SOFTWARE REQUIREMENTS SPECIFICATION

**For**

**Employee Leave Management System**

**Prepared by:-**

*Bhoopesh AG  
Kavin Kannan T  
Pious K George*

# Introduction

## Purpose

The main purpose of our Employee Leave Management System is to improve and expedite the leave management procedure inside businesses. We want to offer a safe, convenient way for people to submit, approve, and monitor their leave requests. We want to do this by utilizing Java and DBMS. Through thorough summaries of leave history and real-time updates on leave balance, this system places a high priority on openness. While Notifications educate stakeholders, the dynamic Working Calendar guarantees accuracy in computations. Accuracy and flexibility are provided via customization choices and False Data Entry Correction tools. In order to serve a varied workforce, we aim for inclusion via Multi-Language Support. Through a feedback and issue reporting structure, the project also promotes ongoing progress. Our ultimate objective is to include a Switch Feature that will enable workers to easily make decisions in response to the changing nature of the workplace.

## Document Conventions

* + - Entire document should be justified.
    - Convention for Main title

Font face: Times New Roman Font style: Bold

Font Size: 14

* + - Convention for Sub title

Font face: Times New Roman Font style: Bold

Font Size: 12

* + - Convention for body

Font face: Times New Roman Font Size: 12

## Scope of Development Project

The Employee Leave Management System makes an effort to increase organizational effectiveness by utilizing necessary techniques. Starting with a secure User Authentication and Access Control system, the project safeguards permitted access to sensitive data. The main element is the Leave Request and Approval System, which makes filing for employees easier and enables supervisors to review requests quickly. Real-time leave balance tracking provides employees with quick access to information about accrued, spent, and unused leave. A program called Comprehensive Leave History and Summary helps with the examination of attendance patterns.

The system has a strong notification system for changes to stakeholders and a dynamic, customized working calendar for the office or firm. Enhancing system flexibility are customization and False Data Entry Correction tools. Diverse language preferences are accommodated via multilingual support. Tools for reporting issues and receiving feedback guarantee both user pleasure and ongoing progress. The Switch Feature facilitates seamless transitions between office and remote work, in line with the trends of flexible work. With a focus on security, scalability, performance optimization, and thorough documentation, the project intends to produce an effective and user-friendly employee leave management system.

## Definitions, Acronyms and Abbreviations

JAVA -> platform independence SQL-> Structured query Language ER-> Entity Relationship

UML -> Unified Modeling Language

IDE-> Integrated Development Environment SRS-> Software Requirement Specification

## References

* + - Books
* Leave Management System: A Comprehensive Guide" by Jagan Mohan Reddy.
* 
  + - Websites

[**http://www.slideshare.net/**](http://www.slideshare.net/)

[**http://ebookily.net/doc/srs-library-management-system**](http://ebookily.net/doc/srs-library-management-system)

# Overall Descriptions

## Product Perspective

Use Case Diagram of Library Management System

*searches*

1

1 *requests*

1

1

1..\*

\*

search\_book



1..\*

check\_limit

check\_availability

User 1

issue\_book

*request\_renew*

<<include>>

*monitors\_request*

1

*monitors\_renew* 1

1

*performs*

*give\_book*

<<include>>

0..\*

1..\*

renew\_book

verify\_member

<<include>>

*take\_book*

1

1 Librarian

Student

0..\*

1..\*

*adds\_new\_book*

*perform\_transaction\_updation*

Staff

\*

\*

return\_book

View\_logs

<<extend>>

add\_book

\*

calculate\_fine

update\_record

This is a broad level diagram of the project showing a basic overview. The users can be either staff or student.. This System will provide a search functionality to facilitate the search of resources. This search will be based on various categories viz. book name or the ISBN. Further the library staff personnel can add/update the resources and the resource users from the

## Product Function

Entity Relationship Diagram of Library Management System



The Online Library System provides online real time information about the books available in the Library and the user information. The main purpose of this project is to reduce the manual work. This software is capable of managing Book Issues, Returns, Calculating/Managing Fine, Generating various Reports for Record-Keeping according to end user requirements. The Librarian will act as the administrator to control members and manage books. The member’s status of issue/return is maintained in the library database. The member’s details can be fetched by the librarian from the database as and when required. The valid members are also allowed to view their account information.

## User Classes and Characteristics

In the context of an Employee Leave Management System, there are several user classes with distinct characteristics. These classes represent the different roles and interactions within the system:

Administrator:

* Has full system access and control.
* Manages user accounts and permissions.
* Generates reports and analytics.
* Resolves system issues and manages configurations.

Supervisor/Manager:

* Approves or denies leave requests.
* Monitors team leave balances and history.
* Generates team-specific reports.
* Communicates leave policies and updates to the team.

Employee/User:

* Submits leave requests.
* Checks leave balances and history.
* Receives notifications on leave request status.
* Edits personal information and preferences.

Human Resources (HR) Representative:

* Manages company-wide leave policies.
* Assists in dispute resolutions related to leaves.
* Conducts training on system usage.
* Generates organization-wide reports.

System Auditor:

* + - Monitors system usage for compliance.
    - Conducts regular system audits.
    - Ensures data integrity and security.
    - Generates audit reports for review.

## Operating Environment

The operating environment for the Employee Leave Management System is designed to be versatile, accessible, and secure. The system operates seamlessly on Java-based platforms, ensuring cross-compatibility and ease of deployment across various operating systems, including Windows, Linux, and macOS. Leveraging a robust Database Management System (DBMS), such as MySQL, ensures efficient data storage and retrieval. The system is web-based, promoting accessibility from different browsers like Chrome, Firefox, and Safari, facilitating user interaction without the need for specific software installations. Security protocols are implemented to safeguard sensitive employee information, adhering to industry standards for data protection. Additionally, the system is optimized for mobile responsiveness, enabling users to access and manage leave-related tasks conveniently from their smartphones or tablets. Integration capabilities with existing HR and payroll systems are considered, fostering a cohesive organizational infrastructure. Regular monitoring and updates are part of the system's environment, ensuring stability, performance, and the incorporation of new features aligned with evolving organizational needs. Lastly, cloud compatibility is embraced, allowing for scalable and efficient deployment, with considerations for potential future migrations to cloud-based infrastructures.

## Assumptions and Dependencies

The successful implementation of the Employee Leave Management System relies on several key assumptions and dependencies. It is assumed that the organization's infrastructure, including servers and network connectivity, is reliable and secure. Additionally, the project assumes that users have access to up-to-date web browsers and devices compatible with the system. Dependencies include timely availability of necessary APIs and libraries, adherence to industry-standard security protocols, and the organization's commitment to providing accurate leave policy information. The project also relies on the availability of skilled personnel for system deployment and ongoing maintenance. Furthermore, it assumes a collaborative effort between the development team and HR representatives for accurate policy integration. Any changes in organizational policies or infrastructure may impact project timelines and require agile adjustments. Regular communication and cooperation between the development team and end-users are assumed for effective feedback loops and issue resolution.

## Requirement

Software Configuration:-

This software package is developed using java as front end which is supported by sun micro system. Microsoft SQL Server as the back end to store the database.

Operating System: Windows NT, windows 98, Windows XP Language: Java Runtime Environment, Net beans 7.0.1 (front end) Database: MS SQL Server (back end)

Hardware Configuration:- Processor: Pentium(R)Dual-core CPU Hard Disk: 40GB

RAM: 256 MB or more

## Data Requirement

The inputs consist of the query to the database and the output consists of the solutions for the query. The output also includes the user receiving the details of their accounts. In this project the inputs will be the queries as fired by the users like create an account, selecting books and putting into account. Now the output will be visible when the user requests the server to get details of their account in the form of time, date and which books are currently in the account.

# External Interface Requirement

## GUI

1. User Authentication Interface:

* Description: The system will feature a secure login interface for employees, supervisors, and administrators.
* Requirements:
* Input fields for username and password.
* "Forgot Password" functionality with a secure reset process.
* Multi-factor authentication support.

2. Leave Request Submission Interface:

* Description: Users should have an intuitive interface to submit leave requests.
* Requirements:
* Calendar-based input for selecting leave start and end dates.
* Dropdown menus for leave type selection.
* Text area for adding leave reasons.

3. Leave Approval Interface:

* Description: Supervisors and administrators need an interface to review and approve/deny leave requests.
* Requirements:
* Clear display of employee leave details.
* Buttons for approval and denial actions.
* Comment section for providing feedback.

4. Leave Balance and History Interface:

* Description: Employees and administrators should easily access leave balances and history.
* Requirements:
* Display of accrued, taken, and remaining leave.
* Detailed historical view with filtering options.

5. Working Calendar Configuration Interface:

* Description: Administrators should be able to configure the working calendar.
* Requirements:
* Calendar interface to mark official working days and holidays.
* Customization options for working hours.

6. Notification Interface:

* Description: Users should receive notifications for leave request status and system updates.
* Requirements:
* In-app notifications.
* Email notifications with customizable preferences.

7. Customization and Settings Interface:

* Description: Administrators should have an interface for system customization.
* Requirements
* Configurable leave policies.
* Options for system theme and language settings.

8. Feedback and Issue Reporting Interface:

* Description: Users should be able to provide feedback and report system issues.
* Requirements:
* Feedback form with categories.
* User-friendly issue reporting with screenshot attachments.

9. Switch Feature Interface:

* + - Description: Employees should have an interface to indicate their work location preference.
    - Requirements:
    - Simple toggle switch for office or remote work.

10. Multi-Language Support Interface:

* Description: The system should support multiple languages for a diverse user base.
* Requirements:
* Language selection dropdown in the user profile.
* Translatable interface elements.

# System Features

The Employee Leave Management System boasts a robust set of features to streamline leave-related processes. Users can efficiently submit and track leave requests, while supervisors seamlessly review and approve them. The system provides real-time leave balance updates and detailed historical summaries. A dynamic working calendar reflects official working days and holidays, and a notification system keeps stakeholders informed. Customization options, multi-language support, and a switch feature for flexible work arrangements enhance the system's adaptability and user experience.

# Other Non-functional Requirements

## Performance Requirement

ELMS is engineered to handle concurrent user interactions with efficiency, ensuring prompt response times for leave requests and data retrieval to sustain a seamless and responsive user experience.

## Safety Requirement

Security is paramount in ELMS, and robust measures are in place to protect sensitive employee data and prevent unauthorized access, instilling confidence in the secure management of information.

## Security Requirement

Security features within ELMS include advanced user authentication, data encryption, and role-based access control, providing a fortified defense against potential breaches and aligning with industry standards.

## Requirement attributes

Each requirement within ELMS is assigned attributes such as priority, status, and target implementation dates, ensuring a systematic and organized approach to development.

## Business Rules

Defined rules within ELMS govern the application of business processes, enhancing the system's adaptability to the unique needs and regulations of the organization.

## User Requirement

The detailed capturing of user requirements within ELMS ensures the end-users' needs and expectations are not only met but exceeded, creating a user-centric and engaging experience

# Other Requirements

## Data and Category Requirement

The specification of data storage requirements and data categorization within ELMS ensures an organized and efficient approach to information retrieval.Similarly there will be different categories of books available. According to the categories of employees their relevant data should be displayed. The categories and the data related to each category should be coded in the particular format.

## Appendix

This section includes additional information, diagrams, and charts that complement the main content, providing stakeholders with a deeper understanding of the intricacies of ELMS.

## Glossary

The glossary within ELMS serves as a comprehensive reference, providing stakeholders with a common understanding of terms used throughout the document and the system.

This detailed document acts as a guiding beacon for the development and comprehension of the Employee Leave Management System, ensuring stakeholders are equipped with the necessary insights to navigate the intricacies of the system with confidence and clarity. Regular updates and reviews are recommended to maintain alignment with project goals and stakeholder expectations.

## Class Diagram

A class is an abstract, user-defined description of a type of data. It identifies the attributes of the data and the operations that can be performed on instances (i.e. objects) of the data. A class of data has a name, a set of attributes that describes its characteristics, and a set of operations that can be performed on the objects of that class. The classes’ structure and their relationships to each other frozen in tim0e represent the static model. In this project there are certain main classes

which are related to other classes required for their working. There are different kinds of relationships between the classes as shown in the diagram like normal association, aggregation, and generalization. The relationships are depicted using a role name and multiplicities. Here ‘employees’, ‘Member’ and ‘leave percentage’ are the most important classes which are related to other classes.

