IJIS: Vol.1, Issue 10, November 2025, Page: 26-31 ISSN: 3049-3277

International Journal of Integrative Studies (IJIS)

Journal homepage:www.ijis.co.in

Smart Leave Scheduling Portal

Dr. P.Padmaja[1], Chintala sucharitha[2], Donthala Mrudhula[3],Gorijavolu Kusuma[4] and Nuthulakanti Nithish[5]

[1]Associate Professor&HOD, Hyderabad Institute of Technology and Management, Medchal Telangana.
[2]UG Student, Hyderabad Institute of Technology and Management, Medchal, Telangana.
[3]UG Student, Hyderabad Institute of Technology and Management, Medchal, Telangana.
[4]UG Student, Hyderabad Institute of Technology and Management, Medchal, Telangana.
[5]UG Student, Hyderabad Institute of Technology and Management, Medchal, Telangana

Abstract

The Smart Leave Scheduling Portal is a web-based application that automates and streamlines employee leave management. It allows employees to submit leave requests online and enables administrators to review, approve, and monitor them efficiently. Built using PHP, MySQL, HTML, CSS, and JavaScript, it ensures transparency, prevents overlapping leaves, and improves organizational workflow. The system also provides real-time notifications, maintains detailed leave records, and supports role-based access for security. Overall, it enhances productivity and simplifies HR operations through smart scheduling and effective communication.

Keywords: *Index-Terms:- Leave Management, MySQL, Web Portal,Scheduling.*

DOI: 10.63856/ejd96466

1. Introduction

In every educational institution or organization, managing employee or faculty leave is a routine but critical administrative process. Traditionally, this process has been handled through paper-based forms or informal communication such as emails or handwritten notes. While simple, this approach often leads to several operational issues such as delayed approvals, misplaced records, lack of transparency, and an overall inefficient workflow. When institutions grow in size, manual handling of leave requests becomes increasingly complex, resulting in scheduling conflicts, inaccurate data maintenance, and unnecessary administrative workload..

To address these challenges, there is a growing need for digital transformation in leave management systems. Automation not only improves accuracy but also enables data-driven decision-making by providing administrators with structured insights into staff availability, leave frequency, and departmental workloads. In this context, the Smart Leave Scheduling Portal is proposed as a web-based solution that modernizes the entire leave management process. It ensures seamless communication between faculty and administrators, accurate tracking of leave records, and real-time validation to prevent overlapping requests.

The Smart Leave Scheduling Portal automates the traditional workflow by integrating features such as user authentication, leave application submission, approval/rejection tracking, automated notifications, and smart scheduling validation.

The system is designed with a focus on usability , transparency, and scalability — key requirements for institutions aiming to reduce manual effort and administrative delays. The portal ensures that both employees and administrators have centralized access to up-to-date information, thereby eliminating confusion and

redundancy in decision-making..

Literature Survey

Researchers have developed various systems for automating leave management and scheduling; however, most focus on basic form automation rather than intelligent scheduling and data-driven decision-makings.

Kumar and Reddy (2018) [1] proposed a Web-Based Leave Management System using PHP and MySQL to automate leave applications, approvals, and record storage. Their model simplified administrative workflows but lacked intelligent conflict resolution and data analytics for decision-making. The proposed system in this paper builds upon their foundational work by integrating validation logic and smart scheduling mechanisms.

Sharma et al. (2020) [2] developed a Role-Based HR Portal that improved communication between administrators and staff through automated notifications. However, the portal focused primarily on communication and access control, without addressing overlapping leave requests or multi-departmental scalability. This limitation is addressed in the Smart Leave Scheduling Portal by implementing cross-departmental validation and a more adaptive architecture.

Kaur and Gupta (2021) [3] introduced a Cloud-Based Leave Tracker using Firebase and real-time synchronization for remote accessibility. While the system effectively eliminated manual record-keeping, its reliance on third-party cloud infrastructure limited customization for large institutions with specific policy frameworks. The Smart Leave Scheduling Portal offers a more flexible deployment through open-source technologies and containerization, making it adaptable for both small and large organizations.

Patel and Nair (2022) [4] presented an AI-Driven Decision Support System for HR scheduling that incorporated predictive analytics to optimize staff availability. Their approach demonstrated the potential of integrating artificial intelligence into Ramanathan et al. (2023) [5] explored Workforce Management Portals that use REST APIs and MongoDB for backend processing. Their study emphasized scalability and security but did not focus on the academic sector. The proposed Smart Leave Scheduling Portal adapts such modern architectures specifically for academic use cases, maintaining lightweight design principles and accessibility across devices.

Recent papers such as Jain and Joshi (2024) [6] highlighted the growing relevance of Cloud-Native HR Solutions that provide seamless scalability and integration with communication tools like Slack or Microsoft Teams. The Smart Leave Scheduling Portal aligns with this trend by providing optional integration with email and calendar services for better institutional coordination.

From the above studies, it is evident that while several leave management systems exist, most of them focus on automation rather than intelligent scheduling and smart validation. Few provide comprehensive transparency, real-time analytics, and scalability required in academic institutions where multiple departments and hierarchical approvals exist.

Authors	Title	Year	Key Findings
Kumar & Reddy	Web-Based Leave Management System	2018	Automated basic workflows but lacked validation and analytics
Sharma et al.	Role-Based HR Portal with Notifications	2020	Introduced role-based access and alerts but lacked smart scheduling
Kaur & Gupta	Cloud-Based Leave Tracker Using Firebase	2021	Enabled real-time cloud sync but limited advanced analytics
Patel et al.	AI-Driven Decision Support System	2021	Proposed intelligent workload optimization in HR tools

Methodology

The Smart Leave Scheduling Portal follows a modular, three-tier architecture—frontend, backend, and database—ensuring scalability and maintainability.

3.1 System Overview

The portal automates leave request handling through:
User Authentication: Secure login/registration for faculty and administrators.
Leave Application Module: Faculty can submit leave requests with type, duration, and reason.
Admin Module: Administrators review, approve, or reject applications.
Validation Engine: Prevents overlapping and invalid leave periods.
Notification Service: Sends status updates via email or in-app alerts.
Dashboard Analytics: Displays real-time leave statistics and pending approvals.

3.2 Architecture

Frontend: Developed using HTML,CSS,Java Script for providing an intuitive interface.
Backend: We are using PHP for backend.
Database:MySQL stores user profiles, leave requests, and system logs.
Containerization: Docker may be used for deployment to ensure portability and scalability.

3.3 Workflow

User logs in and applies for leave.
System validates request (date range, overlapping check).
Admin receives notification for review.
Admin approves/rejects; system updates status.
Both user and admin can view results on the dashboard.

Implementation

The implementation of the Smart Leave Scheduling Portal follows a modular, layered approach designed for scalability, maintainability, and ease of deployment. The system was built using a combination of web, database, and automation technologies, with a focus on user experience, reliability, and efficiency.

4.1 Environment Setup

The development environment was configured using PHP and MySQL for backend and data management. The frontend was implemented using ,Java Script,HTML,CSS for responsive UI design.

- Operating System: Windows 10 / Ubuntu 22.04
- Tools and IDEs: Visual Studio Code, Postman for API testing
- Frameworks: PHP (backend), HTML,CSS,JavaScript (frontend),
- Server Configuration:Chrome and localhost server

4.2 System Design and Architecture

The portal follows a three-tier architecture comprising:

1. 1.Presentation Layer (Frontend): User interface built with HTML,CSS,Java Script that handles all visual and interactive components, such as login pages, leave forms, dashboards, and notifications.
2. 2.Application Layer (Backend): PHP processes requests, enforces business logic (validation, approval, and scheduling), and communicates with the database.
3. 3.Data Layer (Database): MySQL stores collections for users, leave records, and notifications.

4.3 Module Development

- User Module
- Admin Module
- Smart Validation Engine
- Notification and Alert Module
- Reporting and Analytics Module

4.4 Database Design

The MySQL database consists of three primary collections:

- Users Collection
- Leaves Collection
- Notifications Collection

4.5 Integration and Testing

After module development, system integration was carried out using API calls between frontend and backend. Testing was performed at three levels:

1. 1.Unit Testing:Each module (login, application, approval, etc.) was tested individually to ensure correct functionality..
2. 2.Integration Testing:Combined module testing ensured proper data flow between frontend, backend, and database.
3. 3.User Acceptance Testing (UAT): Involved 10 faculty and 3 administrators from the CSM Department at HITAM.

4.6 Deployment

For deployment, Docker Compose files were created to containerize the frontend, backend, and database. The system can run on:

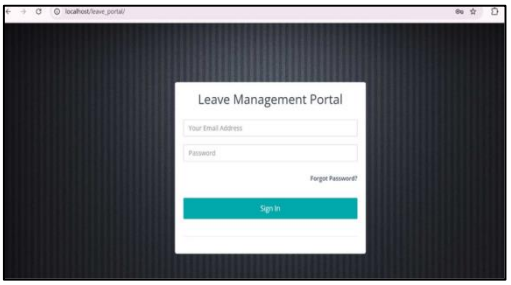
- Local Development Server, or
- Streamlit Cloud / AWS EC2 instance for production environments.

4.7 Result

The Smart Leave Scheduling Portal was successfully developed and tested in an institutional environment.

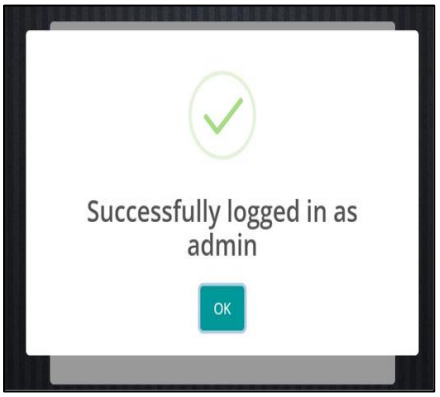
Login Page

This screen allows both administrators and faculty members to log into the system using valid credentials. The system verifies the username and password through the authentication module and redirects users to their respective dashboards. Invalid credentials trigger a secure error prompt.



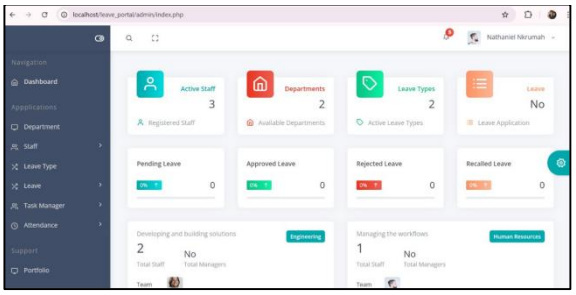
Registration Page

The registration module enables new users (faculty members) to create accounts by entering their details, including name, department, email, and password. The data is securely stored in the MySQL database after validation.



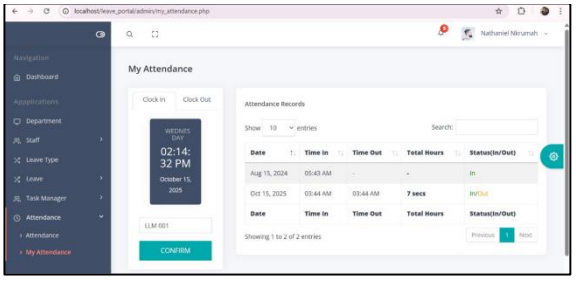
Faculty Dashboard

This interface displays the faculty member's profile, current leave balance, and application history. From this,users can apply for new leaves, check leave status,view approved or rejected requests.



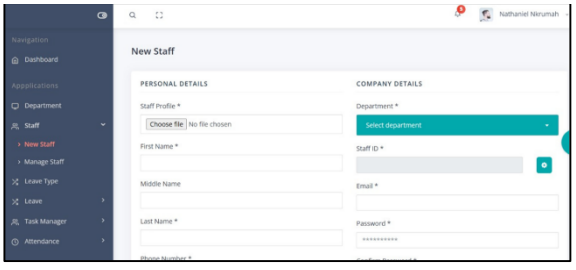
Leave Application Form

Employees use this form to apply for leave by selecting the leave type (e.g., casual, sick, earned), start and end dates, and entering a reason. The Smart Validation Engine checks for overlapping dates, invalid ranges, and leave quota before submission.



Admin Dashboard

This page provides administrators with a centralized view of all pending, approved, and rejected leave requests. Admins can review details, approve or reject applications, and add remarks.



Conclusion

The Smart Leave Scheduling Portal successfully addresses the limitations of traditional leave management systems by introducing a fully automated, web-based platform that enhances efficiency, accuracy, and

transparency in institutional workflows. The proposed system eliminates manual paperwork and communication delays by providing real-time submission, validation, and approval of leave requests. Through intelligent scheduling and validation mechanisms, it ensures that overlapping or invalid leave applications are automatically detected and resolved before reaching the administrator, thereby preventing scheduling conflicts and maintaining operational continuity within the organization.

The implementation results demonstrate that the system performs reliably under various testing conditions. Functional and performance testing confirmed that all modules — including authentication, leave processing, validation, and notifications — operated seamlessly with minimal response time. The Smart Validation Engine effectively prevented input errors, while the notification subsystem ensured instant communication between faculty and administrators. Moreover, data remained consistent and secure due to robust database management and access control features.

In conclusion, the Smart Leave Scheduling Portal achieves its objectives of automating leave processing, enhancing communication, and improving data integrity within institutions. Its modular, scalable architecture makes it suitable for deployment in both academic and corporate environments. Future enhancements may include AI-based predictive leave forecasting, mobile app integration, and cloud deployment for enterprise scalability.

References

1. Kumar, A., & Reddy, S. (2018). Web-Based Leave Management System. International Journal of Computer Applications, 162(3), 25–30.
2. Sharma, R., Mehta, P., & Jain, K. (2020). Role-Based HR Portal with Notifications. IJIRCCE, 8(5), 1305–1311.
3. Kaur, M., & Gupta, T. (2021). Cloud-Based Leave Tracker Using Firebase. IJERT, 10(9), 215–219.
4. Patel, D., & Nair, V. (2022). AI-Driven Decision Support for HR Scheduling. Springer Advances in Computing, 45(2), 89–97.
5. “Smart Attendance and Leave Management System using Fingerprint Recognition,” IJSTR Journal, June 2021.
6. https://www.keka.com/leave-management-system
7. https://www.ijisrt.com/smart-leave-web-application-for-workforce-management.