

Manual de API de Organismos

FACe, Punto General de
Entrada de Facturas
Electrónicas de la
Administración General
del Estado



TLP: CLEAR

AUTOR	SGAD
ÁREA	Catálogos
PROYECTO	FACe, Punto General de Entrada de Facturas de la Administración General del Estado
NOMBRE DEL DOCUMENTO	Manual de API de Organismos

Control de Versiones del Documento

VERSIÓN	AUTOR	FECHA	DESCRIPCIÓN
2.0.0	Equipo FACe	19/11/2025	<ul style="list-style-type: none"> Cambio de plantilla Revisión y corrección de errores
1.0.0	Equipo FACe	03/1/2024	Versión inicial del documento

Índice

1	INTRODUCCIÓN	4
1.1	Objetivos	4
2	CONSIDERACIONES PREVIAS	5
2.1	Alta como integrador	5
2.2	Catálogo de APIs	6
2.3	Conceptos de autenticación	6
3	DESCRIPCIÓN DEL API	9
3.1	Grupo Centros Administrativos	9
3.1.1	Órganos Gestores (/administrative-centres/ogs).....	9
3.1.2	Oficinas Contables (/administrative-centres/ocs).....	13
3.1.3	Unidades Tramitadoras (/administrative-centres/uts).....	19
3.1.4	Relaciones	23
3.2	Grupo Facturas	31
3.2.1	Gestión de facturas (/invoices)	31
3.2.2	Gestión de solicitudes de anulación (/invoices)	51
3.2.3	Gestión de descargas (/files).....	57
3.3	Grupo Notificaciones	65
3.3.1	Gestión de notificaciones y noticias.....	65
3.4	Grupo Auditorías	71
3.4.1	Auditorías	71
3.5	Grupo Usuarios	77
3.5.1	Gestión de usuarios	77
4	ANEXO I	84

1 INTRODUCCIÓN

1.1 Objetivos

El presente documento tiene como objetivo servir de manual de uso de las APIs expuestas por el Punto General de Entrada de Facturas Electrónicas de la AGE, conocido como FACe para los Organismos.

En el apartado 2 Consideraciones previas se tratarán aspectos relacionados con el alta como integrador dentro del sistema FACe y de seguridad como es el modelo de autenticación que expone el API.

2 CONSIDERACIONES PREVIAS

2.1 Alta como integrador

Para poder utilizar los servicios web REST de FACe el usuario se tiene que dar de alta previamente en el **Portal de Integradores** cuyas URL son:

Servicios Estables	Producción
https://se-organismos-face.redsara.es/inicio	https://organismos.face.gob.es/inicio

En dicho portal podrá:

- Darse de alta como integrador
- Generar plataformas, tantas como se necesiten, y asociarles certificados (PEM)
- Gestionar autorizados.

Los certificados tendrán que ser los que se utilicen para firmar las peticiones SOAP para que FACe pueda verificar quién está llamando a los servicios web y si tiene permisos para ello.

2.2 Catálogo de APIs

Puede encontrar las APIs en las siguientes rutas:

- Servicios estables: <https://se-api-face.redsara.es/organisms/doc>
- Producción: <https://api.face.gob.es/organisms/doc>

En su versión V1 FACe expone la siguiente API:

- /v1/organisms/

2.3 Conceptos de autenticación

Las invocaciones a la API están securizadas mediante tokens a través de un JWT firmado con JWS. El integrador deberá generar un token que tendrá que incluirse en la cabecera de las peticiones. El token no es más que la firma cifrada que permita al API identificar al usuario.

- **JWT (JSON Web Token):** El token en sí. Un JWT es un estándar abierto (RFC 7519, <https://datatracker.ietf.org/doc/html/rfc7519>) que define una forma compacta y segura de representar afirmaciones entre dos partes como un objeto JSON.
- **JWS (JSON Web Signature):** El mecanismo de firma. Es un estándar relacionado con JWT que define una forma de firmar digitalmente un mensaje JSON. La firma digital garantiza la integridad y autenticidad del mensaje.

El tiempo de validez del token será de 5 minutos. Una vez pasado este tiempo de validez, el servidor no permitirá más el acceso a recursos con dicho token y deberá generarse un nuevo token.

Un JWT firmado con JWS consta de tres partes codificadas en base 64 y separadas por puntos:

- **Header:** Un JSON que contiene información sobre el tipo de token (JWT), el algoritmo de firma utilizado y otros metadatos. En nuestro caso es un JSON con el tipo de token, su codificación (RSA Signature with SHA-256 (RS256) o HMAC with SHA-256 (HS256)) y una clave xc5 cuyo valor es un array con el que llamaremos PEM limpio, y que es el PEM, o parte publica, del certificado con el que se va a firmar en base 64, en una línea, sin espacios ni saltos de línea, ni los tags de -----BEGIN/END CERTIFICATE-----. Todo ello, codificado a base 64:

```
{  
  "typ": "JWT",  
  "alg": "RS256",  
  "x5c": [ "(PEM limpio )" ]  
}
```

- Payload: Un JSON con los claims o atributos, que son pares clave-valor que definen al token y contienen información sobre la entidad a la que pertenece el token. Todo ello, codificado a base 64:

```
{  
  "username": "(Hash o huella digital SHA1 del PEM limpio)"  
  "iat": "(fecha de creación del token en formato de tiempo  
UNIX)",  
  "exp": "(fecha de expiración del token en formato de  
tiempo UNIX.)"  
}
```

- Signature: Un valor calculado sobre el header y el payload. Se concatenan las representaciones base64 del header y payload, separadas por un punto. Se calcula el hash de la cadena concatenada utilizando el algoritmo SHA-256. Se utiliza la clave privada RSA para firmar el hash. Se codifica en base64 la firma resultante y el resultado es la signature.

Tiene más información, una herramienta de creación de token JWT, decodificadores y generadores de JWT, conversor de tiempo unix y de hash en las páginas:

- <https://jwt.io/>
- <https://jwt.one/https://tribestream.io/tools/jwt/>
- <https://www.jstoolset.com/jwt>
- https://time.is/es/Unix_time_converter
- <https://www.convertstring.com/es/Hash/SHA1>

Para mayor privacidad en la generación del JWT tienen bibliotecas de lenguajes de programación como:

- Node.js: jsonwebtoken o Passport.js
- PHP: Firebase PHP Admin SDK, Lcobucci JWT, Sodium PHP, RobRichards/jwt
- Pythoz: PyJWT

- Java: Nimbus JOSE+JWT jjwt
- .NET: System.IdentityModel.Tokens.Jwt

En el [Anexo I](#) se muestra un ejemplo del código en java para generar el token.

3 DESCRIPCIÓN DEL API

3.1 Grupo Centros Administrativos

Los recursos ubicados bajo la ruta /administrative-centres están relacionados con la gestión de unidades DIR que participan en FAcE como receptoras de facturas electrónicas.

3.1.1 Órganos Gestores (/administrative-centres/ogs)

3.1.1.1 Alta de Órgano Gestor [POST]

Creación de un centro administrativo como un Órgano Gestor.

- API Administrative-centres/ogs

```
/v1/organisms/v1/administrative-centres/ogs
```

- Parámetros de body
 - dirCode (string, obligatorio): código de la unidad DIR
 - alias (string, obligatorio): nombre que tendrá el centro administrativo como OG.
 - description (string, opcional)
- Petición (text/plain)

```
{  
  "dirCode": "A08051032",  
  "alias": "Pruebas_1",  
  "description": "",  
}
```

- Response 201 (application/json)

```
{
  "identifier": "Q4500550A",
  "code": "A08051032",
  "name": "Agencia de Transformaci\u00f3n Digital
de Castilla-La Mancha",
  "alias": "Pruebas_1",
  "description": "",
  "type": "\organo Gestor",
}
```

- Response 400 (application/json)

```
{
  "errors": [
    "string"
  ],
  "code": "400",
  "message": "Bad Request"
}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.1.1.2 Editar Órgano Gestor [PUT]

Permite editar ciertos datos del Órgano Gestor como son el nombre y la descripción

- API Administrative-centres/ogs

```
/v1/administrative-centres/ogs/{dirCode}
```

- Parámetros de path
 - dirCode (string, obligatorio): código de unidad DIR
- Parámetros de body
 - alias (string, obligatorio)
 - description (string, opcional)

- Petición (text/plain)

```
{
  "alias": "Pruebas_234",
  "description": "",
}
```

- Response 200 (application/json)

- Body

```
{
  "code": "A08051032",
  "name": "Agencia de Transformaci\u00f3n Digital  
de Castilla-La Mancha",
  "alias": "Pruebas_234",
  "description": "",
  "type": "\organo Gestor",
}
```

- Response 400 (application/json)

```
{
  "errors": [
    "string"
  ],
  "code": "400",
  "message": "Bad Request"
}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.1.1.3 Borrado de Órgano Gestor [DELETE]

Eliminación de un centro administrativo como Órgano Gestor. Para poder eliminar un órgano gestor, éste no puede formar parte de una relación activa, y dicha relación no puede haber recibido facturas.

- API Administrative-centres/ogs

```
/organisms/v1/administrative-centres/ogs/{dirCode}
```

- Parámetros de path
 - dirCode (string, obligatorio).

- Petición (text/plain)

```
{}
```

- Response 204 (application/json)

- Body

```
{Og deleted}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.1.2 Oficinas Contables (/administrative-centres/ocs)

3.1.2.1 Alta de Oficina Contable [POST]

Creación de un centro administrativo como una Oficina Contable.

- API Administrative-centres/ocs

```
/organisms/v1/administrative-centres/ocs
```

- Parámetros de body
 - dirCode (string, obligatorio)
 - alias (string, obligatorio): nombre que tendrá el centro administrativo como OC.
 - description (string, opcional)
 - email (string, obligatorio)
- Petición (text/plain)

```
{
  "dirCode": " P00000XXX",
  "alias": "OC de pruebas",
  "description": " OC de pruebas ",
  "email": "a@correo.com"
}
```

- Response 201 (application/json)

```
{
  "email": "pruebas@correo.es",
  "rcf": {
    "system": {
      "uuid": "face-system-uuid",
      "name": "FACE"
    }
  },
  "code": "A08051032",
  "name": "Agencia de Transformaci\u00f3n Digital  
de Castilla-La Mancha",
  "alias": "Pruebas_234",
  "description": "",
  "type": "Oficina Contable",
}
```

- Response 400 (application/json)

```
{
  "errors": [
    "string"
  ],
  "code": "400",
  "message": "Bad Request"
}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.1.2.2 Editar Oficina Contable [PUT]

Permite editar ciertos datos del Oficina Contable como son el alias, la descripción o el correo asociado.

- API Administrative-centres/ocs

```
/organisms/v1/administrative-centres/ocs/{dirCode}
```

- Parámetros de path:
 - dirCode (string, obligatorio)
- Parámetros de body
 - alias (string, obligatorio)
 - description (string, opcional)
 - email (string, obligatorio)

- Petición (text/plain)

```
{
  "alias": "Pruebas_23456",
  "description": "",
  "email": "pruebas@correo.es"
}
```

- Response 200 (application/json)

- Body

```
{
  "email": " pruebas@correo.es ",
  "rcf": {
    "system": {
      "uuid": "face-system-uuid",
      "name": "FACE"
    }
  },
  "code": "A08051032",
  "name": "Agencia de Transformaci\u00f3n Digital  
de Castilla-La Mancha",
  "alias": "Pruebas_23456",
  "description": "",
  "type": "Oficina Contable",
}
```

- Response 400 (application/json)

```
{
  "errors": [
    "string"
  ],
  "code": "400",
  "message": "Bad Request"
}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 404 (application/json)

```
{
  "code": "404",
  "message": "Not Found"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```


3.1.2.3 Borrado de Oficina Contable [DELETE]

Eliminación de un centro administrativo como Oficina Contable. Para poder eliminar una oficina contable, éste no puede formar parte de una relación activa, y dicha relación no puede haber recibido facturas.

- API Administrative-centres/ocs

```
/v1/administrative-centres/ocs/{dirCode}
```

- Parámetros de path
 - dirCode (string, obligatorio)

- Petición (text/plain)

```
{}
```

- Response 204 (application/json)

```
{Ocs deleted}
```

- Response 401 (application/json)

```
{  
  "code": "401",  
  "message": "Unauthorized"  
}
```

- Response 403 (application/json)

```
{  
  "code": "403",  
  "message": "Forbidden"  
}
```

- Response 500 (application/json)

```
{  
  "code": "500",  
  "message": "Internal Server Error - 1134356675"  
}
```

3.1.2.4 Modificar los datos de una Oficina Contable [PATCH]

- API Administrative-centres/ocs

```
/v1/administrative-centres/ocs/{dirCode}
```

- Parámetros de path
 - dirCode (string, obligatorio)
- Petición (text/plain)

```
{  
  "base64SignedXml": "string"  
}
```

- Response 200 (application/json)

```
{  
  "email": "string",  
  "rcf": {  
    "system": {  
      "uuid": "string",  
      "name": "string"  
    }  
  },  
  "code": "string",  
  "name": "string",  
  "alias": "string",  
  "description": "string",  
  "type": "string"  
}
```

- Response 400 (application/json)

```
{  
  "errors": [  
    "string"  
  ],  
  "code": "400",  
  "message": "Bad Request"  
}
```

- Response 401 (application/json)

```
{  
  "code": "401",  
  "message": "Unauthorized"  
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.1.3 Unidades Tramitadoras (/administrative-centres/uts)

3.1.3.1 Alta de Unidad Tramitadora [POST]

Creación de un centro administrativo como una Unidad Tramitadora.

- API Administrative-centres/uts

```
/v1/administrative-centres/uts
```

- Parámetros de body
 - dirCode (string, obligatorio).
 - alias (string, obligatorio): nombre que tendrá el centro administrativo como UT.
 - description (string, opcional)
 - email (string, obligatorio)
- Petición (text/plain)

```
{
  "dirCode": " P00000XXX",
  "alias": "UT de pruebas",
  "description": "UT de pruebas",
  "email": "prueba@correo.com"
}
```

- Response 201 (application/json)

```
{
  "email": "string",
  "code": "string",
  "name": "string",
  "alias": "string",
  "description": "string",
  "type": "string"
}
```

- Response 400 (application/json)

```
{
  "errors": [
    "string"
  ],
  "code": "400",
  "message": "Bad Request"
}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.1.3.2 Editar Unidad Tramitadora [PUT]

Permite editar ciertos datos de la Unidad Tramitadora como son el nombre, la descripción y si está activo o no.

- API Administrative-centres/uts

```
/v1/administrative-centres/uts/{dirCode}
```

- Parámetros de path
 - dirCode (string, obligatorio).
- Parámetros de body
 - alias (string, obligatorio)
 - description (string, opcional)
 - email (string, obligatorio)

- Petición (text/plain)

```
{  
  "alias": "UT de pruebas editado",  
  "description": "UT de pruebas",  
  "email": "prueba@correo.com"  
}
```

- Response 200 (application/json)

- Body

```
{  
  "email": "string",  
  "code": "string",  
  "name": "string",  
  "alias": "string",  
  "description": "string",  
  "type": "string"  
}
```

- Response 400 (application/json)

```
{
  "errors": [
    "string"
  ],
  "code": "400",
  "message": "Bad Request"
}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.1.3.3 Borrado de Unidad Tramitadora [DELETE]

Eliminación de un centro administrativo como Unidad Tramitadora. Para poder eliminar una unidad tramitadora, éste no puede formar parte de una relación activa, y dicha relación no puede haber recibido facturas.

- API Administrative-centres/uts

```
/v1/administrative-centres/uts/{dirCode}
```

- Parámetros de query
 - dirCode (string, obligatorio).
- Petición (text/plain)

- Response 204 (application/json)

- Body

```
{Ut deleted}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.1.4 Relaciones

Los recursos ubicados bajo la ruta /relations están relacionados con la gestión de las relaciones entre los centros administrativos (OG, OC y UT).

3.1.4.1 Listado de Relaciones [GET]

Devuelve la información de las relaciones entre centros administrativos.

- API relations

```
/v1/relations
```

- Parámetros de query

- page (string, obligatorio). Número de páginas del resultado. Por defecto está puesto 1.
- limit (string, obligatorio). Número de resultados por página. Por defecto está puesto 1000

- fulltext (string, opcional)
- oc (string, opcional).
- og (string, opcional)
- ut (string, opcional)
- active (string, opcional): los valores admitidos son **n** (inactiva) o **y** (activa)
- Petición (text/plain)

```
{  
  "page": "1",  
  "limit": "1000"  
}
```

- Response 200 (application/json)
 - Body

```
{  
  "items": [  
    {  
      "hash": "string",  
      "oc": {  
        "code": "string",  
        "name": "string",  
        "alias": "string",  
        "rcf": {  
          "system": {  
            "uuid": "string",  
            "name": "string"  
          }  
        }  
      }  
    },  
    {  
      "og": {  
        "code": "string",  
        "name": "string",  
        "alias": "string",  
        "rcf": {  
          "system": {  
            "uuid": "string",  
            "name": "string"  
          }  
        }  
      }  
    },  
    {  
      "ut": {  
        "code": "string",  
        "name": "string",  
        "alias": "string",  
        "rcf": {  
          "system": {  
            "uuid": "string",  
            "name": "string"  
          }  
        }  
      }  
    }  
  ]  
}
```



```
{
  "rcf": {
    "system": {
      "uuid": "string",
      "name": "string"
    }
  },
  "active": true
},
"count": 0,
"total": 0,
"limit": 0,
"page": 0
}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.1.4.2 Crear Relaciones [POST]

Crea la relación (activa o no) entre centros administrativos existentes.

- API relations

```
/v1/relations
```

- Parámetros de body
 - ocCode (string, obligatorio)
 - ogCode (string, obligatorio)

- utCode (string, obligatorio)
- active (opcional, true/false)

- Petición (text/plain)

```
{
  "ocCode": "string",
  "ogCode": "string",
  "utCode": "string",
  "active": true
}
```

- Response 201 (application/json)

```
{
  "hash": "string",
  "oc": {
    "code": "string",
    "name": "string",
    "alias": "string"
  },
  "og": {
    "code": "string",
    "name": "string",
    "alias": "string"
  },
  "ut": {
    "code": "string",
    "name": "string",
    "alias": "string"
  },
  "active": true
}
```

- Response 400 (application/json)

```
{
  "errors": [
    "string"
  ],
  "code": "400",
  "message": "Bad Request"
}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.1.4.3 Actualizar Relaciones [PATCH]

Permite actualizar en la relación existente únicamente si se activa o no, lo que permitirá que pueda recibir facturas o no.

- API relations

```
/v1/relations/{hash}
```

- Parámetros del path
 - hash (string, obligatorio): Hash de la relación para poder identificarla.
- Parámetros de body
 - active (opcional, true/false).
- Petición (text/plain)

```
{
  "hash": 0b61ba393c70eb947303
}
```

- Response 200 (application/json)

- Body

```
{
  "hash": "string",
  "oc": {
    "code": "string",
    "name": "string",
    "alias": "string"
  },
  "og": {
    "code": "string",
    "name": "string",
    "alias": "string"
  },
  "ut": {
    "code": "string",
    "name": "string",
    "alias": "string"
  },
  "active": true
}
```

- Response 400 (application/json)

```
{
  "errors": [
    "string"
  ],
  "code": "400",
  "message": "Bad Request"
}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.1.4.4 Eliminar Relaciones [DELETE]

Permite eliminar una relación, siempre y cuando cumpla con las restricciones definidas, es decir, que no está activa y que no haya recibido facturas.

- API relations

```
/v1/relations/{hash}
```

- Parámetros del path

- hash (string, obligatorio): Hash de la relación para poder identificarla.

- Petición (text/plain)

```
{
  "hash": 0b61ba393c70eb947303
}
```

- Response 200 (application/json)

- Body

```
{
  "items": [
    {
      "hash": "string",
      "oc": {
        "code": "string",
        "name": "string",
        "alias": "string"
      },
      "og": {
        "code": "string",
        "name": "string",
        "alias": "string"
      },
      "ut": {
        "code": "string",
        "name": "string",
        "alias": "string"
      },
      "active": true
    }
  ],
  "count": 0,
  "total": 0,
  "limit": 0,
  "page": 0
}
```

- Response 401 (application/json)

```
{  
  "code": "401",  
  "message": "Unauthorized"  
}
```

- Response 403 (application/json)

```
{  
  "code": "403",  
  "message": "Forbidden"  
}
```

- Response 409 (application/json)

```
{  
  "code": "409",  
  "message": "Conflict"  
}
```

- Response 500 (application/json)

```
{  
  "code": "500",  
  "message": "Internal Server Error - 1134356675"  
}
```

3.2 Grupo Facturas

Los recursos ubicados bajo la ruta /invoices están relacionados con la evolución de las facturas de los proveedores de las administraciones públicas.

3.2.1 Gestión de facturas (/invoices)

3.2.1.1 Listado de facturas [GET]

Permite obtener el listado de todas las facturas de las unidades en las que el usuario logado tiene permisos.

- API Invoices

/v1/invoices

- Parámetros de query string
 - page (string, obligatorio). Por defecto será 1.
 - limit (string, obligatorio). Por defecto será 1000
 - fulltext (string, opcional). Filtrar por número de registro o número de serie de la factura.
 - oc (string, opcional)
 - og (string, opcional)
 - ut (string, opcional)
 - registryDateFrom (string, opcional). Intervalo de fechas de registro, fecha desde. en formato dd/mm/yyyy
 - registryDateTo (string, opcional). Intervalo de fechas de registro, fecha hasta. en formato dd/mm/yyyy
 - factoringAssignment (string, opcional). Estado de la cesión de crédito.
 - hasFactoringAssignment (string, opcional). Filtro por si tiene cesión de crédito (y) o no (n).
 - status (string, opcional). Estado de la factura.
 - cancellationRequestStatus(string, opcional). Estado de solicitud de anulación.
 - format (string, opcional). Tipo de archivo (facturae, UBL, CII, etc)

- seller (string, opcional). Identificador del emisor de la factura
- factoringAssignment (string, opcional). Identificador del cesionario de la factura
- orderBy (string, obligatorio). Criterio de ordenación de los resultados.
Por defecto está el número de registro
- order (string, obligatorio). Definir si el orden será ascendente o descendente.
- Petición (text/plain)

```
{
  "page": "1",
  "limit": "1000",
  .."orderBy": "registry_code",
  .."order": "DESC"
}
```

- Response 200 (application/json)
 - Body

```
{
  "items": [
    {
      "serial": "string",
      "number": "string",
      "serialNumber": "string",
      "expeditionDate": "2025-03-20T15:48:06.241Z",
      "seller": {
        "identifier": "string",
        "fullName": "string"
      },
      "buyer": {
        "identifier": "string",
        "fullName": "string"
      },
      "type": "string",
      "hasRcfFace": true,
      "oc": {
        "code": "string",
        "name": "string",
        "alias": "string"
      },
      "invoicingPointName": "string",
      "rcfName": "string",
      "og": {
        "code": "string",
        "name": "string",
        "alias": "string"
      }
    }
  ]
}
```



```

    },
    "ut": {
      "code": "string",
      "name": "string",
      "alias": "string"
    },
    "cancellationRequestStatus": "string",
    "status": "string",
    "statusCode": "string",
    "cancellationRequestHistory": [
      {
        "requestComment": "string",
        "uuid": "string",
        "requestResponse": "string",
        "requestedAt": "2025-03-
20T15:48:06.241Z",
        "processedAt": "2025-03-
20T15:48:06.241Z",
        "processed": true,
        "accepted": true
      }
    ],
    "registryCode": "string",
    "registryDate": "2025-03-20T15:48:06.241Z",
    "amount": 0,
    "currency": "string",
    "rcfRegistryCode": "string",
    "third": {
      "identifier": "string",
      "fullName": "string"
    },
    "factoringAssignmentDocuments": [
      {
        "isDeleted": true,
        "uuid": "string",
        "filename": "string",
        "mime": "string",
        "hash": "string"
      }
    ],
    "europeanInvoice": {
      "uuid": "string",
      "filename": "string",
      "mime": "string",
      "hash": "string",
      "isDeleted": true
    },
    "facturae": {
      "uuid": "string",
      "filename": "string",
      "mime": "string",
      "hash": "string",

```

```
    "isDeleted": true
  },
  "attachments": [
    {
      "uuid": "string",
      "filename": "string",
      "mime": "string",
      "hash": "string",
      "isDeleted": true
    }
  ],
  "statusHistory": [
    {
      "code": "string",
      "name": "string",
      "comment": "string",
      "createdAt": "2025-03-20T15:48:06.241Z"
    }
  ],
  "factoringAssignmentStatus": "string",
  "factoringAssignmentComment": "string",
  "hasFactoringAssignment": true,
  "factoringAssignment": {
    "identifier": "string",
    "fullName": "string"
  },
  "signer": {
    "identifier": "string",
    "fullName": "string"
  }
},
"count": 0,
"total": 0,
"limit": 0,
"page": 0
}
```

- Response 400 (application/json)

```
{
  "errors": [
    "string"
  ],
  "code": "400",
  "message": "Bad Request"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.2.1.2 Listado de histórico de estado de facturas [GET]

Permite obtener el listado de todos los estados por los que ha pasado una factura.

- API Invoices

```
/v1/invoices/{registryCode}/status-history
```

- Parámetros de query
 - creationDateFrom (string, opcional). El formato fecha será dd/mm/yyyy
 - creationDateTo (string, opcional). El formato fecha será dd/mm/yyyy
 - orderBy (string, obligatorio). Criterio de ordenación de los resultados. Por defecto será "creationDate"
 - Order (string, obligatorio). Definir si el orden será ascendente o descendente.
- Parámetros de path
 - registryCode (string, obligatorio): código único de registro de factura de REGAGE.
- Petición (text/plain)

```
{
  "page": "1",
  "limit": "1000",
  ".."orderBy": "creationDate",
  ".."order": "DESC",
  ".."registry_Code": "TEST102"
}
```

- Response 200 (application/json)

- Body

```
{
  "items": [
    {
      "code": "1200",
      "name": "Registrada",
      "comment": "Molestiae unde natus atque sed laudantium.",
      "createdAt": "2023-11-09T12:59:27+01:00"
    }
  ]
}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.2.1.3 Listado de estados de facturas [GET]

Permite obtener el listado de todos los estados por los que puede transitar una factura, así como la información de los estados de la solicitud de anulación y de la cesión de crédito.

- API Invoices

```
/v1/invoices/invoice-status-list
```

- No hay parámetros de entrada
- Response 200 (application/json)
 - Body

```
{
  "eInvoiceStatus": [
    {
      "code": "string",
      "name": "string",
      "description": "string"
    }
  ],
  "cancellationRequestStatus": [
    {
      "code": "string",
      "name": "string",
      "description": "string"
    }
  ],
  "factoringAssignmentStatus": [
    {
      "code": "string",
      "name": "string",
      "description": "string"
    }
  ]
}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 404 (application/json)

```
{
  "code": "404",
  "message": "Not Found"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.2.1.4 Actualizar datos de una factura registrada en RCF [PATCH]

Permite actualizar datos de una factura registrada en RCF.

- API Invoices

```
/v1/invoices/{registryCode}
```

- Parámetros de body
 - Transition (string, opcional)
 - comment (string, opcional)
 - rcfRegistryCode (string, opcional)
- Parámetros de path
 - registryCode (string, obligatorio): número de registro de la factura en REGAGE que se quiere evolucionar.

- Petición (text/plain)

```
{
  "transition": "string",
  "comment": "string",
  "rcfRegistryCode": "string"
}
```

- Response 200 (application/json)

- Body

```
{
  "status": "string",
  "statusCode": "string",
  "statusHistory": [
    {
      "code": "string",
      "name": "string",
      "comment": "string",
      "createdAt": "2025-03-21T07:53:22.700Z"
    }
  ],
  "registryCode": "string",
  "rcfRegistryCode": "string"
}
```

- Response 400 (application/json)

```
{
  "errors": [
    "string"
  ],
  "code": "400",
  "message": "Bad Request"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.2.1.5 Marcar factura como Registrada en RCF [PATCH]

Permite pasar una factura del estado Registrada a Registrada en RCF.

- API Invoices

```
/v1/invoices/{registryCode}/confirm-download
```

- Parámetros de body
 - comment (string, opcional).
- Parámetros de path
 - registryCode (string, obligatorio): número de registro de la factura que se quiere evolucionar.
- Petición (text/plain)

```
{
  "comment": "factura correcta",
}
```

- Response 200 (application/json)

- Body

```
{
  "status": [
    {
      "code": "1300",
      "createdAt": "2021-01-29T10:00:25+01:00",
      "name": "registered_in_rcf",
      "comment": " factura correcta"
    },
    {
      "code": "1200",
      "createdAt": "2021-01-15T04:47:13+01:00",
      "name": "registered"
    },
  ],
  "statusCode": "1300",
  "registryCode": " REGAGE21e00000010569"
}
```

- Response 400 (application/json)

```
{
  "errors": [
    "string"
  ],
  "code": "400",
  "message": "Bad Request"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```


3.2.1.6 Actualizar el estado de una factura [PATCH]

Permite actualizar el estado de una factura registrada en RCF.

- API Invoices

```
/v1/invoices/{registryCode}/back-to-register-in-rcf
```

- Parámetros de body
 - comment (string, opcional).
- Parámetros de path
 - registryCode (string, obligatorio): número de registro de la factura que se quiere evolucionar.
- Petición (text/plain)

```
{
  "comment": "factura correcta",
}
```

- Response 200 (application/json)
 - Body

```
{
  "registryCode": "string",
  "status": "string",
  "statusCode": "string",
  "statusHistory": [
    {
      "code": "string",
      "name": "string",
      "comment": "string",
      "createdAt": "2025-03-21T08:06:22.226Z"
    }
  ],
  "rcfRegistryCode": "string"
}
```

- Response 400 (application/json)

```
{
  "errors": [
    "string"
  ],
  "code": "400",
  "message": "Bad Request"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.2.1.7 Marcar factura como Verificada en RCF [PATCH]

Permite pasar una factura del estado Registrada en RCF a Verificada en RCF.

- API Invoices

```
/v1/invoices/{registryCode}/verify-in-rcf
```

- Parámetros de body
 - comment (string, opcional).
- Parámetros de path
 - registryCode (string, obligatorio): número de registro de la factura que se quiere evolucionar.
- Petición (text/plain)

```
{
  "comment": "factura correcta",
}
```

- Response 200 (application/json)

- Body

```
{
  "registryCode": "string",
  "status": "string",
  "statusCode": "string",
  "statusHistory": [
    {
      "code": "string",
      "name": "string",
      "comment": "string",
      "createdAt": "2025-03-21T08:16:58.438Z"
    }
  ],
  "rcfRegistryCode": "string"
}
```

- Response 400 (application/json)

```
{
  "errors": [
    "string"
  ],
  "code": "400",
  "message": "Bad Request"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.2.1.8 Marcar factura como Recibida en destino [PATCH]

Permite pasar una factura del estado verificada en RCF a recibida en destino.

- API Invoices

```
/v1/invoices/{registryCode}/receive-in-destination
```

- Parámetros de body
 - comment (string, opcional).
- Parámetros de path
 - registryCode (string, obligatorio): número de registro de la factura que se quiere evolucionar.
- Petición (text/plain)

```
{
  "comment": "factura pasa las validaciones",
}
```

- Response 200 (application/json)
 - Body

```
{
  "registryCode": "string",
  "status": "string",
  "statusCode": "string",
  "statusHistory": [
    {
      "code": "string",
      "name": "string",
      "comment": "string",
      "createdAt": "2025-03-21T08:21:33.908Z"
    }
  ],
  "rcfRegistryCode": "string"
}
```

- Response 400 (application/json)

```
{
  "errors": [
    "string"
  ],
  "code": "400",
  "message": "Bad Request"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.2.1.9 Marcar factura como Conformada [PATCH]

Permite pasar una factura de Recibida en destino a Conformada.

- API Invoices

```
/v1/invoices/{registryCode}/conform
```

- Parámetros de body
 - comment (string, opcional).
- Parámetros de path
 - registryCode (string, obligatorio): número de registro de la factura que se quiere evolucionar.
- Petición (text/plain)

```
{
  "comment": "ok",
}
```

- Response 200 (application/json)

- Body

```
{
  "registryCode": "string",
  "status": "string",
  "statusCode": "string",
  "statusHistory": [
    {
      "code": "string",
      "name": "string",
      "comment": "string",
      "createdAt": "2025-03-21T08:23:11.136Z"
    }
  ],
  "rcfRegistryCode": "string"
}
```

- Response 400 (application/json)

```
{
  "errors": [
    "string"
  ],
  "code": "400",
  "message": "Bad Request"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.2.1.10 Marcar factura como Contabilizada la obligación reconocida [PATCH]

Permite pasar una factura de Conformada a Contabilizada la obligación reconocida.

- API Invoices

```
/v1/invoices/{registryCode}/propose-payment
```

- Parámetros de body
 - comment (string, opcional).
- Parámetros de path
 - registryCode (string, obligatorio): número de registro de la factura que se quiere evolucionar.

- Petición (text/plain)

```
{  
  "comment": "pasa las validaciones",  
}
```

- Response 200 (application/json)

- Body

```
{  
  "registryCode": "string",  
  "status": "string",  
  "statusCode": "string",  
  "statusHistory": [  
    {  
      "code": "string",  
      "name": "string",  
      "comment": "string",  
      "createdAt": "2025-03-21T08:25:23.677Z"  
    }  
  ],  
  "rcfRegistryCode": "string"  
}
```

- Response 400 (application/json)

```
{
  "errors": [
    "string"
  ],
  "code": "400",
  "message": "Bad Request"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.2.1.11 Marcar factura como Pagada [PATCH]

Permite pasar una factura de los estados Conformada o Contabilizada la obligación reconocida a Pagada.

- API Invoices

```
/v1/invoices/{registryCode}/mark-as-paid
```

- Parámetros de body
 - comment (string, obligatorio): dependiendo del estado al que se quiera evolucionar el comentario puede ser opcional.
- Parámetros de path
 - registryCode (string, obligatorio): número de registro de la factura que se quiere evolucionar.
- Petición (text/plain)

```
{
  "comment": "se marca como pagada",
}
```


- Response 200 (application/json)

- Body

```
{
  "registryCode": "string",
  "status": "string",
  "statusCode": "string",
  "statusHistory": [
    {
      "code": "string",
      "name": "string",
      "comment": "string",
      "createdAt": "2025-03-21T08:27:33.244Z"
    }
  ],
  "rcfRegistryCode": "string"
}
```

- Response 400 (application/json)

```
{
  "errors": [
    "string"
  ],
  "code": "400",
  "message": "Bad Request"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.2.1.12 Marcar factura como Rechazada [PATCH]

Permite pasar una factura desde los estados: Registrada en RCF, Verificada en RCF, Recibida en destino y Conformada a Rechazada.

- API Invoices

```
/v1/invoices/{registryCode}/reject
```

- Parámetros de body
 - comment (string, obligatorio): se deberá justificar el motivo del rechazo de la factura.
- Parámetros de path
 - registryCode (string, obligatorio): número de registro de la factura que se quiere evolucionar.
- Petición (text/plain)

```
{  
  "comment": "la factura no pasa las validaciones  
contables",  
}
```

- Response 200 (application/json)
 - Body

```
{  
  "registryCode": "string",  
  "status": "string",  
  "statusCode": "string",  
  "statusHistory": [  
    {  
      "code": "string",  
      "name": "string",  
      "comment": "string",  
      "createdAt": "2025-03-21T08:29:45.114Z"  
    }  
  ],  
  "rcfRegistryCode": "string"  
}
```

- Response 400 (application/json)

```
{
  "errors": [
    "string"
  ],
  "code": "400",
  "message": "Bad Request"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.2.2 Gestión de solicitudes de anulación (/invoices)

3.2.2.1 Obtener solicitudes de anulación de facturas [PATCH]

Permite obtener la información de las solicitudes de anulación de una factura, ya estén aceptadas o rechazadas.

- API Invoices

```
/v1/invoices/{registryCode}/cancellation-requests/{cancellationRequest}
```

- Parámetros de body
 - comment (string, opcional).
 - transition (string, opcional).
- Parámetros de path
 - registryCode (string, obligatorio): número de registro de la factura que se quiere evolucionar.
 - cancellationRequest (string, obligatorio): identificación de la solicitud de anulación.
- Petición (text/plain)

```
{  
  "comment": "string",  
  "transition": "string"  
}
```

- Response 200 (application/json)
 - Body

```
{  
  "registryCode": "string",  
  "registryDate": "string",  
  "uuid": "string",  
  "requestComment": "string",  
  "requestResponse": "string",  
  "accepted": true,  
  "processed": true,  
  "requestedAt": "string",  
  "processedAt": "2025-03-21T08:33:38.731Z"  
}
```

- Response 400 (application/json)

```
{  
  "errors": [  
    "string"  
  ],  
  "code": "400",  
  "message": "Bad Request"  
}
```

- Response 403 (application/json)

```
{  
  "code": "403",  
  "message": "Forbidden"  
}
```

- Response 500 (application/json)

```
{  
  "code": "500",  
  "message": "Internal Server Error - 1134356675"  
}
```

3.2.2.2 Aceptar solicitud de anulación [PATCH]

Permite aceptar la solicitud de anulación de una factura. El estado de la factura quedará como “Anulada” y terminará su procesamiento.

- API Invoices

```
/v1/invoices/{registryCode}/cancellation-requests/{cancellationRequest}/accept
```

- Parámetros de body
 - comment (string, opcional).
- Parámetros de path
 - registryCode (string, obligatorio): número de registro de la factura que se quiere evolucionar.
 - cancellationRequest (string, obligatorio): identificación de la solicitud de anulación.

- Petición (text/plain)

```
{
  "comment": "aceptamos la información para
anular",
}
```

- Response 200 (application/json)

- Body

```
{
  "registryCode": "string",
  "registryDate": "string",
  "uuid": "string",
  "requestComment": "string",
  "requestResponse": "string",
  "accepted": true,
  "processed": true,
  "requestedAt": "string",
  "processedAt": "2025-03-21T08:38:11.637Z"
}
```

- Response 400 (application/json)

```
{
  "errors": [
    "string"
  ],
  "code": "400",
  "message": "Bad Request"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.2.2.3 Rechazar solicitud de anulación [PATCH]

Permite rechazar la solicitud de anulación de una factura. La factura podrá seguir su evolución.

- API Invoices

```
/v1/invoices/{registryCode}/cancellation-requests/{cancellationRequest}/reject
```

- Parámetros de body
 - comment (string, opcional).
- Parámetros de path
 - registryCode (string, obligatorio): número de registro de la factura que se quiere evolucionar.
 - cancellationRequest (string, obligatorio): identificación de la solicitud de anulación.

- Petición (text/plain)

```
{  
  .."comment": " Se rechaza la solicitud por falta de  
información"  
}
```

- Response 200 (application/json)

- Body

```
{  
  "registryCode": "string",  
  "registryDate": "string",  
  "uuid": "string",  
  "requestComment": "string",  
  "requestResponse": "string",  
  "accepted": true,  
  "processed": true,  
  "requestedAt": "string",  
  "processedAt": "2025-03-21T08:40:27.265Z"  
}
```

- Response 400 (application/json)

```
{  
  "errors": [  
    "string"  
  ],  
  "code": "400",  
  "message": "Bad Request"  
}
```

- Response 401 (application/json)

```
{  
  "code": "401",  
  "message": "Unauthorized"  
}
```

- Response 403 (application/json)

```
{  
  "code": "403",  
  "message": "Forbidden"  
}
```

- Response 500 (application/json)

```
{  
  "code": "500",  
  "message": "Internal Server Error - 1134356675"  
}
```

3.2.2.4 Listado histórico de solicitudes de anulación [GET]

Permite rechazar la solicitud de anulación de una factura. La factura podrá seguir su evolución.

- API Invoices

```
/v1/invoices/{registryCode}/cancellation-requests-history
```

- Parámetros de query
 - cancellationRequestAtFrom (string, opcional).
 - cancellationRequestAtTo (string, opcional).
 - cancellationProcessedAtFrom (string, opcional).
 - cancellationProcessedAtTo (string, opcional).
 - orderBy (string, obligatorio)
 - order (string, obligatorio).
- Parámetros de path
 - registryCode (string, obligatorio): número de registro de la factura.
- Petición (text/plain)

```
{
  .."comment": " Se rechaza la solicitud por falta de
información",
}
```

- Response 200 (application/json)

- Body

```
{
  "items": [
    {
      "registryCode": "string",
      "registryDate": "string",
      "uuid": "string",
      "requestComment": "string",
      "requestResponse": "string",
      "accepted": true,
      "processed": true,
      "requestedAt": "string",
      "processedAt": "2025-03-21T09:33:03.915Z"
    }
  ]
}
```


- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.2.3 Gestión de descargas (/files)

3.2.3.1 Obtener ficheros Facturae [GET]

Permite obtener el fichero XML firmado con el contenido de la factura.

- API Invoices

```
/v1/invoices/{registryCode}/facturae/{file}
```

- Parámetros de path

- registryCode (string, obligatorio): número de registro de la factura que se quiere descargar.
- file (string, obligatorio): identificación del fichero xml firmado.

- Petición (text/plain)

```
{}
```

- Response 200 (application/json)

- Body

```
{
  "uuid": "string",
  "filename": "string",
  "content": "string",
  "mime": "string",
  "hash": "string"
}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.2.3.2 Obtener ficheros Facturae en PDF [GET]

Permite obtener la factura en formato PDF.

- API Invoices

```
/v1/invoices/{registryCode}/facturae/{file}/pdf
```

- Parámetros de path

- registryCode (string, obligatorio): número de registro de la factura que se quiere evolucionar.
- file (string, obligatorio): identificación del fichero.

- Petición (text/plain)

```
{}
```

- Response 200 (application/json)

- Body

```
{
  "uuid": "string",
  "filename": "string",
  "content": "string",
  "mime": "string",
  "hash": "string"
}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.2.3.3 Obtener ficheros anejos a la factura [GET]

Permite obtener los ficheros anexos a la factura.

- API Invoices

```
/v1/invoice/{registryCode}/attachments/{file}
```

- Parámetros de path

- registryCode (string, obligatorio): número de registro de la factura que se quiere evolucionar.
- file (string, obligatorio): identificación del fichero.

- Petición (text/plain)

```
{}
```

- Response 200 (application/json)
 - Body

```
{
  "uuid": "string",
  "filename": "string",
  "content": "string",
  "mime": "string",
  "hash": "string"
}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.2.3.4 Obtener ficheros de cesión de crédito [GET]

Permite obtener ficheros que acreditan la cesión de crédito de una factura.

- API Invoices

```
/v1/invoice/{registryCode}/factoring-assignment-
documents/{file}
```

- Parámetros de path
 - registryCode (string, obligatorio): número de registro de la factura que se quiere evolucionar.
 - file (string, obligatorio): identificación del fichero que acredita la cesión de crédito.
- Petición (text/plain)

```
{}
```

- Response 200 (application/json)
 - Body

```
{
  "uuid": "string",
  "filename": "string",
  "content": "string",
  "mime": "string",
  "hash": "string"
}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.2.3.5 Obtener ficheros de facturas europeas [GET]

Permite obtener ficheros XML original de una factura europea (UBL CII).

- API Invoices

```
/v1/invoice/{registryCode}/european-invoice/{file}
```

- Parámetros de path
 - registryCode (string, obligatorio): número de registro de la factura que se quiere evolucionar.
 - file (string, obligatorio): identificación del fichero que acredita la cesión de crédito.

- Petición (text/plain)

```
{}
```

- Response 200 (application/json)

- Body

```
{  
  "uuid": "string",  
  "filename": "string",  
  "content": "string",  
  "mime": "string",  
  "hash": "string"  
}
```

- Response 401 (application/json)

```
{  
  "code": "401",  
  "message": "Unauthorized"  
}
```

- Response 403 (application/json)

```
{  
  "code": "403",  
  "message": "Forbidden"  
}
```

- Response 409 (application/json)

```
{  
  "errors": [  
    "string"  
  ],  
  "code": "400",  
  "message": "Bad Request"  
}
```

- Response 500 (application/json)

```
{  
  "code": "500",  
  "message": "Internal Server Error - 1134356675"  
}
```

3.2.3.6 Obtener ficheros de facturas europeas en PDF [GET]

Permite obtener ficheros PDF original de una factura europea (UBL CII).

- API Invoices

```
/v1/invoice/{registryCode}/european-  
invoice/{file}/pdf
```

- Parámetros de path
 - registryCode (string, obligatorio): número de registro de la factura que se quiere evolucionar.
 - file (string, obligatorio): identificación del fichero que acredita la cesión de crédito.

- Petición (text/plain)

```
{}
```

- Response 200 (application/json)

- Body

```
{  
  "uuid": "string",  
  "filename": "string",  
  "content": "string",  
  "mime": "string",  
  "hash": "string"  
}
```

- Response 401 (application/json)

```
{  
  "code": "401",  
  "message": "Unauthorized"  
}
```

- Response 403 (application/json)

```
{  
  "code": "403",  
  "message": "Forbidden"  
}
```

- Response 409 (application/json)

```
{
  "errors": [
    "string"
  ],
  "code": "400",
  "message": "Bad Request"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```


3.3 Grupo Notificaciones

Los recursos ubicados bajo la ruta /notifications y /news están relacionados con la gestión de las notificaciones del sistema a sus usuarios y las noticias de interés de los mismos.

3.3.1 Gestión de notificaciones y noticias

3.3.1.1 Listado de notificaciones [GET]

Permite obtener el listado de todas las notificaciones que el sistema tiene para el usuario logado.

- API notifications

```
/v1/notifications
```

- Parámetros de query
 - site (string, opcional). Puede ser public o private.
- Petición (text/plain)

```
{  
  "site": "public"  
}
```

- Response 200 (application/json)
 - Body

```
{
  "items": [
    {
      "id": 0,
      "title": "string",
      "description": "string",
      "notificationsButton": {
        "text": "string",
        "link": "string"
      },
      "dateFrom": "2025-03-21T10:15:46.138Z",
      "dateTo": "2025-03-21T10:15:46.138Z",
      "createdAt": "string",
      "active": true,
      "fixed": true,
      "notificationLevel": "string",
      "sites": [
        "string"
      ],
      "position": 0
    }
  ]
}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.3.1.2 Listado de noticias [GET]

Permite obtener el listado de todas las noticias que el sistema tiene publicadas.

- API news

```
/v1/news
```

- Parámetros de query
 - site (string, opcional). Puede ser public o private.
- Petición (text/plain)

```
{  
  "site": "private"  
}
```

- Response 200 (application/json)
 - Body

```
{  
  "items": [  
    {  
      "id": 0,  
      "title": "string",  
      "description": "string",  
      "dateFrom": "2025-03-21T10:17:31.667Z",  
      "dateTo": "2025-03-21T10:17:31.667Z",  
      "createdAt": "string",  
      "active": true,  
      "fixed": true,  
      "sites": [  
        "string"  
      ],  
      "position": 0  
    }  
  ]  
}
```

- Response 401 (application/json)

```
{  
  "code": "401",  
  "message": "Unauthorized"  
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.3.1.3 Listado de noticias cortas [GET]

Permite obtener el listado de todas las noticias cortas que el sistema tiene publicadas.

- API slides

```
/v1/slides
```

- Parámetros de query
 - site (string, opcional). Puede ser *public* o *private*.
- Petición (text/plain)

```
{
  "site": "1"
}
```

- Response 200 (application/json)
 - Body

Slides List

Media type

application/json

Controls Accept header.

Example Value

Schema

```
{
  "items": [
    {
      "id": 0,
      "image": "string",
      "title": "string",
      "description": "string",
      "notificationsButton": {
        "text": "string",
        "link": "string"
      },
      "dateFrom": "2025-03-21T10:18:16.068Z",
      "dateTo": "2025-03-21T10:18:16.068Z",
      "active": true,
      "fixed": true,
      "sites": [
        "string"
      ],
      "position": 0
    }
  ]
}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.3.1.4 Listado de estadísticas [GET]

Permite obtener el listado de todas las noticias cortas que el sistema tiene publicadas.

- API slides

```
/v1/statistics
```

- Petición (text/plain)

```
{}
```

- Response 200 (application/json)

- Body

```
{
  "years": [
    {
      "year": 0,
      "invoices": 0,
      "registered": 0,
      "registeredInRcf": 0,
      "verifiedInRcf": 0,
      "receivedInDestination": 0,
      "conformed": 0,
      "paymentProposed": 0,
      "paid": 0,
      "rejected": 0,
      "cancelled": 0,
      "rejectedFace": 0
    }
  ]
}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.4 Grupo Auditorías

Los recursos ubicados bajo la ruta /audit están relacionadas con las solicitudes de auditoría de datos contables de los Registros Contables de Facturas.

3.4.1 Auditorías

3.4.1.1 Listar Auditorías [GET]

Permite obtener el listado de las auditorías de RCF realizadas por un administrador de una unidad.

- API audit-request

```
/v1/audit-request
```

- Parámetros de query string
 - page (string, opcional). Paginación de los resultados
 - limit (string, opcional). Límite de ítems del listado.
 - dicCode (string, opcional): filtro por código DIR.
 - year (string, opcional): filtro por año fiscal.
 - status (string, opcional): filtro por estado de la solicitud de auditoría.

- Petición (text/plain)

```
{}
```

- Response 200 (application/json)

- Body

```
{
  "hash": "string",
  "dirCode": "string",
  "year": 0,
  "status": "string",
  "requestedAt": "2022-07-11T11:26:21.656Z",
  "processedAt": "2022-07-11T11:26:21.656Z",
  "finishedAt": "2022-07-11T11:26:21.656Z",
  "email": "string"
}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.4.1.2 Solicitar nueva auditoría [POST]

Permite cancelar la solicitud de una auditoría que no esté en proceso todavía.

- API audit-request

```
/v1/audit-request
```

- Parámetros de query string

- dirCode (string, obligatorio). Código de la unidad DIR sobre la que se quiere hacer la auditoría.
- year (string, obligatorio). Año para el que se quiere realizar la auditoría.
- email (string, obligatorio). Dirección de correo electrónico a la que se remitirá la confirmación de que la solicitud se ha procesado correctamente.

- Petición (text/plain)

```
{
  "dirCode": "XXXXXXXXXXXX"
  "year": "2020"
  "email": "a@a.com"
}
```


- Response 200 (application/json)

- Body

```
{
  "hash": "string",
  "dirCode": "string",
  "year": 0,
  "status": "string",
  "requestedAt": "2022-07-11T11:26:21.656Z",
  "processedAt": "2022-07-11T11:26:21.656Z",
  "finishedAt": "2022-07-11T11:26:21.656Z",
  "email": "string"
}
```

- Response 400 (application/json)

```
{
  "dirCode": [
    "string"
  ],
  "year": [
    "string"
  ],
  "email": [
    "string"
  ],
  "errors": [
    "string"
  ],
  "code": "400",
  "message": "Bad Request"
}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 404 (application/json)

```
{
  "code": "404",
  "message": "Not Found"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.4.1.3 Ver detalle de auditoría [GET]

Permite visualizar el detalle de una auditoría.

- API audit-request

```
/v1/audit-request/{hash}
```

- Parámetros de query string
 - hash (string, obligatorio). Hash de la solicitud de auditoría
- Petición (text/plain)

```
{
  "hash": "XXXXXXXXXXXX"
}
```

- Response 200 (application/json)

- Body

```
{
  "hash": "string",
  "dirCode": "string",
  "year": 0,
  "status": "string",
  "requestedAt": "2022-07-11T11:26:21.656Z",
  "processedAt": "2022-07-11T11:26:21.656Z",
  "finishedAt": "2022-07-11T11:26:21.656Z",
  "email": "string"
}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 404 (application/json)

```
{
  "code": "404",
  "message": "Not Found"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.4.1.4 Solicitar cancelación de auditorías [PATCH]

Permite cancelar la solicitud de una auditoría que no esté en proceso todavía.

- API audit-request

```
/v1/audit-request/{hash}/cancel
```

- Parámetros de query string
 - hash (string, obligatorio). Hash de la solicitud de auditoría
- Petición (text/plain)

```
{
  "hash": "XXXXXXXXXXXX"
}
```

- Response 200 (application/json)

- Body

```
{
  "hash": "string",
  "dirCode": "string",
  "year": 0,
  "status": "string",
  "requestedAt": "2022-07-11T11:26:21.656Z",
  "processedAt": "2022-07-11T11:26:21.656Z",
  "finishedAt": "2022-07-11T11:26:21.656Z",
  "email": "string"
}
```

- Response 401 (application/json)

```
{  
  "code": "401",  
  "message": "Unauthorized"  
}
```

- Response 403 (application/json)

```
{  
  "code": "403",  
  "message": "Forbidden"  
}
```

- Response 404 (application/json)

```
{  
  "code": "404",  
  "message": "Not Found"  
}
```

- Response 500 (application/json)

```
{  
  "code": "500",  
  "message": "Internal Server Error - 1134356675"  
}
```

3.5 Grupo Usuarios

Los recursos ubicados bajo la ruta `/user` están relacionadas con los grupos de usuarios.

3.5.1 Gestión de usuarios

3.5.1.1 Asignar rol [POST]

Permite asignar un rol a un usuario en el portal.

- API

```
/v1/user-role
```

- Parámetros de query
 - Identifier (string, opcional)
 - Name (string, opcional)
 - surname1 (string, opcional)
 - surname2 (string, opcional)
 - dirCode (string, opcional)
 - profile (string, opcional)

- Petición (text/plain):

```
{  
  "identifier": "string",  
  "name": "string",  
  "surname1": "string",  
  "surname2": "string",  
  "dirCode": "string",  
  "profile": "string"  
}
```

- Response 201 (application/json)

- Body

```
{
  "identifier": "string",
  "name": "string",
  "surname1": "string",
  "surname2": "string",
  "email": "string",
  "profile": "string",
  "dirCode": "string"
}
```

- Response 400 (application/json)

```
{
  "identifier": [
    "string"
  ],
  "profile": [
    "string"
  ],
  "dirCode": [
    "string"
  ],
  "errors": [
    "string"
  ],
  "code": "400",
  "message": "Bad Request"
}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 404 (application/json)

```
{
  "code": "404",
  "message": "Not Found"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.5.1.2 Listar usuarios de una unidad DIR [GET]

Este recurso listar a los usuarios de una unidad DIR3.

- API

```
/v1/user/{identifier}
```

- Parámetros de path
 - Identifier (string, opcional): identificador del usuario.
- Petición (text/plain)

```
{
  "dirCode": "XXXXXXXXXXXX"
  "year": "2020"
  "email": "a@a.com"
}
```

- Response 200 (application/json)
 - Body

```
{
  "hash": "string",
  "dirCode": "string",
  "year": 0,
  "status": "string",
  "requestedAt": "2022-07-11T11:26:21.656Z",
  "processedAt": "2022-07-11T11:26:21.656Z",
  "finishedAt": "2022-07-11T11:26:21.656Z",
  "email": "string"
}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 404 (application/json)

```
{
  "code": "404",
  "message": "Not Found"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.5.1.3 Listar usuarios de una unidad DIR por perfil [GET]

Permite obtener un listado de usuario de una unidad DIR3 por perfil.

- API audit-request

```
/v1/dir/{dirCode}/user-role
```

- Parámetros de path
 - dirCode (string, obligatorio). Código de la unidad DIR sobre la que se quiere hacer la auditoría.
- Parámetros de la query:
 - Identifier (string, obligatorio): identificador del usuario
- Petición (text/plain)

```
{
  "dirCode": "XXXXXXXXXXXX"
}
```


- Response 200 (application/json)

- Body

```
{
  "count": 0,
  "items": [
    {
      "identifier": "string",
      "name": "string",
      "surname1": "string",
      "surname2": "string",
      "email": "string",
      "profile": "string",
      "dirCode": "string"
    }
  ]
}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 404 (application/json)

```
{
  "code": "404",
  "message": "Not Found"
}
```

- Response 500 (application/json)

```
{
  "code": "500",
  "message": "Internal Server Error - 1134356675"
}
```

3.5.1.4 Eliminar usuario [DELETE]

Permite cancelar la solicitud de una auditoría que no esté en proceso todavía.

- API audit-request

```
/v1/user-profile/{identifier}-{dirCode}-{profile}
```

- Parámetros de path
 - dirCode (string, obligatorio). Código de la unidad DIR sobre la que se quiere hacer la auditoría.
 - identifier (string, obligatorio). Identificador del usuario.
 - profile (string, obligatorio). Rol del identificador del usuario.

- Petición (text/plain)

```
{}
```

- Response 204 (application/json)

- Body

```
{}
```

- Response 401 (application/json)

```
{
  "code": "401",
  "message": "Unauthorized"
}
```

- Response 403 (application/json)

```
{
  "code": "403",
  "message": "Forbidden"
}
```

- Response 404 (application/json)

```
{
  "code": "404",
  "message": "Not Found"
}
```

- Response 500 (application/json)

```
{  
  "code": "500",  
  "message": "Internal Server Error - 1134356675"  
}
```

4 ANEXO I

El XML firmado deberá contener las siguientes etiquetas:

```
<?xml version="1.0" encoding="UTF-8"?><AFIRMA Id="AFIRMA-Root-1cc55eb0-0290-4082-a7f4-7c3a17643ea3">
<association-rcf-proof>
  <responsibility-conditions>
    <conditions>Solicita la vinculación del Registro Contable de Facturas Automatizado</conditions>
  <association-rcf>
    <oc>P000000009</oc>
    <signer>4e156acfb10d56d814534474f21fe825a1745c47</signer>
    <system>
      <rcf>62e2828baadb1</rcf>
    </system>
  </association-rcf>
</responsibility-conditions>
</association-rcf-proof>
```

Donde el <signer> es un hash del certificado que firma la petición es un sha-1 de la clave pública.

5 ANEXO II

El siguiente anexo muestra un ejemplo en java de generación del token.

```
import java.io.File;
import java.io.FileInputStream;
import java.security.KeyStore;
import java.security.PrivateKey;
import java.security.Signature;
import java.util.Arrays;
import java.util.Base64;
import java.util.Enumeration;

def project = context.testCase.testSuite.project

this.getClass().classLoader.addURL(new
File(context.expand(project.resourceRoot) +
"/libs/google-http-client-jackson2-
1.28.0.jar").toURL());
this.getClass().classLoader.addURL(new
File(context.expand(project.resourceRoot) +
"/libs/google-http-client-1.28.0.jar").toURL());

File key_pem = new
File(context.expand(project.resourceRoot) + "Ruta a
la clave publica del certificado");

String pem =
org.apache.commons.io.IOUtils.toString(new
FileInputStream(key_pem));

Signature sig =
Signature.getInstance("SHA256WithRSA");

KeyStore ks =
java.security.KeyStore.getInstance("PKCS12");
File key = new
File(context.expand(project.resourceRoot) + "Ruta
a la clave privada del certificado");
ks.load(new
FileInputStream(key),"alias_certificado".toCharArray());

Enumeration aliases = ks.aliases();
String keyAlias = "";
while (aliases.hasMoreElements()) {
    keyAlias = (String) aliases.nextElement();
    System.out.println(keyAlias);
}
//sig.initSign((PrivateKey)ks.getKey(keyAlias,
"alias_certificado".toCharArray()));
```

```
PrivateKey privateKey ;
com.google.api.client.json.webtoken.JsonWebSignatur
e.Header header = new
com.google.api.client.json.webtoken.JsonWebSignatur
e.Header();
header.setAlgorithm("RS256");
//header.setX509Certificates(Arrays.asList(new
String[]
{Base64.getEncoder().encodeToString((ks.getCertific
ate(keyAlias).getEncoded()))});
def cert_enc =
Base64.getEncoder().encodeToString(ks.getCertificat
e(keyAlias).getEncoded());
def res = [];
res.add(cert_enc);
List certs = Arrays.asList(res);
header.setX509Certificates(certs);
com.google.api.client.json.webtoken.JsonWebToken.Pa
yload payload = new
com.google.api.client.json.webtoken.JsonWebToken.Pa
yload();
long now = System.currentTimeMillis() / 1000L;
long exp = now + 7200;
payload.set("exp", exp);
payload.set("iat", now);
com.google.api.client.json.JsonFactory JSON_FACTORY
=
com.google.api.client.json.jackson2.JacksonFactory.
getDefaultInstance();
def token =
com.google.api.client.json.webtoken.JsonWebSignatur
e.signUsingRsaSha256((PrivateKey)ks.getKey(keyAlias
, "alias_certificado".toCharArray()) ,
JSON_FACTORY, header, payload);
context.testCase.setPropertyValue('token', new
String(token));
log.info(token)
```

