

Fábrica - Servicios Web - Integraciones - MicroservicioConfigApp

Este servicio pemrite generar y proveer configuraciones centralizadas para microservicios. Actualmente soporta la siguientes configuraciones:

- Registro de aplicaciones: Permite registrar configuraciones de aplicaciones las cuales a su vez permitirán el resto de configuraciones. Todas las demas configuraciones van a requerir que el microservicio que la utilice existe como configuración de aplicación.
- Configuraciones centralizadas de Datasources: Provee configuraciones de conexiones a origenes de datos (DB - Datasources) a los microservicios. Estos previamente deben parametrizar cuales datasources pueden ser utilizados por el microservicios.
- Configuraciones de parámetros centralizados: Provee configuraciones de parámetros a los microservicios, estos pueden ser: Texto, Número, Fecha, Texto grande (Lob) y Configuraciones en formato json.

Descripción del Servicio

Nombre	Descripción	Tipo Servicio	Tecnología	Lenguaje	Tipo Desarrollo	Versión Desarrollo	Archivo de Configuración
MicroservicioConfigApp	Microservicio que gestiona y provee configuraciones centralizadas a los microservicios.	Rest	SpringBoot 2.6.3	Java 11.0.315	Nuevo	11.6.0.1	application.properties
Tipo Empaquetado	Gestor de Dependencias	Servidor de Despliegue	IDE Proyecto	Tipo Repositorio	Url Desarrollo	Url Calidad	Url Producción
war	Maven 3.8.1	WildFly 20	SpringTools 4.6	Subversión	SVN branches	SVN trunk	SVN tags
Dependencias							
groupId	artifactId	version	scope	optional	Descripción		
org.springframework.boot	spring-boot-starter-data-jpa	N/A	N/A	N/A	API de persistencia		
org.springframework.boot	spring-boot-starter-data-rest	N/A	N/A	N/A	API de mapeo de rutas		
org.springframework.boot	spring-boot-starter-web	N/A	N/A	N/A	Core Servicios Web		
org.springframework.boot	spring-boot-devtools	N/A	runtime	true	Recarga de aplicaciones		
org.projectlombok	lombok	N/A	N/A	true	API de anotaciones para reducción de código		
org.springframework.cloud	spring-cloud-sleuth-zipkin	N/A	N/A	true	API de gestión de logs centralizados		
org.springframework.boot	spring-boot-starter-test	N/A	test	N/A	Core Springboot		
com.oracle.database.jdbc	ojdbc8	N/A	runtime	N/A	Driver para bases de datos Oracle		
org.postgresql	webjars-locator-jboss-vfs	0.1.0	N/A	N/A	Plugins para despliegue en Wildfly		
org.springframework.boot	spring-boot-starter-tomcat	N/A	provided	N/A	Servidor Embebido Tomcat		
org.apache.maven.plugins	maven-compiler-plugin	3.8.1	N/A	N/A	Gestor de dependencias		
org.springdoc	springdoc-openapi-ui	1.5.12	N/A	N/A	Plugins para generar documentación Swagger		
org.springframework.boot	spring-boot-starter	N/A	N/A	N/A	Core Springboot		
Documentación Operaciones							
Tipo	Documentación					Visor	
Swagger	http://srv-appeon-000-w23.adacsc.co/MicroservicioConfigApp/api-docs					https://editor.swagger.io/	
Ambientes de Despliegue							
Tipo	Url		Observaciones				
Desarrollo:	http://10.1.20.84:3131/SicofMobileServiceADA/api/		Ambiente de desarrollo				
Calidad:			Ambiente de validación				
Producción:			Ambiente de producción				

Información Adicional

Despliegue

Requisitos Técnicos Servidor

- Servidor Wildfly 20 o superior
- Java 11 o OpenJDK 11
- S.O. Linux
- Ram 4GB o superior
- Disco Duro 64 GB
- Publicar la URL y Puerto del WildFly

Pasos Previos: Generar war (IDE Eclipse y Spring Tools)

El proyecto está configurado para generar compilable war. Siga los siguientes pasos para generarlo¹⁾.

- Ubíquese en la raíz del proyecto, click derecho → Run as... → Maven builds...
- En la casilla **Goals** ingrese el comando: **clean install** para generar el war. También puede utilizar en su lugar la instrucción **clean install -DskipTests** si no desea ejecutar las pruebas unitarias.
- Al finalizar el proceso si no hay errores se generará el war en el directorio target del proyecto.

Nota

Para ubicar el war en el directorio del equipo desde el proyecto seleccione el war con click derecho y vaya a propiedades al visualizar la ventana emergente se verá la ubicación del archivo.

Despliegue en el servidor

Actualmente solo hay servidor de desarrollo:

1. Acceder al servidor de despliegue por WSCP o SSH: IP: **10.1.20.84**, Usuario:**wildfly23**, Password:**vwd5ELGR5j7v**
2. Borre el archivo SicoMobileServiceADA.war y espere que el archivo de despliegue quede inválido.
3. Borre el archivo de despliegue SicoMobileServiceADA.war.undeploy
4. Copie el nuevo war en la ruta **/opt/wildfly23/standalone/deployments** se recomienda utilizar WSCP por su interfaz gráfica. Debe esperar mientras se sube el despliegue al servidor el cual terminará correctamente si se crea el archivo SicoMobileServiceADA.war.deployed Si falla este proceso debe ir al paso 2.
5. Por último desde un navegador ingrese la url **<http://10.1.20.84:3131/SicoMobileServiceADA/api/>** y deberá visualizar el mensaje **Online!!!** indicando que el despliegue fue realizado correctamente.

Notas

- Debe contar con VPN
- Ejemplo de Consumo Login Multiempresa:

http://10.1.20.84:3131/SicofMobileServiceADA/api/GetServicio?ping=desarrollo&nombreServicio=Login&callback=__ng_jsonp__.__req0.finished&separador=,¶metros=SICOF,1234,null,null,null&codigoMEmpresa=9999999999

- Ejemplo de consumo Empresas:

http://10.1.20.84:3131/SicofMobileServiceADA/api/GetServicio?ping=fosemed&nombreServicio=GetEmpresas&callback=__ng_jsonp__.__req0.finished&separador=,¶metros=

[←Volver atrás](#)

1)

Se asume que el lector tiene conocimientos previos del uso del lenguaje Java e IDE utilizado

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

Permanent link:
<http://wiki.adacsc.co/doku.php?id=ada:howto:sicoferp:factory:integrations:msconfigapps&rev=1647950507>

Last update: **2022/03/22 12:01**

