



**EBF**

***Employee Storage Service***

***Test Cases***



## **Confidentiality and Copyright**

Proprietary, confidential data. All rights reserved.

Distribution and/or reproduction of these documents or any of their parts, their utilisation or communication of their contents to third parties is not allowed, unless formally conceded.

Contravention will cause damages.

Copyright © 2019 by S4 Integration (Pty) Ltd.

S4 Integration (Pty) Ltd.  
1 Leadwood Crescent  
Fairview  
Port Elizabeth  
South Africa  
6065  
[www.s4.co.za](http://www.s4.co.za)

*Systems · Solutions · Software · Support*



# Table of Contents

<b>Confidentiality and Copyright .....</b>	<b>2</b>
<b>1 System Overview .....</b>	<b>4</b>
<b>2 Test Cases.....</b>	<b>5</b>
2.1 Employees .....	5
2.2 Company.....	7



# 1 System Overview

We're building an employee storage service for companies. A company admin needs to be able to do the following:

- create employee records for company
- view them in a list and detail view for company
- update the employee records and delete them for company
- find the average salary for the company

Your task is to build the backend for this service. It should be a REST API that exposes the required endpoints. The company data can either be autogenerated or configurable or dynamically created (up to you).

## 1.1 Nice to haves:

- Build a frontend for the REST API
- Use JVM based technologies

## 1.2 Data definitions:

- Employee model
  - Name
  - Surname
  - Email
  - Address
  - Salary
  - Company ID
- Company Model
  - Name

Deliver: A publicly available git repository with a README.md that explains how to compile, build and start the project



## 2 Test Cases

### 2.1 Employees

/api/allEmployees	
Trigger	Frontend trigger (home page loaded)
Normal Flow	<ol style="list-style-type: none"><li>1. User navigates to webpage</li><li>2. /api/getAllEmployees gets called</li><li>3. returns all employees</li></ol>
Expected Result	List of employee objects returned
Actual Result	List of employee objects returned
/api/getEmployee	
Trigger	User wants to update employee from frontend
Normal Flow	<ol style="list-style-type: none"><li>1. User clicks to update employee</li><li>2. /api/getEmployee gets called based on employee</li><li>3. Corresponding employee info is fetched</li><li>4. All details linked to an employee is shown</li></ol>
Expected Result	All details linked to an employee is shown
Actual Result	All details linked to an employee is shown
/api/addEmployee	
Trigger	User adds employee from frontend
Normal Flow	<ol style="list-style-type: none"><li>1. User submits employee info in a form</li><li>2. /api/addEmployee called with employee info</li><li>3. Data is validated</li><li>4. Valid employee inserted into the database</li></ol>
Expected Result	Valid employee inserted into the database
Actual Result	Valid employee inserted into the database
/api/updateEmployee	
Trigger	User updates employee from the frontend
1. Normal Flow	<ol style="list-style-type: none"><li>1. User submits new employee information on the form</li><li>2. /api/updateEmployee called to update the Employee's information</li><li>3. Data is validated</li></ol>



Expected Result	4. Employee is updated with valid data Employee is updated with valid data
Actual Result	Employee is updated with valid data
<b>/api/deleteEmployee</b>	
Trigger	User confirms the deletion of an employee
Normal Flow	<ol style="list-style-type: none"><li>1. User clicks to delete an employee</li><li>2. User confirms the deletion of the employee</li><li>3. /api/deleteEmployee with the relevant employee data is sent</li><li>4. The given employee is deleted from the database</li></ol>
Expected Result	The given employee is deleted from the database
Actual Result	The given employee is deleted from the database
<b>/api/getAllEmployeesForCompany</b>	
Trigger	User clicks on a company
Normal Flow	<ol style="list-style-type: none"><li>1. User clicks on a company</li><li>2. /api/getAllEmployeesForCompany with relevant company data is sent</li><li>3. The given employees associated with that company is fetched</li><li>4. All employees associated with that company is displayed</li></ol>
Expected Result	All employees associated with that company is displayed
Actual Result	All employees associated with that company is displayed



## 2.2 Company

/api/getAllCompanies	
Trigger	Frontend trigger (home page loaded)
Normal Flow	<ol style="list-style-type: none"><li>1. User navigates to webpage</li><li>2. /api/getAllCompanies gets called</li><li>3. returns all companies</li></ol>
Expected Result	List of company objects returned
Actual Result	List of company objects returned
/api/getCompanyAverageSalary	
Trigger	Frontend trigger (home page loaded)
Normal Flow	<ol style="list-style-type: none"><li>1. User navigates to webpage</li><li>2. /api/getCompanyAverageSalary gets called</li><li>3. returns each company's average salary per employee</li></ol>
Expected Result	Returns each company's average salary per employee
Actual Result	Returns each company's average salary per employee