

---

# **HR DATA MANAGEMENT – TECHNICAL DESIGN DOCUMENT**

---

# Table of Contents

Table of Contents	1
Introduction	1
Purpose	1
Problem statement	1
Scope	1
Architecture	1
Table Procedures and Execution	2
Specifications	3
Procedures and Functions	3

---

## Introduction

### Purpose

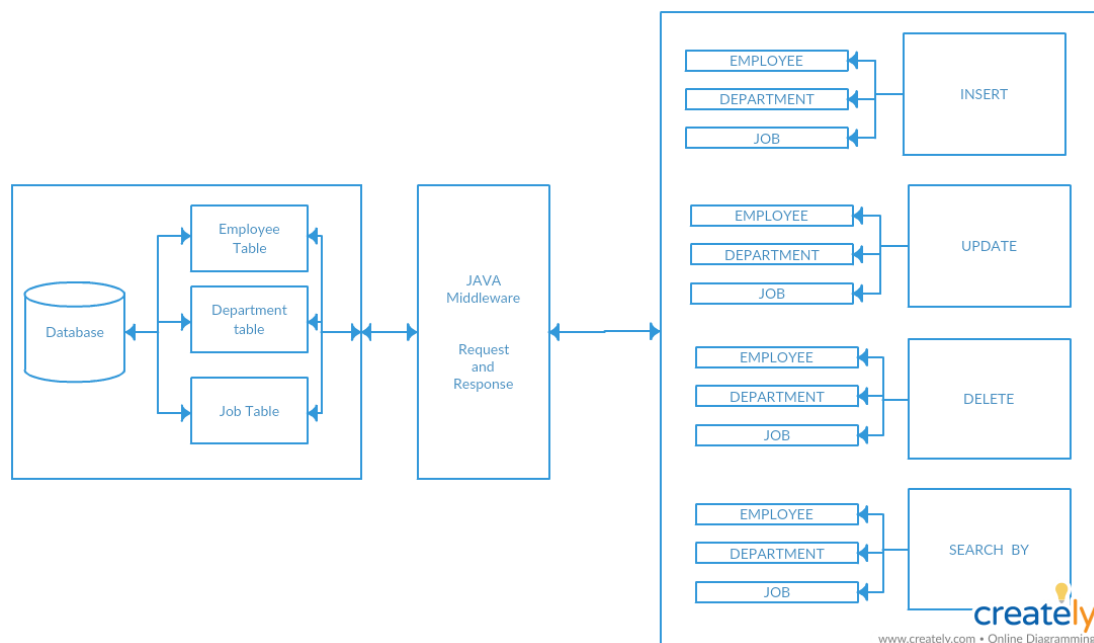
The purpose of this document is to provide a technical specification documentation for the data Management application.

### Scope

- The application will be able to perform add , update, search and delete on employee profile and department records.
- The Create and Search operation of Employee profile are exposed as external Web services.
- The Web services are made secured.
- For each operation on Employee profile one record will be inserted in Employee Transaction Log Table for tracking purpose.

## Architecture

This orchestration will carry the newly created/updated employee records into the MS SQL database. The table “employee”, which hold the employee data will be manipulated by the stored procedures. A table named “department” and a table named “job” has been created in the database to hold the department details and the job details. The schema for the table is found in Specification section.



---

## Table Procedures and Execution

The objectives of the project will be carried out by the following.

### Employee Table Procedures

#### Insert to Employee

1. Choose the insert to employee from the browser
2. Get the values from the user.
3. Validate the input from user
4. IF Error then prompt to re-enter ELSE do nothing
5. Call the stored procedure usp\_employee\_insert to insert the data into the table.
6. Write the transaction details into the transaction log table.

#### Update the employee

1. Choose the update the employee from the browser.
2. Load the values of the record into the form.
3. Get the edited values from the user.
4. Validate the user input
5. IF Error then prompt to re-enter ELSE do nothing
6. Call the stored procedure usp\_employee\_update to update the existing record from the table.
7. Write the transaction details into the transaction log table.

#### Delete Employee

1. Choose the employee to be deleted from the browser.
2. Call the stored procedure usp\_employee\_delete to delete the data from the table.
3. Write the transaction details into the transaction log table.

### Department Table Procedures

#### Insert to Department

1. Choose the insert to department from the browser
2. Get the values from the user.
3. IF Error then prompt to re-enter  
ELSE do nothing
4. Call the stored procedure usp\_department\_insert to insert the data into the table.
5. Write the transaction details into the transaction log table.

#### Update the Department

1. Choose the update the department from the browser.
  2. Load the values of the record into the form.
  3. Get the edited values from the user.
  4. IF Error then prompt to re-enter  
ELSE do nothing
  5. Call the stored procedure usp\_department\_update to update the existing record from the table.
  6. Write the transaction details into the transaction log table.
-

### Delete Department

1. Choose the department to be deleted from the browser.
2. Call the stored procedure usp\_department\_delete to delete the data from the table.
3. Write the transaction details into the transaction log table.

## Specifications

The schema for the table Employee is as follows,

**Employee Table**

#	Column Name	Datatype	Nullable
1	empid	INT IDENTITY(1,1) PK	FALSE
2	empname	VARCHAR(50)	FALSE
3	mgrid	INT	TRUE
4	dept	INT FK	FALSE
5	salary	INT	FALSE
6	hiredate	DATE	FALSE
7	jobcode	INT FK	FALSE
8	phone	INT	FALSE
9	email	VARCHAR(40)	FALSE

**Department Table**

#	Column Name	Datatype	Nullable
1	deptname	VARCHAR(30)	FALSE
2	deptid	INT IDENTITY(1,1)	FALSE

**Job Table**

#	Column Name	Datatype	Nullable
1	jobcode	INT IDENTITY(1,1)	FALSE
2	jobtitle	VARCHAR(20)	FALSE
3	maxsalary	NUMERIC(12,0)	FALSE

**Transaction Log Table**

#	Column Name	Datatype	Nullable
1	empid	VARCHAR(4) PK	FALSE
2	operation	VARCHAR(10)	FALSE
3	empname	VARCHAR(50)	FALSE
4	mgrid	INT	FALSE
5	deptid	INT	FALSE
6	salary	NUMERIC(12,0)	FALSE
7	hiredate	DATE	FALSE
8	jobcode	INT	FALSE
9	phone	NUMERIC(10,0)	FALSE
10	email	VARCHAR(40)	FALSE
11	oldempname	VARCHAR(50)	FALSE
12	oldsalary	INT	FALSE
13	oldjobcode	INT	FALSE
14	oldphone	NUMERIC(10,0)	FALSE
15	olddeptid	INT	FALSE
16	oldemail	VARCHAR(40)	FALSE

## Procedures and Functions

Procedure Name	Functionality	Parameters
usp_employee_insert	Procedure to insert the obtained user data	@empname varchar(50), @mgrid INT, @dept INT, @salary numeric(12,0), @jobcode INT, @phone numeric(10,0), @email varchar(40)
usp_employee_update	Procedure to update the data in the table	@empname varchar(50), @mgrid INT, @dept INT, @salary numeric(12,0), @jobcode INT, @phone numeric(10,0), @email varchar(40)
usp_employee_delete	Procedure to delete the records	@empid INT

in the employee table

Procedure Name	Functionality	Parameters
usp_department_insert	Procedure to insert the obtained user data into the department	@deptcode INT @deptname varchar(20)
usp_department_update	Procedure to update the data in the department table	@deptcode INT @deptname varchar(20)
usp_department_delete	Procedure to delete the records in the department table	@deptcode INT