

Student Leave Management System

1st Dr. M Chandralekha

*Dept. of Computer Science & Engineering
Amrita School of Computing
Amrita Vishwa Vidyapeetham
Chennai, India
m_chandralekha@ch.amrita.edu*

2nd Priyadarshini Jayadurga

*Dept. of Computer Science & Engineering
Amrita School of Computing
Amrita Vishwa Vidyapeetham
Chennai, India
np_jayadurga@ch.students.amrita.edu*

3rd G. Muralidhar Reddy

*Dept. of CSE AIE
Amrita School of Computing
Amrita Vishwa Vidyapeetham
Chennai, India
ch.en.u4aie22014@ch.students.amrita.edu*

4th P. Manoj Kumar

*Dept. of CSE AIE
Amrita School of Computing
Amrita Vishwa Vidyapeetham
Chennai, India
ch.en.u4aie22040@ch.students.amrita.edu*

5th P. Mokshajna Venkata Kruhsna

*Dept. of CSE AIE
Amrita School of Computing
Amrita Vishwa Vidyapeetham
Chennai, India
ch.en.u4aie22041@ch.students.amrita.edu*

6th T. Bhanu Koushik

*Dept. of CSE AIE
Amrita School of Computing
Amrita Vishwa Vidyapeetham
Chennai, India
ch.en.u4aie22061@ch.students.amrita.edu*

7th V.S.V.R Laxman Sai

*Dept. of CSE AIE
Amrita School of Computing
Amrita Vishwa Vidyapeetham
Chennai, India
ch.en.u4aie22068@ch.students.amrita.edu*

Abstract—The Student Leave Management System is an online application that facilitates the automation of students' leave requests. The system allows students to file leaves such as Casual Leave (CL), On-Duty Leave (OD), Medical Leave (ML), and Emergency Relief (EL) with some restrictions and validation processes. In order to promote equity and minimize absences, the system limits CL and OD Leaves combined to four per month, beyond which the students can only opt to use Emergency Leave which requires parental approval via email. Medical Leave also requires documents to be uploaded, which are screened before approval is granted. Each class has an admin (class teacher) who supervises leave application processes and is responsible for each student's administrative department that assures unlikely disorderly approval of applications. The system is developed using Django framework and MySQL as a backend database for systematic filing and retrieval of students' leave records. HTML, CSS and JavaScript are utilized in developing the front end of the application to make sure it is user friendly. Security is further enhanced by establishing a multi factor authentication using one-time passwords for login. The student's dashboard contains a holiday calendar that is user friendly which enhances the student's ability to plan and reduce unnecessary clashes/leaves. Leave statistics, such as total leaves applied, approved, and pending as well as analytics for students who have exceeded leave boundaries, are displayed on a dashboard. At every stage of the leave process from approval, rejection, to additional document requirement, students are notified.

In the case of On-Duty Leave, requests are broken down into Short-Term OD (1 Day – 1 Week) for academic events and Long-Term OD (More than 1 Week) for internships, competitions, or training. Extended OD applications for bulk academic engagements are also supported. A Technical Manager is responsible for troubleshooting any technical issues that arise to ensure continuity of service. Additional system attributes

include cancellation of leave, leave history, delegation of admin rights, and parental emergency leave notifications. These elements increase system reliability and efficiency. Data consistency, security, and accessibility across departments are maintained by the system's centralized architecture. The integrity of the database is maintained through structured query systems and the use of Django ORM (Object Relational Mapping) simplifies the application and database interaction. To avoid any unauthorized access, appropriate measures such as QR codes for login and limited admin access options have been put in place. Students, parents, and administrators can freely communicate with real-time emails through the approval and notification integration. This system autonomously tracks student leaves, making it impractical, effortless, and safe for educational institutions to manage their leaves, which helps integrates technology seamlessly into education. This system in combination with real-time data visualization, an approval hierarchy, and secure login improves the overall efficiency of the administration as well as reduces manual work, automates the error prone processes, and provides a just and open system for leave approval.

Index Terms—System for Managing Student Absences, Django Framework, MySQL Server Database, Web Application, Approval For Leave, One Time Password, Leave Balance, Admin Console, Parent's Consent, Postal Alerts, On Duty Absence, Sick Leave, Urgent Leave, Information Protection, Vacation Schedule, Group Leave Request, Senior Technical Supervisor.

I. INTRODUCTION

SLMS or Student Leave Management Systems have become a necessity for academic institutions globally over the years. The information from MarketsandMarkets shows that the Student Information System domain will experience a compound annual growth rate (CAGR) of 15.9% between 2024 to 2029

due rise in the use of these systems for security, amalgamating enrollments, monitoring attendance[, and recording academic achievements. In addition to this, the global spending on student management systems that stood at USD 8.5 billion in 2023 is estimated to grow to USD 18.3 billion by the year 2032 which translates to a CAGR of 8.9%. These reveal the growing importance and the modernized trends of education SMS systems administrating the institutions. It has been observed that manual and paper bound techniques of keeping track of student's profiles, attendance records, and request forms were dominant amongst educational intuitions. Such processes were slow and took a lot of time as well. These methods also allowed a wide margin of error, which negatively afflicted the communication within students, teachers, and administration. The innovative digital solutions put forward Automated systems, which made the communication simpler along with the management of data to achieve these tasks. Shifting from traditional approaches to advanced software solutions has greatly enhanced the efficiency of operations and the accuracy of data in educational institutions[7]. Our project takes this further with the implementation of a holistic Student Leave Management System aimed at automating and streamlining the processes surrounding the application and approval of leaves. With features like OTP verification for logins, medical leave document uploads, emergency parental approval emails, and real-time notifications, the system provides efficient and transparent workflows. In addition, a statistics dashboard displaying leaves and a holiday calendar was designed to facilitate better decision-making for students, and administrators which is further aided by an informed decision-making process. This solution alleviates administrative workload while enhancing the experience of students who now have a dependable and convenient method of managing their leave requests. Apart from automating leave management, our system utilizes role-based access control (RBAC) [10] to ensure that none unauthorized can access or manage specific data. Every class advisor (admin) is provided with leave applications for their corresponding departments, thus preventing cross-disciplinary approvals from unauthorized personnel. Shifting from traditional approaches to advanced software solutions has greatly enhanced the efficiency of operations and the accuracy of data in educational institutions[7].

Our project takes this further with the implementation of a holistic Student Leave Management System aimed at automating and streamlining the processes surrounding the application and approval of leaves. With features like OTP verification for logins, medical leave document uploads, emergency parental approval emails, and real-time notifications, the system provides efficient and transparent workflows. In addition, a statistics dashboard displaying leaves and a holiday calendar was designed to facilitate better decision-making for students, and administrators which is further aided by an informed decision-making process. This solution alleviates administrative workload while enhancing the experience of students who now have a dependable and convenient method of managing their leave requests. Apart from automating leave

management, our system utilizes role-based access control (RBAC) [10] to ensure that none unauthorized can access or manage specific data. Every class advisor (admin) is provided with leave applications for their corresponding departments, thus preventing cross-disciplinary approvals from unauthorized personnel. Leave requests may be submitted under four categories: Casual and On-Duty Leave (CL), Medical Leave (ML), and Emergency Leave (EL) with their own eligibility criteria. Both types of leaves are limited to four each per month; otherwise, one is required to take Emergency Leave, which must be approved by a parent through email. Medical Leave applications must be backed by additional support, like a hospital receipt which the admin checks before approving. This helps in making sure that no one is discriminated against and the school's policies are followed.

In order to maximize convenience, the system provides real-time email notifications so that students are informed of every step taken on their leave applications. Notifications include confirmation of application sent, admin action (approve or reject), and request for more documentation. The student dashboard also comes with a holiday calendar so that leaves may be planned to avoid clashes with classes. For On-Duty leaves, the system accepts short-term (1 day to 1 week) and long-term (more than 1 week) leaves for academic contests, internship programs, and professional training programs. This feature in particular enables students participating in long-term academic activities to apply for leave in bulk. The administrator's dashboard acts as a central control room where student leave patterns can be monitored with visual analytics. Gauged figures of total applied leaves, pending requests, approvals, and cases where leave limits have been exceeded are portrayed graphically. These trends assist administrators in making decisions, ensuring compliance with institutional policies, and improving them accordingly. The Technical Manager's dashboard also provides additional registered students and faculty members along with system feedback which improves governance of the system. With this system, students, faculty, and administrators will have increased security and transparency while the system consolidates all components of leave management, promoting efficiency.

II. LITERATURE SURVEY

This study presents an Internet-based leave management system designed for academic institutions. Using HTML, PHP, and MySQL, the system automates the submission and approval of leave requests, mitigating issues associated with manual paper-based processes. The implementation is effective in minimizing paperwork for the academic and administrative sectors within the institution.[6].

An Online Leave Management System by Alade et al (2022) states that the goal of the research is to create a Web Based Leave Management System that aims to solve issues regarding the inefficiency and delays in the processing of leaves by human resource staff. The system was built with HTML and CSS for the frontend, and Python with Django as the backend framework, using SQLite for database management.

Evaluation revealed a 76.67% of users being satisfied with the system. The WBLMS leads to enhanced user satisfaction, improved productivity, and effective utilization of resources for processing leaves.[1].

This study optimizes leave management processes for Citra Bonanza Express with a proprietary web-based leave application solution. Alleviating issues associated with traditional leave application systems, Munawir et al. (2024) built the system around an employee leave application portal. Their waterfall model approach to system implementation yielded positive gains in efficiency, precision, and clarity. Furthermore, the shift to an automated system enabled the moderation of leave request submission and acceptance, reduced inaccuracies, and offered managers timely insights about employee leave balance for better decision making[5].

By Ikuomola in 2017, Adaptive Electronic-Leave Management System proposes an adaptive electronic leave management system aimed at administering standard leave rules while automating the capturing and storage of leave data. It also has two tiers of authority for approvals, and is adaptive to different devices without distortion of display. Implemented in PHP and MySQL, with supporting tools jQuery, Bootstrap, JavaScript, HTML, and AJAX, it fosters a 'paperless office' ideology, thereby saving time, human effort, and monetary resources needed for conventional leave management[4].

Online Leave Management System by Praveen et al. (2020), their project focuses on the creation of a web-based application designed for managing leaves in a college. This application is for internal networks only and can be used across the institution. Through the application, users can apply for leave as well as manage approvals from a centralized location. Main features offered are submission of leave application, approval workflows, cancellation of leave, and forwarding of requests to heads of departments. The system stores records of students' leaves for efficient tracking and management of the students' leave history which can be retrieved easily using the database[10]

Employee Leave Management System by Adamu (2020) This research develops a system to manage employee leave through automation to replace the traditional manual approach. I implemented the system with CSS, Javascript, HTML, MySQL, and PHP for the user interface. It is designed to work on the Windows operating system. The system allows requesting leave on time, lessens paperwork, and improves the efficiency of controlling employee leave in an academic institution. The system has been found to work perfectly without errors and users expressed their satisfaction[2]

Withdrawal Scheduling Algorithm to Optimize Services in the Nigerian University System By Stephen Others (2018) discusses an academic staff leave management system that integrates service delivery centric parameters such as staff grade mix and student number per lecturer. The algorithm was developed in MATLAB, it evaluates leave requests, process them, and schedules them during the time which service delivery is least impacted. The results show effective control of academic staff leave without compromising service quality[3]

Design and Implementation of an Electronic Leave Management System: A Case Study of Al-Hikmah University by Olaniyi et al. (2019), this report outlines the creation of a leave management system at Al-Hikmah University. The system defines three major users: Employee, Leave Manager, and Administrator, each assigned particular duties. Employees can file and withdraw leave requests, while Leave Managers process the requests by either granting or denying them. Administrators control employee registration as well as the overall leave activities. Some security control is put in place to block unauthorized access and validate form entries, thereby making the leave management system secure and efficient[9]

Okonigene and Adekanle (2016) in their paper 'Web-Based Staff Management System' proposed an online employee management system to improve transparency and accountability in the organization. The system manages a databank of employee details such as employment history, education, leave balance, and performance evaluations. With the use of PHP and MySQL the system also minimizes time wasted during the employee record and leave application retrieval processes, promoting efficiency in personnel information management[8]

Muhammed et al. (2015) 'An Automated Leave Management System for Federal University of Technology Minna' This study seeks to design and develop an online leave management system for the Federal University of Technology, Minna. The system aims to automate the leave application and approval processes to minimize the problems caused by the use of manual processes. Using several web based techniques, the system is designed to be easy to use with a fully featured leave record database that stores all the relevant information which improves the management functions of the institution's administration[7]

III. METHODOLOGY

The Student Leave Management System is aimed to automate leave applications, approvals, and tracking within educational institutions, enhancing their security. To maintain secure access and prevent unauthorized leave requests, it includes AES encryption, RSA key management, and OTP-based authentication for secure access. Real-time notification systems and dashboards along with database management structures increase the overall effectiveness for students and administrators. The methodology takes a systematic approach by integrating secure logins, automatic decision making, and user-friendly interfaces that provide transparency and ease in the leave management system.

A. System Design and Architecture

The system architecture uses three-tier structure where the front end is separated from the back end and the database. The frontend uses HTML, CSS, and JavaScript which guarantees interactivity and user friendliness of the interface. The backend is developed through the Django framework and takes care of role-based access control, leave request processing, user authentication, and sends email notifications. MySQL is used on the database layer for structured storage of student leave

records, approval history, and user auth details. Security is strengthened and information is received in realtime communication. The local server ensures maximum efficiency

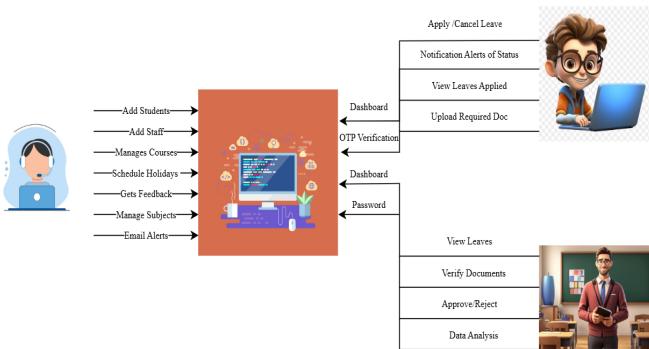


Fig. 1. Model Architecture

in access and processing the data while hosting the system data facilitates for privacy concerns. Both students and staff (admins), along with the technical manager, are no longer joined into looped coordination for approval processes because the leave is applied through a rule based workflow automation. Students can apply for leave through the system[2] alongside stating the type of leave, purpose and requisite documents, if any. In case a student has used up Casual or On-Duty leaves over the limit of four per month, the system prompts them to use Emergency leave by disabling casual options. After a student uses an Emergency Leave, they have to submit a parental approval through email with documents and after that send it to the class advisor for admin verification. During the wait period, the student is kept in the loop of what stage their leave request is through instant email alerts. Inclusion of holiday calendar allows students to plan the leaves easily. Student and staff account automation, feedback collection, and other technical issues is solved easily by the technical manager, ensuring the system functions smoothly.

B. Unified Modeling Language Diagrams Importance

Unified Modeling Language diagrams are crucial for visual modelling from software engineering perspective. They are used to plan, design, and document systems and software architectures. Before any step of implementation takes place, developers, designers, and participants can start to understand the system functionalities thorough interactions. Diagrams make it easy to condition how various elements relate and provide means of evaluating system needs, issues, and development ‘relational’ harmony. UML also assumes prominent place of system developed UML documents collaborative attractiveness and communicative clarity, as well as maintenance and scalability of software solutions. As it happens with words types of UML diagrams serve different purposes within system design. The class diagram lays out the borders of a system and established relationships between classes and their attributes

as well as methods. Focus on time different components interact with each other is captured in sequence diagrams where objects are shown sending and receiving messages in steps. There are different ways of showing how data can move around in a system and those ways are captured by Data Flow Diagrams DFDs. Outputs, inputs, information, and processing storage locations are visually explained. Activity diagrams show workflows and business logic in terms of sequences of activities, decision making, and control flow. Interactions with the system and its users are captured with use case diagrams. It assists in marking the locations for inputs, processes, outputs, and storage. Activity diagrams represent a workflow in a business context showing a sequence of execution and control activities, decisions, and flow of control. Use case diagrams represent users of the system and their corresponding permissions as different actors like students, admins, and the technical manager with respect to some functionalities of the system. UML diagrams are important for the Leave Management System since they capture functional and behavioral features of the application. The class diagram captures system entities like students, staff (admins), and the technical manager with relevant system functions and data. The sequence diagram describes students applying for leave and admins processing their approvals. The DFD indicates the flow of data from leave requisitions to the processing of approvals and subsequent notifications. The activity diagram describes the steps in leave request processing, the steps and decision points there in, and the dependent notifications while the use case diagram abstracts the users and their interactions. All these diagrams help in understanding the system in an integrated form for effective implementation and maintenance.

1) *Sequence Diagram:* The illustration of the sequence diagram depicts the components and functionality of the Leave Management System, it focuses on the parents, students, the technical manager, administrators, the email service, the database and the system itself. The process begins when the student initiates a leave application whether it is OD, ML, Casual or an Emergency leave. The system then checks the leave balance of the student in question using a query to the database. Should the leave balance be within the allowable account limits (≤ 4), the system will send the application to the administrator for an approval or rejection of the request, it then updates the leave status in the database and sends a notification to the student regarding the approval or rejection. If the students leave balance has exceeded the limits (> 4), then the system will suggest Emergency Leave Integrated with a parental approval request that is sent via email. A parent composes an email directed to the administrator containing the students' details, reason, and evidence. The system checks the parent's email in the database and returns a verification status. The admin either grants or denies the Emergency Leave application, modifies the leave balance, and notifies the student. Besides, students can inform the technical manager with system issues, who resolves them and informs the student when done. The system guarantees immediate changes, preserves data integrity, and automates notifications through

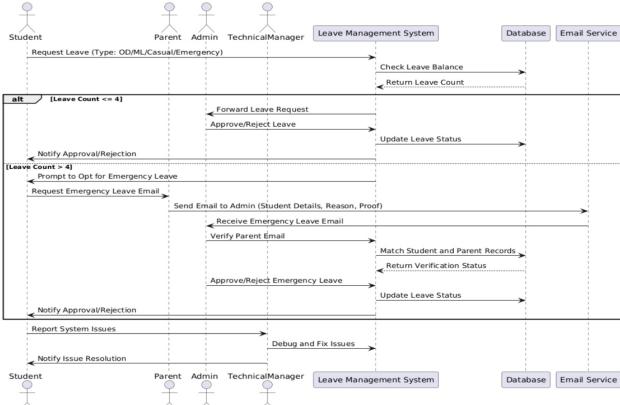


Fig. 2. Sequence Diagram

emails.

2) *Use Case Diagram*: A Use Case Diagram outlines how users interact with specific functions of a system. It demonstrates how various users may execute different actions. Use case diagrams are crucial in software engineering for establishing system requirements as they aid in providing structure, enhancing precision, and ascertaining that all user roles and systems' interactions are sufficiently captured. The main actors in the Leave Management System are Students, Admin (Class Adviser), Parents, and Technical Manager.

Students are able to apply for leave, check their leave status, and receive notifications for approvals or rejections. In cases where a student's leave exceeds the allowed limit, they have the option to request an Emergency Leave that requires parental email verification. Admins manage student leave records and also approve or reject leave requests. Parents can approve emails on Emergency Leave verification. Technical Managers resolve system issues and register students/staff and collect feedback about the system. The system also communicates with the email service to send notifications at different stages. All interactions provided in this diagram help design and implement the system efficiently.



Fig. 3. Use Case Diagram

3) *Class Diagram*: The class diagram of the Leave Management System gives a clear structure on how student leave applications can be managed. The Student class enables students to apply for leave and subsequently track its status. The Admin approves or disapproves the requests, processes them into reports, and modifies leave policies. The Leave Request class has attributes such as student id, type of leave, date range, and status. The System class sends email alerts to students and

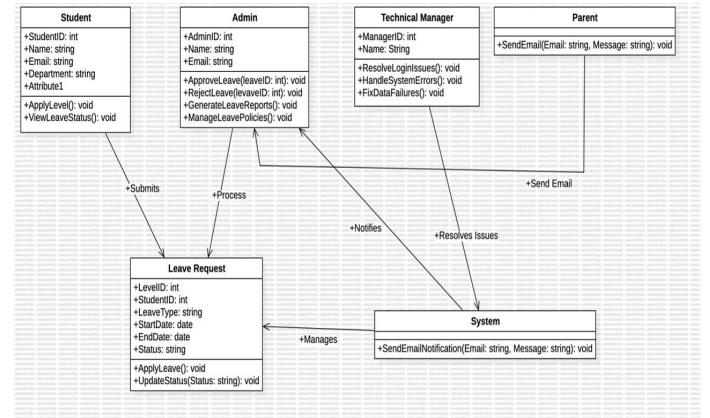


Fig. 4. Class Diagram

their guardians, while the Parent class is only informed through emails. The Technical Manager's role is to make certain the system remains operational by fixing login problems, resolving any errors, and correcting data failures. All these classes are interconnected to provide complete leave management, where students submit requests, admins process them, the system manages notifications, and technical managers address system problems so that the Leave Management System is operational.

C. Data Flow Diagram-Level 0

A Level 0 Data Flow Diagram (DFD) or Context Diagram captures the essence of the entire system. In this case, the Leave Management System is represented as a single process. It identifies major processes in conjunction with the system including Students, Admin (Class Advisor), Parents, Technical Manager, Database, Email Service. In our Leave Management System, the process begins when students apply for leave which is then sent to an admin for approval. If the student stays within the leave count, then the request will be accepted and the student will be informed of his approval or rejection to the request. Should the number of leaves exceed four however, the student will need to apply for an Emergency Leave which will require parental approval initiated over email. It is the responsibility of the admin to verify the parental email along with the necessary documents before deciding. The email service is used to notify users of updates in regards to their leave. In addition to this, the technical manager attends to system problems, controls registration of students and staff, and makes sure that everything is functioning properly. The database is the most secured area for all the leaves applications and their status, user information, and system logs. The Level

0 DFD enables users to see how other users interact with the system in relation to the data and notes that these interactions are important reconciled combination to manage leave.

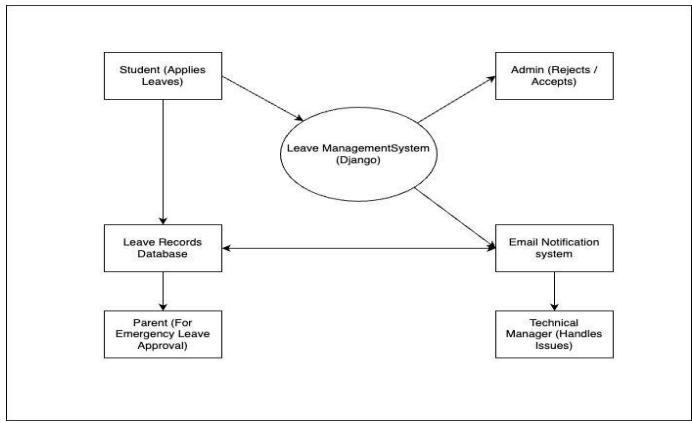


Fig. 5. DFD Diagram

D. Flowchart of the Leave Management System

The **Leave Management System** follows a structured flow, ensuring seamless handling of student leave applications, approvals, and notifications. The system is designed with three primary roles: **Student, Admin (Class Advisor), and Technical Manager**, each performing specific tasks to maintain efficiency and security.

1. Student Workflow:

- The **student logs in** using OTP verification.
- After successful authentication, the student **applies for leave** by selecting the type (Casual, On-Duty, Medical, or Emergency).
- The system **checks the leave balance**:
 - If the student has **not exceeded 4 leaves**, the request is forwarded to the **admin** for approval.
 - If the limit is **exceeded**, the student is prompted to **apply for Emergency Leave**, which requires **parental email approval** and **document submission**.
- The student **receives email updates** at every stage, including submission, approval, rejection, or additional document requests.

2. Admin (Class Advisor) Workflow:

- The **admin reviews leave applications** and checks supporting documents (if applicable).
- Based on the details, the **admin approves or rejects the request**.
 - If rejected, the **student is notified** with a reason.
 - In case of **Emergency Leave**, the admin verifies the **parental email** before making a decision.

3. Technical Manager Workflow:

- The **technical manager** is responsible for system maintenance and resolving reported issues.
- He manages **student and staff registrations**, ensuring every user has proper credentials.

- Additionally, he maintains **database security and email service integration**.
- The manager also receives and addresses **feedback from both students and admins**.

Final Process:

- The **leave status is updated** in the database, and students can track their leave history.
- The system **prevents students from applying for additional Casual or On-Duty Leaves** once they exceed the monthly limit.
- **Admins can delegate approvals** if they are unavailable.
- The **dashboard displays analytics**, including approved, pending, and rejected leaves.

This **flowchart-based approach** ensures an efficient, transparent, and well-structured **Leave Management System** that automates leave processing while maintaining security and accountability.

IV. RESULTS AND FEATURES

A. User Role Management and Authentication

The specified system adopts Role-Based Access Control (RBAC) technique to restrict and organize access to the portal in the system by assigning permissions to students, staff, and the Technical Manager according to their duties. The Technical Manager is at the highest level in the hierarchy. He ensures the system is working well and oversees its operations. This role is concerned with solving issues pertaining to the portal and supervising both students and staff to ensure the portal is not congested. He is the sole administrator and as such, is the one who makes the decisions regarding the integrity and performance of the system. Furthermore, he liaises with students and staff to ensure their concerns are addressed in a timely manner. Staff (Class Advisors), who are also the second authority, have important responsibilities with regards to students' leave allowing them to check, approve or deny the leave requests. Their decision-making process involves confirming documents uploaded through the system as Medical Leave (ML) and checking Emergency Leave (EL) requests where parents have to approve through email. Staff members also analyze leave trends, monitor students that go over the threshold of 4 Casual or On-Duty Leaves a month and check for attendance periods not taught.

Every staff member is assigned to a specific department to prevent teachers from one department from handling leave requests of another department. These staff members can also suggest improvements to the Technical Manager which provides further development to the system. Students, the main users of the system, have a personalized dashboard from which they can apply for leaves, monitor their leave history, and get information on the status of their leave in real time. The system has stringent policies regarding students' leaves of absences - students cannot apply for Casual or On-Duty Leaves after surpassing the limit of 4 leaves in a particular month, and is subsequently forced to take Emergency Leave which has

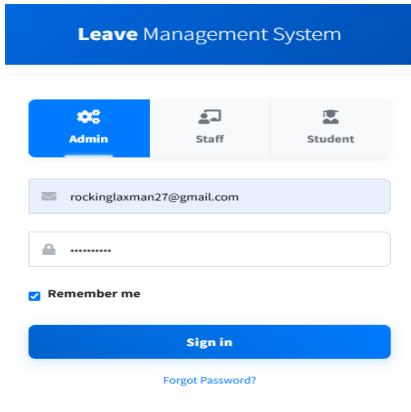


Fig. 6. Login Page

parental email verification as a prerequisite. Moreover, students are able to use the holiday calendar to check for conflicts with college holidays before submitting leave requests and analyze their leave trends visually. Through RBAC, the system ensures that every user has certain permissions that must be followed increasing security, transparency, and efficiency when managing student leaves.

1) Technical Manager Role and Functionalities: A specific login page is provided for the Technical Manager, who is expected to conveniently choose one out of the three options presented on the page – Technical Manager, Staff, and Student. The user takes the first step by clicking the Technical Manager (Admin) button, after which they must authenticate themselves by entering their email and password. When logging in for the first time, users are taken to the Admin Dashboard which is their main hub for managing the entire leave management system. On the Admin Dashboard, the Technical Manager

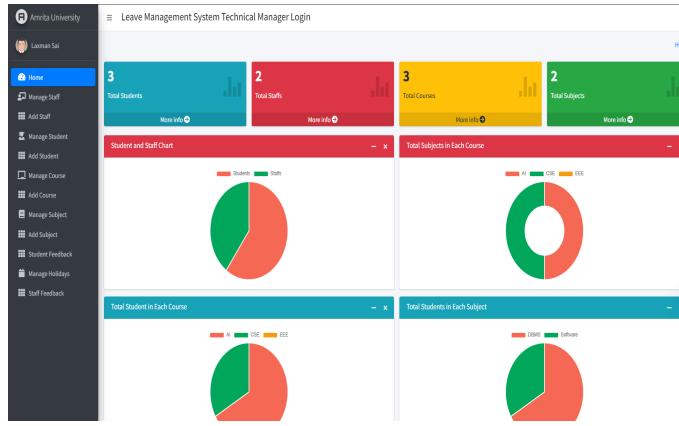


Fig. 7. Technical Manager Dashboard

has access to intricate details in the form of pie charts, bar graphs, and other statistical illustrations. Such visual analytics summary the different departments, the number of students in each department, number of faculty members, subjects available, and the trends regarding leave applications. With these analytics, the Technical Manager is able to effectively

track system activity, leverage patterns in leave requests, and maintain the flow of operations smoothly. Other than data

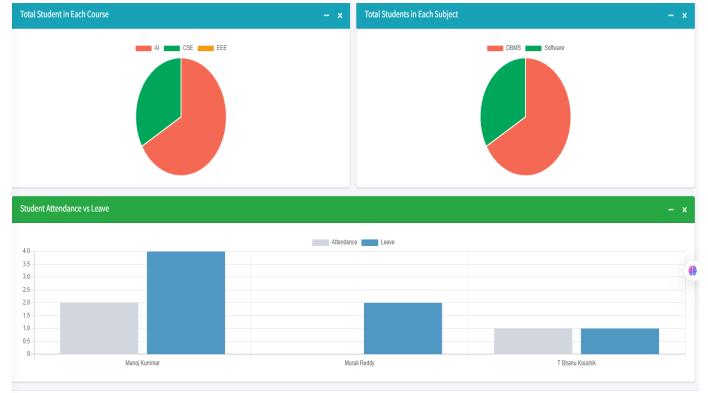


Fig. 8. Technical Manager Dashboard

analysis, the Technical Manager holds the highest administrative position, which includes responsibility for managing student and staff accounts, providing resolution for technical and operational issues, and guaranteeing that the portal is in a functional state devoid of interruptions. They are also the recipients of other comments and suggestions from students and staff regarding leave policies, systems performance, and general system usability. This function particularly is important for the overall system integrity, security, and efficiency of the Leave Management System. As with most other systems,

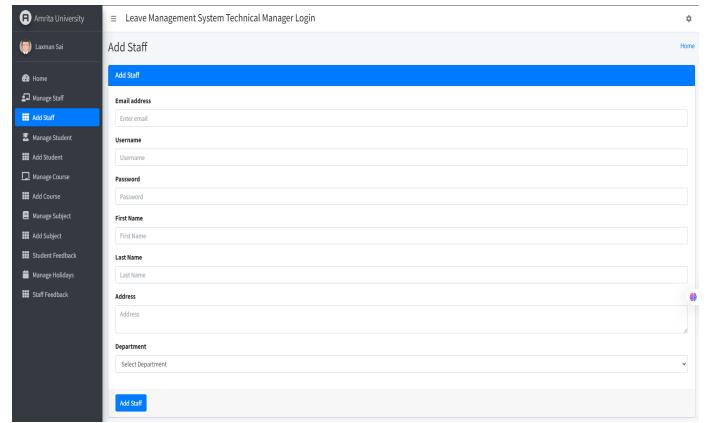


Fig. 9. Adding Staff Members

the Technical Manager can add and control courses within the system. To add a new course, the Technical Manager simply needs to type the name of the course to an input box, and press the Ok button. The procedure is simple. Once submitted, the course will automatically register in the system and update the database. The manager is also able to view and control all existing courses, seeing the total amount of courses available along with their details. This feature makes sure course related data is organized which in turn makes it easier for students and staff to associate themselves with

the right course. Aside from managing courses, the Technical Manager also recruits new staff and allocates subjects. While creating a staff account, the manager has to fill in the name of the staff, email, login username, password, department (where the course is assigned), and address of the staff. This makes

Fig. 10. Adding Students

certain that the staff's accounts are created with clearly defined roles as credentials for the portal are assigned to the staff. In the same way, the manager should also specify the name of the subject, the course that is associated, and the staff member that is assigned with the subject as the manager must add subjects. These features enable the Technical Manager to effectively structure and sustain the academic base of the system which positively contributes to the organized and automated leave approval process. The Technical Manager is enabled to create

Fig. 11. Manage Subjects

and control holiday schedules in the system for both the staff and students. He is able to insert holidays into the system through the admin console by selecting a date and inputting a name as well as description for the holiday. The holidays are then displayed in the dashboards of staff and students for them to strategize their actions. This approach is effective in averting unreasonable clashes between college holidays and leave applications by ensuring students do not request leave on declared holidays. Moreover, this information is useful

to staff members as well in scheduling academic activities, exams, or other important events. The non-working days or holidays are marked on the calendar which makes it easy for users to view and track current and official non-working days. This guarantees improved communication and management of holidays in the institution.

Fig. 12. Manage Holidays

2) Staff Role and Functionalities: Following the standard procedure of logging in with their email and password, the individual staff users are seamlessly redirected into their respective dashboards which provide a detailed summary of the activities assigned to them. The dashboard shows the staff members the total number of students assigned to them, total attendance, total leaves taken by the students, and the subjects taught by them. In addition, the staff is presented with graphical analysis as leave status charts, subjects attended to, and students' attendance data so that trends can visually be analyzed. Team members have a number of primary func-

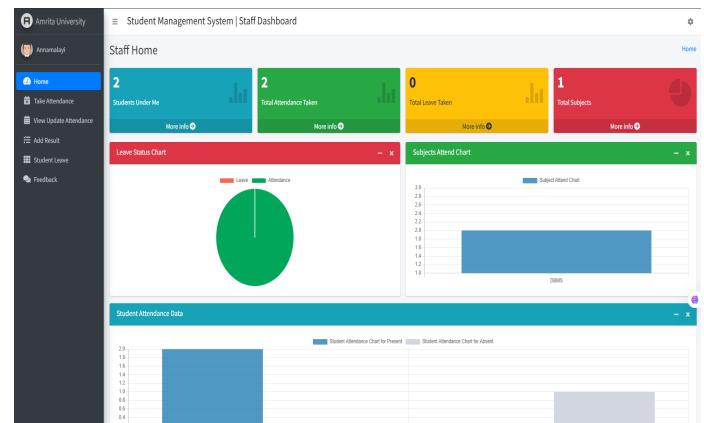


Fig. 13. Staff Dashboard

tions, such as marking attendance, editing attendance records, inputting student grades, and giving feedback for all aspects of the learning process. One important function of staff personnel is to process and supervise applications for students' leaves of absences. They are able to grant or deny leaves on the basis of adequate examination of the documents presented for

a leave. When processing leaves, staff members are able to see what type of leave has been requested, namely, Casual Leave (CL), On-Duty Leave (OD), Medical Leave (ML), or Emergency Leave (EL). On-Duty Leave (OD) is further divided in two categories; short term OD (1 day to 1 week) for hackathons and college events and long term OD (greater than a week) for employment internship and job related activities. In order to keep a systemized record of absences,

ID	Student	Leave Type	Period	Reason	Documents	Applied On	Status	Action
7	Murali Reddy 10-5	Casual Leave	From March 30, 2025 To March 31, 2025	View Reason	No documents	Mar 28, 2025	Rejected	Rejected
6	Murali Reddy 10-5	Casual Leave	From March 29, 2025 To March 30, 2025	View Reason	No documents	Mar 28, 2025	Approved	Approved
5	Murali Reddy 10-5	Casual Leave	From March 29, 2025 To March 30, 2025	View Reason	No documents	Mar 28, 2025	Rejected	Rejected
4	Manoj Kumar 10-4	On-Duty Leave Short Term	From April 01, 2025 To April 10, 2025	View Reason	No documents	Mar 28, 2025	Approved	Approved
3	Manoj Kumar 10-4	Casual Leave	From March 29, 2025 To March 29, 2025	View Reason	No documents	Mar 28, 2025	Approved	Approved
2	Manoj Kumar 10-4	On-Duty Leave Short Term	From March 30, 2025 To March 31, 2025	View Reason	No documents	Mar 28, 2025	Approved	Approved
1	Manoj Kumar 10-4	Casual Leave	2025-03-30	View Reason	No documents	Mar 28, 2025	Approved	Approved

Fig. 14. Student Leave Application

students are limited to a total of 4 absences per month for both Casual Leave (CL) and On-Duty Leave (OD) so that excessive lost time is avoided. Holidays are automatically part from the leave count in order to ensure there are no conflicts with official college holidays. In the case of Medical Leave (ML), staff cannot just grant approval without first having authentic receipts from the hospital uploaded, which aid in proving the patient indeed visited the facility during the stated time period. Equally, teaching staff for Emergency Leave (EL)[8] must ensure that parental consent emails were sent from the student's guardians prior to allowing the leave request. After the decision is made, the student is automatically informed via email whether the leave was granted or denied. Alongside overseeing student leaves, staff are also able to

Fig. 15. Student Attendance

create elaborate reports regarding leave summaries, attendance

records, and student scores. These reports may be exported for documentation purposes and further analysis in Excel. Being able to analyze leave trends along with attendance data enables staff to make informed decisions and offer necessary support in cases where students's leave limits or irregular attendance patterns are met. The system guarantees transparency and accountability making total efficiency in the previously time-consuming processes.

ID	Student	Leave Type	Period	Reason	Documents	Applied On	Status	Action
7	Murali Reddy 10-5	Casual Leave	From March 30, 2025 To March 31, 2025	View Reason	No documents	Mar 28, 2025	Rejected	Rejected
6	Murali Reddy 10-5	Casual Leave	From March 29, 2025 To March 30, 2025	View Reason	No documents	Mar 28, 2025	Approved	Approved
5	Murali Reddy 10-5	Casual Leave	From March 29, 2025 To March 30, 2025	View Reason	No documents	Mar 28, 2025	Rejected	Rejected
4	Manoj Kumar 10-4	On-Duty Leave Short Term	From April 01, 2025 To April 10, 2025	View Reason	No documents	Mar 28, 2025	Approved	Approved
3	Manoj Kumar 10-4	Casual Leave	From March 29, 2025 To March 29, 2025	View Reason	No documents	Mar 28, 2025	Approved	Approved
2	Manoj Kumar 10-4	On-Duty Leave Short Term	From March 30, 2025 To March 31, 2025	View Reason	No documents	Mar 28, 2025	Approved	Approved
1	Manoj Kumar 10-4	Casual Leave	2025-03-30	View Reason	No documents	Mar 28, 2025	Approved	Approved

Fig. 16. Student Leave Report

3) *Email Notification Alerts:* For automated real-time email notifications through the Leave Management System, we implement the built-in class `django.core.mail` of Django. The sending process of the email us performed using the class `EmailMessage` with the function `send_mail()` which are set via SMTP backend of django through `EMAIL_HOST`, `EMAIL_PORT`, `EMAIL_USE_TLS`, `EMAIL_HOST_USER`, `EMAIL_HOST_PASSWORD` and other necessary configuration settings. As the student applies for a leave, the system saves the request and at the same time sends a confirmation email which is followed by staff checking the request against attached documents for Medical Leave (ML) or, in the case of Emergency Leave (EL) parental email approval, for some students after the limit of 4 CL/OD leaves per month are exceeded. Email notification is automatically done by the system based on the students leave status, either being approved or rejected of which in turn they get an email update. Where there is a leave limit exceed, an automated notification is issued and the student instructed to send a parental email to verify. Staff acting on verified requests issue email an automation instruction and the students being sent the email after being informed about the processing of the request. With no need of checking the portal manually, students get their emails immediately informing them about the status.

4) *Student Role and Functionalities:* Students hold an important function in the Student Leave Management System[6]. They possess much less permissions than the Technical Manager (admin) and Staff (class advisors), but their participation is critical to the system functioning. When students log in the portal, they click on the "Student" button and type the registered email and password. The authentication process entails an OTP verification, where a one time password is sent to the students email. Students are then prompted to log in to the system after successfully entering the OTP. The

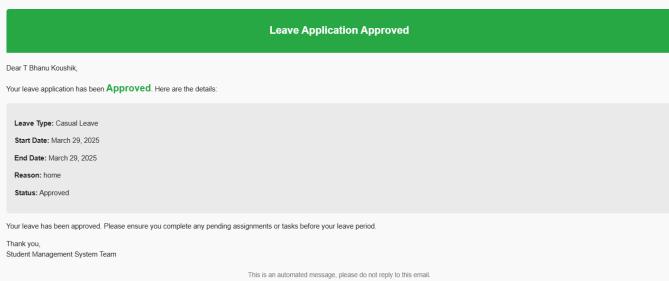


Fig. 17. Email Notification

Fig. 19. Students Leave Application

student dashboard displays important information such as total attendance, total classes attended, total classes missed, and total registered subjects. These numerical values are converted into visual representations such as pie charts and bar graphs to aid the students in having a better understanding. On

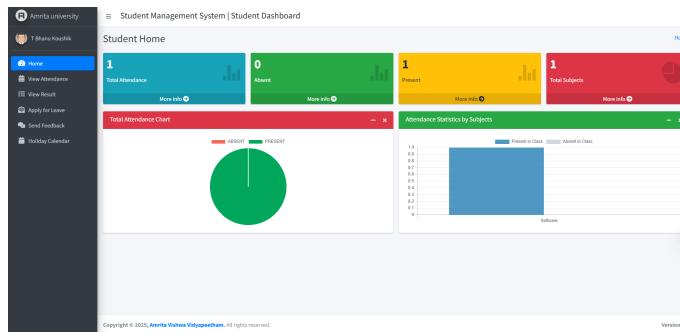


Fig. 18. Students Dashboard Visualization

the left side of the dashboard, students have a navigation panel with function options such as Home, View Attendance, View Results, Apply For Leave, Send Feedback To Technical Manager, and View Holiday Calendar. The system allows its users to follow a Role Based Access Control (RBAC) model which implies that students are able to view their attendance records, results, and leaves history without having the capability of changing administrative settings. With regard to leave, however, students are in full control. Students can request different leave types such as Casual Leave (CL), On-Duty Leave (OD), Medical Leave (ML), and Emergency Leave (EL). In order for Medical and Emergency Leave to be granted, appropriate medical documents must be uploaded. The system makes it simple by calculating the total number of leaves taken automatically per month. If the student has used four leaves, they will go over their limit, whereby the option for Casual Leave will be disabled. This only enables Emergency Leave, which is parental permission via email. After submission, the appropriate personnel (class advisor) either approves or denies the leave and notifies the student via email. When applying for leave, students must enter details such as leave type, start and end dates, and a valid reason. If Medical or Emergency Leave is selected, the document upload becomes mandatory; otherwise, the leave request cannot be submitted. Students can

also track their leave history and check their holiday calendar to plan accordingly. The system ensures complete privacy and security by storing all details in a MySQL database. Students cannot view other students' information, as data privacy is enforced through secure authentication mechanisms, including email-based notifications and access controls.

ID	Type	Dates	Reason	Status	Applied On
11	Casual Leave	2025-03-29 to 2025-03-30	home	Pending	March 28, 2025, 3:24 a.m.
10	On-Duty Leave (Short-term)	2025-03-29 to 2025-04-03	hacathon	Rejected	March 28, 2025, 2:57 a.m.
9	Casual Leave	2025-03-29 to 2025-03-30	out	Rejected	March 28, 2025, 2:54 a.m.
8	Casual Leave	2025-03-29	home	Approved	March 28, 2025, 2:38 a.m.

Copyright © 2025, Amrita Vishwa Vidyapeetham. All rights reserved.

Version 1.0

Fig. 20. Students Leave History

Overall, the student module is designed to provide a seamless and secure experience, allowing students to efficiently apply for leave while ensuring transparency, accountability, and data security. The integration of email alerts ensures that students are kept up to date about their leave application status without needing to constantly check the portal. The system's structured workflow enhances efficiency, security, and usability, making it a highly effective solution for leave management in an academic environment.

Academic Calendar - March 2025						
Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

Fig. 21. Students Holidays Page

V. DATABASE DESIGN

Organizing a database properly is vital because the information can be retrieved and stored in the most efficient way possible in our Student Leave Management System (SLMS). The system does not only need secure authentication and role-based access, but also provides leave management, therefore making the database as the main part of the system. The database is meant to keep student profiles, student authentication information (which is encrypted), leave records, and administrative information. It has different tables under Students, Admins, LeaveRequests, OTPVerification, and EncryptedKeys to take care of specific tasks. The Students table contains a student record with his/her ID, name, email address, a profile picture, and encrypted password. In the LeaveRequests table, leave applications do not only have casual leave as a category, but also medical leave, emergency leave, and on duty leave for easier tracking and approval.

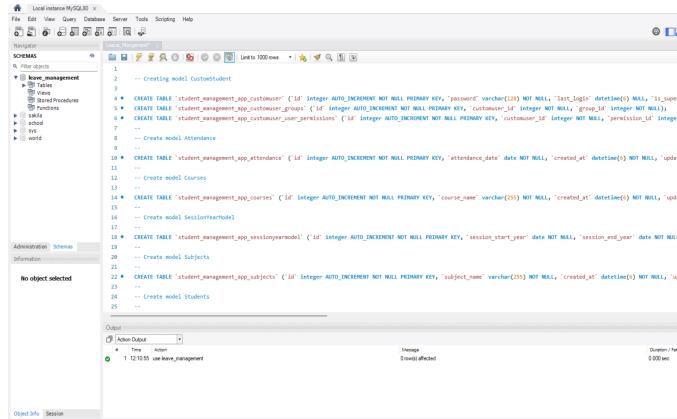
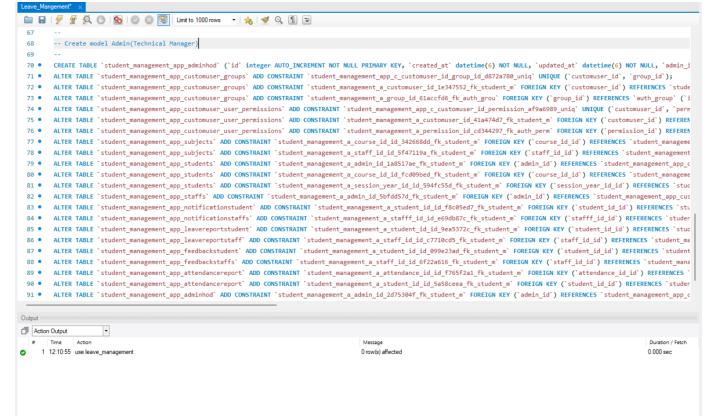


Fig. 22. MySQL Workbench

The Admins table includes the class advisors and the technical managers who verify and manage the requests. Reeth of the Admin orally store the RSA and AES encryption keystward at the same time as pretending toward keep up the integrity of the information. These tables communicate with one another effectively, making it possible for effective data, stated security and consistency to be attained, thus facilitating students and admins thwarting requests information while barring unauthorized users. We prefer MySQL over SQLite and MongoDB as it is more secure, scalable, and provides support for relational integrity. Although SQLite is lightweight and file based, it does not support high level of complex single user transactions or even multi-user access, which makes it unsuitable for our project. The project requires multiple user roles and an secure user authentication system. On the other hand, MongoDB, which is a NoSQL database, is highly recommended for unstructured data, however it lacks the ability to enforce strict relational integrity which makes it infeasible for handling leave and user authentication in a structured manner. MySQL has ACID (Atomicity, Consistency, Isolation, Durability) transactions, guaranteeing that operations performed on the information is reliable and consist-

tent. Moreover, with strong indexing and query optimization together with supporting ACID transactions, allows MySQL to quickly and accurately retrieve information regarding students and their leave history records. In pursuing the elimination of



redundancy, maintenance of atomic values, and avoidance of anomalies, we normalized the database to the Third Normal Form (3NF). First, we ensured that each table only contains relevant attributes (1NF). Then, we eliminated the partial dependencies of leave types and students by grouping them into different tables (2NF). Finally, we make sure that all the non-key attributes are dependent solely on the primary key to get rid of transitive dependencies (3NF). A case in point involves not keeping student information directly in the LeaveRequests table which would cause redundancy. Instead, we only keep the StudentID which serves as a pointer to the Students table. This, as a result, minimizes storage expenses, averts insert, update, and delete anomalies, while providing data integrity.

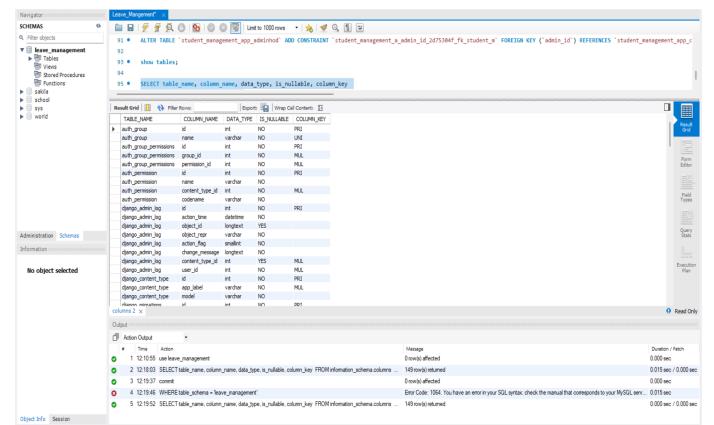


Fig. 24. Data Base Schema

VI. TESTING

A. Unit Testing

In software development, unit testing involves checking individual parts of a program, also referred to as components and functions, in isolation. This process is usually automated and

concentrated on a particular portion of the code like a method, function, or class, independently from external factors. Unit testing is significant as it allows software developers to locate and fix defects during the initial stages of development. This makes it possible for the developer to make considerable changes to the code without harming previously working code.

```
-- Running Unit Tests --
test session starts
[...]
test session ends
8 passed, 0 failed, 0 errors, 0 skips
Time: 0.000s
```

Fig. 25. Unit Testing

B. Integration Testing

As its name suggests, integration testing checks the different parts of the system to see how well they work together after being integrated. Its goal is to check the interaction between different modules, ensuring that there is data transfer between them and that it occurs seamlessly. Integration testing plays an important role in checking if there are problems with data communication, format mismatch, or improper interaction with APIs that have not been revealed in unit tests. Integration testing is flexible and can be done in steps, for example, in a successive series of top down and bottom-up integration tests, or in sandwich testing.

C. Mutation Testing

Mutation testing determines how effective a test is by adding challenges to it and seeing if the test can overcome the challenge. In the context of software inferring from its name, a test is considered a failure if during execution a fault is triggered, then the system performs a function and a test is performed to cross-check if it can identify the defect that has either been camouflaged or hidden from it. If the answer is yes, then the test suite is said to be strong. Anything else means there are gaps in the test. Mutation testing allows improving reliability by clearly identifying and filling in gaps.

D. Blackbox Testing

In this kind of testing, functionality is the main focus rather than the logic behind the code. In black box testing, the tester doesn't need to understand how the program or application operates on the inside. Instead the tester only needs to understand how the program interface looks like and what details are necessitated. Blackbox testing includes how users

interact with the system; how the system interacts with other systems; and basic system functionalities. Other techniques of the black box are equivalence partitioning, boundary value, analysis and decision table.



Fig. 26. BlackBox Testing

VII. DISCUSSION

Many college leave management portals like Amrita Vishwa Vidyapeetham's are devoid of features that enhance usability and automation. For example, in Amrita's case, students do not get email notifications. Students have to check the portal for leave status updates. Besides, when students who are allowed to take four casual leaves wish to take more, the system should auto-default to an emergency leave. Instead, students have to go to the warden's office and get emergency manual approval. Also, there are no options for document uploads which are necessary for verification of medical leaves. Our portal bridges these gaps through real-time email notifications on leave and OTP verification for added security. We also classify On Duty (OD) leaves as short term (like attending hackathons) and long-term (like internships) giving more options. Our system changes excess casual leaves to emergency ones automatically and also implements document upload requirements for ease with no manual work needed. In addition, a Technical Manager oversees system performance and addresses issues that arise between students and administrators ensuring that the system performs better, is more secure, and more user friendly than other portals of the university.

VIII. FURTHER WORK

The portal can be improved greatly in terms of efficacy, security, and user experience. We aim to implement AI-based suggestions for leave approvals where the system passes previously approved or denied leave requests and automatically recommends actions to the admin. In addition, students will be able to see their leave history and upcoming leaves in a simplified manner through calendar integration. For security purposes, we plan on adding biometric authentication along with OTP verification to offer greater user validation. To simplify the approval process even further, we can deploy a

chatbot to aid students in submitting requests and resolving queries. With a mobile app version, access to the portal on the go will be seamless, minimizing the reliance on the web. In addition, utilizing blockchain technology can prevent unauthorized modifications of leave records which enhances data integrity. To aid in decision making, having advanced analytics and reports will give admins insights into leave trends which helps in policy-making and resource allocation. This will make the system turn from being student oriented to faculty and staff enabling them to apply for leaves and check their approvals thereby making it a responsive solution for universities and institutions.

IX. CONCLUSION

The Management Student Leave System (MSLS) improves the efficiency and clarity of the leave management within institutions of learning. Students can now apply for leave from any location and administrators can conveniently manage requests due to the digitization of the leave processes and the existence of a centralized platform. OTP based authentication and document email notifications improves the security and structure of the approval workflow, reducing administrative burdens while eliminating paperwork.[1] These systems also allow for reduced burden and replication of administrative work. Repetitive tasks that require a high level of undertaking are simplified to enhance personal efficiency and time managing Whatever could be considered an inconvenience in a modern world is easily fixed with automation. Traditional leave management systems lack cohesion and responsibility and this often leaves weak points for obstructions in checking attendance and managing records. The previously described challenges have been resolved through the systematic intervention that includes verification of data processing with 'leaves' records with automation. When there is need for submission of such leave records, they will always be accurate. Need for cooperation also eliminates the scope for misuse by setting preconditions that are likely to fail without utmost parental input of granting an emergency parental leave or upload of medical documents to grant medical leave. The holistic approach of addressing educational ethics is problematic without factoring in the role of school calendars where students are disadvantaged. With the implemented calendar holidays, bulk On-Duty Leave application as well as attendance combined with system analytics, there is easier access for more appropriate decisions to be made concerning attending courses in which students are allowed to learn over simply attending classesIn general, the Student Leave Management System enhances the leave approval process, providing equity, safety, and efficiency for students, faculty, and administrative personnel. With the elimination of most manual errors, paperwork, and system accessibility, this framework lays the groundwork for more advanced student leave tracking automation. Further development may implement AI analytics, mobile application integration, and cross-institutional expansion, improving the leave management convenience while accommodating the changing advancement of educational institutions' requirements.

REFERENCES

- [1] Samuel Aládé, Adejumo Samuel, and Temitope Alade. "Design and Implementation of a Web Based Leave Management System". In: *International Journal of Computer Applications Technology and Research* 11 (Apr. 2022), pp. 123–144. DOI: 10.7753/IJCCTR1104.1006.
- [2] R Haumshini, Sathia Dev, and R Mahendran. "Digitalized Hostel Leave Management System". In: *International Journal of Emerging Technology and Innovative Engineering* 6.01 (2020).
- [3] M Iswarya, C Saikarthick, and K Sudarshan. "Leave Management portal". In: *2022 1st International Conference on Computational Science and Technology (ICCST)*. IEEE. 2022, pp. 211–214.
- [4] Hiyam Hatem Jabbar and Raed Majeed. "Electronic Leave Documentation System for College". In: *Sumer Journal for Pure Science* (2024).
- [5] Sangram D. Jadhav and Amarsinha A. Ranaware. "Design Steps of Online Leave Management Application System for Academic Institution". In: *International Journal of Innovations in Engineering Research and Technology* 2023.NCETSAE - July (2023), pp. 82–86. URL: <https://repo.ijiert.org/index.php/ijiert/article/view/3718>.
- [6] Vikrant Kumar Kaushik et al. "Student leave management system". In: *Int. J. Adv. Res. Inno. Ideas Educ* 3 (2017), pp. 124–131.
- [7] Ahmad Munawir et al. "Enhancing Leave Management through an Online Employee Leave Application System". In: *ARRUS Journal of Engineering and Technology* 4.1 (2024), pp. 26–36.
- [8] T. Praveen, H. M. Vetrivel, and B. Arumugasamy. "Online Leave Management System". In: *International Journal of Advanced Science and Technology* 29.7 (2020), pp. 2785–2791.
- [9] M Ramanan. "Web Based Leave Management System for University College of Jaffna". In: *J. Res. Technol. Eng* 2.3 (2021), pp. 106–113.
- [10] BU Stephen et al. "Improved leave scheduling algorithm for improved service delivery in the Nigerian university system". In: *Nigerian Journal of Technology* 37.4 (2018), pp. 1039–1048.