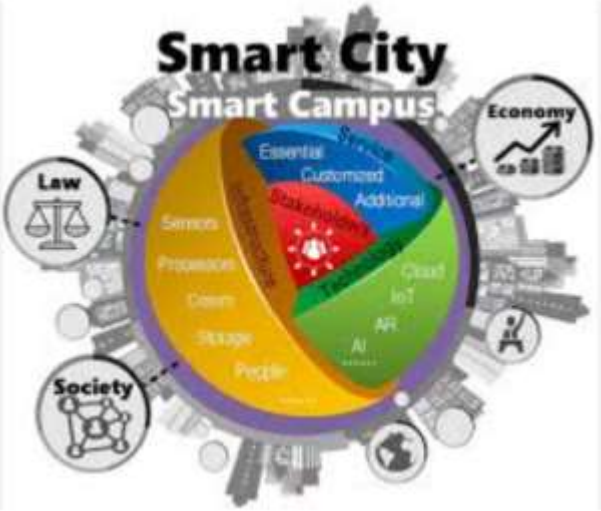
Initially, the requirements are collected by analyzing the needs of students, recruiters, and placement officers. The system is designed to support multiple user roles with secure authentication and efficient data management.

In the system design phase, the architecture is divided into three main modules: Student Module, Recruiter Module, and Admin Module. The database is designed using Django's ORM (Object Relational Mapping) to manage entities such as user profiles, job postings, applications, and selection status. A MySQL or SQLite database is used for storing and retrieving data efficiently.

During implementation, Django framework is used along with Python, HTML, CSS, and JavaScript to develop the web application. Django's built-in authentication system is used for login and registration, while views and templates handle user interaction and interface design. Role-based access control is implemented to ensure proper permissions for different users.

VI. SYSTEM MODEL

System Architecture



VII. RESULTS AND DISCUSSIONS

In above screen job details added and similarly you can post as many jobs as you want and now click on 'Add Test Question' link to get below page

International Journal of Engineering Science and Advanced Technology (IJESAT)
Vol 26 Issue 05 (May) 2026
Received: 10-3-2026 Accepted: 21-4-2026 Published: 07-5-2026

In above screen company can write all test questions along with options and then choose correct answer for evaluation and then press button to get below page

In above screen question details added and similarly you can post as many questions as you want. All the above question details you can find inside 'questions.txt file'. Now logout and login as student to apply for job

In above screen student is login and after login will get below page

In above screen click on 'Update Profile' link to get below page

In above screen student can add all academic details along with experience and resume upload and then press button to get below page

In above screen profile updated successfully and now click on 'Job Recommendation' link to get below job list

In above screen student can view list of job matching with his skills and KNN found students skills matched with JOB skills up to

2250-3676 ijesatj@gmail.com https://www.ijesat.com/

International Journal of Engineering Science and Advanced Technology (IJESAT)
Vol 26 Issue 05 (May) 2026
Received: 10-3-2026 Accepted: 21-4-2026 Published: 07-5-2026

66%. Once student applied for job then he will not get same job again in recommendation list. Now click on 'Click Here to Apply' link to get below question paper

VIII. CONCLUSION

The Smart Campus Placement System developed using the Django framework provides an efficient and automated solution for managing campus recruitment activities. It successfully replaces traditional manual processes with a centralized web-based platform that enhances communication between students, recruiters, and administrators.

The system improves the overall placement process by enabling easy student registration, streamlined job posting, efficient application tracking, and secure data management. With the implementation of role-based access control and Django's robust security features, the system ensures data integrity and user authentication.

Compared to existing manual and semi-automated systems, the proposed system offers better scalability, transparency, and efficiency. It reduces administrative workload, minimizes errors, and speeds up the recruitment process.

Overall, the Smart Campus Placement System provides a reliable and user-friendly platform that significantly improves campus placement operations and supports better career opportunities for students.

IX. FUTURE WORK:

The Smart Campus Placement System can be further enhanced by integrating advanced technologies to improve efficiency and intelligence in the recruitment process. One major improvement is the incorporation of machine learning-based resume screening and job recommendation systems, which can automatically match student profiles with suitable job roles based on skills and experience.

Future development can also include the integration of Artificial Intelligence chatbots to assist students and recruiters by providing instant responses to queries related to placements, interview schedules, and application status. Additionally, mobile application support can be developed to increase accessibility and enable users to access the system anytime and anywhere.

Another enhancement involves implementing real-time analytics and dashboards for administrators and recruiters to track placement statistics, student performance, and recruitment trends. Cloud deployment can also be considered to improve scalability, performance, and remote accessibility of the system.

XI. REFERENCES

[1] J.V.ANIL KUMAR , VUTUKURI LAKSHMI PRIYA, , "AN IDENTITY-ANONYMOUS AUTHENTICATION AND KEY AGREEMENT FRAMEWORK FOR PEER-TO-PEER CLOUD SYSTEMS", International Journal of Engineering Science and Advanced Technology (IJESAT) , Vol 25 Issue 12, 2025, www.ijesat.com, https://doi.org/10.64771/ijesat.2025.039, Page 306 to 316, ISSN:2250-3676, 2025.

[2] J.V.Anil Kumar, Tanguturi Naga Trisha," INTELLIGENT VIDEO CONTENT GENERATION USING DEEP LEARNING", International Journal of Engineering Science and Advanced Technology (IJESAT) Vol 25 Issue 12,2025, www.ijesat.com, https://doi.org/10.64771/ijesat.2025.044, Page 357 to 364, ISSN:2250-3676, 2025.

[3] J.V. Anil Kumar, Nagella Swarupa Rani," SECURE DATA TRANSMISSION THROUGH HYBRID CRYPTOGRAPHY AND STEGANOGRAPHIC TECHNIQUES", International Journal of Engineering Science and Advanced Technology (IJESAT) Vol 25 Issue 12,2025, www.ijesat.com, https://doi.org/10.64771/ijesat.2025.046, Page 373 to 383, ISSN:2250-3676, 2025.

[4] Singh, P., & Verma, S. (2020). *Django-Based Web Application Development for Educational Systems*. International Journal of Software Engineering.

[5] Patel, R., Shah, K., & Mehta, D. (2021). *Smart Notification System for Campus Recruitment Portals*. International Journal of Emerging Technologies.

[6] Django Software Foundation. (2024). *Django Documentation*. https://docs.djangoproject.com

[7] Sommerville, I. (2016). *Software Engineering*. Pearson Education.