

# Migración SICOF ERP - Proceso: Guía de Documentación de Microservicios

La siguiente sección define las configuraciones que se deben tener presente en el desarrollo de los microservicios de la fábrica de desarrollo para la documentación de los microservicios.

## Consideraciones Previas

- Los microservicios deben aplicar las configuraciones definidas en la mayoría de los casos a todas las clases que definan la lógica del negocio como Entidades, Controladores, Configuraciones etc.
- Si un microservicio requiere configuraciones especiales, estas deben ser validadas con los líderes de desarrollo (Pablo Quintana, Daberson Henao, Carlos Torres, Gersain Castañeda).

## Información Importante

Se ha definido para el proceso de documentación de los microservicios el estándar OpenAPI Specification (OAS) 3.0.3. <sup>e<sup>1</sup></sup> la cual define una descripción de interfaz estándar, independiente del lenguaje de programación para las API REST, que permite que tanto humanos como computadoras descubran y comprendan las capacidades de un servicio sin requerir acceso al código fuente, documentación adicional o inspección del tráfico de red. Cuando se define adecuadamente a través de OpenAPI, un consumidor puede comprender e interactuar con el servicio remoto con una cantidad mínima de lógica de implementación. Similar a lo que han hecho las descripciones de interfaz para la programación de nivel inferior, la especificación OpenAPI elimina las conjeturas al llamar a un servicio.

## Documentación de Soporte

En la siguiente web se relacionan la forma de implementación de la especificación OAS<sup>2</sup> con la tecnología Sprintboot [Documentación springdoc](#)

## Configuraciones del Microservicio

Todo microservicio que implemente documentación OAS<sup>3</sup> debe incluir en su archivo de configuración Maven **POM**<sup>4</sup> la dependencia del proyecto del Dominio de clases y entidades Comunes el cual se describe a continuación:

```
<dependency>
  <groupId>co.ada.models</groupId>
  <artifactId>ModelsClassesADA</artifactId>
  <version>0.0.1-SNAPSHOT</version>
```

</dependency>

El desarrollador debe utilizar la versión más reciente del proyecto la cual puede ser consultada en la sección [Versionamiento de APIs](#)

## Como implementar documentación OAS el Dominio de Clases de Entidades Comunes

Para usar el Dominio de Clases de Entidades Comunes en el proceso de documentación siga los siguientes pasos:

### Paso 1

Registre la dependencia actualizada del Dominio de Clases de Entidades Comunes.

### Paso 2

cree un paquete llamado config

### Paso 3

Cree la clase de configuración como se muestra continuación.

```
package co.ada.test.microservicio.pruebamodel.config;

import org.springframework.context.annotation.Configuration;
import co.ada.doc.config.SpringOpenApiConfigBase;

@Configuration
public class MySpringOpenApiConfig extends SpringOpenApiConfigBase {}
```

### Paso 4

Configure en el bootstrap.yml del proyecto los parametros de documentación del microservicio. Apoyese del siguiente modelo de referencia extraido del proyecto modelo.

```
openapi-document:
  info:
    title: "Prueba Model ADA API"
    description: "Ejemplo de Microservicio utilizando el Dominio de clases y entidades comunes"
```

```

termsOfService: ""
contact:
  name: "Carlos Torres"
  url: "http://ada.co"
  email: "carlos.torres@ada.co"
license:
  name: "ADA 1.0"
  url: ""
version: "0.0.1"
servers:
  public:
    url: "http://zuul-mig.adacsc.co/api/prueba"
    description: "Development integration public server"
externalDocs:
  description: ""
  url: ""

```

## Paso 5

Documento las clases del microservicios con la especificación OAS<sup>5)</sup>. Apoyese del proyecto modelo el cual tiene un ejemplo de configuración.

```

package co.ada.test.microservice.prueba.dbconnect;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.beans.factory.annotation.Qualifier;
import org.springframework.boot.orm.jpa.EntityManagerFactoryBuilder;
import org.springframework.context.annotation.Bean;
import org.springframework.context.annotation.Configuration;
import org.springframework.data.jpa.repository.config.EnableJpaRepositories;
import org.springframework.orm.jpa.JpaTransactionManager;
import org.springframework.orm.jpa.LocalContainerEntityManagerFactoryBean;
import org.springframework.transaction.annotation.EnableTransactionManagement;

import co.ada.db.config.DataSourceConfigBase;
import co.ada.db.config.IDataSourceConfig;

@Configuration
@EnableJpaRepositories(
    basePackages = "co.ada.test.microservice.prueba.repo", //Ver Nota 1
    transactionManagerRef = "transcationManager",
    entityManagerFactoryRef = "entityManager")
@EnableTransactionManagement
public class MyDataSourceConfig extends DataSourceConfigBase {

    @Bean(name = "entityManager")
    public LocalContainerEntityManagerFactoryBean
entityManagerFactoryBean(EntityManagerFactoryBuilder builder) {

```

```
        return  
builder.dataSource(dataSource()).packages("co.ada.models.microservicio.usuario").build();//Ver Nota 2  
    }  
  
    @Bean(name = "transcationManager")  
    public JpaTransactionManager transactionManager(@Autowired  
    @Qualifier("entityManager") LocalContainerEntityManagerFactoryBean  
    entityManagerFactoryBean) {  
        return new  
JpaTransactionManager(entityManagerFactoryBean.getObject());  
    }  
}
```

### Nota 1

Registre el paquete local donde tiene los repositorios / dao de acceso a datos.

### Nota 2

Registre el paquete de entidades del Dominio de Clases de Entidades Comunes que utilizará con los repositorio.

## Repositorios

Se define en el Dominio de Clases de Entidades Comunes el repositorio estandar para acceso a datos el cual debe ser utilizado en todos los microservicios que requieran esa funcionalidad. El repositorio esta ubica en el paquete co.ada.db.config.repository y contiene la clase que a continuación se describe:

```
package co.ada.db.config.repository;  
  
import org.springframework.data.jpa.repository.JpaRepository;  
import org.springframework.stereotype.Repository;  
  
@Repository  
public interface JpaAdaRepository<T, ID> extends JpaRepository<T, ID> {}
```

[←Volver atrás](#)

1)

<https://github.com/OAI/OpenAPI-Specification>

2) 3) 5)

<https://github.com/OAI/OpenAPI-Specification/blob/master/versions/3.0.3.md>

4)

[https://www.tutorialspoint.com/maven/maven\\_pom.htm](https://www.tutorialspoint.com/maven/maven_pom.htm)

From:  
<http://wiki.adacsc.co/> - Wiki

Permanent link:  
<http://wiki.adacsc.co/doku.php?id=ada:howto:sicoferp:factory:migracionsicoferp:process:backend:guiadocmicroservicios&rev=1589914297>

Last update: 2020/05/19 18:51

