PERSONEL MANAGEMENT SYSTEM
SOFTWARE ENGINEERING & AGILE DEVELOPMENT REPORT

Submitted by

Anusha - 2022BCSE07AED796

Polamreddy Rohini - 2022BCSE07AED733

K Sai Srivalli - 2022BCSE07AED969

Nanditha D N – 2022BCSE07AED803

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY
IN
COMPUTER SCIENCE AND ENGINEERING

Under the Supervision of

Mano J Shankari Professor, Department of CSE

Department of Computer Science and Engineering

ALLIANCE COLLEGE OF ENGINEERING AND DESIGN

ALLIANCE UNIVERSITY

Private University established in Karnataka State by Act No.34 of year 2010
Recognized by the University Grants Commission (UGC), New Delhi

BONAFIED CERTIFICATE

This is to certify that the project work entitled **"PERSONNEL MANAGEMENT SYSTEM"** is the Bonafide work done by

Anusha - 2022BCSE07AED796,

Polamreddy Rohini - 2022BCSE07AED733,

K Sai Srivalli - 2022BCSE07AED969,

Nanditha D N - 2022BCSE07AED803, submitted in partial fulfilment of the requirements for the award of the degree Bachelor of Technology in **COMPUTER SCIENCE AND ENGINEERING** during the Academic year 2025- 2026.

Miss.Mano J Shankari
Supervisor

Dr. Ramalakshmi K
Head of Department

ALLIANCE UNIVERSITY

Private University established in Karnataka State by Act No.34 of year 2010
Recognized by the University Grants Commission (UGC), New Delhi

DECLARATION

This is to declare that the report titled **"PERSONNEL MANAGEMENT SYSTEM"** has been made for the partial fulfilment of the Course Bachelor of Technology in Computer Science Engineering, under the guidance of **Mano J Shankari**. We confirm that this report truly represents our work as a part of our **Software Engineering and Agile Development** project work. This work is not a replication of work done previously by any other person. We also confirm that the contents of the report and the views contained therein have been discussed and deliberated with the faculty guides.

Student Name	**Reg.no**	**Signature**
Anusha	2022BCSE07AED796	
Polamreddy Rohini	2022BCSE07AED733	
K Sai Srivalli	2022BCSE07AED969	
Nanditha D N	2022BCSE07AED803	

ALLIANCE UNIVERSITY

Private University established in Karnataka State by Act No.34 of year 2010
Recognized by the University Grants Commission (UGC), New Delhi

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of the task would be put incomplete without the mention of the people that made it possible, whose constant guidance and encouragement crown all the efforts successfully.

We are very thankful to our supervisor **Mano J Shankari** for her sustained inspiring guidance and cooperation throughout the process of this project. Her wise counsel and valuable suggestions are invaluable.

We avail this chance to express our deep sense of gratitude and hearty thanks to the Management of Alliance University, for providing world class infrastructure, congenial atmosphere, and encouragement.

We express our deep sense of gratitude and because of the teaching and non-teaching staff at our department who stood with us during the project and helped us to make it a successful venture.

We place highest regards to our parents, our friends and well-wishers who helped plenty in making the report of this project.

Anusha

Polamreddy Rohini

K Sai Srivalli

Nanditha D N

Table of Content

1.Problem Statement

In modern organizations, managing personnel efficiently is critical for smooth operations and employee satisfaction. Traditional methods of maintaining employee records, processing payrolls, tracking attendance, and managing leaves often involve manual paperwork or outdated software, leading to inefficiencies, errors, and security vulnerabilities. Organizations struggle with challenges such as data redundancy, miscommunication, delays in leave approvals, payroll miscalculations, and difficulties in performance evaluation.

A **Personnel Management System (PMS)** is required to automate and centralize HR-related tasks, ensuring seamless employee record-keeping, attendance tracking, leave management, payroll processing, and performance monitoring. The system will provide role-based access, enabling HR managers to manage employee data securely while allowing employees to access their own information and submit requests.

By digitizing personnel management, the proposed system will improve accuracy, save time, enhance transparency, and provide real-time data for better decision-making. It will also ensure compliance with labour laws and organizational policies while maintaining data security and confidentiality. The goal is to create a user-friendly, efficient, and scalable system that optimizes human resource management and contributes to the overall productivity of the organization.

2.Software Requirements Specifications (SRS)

1. Introduction

1.1 Purpose

The **Personnel Management System (PMS)** is designed to help organizations efficiently manage employee records, leave requests, payroll, and other HR-related activities. It aims to reduce manual errors, improve data accessibility, and enhance the decision-making process.

1.2 Scope

The system will store and manage employee details, including personal information, job roles, and salary details.

- It will allow employees to request leave and track attendance.
- The payroll module will calculate salaries based on working hours and deductions.
- The system will generate reports on employee performance, leave history, and salary distribution.
- Role-based access control will be implemented for security.

1.3 Definitions, Acronyms and Abbreviations

- **PMS –** Personnel Management System
- **HR –** Human Resources
- **UI –** User Interface
- **DBMS –** Database Management System

1.4 References

- **IEEE Software Requirements Specification (SRS) standard**
- **HR Management Best Practices**

1.5 Overview

This document describes the functional and non-functional requirements of the system, including user roles, design constraints, and system features.

2. General Description

2.1 Product Perspective

The system will be a web-based application that integrates with an existing payroll system (if required). It will replace manual personnel management methods with a digital, automated solution.

2.2 User Characteristics

- **HR Managers:** Manage employee records, payroll, and performance reviews.
- **Employees:** View personal details, apply for leaves, and track attendance.
- **Admin:** Configure system settings, manage roles, and generate reports.

2.3 Constraints

- The system must be secure and accessible only by authorized personnel.
- It should handle concurrent users efficiently.
- should compile with data protection laws.

2.4 Assumptions and Dependencies

- Internet connection is required for cloud-based deployment.
- Employees have unique IDs for system authentication.

3. Specific Requirements

3.1 Functional Requirements

1. **User Authentication & Role Management**
    - Admin, HR, and employees should have different access levels.
    - Secure login with password hashing.
2. **Employee Management**
    - Add, update, delete employee records.
    - Assign employees to departments.
3. **Leave Management**
    - Employees can apply for leave.
    - HR can approve/reject leave requests.
4. **Attendance & Payroll**
    - Track employee work hours.
    - Calculate salary based on attendance and deductions.

5. Reports Generation

- Generate monthly payroll reports.
- View employee performance analytics.

3.2 Non-Functional Requirements

1. **Performance:** The system should handle up to 1000 concurrent users.
2. **Security:** Implement encryption for sensitive data.
3. **Usability:** The UI should be intuitive and easy to use.
4. **Scalability:** Should support the addition of more modules in the future.

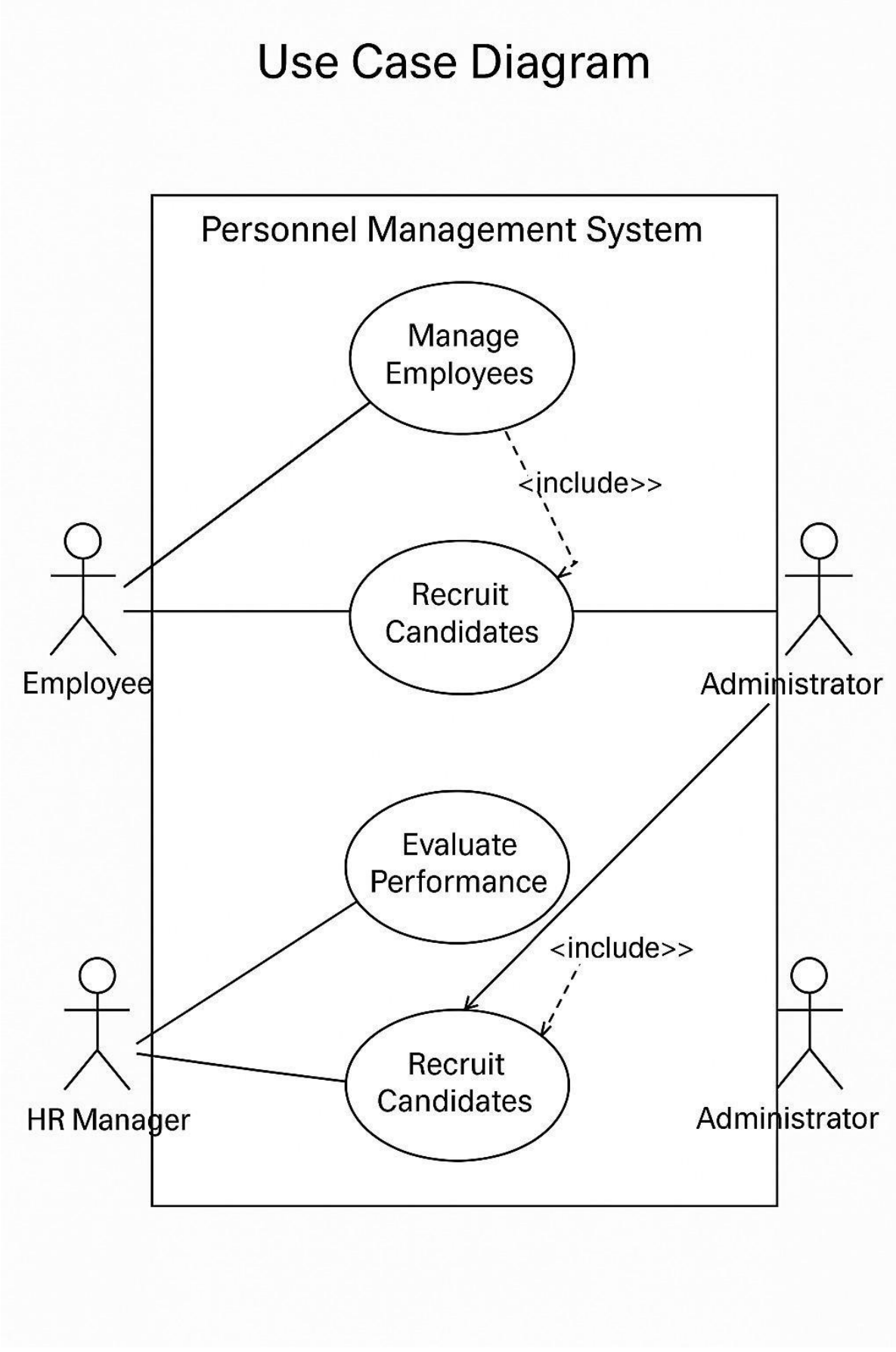
3.Software Requirements

1. **Backend & Database**
   - **XAMP-**Includes Apache (web server), MySQL (database), PHP (server-side scripting), and phpMyAdmin (database management GUI).
   - **PHP 8+ –** Used for server-side scripting and backend logic**.**
   - **MySQL –** Relational Database Management System (RDBMS) for storing and managing personnel data.
2. **Frontend**
   - **HTML5 & CSS3** – Used for structuring and styling the web pages.
   - **JavaScript** – Enhances interactivity and provides client-side form validation.
3. **Development Tools**
   - **VS Code / Sublime Text / PHPStorm** – Code editors for writing and managing the project files.
   - **phpMyAdmin** – Web-based GUI for managing MySQL databases efficiently.

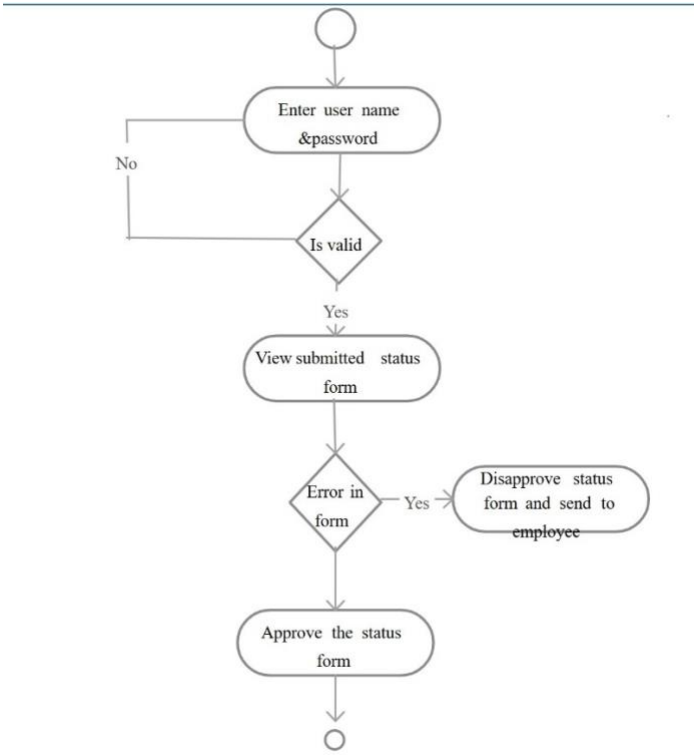
4.UML Tools

The UML Tool that we have used in this project is **AgroUML.** AgroUML, now known as diagrams.net, is a versatile online diagramming tool that allows users to create a wide range of diagrams, including UML (Unified Modeling Language) diagrams. It features a user-friendly interface with drag-and-drop functionality, making it accessible for both beginners and experienced users. With a rich library of shapes and templates, AgroUML supports various diagram types such as class diagrams, sequence diagrams, and use case diagrams. Additionally, it offers collaboration features, enabling multiple users to work on diagrams in real-time, and integrates seamlessly with cloud storage services like Google Drive and Dropbox for easy sharing and storage.

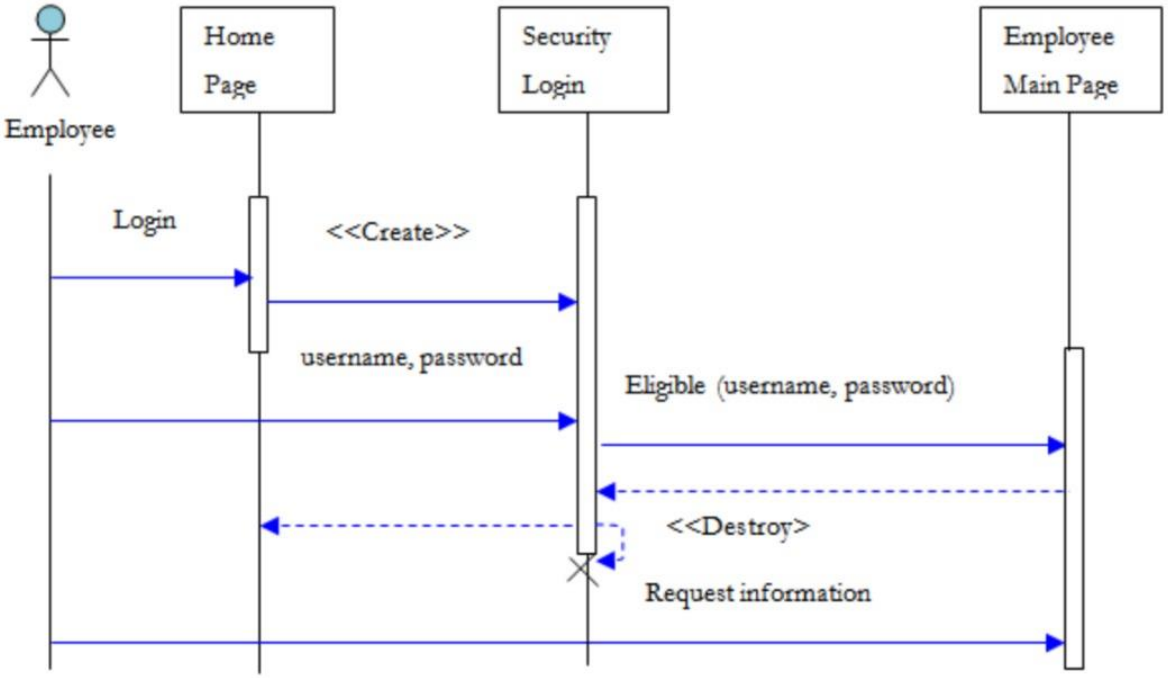
5.Use Case Diagram



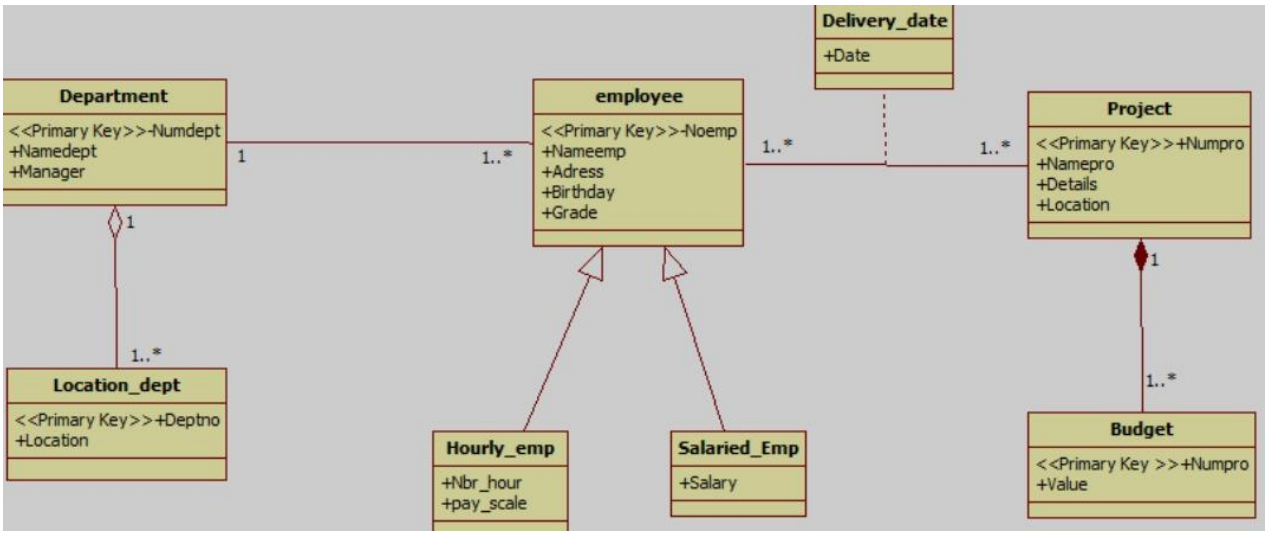
Activity Diagram



Sequence Diagram



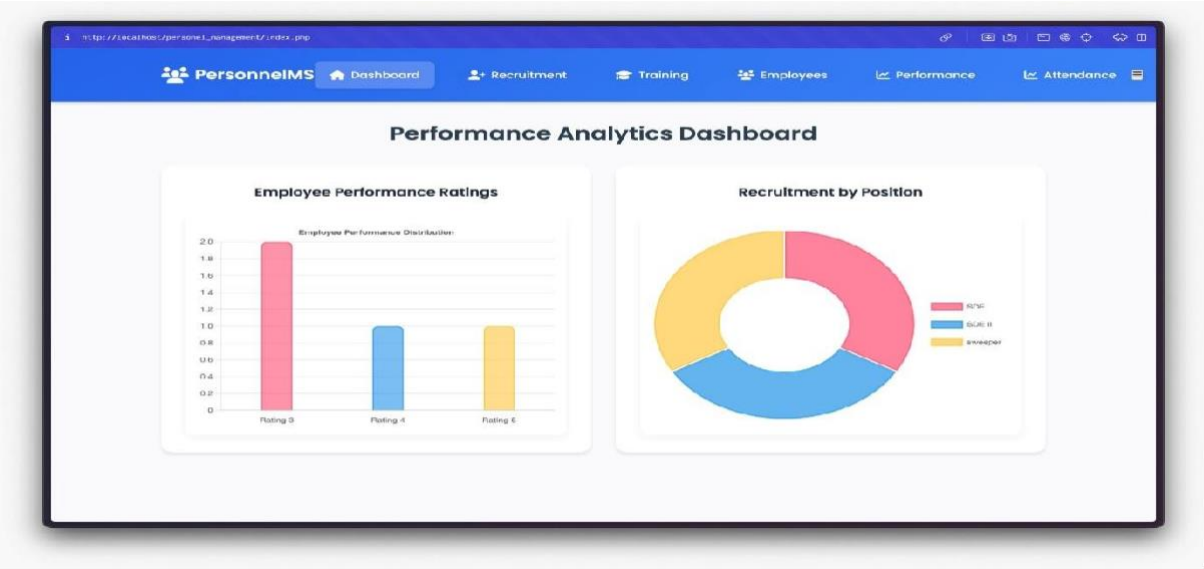
Class Diagram



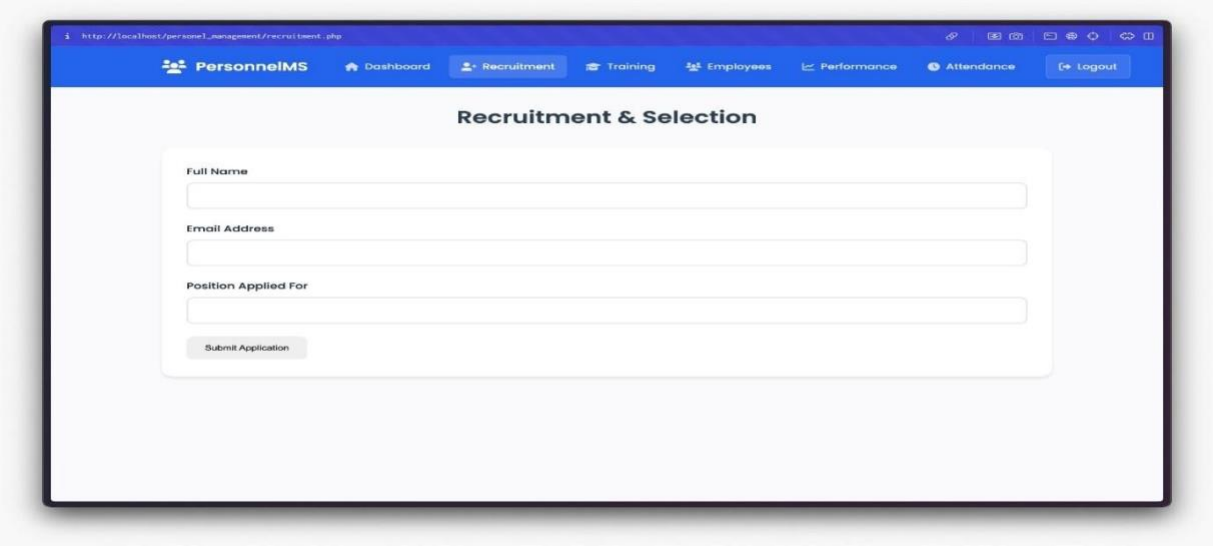
6.Connectivity Code

```php
<?php
$servername = "localhost";
$username = "root"; // Default username in XAMPP
$password = ""; // No password by default
$database = "personnel_management";

// Create connection
$conn = new mysqli($servername, $username, $password, $database);

// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
?>
```

7.Frontend Form

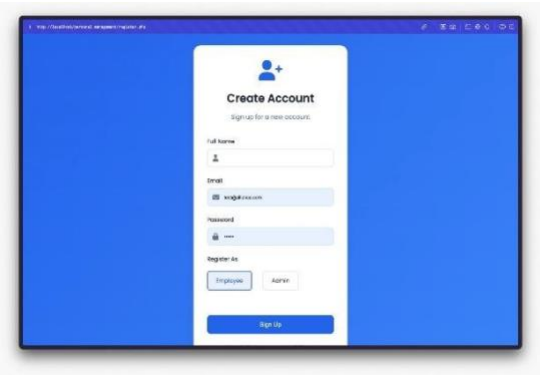


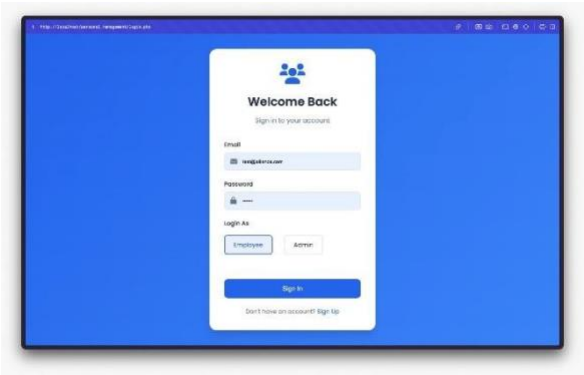
Dashboard



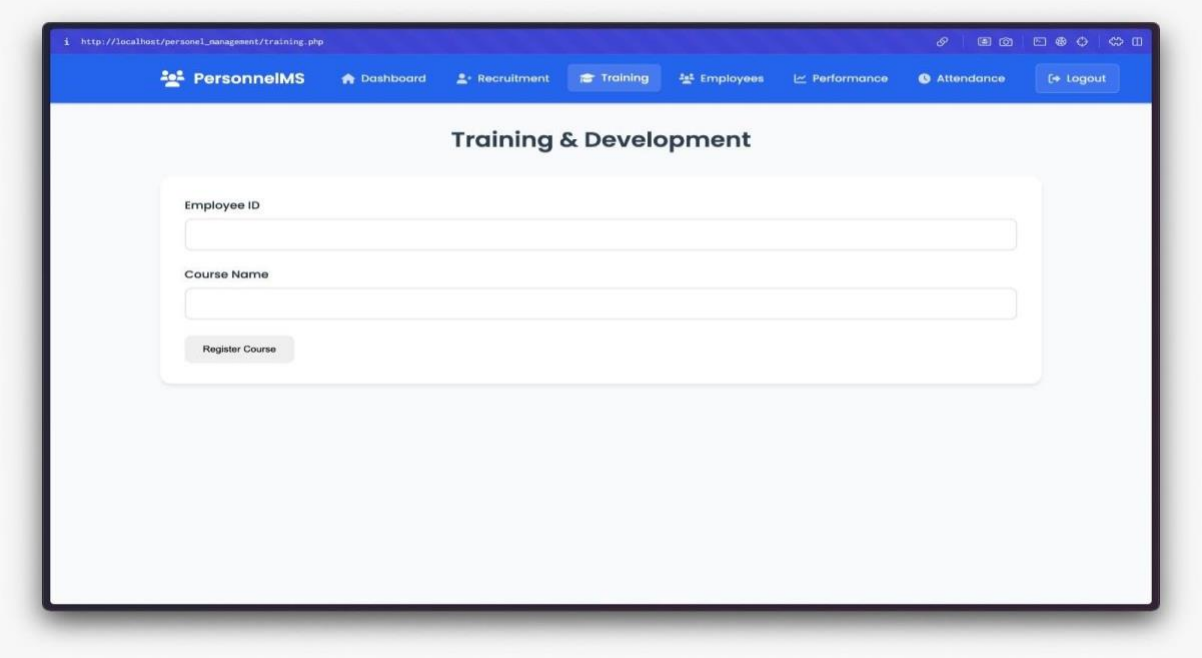
Recruitment Page for Admin

Employee and Admin Login



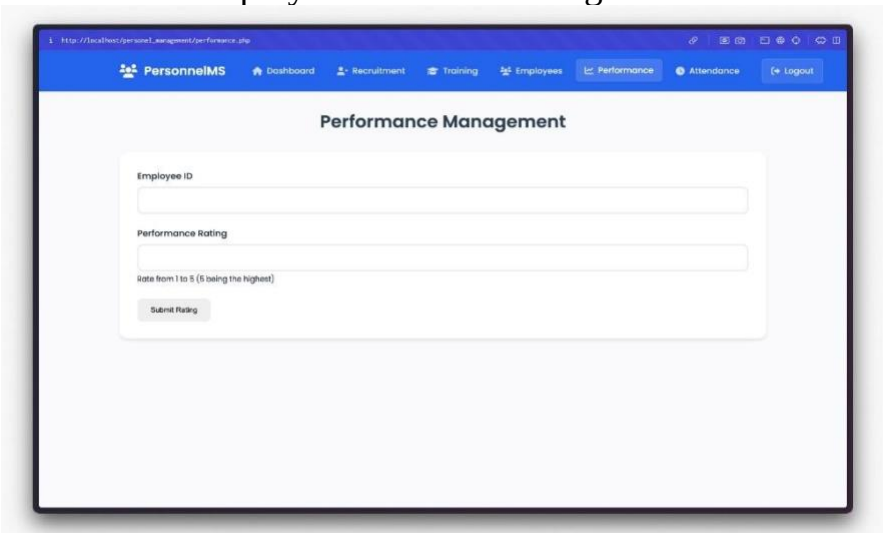


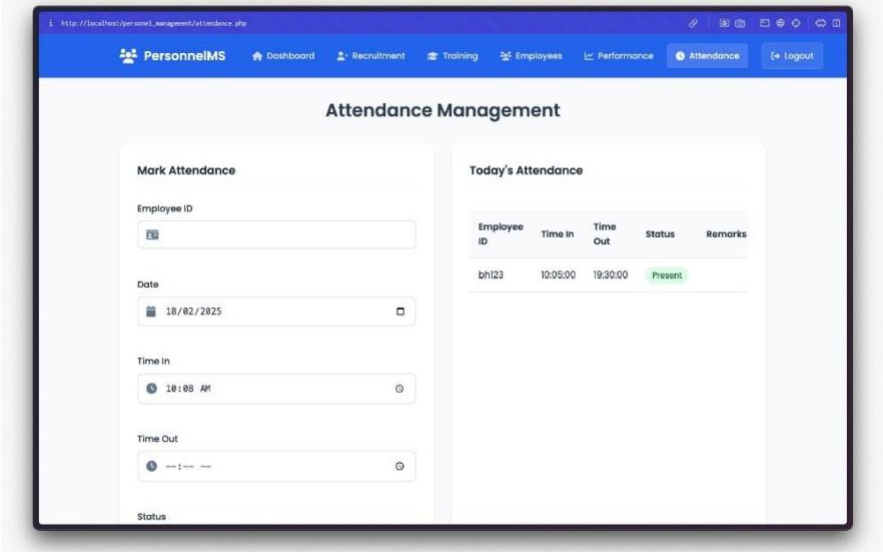
Sign in Page



Training Form

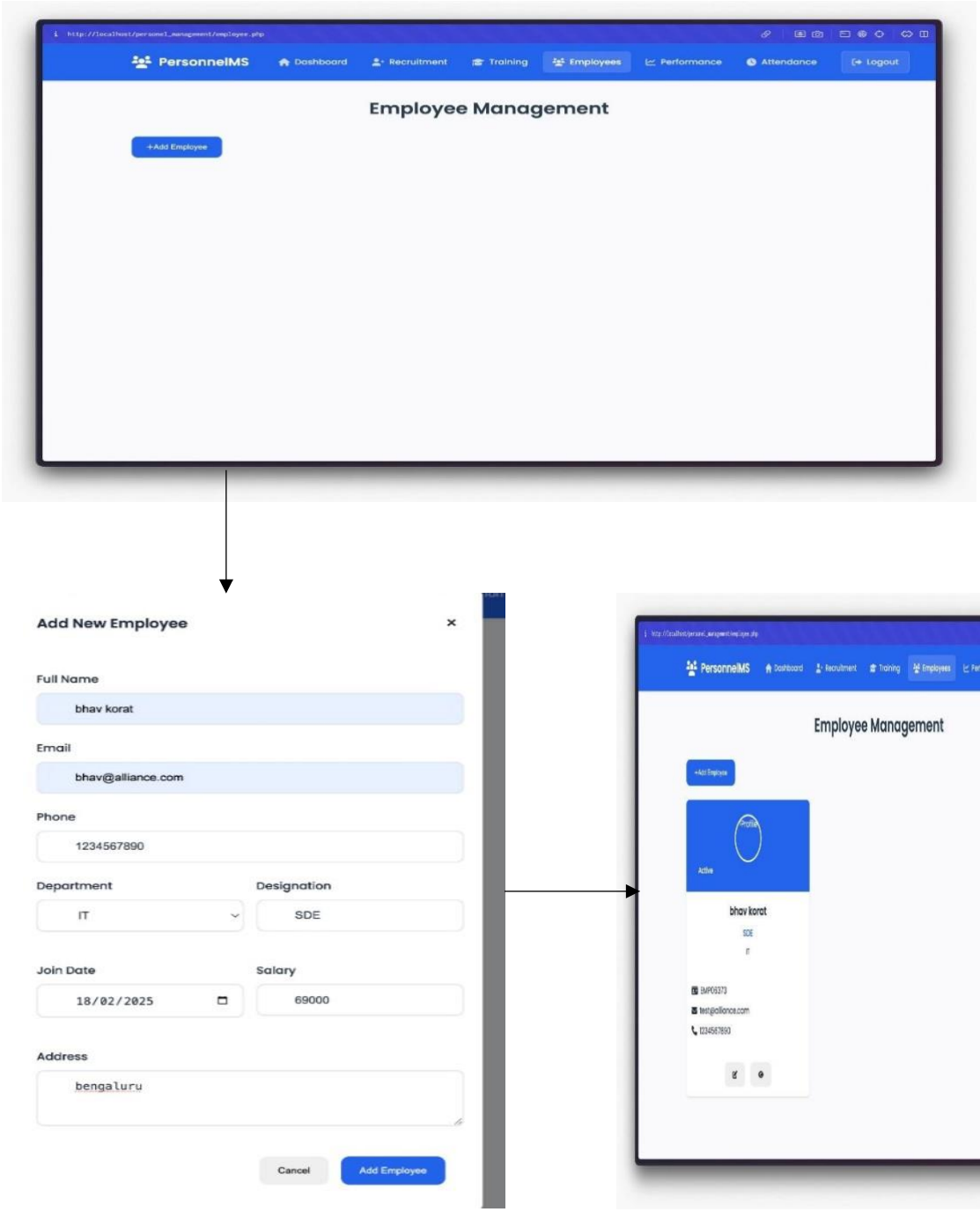
Employee Performance Page





Attendance Form

Employee Management



Test Cases

Test ID	Test Condition	Test Steps	Test I/P	Expected Result	Actual Result	Status	Remarks
1	Check login functionality	Enter valid credentials → Click login	Valid username & password	Dashboard should load	Dashboard loaded successfully	Pass	Login working as expected
2	Check invalid login	Enter invalid credentials → Click login	Invalid username & password	Invalid credentials message	Invalid credentials message displayed	Pass	Proper error handling in place
3	Check session timeout	Login → Stay idle for X minutes → Try navigating	None	Users should be logged out	User was not logged out	Fail	Session timeout not implemented successfully
4	Check forgot password	Click forgot password → Enter registered email → Submit	Registered email	Password reset link sent	No email received	Fail	Issue with email delivery system
5	Check add employee	Navigate to add employee → Enter valid details → Click "Save"	Employee Details	Employee should be added successfully	Employee record stored successfully	Pass	Employee record stored successfully
6	Check duplicate employee	Enter an existing employee ID → Click "Save"	Existing Employee ID	"Employee already exists" message displayed	Employee already exists	Pass	Duplicate validation works

7	Check edit employee	Select an employee → Edit details → Update	Updated Details	Employee details should be updated	Updated query issue in DB	Pass	Updated query issue in Database

8	Check delete employee	Select an employee → Click "Delete" → Confirm action	Employee ID	Employee should be deleted successfully	Deletion is working fine	Pass	Deletion is working fine
9	Check search employee	Enter name ID in search bar → Click "Search"	Name ID	Relevant employee record displayed	Records not displayed	Fail	Search is not functioning
10	Check role-based access	Login with an admin → Try accessing admin-only page	Admin credentials	Access Denied message	Non-admin was able to access	Fail	Security not working fine
11	Check profile upload	Navigate to Profile → Edit details → Save changes	Updated User Info	Profile should update successfully	Profile not updated successfully	Fail	Profile edit is not working

12	Check file upload	Navigate to "Upload Doc" → Upload a valid document	Valid File (PDF, DOC)	File should be uploaded	File uploaded successfully	Pass	File storage works properly
13	Check invalid file upload	Upload an unsupported file format	Unsupported Format	"Unsupported file format" message displayed	Unsupported file format message displayed	Pass	Proper validation in place
14	Check department creation	Navigate to "Department" → Add a new department	Department Details	Department should be added correctly	Department added successfully	Pass	Department module fine
15	Check department deletion	Select a department → Click "Delete"	Department ID	Department should be removed	Department deleted successfully	Pass	No issues found
16	Check employee leave request	Navigate to "leave request" → Submit/Approve	Leave Details	Leave request should be submitted	Leave request submitted	Pass	Leave request feature working fine
17	Check leave approval	Admin navigate to "leave request" → Approve	Leave ID	Leave status should be updated	Status remained unchanged	Fail	Approval logic not working

18	Check payroll processing	Navigate to payroll → Generate payroll	Employee ID	Payroll should be generated	Payroll is not generated	Fail	Payroll generation issue
19	Check email notification	Perform an action that triggers mail	Various Triggers	Email should be sent	Email not sent	Fail	SMTP configuration needed
20	Check system logs	Admin navigate to logs	None	Logs should be displayed	Logs displayed successfully	Pass	Login works as expected
21	Check logout functionality	Click logout	None	User should be logged out	User was not logged out	Fail	Session terminated successfully

22	Check mobile responsiveness	Open system on mobile → Surf page	Mobile Device	Page should load correctly	Page loaded correctly	Pass	CSS query working
23	Check cross-browser compatibility	Open system in different browsers	Systems on various browsers	System should work across browsers	System works fine in every browser	Pass	Browser compatible system
24	Check system performance	Load dashboard with 1000+ employees	Large data	System should load within 10 seconds	Takes too long to load	Fail	Performance optimization needed

25	Check DB Integrity	Program reason with multiple connections	SQL Query	Data should be consistent	Data is consistent	Pass	DB integrity is working fine

Conclusion and Future Scopes

Conclusion

The Personnel Management System enhances efficiency by streamlining key HR processes such as employee management, leave approvals, payroll processing, and access control. Through systematic testing, we identified functional strengths and areas requiring improvement, ensuring the system meets essential operational needs. Security measures like role-based access control further strengthen data protection, while features like document uploads and email notifications improve workflow automation. Despite minor issues in session handling and performance optimization, the system provides a solid foundation for digital HR management.

Future Scopes

To further improve functionality and adaptability, the system can be enhanced with:

- **AI-Based Recruitment Screening:** Automating candidate filtering based on resumes for efficient hiring.
- **Automated Performance Analysis:** Leveraging AI-driven analytics to assess employee productivity.
- **Enhanced Role-Based Access Control:** Implementing stricter access levels for Admins, HR, and Employees to enhance security.
- **Payroll Management Integration:** Extending the system to manage salaries, benefits, and tax calculations.
- **Mobile App Development:** Developing an Android/iOS application for remote access and enhanced usability.

These future enhancements will ensure scalability, security, and efficiency, making the Personnel Management System a comprehensive solution for modern HR operations.

8.Agile Development

Experiment:1

Aim: To understand and implement Agile methodology in developing a ***Smart Personnel Management System,*** focusing on faster delivery, continuous feedback, and better adaptability.

Procedure:

Step-1

Requirement Gathering:

Collected user needs (e.g., managing employee records, login, leave management).

Step-2

Sprint Planning:

Divided the work into 2-week sprints (e.g., Sprint 1: Login & Signup, Sprint 2: Dashboard, etc.).

Step-3

Followed Agile ceremonies like:

- Daily stand-up meetings
- Sprint reviews
- Retrospectives for improvements

Step-4

Development & Testing:

- Developed and tested features during each sprint.
- Handled feedback quickly and made changes accordingly.

Step-5

Refactoring:

Improved code after feedback without affecting features.

Step-6

Deployment:

After final testing, the system.

Output:

A working *Smart Personnel Management System* with:

- Login & Signup
- Employee Dashboard
- Data storage and update
- Admin panel (optional)

Result:

The Agile methodology helped in developing the system step-by-step. Regular feedback and early testing improved software quality. Changes were handled smoothly within sprints, ensuring the final product met user expectations.

Experiment:2

Aim: To understand the business value of adopting the Agile methodology in developing the *Smart Personnel Management System*, focusing on early delivery, feedback-driven changes, and maximizing value to end-users.

Procedure:

Step-1

Prioritized Features:

- Identified key business needs like employee management, attendance tracking, and leave applications.
- Prioritized these features based on importance to HR/Admin staff.

Step-2

Agile Planning:

- Broke down tasks into sprints (e.g., Sprint 1: Employee Login, Sprint 2: Dashboard).
- Delivered the most important features first.

Step-3

Early Feedback:

- Showed working parts of the system to users (e.g., staff or mock clients) after every sprint.
- Collected their input and made quick changes.

Step-4

Value Delivery:

- Delivered working software modules at the end of every sprint.
- Focused on the features that saved time and effort for HR staff.

Step-5 Adaptability:

Respond to changes like UI improvements, new data fields, or bug fixes without disturbing the whole project.

Output:

A functional Smart Personnel Management System with:

- Core features prioritized and delivered early.
- User-friendly interface based on real-time feedback.
- Minimal delays and maximum value in each sprint.

Result:

Using the Agile approach added high business value to the project. The development process was smooth and flexible. Feedback from users was

implemented quickly, which improved user satisfaction. The final product aligned closely with actual business needs, leading to better usability and effectiveness.

Experiment:3

Aim: To understand and implement core Agile development practices in building a *Smart Personnel Management System*, ensuring efficient collaboration, fast delivery, and high-quality outcomes.

Procedure:

Step-1

Defined User Stories:

Created user stories like: "As an employee, I want to log in securely" and "As an admin, I want to view employee records."

Step-2

Planned Sprints:

Divided the work into 2-week sprints:

- Sprint 1: Login, Signup
- Sprint 2: Dashboard and Profile Management
- Sprint 3: Attendance and Leave Module

Step-3

Daily Stand-Ups (Mock or Real):

Discussed tasks, blockers, and progress regularly (as a team or self-reviewed).

Step-4

Development with Continuous Integration:

Code was frequently updated, tested, and integrated using Git.

Step-5

Refactoring:

Improved code readability and structure during each sprint without changing outputs.

Step-6

Sprint Reviews & Retrospectives:

At the end of each sprint, reviewed completed tasks and noted what could be improved.

Result:

The use of Agile development practices helped in building the system in smaller, manageable parts. Regular reviews and feedback improved the design and functionality. These practices ensured better teamwork, adaptability to changes, and delivery of a system that meets real user needs.

Experiment:4

Aim: To propose and design a sample project titled *Smart Personnel Management System* by defining its objectives, vision statement, use cases, and visualizing it using standard UML diagrams.

Procedure:

Step-1

Defined Project Objectives:

- Clarified the core purpose of the system.
- Identified key features required by end-users (HR/Admins & Employees).

Step-2

Created Vision Statement:

Wrote a concise and clear long-term goal for the system.

Step-3

Identified Use Cases:

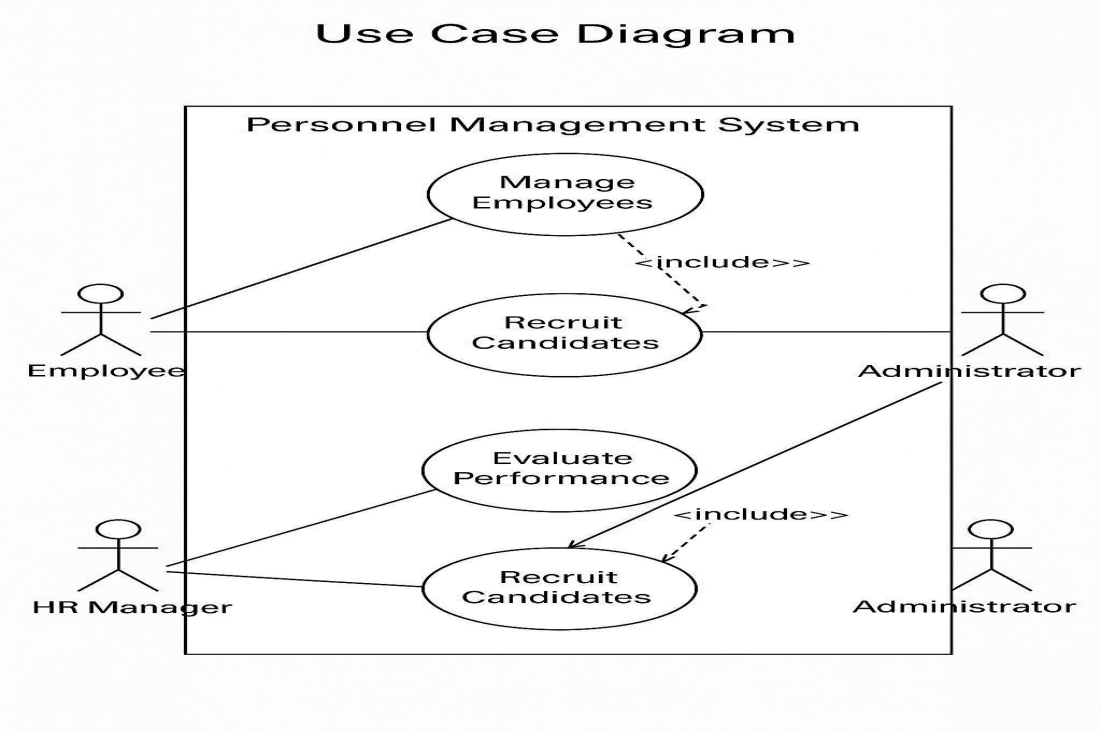
Listed key functions the system should perform from the user's perspective.

Step-4

Designed UML Diagrams:

- Created UML diagrams including:
  - Use Case Diagram

Output Diagram:



Result:

The Smart Personnel Management System was successfully proposed with detailed objectives, a strong vision, relevant use cases, and meaningful UML diagrams. This structured design helped in better planning, development, and understanding of the system architecture.

Experiment:5

Aim: To compile a structured release map and user stories for the *Smart Personnel Management System*, ensuring smooth planning and Agile development aligned with user needs.

Procedure:

Step-1

Identified Core Modules:

- Split the entire project into functional modules like login, profile management, leave management, etc.

Step-2

Created a Project Release Map:

- Divided the project timeline into sprints (2-week cycles).
- Mapped features to each sprint based on priority.

Step-3

Wrote User Stories:

Defined system features from the perspective of both employees and admins.

Step-4:

Mapped Stories to Releases:

Assigned each user story to a specific sprint/release to track progress.

Project Release Map:

Sprint/Release	Planned Features
Sprint-1	User Login/Signup, Role-based access
Sprint-2	Employee Dashboard, Profile View/Edit
Sprint-3	Leave Application & Approval system
Sprint-4	Attendance Tracking, Report Generation
Sprint-5 (Final)	Admin Controls, UI Enhancements, Bug Fixes

User Stories:

Employee User Stories:

1. **Login:**
   As an employee, I want to log in securely so that I can access my profile.

2. **View Profile:**
   As an employee, I want to view and update my profile so that my information stays current.
3. **Apply for Leave:**
   As an employee, I want to apply for leave so that I can request time off easily.
4. **Mark Attendance:**
   As an employee, I want to mark my daily attendance so that it is recorded digitally.

Admin User Stories:

5. **Approve/Reject Leave:**
   As an admin, I want to review and act on leave requests so that leave can be managed effectively.
6. **Add New Employee:**
   As an admin, I want to add new employees' data so that the system stays up to date.
7. **Generate Reports:**
   As an admin, I want to generate attendance and leave reports so that HR decisions can be made based on data.

Output:

- A detailed project release map with clear sprint planning.
- Well-structured user stories for both employees and admins.
- Stories are assigned to specific sprints, improving traceability and planning.

Result:

The release map and user stories provided a solid roadmap for development. Each feature was planned, prioritized, and aligned with user needs, helping the Agile team stay focused and deliver valuable software within each sprint. The process ensured clarity, collaboration, and faster delivery of the Smart Personnel Management System.

Experiment:6

Aim:
To identify storyboarding tasks and develop a detailed release plan for the *Smart Personnel Management System,* ensuring clear visualization of user interaction and systematic delivery of software features.

Procedure:

Step-1

Created Storyboards:

- Sketched key user interactions (screens like Login, Dashboard, Leave Application).
- Mapped user navigation between screens (step-by-step actions).

Step-2

Identified Tasks for Each Screen:

- Broke down UI elements and backend processes for each screen.
- Linked tasks to corresponding user stories.

Step-3

Planned Releases Based on Tasks:

- Organized development into sprints with realistic timelines.
- Prioritized core functionalities first, followed by enhancements.

Step-4

Assigned Resources & Set Milestones:

- Estimated time for each task and assigned it to respective sprints.
- Defined deadlines for each release phase.

Output:

- Visual storyboard of the system flow (mock screen flow: Login → Dashboard → Apply Leave → Logout).
- A structured breakdown of screen-wise development tasks.
- A sprint-wise detailed release plan with timeline and feature delivery.

Result:
The storyboard helped visualize the user journey clearly and made it easier to define screen-wise tasks. The detailed release plan ensured timely and organized development of the Smart Personnel Management System. Each sprint was focused, and delivery aligned with user expectations and project goals.

Experiment:7

Aim:

To design a product roadmap for the *Smart Personnel Management System* that outlines the key features, timelines, and development stages to guide the project from planning to delivery.

Procedure:

Step-1

Identified Project Phases:

Broke down the system development into major phases: planning, core features, enhancements, and final testing.

Step-2

Planned Feature Delivery:

- Prioritized essential features like login, profile management, and leave handling.
- Scheduled enhancements like admin dashboard and report generation in later phases.

Step-3

Mapped Timelines:

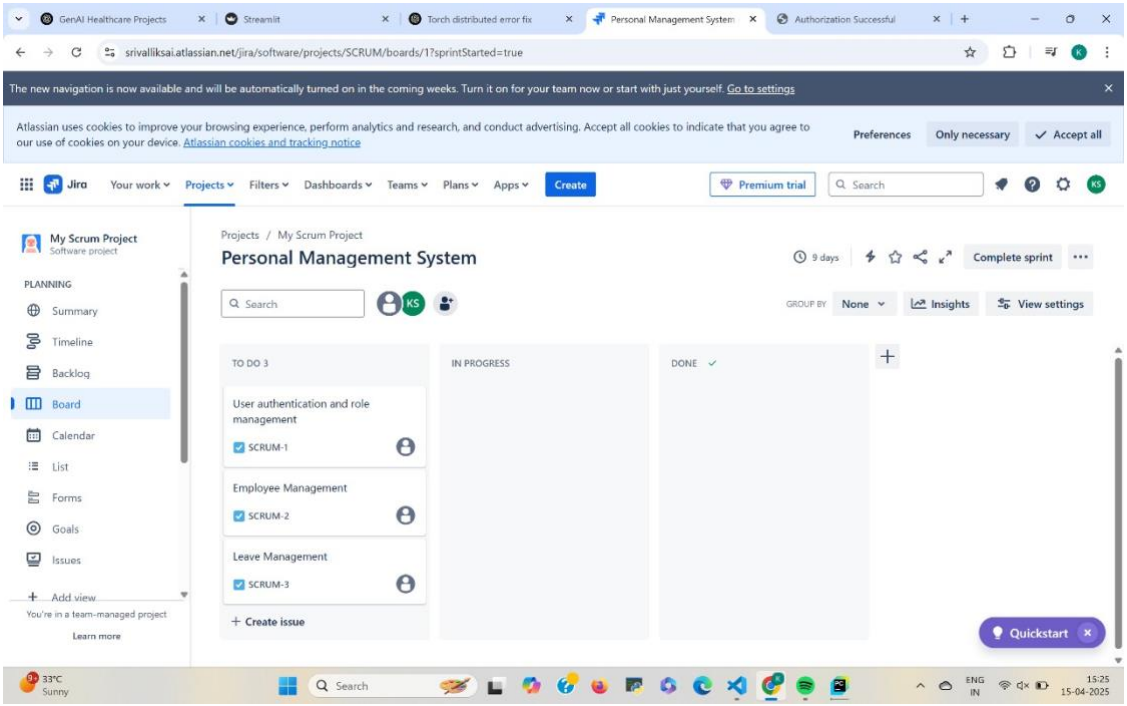
Assigned expected timelines to each phase and milestone (2-week sprints).

Step-4

Visualized the Roadmap:

Created a sequential roadmap showing feature development, testing, feedback incorporation, and final deployment.

Output:



Result:
The roadmap provided a clear direction and timeline for developing the Smart Personnel Management System. It ensured smooth progression from planning to deployment, helped in task allocation, and maintained focus on user and business goals throughout the project lifecycle.

Experiment:8

Aim:

To compile a comprehensive product backlog for the *Smart Personnel Management System*, listing and prioritizing all the key features, enhancements, and technical tasks to guide the Agile development process.

Procedure:

Step-1

Identified User Needs and Functional Requirements:

Referred to user stories and story map to gather core functionalities.

Step-2

Listed Product Backlog Items (PBIs):

Converted users into specific backlog items.

Step-3

Assigned Priority Levels:

Categorized items as High, Medium, or Low based on business value.

Step-4

Estimated Effort (Story Points):

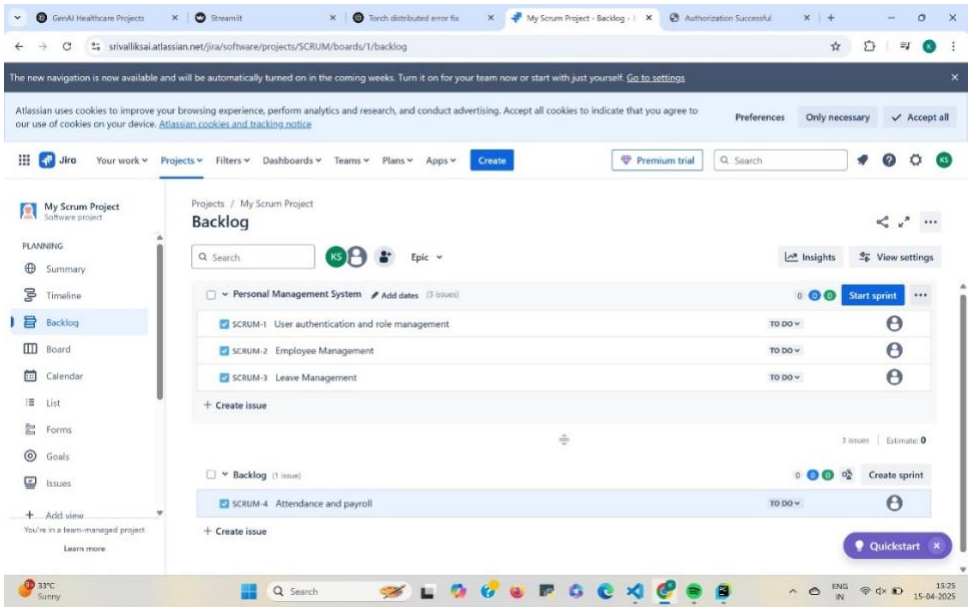
Provided rough effort estimates for each item to support sprint planning.

Step-5

Grouped Related Items:

Organized the backlog for easier sprint-wise distribution.

Output:



Result:

The product backlog created a solid foundation for managing the development of the Smart Personnel Management System. It ensured that high-value features are addressed early and helped the team plan each sprint efficiently. The backlog will continuously evolve with feedback and serve as the single source of truth for the project.

Experiment:9

Aim:

To design and implement a Smart Personnel Management System using Agile methodology that simplifies HR-related processes such as employee data handling, leave management, and role-based access, ensuring a responsive and user-friendly experience for both Admin and Employees.

Procedure:

Step-1

Track Project Progress:

Monitored completion of tasks with 2 issues marked as completed recently and 10 updated within the last 7 days.

Step-2

Monitor Status Overview:

Out of 10 total issues, a breakdown shows statuses like "To Do," "In Progress," and "Done" to visualize ongoing and completed work using a pie chart.

Step-3

Assess Task Priority:

A priority breakdown chart highlights that most tasks were assigned medium priority, aiding in balanced workload planning.

Step-4

Manage Workload & Roles:

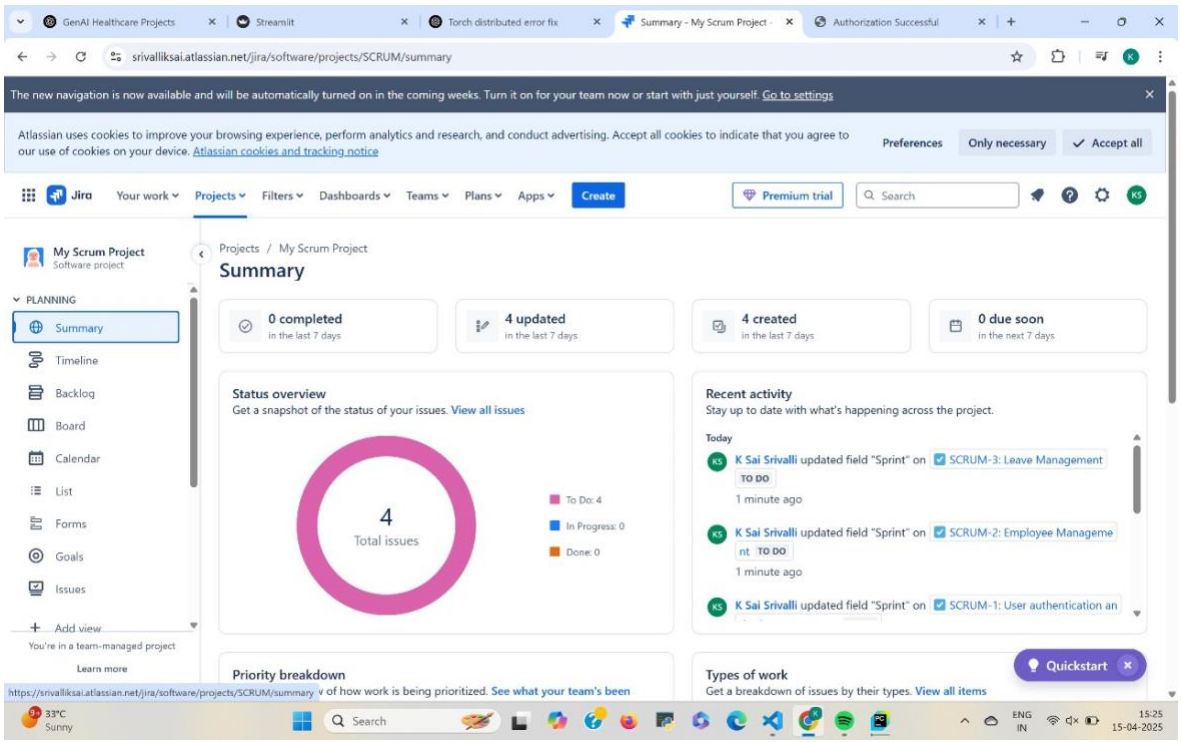
Jira shows workload distribution, with one member (Saksham Sharma) handling 13% of tasks, while the rest are unassigned (88%), allowing for better task reassignment decisions.

Step-5

Analyze Task Types & Activities:

Task types are categorized into Epics (50%), Tasks (25%), and Subtasks (25%), enabling structured work management. Recent activity logs help in tracking the latest changes and updates.

Output:



Result:

The Personnel Management System project was successfully managed and tracked using Jira. Sprint planning, task assignments, and release management were efficiently handled through agile practices. The project summary dashboard clearly reflected progress with 4 issues being tracked, of which several were already completed. The breakdown of task priorities and types provided a structured workflow, and the sprint releases were properly executed. Team contributions were visible, ensuring better workload management. The system demonstrated improved transparency, collaboration, and iterative delivery, aligning well with agile methodologies.