APPLIED INFORMATION TECHNOLOGY AND COMPUTER SCIENCE
e-ISSN: 2773-5141

AITCS

Vol. 5 No. 2 (2024) 500-519
https://publisher.uthm.edu.my/periodicals/index.php/aitcs

Tan Star HR Management System

Yubaasini Visvanathan[1], Nurul Aswa Omar[2]

[1,2] *Fakulti Sains Komputer dan Teknologi Maklumat,*
Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, 86400, Malaysia

*Corresponding Author: vyuba24@gmail.com
DOI: https://doi.org/10.30880/aitcs.2024.05.02.027

Article Info

Received: 13 June 2024
Accepted: 28 September 2024
Available online: 15 December 2024

Keywords

HR management, web-based system, salary management, waterfall methodology, HR processes, automated payroll, Project phases

Abstract

The Tan Star HR Management System (TS-HRMS) is a web-based system designed to revolutionize human data management in organizations. The system aims to address issues such as time-consuming data retrieval, errors, and potential employee misidentification. The system includes features like transparent access to salary history, streamlined profile maintenance, quick processing of leave requests, and user-friendly dashboards. The project uses advanced technology which is Hypertext Preprocessor (PHP) for programming language and XAMPP Control Panel for database where by using a methodical approach with waterfall design phases and rigorous testing to ensure a reliable and error-free system. TS-HRMS accommodates the unique needs of various organizations through scalability and customization, promoting improved operational agility and organizational performance. The project is expected to change HR management, bringing about a new era of accessible, accurate, and efficient personnel data management.

1. Introduction

Pasar Mini Tan Star Enterprise, a 24-hour grocery store where staff and employee at Pasar Mini Tan Star are now applying for leaves, maintaining employee records manually using a form, and the company is dealing with a data loss situation in which employee records are lost. The system has transformed the store's operations, reducing manual labor and costs, improving employee satisfaction, and ensuring data security and compliance. The web-based system simplifies leave management, allowing employees to request leave and approve requests swiftly. It also automates payroll calculations, saving time and reducing errors. The system also offers robust reporting capabilities, enabling data-driven decisions for management and staffing. The user-friendly interface requires minimal training for staff, minimizing disruptions to daily operations. As a result, Pasar Mini Tan Star Enterprise has become an integral part of the community's daily life, providing essential products efficiently and effectively.

1.1 Problem Statement

Tan Star HR faces challenges in planning, processing, keeping records, and retrieving information, which can lead to delays and reduced productivity. To improve efficiency and operational effectiveness, the organization should address several critical concerns.

The lengthy data retrieval process is a major issue, requiring human labor and time. A streamlined system for data retrieval can help provide quick access to employees' information. Data inaccuracy is another issue, impacting decision-making and operational operations. A strong system for data validation and verification, including frequent audits and staff self-service capabilities, can improve data trustworthiness and ensure correct personnel records.

An effective payroll system is also crucial for employee happiness, as accurate and timely wage disbursement is crucial for employee satisfaction. An efficient payroll administration system can ensure accurate payments while providing transparency and flexibility.

Misidentification of employees poses a serious risk to the company, causing identity confusion and potential operational interruptions. A strict system for personnel identification and verification, including biometric authentication and unique identifiers, can help prevent misidentification and ensure operational safety.

To address these challenges, a Human Resource Management System should be developed to facilitate effective data access, ensure data correctness, automate payroll processes, and protect against employee misidentification.

1.2 Objective

The objectives of developing the Tan Star HR Management System (TS-HRMS) are as follows:

i. To design the Tan Star HR Management System (TS-HRMS) based on structured approach.
ii. To develop a web-based management system by using sublime IDE, PHP technology and MySQL as back-end.
iii. To test functionality of system and user acceptance.

1.3 Scope of Project

An extensive web-based system created to optimize and simplify an organization's human resources processes is called the Tan Star HR Management System. This project encompasses the following key features and functionalities. Table 1 and 2 shows the functionalities of system that is designated for admin and employee.

Table 1 Designated functionalities for Admin explained by modules

Admin: - Admin is responsible for managing the entire system. Key features for the admin are as follows:	
Dashboard	: Displaying essential metrics and system status
Manage department list	: Add, edit or delete department
Manage designation list	: Add, edit, or delete designations
Manage list of leave types	: Add, edit, or delete leave types
Manage employee list	: Add, edit or delete information of the employee
Manage employee salary	: Add, edit, or delete employee salary details
Manage leave privilege	: Manage employee leave privileges
Manage leave application	: Manage leave requests and update leave requests
Manage leave application's status	: Update the status of leave applications
Print leave records	: Print leave records of employees
Generate leave application record	: Generate reports on leave applications
Manage system setting	: Manage overall system settings
Notification	: Notify the admin when a leave application is submitted
Manage account credentials	: Admin can also view profiles, change passwords, and recover passwords

Table 2 Designated functionalities for Employee explained by modules

Employee: - Employee is responsible for managing the designated roles. Key features for the employee are as follows:	
Dashboard	: A central hub displaying essential metrics and system status
View salary records	: View history of salary
View leave records	: View leave records
Create leave application	: Can apply for leave
Manage leave application	: Manage leave applications, including viewing and updating leave requests

Notification	: Notify the employee when the status of their leave application is updated
Manage account credentials	: Employee can also view profile and change password

2. Related Work

Related work discusses the current understanding of a computerized system and compares it with the proposed system through a literature review. The review aims to gather data and identify issues, while also serving as the project's source material. The literature review helps understand linked initiatives and solve important issues. TCIS Total Campus Integrated System, E-Penyata Gaji & Laporan (Jabatan Akauntan Negara Malaysia), and HRMIS Sistem Pengurusan Maklumat Sumber Manusia are the systems that are compared I this section.

2.1 Study of the Existing Human Resource Management System

To find flaws in the current system or learn more about this project that has been produced with more precision and clarity, this research of the current system is being conducted. Three systems that are almost comparable to the system that will be constructed are used as benchmarks in this study. The current systems are the HRMIS Sistem Pengurusan Maklumat Sumber Manusia, TCIS Total Campus Integrated System, and E-Penyata Gaji & Laporan (Jabatan Akauntan Negara Malaysia).

2.1.1 HRMiS Sistem Pengurusan Maklumat Sumber Manusia

HRMiS 2.0 is a significant advancement in human resource management information systems, aiming to revolutionize employee data handling and streamline administrative procedures. Despite persistent user complaints, the system's flexibility has expanded, allowing employees to handle leave requests from anywhere, despite being limited to a single network.

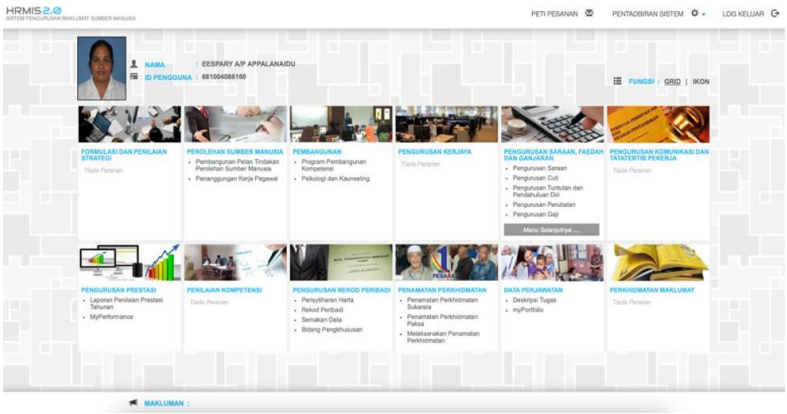


Fig. 1 *Homepage for HRMIS*

2.1.2 TCIS Total Campus Integrated System

TCSI is a flexible software program designed to integrate academic and administrative functions in educational institutions. It combines modules for admissions, financial aid, academic records, student information, and human resources. TCSI improves productivity, reduces duplication, and creates a unified environment for teachers, administrators, and students. It streamlines procedures, maximizes resource use, and enhances the quality of education by reducing mistakes and adapting to technology developments.

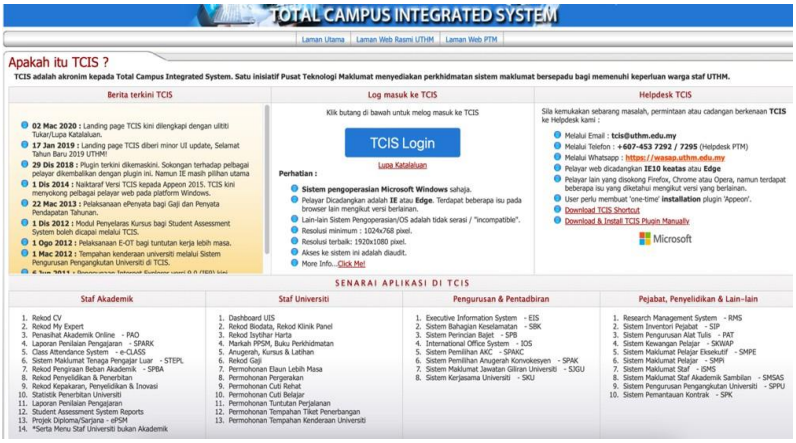


Fig. 2 *Homepage for TCIS*

2.1.3 E-Penyata Gaji & Laporan (Jabatan Akauntan Negara Malaysia)

E-Penyata Gaji & Laporan (Jabatan Akauntan Negara Malaysia) is a government salary management system that manages employee information and compensation. It includes payroll processing, which calculates monthly pay, benefits, and expenses, and other parts that handle insurance, health benefits, and deductions. The system ensures compliance with regulations and connects to accounting systems, allowing workers to control their work environment.

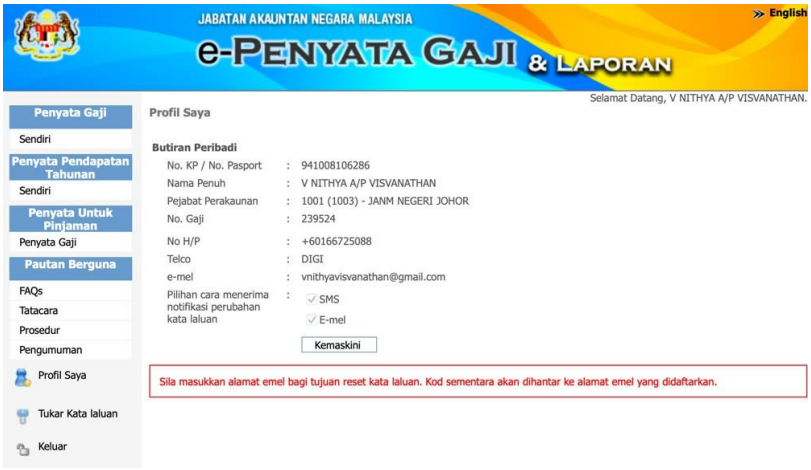


Fig. 3 *Homepage for e-Penyata Gaji & Laporan*

Comparison between related system:

Table below shows the module and system comparison between the existing system and the system that will be developed for Tan Star Enterprise

Table 3 *Comparison between the proposed system and existing similar systems in terms of functionalities*

Comparison / System	**HRMIS**	**TCIS**	**E-Penyata Gaji & Laporan**	**Proposed System (TS-HRMS)**
Technology:				
Web-based system	Yes, web based	Yes, web based	Yes, web based	Yes, web based
Web-based app	Yes, web based app	No	No	No
Online system	Yes, online system	Yes, , online system	Yes, , online system	Yes, , online system
Modules:				
Attendance	No	Yes, student attendance	No	No

Employee Data	Yes Update Details)	Yes Update Details)	Yes Update Details)	Yes (Update Details)
Department	Yes	Yes	No	Yes
Payroll	No	Yes, generate pay slip	Yes, generate pay slip	Yes, generate pay slip
Leave	Yes (Apply Leave)	Yes (Apply Leave)	No	Yes (Apply Leave)
Claim	Yes	Yes	No	No
Notification	Yes (Leave Approval)	No	No	Yes (Employee Leave Application)
Performance	Yes (Monthly)	Yes	No	Yes
Report	Yes (Employee details)	Yes (Salary slips)	Yes (Salary slips)	Yes (Employee details)
Platform Language	Java, PHP, C#, Python, JavaScript	Java, Python, JScript, C#,	Java, Python, JavaScript, C#, PHP	PHP, HTML, CSS, JScript, SQL

3. Methodology

Figure 4 shows the Waterfall Process Model where a methodology used in Human Resource Management Systems for projects with clearly defined stages and requirements. It comprises five stages: requirement analysis, design, implementation, testing, and maintenance. The model is connected, with outcomes influencing each other. It allows for departmentalization and administrative control, ensuring project completion on schedule. The waterfall approach is particularly beneficial for software development.

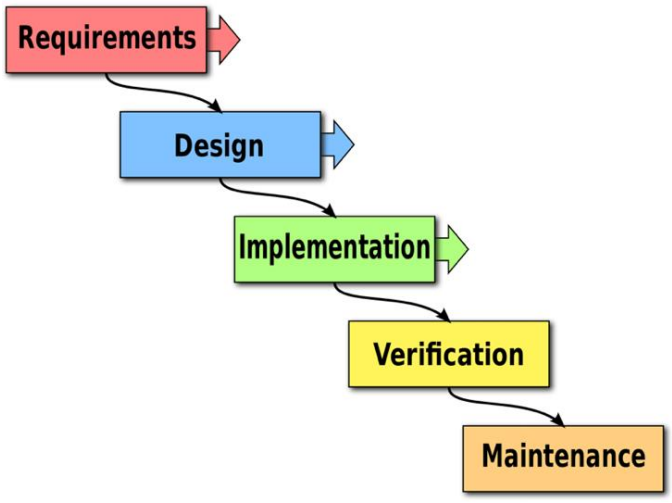


Fig. 4 *Waterfall Development Model*

Table 4 shows that the tasks completed in each step utilizing the waterfall model methodologies are explained. The requirement and analysis phase, design phase, implementation phase, testing phase, and maintenance phase are the five stages of this process., durations, and the overall project timeline.

Applied Information Technology and Computer Science Vol. 5 No. 2 (2024) p. 500-519

Table 4 *Specification of activities to be carried out in each phase with its deliverables and the tools it used*

Phases	Task/Activities	Deliverables	Tools
Requirement & Analysis	- Identity system scope and requirements - Analyze requirements: i. Data collection – interview Pasar Mini Tan Star's supervisor to identify system requirements - Project timeline – Gantt Chart	i. Proposal ii. Gantt Chart	i. MS Word ii. TeamGantt.com
Design	- Design user interface as per requirement - Design database as per requirement - Design Data Flow Diagram (DFD) and Entity Relationship Diagram (ERD)	i. Context Diagram (CD) ii. Data Flow Diagram (DFD) Entity Relationship Diagram (ERD) iv. Schema Table iii. Interfaces	i. Creately.com ii. PowerPoint
Implementation	The designated design will be converted to actual code	To execute system	i. Sublime IDE ii. XAMPP Control Panel
System Testing	Alpha testing	Test the functionalities and acceptance test	i. Sublime IDE ii. XAMPP Control Panel
Maintenance	Involve correcting errors and upgrading due to current trends and needs	Make sure the system is ready to use by the stakeholder and ensures the quality of use which provides effectiveness, efficiency, satisfaction, reliable and secure	i. Sublime IDE ii. XAMPP Control Panel

4. Analysis and Design

This chapter covers the analysis and design phase of a system's development, which is completed before the implementation phase. It discusses a comprehensive system specification procedure, model definition, data flow diagrams (DFD), sketches, and architecture. The system's operation is demonstrated by its architecture, which includes interface and database design, reflecting the data stored in the system.

4.1 DFD Context Diagram (DFD CD)

Context diagrams show the surroundings of a software system. The Human Resource Management System's context diagram is displayed in Figure 5 below. The system is interacted with by two external entities, namely employees and administrators.

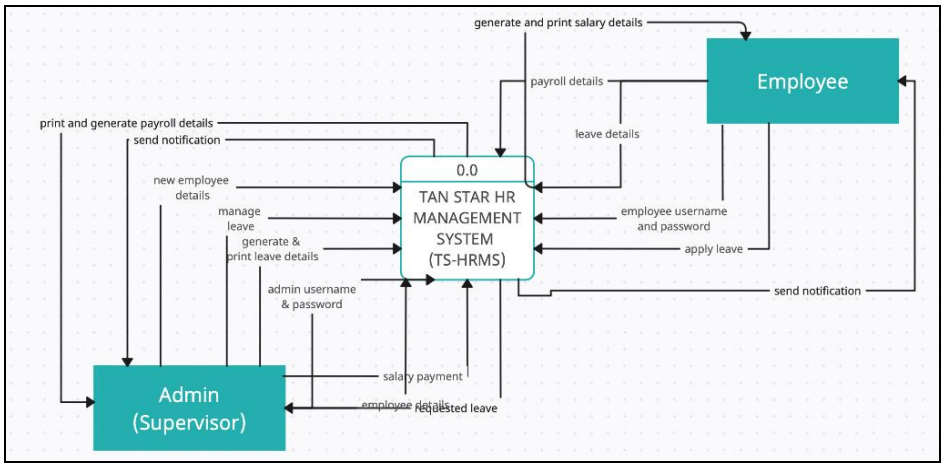


Fig. 5 *Data Flow Diagram Context Diagram (DFD CD)*

4.2 DFD Level 0

The Tan Star HR Management System's level 0 data flow diagram may be seen in Figure 6. The data flow involved in figure 4 is described in the Data flow Diagram Level 0. The relationships between each process and the entities at level 0 of the data flow diagram are depicted below the figure.

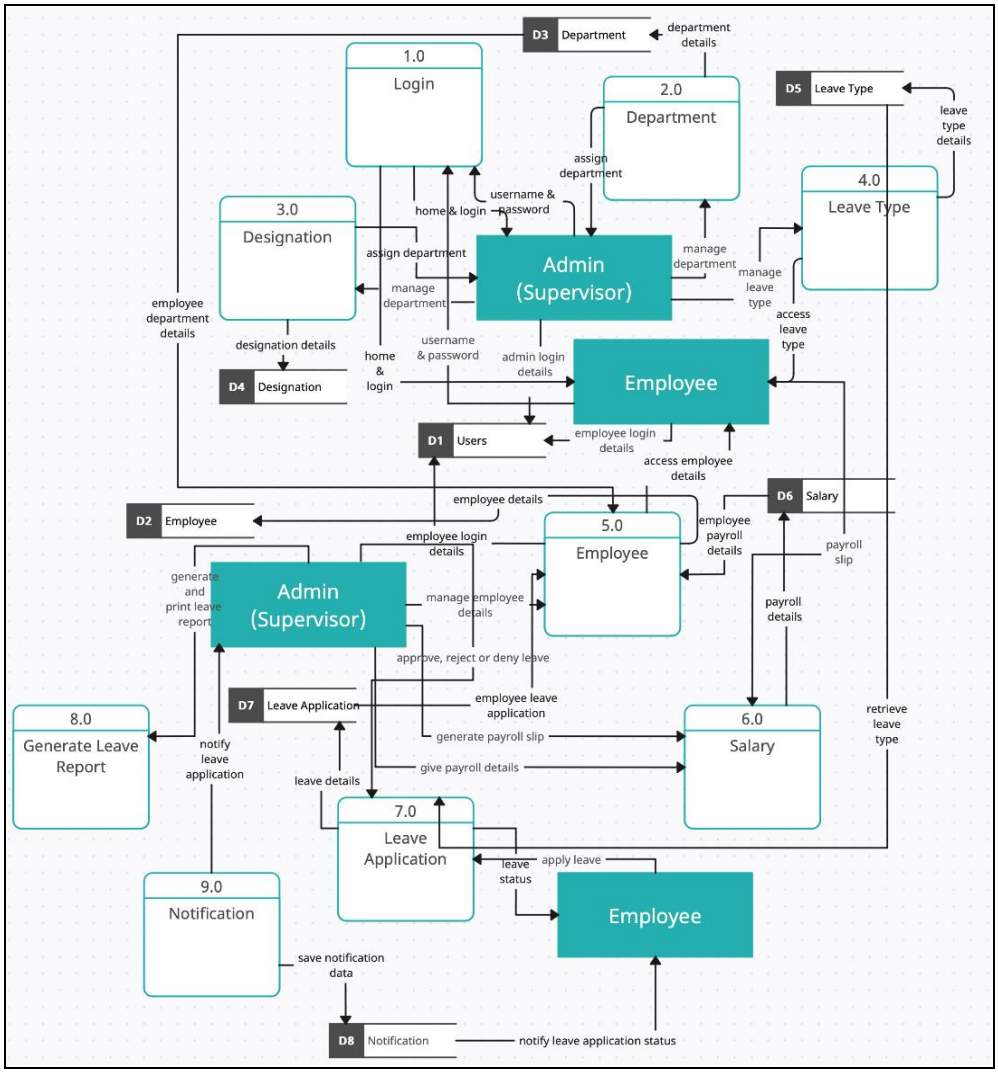


Fig. 6 *Data Flow Diagram Level 0 (DFD 0)*

4.3 Entity Relationship Diagram (ERD)

An entity-attribute relationship diagram is a comprehensive diagram that illustrates the relationship between the two. An entity relationship diagram is created to show how external entities and the system are related. A high-level conceptual design database uses the ER-model to identify important things and provide links that explain the relationships between these elements. Based on the chosen data model, the ER model is transformed into a particular database scheme during database design. Gordon Everest first introduced the Crow's Foot notation, which was formerly known as Inverted Arrow. The Entity Relationship Diagram below, which details the relationship between the HRM System and external entities including employees and administrators.

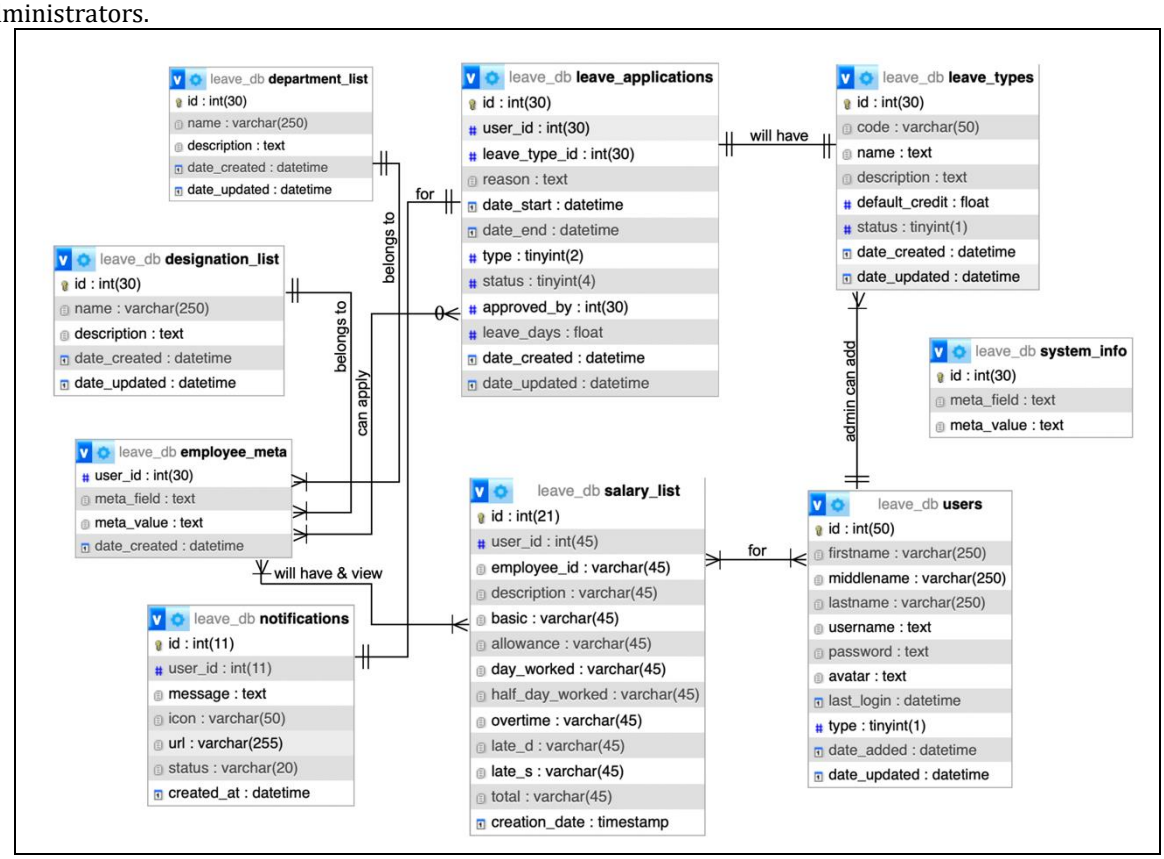


Fig.7 *Entity Relationship Diagram of TS-HRMS*

5. Implementation and Testing

This part presents a detailed analysis of the implementation of the Tan Star HR Management System, focusing on its effectiveness, efficiency, and impact on the organization's operations. The analysis includes a detailed examination of key performance indicators, user feedback, and system performance metrics. The aim is to understand the system's strengths and weaknesses, identify areas for improvement, and assess its overall contribution to enhancing HR processes. The findings are contextualized within the broader HR management technology landscape, providing valuable implications for practitioners and researchers in the field.

5.1 System Implementation

The web application system, which was developed using PHP and PDO for database administration and HTML, CSS, and JavaScript for design and style, will be described in this part. Sample data will be utilized to demonstrate the coding and implementation of key aspects in the E-form system, as well as its flow. Nevertheless, just certain portions of the main program for every function will be explained.

5.1.1 Database Connection

The Tan Star HR Management System uses a single file called DBConnection.php to simplify file management. This file contains the database name of the system and is necessary for any files requiring database connectivity to function correctly. The code for creating the connection is provided.

```
class DBConnection{

    private $host = DB_SERVER;
    private $username = DB_USERNAME;
    private $password = DB_PASSWORD;
    private $database = DB_NAME;
    public $conn;
    public function __construct(){
        if (!isset($this->conn)) {
$this->conn = new mysqli($this->host, $this->username, $this->password,
                            $this->database);
            if (!$this->conn) {
                echo 'Cannot connect to database server';
                exit; }}}
    public function __destruct(){
        $this->conn->close();
    }
```

5.1.2 Login Management

The login module restricts authorized users' access to specific pages, distinguishing between administrators, workers, and managers. Users must enter passwords and usernames, with the module's interface and partial code determining user type.

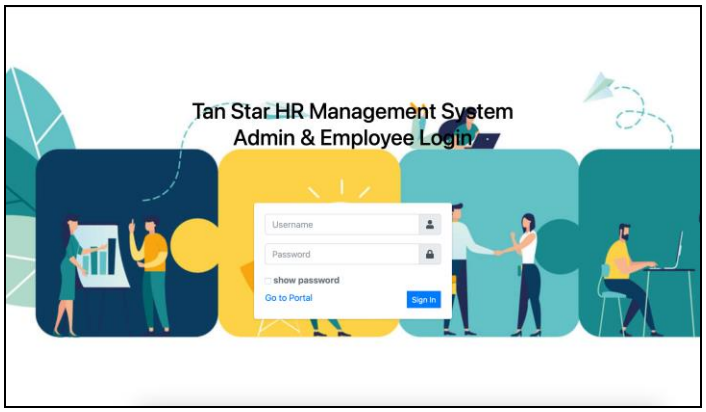


Fig. 8 *Login page for Admin/Employee*

5.1.3 Department Management

The department management module streamlines administrative tasks by creating, modifying, and eliminating departments, as well as assigning staff, while providing quick updates and user-friendly interfaces. It ensures efficient use of resources and maintains accuracy.

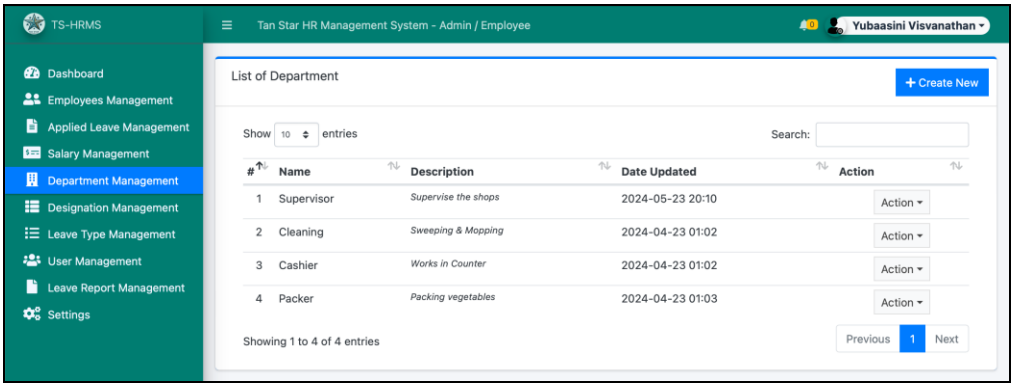


Fig. 9 *Department Management Admin*

5.1.4 Designation Management

The designation administration module is crucial for managing duties like adding, modifying, and eliminating outdated designations, ensuring efficient task allocation. Its user-friendly interfaces and effective data handling make it easy for managers to manage designations, maintaining data integrity.

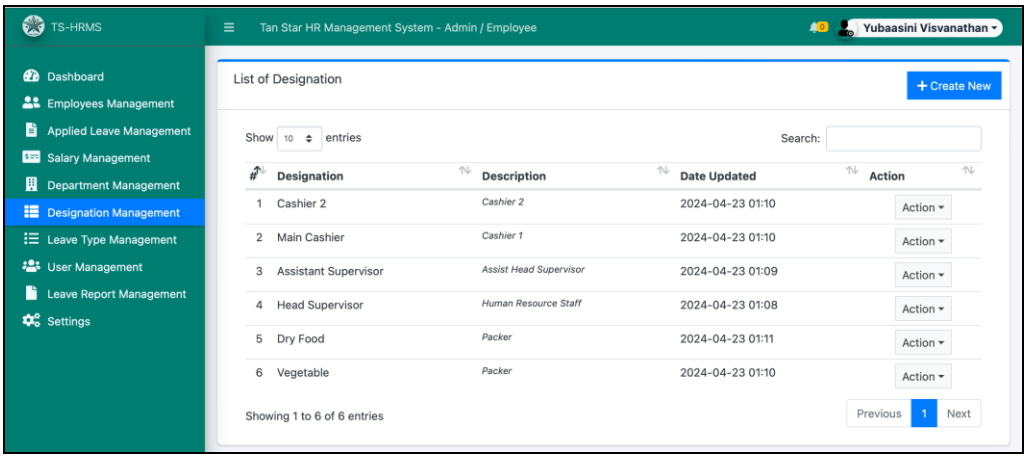


Fig. 10 *Designation Management Admin*

5.1.5 Leave Type Management

The module manages leave types, including sick leave and yearly holidays, ensuring correct data entry and neat arrangement. It eliminates duplication and provides real-time changes, ensuring seamless operations and clear feedback. Code snippets guide adding, updating, and deleting leave types.

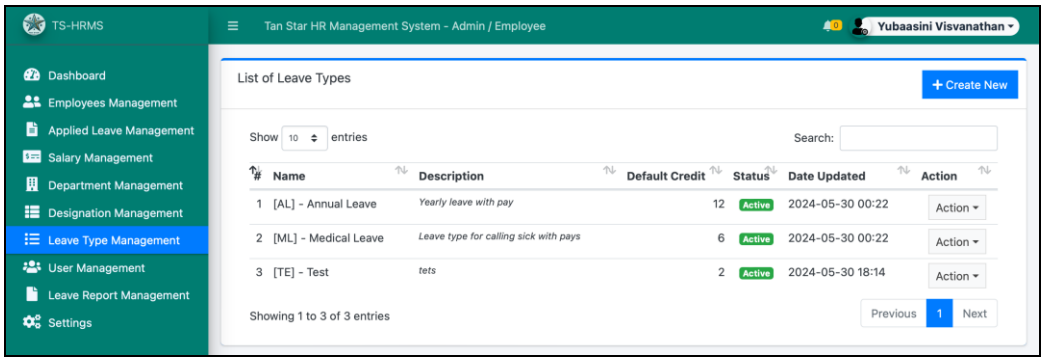


Fig. 11 *Leave Type Management Admin*

5.1.6 Employee Management

The system's staff management features, such as `save_employee()' help maintain data integrity, streamline user administration, and securely change employee passwords based on their ID, ensuring successful workforce management.

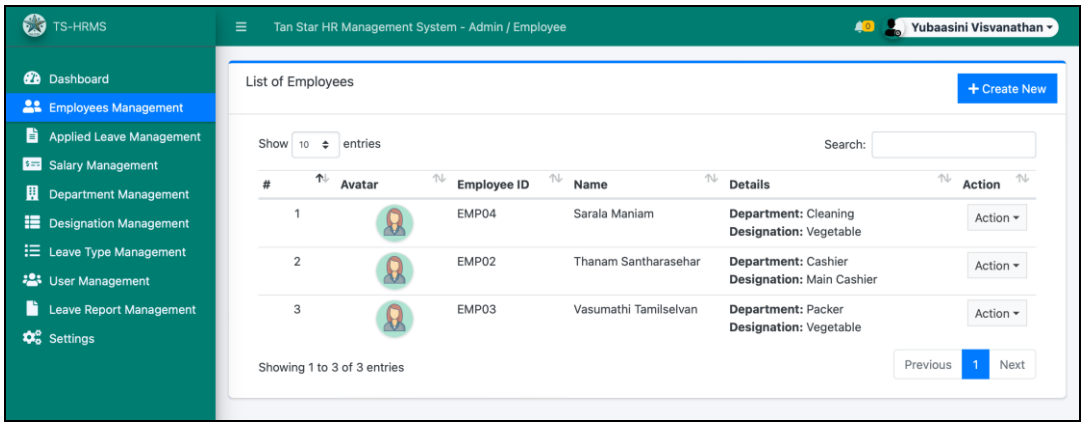


Fig. 12 *Employee Management Admin*

5.1.7 Applied Leave Management

The `save_application()` function is crucial in our leave management system, reviewing and approving staff leave requests, ensuring they align with allotted credits and timetables. It also manages new requests, updates existing ones, and assesses the operation's success.

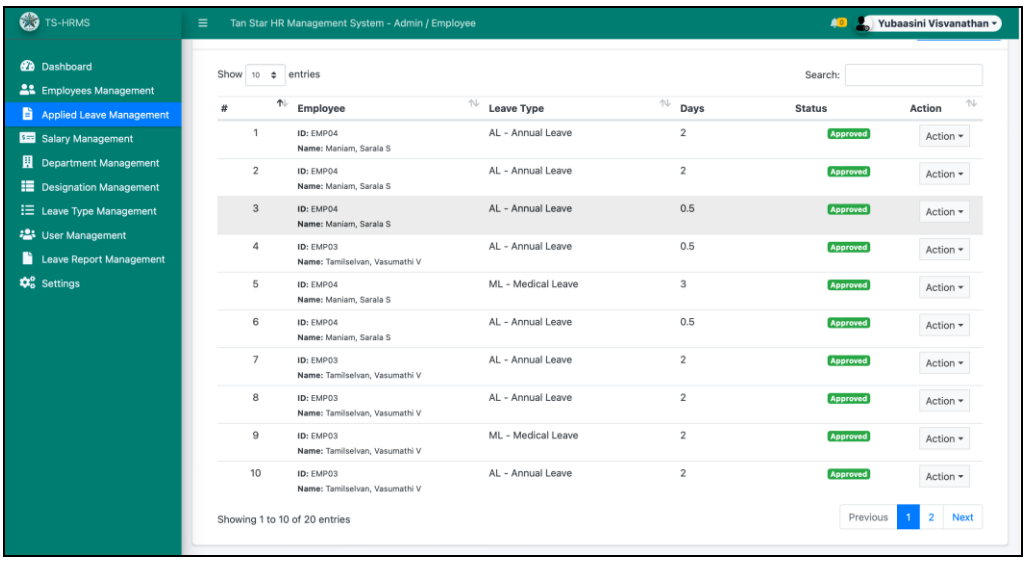


Fig. 13 *Applied Leave Management Admin*

5.1.8 Salary Management

The `save_salary()` method is crucial for our system's salary administration, processing and validating salary data. It retrieves user data like department and designation to calculate total compensation.

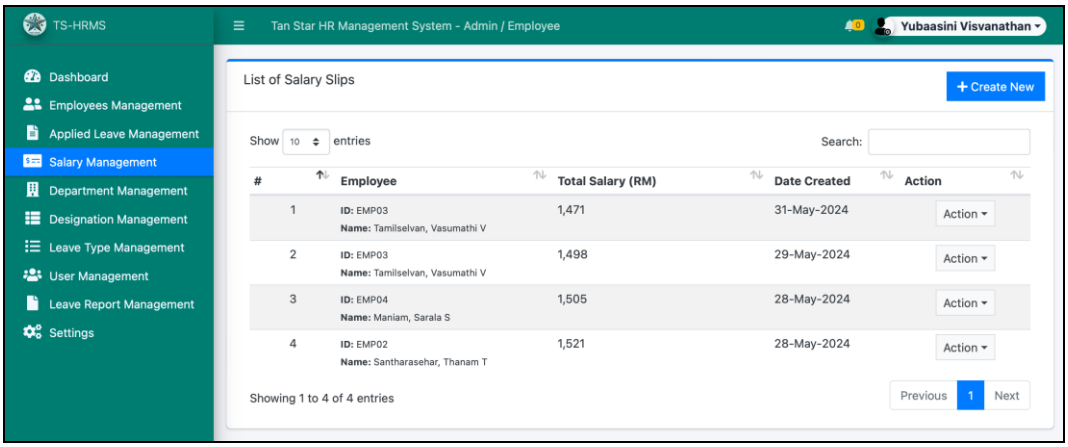


Fig. 14 *Salary Management Admin*

5.1.9 User Management

This PHP code has two main functions: `save_users` and `delete_users`. It handles password hashing, file upload management, and user information adding. It removes user records and avatar files from the database using SQL queries and session data.

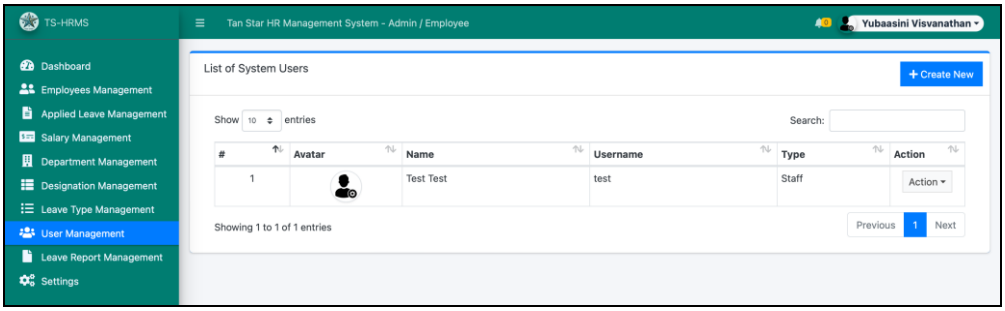


Fig. 15 *User Management Admin*

5.1.10 Leave Report Management

The code snippet displays a list of leave applications from a database, filtering records based on start and end dates. The table includes employee ID, leave type, dates, status, and reason. Colored badges graphically show leave status, and "No Records" is displayed if no records are found.

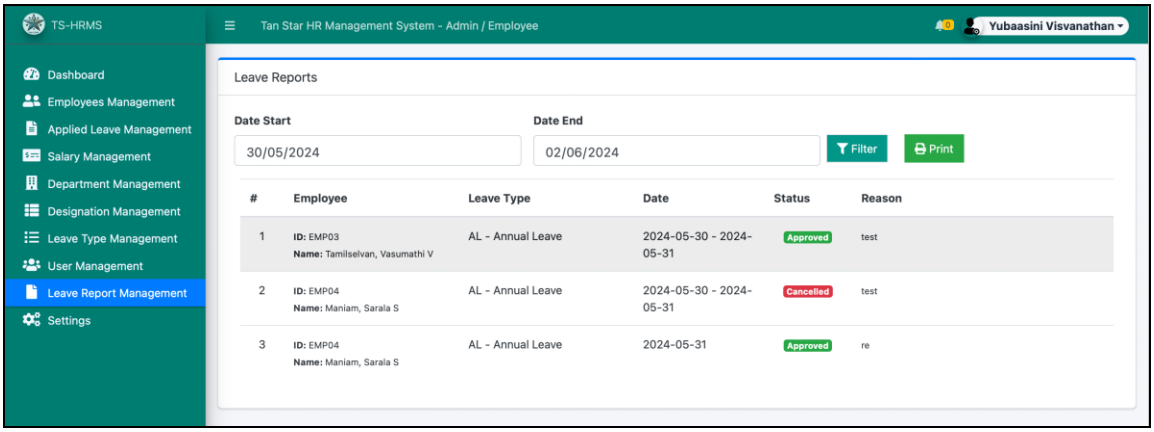


Fig. 16 *Leave Report Management Admin*

5.2 System Testing

System testing is a crucial stage in the software development lifecycle, assessing the entire integrated system's compliance with specified requirements. It evaluates the system's functionality, performance, reliability, and security in real-world conditions, ensuring all components interact correctly and identifying potential issues.

This comprehensive testing enhances product quality and customer satisfaction by ensuring robustness and readiness for deployment.

5.2.1 System Functionality Testing

System functionality testing is a crucial phase in the development and acceptance of a system, ensuring it meets all requirements and design specifications. Independent testers evaluate the system's overall functionality, including interfaces between modules. Test cases are created to verify the features of the Tan Star HR Management System, assessing components from department administration to leave application procedures and pay check generation. This methodical methodology ensures a solid and reliable HR management system.

Table 5 *List of Test Cases*

No.	Test Cases	Description
TC_100 Login Management		
1.	TC_100_01	Verify that the user can log in with valid credentials.
2.	TC_100_02	Verify that the system rejects invalid login attempts.
3.	TC_100_03	Verify that the system locks the account after multiple failed login attempts.
4.	TC_100_04	Verify that the password reset functionality works correctly.
5.	TC_100_05	Verify that the system maintains session security
TC_200 Department Management		
6.	TC_200_06	Verify that the admin can create new departments.
7.	TC_200_07	Verify that the admin can update department details.
8.	TC_200_08	Verify that the admin can delete a department.
9.	TC_200_09	Verify that the admin can view the list of all departments.
10.	TC_200_10	Verify that department details are validated before saving.
TC_300 Designation Management		
11.	TC_300_11	Verify that the admin can create new designations.
12.	TC_300_12	Verify that the admin can update designation details.
13.	TC_300_13	Verify that the admin can delete a designation.
14.	TC_300_14	Verify that the admin can view the list of all designations.
15.	TC_300_15	Verify that designation details are validated before saving.
TC_400 Leave Types Management		
16.	TC_400_16	Verify that the admin can create new leave types.
17.	TC_400_17	Verify that the admin can update leave type details.
18.	TC_400_18	Verify that the admin can delete a leave type.
19.	TC_400_19	Verify that the admin can view the list of all leave types.
20.	TC_400_20	Verify that leave type details are validated before saving.
TC_500 Employee Management		
21.	TC_500_21	Verify that the admin can add a new employee.
22.	TC_500_22	Verify that the admin can update employee details.
23.	TC_500_23	Verify that the admin can delete an employee.
24.	TC_500_24	Verify that the admin can view the list of all employees.

25.	TC_500_25	Verify that employee details are validated before saving.
TC_600 Applied Leave Management		
26.	TC_600_26	Verify that an employee can apply for leave.
27.	TC_600_27	Verify that the admin can approve leave applications.
28.	TC_600_28	Verify that the admin can reject or deny leave applications.
29.	TC_600_29	Verify that an employee can view the status of their leave applications.
30.	TC_600_30	Verify that leave application details are validated before submission.
TC_700 Salary Management		
31.	TC_700_31	Verify that the admin can generate payroll slips.
32.	TC_700_32	Verify that employees can view their payroll details.
33.	TC_700_33	Verify that payroll details are accurate.
34.	TC_700_34	Verify that the system calculates payroll accurately.
35.	TC_700_35	Verify that the admin can update payroll details.
TC_800 User Management		
36.	TC_800_36	Verify that the admin can create new user accounts.
37.	TC_800_37	Verify that the admin can update user details.
38.	TC_800_38	Verify that the admin can delete a user account.
39.	TC_800_39	Verify that the admin can assign roles to users.
40.	TC_800_40	Verify that the admin can view the list of all users.
TC_900 Leave Report Management		
41.	TC_900_41	Verify that the admin can generate leave reports.
42.	TC_900_42	Verify that the admin can filter leave reports by date range.
43.	TC_900_43	Verify that the admin can export leave reports.
44.	TC_900_44	Verify that leave report details are accurate.

5.2.2 User Acceptance Testing

The software, TS-HRMS, undergoes user acceptance testing (UAT) before its official release. This process involves real-world user testing in a controlled environment, including a representative user group. The assessment identifies flaws, usability issues, and missing features. Feedback is then examined, and necessary adjustments made to ensure the software meets all user needs and expectations.

Table 6 *User Acceptance Testing Evaluation*

No	Statement	Scale					Total
		1	2	3	4	5	
1.	How easy was it to navigate through the Tan Star HR Management System?	0	0	0	8	7	15
2.	How easy was it to navigate through the Tan Star HR Management System?	0	0	1	4	10	15
3.	Were you able to easily apply for leave using the system?	0	0	1	6	8	15
4.	Did you find the system's interface visually appealing and well-organized?	0	0	1	4	10	15

5.	How straightforward was it to access and review your pay slips documents to print?	0	0	1	5	9	15
6.	Did the system provide adequate notifications and reminders for leave application?	0	0	0	7	8	15
7.	How satisfied are you with the response time of the system's features and functionalities?	0	0	0	5	10	15
8.	How severe were the bugs or technical issues you encountered while using the system?	0	0	0	1	14	15
9.	To what extent did you find all the expected functionalities or features in the system?	0	0	0	5	10	15
10.	Were you able to successfully log in and log out of the system without any issues?	**YES** / NO					15

6. Conclusion

In this area the system focal points, drawbacks and suggestion will be talked about. In general, Human Resource Management System for Tan Star Enterprise was meet the targets that has been built in Chapter 1. This system meets the most objective of this system is to create an accessible Human Resource employee administration system with help ought to not take a toll more than the advantage rendered, which is this system was utilitarian within the online stage. The system development stages were utilizing the model strategy, this technique comprises of arranging, examination, plan and usage. These stages comprise of exercises that intentionally to improve the system advancement prepare such as meeting and give survey, compare with existing method and alluding a few diaries and books. After the testing stage has been done by choosing the analyzer from target user, the test result will be talked about and recorded the proposal based on this result. Close to that the points of interest and drawbacks too recorded to be clarified in assist in this chapter.

6.1 Achievement of Overall Objectives

Though it offers several noteworthy benefits, the Tan Star Enterprise HRM system has drawbacks that affect how successful it is.

6.1.1 System Advantages

There are some widely recognised benefits of Tan Star Enterprise's HRM system:

i. The system updates supervisors of employee leave requests and updates employees on their leave status upon login. As a result, submitting a leave application does not require direct supervisor consultation.
ii. The system, which is overseen by HR professionals, serves as a single location for all employee data. HR representatives can quickly access and manage employee information thanks to its centralized storage.
iii. To simplify pay management, the technology automates several process processes. It can provide accurate pay calculations based on relevant information. Task automation reduces human error and ensures timely and accurate employee salary payments. In-depth pay statements and reports, which provide management and HR experts with crucial data on salary trends and costs, may also be generated by the system.

6.1.2 System Disadvantages

Like any other system, the Tan Star Enterprise HRM System has limits, which were made apparent when the system was being developed. These limitations include:

i. The system's payroll module has limitations; although it can calculate salaries and generate pay slips, it is unable to handle online salary payments. Since the technology isn't compatible with internet banking, income distribution still needs to be done manually.
ii. Another drawback is the lack of attendance management features. This means that the system does not include functionalities for tracking and managing employee attendance, such as recording work hours, breaks, and absences.

6.2 Recommendation

Based on testing that has been carried out, there are several suggestions for improvement have been identified. Among the proposals are as follows:

i. Use biometric technologies, such as fingerprint scanners, to automate the recording of staff attendance. By integrating these gadgets with the system, manual data input may be removed, expediting and saving time throughout the attendance recording process.
ii. Expand the capabilities of the system to allow online salary payments via internet banking. Employee convenience and direct bank account deposit of pay are guaranteed by this enhancement, which will expedite the salary distribution procedure.

Tan Star Enterprise's HRMS meets goals and satisfies user demands, demonstrated by its functionality and test results. Enhancements aim to improve organization. Future improvements will ensure the system meets user demands and maintains quality and performance through ongoing studies and research.

Acknowledgement

The authors express their gratitude to the Faculty of Computer Science and Information Technology at University Tun Hussein Onn Malaysia for their invaluable assistance and motivation during the study's conduct. Authors including an appendix section should do so before the References section. Multiple appendices should all have headings in the style used above. They will automatically be ordered A, B, C etc.

References

[1] ADP. (2021, January 15). Employee Management and Employee Management Systems. Retrieved from https://www.adp.com/resources/articles-and-insights/articles/e/employee-management.aspx

[2] Top features of Employee Management Software | TEC. (n.d.). Retrieved from https://www3.technologyevaluation.com/research/article/top-features-of-employee-management-software.html

[3] Pendidik2u. (2017, October 2). HRMIS Sistem Pengurusan Maklumat Sumber Manusia - Pendidik2u. Retrieved from https://www.pendidik2u.my/hrmis-login/

[4] HR.my for free. lifetime. (n.d.). Retrieved from https://hr.my/doc/employee-management.html

[5] Saha, P. (2022, December 5). Employee Management Portal - ENR Consultancy Services. Retrieved from https://engagenreap.com/employee-management-portal/

[6] Lee et al. (1999). Relations between the practical aspects of the implementation and the conceptual viewpoints of the implementation.

[7] Laursen, K. and Foss, N. J. (2003). New human resources management practices, complementarities and the impact on innovation performance. *Cambridge Journal of Economics, 27*(2), 243-263.

[8] Gupta, A. and Singhai, A. (1993). Managing human resources for innovation and creativity. *Research Technology Management, 36*(3), 8-41.

[9] Dyer, L. (1985). Strategic human resource and planning. In K. Rowland & G. Ferris (Eds.), *Research in Personnel and Human Resource Management, 18*(2), 295-320.

[10] Huselid, M. A. (1995). The impact of human resource management practices on turnover, productivity and corporate financial performance. *Academy of Management Journal, 38*(3), 635-672.

Appendix A

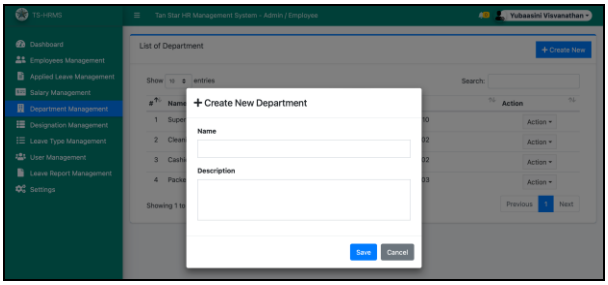
Login Management: The login module restricts authorized users' access to specific pages, distinguishing between administrators, workers, and managers. Users must enter passwords and usernames, with the module's interface and partial code determining user type.


```
<script>
  $(document).ready(function(){
    end_loader();
  })
</script>
<script>
  $(document).ready(function(){
    end_loader();
    // Show/hide password
    $('#show-password').click(function() {
      var passwordField = $('input[name="password"]');
      var passwordFieldType = passwordField.attr('type');

      if (passwordFieldType == 'password') {
        passwordField.attr('type', 'text');
      } else {
        passwordField.attr('type', 'password');
      }});});</script>
```

Fig. 26 *Homepage and partial code for Login Management*

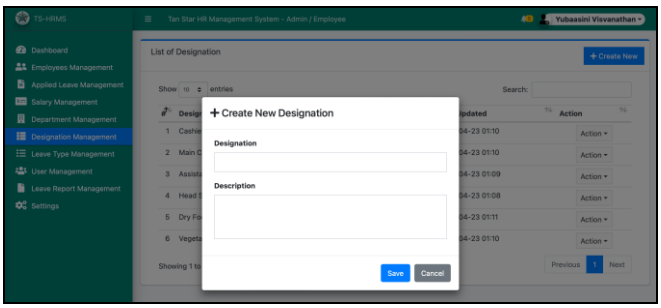
Department Management: The department management module streamlines administrative tasks by creating, modifying, and eliminating departments, as well as assigning staff, while providing quick updates and user-friendly interfaces. It ensures efficient use of resources and maintains accuracy.




```
$check = $this->conn->query("SELECT * FROM `department_list` where
`name` = '{$name}' ".(!empty($id) ? " and id != {$id} " : "")." )-
>num_rows; if($this->capture_err())return $this->capture_err();
            if($check > 0){
                  $resp['status'] = 'failed'; $resp['msg'] = "
Department already exist.";
                  return json_encode($resp);exit;}
            if(empty($id)){
                  $sql = "INSERT INTO `department_list` set {$data}";
                  $save = $this->conn->query($sql); }else{
                  $sql = "UPDATE `department_list` set {$data} where
id = '{$id}' "; $save = $this->conn->query($sql);}
            if($save){
                  $resp['status'] = 'success';
                  if(empty($id))
                        $this->settings->set_flashdata('success',"New
Department successfully saved.");else $this->settings-
>set_flashdata('success',"Department successfully updated.");
```

Fig. 27 *Create new department and partial code for department management*

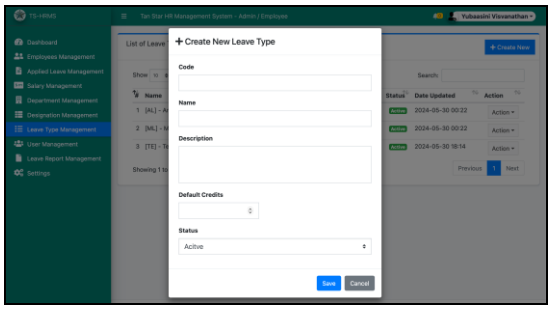
Designation Management: The designation administration module is crucial for managing duties like adding, modifying, and eliminating outdated designations, ensuring efficient task allocation. Its user-friendly interfaces and effective data handling make it easy for managers to manage designations, maintaining data integrity.




```
$check = $this->conn->query("SELECT * FROM `designation_list` where
`name` = '{$name}' ".(!empty($id) ? " and id != {$id} " : "")." )-
>num_rows;
            if($this->capture_err())return $this->capture_err();
            if($check > 0){ $resp['status'] = 'failed';
                  $resp['msg'] = " Designation already exist.";
                  return json_encode($resp);exit;}
            if(empty($id)){
                  $sql = "INSERT INTO `designation_list` set {$data}
";
                  $save = $this->conn->query($sql);
            }else{
                  $sql = "UPDATE `designation_list` set {$data} where
id = '{$id}' ";$save = $this->conn->query($sql);}
            if($save){
                  $resp['status'] = 'success';
                  if(empty($id))
                        $this->settings->set_flashdata('success',"New
Designation successfully saved.");
                  Else $this->settings-
>set_flashdata('success',"Designation successfully updated.");
```

Fig. 28 *Create new designation and partial code for designation management*

Leave Type Management: The module manages leave types, including sick leave and yearly holidays, ensuring correct data entry and neat arrangement. It eliminates duplication and provides real-time changes, ensuring seamless operations and clear feedback. Code snippets guide adding, updating, and deleting leave types.




```
$check = $this->conn->query("SELECT * FROM `leave_types` where `name` =
'{$name}' ".(!empty($id) ? " and id != {$id} " : "")." ")->num_rows;
            if($this->capture_err())return $this->capture_err();
            if($check > 0){ $resp['status'] = 'failed';
                  $resp['msg'] = " Leave Type already exist.";
                  return json_encode($resp); exit;}
            if(empty($id)){
                  $sql = "INSERT INTO `leave_types` set {$data} ";
                  $save = $this->conn->query($sql); }else{
                  $sql = "UPDATE `leave_types` set {$data} where id =
'{$id}' "; $save = $this->conn->query($sql); }
            if($save){
                  $resp['status'] = 'success';
                  if(empty($id)) $this->settings-
>set_flashdata('success',"New Leave Type successfully saved."); else
                        $this->settings->set_flashdata('success',"Leave
Type successfully updated.");
```

Fig. 29 *Create new leave type and partial code for leave type management*

Employee Management: The system's staff management features, such as `save_employee()' help maintain data integrity, streamline user administration, and securely change employee passwords based on their ID, ensuring successful workforce management.

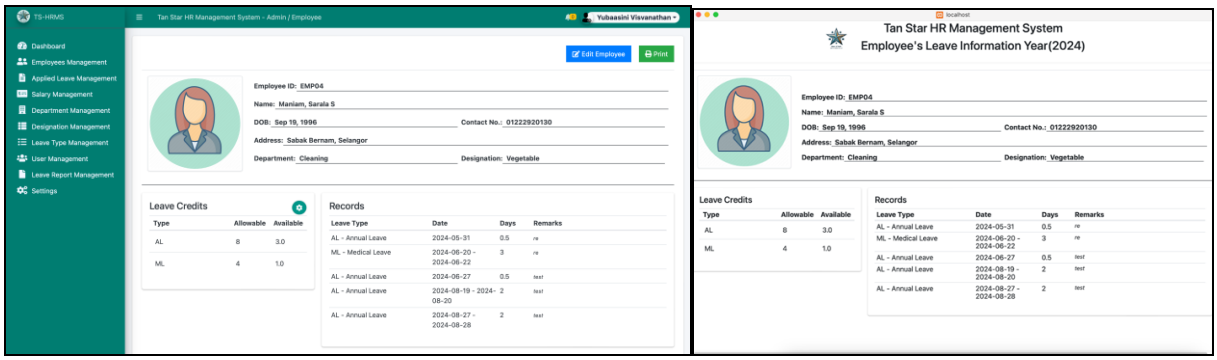
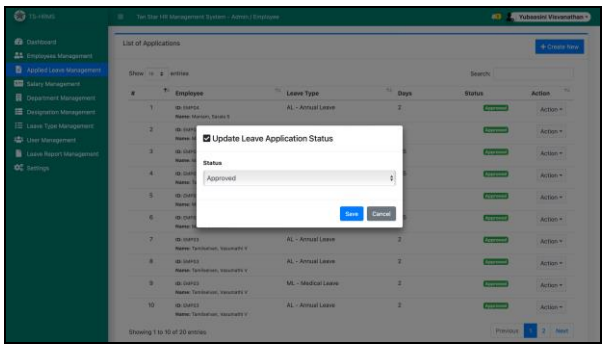


Fig. 30 *View and print employee details*

```
$chk2 = $this->conn->query("SELECT * FROM `users` where username
='{$username}' ".($id>0? " and id!= '{$id}' " : ""))->num_rows; $this-
>capture_err();
            if($chk2 > 0){$resp['status'] = 'failed';
                  $resp['msg'] = "Username is not available. Please
review and try again.";return json_encode($resp);
                  exit; }$data = "";
            foreach($_POST as $k =>$v){
if(in_array($k,array('firstname','lastname','middlename','username','ty
pe'))){
                        if(!empty($data)) $data.=" , "; $data .= "
`{$k}` = '{$v}' ";}}if(empty($id))
            $data .= ", `password` = md5('{$employee_id}') ";
            if(empty($id))
                  $sql1 = "INSERT INTO `users` set {$data} ";
            else
                  $sql1 = "UPDATE `users` set {$data} where id =
'{$id}' ";$save1 = $this->conn->query($sql1);
            $this->capture_err();
            if(!$save1){
                  $resp['status'] = 'failed';
                  $resp['error_sql'] = $sql1;
            }$user_id = empty($id) ? $this->conn->insert_id : $id ;
            $this->conn->query("DELETE FROM `employee_meta` where
user_id = '{$user_id}' and meta_field not in
('leave_type_ids','leave_type_credits') ");
```

Applied Leave Management: The `save_application()` function is crucial in our leave management system, reviewing and approving staff leave requests, ensuring they align with allotted credits and timetables. It also manages new requests, updates existing ones, and assesses the operation's success.




```
$meta_qry = $this->conn->query("SELECT * FROM employee_meta where
user_id = '{$user_id}' ");while($row = $meta_qry->fetch_assoc()){
                $meta[$row['meta_field']] = $row['meta_value']; }
            $leave_type_credits = isset($meta['leave_type_credits']) ?
json_decode($meta['leave_type_credits']) : array();$ltc = array();
            foreach($leave_type_credits as $k=> $v){
                $ltc[$k] = $v; } $used = $this->conn->query("SELECT
COALESCE(sum(`leave_days`),0) as total FROM leave_applications where
user_id = '{$user_id}' and `leave_type_id` = '{$leave_type_id}' and
date_format(date_start,'%Y') = '".date('Y')."' and
date_format(date_end,'%Y') = '".date('Y')."' and status = 1 ")-
>fetch_array()['total']; $allowed = (isset($ltc[$leave_type_id])) ?
$ltc[$leave_type_id] : 0; $available =  $allowed - $used;
            if(!isset($ltc[$leave_type_id])){$resp['status'] =
'failed';$resp['msg'] = " Selected employee does not have previlege
for the selected leave type.";return json_encode($resp);exit; }
            if($leave_days > $available){$resp['status'] = 'failed';
                $resp['msg'] = " Days of Leave is greated than
available days of selected leave type. Available
(($available)).";return json_encode($resp); exit; }
```

Fig. 31 *Update leave application status and partial code for applied leave management*

Salary Management: The `save_salary()` method is crucial for our system's salary administration, processing and validating salary data. It retrieves user data like department and designation to calculate total compensation.

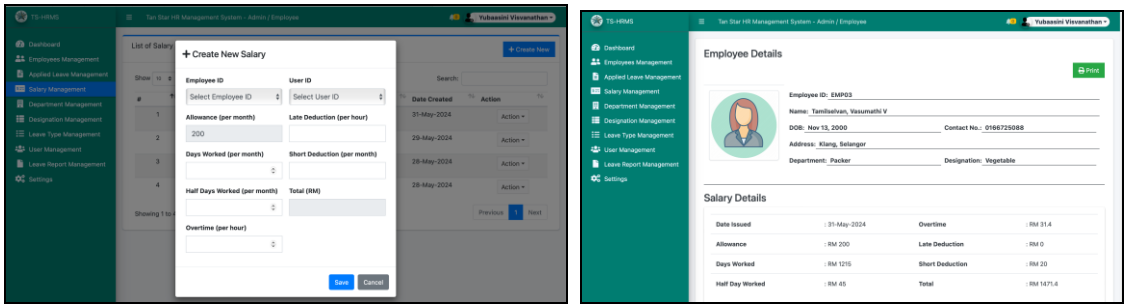


Fig. 32 *Create new and view salary slip*

```
$meta_qry = $this->conn->query("SELECT meta_field, meta_value FROM
employee_meta WHERE user_id = '{$employee_id}'");
    $user_meta = array();
    while($row = $meta_qry->fetch_assoc()){
        $user_meta[$row['meta_field']] = $row['meta_value']; }
    // For example, if you need the 'department' and 'designation'
fields:
    $department_id = isset($user_meta['department']) ?
$user_meta['department'] : '';
    $designation_id = isset($user_meta['designation']) ?
$user_meta['designation'] : '';
    // Calculate the total salary
    $total = $allowance + ($day_worked * 45) + ($half_day_worked *
22.5) + ($overtime * 7.85) - $late_d - $late_s;
    // Save the salary record
    $sql = "INSERT INTO salary_list (employee_id, user_id, basic,
allowance, day_worked, half_day_worked, overtime, late_d, late_s,
total)
            VALUES ('$employee_id', '$user_id', '$basic', '$allowance',
'$day_worked', '$half_day_worked', '$overtime', '$late_d', '$late_s',
'$total')";
```

User Management: This PHP code has two main functions: `save_users` and `delete_users`. It handles password hashing, file upload management, and user information adding. It removes user records and avatar files from the database using SQL queries and session data.

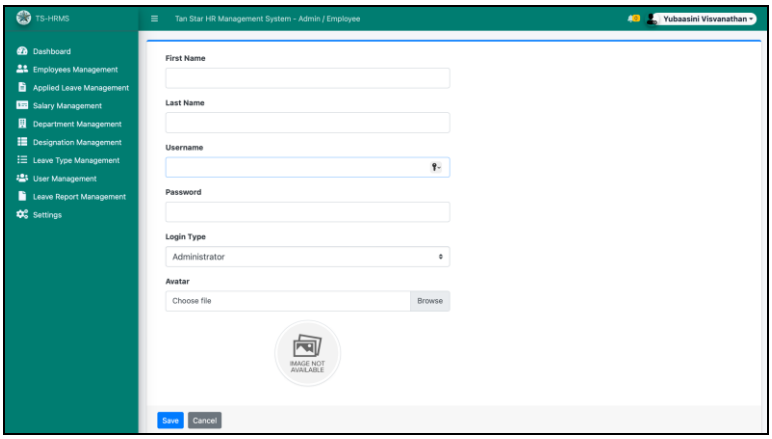
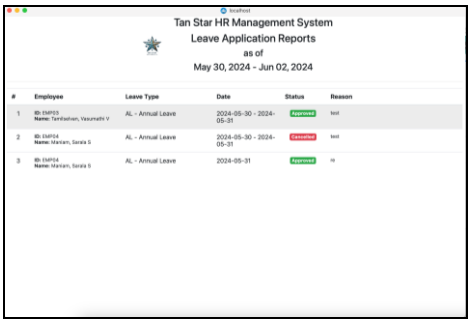


Fig. 34 *Create new system user*

```
$chk = $this->conn->query("SELECT * FROM `users` where username
='{$username}' ".($id>0? " and id!= '{$id}' " : ""))->num_rows;
if($chk > 0){
                    return 3;exit;}
                foreach($_POST as $k => $v){
if(!in_array($k,array('id','password'))){if(!empty($data)) $data .=" ,
";
                        $data .= " {$k} = '{$v}' ";}}
                if(!empty($password)){
                    $password = md5($password);
        if(!empty($data)) $data .=" , ";$data .= " `password` =
'{$password}' ";}
                if(isset($_FILES['img']) && $_FILES['img']['tmp_name'] !=
''){
                    $fname = 'uploads/'.strtotime(date('y-m-d
H:i')).'_'.$_FILES['img']['name'];
                    $move =
move_uploaded_file($_FILES['img']['tmp_name'],'../'. $fname);
                    if($move){$data .=" , avatar = '{$fname}' ";

        if(isset($_SESSION['userdata']['avatar']) &&
is_file('../'.$_SESSION['userdata']['avatar']) &&
$_SESSION['userdata']['id'] ==
$id)unlink('../'.$_SESSION['userdata']['avatar']);}}
                if(empty($id)){
                    $qry = $this->conn->query("INSERT INTO users set
{$data}");
                    if($qry){
                        $this->settings->set_flashdata('success','User
Details successfully saved.');return 1;}else{return 2;}}else{
        $qry = $this->conn->query("UPDATE users set $data where id =
{$id}");
$qry = $this->conn->query("DELETE FROM users where id = $id");
                if($qry){$this->settings
```

Leave Report Management: The code snippet displays a list of leave applications from a database, filtering records based on start and end dates. The table includes employee ID, leave type, dates, status, and reason. Colored badges graphically show leave status, and "No Records" is displayed if no records are found.




```
$sql = "SELECT l.*,concat(u.lastname,', ',u.firstname,' ',u.middlename)
as `name`,lt.code,lt.name as lname from `leave_applications` l inner
join `users` u on l.user_id=u.id inner join `leave_types` lt on lt.id =
l.leave_type_id where ((date(l.date_start) BETWEEN '$date_start'  and
'$date_end') OR (date(l.date_end) BETWEEN '$date_start'  and
'$date_end') ) order by unix_timestamp(l.date_start)
asc,unix_timestamp(l.date_end) asc";$qry = $conn->query($sql);
while($row = $qry->fetch_assoc()): $lt_qry = $conn->query("SELECT
meta_value FROM `employee_meta` where user_id = '{$row['user_id']}' and
meta_field = 'employee_id' "); $row['employee_id'] = ($lt_qry->num_rows
> 0) ? $lt_qry->fetch_array()['meta_value'] : "N/A";
```

Fig. 35 *Print leave reports and partial code for leave report management*