**Solution Approach Document**

Solution Approach Document: **Employee CRUD**

|  |  |  |  |
| --- | --- | --- | --- |
| Version: | 1.0 | Version Release Date: | 12/07/2021 |
| Author(s): | Khalid Qassem | Document Status: | **FINAL** |

Contents

[1 Introduction 3](#_Toc89790292)

[1.1 Purpose 3](#_Toc89790293)

[1.2 Reference documents 3](#_Toc89790294)

[1.3 Assumptions 3](#_Toc89790295)

[1.4 Architectural Decisions 3](#_Toc89790296)

[2 Solution APPROACH 4](#_Toc89790297)

[2.1 System Architecture Overview 4](#_Toc89790298)

[2.2 API Interface Document 4](#_Toc89790299)

[2.3 Data Model 5](#_Toc89790300)

[2.4 How to use 6](#_Toc89790301)

[2.5 Repo link 7](#_Toc89790302)

# Introduction

## Purpose

The purpose of this document is to define the solution approach for a sample application contains create, read, update and delete operations for an Employee Object, using Java rest backend (Spring) and swagger-ui as a client.

## Reference documents

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Name | Version | Author | Location and Attachment |
| 1 | Requirements Specification Document | V1.0 |  |  |
| 2 | Github code branch |  | Khalid | <https://github.com/khaledkasm/Sample-Project-Assignment-Solution-Architect.git> |

## Assumptions

|  |  |  |
| --- | --- | --- |
| ID | Details | Approved by |
| 1 | Employee salary has to be 1 or more |  |
| 2 | While creating the employee, we assume the department already created and the manager already created. |  |

## Architectural Decisions

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Issue | Reasoning & Decision | Approved by |
|  |  |  |  |

# Solution APPROACH

## System Architecture Overview

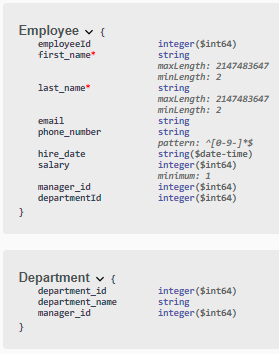


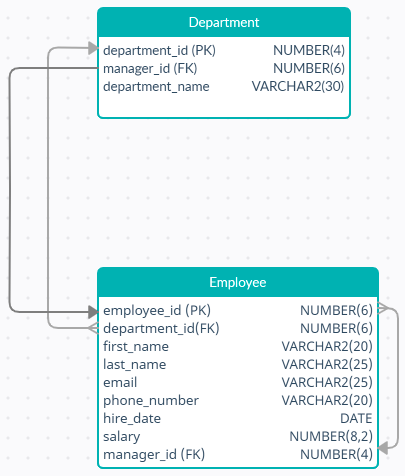
## API Interface Document

You can use swagger documentation from the below link

<http://localhost:8080/swagger-ui/index.html?configUrl=/v3/api-docs/swagger-config#/>

## Data Model





## How to use

1. Run the application then go to the link <http://localhost:8080/swagger-ui/index.html?configUrl=/v3/api-docs/swagger-config#/>
2. Create an employee with the following request

{

"first\_name": "Khalid",

"last\_name": "Qassem",

"email": "khaledkasm@gmail.com",

"phone\_number": "9435-4841",

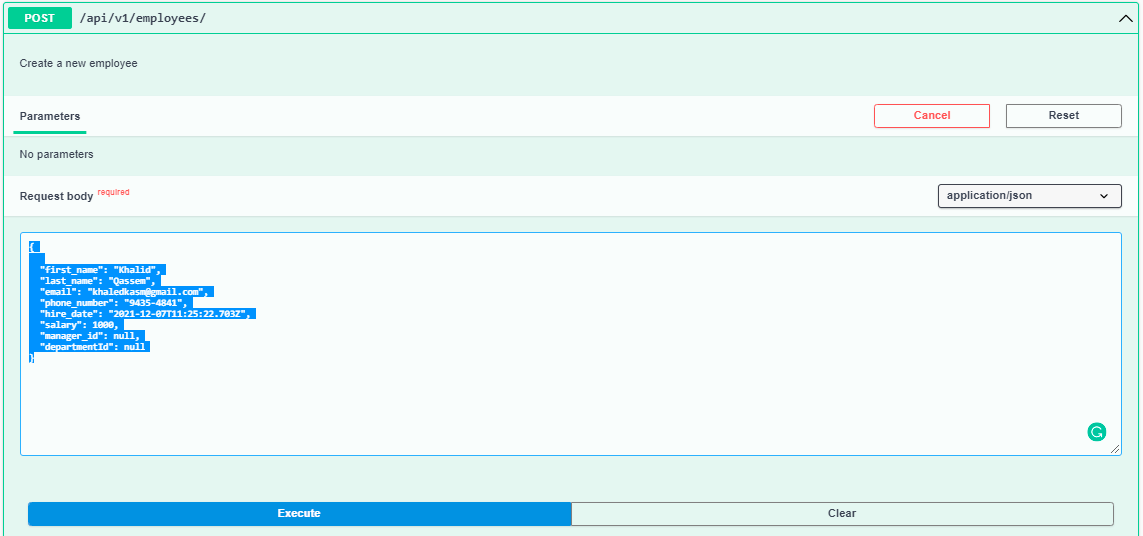
"hire\_date": "2021-12-07T11:25:22.703Z",

"salary": 1000,

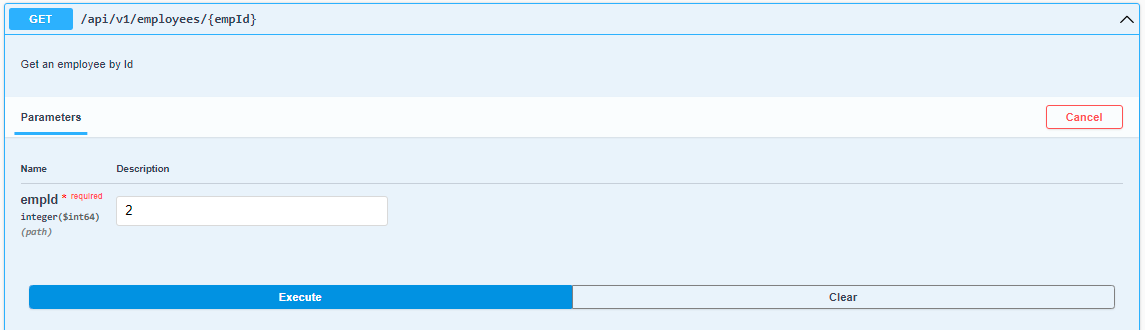
"manager\_id": null,

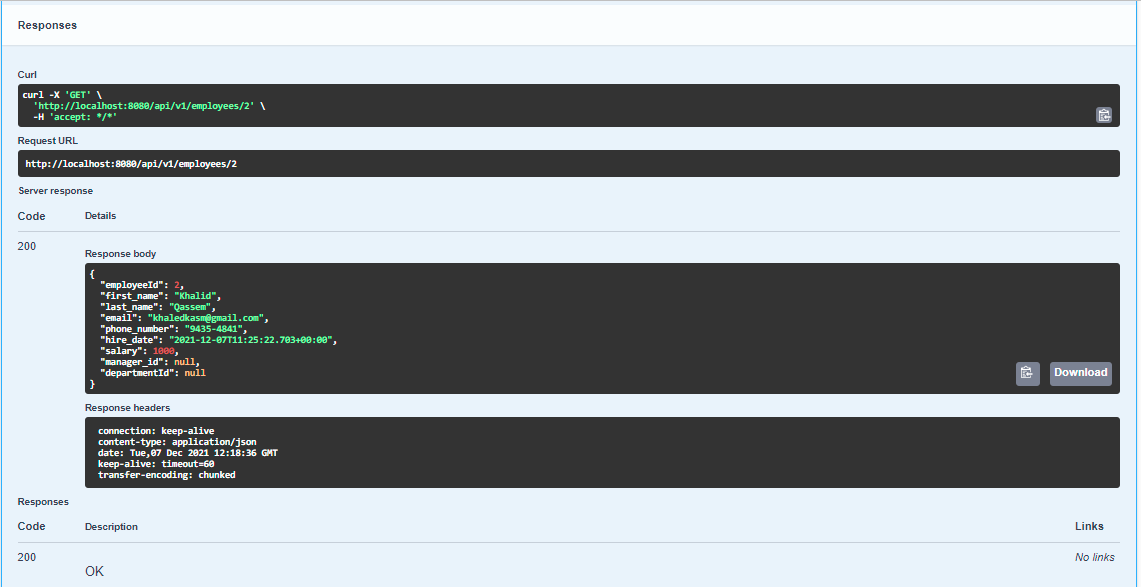
"departmentId": null

}



1. You can retrieve the employee by executing the following request





## Repo link

<https://github.com/khaledkasm/Sample-Project-Assignment-Solution-Architect.git>