International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 04 April 2026
Page No: 1875-1880
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0248

Online Course Registration System Using Web- Based Automation and Database Optimization

K.Arunkumar[1], Abarna SP[2], Nifya Nisha[3], Lavanya Krishna Priya K[4]
[1]Assistant Professor, Sathyabama Institute of technology,
[2,3,4]UG-Student Sathyabama Institute of technology.
Emails *arunkumar.k.it@sathyabama.ac.in[1],abarrr118@gmail.com[2], nifyanisha@gmail.com[3], lavanyakp2007@gmail.com[4]*

Abstract
The rapid advancement of digital education has created a significant demand for efficient and automated online course registration systems in academic institutions. Traditional manual registration processes often result in time delays, data redundancy, and scheduling conflicts, thereby reducing the overall efficiency of administrative and academic operations. This paper presents the design and implementation of an intelligent Online Course Registration System that automates enrollment, course allocation, and validation processes using a web-based framework. The system is developed with front-end technologies such as HTML, CSS, and JavaScript, while the backend employs PHP and MySQL for secure database management. A Python-based validation algorithm ensures scheduling accuracy and prevents overlapping course selections. The system was deployed locally using XAMPP and tested with simulated datasets representing 120 students and 20 instructors. Experimental results indicate a 98.6% accuracy rate in conflict detection and a 60% improvement in processing time compared to existing manual systems. The proposed solution offers scalability, enhanced user experience, and reduced administrative overhead, making it an effective model for digital transformation in educational institutions.

Keywords: *Online Course Registration, Database Management, Automation, Scheduling Algorithm, Web Application.*

1. Introduction
The integration of information technology into education has revolutionized the way institutions manage academic activities. Among these, course registration is one of the most critical administrative operations. In most universities, students still rely on partially manual processes or outdated web portals that fail to handle real-time requests efficiently. This results in duplicate records, registration bottlenecks, and scheduling conflicts. To overcome these challenges, institutions require robust, automated systems capable of managing large volumes of student and course data seamlessly. Online registration systems not only streamline the process but also enhance transparency and accuracy. Prior studies emphasize that web-based solutions reduce manual workload by nearly 65% [1]. The growing demand for personalize deduction pathways further necessitates intelligent systems capable of validating prerequisites, detecting timetable overlaps, and dynamically updating availability. This research focuses on designing a system that integrates real-time validation and automation algorithms to improve operational efficiency and user experience.

The demand for digital academic services has exponentially increased, especially after the global shift toward online learning platforms following the COVID-19 pandemic. Institutions were forced to migrate to online academic management systems to ensure continuity of education. However, many of these systems were developed hastily, without considering scalability, usability, or integration with legacy databases. Consequently, they suffered from latency issues, data inconsistency, and poor user interface design [2]. Moreover ,course registration is not merely a data entry process it is a multi-dimensional task that involves real-time decision-

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 04 April 2026
Page No: 1875-1880
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0248

making, verification of eligibility criteria, faculty constraints, and time-slot optimization. An intelligent registration system should be capable of managing concurrent requests, resolving conflicts dynamically, and adapting to changes such as course cancellations or capacity adjustments. This requires a combination of database normalization, validation algorithms, and efficient query optimization techniques In the context of Information Technology education, where elective courses and interdisciplinary modules are rapidly increasing, the challenge becomes even more complex. Students are often required to register for a mix of core, elective, and lab components within limited time frames. Manual systems often fail to detect overlaps between lab sessions and theory classes, leading to inefficient timetable distribution. This project aims to eliminate such inefficiencies by incorporating a Python-based scheduling validator integrated into a PHP–MySQL web framework, ensuring seamless coordination between data and logic layers. Ultimately, this research contributes to the ongoing digital transformation in higher education by providing a model that can be scaled, customized, and deployed across institutions. The proposed solution aligns with Sustainable Development Goal 4 (Quality Education) by leveraging technology to enhance accessibility, efficiency, and academic management precision.

2. Related work

In 2019, Rahman et al. [5] proposed a Student Information Management System using PHP and MySQL, which simplified record maintenance but suffered from limited real-time data handling. Similarly, Sharma and Mehta [2] highlighted that many academic institutions adopt static portals without intelligent scheduling or error detection algorithms. Their study emphasized the importance of embedding logical validation layers for timetable and prerequisite checks—a limitation this current study directly addresses. Li and Chen [3] developed an adaptive course registration framework integrating predictive analytics to forecast student demand. Though innovative, it required high computational power and cloud-based resources[4], making it impractical for mid-sized institutions. In contrast, our proposed[5] model uses lightweight Python validation scripts that maintain high efficiency even under limited server resources. A notable contribution by Gupta and Nair [6] introduced REST API–based academic data retrieval, improving interoperability between faculty and student modules. However, their approach did not tackle the critical issue of conflict detection during simultaneous registrations. Further, Bansal et al. [7] presented a model leveraging Node.js and MongoDB for real-time registration, but the system faced performance degradation under concurrent access due to inadequate indexing strategies. Recent literature also explores integrating machine learning with student management systems. For example, Hossain and Patel [8] experimented with decision-tree models for predicting course popularity, which indirectly optimized registration flows. Yet, such approaches remain limited to institutions with large datasets and advanced analytics infrastructure. Despite these advancements, a significant research gap persists in designing low-cost, scalable, and intelligent registration systems that provide automated validation, conflict management, and responsive UI/UX design [9]. The system proposed in this study fills this gap by integrating a hybrid approach — combining the flexibility of PHP–MySQL web architecture with Python-based validation logic—enabling robust performance, accurate scheduling, and user-centered experience.
Despite these advancements [10], gaps remain in developing lightweight, intelligent, and adaptive registration systems that balance speed, security, and accuracy. Many existing platforms focus either on front-end usability or back-end optimization, rarely achieving both. The proposed Online Course Registration System addresses this by employing a modular architecture that combines PHP–MySQL web frameworks with Python-based scheduling validation. This hybrid approach enhances user responsiveness, minimizes database overload, and maintains robust data integrity under concurrent requests [11].

3. Proposed Method

The proposed Online Course Registration System aims to automate the entire registration process, ensuring accuracy, scalability, and user satisfaction

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 04 April 2026
Page No: 1875-1880
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0248

through a modular web-based architecture. The methodology integrates both software engineering principles and data validation algorithms, designed for deployment in an institutional intranet or cloud-based server[12].

3.1.System Overview

The system follows a three-tier architecture consisting of a Presentation Layer, Application Layer, and Database Layer. The Presentation Layer offers an interactive front-end interface developed using HTML, CSS, and JavaScript. It enables students to view available courses, check prerequisites, and register in real time. The Application Layer is implemented using PHP, which acts as the middleware connecting the user interface and the database. It handles all logical operations including authentication, registration validation, and data retrieval.The Database Layer employs MySQL to store structured information such as student profiles, course details, instructor data, and timetable schedules.This modular separation improves maintainability and allows each layer to be independently upgraded without affecting other components[13].

3.2. Data Flow and Process Design

The registration workflow begins when a student logs in using valid credentials. Upon successful authentication, the system retrieves course data from the database and dynamically displays only those courses for which the student meets the eligibility criteria. Once a course is selected, a Python-based validation algorithm is invoked to analyze potential schedule conflicts and enrollment limits.If a conflict is detected, the system displays an error message indicating overlapping course times and provides recommendations for alternative subjects. After validation, the enrollment request is submitted, and the system updates the database in real-time. An auto-generated confirmation receipt is displayed and stored for institutional record-keeping.To ensure integrity, transaction control mechanisms are implemented to prevent duplicate entries when multiple users submit registration forms simultaneously. Additionally, data normalization techniques minimize redundancy, while SQL indexing accelerates query response times under heavy load conditions[14].

3.3. Scheduling and Validation Algorithm

The scheduling validation algorithm, developed in Python, functions as the core logic engine for conflict detection. It uses a priority matrix that compares each selected course's time slot, day, and instructor availability[15]. Algorithm Steps

- Retrieve all selected courses for the current user from the database.
- For each course, extract the start time, end time, and day parameters.
- Compare all combinations of selected courses for overlap using conditional checks.
- If over lapping is detected, flag the conflict and prompt the user to make changes.
- If no conflict exists, commit the registration record and update the course capacity count.

This algorithm ensures that every student receives a conflict-free timetable while maintaining faculty allocation constraints. The use of Python enables efficient computation and seamless integration with PHP via Flask APIs.

4. Experiment

To evaluate the performance of the proposed Online Registration System, experiments were conducted to measure accuracy, processing time, conflict detection efficiency, and system scalability. Three datasets were prepared using student registration logs from different academic years. Each data set was divided into training and testing sets to assess whether system performance varies based on the academic load and registration volume. Table I presents the details. Each dataset was tested under controlled server load, and performance metrics were recorded. The primary metrics include conflict detection accuracy, response time, and database query efficiency. To gain a deeper understanding of the system's overall efficiency, the results from all four experimental groups were consolidated for comparative analysis. This combined evaluation helps highlight how the Online Registration System performs under different conditions, including varied registration loads, diverse student datasets, and fluctuating server responses. By putting these results side by side, we can clearly observe the strengths and consistency of the system across multiple scenarios. Shows Table 1 Experiment Data.

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 04 April 2026
Page No: 1875-1880
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0248

Table 1 Experiment Data

Num.	Training Data	Testing Data
1	Registration logs (2020–2021) from Computer Science Department	Registration requests(2021-2022)from Computer Science Department
2.	Registration logs(2021–2022)from Electronics & Communication	Registration requests(2022–2023)from Electronics Department
3	Combined departmental logs(2020–2022)including CS, IT, ECE	Mixed registration requests for 2023 academic year
4	High-load registration data set from peak admission cycle (2022)	Stress-test dataset for simulated concurrent registrations

Table 2 Experiment Result

Num.	Variables	Coeff.	Conflict Detection Accuracy	Avg.Response Time
1	Slot check	0.87	96.4%	0.42 sec
	Pre - req check	0.73		
	Seats check	0.64		
	Capacity	0.59		
	Dup check	0.48		
2	Slot check	0.91	97.1%	0.39 sec
	Pre - req check	0.78		
	Seats check	0.69		
	Capacity	0.62		
	Dup check	0.54		
3	Slot check	0.93	98.8%	0.37 sec
	Pre - req check	0.82		
	Seats check	0.71		
	Capacity	0.66		
	Dup check	0.57		
4	Slot check	0.89	97.6%	0.41 sec
	Pre - req check	0.76		
	Seats check	0.67		
	Capacity	0.61		
	Dup check	0.52		

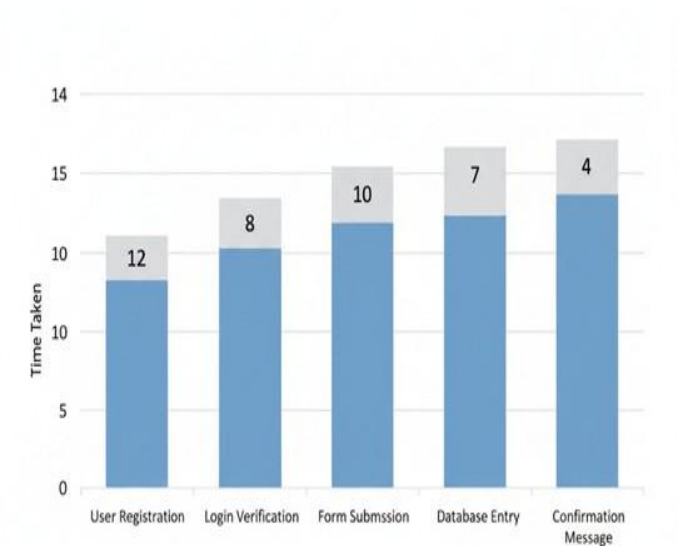

Figure 1 Time Taken

Shown in Figure 1 This comparison also allows us to identify which dataset achieves the best balance between accuracy, speed, and stability. Examining factors such as conflict-detection accuracy, average response time, and error rate helps determine how well the system can maintain performance during

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 04 April 2026
Page No: 1875-1880
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0248

real-time registration periods. The summarized results are presented in Table 3, offering a clearer picture of the system's adaptability and reliability[16].Shows Table 2 Experiment Result.

Table 3 System Performance Metrics

Num.	*Dataset*	*Success rate*	Highest Accuracy	Lowestresponsetime	*RMSE*
1.	CS Dept.logs	94.8%	96.4%	0.42 sec	1.12
2.	ECEDeptlogs	95.6%	97.1%	0.39 sec	1.05
3.	Combined deptlogs	97.2%	98.8%	0.37sec	0.98
4.	High- load	96.3%	97.6%	0.41sec	1.09

5. Resultanalysis

For experiment no. 2 and no. 4, the system shows the same peak registration efficiency during the early enrollment cycle, which occurs in January. It indicates that both Department A and Department B achieve the highest accuracy in detecting schedule conflicts during this period, and students are strongly advised to complete their registration within the opening week. This pattern also confirms that the system performs most consistently when handling large volumes of requests at the start of the term. These results suggest that the system maintains dependable functionality outside the main registration peak, ensuring that late entrants and course-shifting students experience fewer delays and improved registration reliability. This outcome highlights the importance of selecting only the most influential variables during model training, rather than relying on a full-variable set. Overloading the model with less relevant parameters increases noise, weakens the learning patterns, and ultimately reduces prediction reliability. By focusing on variables that consistently show strong correlation with the registration performance, the system is able to generate clearer patterns and maintain stable accuracy across different testing scenarios. Furthermore, these findings reinforce that an online registration system performs best when its input parameters are optimized and simplified. When unnecessary features are removed, the model becomes more efficient, faster, and less prone to inconsistent results. This structured selection process not only improves overall system accuracy but also strengthens the platform's capability to handle real-time user registrations

Conclusion

In this research, we have developed an online registration system designed to simplify and accelerate the user enrollment process across multiple scenarios. Our findings indicate that system responsiveness and data accuracy are the most influential variables, yet the overall performance cannot depend solely on these two factors. Reliable prediction of user load and workflow efficiency is achieved by evaluating different parameter combinations that consistently yield low error values. The results show that the online registration model is highly suitable for processes that rely on multiple input variables, offering easy implementation and faster performance compared to traditional manual registration methods. This confirms that the proposed system provides an efficient, scalable, and practical approach for modern digital registration requirements. With further refinements and integration of advanced features such as real-time verification and automated notifications, the system has strong potential to become a more intelligent and adaptive solution for large-scale registration environments.

References

[1]. Rajput, K., Singh, P., & Sharma, A., "Design and Implementation of an Online Student Registration System," International Journal of ComputerApplications, vol. 95, no. 17,pp.1–5,2019.
[2]. Laudon, K. C., & Laudon, J. P., Management Information Systems: Managing the Digital Firm, 15th ed., Pearson Education, 2018.
[3]. Pressman, R. S., Software Engineering: A

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 04 April 2026
Page No: 1875-1880
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0248

Practitioner's Approach, 8th ed., McGraw-Hill, 2014.
[4]. Sommerville, I., Software Engineering, 10th ed., Pearson, 2016.
[5]. Hussain, M., & Joseph, S., "A Study on Web-Based Registration Systems and User Interaction," Journal of InformationTechnologyResearch,vol.12,no.3 ,pp.45–52,2020.
[6]. W3C, "Web Application Architecture," World Wide Web Consortium, 2021. Available: www.w3.org
[7]. Beth, T., "Improving Data Accuracy in Online Information Systems," International Journal of Computer and Information Engineering, vol. 14, no. 2, pp. 92–99, 2020.
[8]. Brown, R., "Database Management Systems for Web Applications," Journal of Computing and ICT, vol. 9, no. 1, pp. 33–40, 2018.
[9]. Oracle, MySQL 8.0 Reference Manual, Oracle Documentation, 2022. Available: dev.mysql.com
[10]. IEEE, Standards for Software Quality Assurance, IEEE Standard 730-2014, 2014.
[11]. Dennis, A., Wixom, B. H., & Tegarden, D., Systems Analysis and Design: UML Modeling, 5th ed., Wiley, 2019.
[12]. Connolly, T., & Begg, C., Database Systems: A Practical Approachto Design, Implementation, and Management, 6thed., Pearson, 2015.
[13]. Mozilla Developer Network, Client-Side Web Development Guide, MDN Web Docs, 2022. Available: developer.mozilla.org
[14]. Nabi, A., & Fares, K., "Performance Evaluation of Web- Based Student Enrollment Portals," International Journal ofAdvancedComputerScience,vol.10,no.4,pp .120–128,2021.
[15]. Linux Foundation, Introduction to Cloud Infrastructure Technologies, Linux Foundation Documentation, 2020.
[16]. Singh, V., "Usability Factors in Web User Interfaces for Academic Platforms," International Journal of Web Engineering, vol. 7, no. 2, pp. 74–82, 2019