

Fábrica - Integraciones - Servicios Web - WsLogs

Este servicio tiene la funcionalidad de exponer endpoint para el registro de logs.

Descripción del Servicio

Nombre	Descripción	Tipo Servicio	Tecnología	Lenguaje	Tipo Desarrollo	Versión Desarrollo	Archivo de Configuración
Ada	Funcionalidad exponer microservicios para registro de logs	Rest	Spring Boot 4.4	Java 11.0.315	Nuevo	11.0.1.0	application.properties - config.properties
Tipo Empaquetado	Gestor de Dependencias	Servidor de Despliegue	IDE Proyecto	Tipo Repositorio	Url Desarrollo	Url Calidad	Url Producción
jar	Maven 3.8.1	Tomcat 6	Eclipse 2021-03	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-web		N/A	N/A	N/A	Core Servicios Web	
org.springframework.boot	spring-boot-devtools		N/A	runtime	true	Recarga de aplicaciones	
com.oracle.ojdbc	ojdbc8		N/A	runtime	N/A	Driver DB Oracle	
org.springframework.boot	spring-boot-starter-test		N/A	test	N/A	Core	
org.springframework.boot	spring-boot-starter-data-jdbc		N/A	N/A	N/A	API JDBC para conexión a Base de Datos	
org.apache.tomcat.embed	tomcat-embed-jasper		N/A	provided	N/A	Servidor embebido tomcat	
org.springframework.boot	spring-boot-starter-tomcat		N/A	provided	N/A	Core tomcat	
org.apache.maven.plugins	maven-compiler-plugin		3.8.1	N/A	N/A	Gestor de dependencias	
io.springfox	springfox-swagger2		2.9.2	N/A	N/A	Core Documentación Web Services RESTful	
io.springfox	springfox-swagger-ui		2.9.2	N/A	N/A	GUI Documentación Web Services RESTful	
Operaciones							
Tipo Petición Http	Nombre operación		Descripción			Documentación	
POST	/transactional/save		Se encarga de guardar el registro de una transacción.			Ver Contrato de Servicio	
GET	/transactional/getRegisters		Se encarga de obtener registros según el parámetro enviado.			Ver Contrato de Servicio	
GET	/transactional/{id}/getRegister		Se encarga de obtener un registro por ID.			Ver Contrato de Servicio	
POST	/session/save		Se encarga de guardar el registro de una sesión.			Ver Contrato de Servicio	
GET	/session/getRegisters		Se encarga de obtener registros según el parámetro enviado.			Ver Contrato de Servicio	
GET	/session/{id}/getRegister		Se encarga de obtener un registro por ID.			Ver Contrato de Servicio	
PUT	/session/{id}		Se encarga de actualizar un registro por ID, para cuando se realiza un finalizar sesión.			Ver Contrato de Servicio	
POST	/upload/save		Se encarga de guardar el registro al realizar una subida de archivo.			Ver Contrato de Servicio	
GET	/upload/getRegisters		Se encarga de obtener registros según el parámetro enviado.			Ver Contrato de Servicio	
POST	/error/save		Se encarga de guardar el registro de error.			Ver Contrato de Servicio	

Nombre	Descripción	Tipo Servicio	Tecnología	Lenguaje	Tipo Desarrollo	Versión Desarrollo	Archivo de Configuración
GET	/error/getRegisters		Se encarga de obtener registros según el parámetro enviado.			Ver Contrato de Servicio	
GET	/error/{id}/getRegister		Se encarga de obtener un registro por ID.			Ver Contrato de Servicio	
Ambientes de Despliegue							
Tipo	Url	Observaciones					
Desarrollo:	http://10.1.140.20:8001/	Ambiente de desarrollo					
Calidad:		Ambiente de validación					
Producción:		Ambiente de producción					

Información Adicional

Contrato de Servicio

A continuación se presenta el contrato de servicio con las operaciones implementadas. La documentación está realizada en el estandar [OpenAPI V3.0.1](#). Para visualizar la documentación (Vista html) copie el código [Yaml](#) en un visor [Swagger](#).

Para más información acerca de la estructura de los campos, puede consultar el [Diccionario de datos](#)

```
openapi: 3.0.1
info:
  title: Service Log
  contact:
    name: ADA.S.A
    url: https://ada.co/
    email: info@ada.co
  license:
    name: Apache License Version 2.0
    version: "1.0"
servers:
- url: //10.1.140.20:8001/
tags:
- name: file-upload-log-controller
  description: File Upload Log Controller
- name: mov-log-errores-controller
  description: Mov Log Errores Controller
- name: session-log-app
  description: Session Log App
- name: transactional-log-app
  description: Transactional Log App
paths:
  /error/getRegisters:
    get:
      tags:
      - mov-log-errores-controller
      summary: getRegisters
```

```
operationId: getRegistersUsingGET_1
parameters:
- name: codigoInterno
  in: query
  description: codigoInterno
  schema:
    type: integer
    format: int32
    default: 0
- name: codOracle
  in: query
  description: codOracle
  schema:
    type: integer
    format: int32
    default: 0
- name: msjOracle
  in: query
  description: msjOracle
  schema:
    type: string
    default: "0"
- name: nomPrecedimiento
  in: query
  description: nomPrecedimiento
  schema:
    type: string
    default: "0"
responses:
  200:
    description: OK
    content:
      '*/*':
        schema:
          type: array
          items:
            $ref: '#/components/schemas/MovLogErrores'
  401:
    description: Unauthorized
    content: {}
  403:
    description: Forbidden
    content: {}
  404:
    description: Not Found
    content: {}
/error/save:
  post:
    tags:
      - mov-log-errores-controller
    summary: save
```

```
operationId: saveUsingPOST
requestBody:
  description: sessionLog
  content:
    application/json:
      schema:
        $ref: '#/components/schemas/MovLogErrores'
  required: true
responses:
  200:
    description: OK
    content:
      '*/*':
        schema:
          type: integer
          format: int32
  201:
    description: Created
    content: {}
  401:
    description: Unauthorized
    content: {}
  403:
    description: Forbidden
    content: {}
  404:
    description: Not Found
    content: {}
x-codegen-request-body-name: sessionLog
/error/{id}/getRegister:
get:
  tags:
    - mov-log-errores-controller
  summary: getRegisterById
  operationId: getRegisterByIdUsingGET
  parameters:
    - name: id
      in: path
      description: id
      required: true
      schema:
        type: string
  responses:
    200:
      description: OK
      content:
        '*/*':
          schema:
            $ref: '#/components/schemas/MovLogErrores'
    401:
```

```
    description: Unauthorized
    content: {}
  403:
    description: Forbidden
    content: {}
  404:
    description: Not Found
    content: {}
/session/getRegisters:
  get:
    tags:
      - session-log-app
    summary: getRegisters
    operationId: getRegistersUsingGET_2
    parameters:
      - name: codigoAplicacion
        in: query
        description: codigoAplicacion
        schema:
          type: integer
          format: int64
      - name: codigoDependencia
        in: query
        description: codigoDependencia
        schema:
          type: integer
          format: int64
      - name: codigoUsuario
        in: query
        description: codigoUsuario
        schema:
          type: integer
          format: int64
      - name: id
        in: query
        description: id
        schema:
          type: string
      - name: nitEmpresa
        in: query
        description: nitEmpresa
        schema:
          type: integer
          format: int64
      - name: tipoRegistro
        in: query
        description: tipoRegistro
        schema:
          type: string
    responses:
      200:
```

```
    description: OK
    content:
      '*/*':
        schema:
          type: array
          items:
            $ref: '#/components/schemas/SessionLog'
  401:
    description: Unauthorized
    content: {}
  403:
    description: Forbidden
    content: {}
  404:
    description: Not Found
    content: {}
/session/save:
  post:
    tags:
      - session-log-app
    summary: saveSessionLog
    operationId: saveSessionLogUsingPOST_1
    requestBody:
      description: sessionLog
      content:
        application/json:
          schema:
            $ref: '#/components/schemas/SessionLog'
      required: true
    responses:
      200:
        description: OK
        content:
          '*/*':
            schema:
              type: string
      201:
        description: Created
        content: {}
      401:
        description: Unauthorized
        content: {}
      403:
        description: Forbidden
        content: {}
      404:
        description: Not Found
        content: {}
    x-codegen-request-body-name: sessionLog
/session/{id}:

```

```
put:
  tags:
    - session-log-app
  summary: updateRegister
  operationId: updateRegisterUsingPUT
  parameters:
    - name: id
      in: path
      description: id
      required: true
  schema:
    type: string
  requestBody:
    description: sessionLog
    content:
      application/json:
        schema:
          $ref: '#/components/schemas/SessionLog'
    required: true
  responses:
    200:
      description: OK
      content:
        '*/*':
          schema:
            $ref: '#/components/schemas/SessionLog'
    201:
      description: Created
      content: {}
    401:
      description: Unauthorized
      content: {}
    403:
      description: Forbidden
      content: {}
    404:
      description: Not Found
      content: {}
  x-codegen-request-body-name: sessionLog
/transactional/getRegisters:
  get:
    tags:
      - transactional-log-app
    summary: getRegisters
    operationId: getRegistersUsingGET_3
    parameters:
      - name: codigoUsuario
        in: query
        description: codigoUsuario
    schema:
      type: integer
```

```
    format: int64
- name: id
  in: query
  description: id
  schema:
    type: integer
    format: int64
- name: logColumn
  in: query
  description: logColumn
  schema:
    type: string
- name: logTabla
  in: query
  description: logTabla
  schema:
    type: string
- name: nitEmpresa
  in: query
  description: nitEmpresa
  schema:
    type: integer
    format: int64
- name: nombreProceso
  in: query
  description: nombreProceso
  schema:
    type: string
responses:
  200:
    description: OK
    content:
      '*/*':
        schema:
          type: array
          items:
            $ref: '#/components/schemas/TransactionalLog'
  401:
    description: Unauthorized
    content: {}
  403:
    description: Forbidden
    content: {}
  404:
    description: Not Found
    content: {}
/transactional/save:
  post:
    tags:
      - transactional-log-app
```

```
summary: saveTransactional
operationId: saveTransactionalUsingPOST
requestBody:
  description: log
  content:
    application/json:
      schema:
        $ref: '#/components/schemas/TransactionalLog'
  required: true
responses:
  200:
    description: OK
    content:
      '*/*':
        schema:
          type: string
  201:
    description: Created
    content: {}
  401:
    description: Unauthorized
    content: {}
  403:
    description: Forbidden
    content: {}
  404:
    description: Not Found
    content: {}
x-codegen-request-body-name: log
/transactional/{id}/getRegister:
get:
  tags:
    - transactional-log-app
  summary: getRegisterById
  operationId: getRegisterByIdUsingGET_1
  parameters:
    - name: id
      in: path
      description: id
      required: true
      schema:
        type: integer
        format: int64
  responses:
    200:
      description: OK
      content:
        '*/*':
          schema:
            $ref: '#/components/schemas/TransactionalLog'
    401:
```

```
    description: Unauthorized
    content: {}
403:
    description: Forbidden
    content: {}
404:
    description: Not Found
    content: {}
/upload/getRegisters:
  get:
    tags:
      - file-upload-log-controller
    summary: getRegisters
    operationId: getRegistersUsingGET
    parameters:
      - name: id
        in: query
        description: id
      schema:
        type: integer
        format: int32
        default: 0
      - name: nombreArchivo
        in: query
        description: nombreArchivo
      schema:
        type: integer
        format: int32
        default: 0
      - name: nombreProceso
        in: query
        description: nombreProceso
      schema:
        type: string
        default: "0"
    responses:
      200:
        description: OK
        content:
          '*/*':
            schema:
              type: array
              items:
                $ref: '#/components/schemas/FileUploadLog'
      401:
        description: Unauthorized
        content: {}
      403:
        description: Forbidden
        content: {}
```

```
    404:
      description: Not Found
      content: {}
/upload/save:
  post:
    tags:
      - file-upload-log-controller
    summary: saveSessionLog
    operationId: saveSessionLogUsingPOST
    requestBody:
      description: sessionLog
      content:
        application/json:
          schema:
            $ref: '#/components/schemas/FileUploadLog'
      required: true
    responses:
      200:
        description: OK
        content:
          '*/*':
            schema:
              type: integer
              format: int32
      201:
        description: Created
        content: {}
      401:
        description: Unauthorized
        content: {}
      403:
        description: Forbidden
        content: {}
      404:
        description: Not Found
        content: {}
    x-codegen-request-body-name: sessionLog
components:
  schemas:
    FileUploadLog:
      title: FileUploadLog
      type: object
      properties:
        codigoAplicacion:
          type: integer
          format: int32
        codigoMEmpresa:
          type: string
        codigoUsuario:
          type: integer
          format: int32
```

```
columnaGrupo:
  type: string
fecha:
  type: string
  format: date-time
fechaRegistro:
  type: string
  format: date-time
hostCliente:
  type: string
id:
  type: integer
  format: int32
infoApp:
  type: string
nitEmpresa:
  type: integer
  format: int64
nombreArchivo:
  type: string
nombreProceso:
  type: string
notificacion:
  type: string
numeroLinea:
  type: integer
  format: int32
sessionBrowserVersion:
  type: string
sessionMac:
  type: string
sessionOstype:
  type: string
MovLogErrores:
  title: MovLogErrores
  type: object
properties:
  codRacle:
    type: integer
    format: int32
  codigoAplicacion:
    type: string
  codigoEmpresa:
    type: string
  codigoInterno:
    type: integer
    format: int32
  codigoUsuario:
    type: integer
    format: int64
```

```
    errorLog:
      type: string
    errorType:
      type: string
    fecha:
      type: string
      format: date-time
    fechaRegistro:
      type: string
      format: date-time
    hostCliente:
      type: string
    infoApp:
      type: string
    msjRacle:
      type: string
    nitEmpresa:
      type: integer
      format: int64
    nomProcedimiento:
      type: string
    observacion:
      type: string
    sessionBrowserVersion:
      type: string
    sessionMac:
      type: string
    sessionOstype:
      type: string
    usuarioBD:
      type: string
  SessionLog:
    title: SessionLog
    type: object
    properties:
      codigoAplicacion:
        type: integer
        format: int64
      codigoDependencia:
        type: integer
        format: int64
      codigoEmpresa:
        type: number
        format: double
      codigoUsuario:
        type: integer
        format: int64
      fechaLogout:
        type: string
        format: date-time
      fechaRegistro:
```

```
    type: string
    format: date-time
fechaSessionActiva:
    type: string
    format: date-time
id:
    type: string
nitEmpresa:
    type: integer
    format: int64
sessionBrowserVersion:
    type: string
sessionId:
    type: string
sessionMac:
    type: string
sessionOsType:
    type: string
tipoApp:
    type: string
tipoRegistro:
    type: string
usuarioDb:
    type: string
TransactionalLog:
    title: TransactionalLog
    type: object
    properties:
        codigoAplicacion:
            type: integer
            format: int64
        codigoEmpresa:
            type: string
        codigoUsuario:
            type: integer
            format: int64
        fecha:
            type: string
            format: date-time
        fechaRegistro:
            type: string
            format: date-time
        hostCliente:
            type: string
        id:
            type: integer
            format: int64
        infoApp:
            type: string
        logColumna:
```

```
    type: string
  logPeticion:
    type: string
  logTabla:
    type: string
  logValorAct:
    type: string
  logValorAnt:
    type: string
  loginUsuario:
    type: string
  nitEmpresa:
    type: integer
    format: int64
  nombreProceso:
    type: string
  sessionBrowserVersion:
    type: string
  sessionMac:
    type: string
  sessionOsType:
    type: string
  tipoTransaccion:
    type: string
```

[←Volver atrás](#)

From:

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

Permanent link:

<http://wiki.adacsc.co/doku.php?id=ada:howto:sicoferp:factory:integrations:logs>



Last update: **2021/08/10 22:24**