|  |  |
| --- | --- |
|  | **Employee Management System Application Development** |
|  |  |
|  | ELF / HTD Training Programs  Java Full Stack - Use Cases |

|  |  |
| --- | --- |
|  | Software Solutions |

 

Table of Contents

[1. INTRODUCTION 2](#_Toc27565091)

[2. PROBLEM STATEMENT 2](#_Toc27565092)

[3. IMPLEMENTATION 2](#_Toc27565093)

[4. Database Specifications 3](#_Toc27565094)

# 1. INTRODUCTION

This document outlines a Project. The project is to develop online EMPLOYEE MANAGEMENT SYSTEM (EMS). This document contains the work flow of the system and gives guidelines on how to build the functionality gradually in each of the course modules.

# 2. PROBLEM STATEMENT

The problem definition for designing the system is to maintain the data of employee, to make easy controlling of employees and their leaves.

This project is aimed at developing Employee Maintenance System. This is an Intranet based application that can be accessed throughout the organization This system can be used to search for Employees based on search condition, add individual employee, modify an existing employee details and display all employee details across locations within an organization. This is an integrated system that contains both the user (Employee) component and the Admin component

# 3. IMPLEMENTATION

Following is a list of functionalities of the system. Wherever, the description of functionality is not adequate, you can make appropriate assumptions and proceed. There are two categories of people who would access the system viz. employee & Admin. Each one of them would have some exclusive privileges (for e.g. Employees can just search an employee details based on search condition)

1. Admin should be able to
   * Login to the system using his/her credentials
   * Add individual employee details by accepting all the field values from end user as listed below and inject the values into database table if data are valid else display an appropriate error messages.
     + - Employee ID
       - First and Last name should be alphabets only (Cannot be empty)
       - Date of joining and Date of birth need to undergo regular date validation (DD-MMM-YY) - (Cannot be empty)
       - The date of joining should be > Date of birth and the employee’s age has to be >=18 and <= 58 as an eligibility criterion for employment
       - Department name values has to be populated from database
       - Password (Cannot be empty)

* Display all employee details
* Approve or reject leaves from the employees

1. Employees/User should be able to
   * Login to the system using his/her credentials
   * Employees should be able to apply for a leave using from date and to date with these conditions
     + - From date must be greater than today’s date
       - To date must greater than or equal to from date

# 4. Database Specifications

User\_Info

|  |  |
| --- | --- |
| User\_id | int (primary key)(auto generate) |
| First\_name | Varchar(255) |
| Last\_name | Varchar(255) |
| Password | Varchar(255) |
| User\_Type | Varchar(255) |

User\_Other\_Info

|  |  |
| --- | --- |
| id | int (primary key)(auto generate) |
| User\_id | int (foreign key) (unique) |
| Department\_name | Varchar(255) |
| Joinning\_Date | Date |
| Date\_Of\_Birth | Date |

Leave\_Status\_Info

|  |  |
| --- | --- |
| id | int (primary key)(auto generate) |
| User\_id | int (foreign key) |
| From\_Date | Date |
| To\_Date | Date |
| Leave\_Status | Varchar(255) |