

Fábrica - Servicios Web - Integraciones - AdaEmailService

Este servicio permite el envío de correos electrónicos. Actualmente soporta los siguientes servidores de correo:

- Ionos
- Gmail*
- Outlook

*Requiere generación de clave de aplicaciones

Descripción del Servicio

Nombre	Descripción	Tipo Servicio	Tecnología	Lenguaje	Tipo Desarrollo	Versión Desarrollo	Archivo de Configuración
AdaEmailServie	Permite el envío de correos electrónicos.	Rest	SpringBoot 2.6.4	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-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		
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		
com.sun.mail	javax.mail	1.6.2	N/A	N/A	Envío de correos		
Documentación Operaciones							
Tipo	Documentación						Visor
Swagger	Ver Documento Json						https://editor.swagger.io/
Javadoc	Directorio Javadoc						Navegador Web
Ambientes de Despliegue							
Tipo	Url		Observaciones				
Desarrollo:	http://localhost:8080		Ambiente de desarrollo				
Calidad:	http://srv-appeon-000-w23.adacsc.co/AdaEmailService		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

Notas

- Debe contar con VPN
- Ejemplo de Consumo QA - Index: <http://srv-appeon-000-w23.adacsc.co/AdaEmailService>
- Ejemplo de consumo QA - Doc Swagger: <http://srv-appeon-000-w23.adacsc.co/AdaEmailService/api-docs>

Ejemplos de envío de correos

Envío servidor Ionos

```
{
  "smtpHost": "smtp.ionos.com",
  "smtpPort": 587,
  "smtpAuth": "true",
  "smtpMailUser": "email.test@ada.co",
  "smtpMailPassword": "password",
  "smtpStartTlsEnabled": false,
  "smtpSessionDebug": true,
  "subject": "Título de Prueba",
  "content": "Contenido de prueba",
  "html": false,
  "toEmail": [
    {
      "address": "notificacionTo.test@ada.co",
      "persona": "Notificación To"
    }
  ],
  "ccEmail": [
    {
      "address": "notificacionCc.test@ada.co",
      "persona": "Notificación Cc"
    }
  ],
  "bccEmail": [
    {
      "address": "notificacionBcc.test@ada.co",
      "persona": "Notificación Bcc"
    }
  ]
}
```

```
}
```

Envio servidor Outlook

```
{
  "smtpHost": "smtp.outlook.com",
  "smtpPort": 587,
  "smtpAuth": "true",
  "smtpMailUser": "correo",
  "smtpMailPassword": "password",
  "smtpStartTlsEnabled": true,
  "smtpSessionDebug": true,
  "subject": "Prueba",
  "content": "Ejemplo",
  "html": false,
  "toEmail": [
    {
      "address": "correo",
      "persona": "persona"
    }
  ],
  "ccEmail": [
    {
      "address": "correo copia",
      "persona": "Persona Copia"
    }
  ],
  "bccEmail": [
    {
      "address": "Correo copia oculta",
      "persona": "Persona copia oculta"
    }
  ]
}
```

Envio servidor Gmail

```
{
  "smtpHost": "smtp.gmail.com",
  "smtpPort": 587,
  "smtpAuth": "true",
  "smtpMailUser": "correo",
  "smtpMailPassword": "clave aplicacion",
  "smtpStartTlsEnabled": true,
  "smtpSessionDebug": true,
  "subject": "Prueba",
  "content": "Ejemplo",
  "html": false,
```

```
"toEmail": [  
  {  
    "address": "Correo",  
    "persona": "Persona"  
  }  
],  
"ccEmail": [  
  {  
    "address": "Correo copia",  
    "persona": "Persona copia"  
  }  
],  
"bccEmail": [  
  {  
    "address": "Correo copia oculta",  
    "persona": "Persona copia oculta"  
  }  
]  
]
```

Liberación Versión 2.0.0 Correos Masivos

Se libera versión con soporte de envíos masivos sin necesidad de servidor SMTP.

Documentación Técnica Javadoc

Se libera la documentación técnica javadoc la cual puede ser consultada desde la carpeta de documentación del proyecto en la ruta del [SVN](#)

Nota

Debe descargar el proyecto para visualizar la documentación.

Documentación Técnica OpenAPI

Se libera documentación OpenAPI V3 la cual se comparte a continuación y puede se utiliza en el [visor swagger](#)

```
openapi: 3.0.1  
info:  
  title: OpenAPI definition  
  version: v0  
servers:  
  - url: http://localhost:8088
```

```
description: Generated server url
tags:
  - name: EmailController
    description: Controlador que gestiona los envíos de correos electrónicos
    en las aplicaciones.
paths:
  /email/v2/sendMasive:
    post:
      tags:
        - EmailController
      summary: sendMasive
      description: Envía correos electrónicos de forma masiva.
      operationId: sendMasive
      parameters:
        - name: token
          in: header
          description: Token API utilizado para realiza el envío. Si no se
            especifica se utiliza uno genérico.
          required: false
          schema:
            type: string
            example: eyFDSFDSF4%$#%$#%VFDvcxvkxkvljxkljelkrjklcjvklcxjvklxjv
        - name: emailDto
          in: query
          description: Objeto con la configuración del envio y contenido del
            correo electrónico que será enviado
          required: true
          schema:
            $ref: '#/components/schemas/EmailDto'
      responses:
        '202':
          description: Correo electrónico enviado correctamente.
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/GenericResponseDto'
        '500':
          description: Se han generado inconsistencias al crear el registro.
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/GenericResponseDto'
        '503':
          description: No se puede enviar el correo electrónico.
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/GenericResponseDto'
  /email/v1/send:
    post:
      tags:
```

```
- EmailController
summary: send
description: Envía correos electrónicos.
operationId: send
parameters:
  - name: emailConfigDto
    in: query
    description: Objeto con la configuración del envío y contenido del correo electrónico que será enviado
    required: true
    schema:
      $ref: '#/components/schemas/EmailConfigDto'
example:
  smtpHost: smtp.ionos.com
  smtpPort: 587
  smtpAuth: 'true'
  smtpMailUser: email.test@ada.co
  smtpMailPassword: password
  smtpStartTlsEnabled: false
  smtpSessionDebug: true
  subject: Título de Prueba
  content: Contenido de prueba
  html: false
  toEmail:
    - address: notificacionTo.test@ada.co
      persona: Notificación To
  ccEmail:
    - address: notificacionCc.test@ada.co
      persona: Notificación Cc
  bccEmail:
    - address: notificacionBcc.test@ada.co
      persona: Notificación Bcc
responses:
  '202':
    description: Correo electrónico enviado correctamente.
    content:
      application/json:
        schema:
          $ref: '#/components/schemas/GenericResponseDto'
  '500':
    description: Se han generado inconsistencias al crear el registro.
    content:
      application/json:
        schema:
          $ref: '#/components/schemas/GenericResponseDto'
  '503':
    description: No se puede enviar el correo electrónico.
    content:
      application/json:
        schema:
```

```
    $ref: '#/components/schemas/GenericResponseDto'
  deprecated: true
  /email/:
    get:
      tags:
        - EmailController
      summary: index
      description: Método de validación de estado del microservicio
      operationId: index
      responses:
        '200':
          description: Microsericio en linea
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/EmailController'
  components:
    schemas:
      AddressConfigDto:
        type: object
        properties:
          address:
            type: string
          persona:
            type: string
        description: Clase Dto que contiene la configuración de la dirección de correo electrónico.
      AttachmentConfigDto:
        type: object
        properties:
          fileName:
            type: string
          fileContentStringBase64:
            type: string
          typeContent:
            type: string
        description: Clase Dto que contiene la configuración del archivo que se adjuntará al correo electrónico.
      EmailDto:
        type: object
        properties:
          from:
            $ref: '#/components/schemas/AddressConfigDto'
          subject:
            type: string
          content:
            type: string
          html:
            type: boolean
          attachmentBase64:
            type: array
```

```
    items:
      $ref: '#/components/schemas/AttachmentConfigDto'
  toEmail:
    type: array
    items:
      $ref: '#/components/schemas/AddressConfigDto'
  ccEmail:
    type: array
    items:
      $ref: '#/components/schemas/AddressConfigDto'
  bccEmail:
    type: array
    items:
      $ref: '#/components/schemas/AddressConfigDto'
  description: Clase Dto que contiene la configuración e información del correo electrónico.
  GenericResponseDto:
    type: object
    properties:
      code:
        type: integer
        format: int64
      message:
        type: string
      content:
        type: object
    description: Dto generico utilizado para respuestas en servicios.
  EmailConfigDto:
    type: object
    properties:
      smtpHost:
        type: string
      smtpPort:
        type: string
      smtpAuth:
        type: string
      smtpMailUser:
        type: string
      smtpMailPassword:
        type: string
      smtpStartTlsEnabled:
        type: string
      smtpSslTrust:
        type: string
      smtpSessionDebug:
        type: boolean
      subject:
        type: string
      content:
        type: string
```

```
    html:
      type: boolean
    attachmentBase64:
      type: array
      items:
        $ref: '#/components/schemas/AttachmentConfigDto'
    toEmail:
      type: array
      items:
        $ref: '#/components/schemas/AddressConfigDto'
    ccEmail:
      type: array
      items:
        $ref: '#/components/schemas/AddressConfigDto'
    bccEmail:
      type: array
      items:
        $ref: '#/components/schemas/AddressConfigDto'
  description: Clase Dto que contiene las configuraciones de
  autenticación y envío del correo electrónico.
  EmailController:
    type: object
```

Ejemplo de Consumo

El siguiente es un ejemplo de consumo del nuevo servicio. Los parametros de consumo se describen como se configurarían en un cliente REst como Postman.

```
//Http: POST
//URL: localhost:8088/email/v2/sendMasive
//Header: token = Token API utilizado para realiza el envío. Si no se
especifica se utiliza uno genérico.
//Parametro en el Body: Ver el json a continuación

{
  "from": {
    "address": "dirección de correo de origen",
    "persona": "Nombre de la cuenta de origen"
  },
  "subject": "Asunto del correo",
  "content": "Contenido del correo, puede ser un bloque html o texto plano",
  "html": true,
  "attachmentBase64": [
    {
      "fileName": "Nombre del archivo con extensión",
      "fileContentStringBase64": "Contenido del archivo en Base64",
      "typeContent": "Mime type del archivo Ej: Para un PDF sería
application/pdf"
    }
  ],
}
```

```
"toEmail": [  
  {  
    "address": "Correo de destino",  
    "persona": "Nombre asociado al correo de destino."  
  }  
],  
"ccEmail": [  
  {  
    "address": "Correo de copia",  
    "persona": "Nombre asociado al correo de copia."  
  }  
],  
"bccEmail": [  
  {  
    "address": "Correo de copia oculta",  
    "persona": "Nombre asociado al correo de copoa oculta."  
  }  
]  
}
```

[←Volver atrás](#)

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

Permanent link:
<http://wiki.adacsc.co/doku.php?id=ada:howto:sicoferp:factory:integrations:msadaemailservice&rev=1685739667>

Last update: 2023/06/02 21:01

