Software Requirements Specification – Employee Leave Management System

Document ID: ASP-CMS-SRS-02.doc

Author(s): PRIYADHARSHINI S

Version: 0.1

Date: March 26, 2024

1. Abstract

The Employee Leave Management System is a comprehensive software solution designed to streamline and automate the leave management process within organizations. This system offers a user-friendly interface for both employees and administrators, facilitating efficient leave requests, approvals, and tracking. Employees can easily submit leave requests and check the status of their applications. Administrators have access to tools for configuring leave policies, managing employee information, generating reports, and overseeing the entire leave workflow. The system ensures accuracy in leave calculations, adherence to company policies, and timely notifications for pending approvals. By digitizing the leave management process, this system enhances productivity, transparency, and employee satisfaction while reducing administrative burden and errors.

2. Objective and Scope

The objective and scope of Employee Leave Management System is to automate and streamline the process of leave requests within an organization, ensuring efficiency, transparency, and accuracy.

The scope of the product includes the following basic features:

- Employees will have the ability to submit leave requests through the system. Different types of leaves such as sick leave, vacation, maternity/paternity leave, etc., will be supported. Employees can specify the dates, duration, and reason for the leave application.

- Supervisors and managers can review, approve and reject leave requests based on predefined rules and policies. The system will maintain a transparent workflow, showing the status of each request.

- Supervisors/Managers able to add, edit or delete the employee belonged to particular department.

- The system will be accessible via web and/or mobile interfaces for convenient usage. It will implement stringent security measures to protect employee data and ensure confidentiality.

- Administrators will have tools to define and manage various leave types, set policies, and oversee the entire leave workflow.

- By centralizing leave management, the system seeks to increase transparency, reduce administrative overhead, and improve overall productivity within the organization.

- Administrators will benefit from comprehensive tools to define leave policies, manage employee records, and generate insightful reports for better decision-making.

3. Project End Users

The project end users are the individuals or groups who will interact with and benefit from the Employee leave management System.

Employees: They are the primary users of the system, responsible for submitting leave• applications with necessary details such as leave type, duration, and reason. Viewing the status of their leave requests. Receiving notifications and reminders regarding leave approvals, rejections, or modifications.

Supervisors / Managers: They are responsible for managing their team's leave requests,• ensuring adequate coverage, and approving or rejecting leave applications. Planning work schedules to accommodate approved leaves and maintain productivity. Receiving notifications and reminders for pending leave requests requiring approval.

4. Features

4.1Login to the system

Each and every user should be authenticated with a User Name and Password to login into the system.

Validations for User Name and Password.

User Name: It accepts only Alphabets, Numbers, Dot (.) symbol and Underscore (_) symbol.

Password: It can be anything of the users’ choice.

4.2Add/Update/Delete Employee

This feature provides tools for supervisors to manage employee profiles, information, and interactions within the system. Add new employees to the system and set up their profiles with relevant details. Update employee information such as job titles and departments. Delete the employee.

4.3Manage Leave Type

This feature enables administrators to define and customize different types of leave such as vacation leave, sick leave, maternity/paternity leave, etc.

4.4Manage Leave Application

This feature provides administrators with tools to oversee and manage all leave applications submitted by employees. Review pending leave requests and take action (approve, reject, modify) as necessary. Monitor leave balances and ensure compliance with company policies.

4.5Apply for Leave

This feature enables employees to request time off for various reasons such as vacation, sick leave, personal reasons, etc. Fill out a leave request form with details such as leave type, duration, start date, end date, and reason. Submit the leave request for approval by the relevant manager or supervisor.

5. Functional and Non-Functional Requirements

5.1Functional Requirements

Login: Users must be login to the system by using their username and password.

Employee Management: The admin should be able to add, edit, and delete employee profiles. Assign roles and permissions to users, such as manager, HR, or regular employee.

Leave Policy Management: Define different types of leaves (e.g., sick leave, vacation, maternity/paternity leave, etc.).

Leave Application Processing: View all leave requests in a centralized dashboard. Approve or reject leave requests submitted by employees.

Apply for Leave: Employee can able to request for leave be specifying the type of leave, duration with description.

5.2Non-Functional Requirements

Performance: The system should respond to user actions promptly, with minimal latency.

Scalability: The system should be scalable to accommodate an increasing number of employees and leave requests.

Availability: The system should be available 24/7, especially for global organizations with employees in different time zones.

Security: The system should adhere to industry-standard security practices to protect sensitive employee data.

Usability: The user interface should be intuitive and user-friendly, requiring minimal training for admins.

6. High Level Design

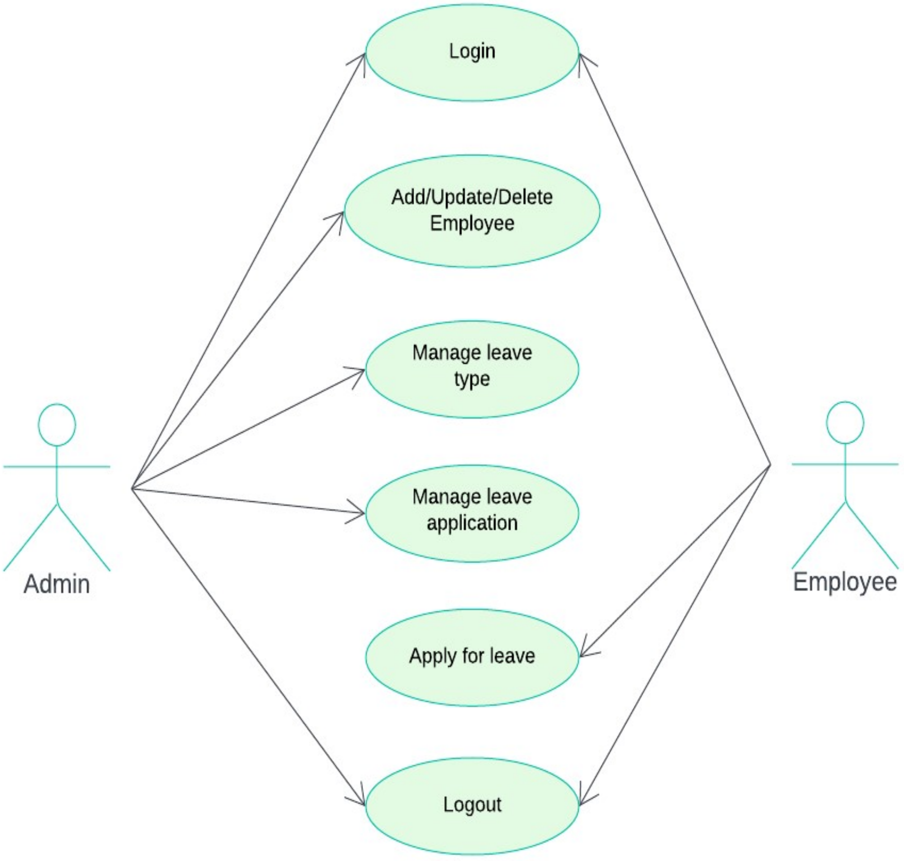
The high-level design of the Employee Leave Management System includes defining user roles (Admin, Employee), modules (Authentication, Leave Requests, Approval Workflow, Leave Types), and a database schema (Users, Employees, Leave Applications). It outlines a user-friendly interface for employees to apply for leave and check statuses. The frontend development incorporates frameworks such as HTML, CSS, React.js, Angular.js, offering a dynamic and intuitive user interface for seamless interaction. For the backend, a resilient technology stack such as Node.js, Python Django, or Ruby is employed to develop RESTful APIs, enabling efficient communication with the frontend interface. Data persistence is managed using a reliable relational database management system (RDBMS) such as PostgreSQL or MySQL, featuring a well-structured schema encompassing tables for employee profiles, leave types, leave applications and approvals. Admins can manage employees, leave types, and approval workflows. Security measures include role-based access control and encrypted data transmission. The system architecture follows a modular design pattern for scalability and easy maintenance.

7. Low Level Design

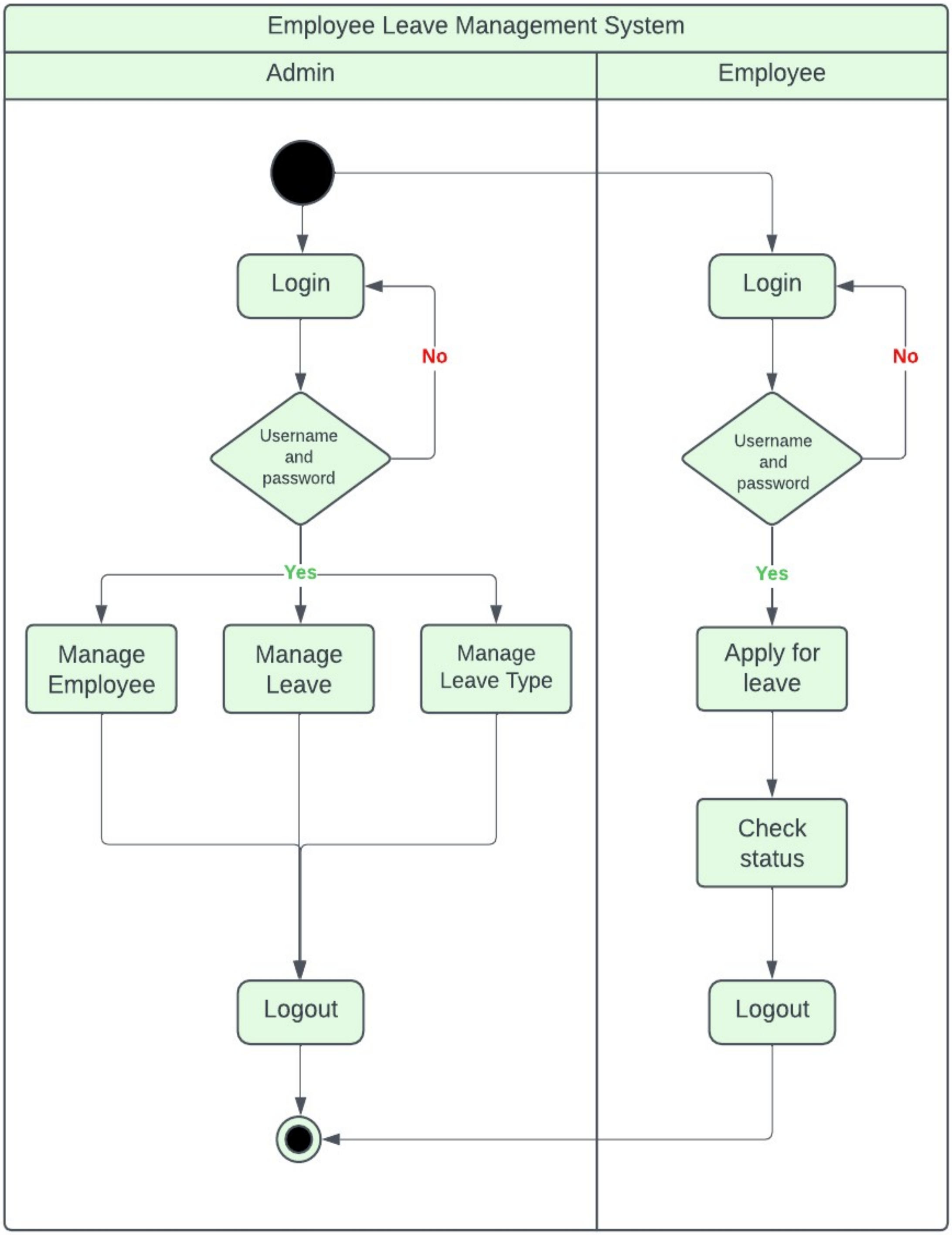
The low-level design of the Employee Leave Management System delves into the detailed implementation of the system's components defined in the high-level design. It includes class diagrams, sequence diagrams, database schema, and algorithm descriptions. Middleware functions handle crucial tasks such as authentication, authorization, and error handling, ensuring smooth database interactions and data integrity. Each module, such as user authentication, leave application processing, and database interactions, is broken down into specific functions, methods, and their interactions. This phase focuses on coding specifics, like naming conventions, error handling, and data validation. The database schema represents entities like employees, leave types, and leave applications, employing normalized tables and establishing appropriate relationships. It also outlines how each module communicates with others and the data flow between them. The low-level design provides the development team with a clear roadmap for coding and ensures a cohesive and efficient system implementation.

8. Diagrams

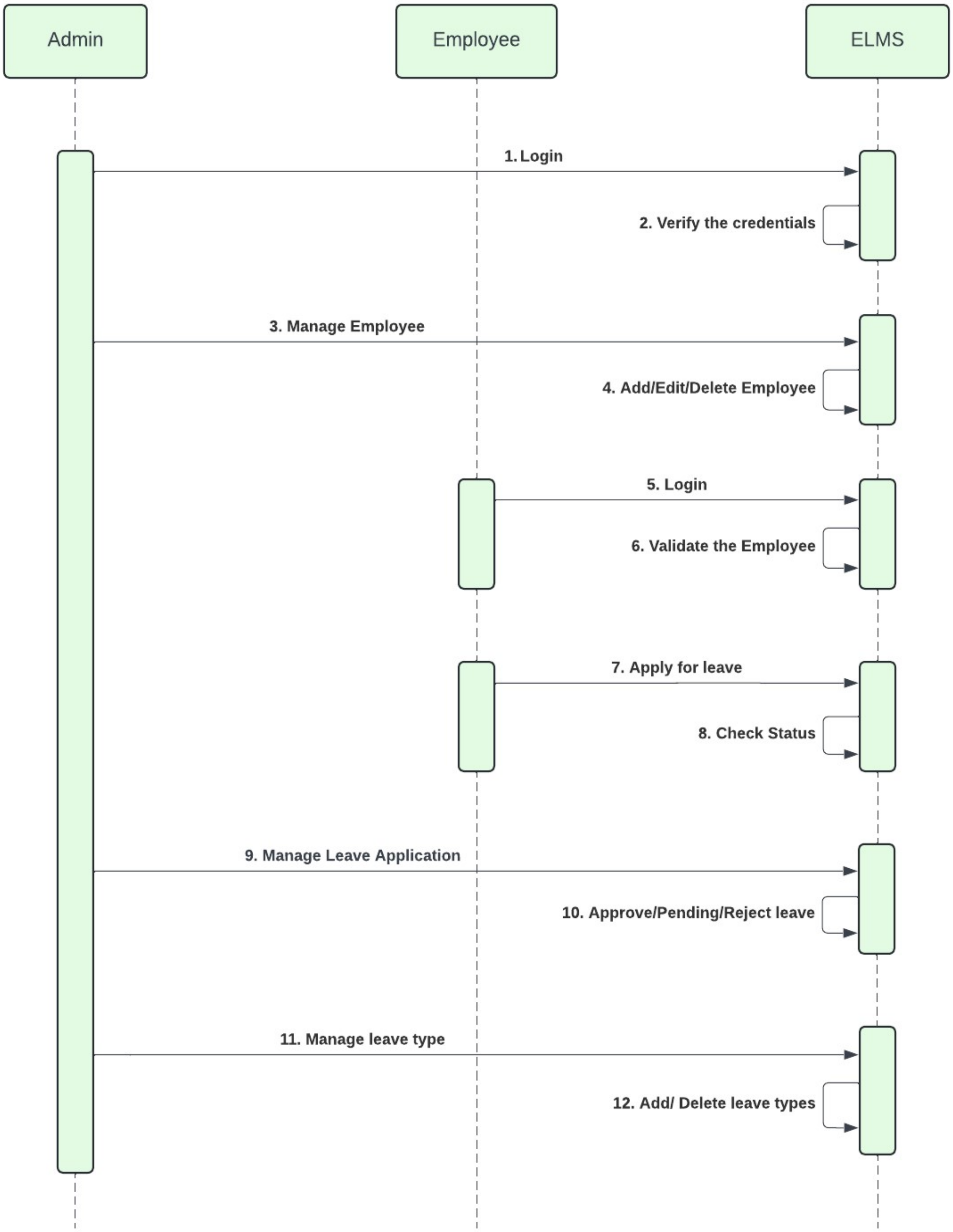
8.1Use Case Diagram



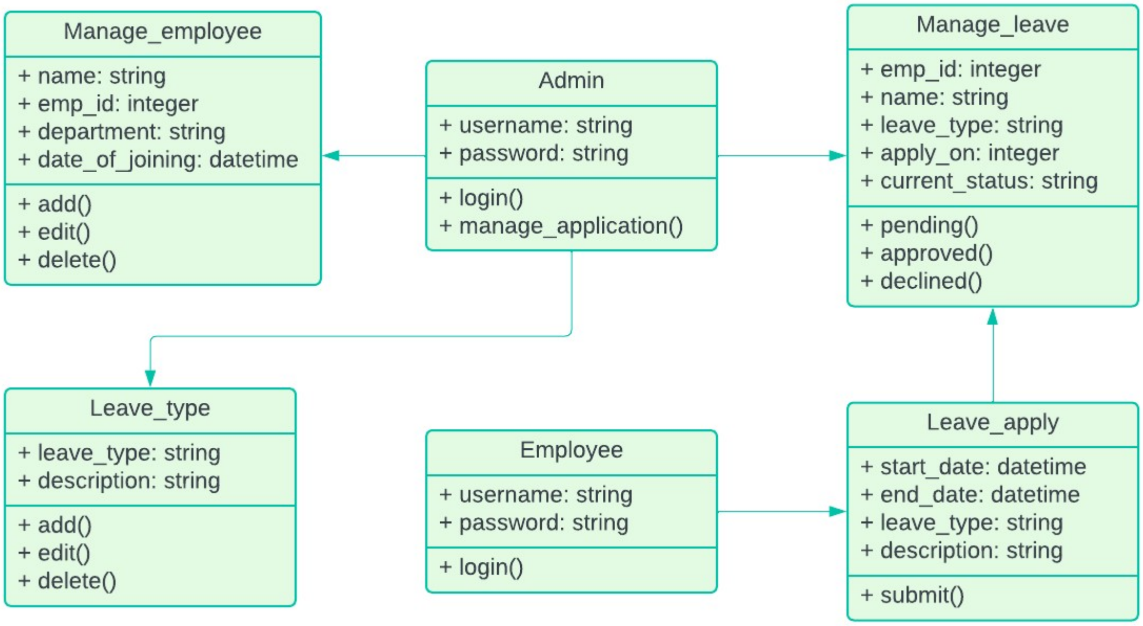
8.2Flow Chart



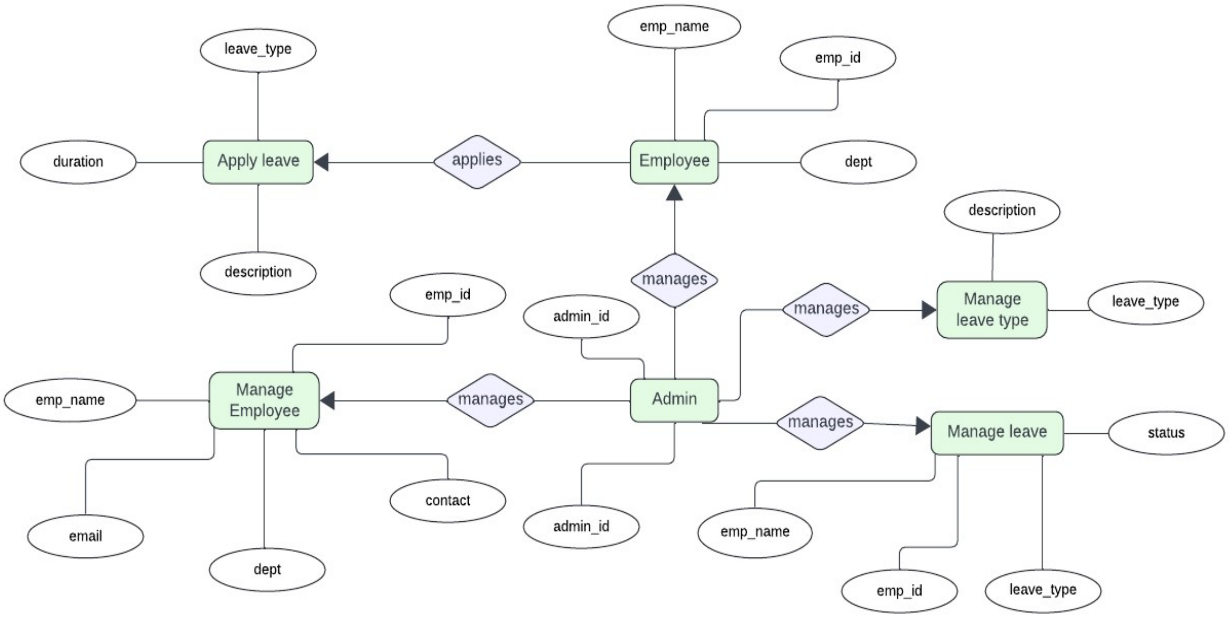
8.3Sequence Diagram


Admin
Employee
ELMS
1. Login
2. Verify the credentials
3. Manage Employee
4. Add/Edit/Delete Employee
5. Login
6. Validate the Employee
7. Apply for leave
8. Check Status
9. Manage Leave Application
10. Approve/Pending/Reject leave
11. Manage leave type
12. Add/ Delete leave types

8.4Class Diagram



9. ER Diagram



10. Test Cases

The Test Plan is derived from the Requirements, Functional Specifications and Detailed Design Specification.

Test Case	Test Purpose	Test Condition	Expected Outcome	Actual Result
User login	To ensure users can log in successfully.	User provides valid login credentials (username/email and password).	Successful login with valid credentials.	User is successfully authenticated and directed to the main dashboard.
Leave Submission	To verify employees can submit leave applications.	Valid leave type selected. Start date and end date within allowed range. Reason provided.	Leave application successfully submitted.	Employee can successfully submit their leave application and view the status.
Manage Leave	To ensure managers can approve or reject leave applications.	Access to pending leaves requests. Approval or rejection action taken.	Approved or Rejected leave applications reflect in employee's status.	Supervisor/ Manager able to manage leave successfully.

Manage Employee	To confirm administrators can manage employee records.	Add a new employee with valid details. Edit existing employee information. Delete an employee record.	Successful addition of new employee. Updated employee information reflects changes. Employee record removed from the system.	Supervisor/ Manager able to manage employee successfully.
Manage leave type	To ensure administrators can configure leave policies.	Define new leave types and also edit existing leave types.	New leave types visible for employees.	Supervisor/ Manager able to manage leave types successfully.