

High Level Design(HLD)

Leave Management System Web Application

Revision Number : 1.0

Last Date of Revision : 30/07/2022

Contents

Document version control	3
Abstract	4
1 Introduction	5
1.1 Why this High-Level-Design Document?	5
1.2 Scope	5
1.3 Definitions	5
2 General Description	6
2.1 Product Perspective	6
2.2 Problem Statement	6
2.3 Proposed Solution	6
2.4 Tool Used	7
2.5 Constraints	7
2.6 Assumptions	8
3 Design Details	8
3.1 Process Flow	8
3.2 Event log	8
3.3 Error Handling	8
4 Performance	9
4.1 Reusability	9
4.2 Application Compatibility	9
4.3 Resource Utilisation	9
4.4 Deployment	9
5 KPIS (Key Performance Indicators)	10
6 Conclusion	10
7 References	10

Document Version Control

Date Issued	Version	Description	Author
30/07/2022	1	HLD-Ver1.0	Billa Mounika

Abstract

Every organization has a challenge to manage their employees. If it is not done properly, some problems may happen to deliver projects/work. That's why, leave management system is necessary for an organization. This work discusses the leave management system web application. Here employees can apply for leave and supervisors can manage those leave easily.

1 Introduction

1.1 Why this High Level Design Document

The purpose of this High-Level-Design(HLD) Document is to add necessary details to the current project description to represent a suitable model for coding. This document is also intended to help detect contradictions prior to coding, and can be used as a reference manual for how the modules interact at a high level.

The HLD will:

- Present all of the design aspects and define them in a detail
- Describe the user interface being implemented
- Includes design features and architecture of the project
- List and describe the non functional attribute like:
 - Reliability
 - Maintainability
 - Portability
 - Reusability
 - Application compatibility
 - Serviceability

1.2 Scope

The HLD documentation presents the structure of the system, such as the database architecture, application architecture(layers), application flow(navigation), and technology architecture. The HLD uses non-technical to mildly-technical terms which should be understandable to the administrator of the system.

1.3 Definitions

Term	Definition
Database	Collection of all the information uploaded by the system or used by the system
IDE	Integrated Development Environment
AWS	Amazon Web Services

2 General Description

2.1 Product Perspective

Leave management system is a python-based application where employees can apply for leave and supervisors can manage those leave easily.

2.2 Problem Statement

To create a web application for leave management system of corporate office and to implement the following feature:

- Supervisors can add, remove, update details of employee
- Every employee should be credited two leave by default each month
- Employees can apply for half or full day leave
- Leave types should be available for employees to choose from
- Each applied leave should be approved by the supervisors
- Automatic approval should be provided on applied leave, it is not disapproved by the supervisors before 24 hours of the leave date
- After every year if employees have more than 22 leave left, all leave above 22 days should be converted into money and employee should be able to claim their money into bank account

2.3 Proposed Solution

The solution proposed here is a leave management system that can be implemented to perform the above-mentioned use cases. Here employees can apply for leave and then notification will be sent to supervisors, he/she can take action on that. Also supervisors can edit, delete, and update the details of the employees. There will be an option for employees where they can claim the equivalent money if they have more than 22 leave left.

2.4 Tool used



- Python3.7 is used as programming language
- Flask2.0 is used as backend development
- Pycharm is used as IDE
- Front end development is done by using HTML/CSS/JS/Bootstrap5.2
- Cassandra is used as a database where all information will be stored
- Github is used as version control
- AWS is used to deploy the application

2.5 Constraints

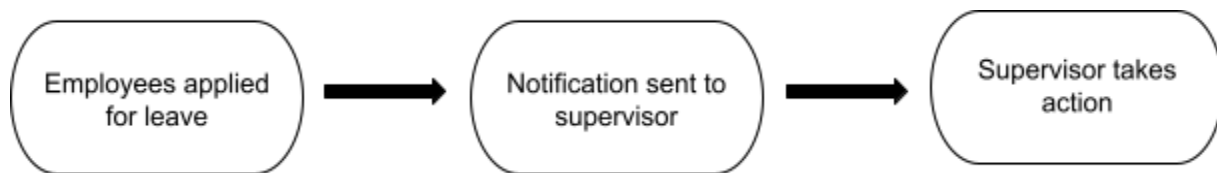
The Leave Management System must be user-friendly, as automated as possible, and the user should not be required to know any other work.

2.6 Assumption

The main objective of this project is to implement the use cases as previously mentioned(2.2 Problem Statement) as fast as possible, so that organizations can manage their employees. It is also assumed that all aspects of the project have the ability to work together in the way the organization is expecting.

3 Design Details

3.1 Process Flow



3.2 Event Log

The system should log every event so that the user will know what process is running internally.

Initial Step-By-Step Description

1. The system identifies at what step logging required
2. The system should be able to log each and every system flow
3. Developers can choose a logging method. You can choose database logging or file logging as well
4. System should not hang even after so many loggins. Logging just because we can debug issues, so logging is mandatory.

3.3 Error Handling

Should errors be encountered, an explanation will be displayed as to what went wrong? An error will be defined as anything that falls outside the normal and intended usage.

4 Performance

This leave management system helps organizations to manage their employees so that they don't have any problem to deliver any projects/work on time. Also here all the details of the employees will be stored securely and they can extract the details at any time.

4.1 Reusability

The code written should have the ability to be reused with no problems.

4.2 Application Compatibility

The different components for this project will be using python as an interface between them. Each component will have its own task to perform, and it is the job of the python to ensure the proper transformation of information.

4.3 Resource Utilisation

When any task is performed, it will likely use all the processing power available until that function is finished.

4.4 Deployment



5 KPIS (Key Performance Indicators)

1. Employees can apply for leave easily
2. Supervisors can manage the employees properly
3. Details of each employee is stored securely
4. Delete, edit of employees's information can be done quickly
5. Everyone can claim the money equivalent to leave left securely

6 Conclusion

This system optimizes the employee management in an organization. It helps both employees and the organization to fulfill their needs. This way an organization can save both time and cost.

7 References

1. <https://docs.python.org/3/>
2. <https://getbootstrap.com/>