

## POC Micro services

### 1. Introduction

I created this project with the name **POC** which stands for “**Proof Of Concept**”. It is created based on **Micro Service Architecture**. **Micro service** is a method of developing software applications as a suite of independently deployable, small, modular services in which each service runs a unique process and communicates through a well-defined, lightweight mechanism to serve a business goal.

So I have designed two important services called “**Employee Service**” and “**Student Service**”. Both are independently deployable and runs unique process.

This application contains 3 important modules. They are,

- i. **poc-core.**
- ii. **poc-employee.**
- iii. **poc-student.**

The module **poc-core** contains common functionalities for both **poc-employee** and **poc-student**. The poc-employee is for “Employee Service” and poc-student is for “Student Service”.

### 2. Environment:

- i. Jdk 1.8
- ii. Spring 4.3.4
- iii. Spring-boot 1.4.2. (It is for Micro Service)
- iv. Gradle 3.2.1
- v. Postgres (Not using Oracle since it is Licensed version)
- vi. Tomcat – Embedded with Spring Boot.

### 3. Steps to Install:

Please follow the below steps to install this application.

Step1: Clone the source code from the hithub.com using the url <https://github.com/vayalaivadivel/poc.git>. The project directory is now <poc>.

Step2: Install softwares jdk1.8, postgres and gradle3.2. Set GRADLE\_HOME for gradle in environment variable. For more information, please refer <https://gradle.org/>.

Step3: Configuration

- i. Go to the project directory <poc> and copy demodb.sql run the script in postgres and make note of username/password.
- ii. Update data base username/password accordingly in  
<poc\poc-employee\src\main\resources>/application.yml and  
<poc\poc-student\src\main\resources>/application.yml

- iii. Currently “Employee Service” url is <http://localhost:9001/employee>. If you want to change this port-no and context path, change in <poc\poc-employee\src\main\resources>/application.yml.
- iv. Currently “Student Service” url is <http://localhost:9002/student>. If you want to change this port-no and context path, change in <poc\poc-employee\src\main\resources>/application.yml.

#### 4. Run Application:

- i. To run “Employee-Service”, navigate directory <poc\poc-employee> in DOS-Prompt/Unix and type >gradle clean build bootRun.
- ii. To run “Student-Service”, navigate directory <poc\poc-student> in DOS-Prompt/Unix and type >gradle clean build bootRun.

I have used spring-actuator api. So you can use below url to ensure both the service running successfully or not,

<http://localhost:9001/employee/health> and <http://localhost:9002/student/health>.

#### 5. Test Application:

Please use the tool Postmaster/Soap-UI to test and also set below request headers for this testing,

username-success  
password-success@123

##### a. Employee-Service:

Below are the list of requests for this service,

- i. Create  
POST <http://localhost:9001/employee/save>  
{  
  "name": "Raja",  
  "age": 22,  
  "salary": 45000.00  
}
- ii. Update  
PUT <http://localhost:9001/employee/update>  
{  
  "empNo": 1,  
  "name": "Rani",  
  "age": 44,  
  "salary": 55000.00  
}

iii. Retrieve

GET [http://localhost:9001/employee /get/{primaryKey}](http://localhost:9001/employee/get/{primaryKey})

iv. Retrieve All

GET [http://localhost:9001/employee /getAll](http://localhost:9001/employee/getAll)

v. Delete

DELETE [http://localhost:9001/employee /delete/{primaryKey}](http://localhost:9001/employee/delete/{primaryKey})

vi. Delete All

DELETE <http://localhost:9001/employee/deleteAll>

b. Student-Service:

Below are the list of requests for this service,

i. Create

POST <http://localhost:9002/student/save>

```
{  
  "name": "Mani",  
  "age": 21,  
  "totalMark": 350  
}
```

ii. Update

PUT <http://localhost:9002/student/update>

```
{  
  "regNo": 1,  
  "name": "Velu",  
  "age": 22,  
  "totalMark": 450  
}
```

iii. Retrieve

GET <http://localhost:9002/student/get/{primaryKey}>

iv. Retrieve All

GET <http://localhost:9002/student/getAll>

v. Delete

DELETE <http://localhost:9002/student/delete/{primaryKey}>

vi. Delete All

DELETE <http://localhost:9002/student/deleteAll>.

Note: I have not covered test suite for this application. I will update those in next release if you want.