SWE Challenge – Employee Salary Management (ESM) Web Application

# Introduction

Employee Salary Management (ESM) Web Application is to help the HR department in order to manage the employees’ salaries. Using this application, an HR can carry out the following actions:

1. Upload all employees’ data using CSV file
2. View all employee records in the dashboard
3. Add new employee record
4. Edit an employee record
5. Delete an employee record
6. Search an employee record

# Assumptions

Below are the list of assumptions made in order to proceed with the web application development.

1) The is no support for parallel employee file upload

2) The maximum salary of an employee is assumed to be not more than SGD 50000.00 as this is the maximum salary cap used to fetch the records from database for Dashboard screen. However, this amount will not be considered as part of data inserting or updating.

For example, while uploading employees’ data CSV file, adding new employee data and updating existing employee data, there is no validation added to salary amount. However, maximum salary limit of SGD 50000 has been included for the employee records displayed in the Dashboard screen.

If an HR wants to view records of those employee’s whose salary may possibly be above SGD 50000, then it can be done by using the Search Employee feature.

3) Mention of user and employee means one and the same.

4) The CSV file used to upload employees’ data will have the first line as column header and following lines with employee records only. File cannot have a footer line as well unless it’s a comment starting with # in which case the line will be ignored from processing.

5) The application is configured to only run in local machine.

# Application Features

Below are the list of features that is available in ESM web application.

## Upload Employees Data

This functionality can be used to choose and upload a CSV file which consist of employees’ data.

|  |  |  |
| --- | --- | --- |
| API Details | | |
| Endpoint URL | POST http://localhost:8080/users/upload | |
| Request parameters | Parameter | Data Type |
| file | File |
| Response parameters | responseCode | String |
| message | String |
| Sample Backend Request and Response (Success scenario) | | |
|  | | |
| Sample Backend Request and Response (Error scenario) | | |
|  | | |

## Add Employee Data

This functionality can be used to add an employee record to the database

|  |  |  |
| --- | --- | --- |
| API Details | | |
| Endpoint URL | POST http://localhost:8080/users/e10011 | |
| Request parameters | Parameter | Data Type |
| id | String |
|  | login | String |
|  | name | String |
|  | salary | Float |
| Response parameters | responseCode | String |
| message | String |
| Sample Backend Request and Response (Success scenario) | | |
|  | | |
| Sample Backend Request and Response (Error scenario) | | |
|  | | |

## Update Employee Data

This functionality can be used to update an existing employee data.

|  |  |  |
| --- | --- | --- |
| API Details | | |
| Endpoint URL | PATCH http://localhost:8080/users/e10011 | |
| Request parameters | Parameter | Data Type |
| id | String |
|  | login | String |
|  | name | String |
|  | salary | Float |
| Response parameters | responseCode | String |
| message | String |
| Sample Backend Request and Response (Success scenario) | | |
|  | | |
| Sample Backend Request and Response (Error scenario) | | |
|  | | |

## Delete Employee Data

This functionality can be used to delete an existing employee data by id.

|  |  |  |
| --- | --- | --- |
| API Details | | |
| Endpoint URL | DELETE http://localhost:8080/users/e10011 | |
| Path parameters | Parameter | Data Type |
| id | String |
| Response parameters | responseCode | String |
| message | String |
| Sample Backend Request and Response (Success scenario) | | |
|  | | |
| Sample Backend Request and Response (Error scenario) | | |
|  | | |

## Search Employee Data

This functionality can be used to delete an existing employee data by id.

|  |  |  |
| --- | --- | --- |
| API Details | | |
| Endpoint URL | GET http://localhost:8080/users/e10011 | |
| Path parameters | Parameter | Data Type |
| id | String |
| Response parameters | responseCode | String |
| message | String |
| totalElements | int |
| results | Employee |
| empDataRecord | List< Employee> |
| Sample Backend Request and Response (Success scenario) | | |
|  | | |
| Sample Backend Request and Response (Error scenario) | | |
|  | | |

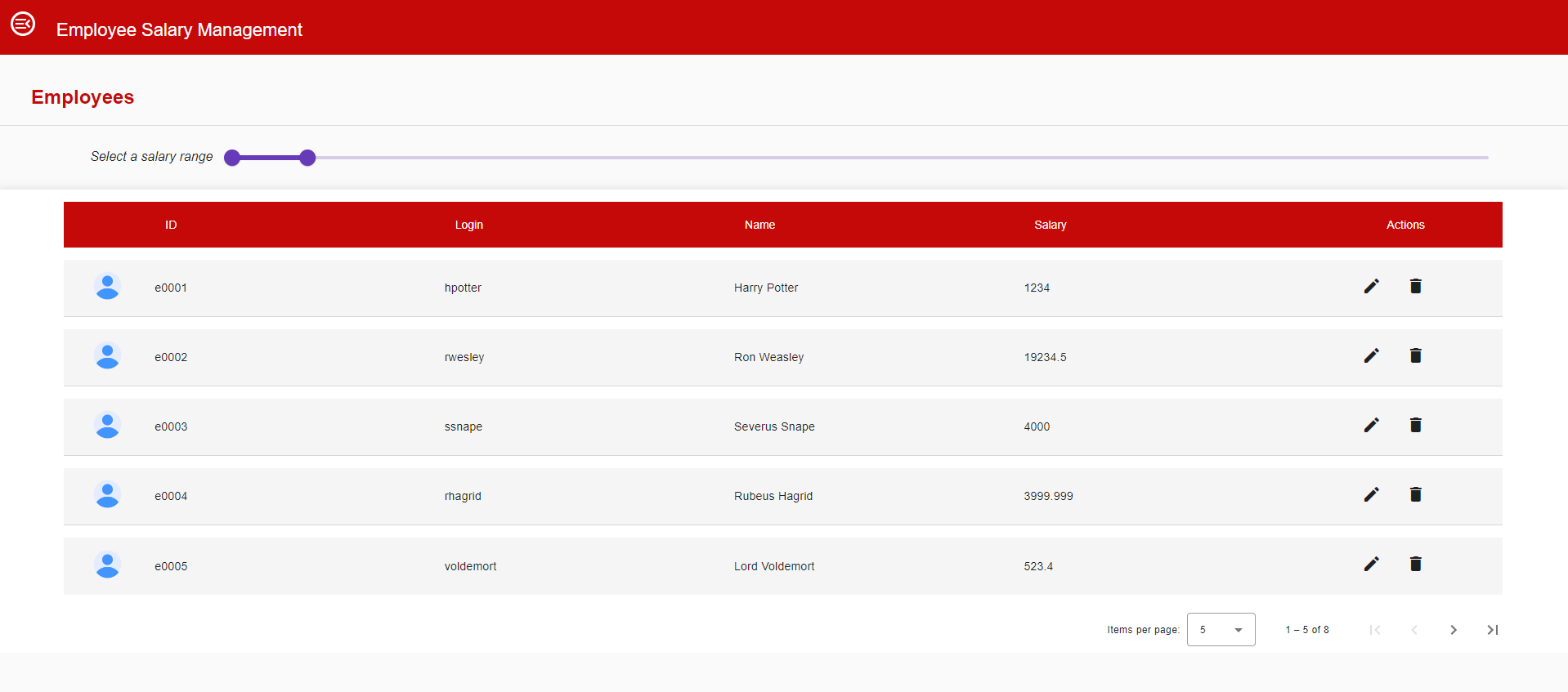
## Fetch All Employees’ Data

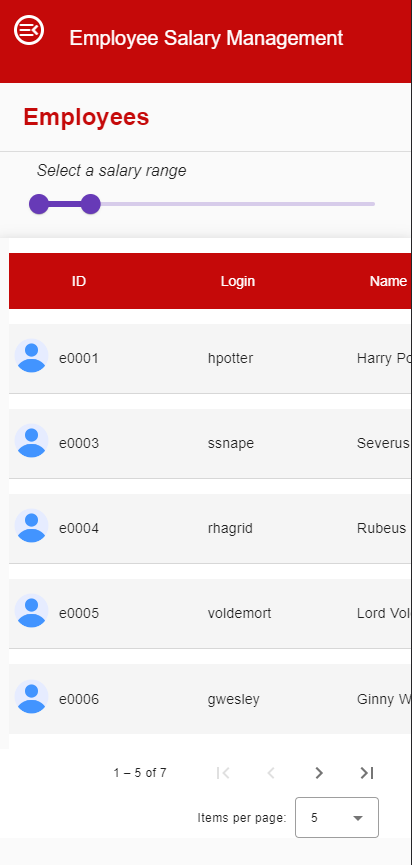
This functionality can be used to delete an existing employee data by id.

|  |  |  |
| --- | --- | --- |
| API Details | | |
| Endpoint URL | GET http://localhost:8080/users?minSalary=0&maxSalary=4000&offset=0&limit=30&sort=+name | |
| Path parameters | Parameter | Data Type |
| minSalary | Float |
|  | maxSalary | Float |
|  | offset | 0 |
|  | limit | 30 |
|  | sort | +name |
| Response parameters | responseCode | String |
| message | String |
| totalElements | int |
| results | Employee |
| empDataRecord | List< Employee> |
| Sample Backend Request and Response (Success scenario) | | |
|  | | |
| Sample Backend Request and Response (Error scenario) | | |
|  | | |

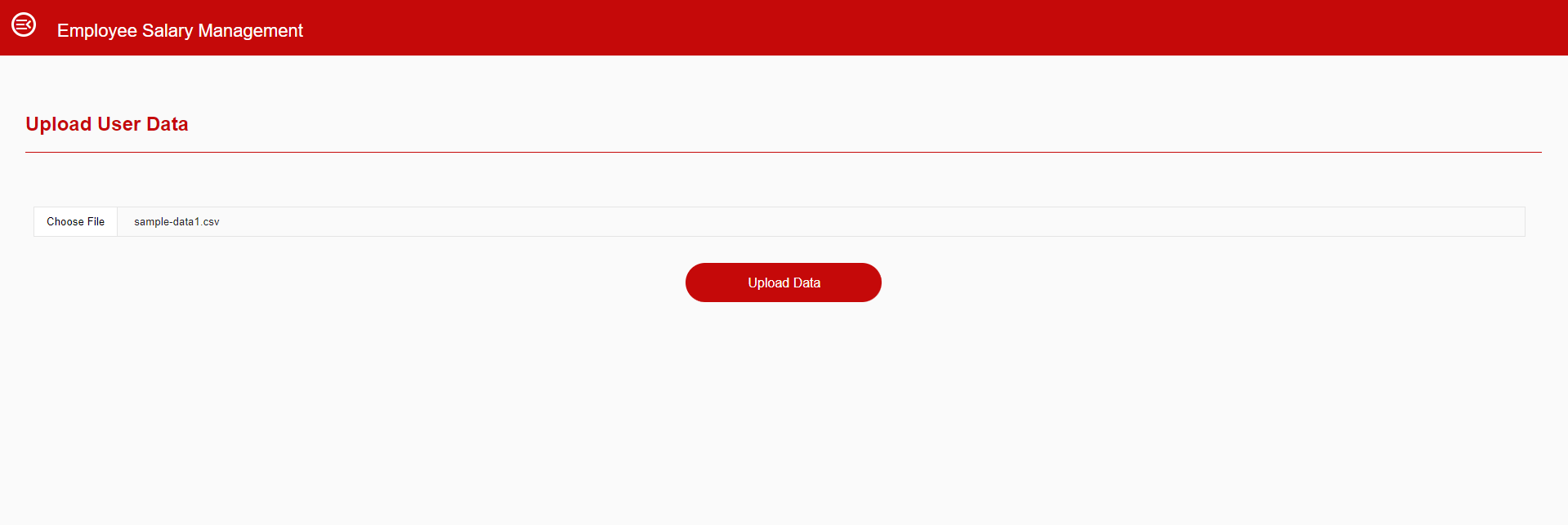
# UI Screens

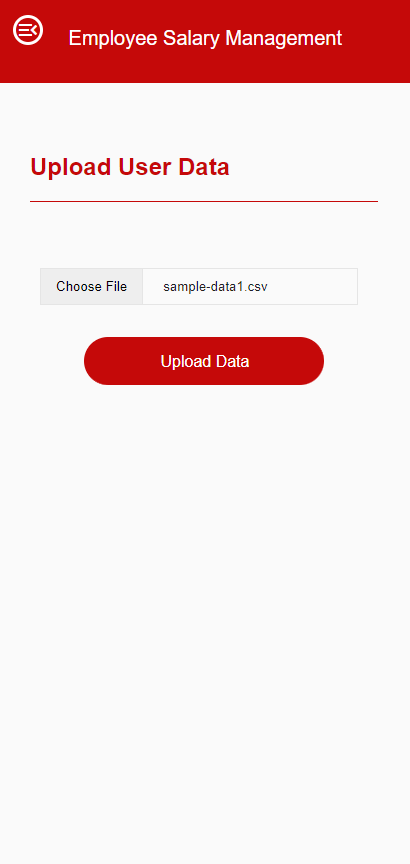
## Dashboard



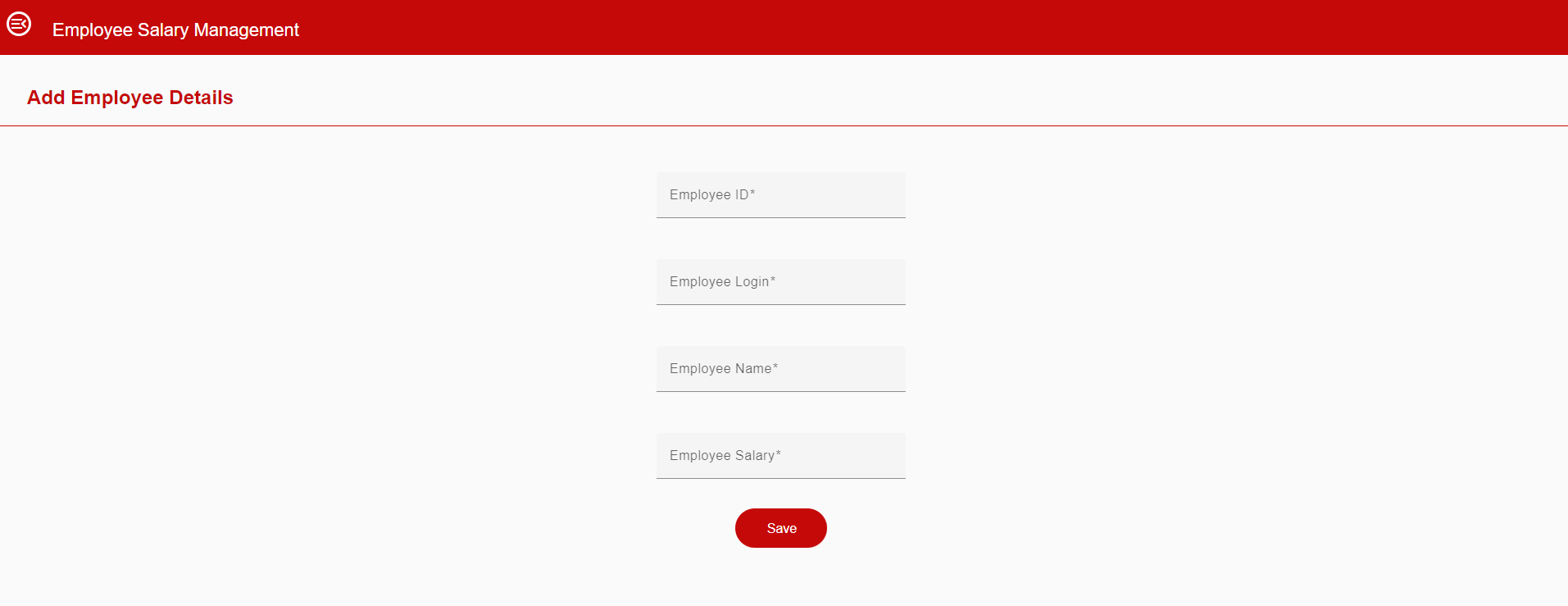


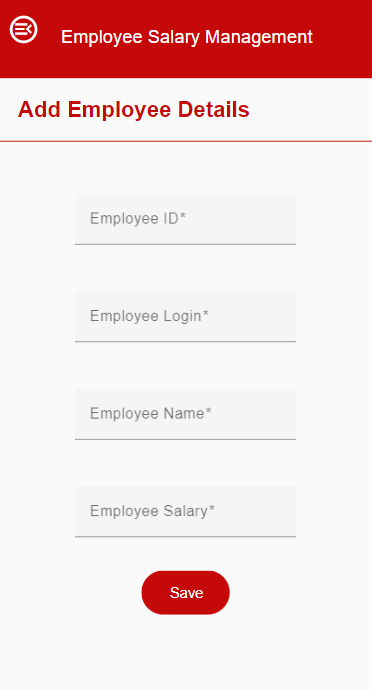
## Employees’ Data Upload



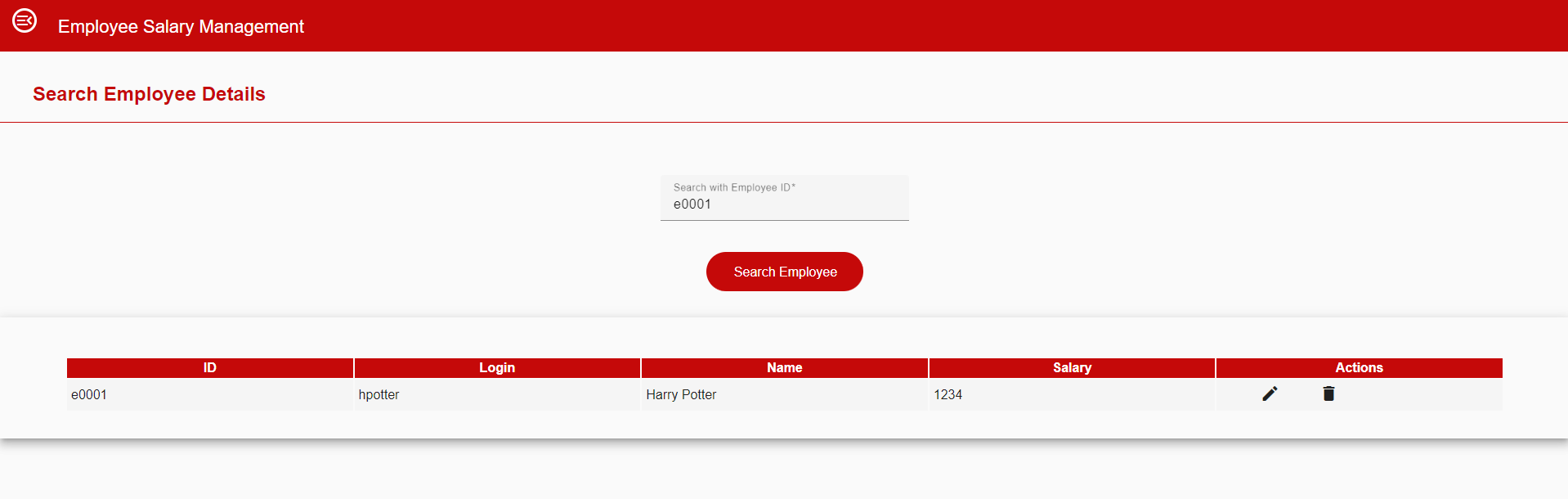


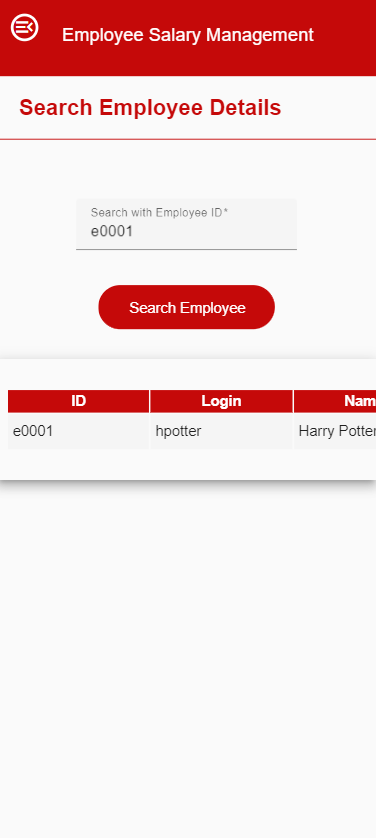
## Add Employee Details



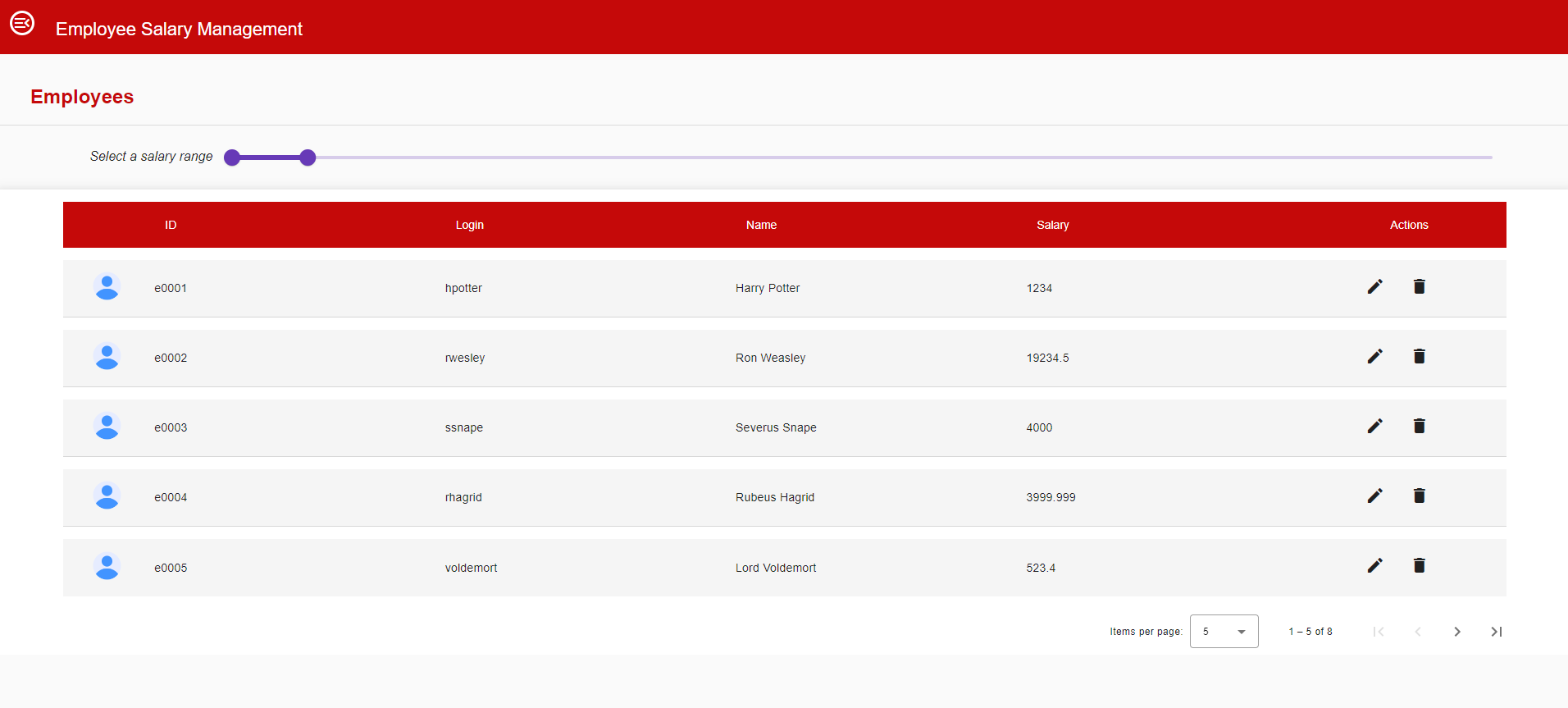


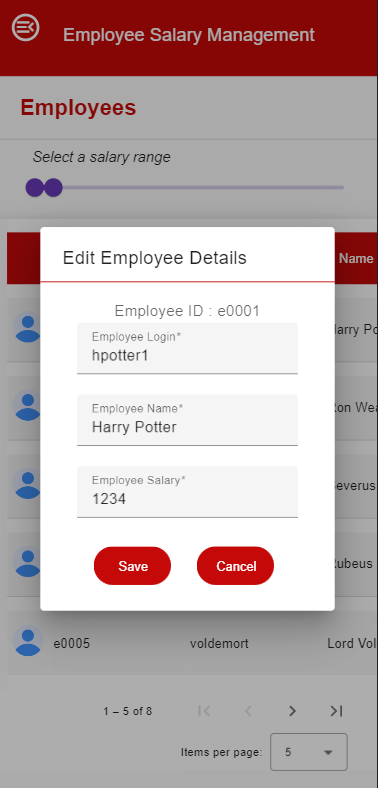
## Search Employee Data



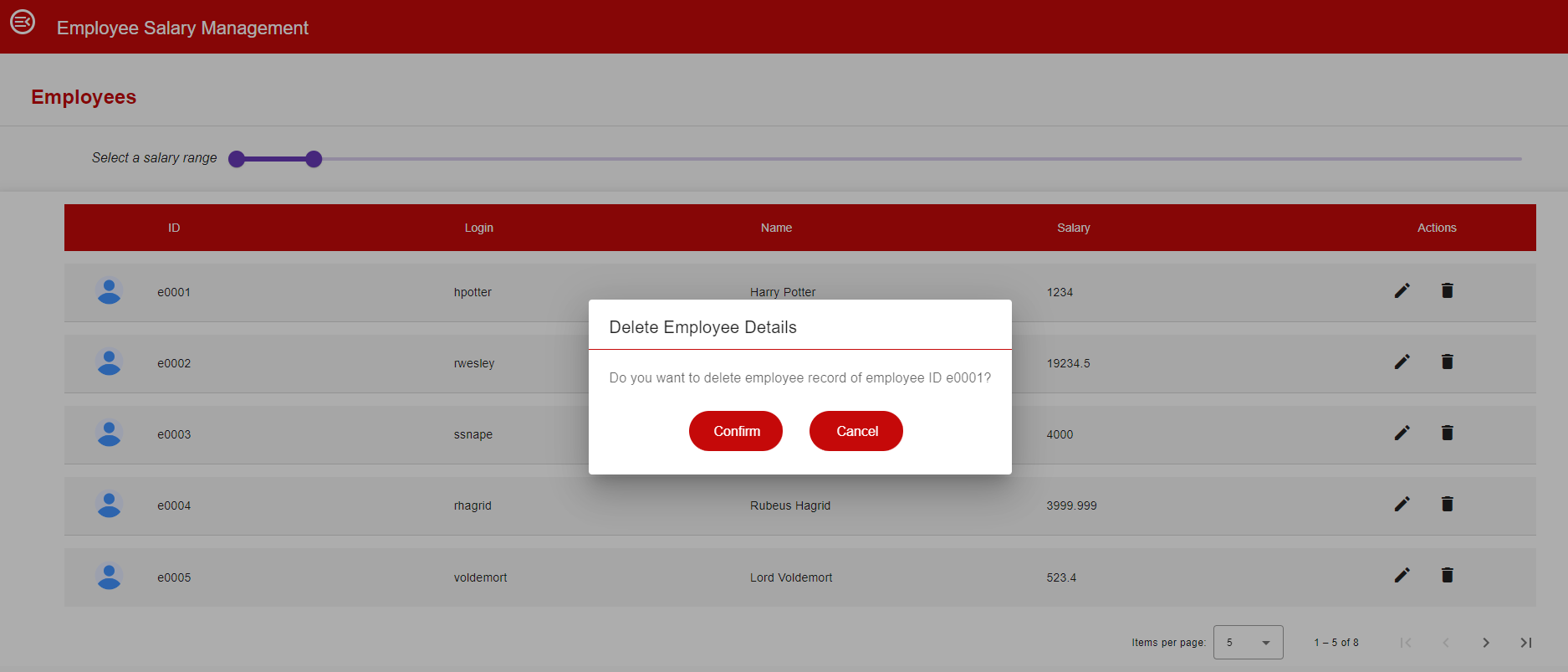


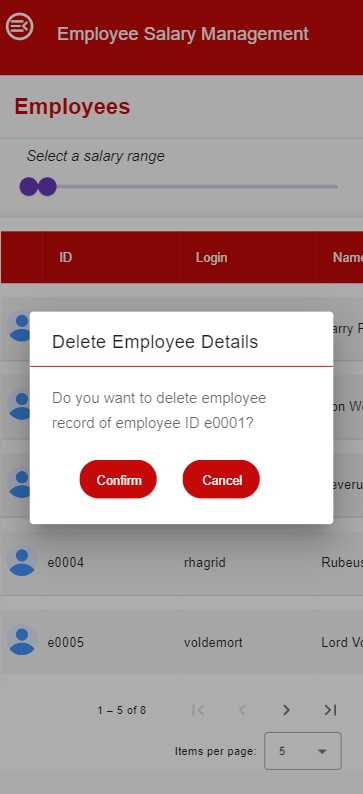
## Update Employee Details





## Delete Employee Data





## Side Navigation

