**Case Study Document: Integrate Java with React**

**Table of Contents**

1. Introduction

- 1.1 Purpose

- 1.2 Scope

- 1.3 Technologies Used

2. Business Scenario

- 2.1 Background

- 2.2 Problem Statement

- 2.3 Objectives

3. System Architecture

- 3.1 High-Level Overview

- 3.3 Data Flow

4. Features

- 4.1 Employee Management

- 4.2 Data Storage and Retrieval

- 4.3 CRUD Operations

**1. Introduction**

**1.1 Purpose**

The purpose of this case study is to demonstrate the integration of java api with react. The assignment aims to showcase the implementation of essential features using this technology stack.

**1.2 Technologies Used**

- Java Spring

- React

-Oracle

**2. Business Scenario**

**2.1 Background**

The assignment is based on a scenario where a company requires a web application to manage employee data. The data consists of employee’s basic information, such as name and email. The application should support storing, retrieving, updating, and deleting employee records.

**2.2 Problem Statement**

The company needs an efficient solution to manage employee data.

**2.3 Objectives**

The main objectives of this case study are:

1. write a Java application to perform the below operations
   1. we should be able to get the employee details
   2. employee details should be editable
   3. we should be able to delete the employee details

**3. System Architecture**

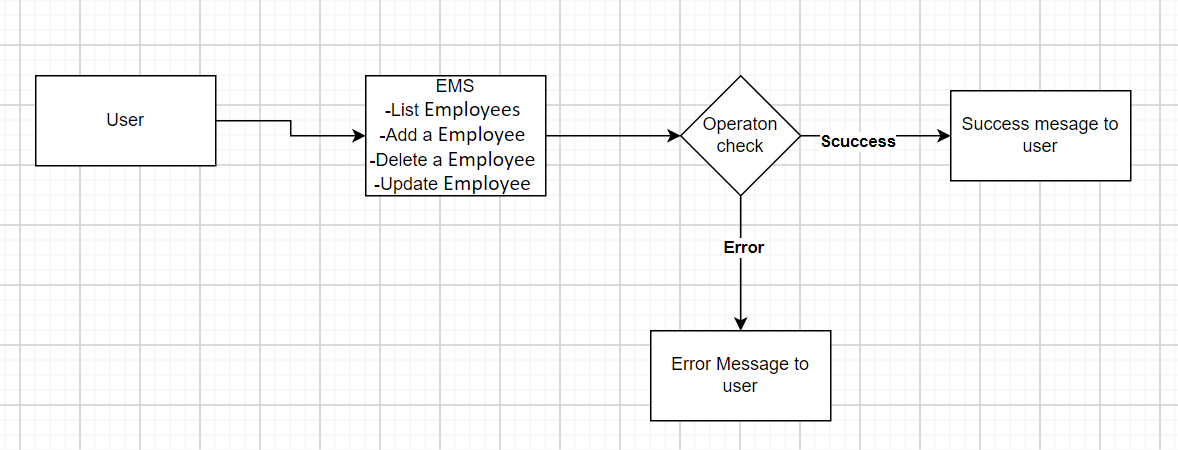
**3.1 High-Level Overview**

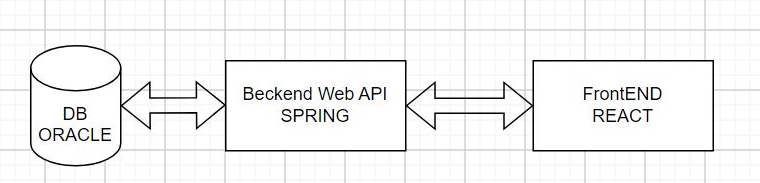
The system will follow a typical have the following:

1. Presentation Layer: React frontend application that will help user to perform the operations

2. Business Logic Layer: Contains application logic and coordinates data access.

3. Data Access Layer: Manages interaction with oracle database.





**3.3 Data Flow**

The data flow within the system is as follows:

1. User will select on add button to add employee’s

2. Take an input from user.

3.Perform the operation

**4. Features**

The web application will offer the following features:

**4.1 Employee Management**

- Create new user records with name, email, employeeID

- Retrieve user records by ID or email.

- Update existing employee information.

- Delete employee records.

**4.2 Data Storage and Retrieval**

- Retrieve user data from oracle database.

**4.3 CRUD Operations**

- Implement Create, Read, Update, and Delete operations using java spring and oracle in backend.

- Validate inputs and handle exceptions gracefully.