



RUN

Reductor de URLs

Manual de API REST

Especificaciones del servicio de API de RUN.

Versión

1.1-1.4

Fecha de revisión

26/05/2023

Realizado por

Equipo RUN & Sist. Desarrollo



1 ÍNDICE

1	ÍNDICE	2
2	HISTÓRICO DE VERSIONES	3
3	OBJETIVO DEL DOCUMENTO.....	4
	Tabla resumen del estado actual de los endpoints	4
4	REQUISITOS	5
4.1	CONSIDERACIONES GENERALES.....	5
5	SERVICIOS API REST DE RUN	6
5.1	APPLICATIONUSER.....	6
5.1.1	POST /api/login_check.....	6
5.2	QR	7
5.2.1	POST - /api/v1/qr.....	7
5.3	REFRESH_TOKEN	9
5.3.1	POST - /api/v1/token/refresh	9
5.4	URL	10
5.4.1	POST - /api/v1/url/edit	10
5.4.2	GET - /api/v1/urls-list/{groupId}/{page}/{pageSize}.....	12
5.4.3	POST - /api/v1/url	13
5.4.4	DELETE - /api/v1/url/delete	15
5.5	USER.....	16
5.5.1	GET - /api/v1/user/details	16
5.6	GROUPBYUSER	16
5.6.1	GET - /api/v1/user/groups.....	17
5.7	DOCUMENTACIÓN (DEFAULT)	17
5.7.1	GET - /api/doc.json	17
5.7.2	POST - /api/v1/events/dispatch.....	18



2 Histórico de versiones

Versión Documento	Versión RUN	Fecha	Revisado por	Descripción														
1.1-1.0	1.1.1	12/07/2022	Equipo RUN	Elaboración de la plantilla inicial del documento. Incorporación de información inicial.														
1.1-1.1	1.1.1	12/07/2022	Equipo RUN	Incorporación de información.														
1.1-1.2	1.1.1	27/09/2022	Equipo RUN	Actualización de la información de las API.														
1.1-1.3	1.1.1	28/09/2022	Equipo RUN	<div>Actualización de la información de las API según nuevas correcciones e</div> <div>Se incorpora <i>endpoint</i> User</div> <div>Este servicio contiene el <i>endpoint</i> responsable de mostrar l</div> <div>2.1.1 GET - /api/v1/user/details</div> <div>Petición - Estructura de campos</div> <div>HEADER</div> <table><thead><tr><th>Parámetro</th><th>Tipo</th><th>Obligatorio</th></tr></thead><tbody><tr><td>Authorization</td><td>string</td><td>NO</td></tr></tbody></table> <div>Ejemplo de Petición:</div> <table><tr><td>Método</td><td>GET</td></tr><tr><td>header</td><td>Authorization: Bearer eyJ0eXAiOiJKV1QiLC</td></tr></table> <div>Respuesta – Estructura de campos</div> <table><thead><tr><th>Códigos HTTP de respuesta</th><th>Responses model</th></tr></thead><tbody><tr><td>200</td><td><pre>{ "status": "success", "data": { "userName": "RICARDO", "app-name": "RICARDO", "roles": "ROLE_USER", "Auth-method": "password" } }</pre></td></tr></tbody></table>	Parámetro	Tipo	Obligatorio	Authorization	string	NO	Método	GET	header	Authorization: Bearer eyJ0eXAiOiJKV1QiLC	Códigos HTTP de respuesta	Responses model	200	<pre>{ "status": "success", "data": { "userName": "RICARDO", "app-name": "RICARDO", "roles": "ROLE_USER", "Auth-method": "password" } }</pre>
Parámetro	Tipo	Obligatorio																
Authorization	string	NO																
Método	GET																	
header	Authorization: Bearer eyJ0eXAiOiJKV1QiLC																	
Códigos HTTP de respuesta	Responses model																	
200	<pre>{ "status": "success", "data": { "userName": "RICARDO", "app-name": "RICARDO", "roles": "ROLE_USER", "Auth-method": "password" } }</pre>																	



				<div>500</div> <div>{ "status": "error", "data": "string" }</div>
				GroupByUser
1.1-1.4	1.1.1	26/05/2023	Equipo RUN	<p>Actualización de la información de las API según nuevas correcciones e</p> <p>Actualización de los apartados:</p> <ul style="list-style-type: none">• Tabla resumen del estado actual de los endpoints• GET - /api/v1/urls-list/{groupId}/{page}/{pageSize} <p>Se incorpora los siguientes endpoints:</p> <ul style="list-style-type: none">• RefreshToken → POST - /api/v1/token/refresh• POST - /api/v1/url/edit• DELETE - /api/v1/url/delete• User → GET - /api/v1/user/details• POST - /api/v1/events/dispatch

Tabla 1. Histórico de versiones



3 Objetivo del documento

Este documento describe los *endpoints* que RUN ofrece para su gestión a través de la interfaz de programación de aplicaciones API REST.

Tabla resumen del estado actual de los *endpoints*

Para que los usuarios de API REST de RUN tengan una mayor información en cuanto al estado de los *endpoints*, a continuación, se presenta la siguiente tabla resumen:

Leyenda:

Disponible: Son aquellos *endpoint* que están disponibles y activos.

En desuso: *Endpoint* que están disponibles, pero se desaconseja su uso.

En proceso de actualización: *Endpoint* que están siendo actualizados y en próximas versiones se verán reflejados dichos cambios o correcciones.

En desarrollo: Nuevos *endpoints* que se están creando para poner a disposición de los usuarios. Estarán disponibles en próximas versiones.

Servicio	Endpoint	Estado actual
ApplicationUser	/api/login_check	Disponible
Qr	/api/v1/qr	Disponible
RefreshToken	/api/v1/token/refresh	Disponible
Url	/api/v1/url/edit	Disponible
	/api/v1/urls-list/{groupId}/{page}/{pageSize}	Disponible
	/api/v1/url	Disponible
	/api/v1/url/delete	Disponible
User	/api/v1/user/details	Disponible
GroupByUser	/api/v1/user/groups	Disponible
default	/api/doc.json	Disponible
	/api/v1/events/dispatch	Disponible



RUN-API <small>1.0.0</small>	
API para RUN	
ApplicationUser ▾	
POST	/api/login_check
Qr ▾	
POST	/api/v1/qr
RefreshToken ▾	
POST	/api/v1/token/refresh
Url ▾	
POST	/api/v1/url/edit
GET	/api/v1/urls-list/{sgroupId}/{page}/{pageSize}
POST	/api/v1/url
DELETE	/api/v1/url/delete
User ▾	
GET	/api/v1/user/details
GroupByUser ▾	
GET	/api/v1/user/groups
default ▾	
GET	/api/doc.json
POST	/api/v1/events/dispatch

4 Requisitos

4.1 Consideraciones generales

Para poder hacer uso del catálogo de servicios de RUN es necesario que se solicite usuario a través del CAID (Centro de Ayuda a Integradores y Desarrolladores), usando la siguiente URL:

<https://ssweb.seap.minhap.es/ayuda/consulta/run>

En la solicitud se debe indicar el tipo de incidencia.

Para la correcta integración con el API REST, se ofrece una página web centralizada desde donde se puede acceder al catálogo de los servicios activo. La URL para poder ingresar al catálogo en el entorno de preproducción es la siguiente:

<https://pre-run-api.redsara.es/api/doc>

Además, se ofrece la URL con la descripción del catálogo basada en el estándar Swagger 2.0, en formato Json:

<https://pre-run-api.redsara.es/api/doc.json>

Después de la correcta integración en el entorno de preproducción, los responsables de la aplicación integradora tendrán que solicitar credenciales de acceso para el entorno de producción.

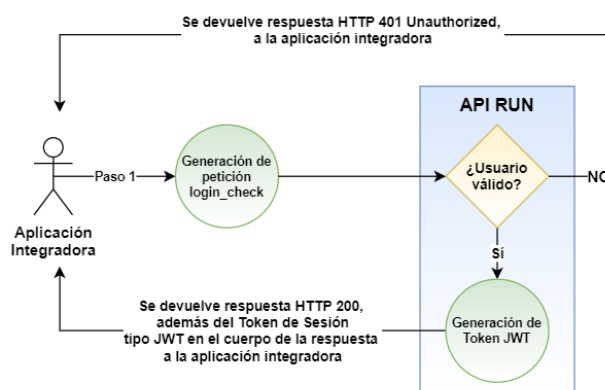


5 Servicios API REST de RUN

5.1 AplicationUser

Este servicio permite autenticarse dentro de la aplicación RUN.

A continuación, se describe el proceso de autenticación en el API:



Este servicio ofrece el siguiente endpoint:

5.1.1 POST /api/login_check

Petición - Estructura de campos

HEADER

Parámetro	Tipo	Obligatorio	Descripción
Auth-Signature	string	SI	Permite añadir la firma de autorización para ingresar a la app. NOTA: No es necesario añadir cabeceras adicionales.

MESSAGE BODY

Formato	Parámetro	Tipo	Obligatorio	Descripción
Application/json	applicationId	string	SI	Identificador de la aplicación.
	date	string	SI	Fecha y hora de creación del certificado. Debe ir en el siguiente formato YYYY-MM-DD HH:mm:ss



Ejemplo de Petición:

Método	POST
header	Auth-Signature:GT8ptcJeu9czt8yiLHsGMTXIy6Wgoyms2e5SLYpTaHppA...
body	{ "applicationId": "apptest", "date": "2022-09-27 13:20:34" }

Respuesta – Estructura de campos

Códigos HTTP de respuesta	Objeto de Respuesta	Ejemplo de respuesta	Descripción
200	{ "status": "success", "data": { "token": "string" } }	{ "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzUxMiJ9..." }	Usuario logado correctamente. Token de sesión de tipo JWT usado para realizar peticiones sobre el resto de los métodos del API
401		{ "status": "fail", "data": { "message": "string" } }	Información Corrupta. Las credenciales no son correctas. Usuario no autorizado
500			Usuario NO logado. Ocurrió un error durante el proceso de autenticación

5.2 Qr

Este servicio genera una imagen QR que contiene la url indicada en la petición. Además, recibe un parámetro opcional para descargar la imagen codificada en Base64 o como fichero binario.

5.2.1 POST - /api/v1/qr

Petición - Estructura de campos

HEADER

Parámetro	Tipo	Obligatorio	Descripción
Authorization	string	NO	Es necesario indicar el Token de sesión tipo Bearer en la cabecera Authorization.



			IMPORTANTE: El token debe ir precedido de la palabra Bearer y un espacio.
--	--	--	--

MESSAGE BODY

Formato	Parámetro	Tipo	Obligatorio	Descripción
Application/json	surl	string	SI	Contiene una surl existente de RUN.
	getImage	boolean	NO	Parámetro que indica si se desea descargar la imagen del QR en binario (true) o en Base64 (false). Por defecto se descarga en Base64.

Ejemplo de Petición:

Método	POST
header	Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzUxMiJ9...
body	{ "surl": "kzjd2454", "getImage": "true" }

Respuesta – Estructura de campos

Códigos HTTP de respuesta	Responses model	Descripción
201	{ "status": "success", "data": { "image_base64": "string" } }	Código QR generado. Si el parámetro getImage es <i>True</i> se descargará directamente la imagen.
401	{ "code": "401", "message": "Expired JWT Token" }	JWR Token ha expirado. La sesión ha expirado.
402	{ "status": "fail", "data": "string" }	Información corrupta. Los datos de entrada no son correctos o presentan errores.
500	{ "status": "error", "data": "string" }	Ocurrió un error durante el proceso.



5.3 RefreshToken

Esta sección contiene el *endpoint* responsable de crear un nuevo token para logarse.

5.3.1 POST - /api/v1/token/refresh

Permite obtener un nuevo *token* JWT utilizando para ello el *refresh token* obtenido en el momento del inicio de la sesión. También ofrece un nuevo *refresh token* que podrá ser utilizado posteriormente.

Petición - Estructura de campos

MESSAGE BODY

Formato	Parámetro	Tipo	Obligatorio	Descripción
Application/json	Refresh_token	string	SI	Nuevo token para logarse.

Ejemplo de Petición:

Método	POST
header	Authorization: "Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJz...
path	/api/v1/token/refresh
body	{ "refresh_token": "c48db68233c5677bb633e342790036ba151322d2435c15b0ccddc0c62526410778e02c300cf4ae4a76f074664bb346deb502296d5b9d526711b767e3fa95d2a1" }

Respuesta – Estructura de campos

Códigos HTTP de respuesta	Responses model	Descripción
200	{ "status": "success", "data": { "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9." }, "refresh_token": "c48db68233c5677bb633e342790036ba151322d2435c15b0ccddc0c62526410778e02c300cf4ae4a76f074664bb346deb502296d5b9d526711b767e3fa95d2a1" }	<i>Refresh token.</i> Nuevo token creado correctamente.
401	{ "status": "fail", "data": { "message": "string" } }	<i>Información corrupta. Las credenciales no son válidas.</i>



500	<pre>{ "status": "error", "data": { "message": "error message" } }</pre>	<i>Returned on any other error.</i> Ocurrió un error durante el proceso.
-----	--	---

5.4 URL

Este servicio contiene los *endpoints* responsables de la generación de urls cortas a partir de una lista de urls suministradas en el cuerpo de la petición.

5.4.1 POST - /api/v1/url/edit

Este *endpoint* permite editar una URL corta ya existente.

Petición - Estructura de campos

HEADER

Parámetro	Tipo	Obligatorio	Descripción
Authorization	string	NO	Es necesario indicar el Token de sesión tipo Bearer en la cabecera Authorization. IMPORTANTE: El token debe ir precedido de la palabra Bearer y un espacio.

MESSAGE BODY

Formato	Parámetro	Tipo	Obligatorio	Descripción
Application/json	-	array	SI	Colección de objetos que contienen una url a acortar y opcionalmente la cadena de texto a usar para la url acortada.

Estructura de campo.

Parámetro	Tipo	Obligatorio	Descripción
surl	string	NO	Recoge una lista con las propuestas de la urls acortadas (propuesta opcional).
url	string	SI	Recoge las urls (ruta larga) que luego serán devueltas de manera acortada. Debe contener una lista json con las urls a acortar.
description	string	NO	Permite añadir una descripción del porqué de la edición de la URL acortada.
active	boolean	SI	Permite activar la URL editada.



Ejemplo de Petición:

Método	POST
header	Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzUxMiJ9...
body	<pre>{ "surl": "string", "url": "string", "description": "string", "active": "boolean" }</pre>

Respuesta – Estructura de campos

Códigos HTTP de respuesta	Responses model	Descripción
201	<pre>{ "status": "string", "data": { "sUrl": "string", "url": "string", "description": "string", "active": "boolean" } }</pre>	Operación correcta y url dada de alta. Se generan las urls acortadas suministradas, o se devuelve respuesta de tipo 2XX indicando que la url ya se generó previamente.
401	<pre>{ "code": "401", "message": "Expired JWT Token" }</pre>	JWR Token ha expirado. La sesión ha expirado.
402	<pre>{ "status": "fail", "data": "string" }</pre>	Información corrupta. Los datos de entrada no son correctos o presentan errores.
500	<pre>{ "status": "error", "data": "string" }</pre>	Ocurrió un error durante el proceso.



5.4.2 GET - /api/v1/urls-list/{groupId}/{page}/{pageSize}

Petición - Estructura de campos

HEADER

Método	Parámetro	Tipo	Obligatorio	Descripción
Bearer Token	Authorization – Bearer Token	string	NO	Es necesario indicar el Token de sesión tipo Bearer en la cabecera Authorization. IMPORTANTE: El token debe ir precedido de la palabra Bearer y un espacio.

PATH

Parámetro	Tipo	Obligatorio	Descripción
sGroupId	string	SI	Identificador del grupo (slug) del que se requiere listar urls.
page	integer	SI	Número de página para visualizar. Si no se añade un valor, por defecto el sistema mostrará la página 1.
pageSize	integer	SI	Número de resultados a mostrar por página. Por defecto se muestran 10 resultados por página.

Ejemplo de Petición:

Método	GET
header	Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzUxMiJ9...
path	/api/v1/urls-list/vri/4/13

Respuesta – Estructura de campos

Códigos HTTP de respuesta	Responses model	Descripción
201	{ "status": "success", "data": [{ "url": "string", "surl": "string" }] }	Obtiene las urls filtradas
400	{ "status": "fail", "data": "string" }	Wrong value. El número de página o tamaño de página es erróneo.



401	{ "status": "fail", "data": "string" }	Expired JWT Token. La sesión ha expirado.
403	{ "status": "fail", "data": "string" }	Forbidden. El usuario logeado no pertenece al grupo introducido.

5.4.3 POST - /api/v1/url

Petición - Estructura de campos

HEADER

Parámetro	Tipo	Obligatorio	Descripción
Authorization	string	NO	Es necesario indicar el Token de sesión tipo Bearer en la cabecera Authorization. IMPORTANTE: El token debe ir precedido de la palabra Bearer y un espacio.

MESSAGE BODY

Formato	Parámetro	Tipo	Obligatorio	Descripción
Application/json	-	array	SI	Colección de objetos que contienen una url a acortar y opcionalmente la cadena de texto a usar para la url acortada.

*payload – Estructura de campo.

Parámetro	Tipo	Obligatorio	Descripción
url	string	SI	Recoge las urls (ruta larga) que luego serán devueltas de manera acortada. Debe contener una lista json con las urls a acortar.
surl	string	NO	Recoge una lista con las propuestas de la urls acortadas (propuesta opcional).



Ejemplo de Petición:

Método	POST
header	Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzUxMiJ9...
body	<pre>[{ "url": "https://www.google.es", "surl": "goo" }, { "url": "https://www.ure.es" }]</pre>

Respuesta – Estructura de campos

Códigos HTTP de respuesta	Responses model	Descripción
201	<pre>{ "status": "string", "data": { "shortenedUrls": "array", "nonShortenedExistentUrls": [{ "url": "string", "error": "string" }], "nonShortenedUrl": [{ "url": "string", "error": "string" }] } }</pre>	Operación correcta y url dada de alta. Se generan las urls acortadas suministradas, o se devuelve respuesta de tipo 2XX indicando que la url ya se generó previamente.
401	<pre>{ "code": "401", "message": "Expired JWT Token" }</pre>	JWR Token ha expirado. La sesión ha expirado.
402	<pre>{ "status": "fail", "data": "string" }</pre>	Información corrupta. Los datos de entrada no son correctos o presentan errores.
500	<pre>{ "status": "error", "data": "string" }</pre>	Ocurrió un error durante el proceso.



5.4.4 DELETE - /api/v1/url/delete

Permite eliminar una o varias URLs.

Petición - Estructura de campos

REQUEST HEADERS

Nombre	Tipo	Obligatorio	Descripción
Authorization	string	SI	Token de sesión de tipo JWT (https://jwt.io/). Debe ir precedido de "Bearer ". Ej: Bearer eyJhbGciOiJ[...]

MESSAGE BODY

Formato	Parámetro	Tipo	Obligatorio	Descripción
Application/json	sUrl	string	SI	Identificador del grupo a eliminar.

Ejemplo de Petición:

Método	DELETE
header	Authorization: "Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJz..."
Path	api/v1/url/delete

Respuesta – Estructura de campos

Códigos HTTP de respuesta	Responses model	Descripción
200	{ "status": "success", "data": { "sUrls": "array" } }	<i>URLs deleted</i> Eliminación de URL mediante Short URL.
401	{ "code": "401", "message": "Expired JWT Token" }	<i>Expired JWT Token.</i> La sesión ha expirado.
404	{ "status": "fail", "data": "SURL no existe" }	<i>Bad Request.</i> <i>Short URL No existe.</i>
500	{ "status": "error", "data": "string" }	<i>Returned on any other error.</i> Ocurrió un error durante el proceso.



5.5 User

Este servicio contiene el *endpoint* responsable de mostrar los datos del usuario.

5.5.1 GET - /api/v1/user/details

Petición - Estructura de campos

HEADER

Parámetro	Tipo	Obligatorio	Descripción
Authorization	string	NO	Es necesario indicar el Token de sesión tipo Bearer en la cabecera Authorization. IMPORTANTE: El token debe ir precedido de la palabra Bearer y un espacio.

Ejemplo de Petición:

Método	GET
header	Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzUxMiJ9...

Respuesta – Estructura de campos

Códigos HTTP de respuesta	Responses model	Descripción
200	{ "status": "success", "data": { "userName": "RICARDO", "app-name": "RICARDO", "roles": "ROLE_USER", "Auth-method": "password" } }	Operación correcta. Obtiene los datos del usuario.
500	{ "status": "error", "data": "string" }	Ocurrió un error durante el proceso.

5.6 GroupByUser

Este servicio contiene el *endpoint* responsable de mostrar el identificador de los grupos a los que pertenece el usuario.



5.6.1 GET - /api/v1/user/groups

Petición - Estructura de campos

HEADER

Parámetro	Tipo	Obligatorio	Descripción
Authorization	string	NO	Es necesario indicar el Token de sesión tipo Bearer en la cabecera Authorization. IMPORTANTE: El token debe ir precedido de la palabra Bearer y un espacio.

Ejemplo de Petición:

Método	GET
header	Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzUxMiJ9...

Respuesta – Estructura de campos

Códigos HTTP de respuesta	Responses model	Descripción
201	{ "status": "string", "data": ["string"] }	Operación correcta. Se genera el listado de grupos de un usuario de manera correcta.
500	{ "status": "error", "data": "string" }	Ocurrió un error durante el proceso.

5.7 Documentación (default)

5.7.1 GET - /api/doc.json

Petición - Estructura de campos

IMPORTANTE: No requiere ningún parámetro adicional.

Ejemplo de Petición:

Método	GET
header	-



path	/api/doc.json
------	---------------

5.7.2 POST - /api/v1/events/dispatch

Petición - Estructura de campos

IMPORTANTE: No requiere ningún parámetro adicional.

Ejemplo de Petición:

Método	POST
header	-
path	/api/v1/events/dispatch