

**Ms Trade and Working Capital Documents – Capa Exposición - ApiGee**

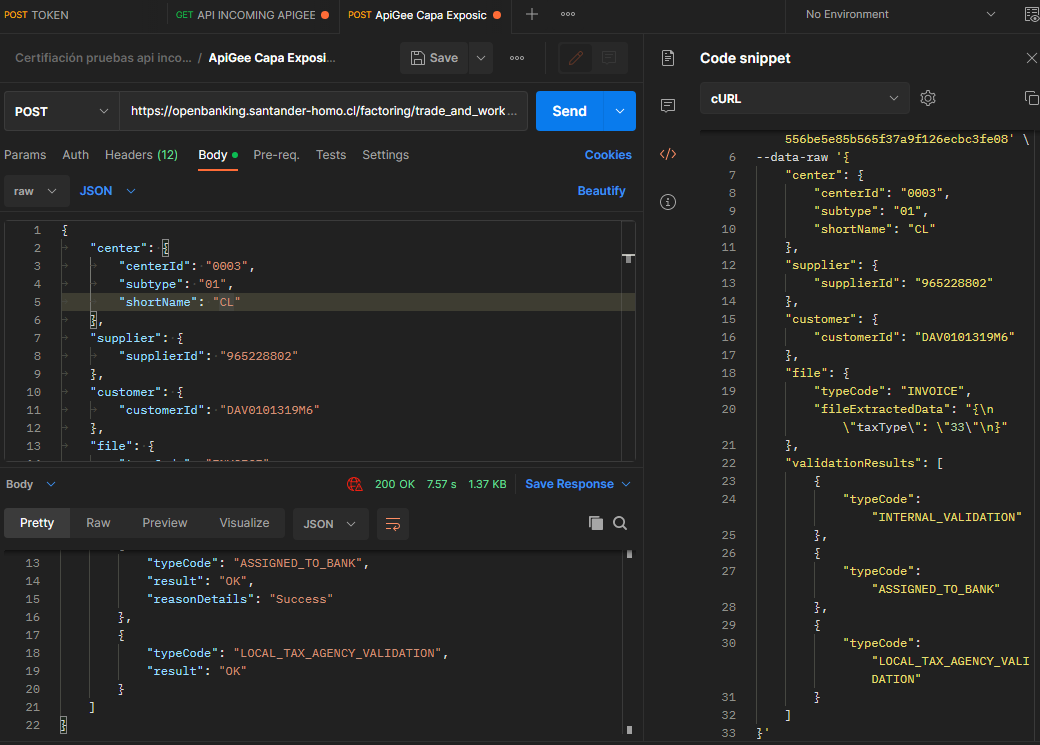
**Pruebas de Integración**

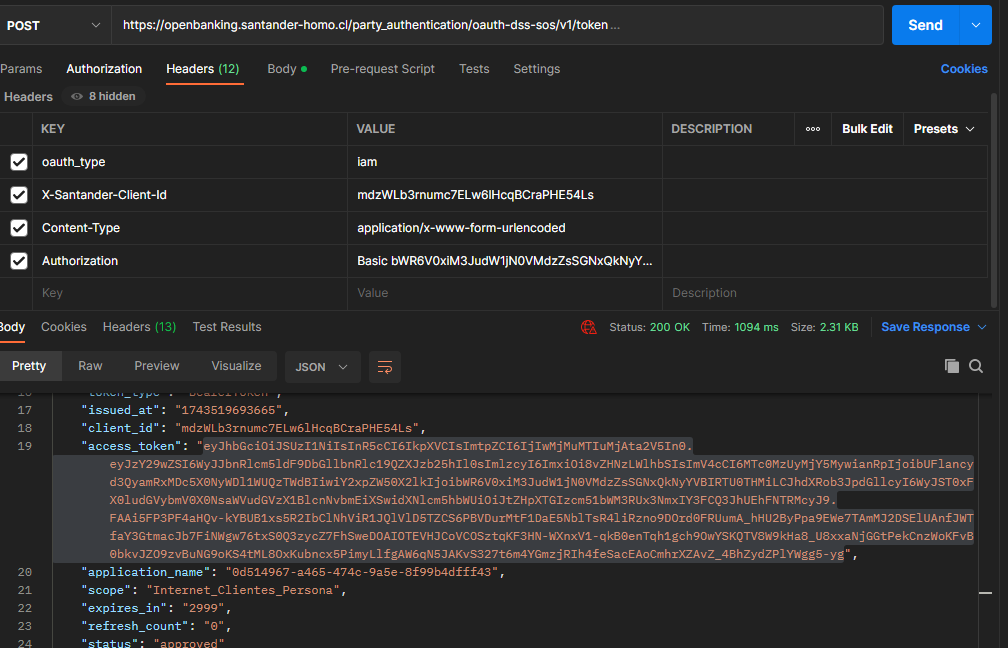
Versión 1.0.0

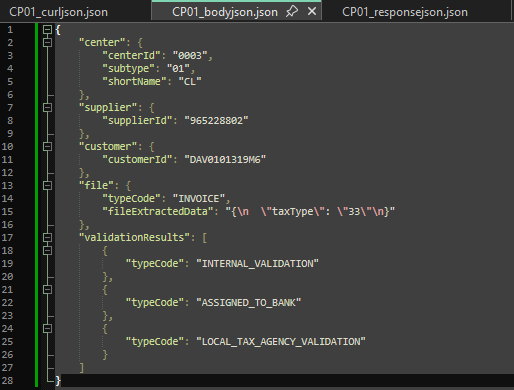
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | CP 01 |  | | ¿Despliegue? | No |   Caso de prueba correcto: shortName CL |
| **Descripción:**  A través de postman, haremos una petición correcta, con token válido, “x\_santander\_client\_id” válido, content-type como application-json válido, un bearer token válido y un body igualmente válido al endpoint del ms para obtener la información relacionada al contexto shortName CL |
| **Prerrequisitos:**   1. Generar un token válido. 2. Tener el dato “x\_santander\_client\_id” correcto. 3. Tener un body valido. |
| **Pasos:**   1. Generar nuevo token válido. 2. Armar el “request” en postman. 3. Ejecutar. |
| **Resultado esperado:**  Código de respuesta http 200 “ok” con la data consistente según el “id” de cuenta para ms trading and working capital documents |
| **Resultado obtenido:**  Código de respuesta http 200 “ok” con la data consistente según el “id” de cuenta para ms trading and working capital documents |

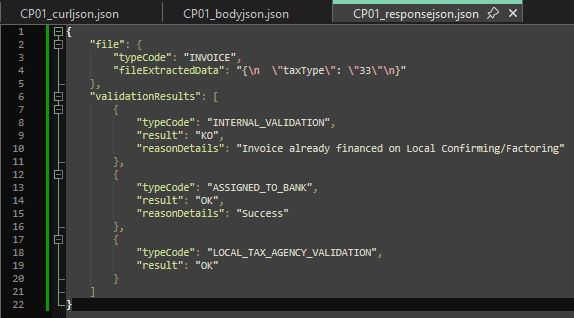
**Archivos relacionados**

* CP01\_CL\_curljson.json
* CP01\_CL\_bodyjson.json
* CP01\_CL\_responsejson.json

w



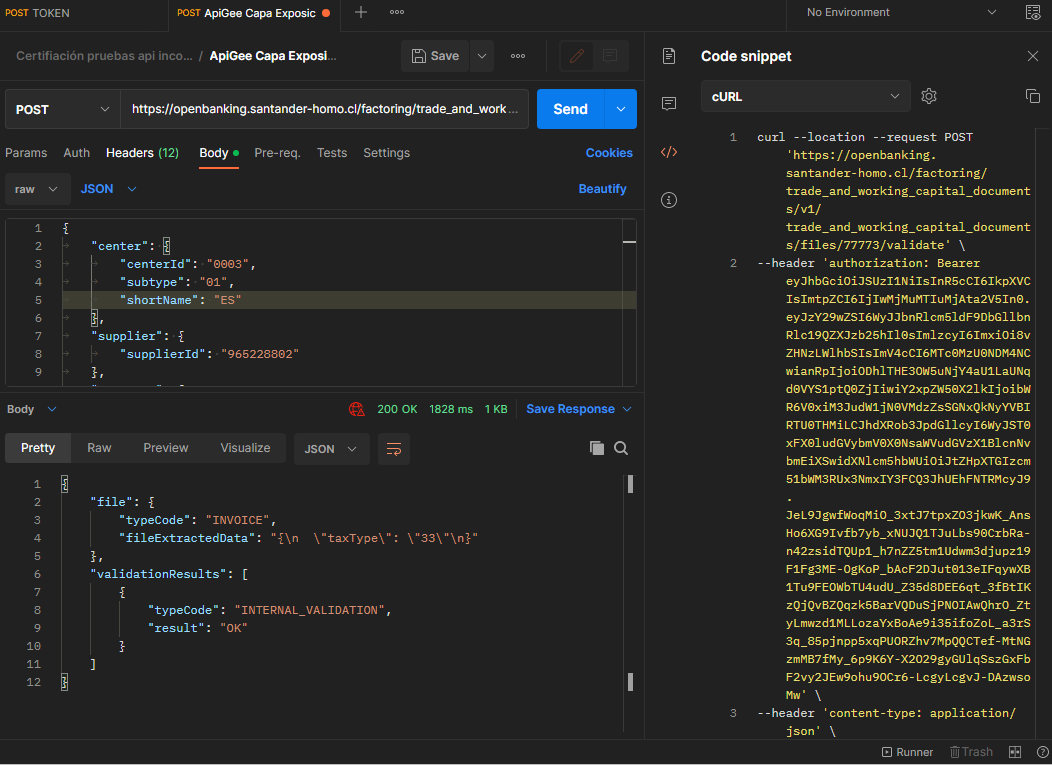


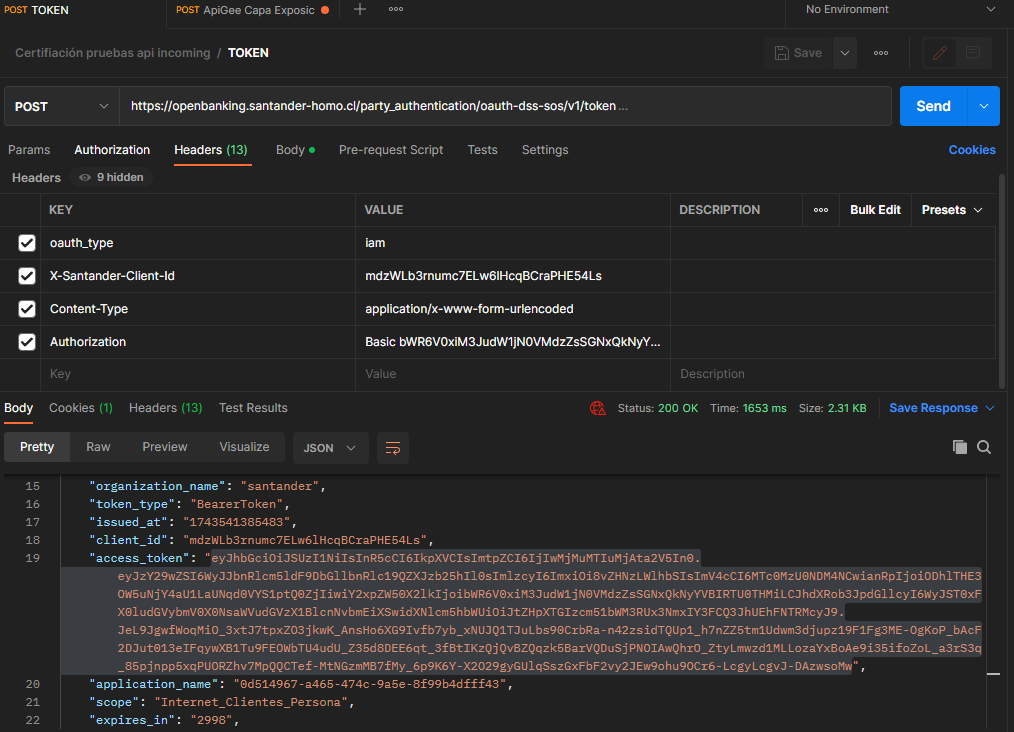


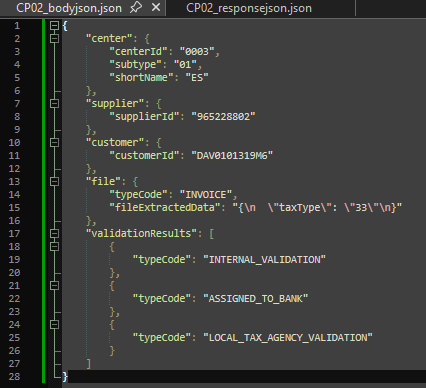
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | CP 02 |  | | ¿Despliegue? | No |   Caso de prueba correcto: shortName ES |
| **Descripción:**  A través de postman, haremos una petición correcta, con token válido, “x\_santander\_client\_id” válido, content-type como application-json válido, un bearer token válido y un body igualmente válido al endpoint del ms para obtener la información relacionada al contexto shortName ES |
| **Prerrequisitos:**   1. Generar un token válido. 2. Tener el dato “x\_santander\_client\_id” correcto. 3. Tener un body valido. |
| **Pasos:**   1. Generar nuevo token válido. 2. Armar el “request” en postman. 3. Ejecutar. |
| **Resultado esperado:**  Código de respuesta http 200 “ok” con la data consistente según el “id” de cuenta para ms trading and working capital documents |
| **Resultado obtenido:**  Código de respuesta http 200 “ok” con la data consistente según el “id” de cuenta para ms trading and working capital documents |

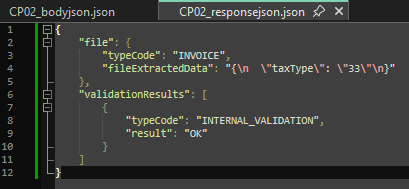
**Archivos relacionados:**

* CP02\_ES\_curljson.json
* CP02\_ES\_bodyjson.json
* CP02\_ES\_responsejson.json





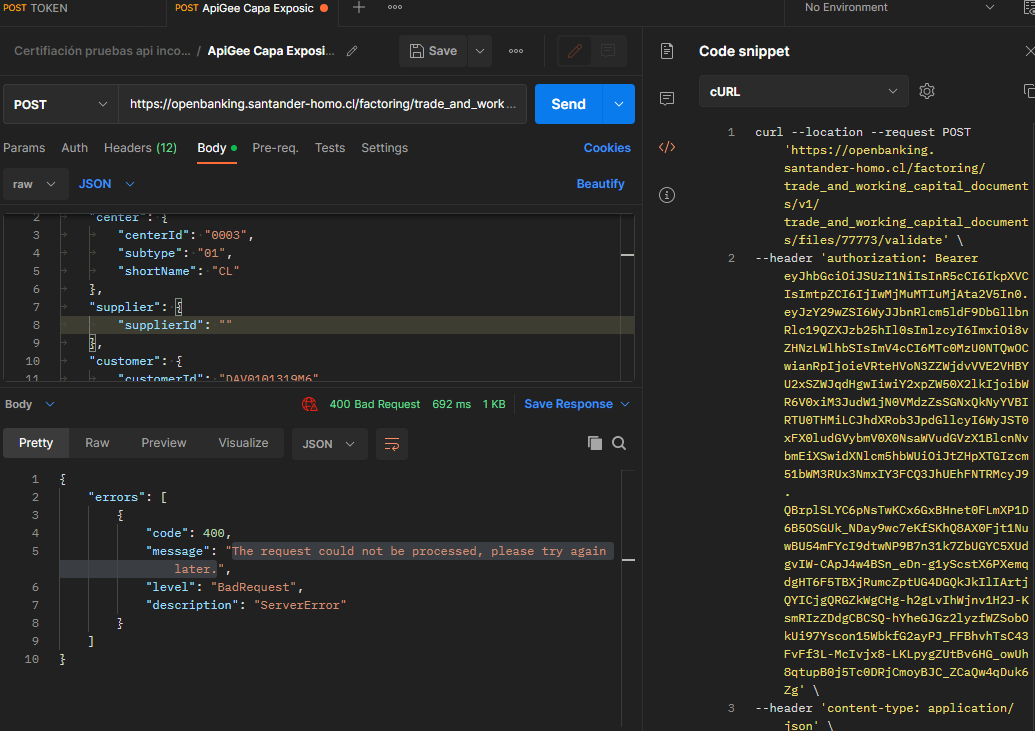


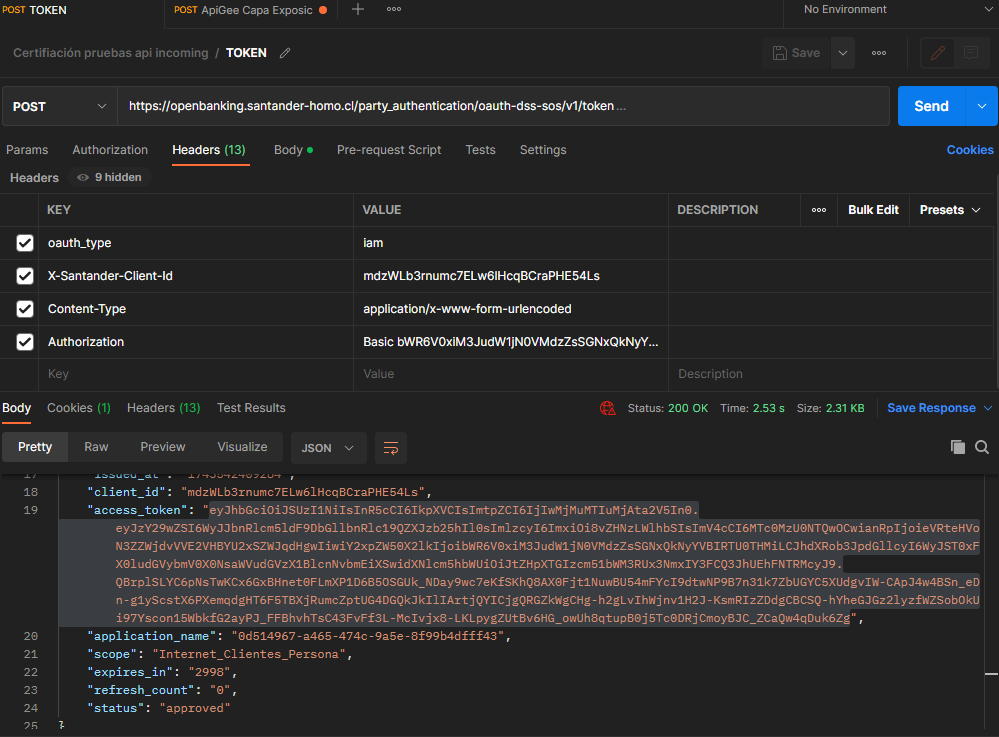


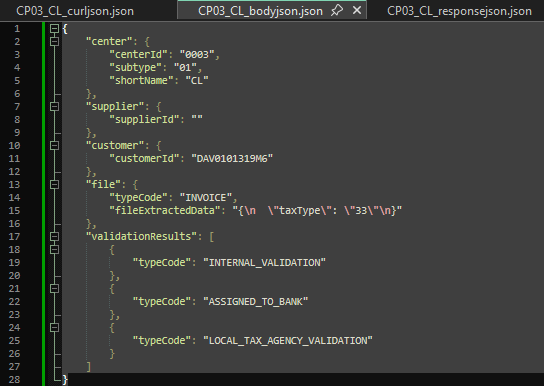
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | CP 03 |  | | ¿Despliegue? | No |   Caso de prueba incorrecto: shortName CL supplierId vacío |
| **Descripción:**  A través de postman, haremos una petición incorrecta, con token válido, “x\_santander\_client\_id” válido, content-type como application-json válido, un bearer token válido y un body inválido cuando el supplierId es vacío, al endpoint del ms para obtener la información relacionada al contexto shortName CL |
| **Prerrequisitos:**   1. Generar un token válido. 2. Tener el dato “x\_santander\_client\_id” correcto. 3. Tener un body valido. |
| **Pasos:**   1. Generar nuevo token válido. 2. Armar el “request” en postman. 3. Ejecutar. |
| **Resultado esperado:**  Código de respuesta http 400 “bad request” con mensaje tipo “The request could not be processed, please try again later.”, level y description, asociados a BadRequest y a ServerError respectivamente ms trading and working capital documents |
| **Resultado obtenido:**  Código de respuesta http 400 “bad request” con mensaje tipo “The request could not be processed, please try again later.”, level y description, asociados a BadRequest y a ServerError respectivamente ms trading and working capital documents |

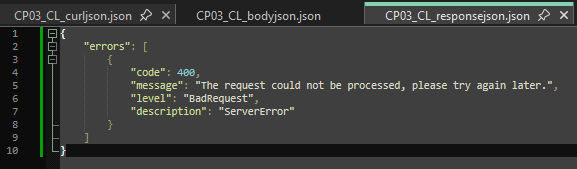
**Archivos relacionados:**

* CP03\_CL\_curljson.json
* CP03\_CL\_bodyjson.json
* CP03\_CL\_responsejson.json





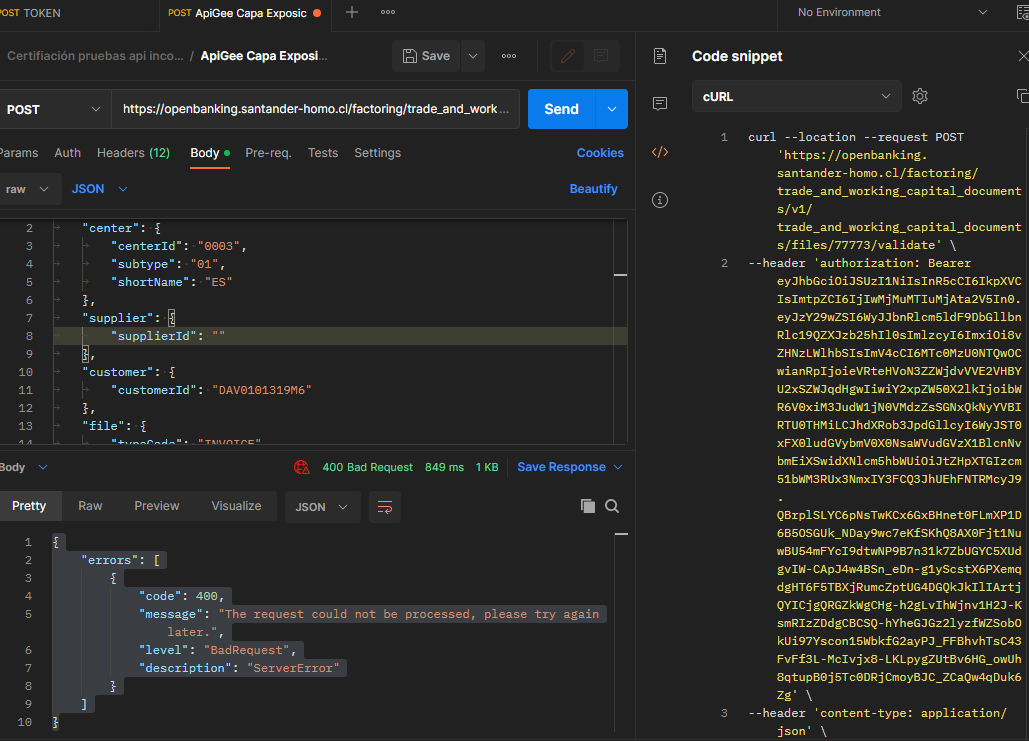


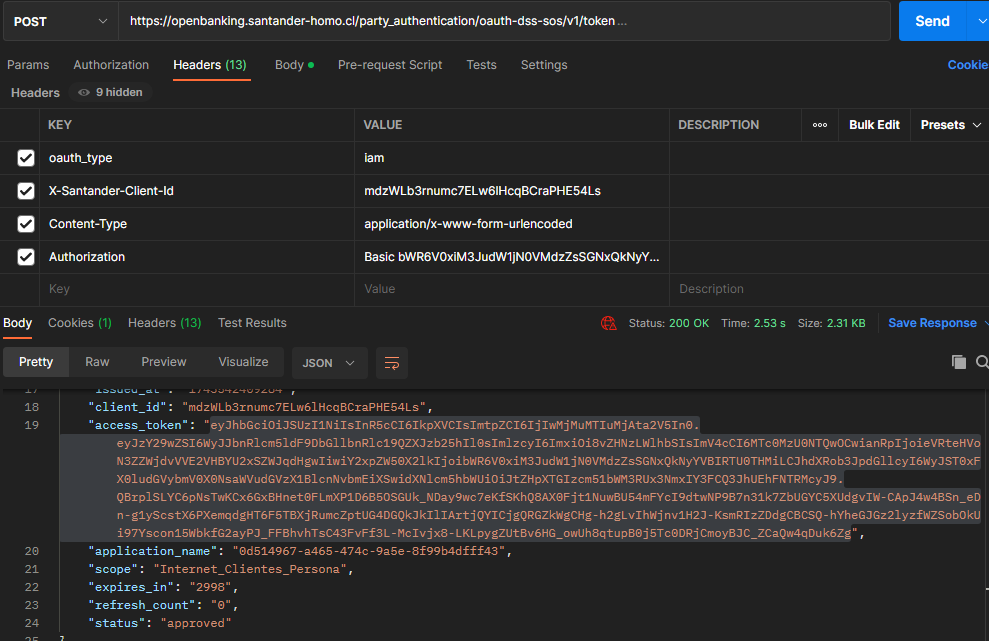


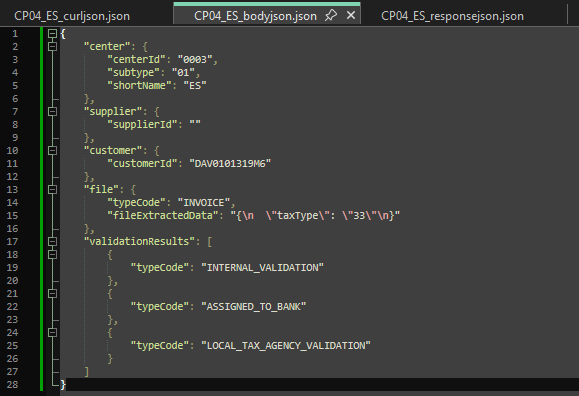
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | CP 04 |  | | ¿Despliegue? | No |   Caso de prueba incorrecto: shortName ES supplierId vacío |
| **Descripción:**  A través de postman, haremos una petición incorrecta, con token válido, “x\_santander\_client\_id” válido, content-type como application-json válido, un bearer token válido y un body inválido cuando el supplierId es vacío, al endpoint del ms para obtener la información relacionada al contexto shortName ES |
| **Prerrequisitos:**   1. Generar un token válido. 2. Tener el dato “x\_santander\_client\_id” correcto. 3. Tener un body valido. |
| **Pasos:**   1. Generar nuevo token válido. 2. Armar el “request” en postman. 3. Ejecutar. |
| **Resultado esperado:**  Código de respuesta http 400 “bad request” con mensaje tipo “The request could not be processed, please try again later.”, level y description, asociados a BadRequest y a ServerError respectivamente ms trading and working capital documents |
| **Resultado obtenido:**  Código de respuesta http 400 “bad request” con mensaje tipo “The request could not be processed, please try again later.”, level y description, asociados a BadRequest y a ServerError respectivamente ms trading and working capital documents |

**Archivos relacionados:**

* CP04\_ES\_curljson.json
* CP04\_ES\_bodyjson.json
* CP04\_ES\_responsejson.json





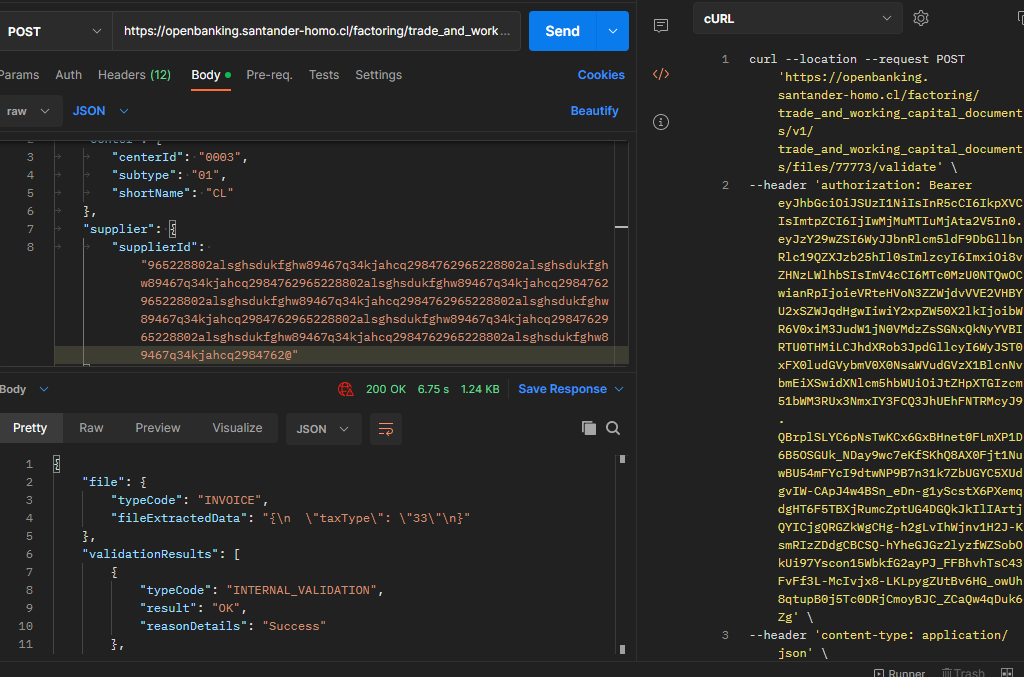


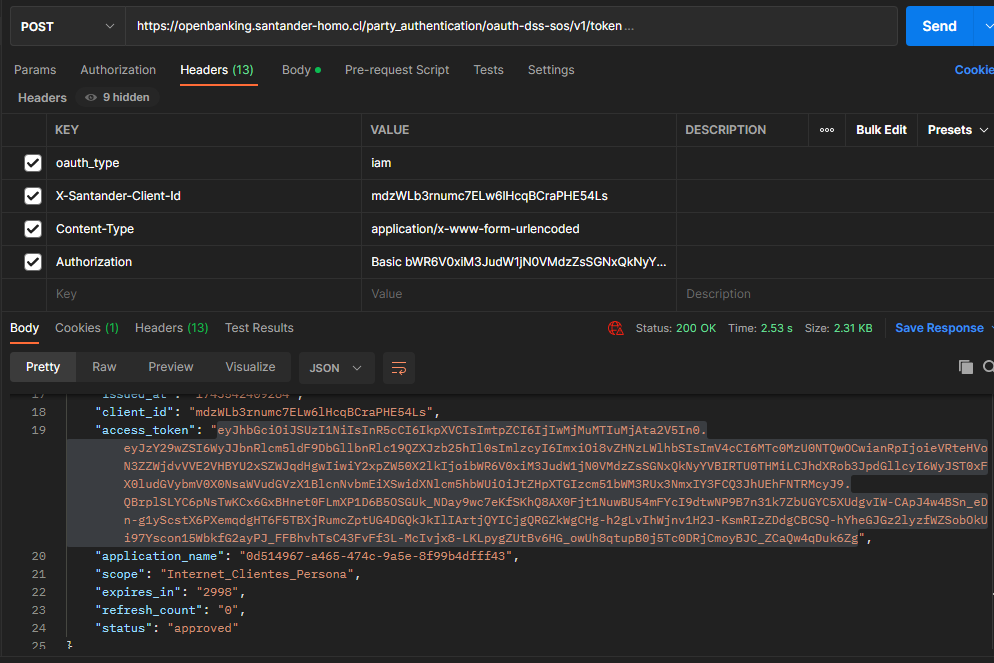


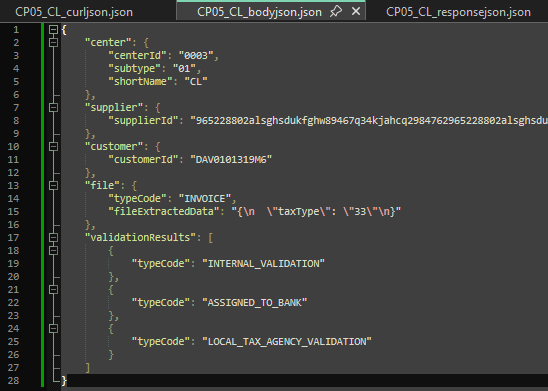
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | CP 05 |  | | ¿Despliegue? | No |   Caso de prueba incorrecto: shortName CL supplierId muy largo, alfanumérico y con carácter(es) especial(es) |
| **Descripción:**  A través de postman, haremos una petición correcta, con token válido, “x\_santander\_client\_id” válido, content-type como application-json válido, un bearer token válido y un body incorrecto cuando el supplierId es muy largo y del tipo alfanumérico con carácter(es) especial(es) inclusive, al endpoint del ms para obtener la información relacionada al contexto shortName CL |
| **Prerrequisitos:**   1. Generar un token válido. 2. Tener el dato “x\_santander\_client\_id” correcto. 3. Tener un body valido. |
| **Pasos:**   1. Generar nuevo token válido. 2. Armar el “request” en postman. 3. Ejecutar. |
| **Resultado esperado:**  Código de respuesta http 200 “ok” con la data siguiente en el apartado validationResults: KO para LOCAL\_TAX\_AGENY\_VALIDATION y para ASSIGNED\_TO\_BANK, y "ok” para INTERNAL\_VALIDATION |
| **Resultado obtenido:**  Código de respuesta http 200 “ok” con la data siguiente en el apartado validationResults: KO para LOCAL\_TAX\_AGENY\_VALIDATION y para ASSIGNED\_TO\_BANK, y "ok” para INTERNAL\_VALIDATION |

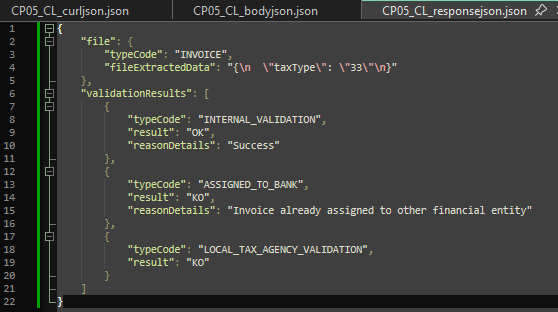
**Archivos relacionados:**

* CP05\_CL\_curljson.json
* CP05\_CL\_bodyjson.json
* CP05\_CL\_responsejson.json





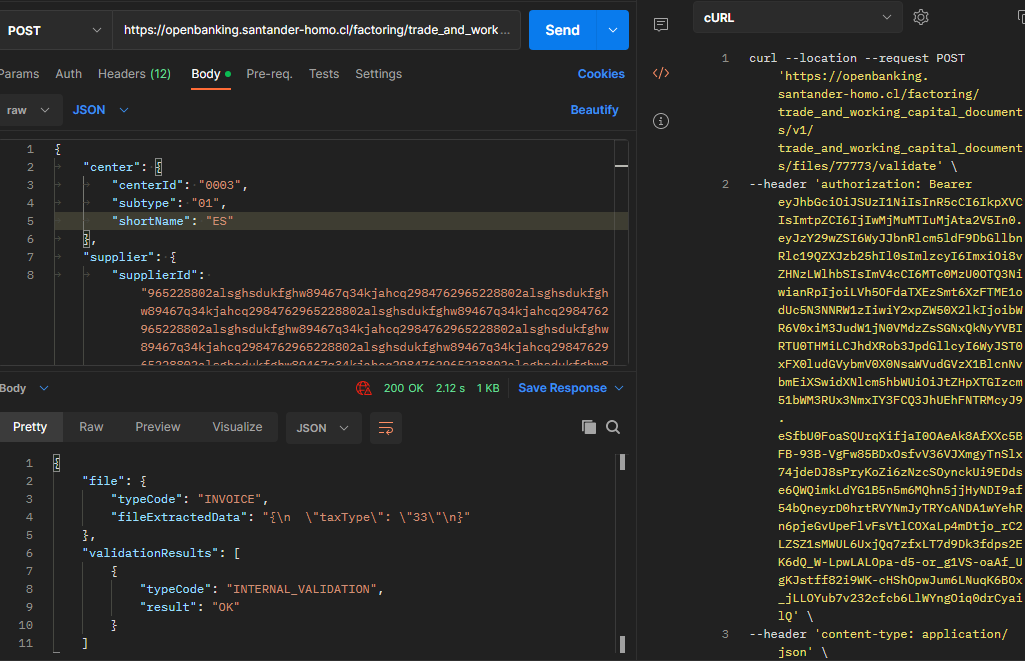


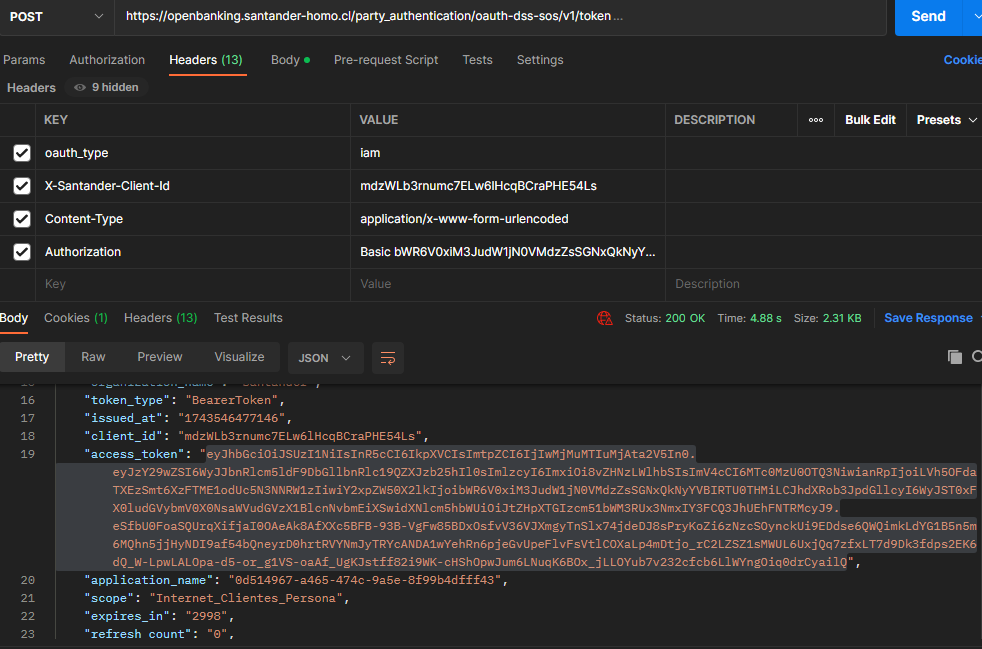


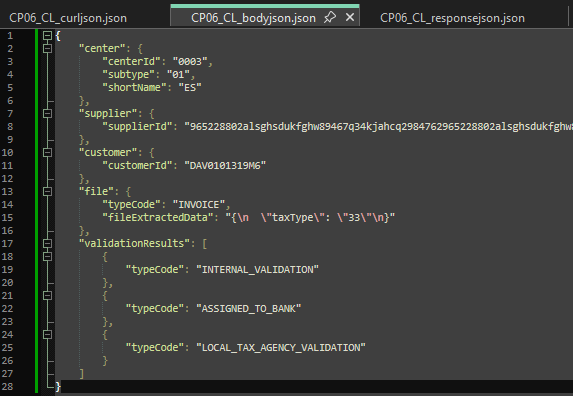
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | CP 06 |  | | ¿Despliegue? | No |   Caso de prueba incorrecto: shortName ES supplierId muy largo, alfanumérico y con carácter(es) especial(es) |
| **Descripción:**  A través de postman, haremos una petición correcta, con token válido, “x\_santander\_client\_id” válido, content-type como application-json válido, un bearer token válido y un body incorrecto cuando el supplierId es muy largo y del tipo alfanumérico con carácter(es) especial(es) inclusive, al endpoint del ms para obtener la información relacionada al contexto shortName ES |
| **Prerrequisitos:**   1. Generar un token válido. 2. Tener el dato “x\_santander\_client\_id” correcto. 3. Tener un body valido. |
| **Pasos:**   1. Generar nuevo token válido. 2. Armar el “request” en postman. 3. Ejecutar. |
| **Resultado esperado:**  Código de respuesta http 200 “ok” con la data siguiente en el apartado validationResults: "ok” para INTERNAL\_VALIDATION |
| **Resultado obtenido:**  Código de respuesta http 200 “ok” con la data siguiente en el apartado validationResults: "ok” para INTERNAL\_VALIDATION |

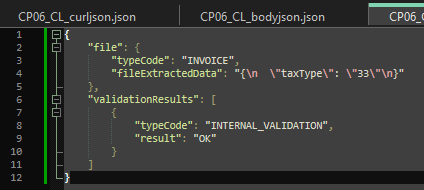
**Archivos relacionados:**

* CP06\_ES\_curljson.json
* CP06\_ES\_bodyjson.json
* CP06\_ES\_responsejson.json





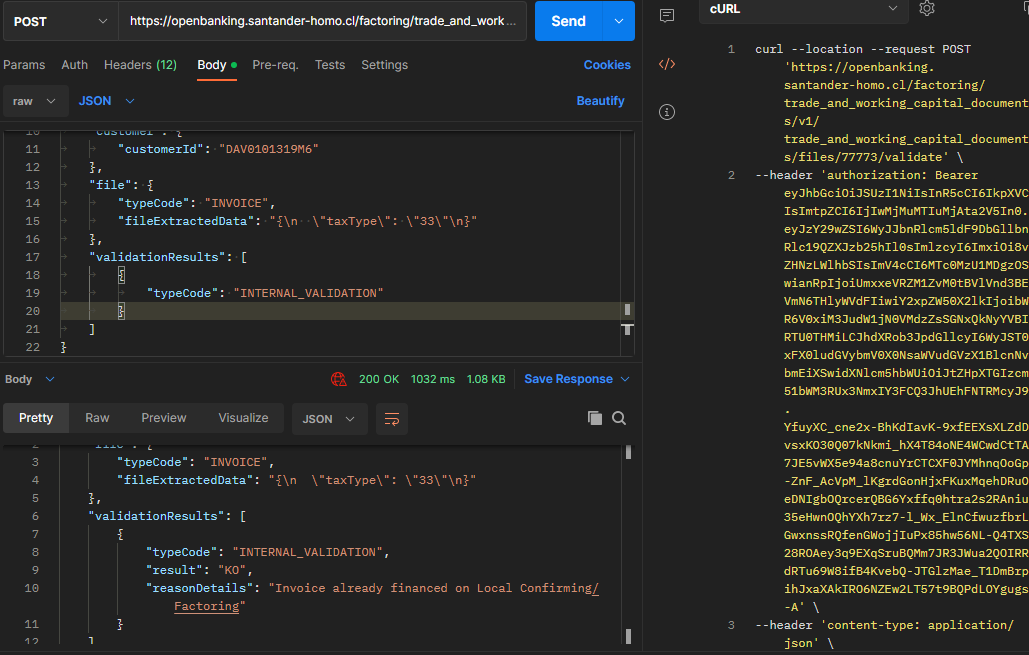


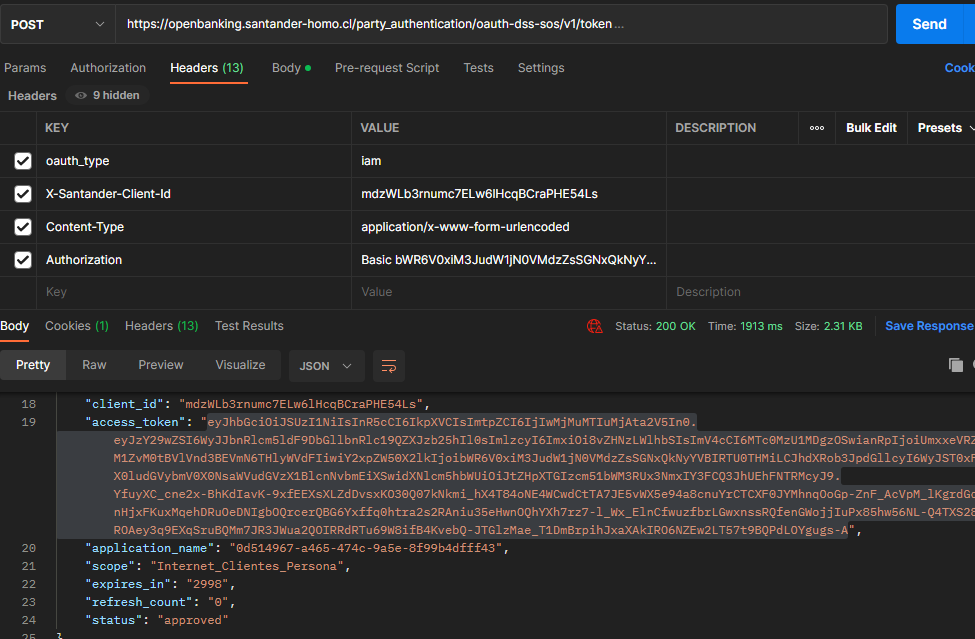


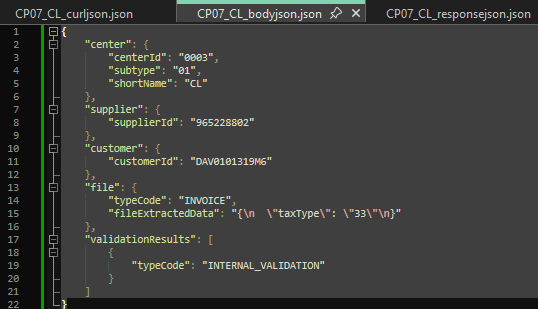
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | CP 07 |  | | ¿Despliegue? | No |   Caso de prueba incorrecto: shortName CL con validationResults en typeCode “INTERNAL\_VALIDATION” (y solo éste) |
| **Descripción:**  A través de postman, haremos una petición correcta, con token válido, “x\_santander\_client\_id” válido, content-type como application-json válido, un bearer token válido y un body correcto cuando validationResults en typeCode “INTERNAL\_VALIDATION” (y solo éste), al endpoint del ms para obtener la información relacionada al contexto shortName CL |
| **Prerrequisitos:**   1. Generar un token válido. 2. Tener el dato “x\_santander\_client\_id” correcto. 3. Tener un body valido. |
| **Pasos:**   1. Generar nuevo token válido. 2. Armar el “request” en postman. 3. Ejecutar. |
| **Resultado esperado:**  Código de respuesta http 200 “ok” con el response validationResults en INTERNAL\_VALIDATION dado en el body un validationResults en INTERNAL\_VALIDATION, es decir, un INTERNAL\_VALIDATION para un INTERNAL\_VALIDATION (y no más, 1:1) |
| **Resultado obtenido:**  Código de respuesta http 200 “ok” con el response validationResults en INTERNAL\_VALIDATION dado en el body un validationResults en INTERNAL\_VALIDATION, es decir, un INTERNAL\_VALIDATION para un INTERNAL\_VALIDATION (y no más, 1:1) |

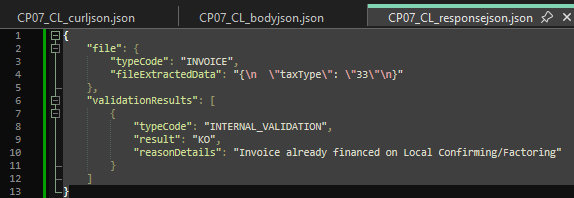
**Archivos relacionados:**

* CP07\_CL\_curljson.json
* CP07\_CL\_bodyjson.json
* CP07\_CL\_responsejson.json





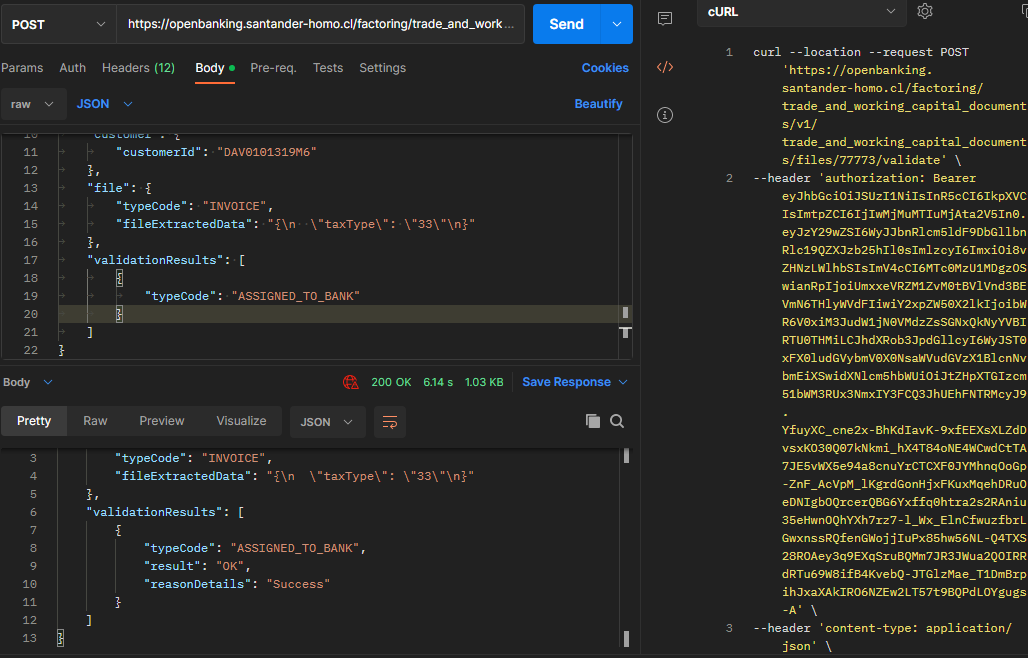


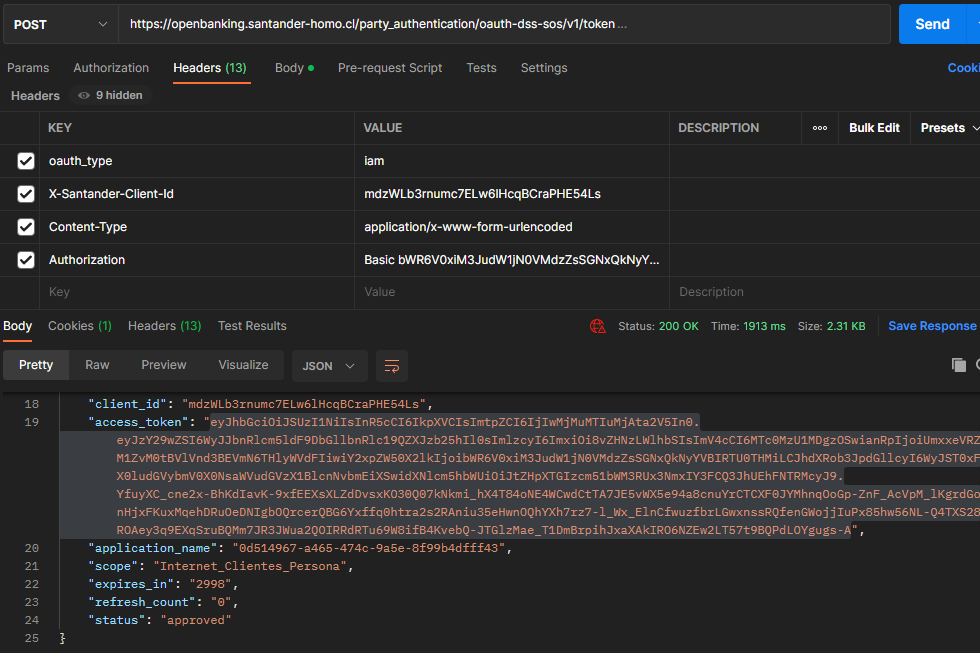


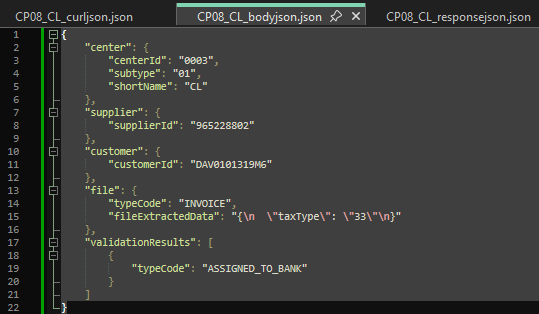
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | CP 08 |  | | ¿Despliegue? | No |   Caso de prueba incorrecto: shortName CL con validationResults en typeCode “ASSIGNED\_TO\_BANK” (y solo éste) |
| **Descripción:**  A través de postman, haremos una petición correcta, con token válido, “x\_santander\_client\_id” válido, content-type como application-json válido, un bearer token válido y un body correcto cuando validationResults en typeCode “ASSIGNED\_TO\_BANK” (y solo éste), al endpoint del ms para obtener la información relacionada al contexto shortName CL |
| **Prerrequisitos:**   1. Generar un token válido. 2. Tener el dato “x\_santander\_client\_id” correcto. 3. Tener un body valido. |
| **Pasos:**   1. Generar nuevo token válido. 2. Armar el “request” en postman. 3. Ejecutar. |
| **Resultado esperado:**  Código de respuesta http 200 “ok” con el response validationResults en ASSIGNED\_TO\_BANK dado en el body un validationResults en ASSIGNED\_TO\_BANK, es decir, un ASSIGNED\_TO\_BANK para un ASSIGNED\_TO\_BANK (y no más, 1:1) |
| **Resultado obtenido:**  Código de respuesta http 200 “ok” con el response validationResults en ASSIGNED\_TO\_BANK dado en el body un validationResults en ASSIGNED\_TO\_BANK, es decir, un INTERNAL\_VALIDATION para un ASSIGNED\_TO\_BANK (y no más, 1:1) |

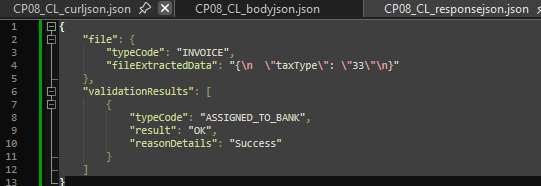
**Archivos relacionados:**

* CP08\_CL\_curljson.json
* CP08\_CL\_bodyjson.json
* CP08\_CL\_responsejson.json





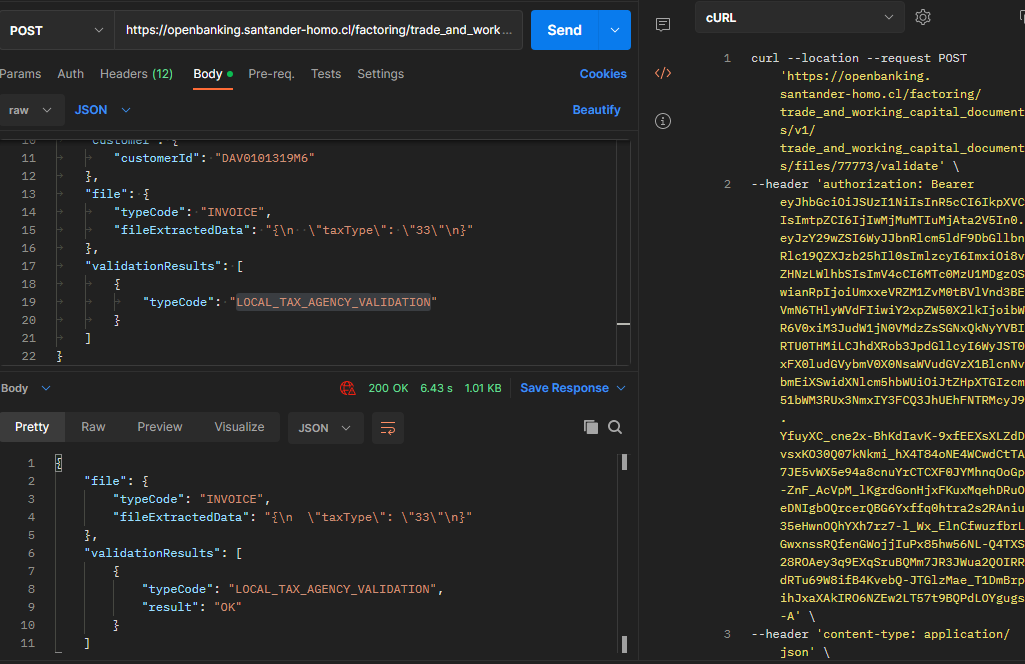


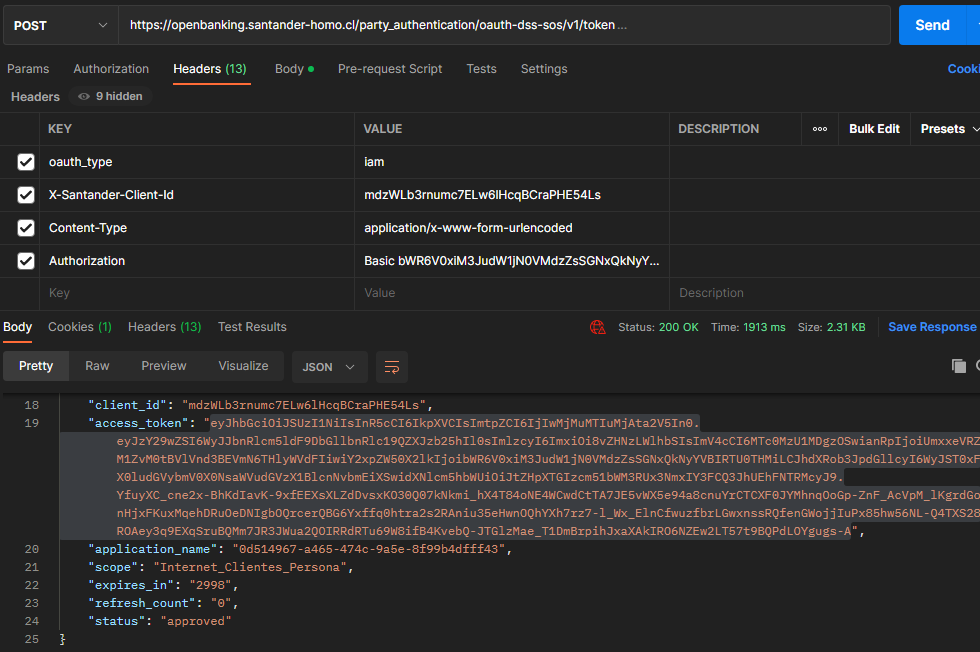


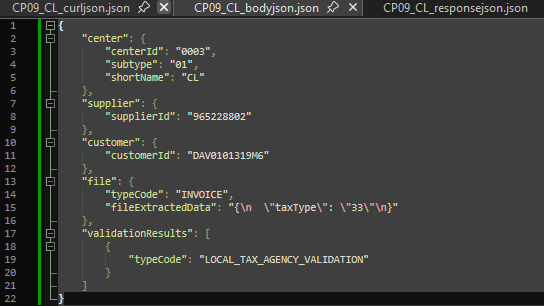
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | CP 09 |  | | ¿Despliegue? | No |   Caso de prueba incorrecto: shortName CL con validationResults en typeCode “LOCAL\_TAX\_AGENCY\_VALIDATION” (y solo éste) |
| **Descripción:**  A través de postman, haremos una petición correcta, con token válido, “x\_santander\_client\_id” válido, content-type como application-json válido, un bearer token válido y un body correcto cuando validationResults en typeCode “LOCAL\_TAX\_AGENCY\_VALIDATION” (y solo éste), al endpoint del ms para obtener la información relacionada al contexto shortName CL |
| **Prerrequisitos:**   1. Generar un token válido. 2. Tener el dato “x\_santander\_client\_id” correcto. 3. Tener un body valido. |
| **Pasos:**   1. Generar nuevo token válido. 2. Armar el “request” en postman. 3. Ejecutar. |
| **Resultado esperado:**  Código de respuesta http 200 “ok” con el response validationResults en LOCAL\_TAX\_AGENCY\_VALIDATION dado en el body un validationResults en LOCAL\_TAX\_AGENCY\_VALIDATION, es decir, un LOCAL\_TAX\_AGENCY\_VALIDATION para un ASSIGNED\_TO\_BANK (y no más, 1:1) |
| **Resultado obtenido:**  Código de respuesta http 200 “ok” con el response validationResults en LOCAL\_TAX\_AGENCY\_VALIDATION dado en el body un validationResults en LOCAL\_TAX\_AGENCY\_VALIDATION, es decir, un LOCAL\_TAX\_AGENCY\_VALIDATION para un LOCAL\_TAX\_AGENCY\_VALIDATION (y no más, 1:1) |

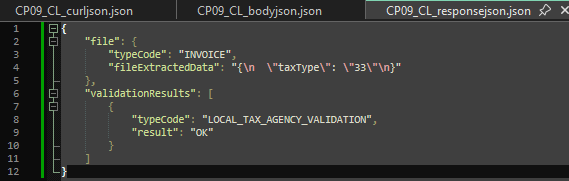
**Archivos relacionados:**

* CP09\_CL\_curljson.json
* CP09\_CL\_bodyjson.json
* CP09\_CL\_responsejson.json





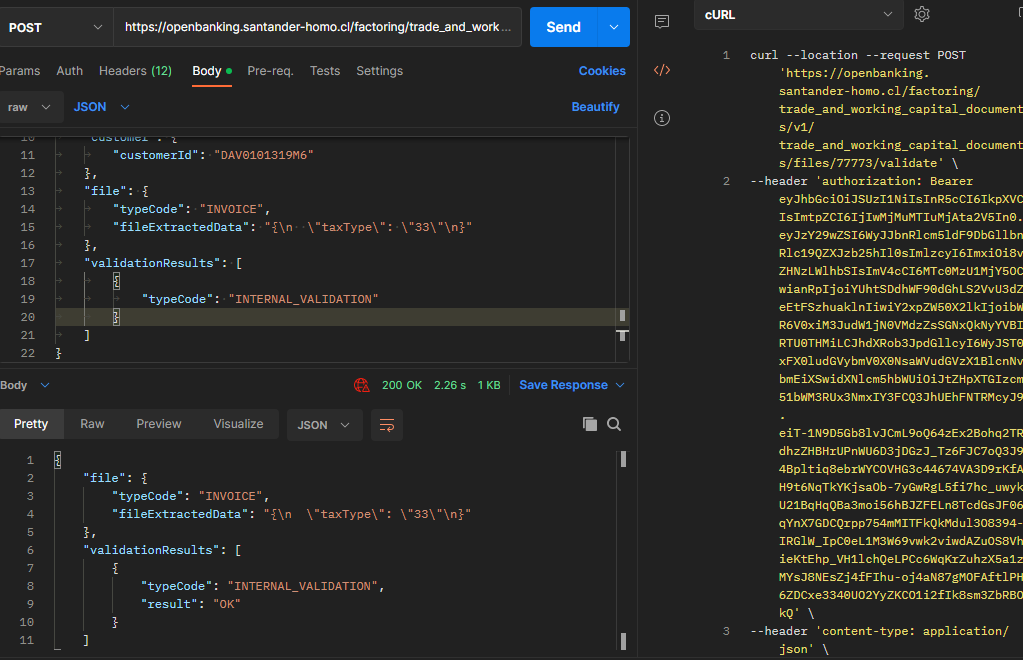


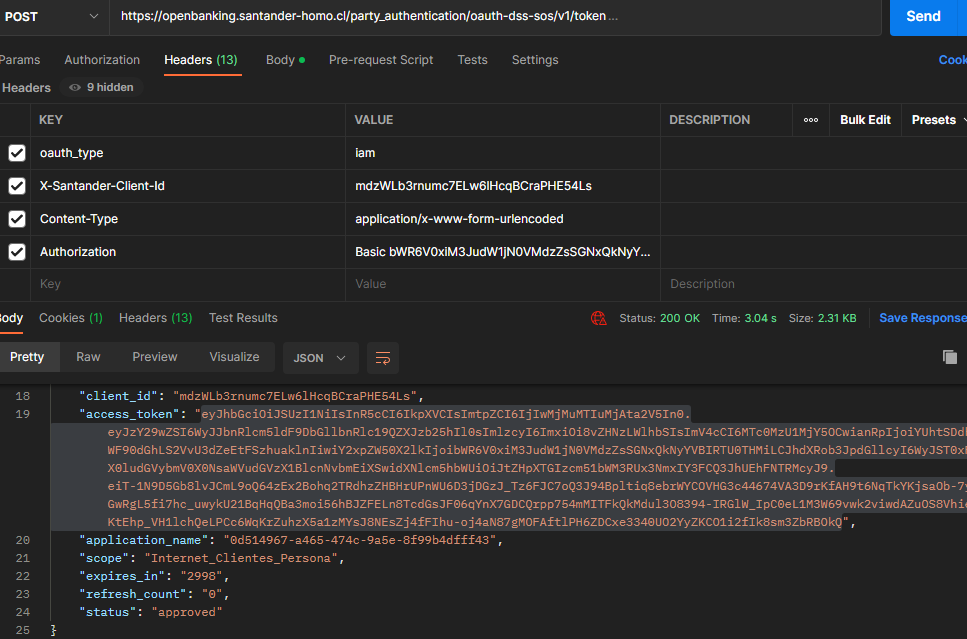


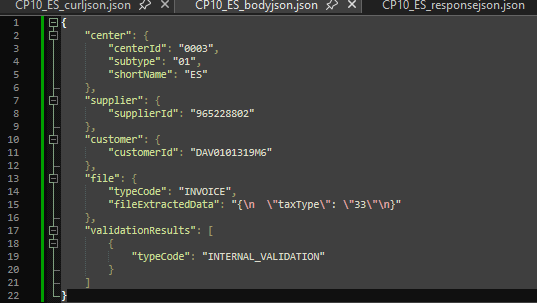
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | CP 10 |  | | ¿Despliegue? | No |   Caso de prueba incorrecto: shortName ES con validationResults en typeCode “INTERNAL\_VALIDATION” (y solo éste) |
| **Descripción:**  A través de postman, haremos una petición correcta, con token válido, “x\_santander\_client\_id” válido, content-type como application-json válido, un bearer token válido y un body correcto cuando validationResults en typeCode “INTERNAL\_VALIDATION” (y solo éste), al endpoint del ms para obtener la información relacionada al contexto shortName ES |
| **Prerrequisitos:**   1. Generar un token válido. 2. Tener el dato “x\_santander\_client\_id” correcto. 3. Tener un body valido. |
| **Pasos:**   1. Generar nuevo token válido. 2. Armar el “request” en postman. 3. Ejecutar. |
| **Resultado esperado:**  Código de respuesta http 200 “ok” con el response validationResults en INTERNAL\_VALIDATION dado en el body un validationResults en INTERNAL\_VALIDATION, es decir, un INTERNAL\_VALIDATION para un INTERNAL\_VALIDATION (y no más, 1:1) |
| **Resultado obtenido:**  Código de respuesta http 200 “ok” con el response validationResults en INTERNAL\_VALIDATION dado en el body un validationResults en INTERNAL\_VALIDATION, es decir, un INTERNAL\_VALIDATION para un INTERNAL\_VALIDATION (y no más, 1:1) |

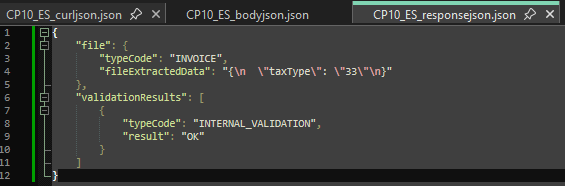
**Archivos relacionados:**

* CP10\_ES\_curljson.json
* CP10\_ES\_bodyjson.json
* CP10\_ES\_responsejson.json





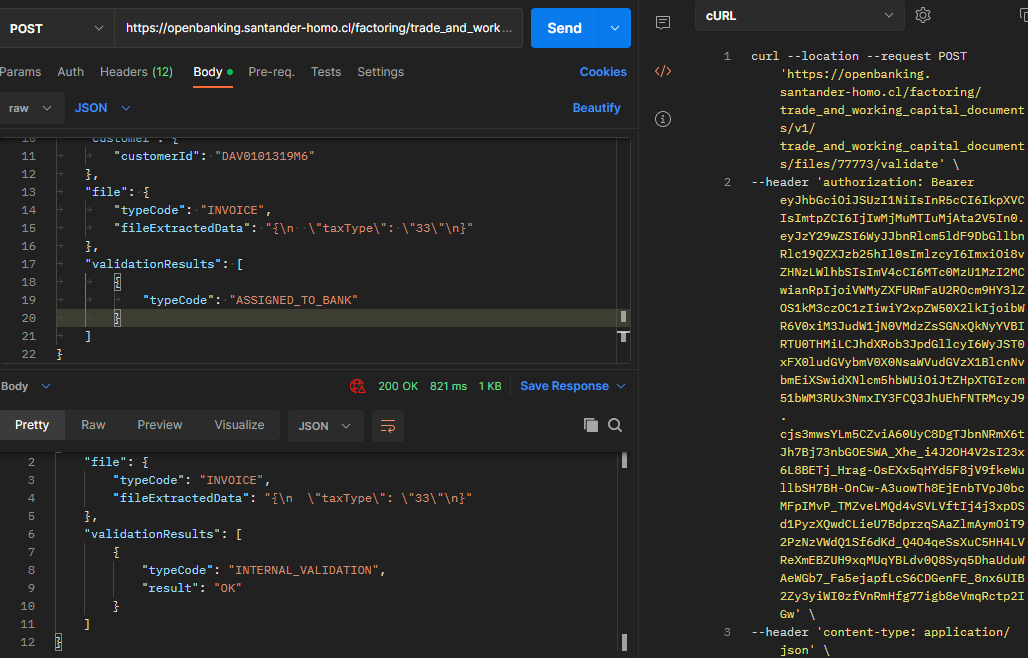


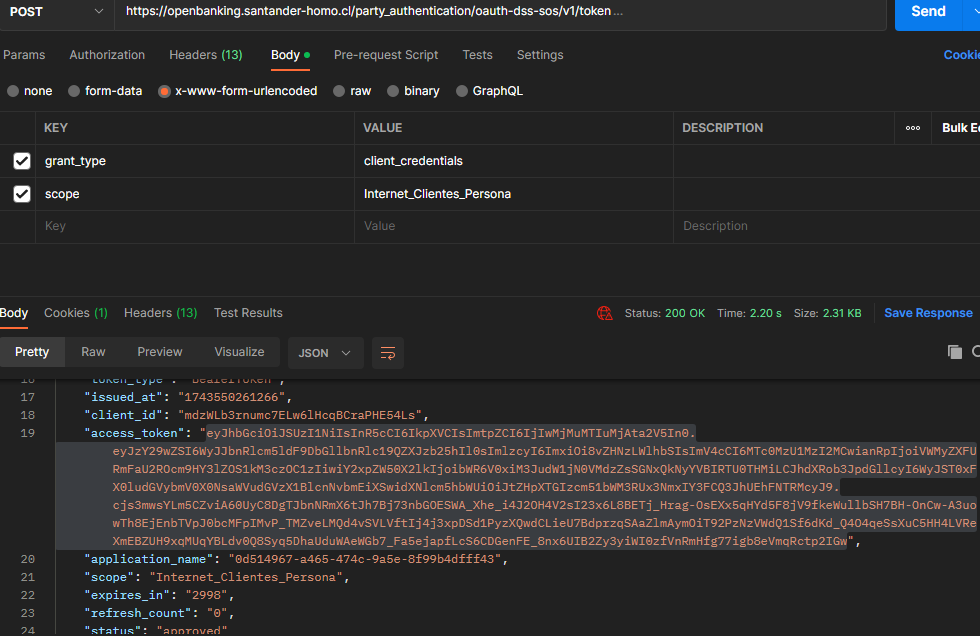


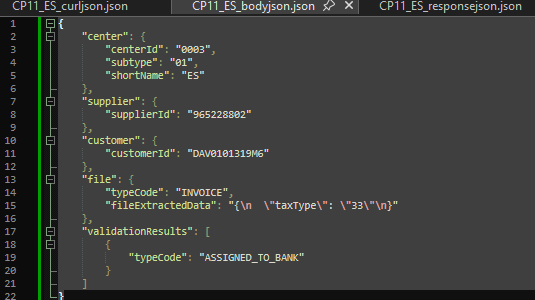
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | CP 11 |  | | ¿Despliegue? | No |   Caso de prueba incorrecto: shortName CL con validationResults en typeCode “ASSIGNED\_TO\_BANK” (y solo éste) |
| **Descripción:**  A través de postman, haremos una petición correcta, con token válido, “x\_santander\_client\_id” válido, content-type como application-json válido, un bearer token válido y un body correcto cuando validationResults en typeCode “ASSIGNED\_TO\_BANK” (y solo éste), al endpoint del ms para obtener la información relacionada al contexto shortName ES |
| **Prerrequisitos:**   1. Generar un token válido. 2. Tener el dato “x\_santander\_client\_id” correcto. 3. Tener un body valido. |
| **Pasos:**   1. Generar nuevo token válido. 2. Armar el “request” en postman. 3. Ejecutar. |
| **Resultado esperado:**  Código de respuesta http 200 “ok” con el response validationResults en INTERNAL\_VALIDATION dado en el body un validationResults en ASSIGNED\_TO\_BANK, es decir, un ASSIGNED\_TO\_BANK para un INTERNAL\_VALIDATION (y no más, 1:1) |
| **Resultado obtenido:**  Código de respuesta http 200 “ok” con el response validationResults en ASSIGNED\_TO\_BANK dado en el body un validationResults en INTERNAL\_VALIDATION es decir, un ASSIGNED\_TO\_BANK para un INTERNAL\_VALIDARION (y no más, 1:1) |

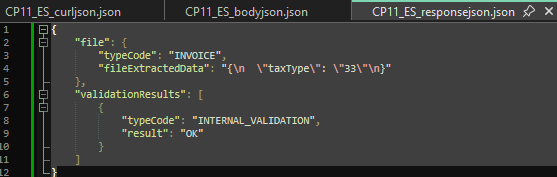
**Archivos relacionados:**

* CP11\_ES\_curljson.json
* CP11\_ES\_bodyjson.json
* CP11\_ES\_responsejson.json





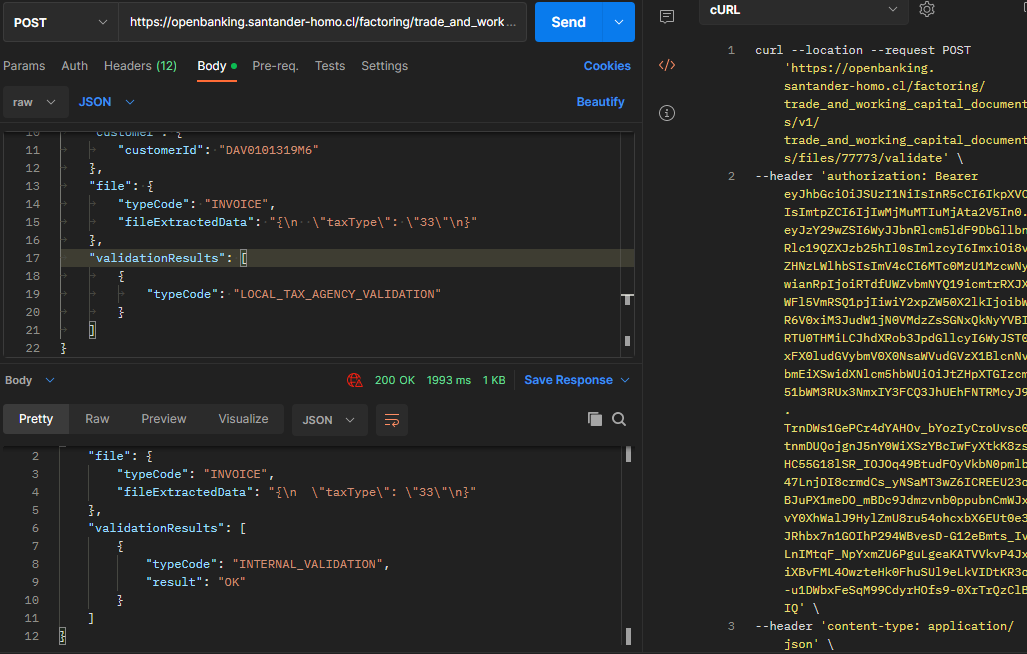


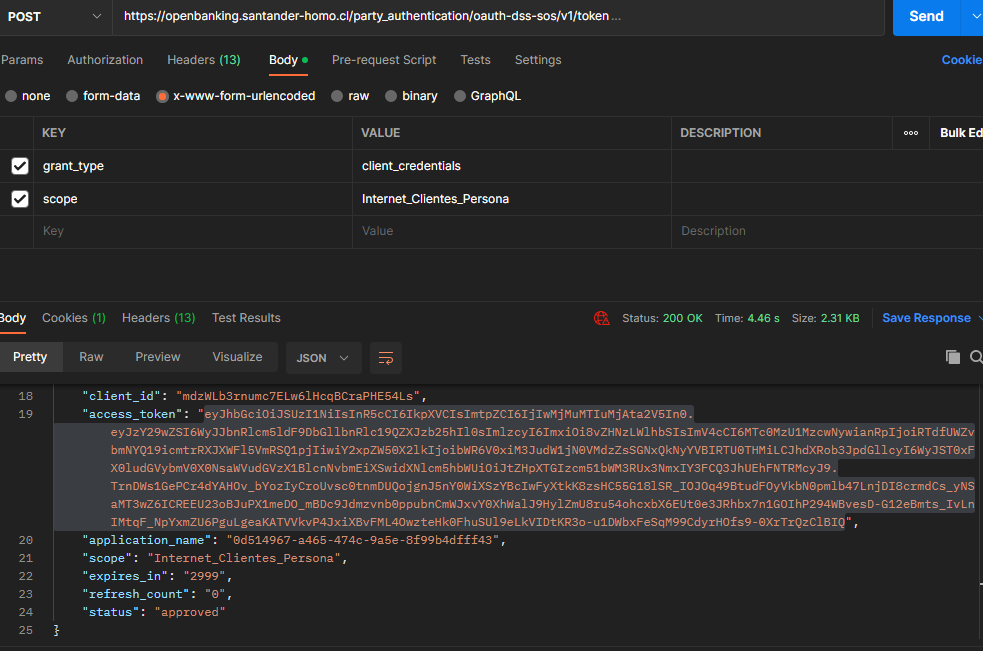


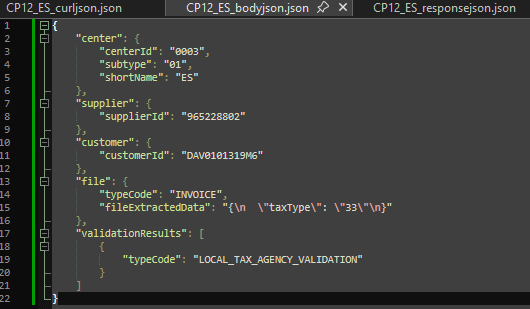
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | CP 12 |  | | ¿Despliegue? | No |   Caso de prueba incorrecto: shortName CL con validationResults en typeCode “LOCAL\_TAX\_AGENCY\_VALIDATION” (y solo éste) |
| **Descripción:**  A través de postman, haremos una petición correcta, con token válido, “x\_santander\_client\_id” válido, content-type como application-json válido, un bearer token válido y un body correcto cuando validationResults en typeCode “LOCAL\_TAX\_AGENCY\_VALIDATION” (y solo éste), al endpoint del ms para obtener la información relacionada al contexto shortName ES |
| **Prerrequisitos:**   1. Generar un token válido. 2. Tener el dato “x\_santander\_client\_id” correcto. 3. Tener un body valido. |
| **Pasos:**   1. Generar nuevo token válido. 2. Armar el “request” en postman. 3. Ejecutar. |
| **Resultado esperado:**  Código de respuesta http 200 “ok” con el response validationResults en LOCAL\_TAX\_AGENCY\_VALIDATION dado en el body un validationResults en INTERNAL\_VALIDATION, es decir, un LOCAL\_TAX\_AGENCY\_VALIDATION para un INTERNAL\_VALIDATION (y no más, 1:1) |
| **Resultado obtenido:**  Código de respuesta http 200 “ok” con el response validationResults en LOCAL\_TAX\_AGENCY\_VALIDATION dado en el body un validationResults en INTERNAL\_VALIDATION, es decir, un LOCAL\_TAX\_AGENCY\_VALIDATION para un INTERNAL\_VALIDATION (y no más, 1:1) |

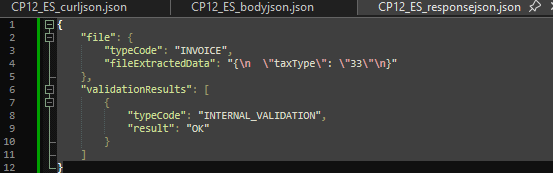
**Archivos relacionados:**

* CP12\_ES\_curljson.json
* CP12\_ES\_bodyjson.json
* CP12\_ES\_responsejson.json





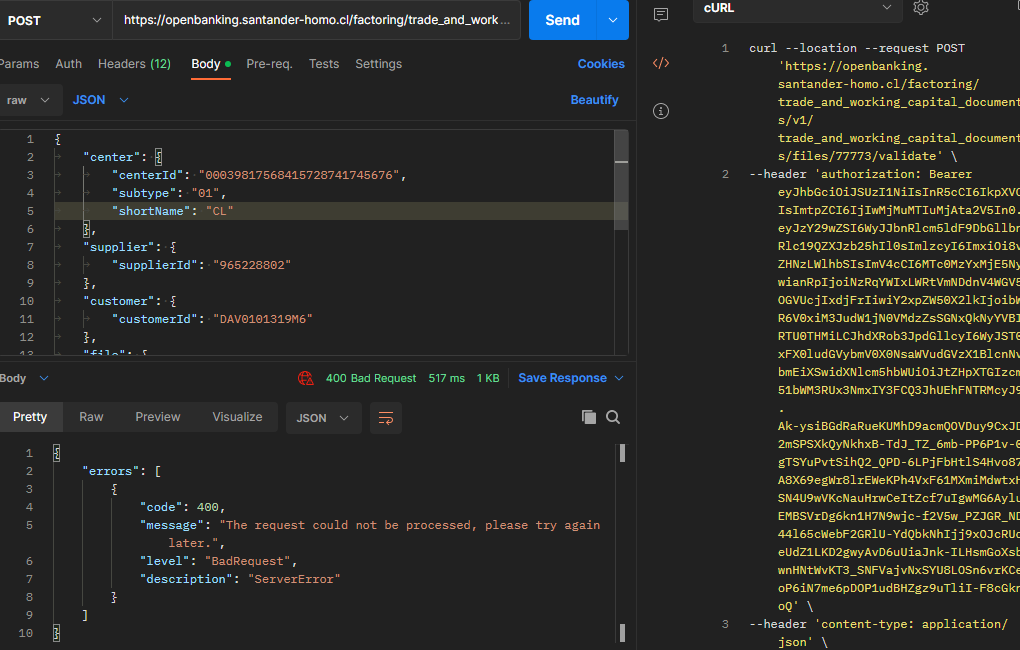


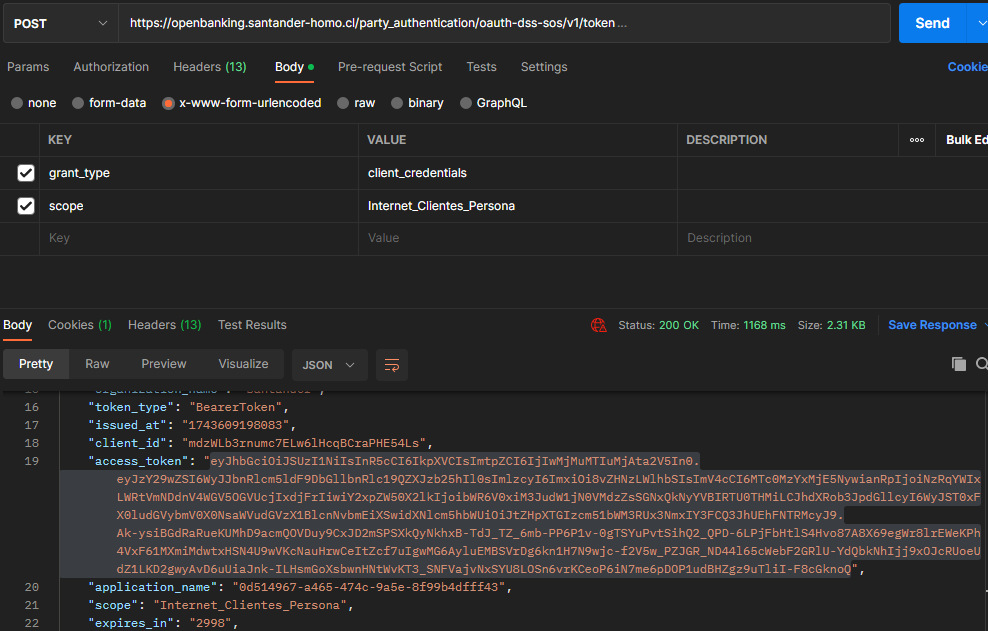


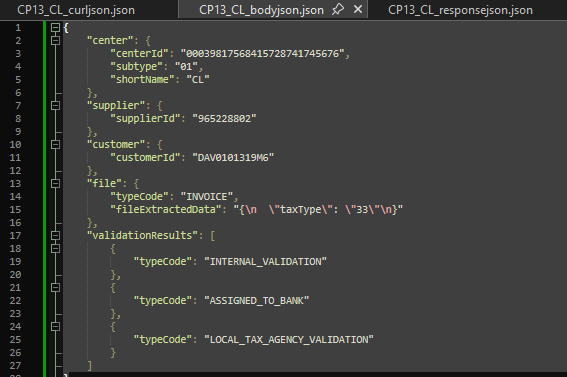
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | CP 13 |  | | ¿Despliegue? | No |   Caso de prueba correcto: shortName CL centerId inválido |
| **Descripción:**  A través de postman, haremos una petición correcta, con token válido, “x\_santander\_client\_id” válido, content-type como application-json válido, un bearer token válido y un centerId inválido en el body, al endpoint del ms para obtener la información relacionada al contexto shortName CL |
| **Prerrequisitos:**   1. Generar un token válido. 2. Tener el dato “x\_santander\_client\_id” correcto. 3. Tener un body valido. |
| **Pasos:**   1. Generar nuevo token válido. 2. Armar el “request” en postman. 3. Ejecutar. |
| **Resultado esperado:**  Código de respuesta http 400 “bad request” con la data consistente según el “id” de cuenta para ms trading and working capital documents |
| **Resultado obtenido:**  Código de respuesta http 400 “bad request” con la data consistente según el “id” de cuenta para ms trading and working capital documents |

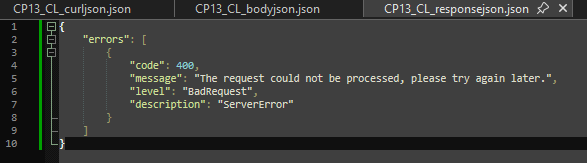
**Archivos relacionados:**

* CP13\_CL\_curljson.json
* CP13\_CL\_bodyjson.json
* CP13\_CL\_responsejson.json





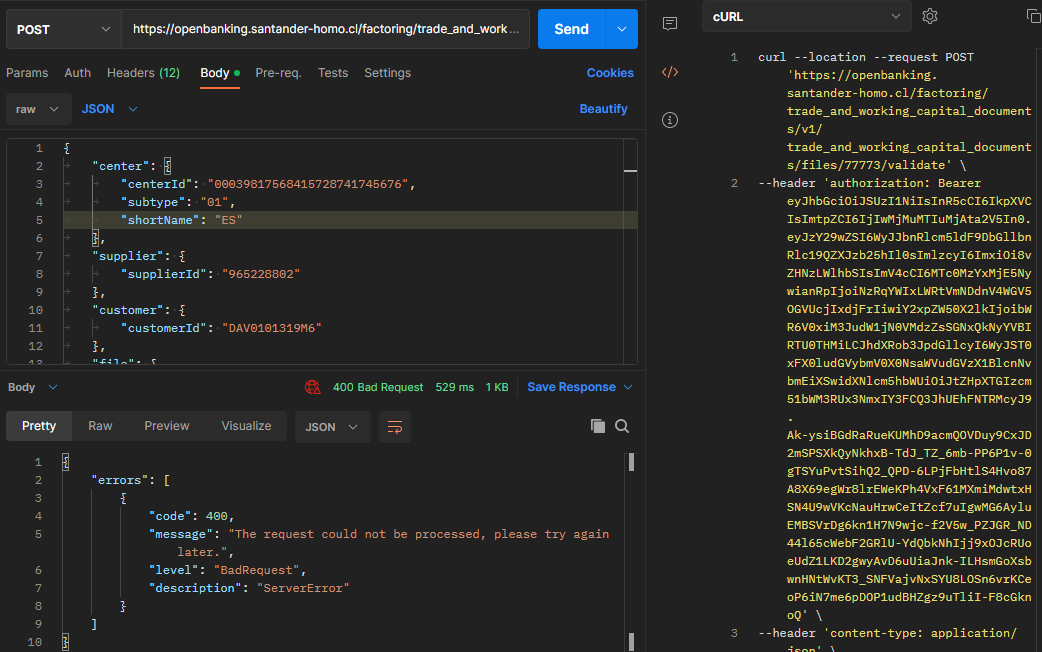


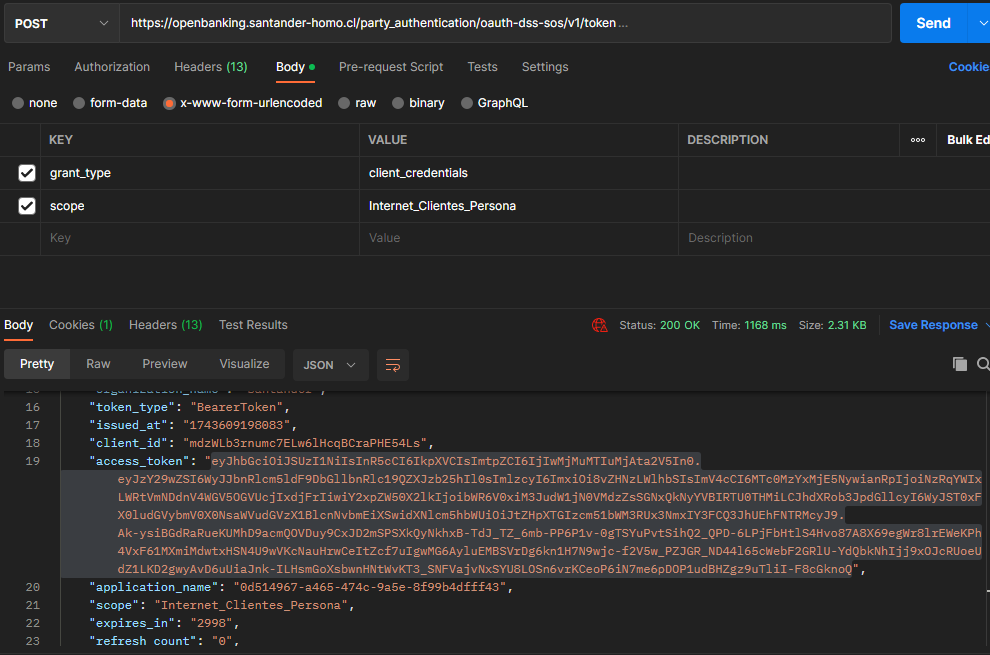


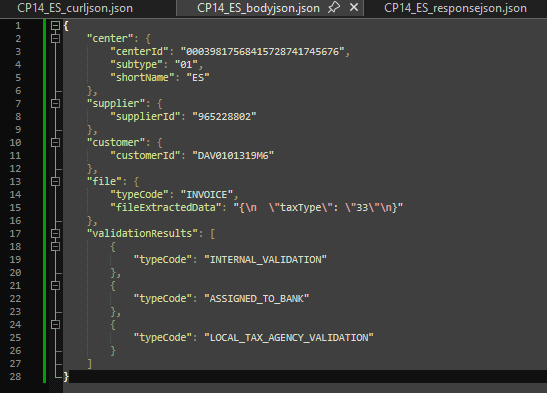
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | CP 14 |  | | ¿Despliegue? | No |   Caso de prueba correcto: shortName ES centerId inválido |
| **Descripción:**  A través de postman, haremos una petición correcta, con token válido, “x\_santander\_client\_id” válido, content-type como application-json válido, un bearer token válido y un centerId inválido en el body, al endpoint del ms para obtener la información relacionada al contexto shortName ES |
| **Prerrequisitos:**   1. Generar un token válido. 2. Tener el dato “x\_santander\_client\_id” correcto. 3. Tener un body valido. |
| **Pasos:**   1. Generar nuevo token válido. 2. Armar el “request” en postman. 3. Ejecutar. |
| **Resultado esperado:**  Código de respuesta http 400 “bad request” con la data consistente según el “id” de cuenta para ms trading and working capital documents |
| **Resultado obtenido:**  Código de respuesta http 400 “bad request” con la data consistente según el “id” de cuenta para ms trading and working capital documents |

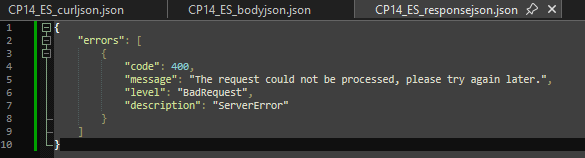
**Archivos relacionados:**

* CP14\_ES\_curljson.json
* CP14\_ES\_bodyjson.json
* CP14\_ES\_responsejson.json





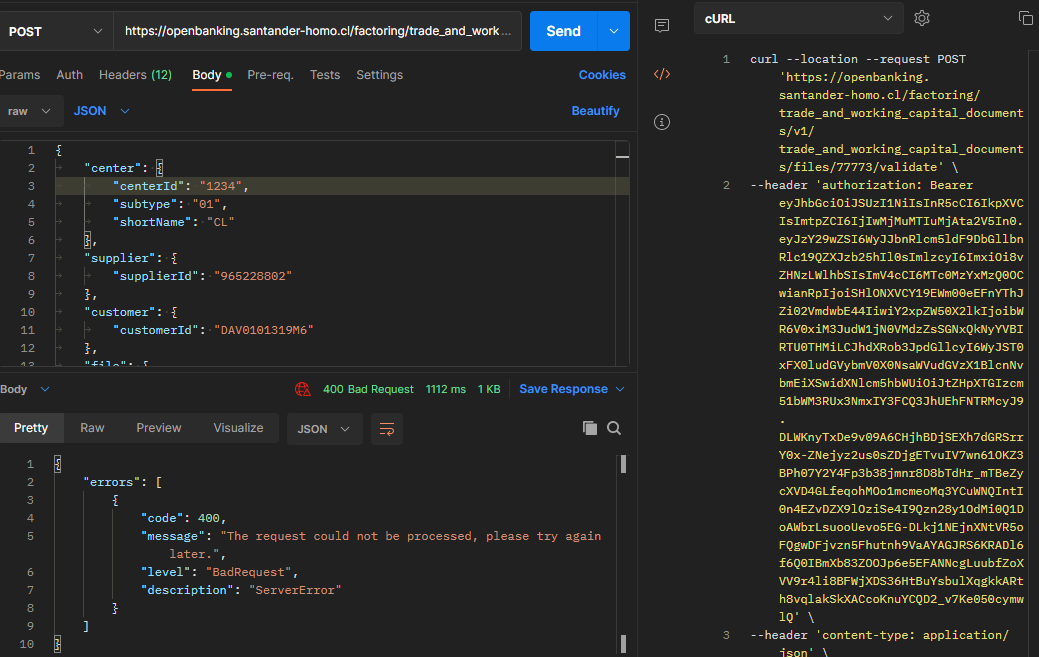


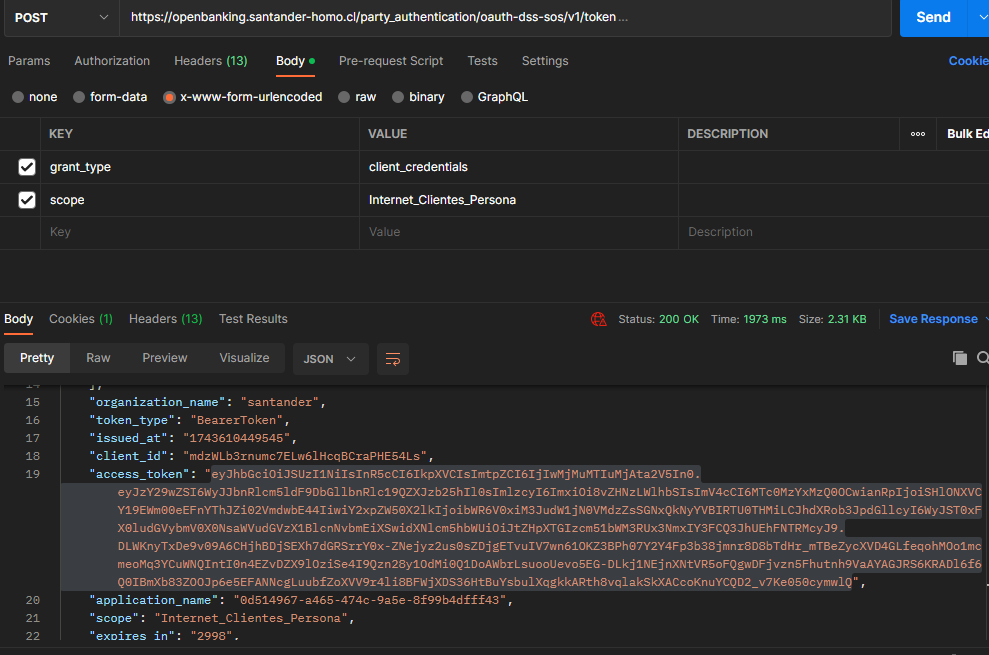


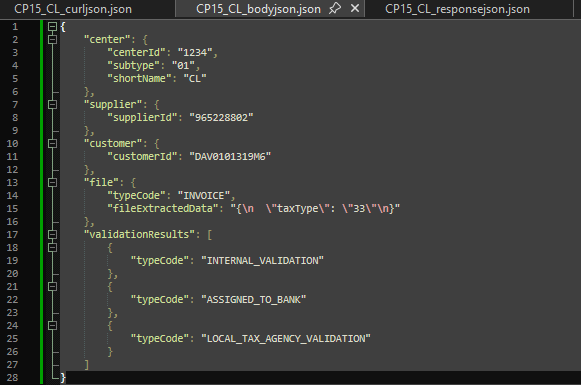
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | CP 15 |  | | ¿Despliegue? | No |   Caso de prueba correcto: shortName CL centerId inválido |
| **Descripción:**  A través de postman, haremos una petición correcta, con token válido, “x\_santander\_client\_id” válido, content-type como application-json válido, un bearer token válido y un centerId inválido en el body, al endpoint del ms para obtener la información relacionada al contexto shortName CL |
| **Prerrequisitos:**   1. Generar un token válido. 2. Tener el dato “x\_santander\_client\_id” correcto. 3. Tener un body valido. |
| **Pasos:**   1. Generar nuevo token válido. 2. Armar el “request” en postman. 3. Ejecutar. |
| **Resultado esperado:**  Código de respuesta http 400 “bad request” con la data consistente según el “id” de cuenta para ms trading and working capital documents |
| **Resultado obtenido:**  Código de respuesta http 400 “bad request” con la data consistente según el “id” de cuenta para ms trading and working capital documents |

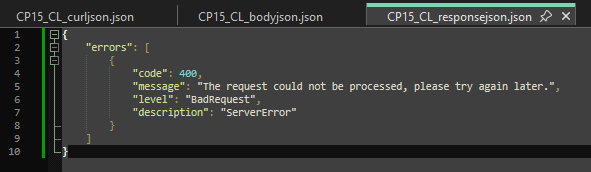
**Archivos relacionados:**

* CP15\_CL\_curljson.json
* CP15\_CL\_bodyjson.json
* CP15\_CL\_responsejson.json





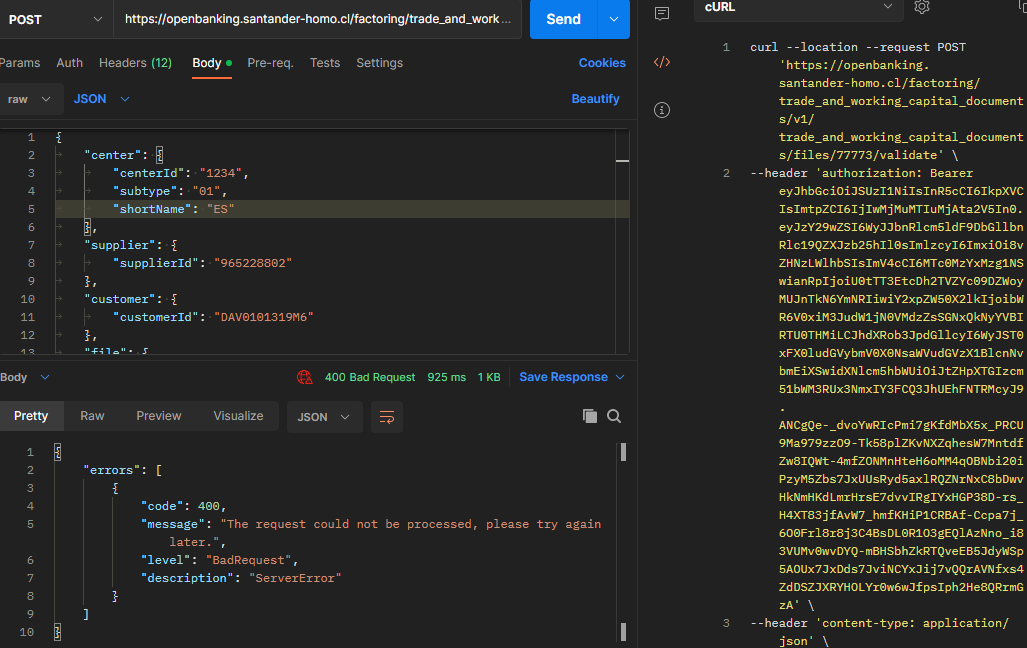


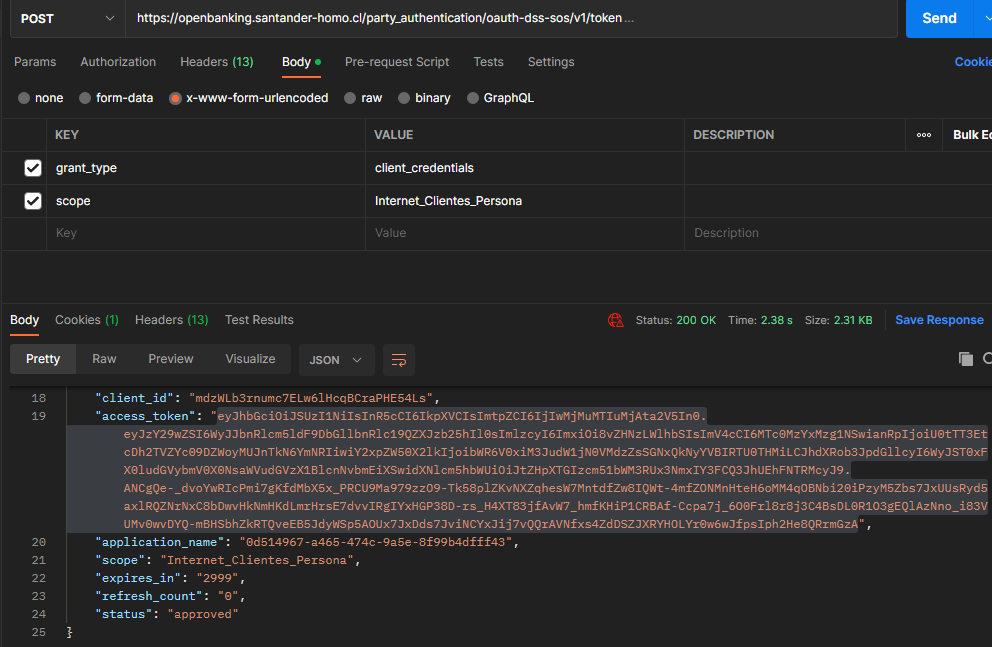


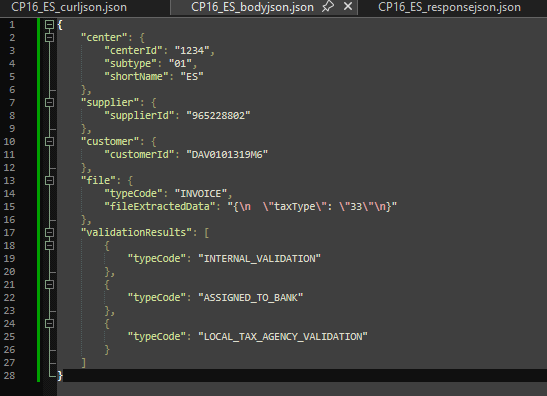
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | CP 16 |  | | ¿Despliegue? | No |   Caso de prueba correcto: shortName ES centerId inválido |
| **Descripción:**  A través de postman, haremos una petición correcta, con token válido, “x\_santander\_client\_id” válido, content-type como application-json válido, un bearer token válido y un centerId inválido en el body, al endpoint del ms para obtener la información relacionada al contexto shortName ES |
| **Prerrequisitos:**   1. Generar un token válido. 2. Tener el dato “x\_santander\_client\_id” correcto. 3. Tener un body valido. |
| **Pasos:**   1. Generar nuevo token válido. 2. Armar el “request” en postman. 3. Ejecutar. |
| **Resultado esperado:**  Código de respuesta http 400 “bad request” con la data consistente según el “id” de cuenta para ms trading and working capital documents |
| **Resultado obtenido:**  Código de respuesta http 400 “bad request” con la data consistente según el “id” de cuenta para ms trading and working capital documents |

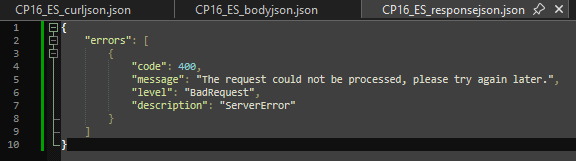
**Archivos relacionados:**

* CP16\_ES\_curljson.json
* CP16\_ES\_bodyjson.json
* CP16\_ES\_responsejson.json





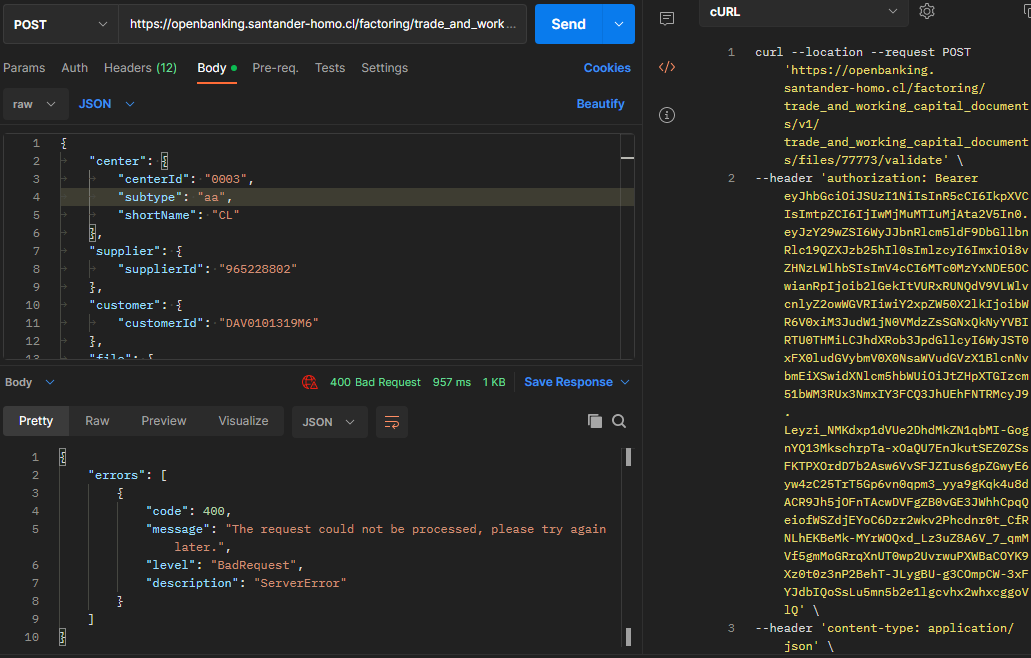


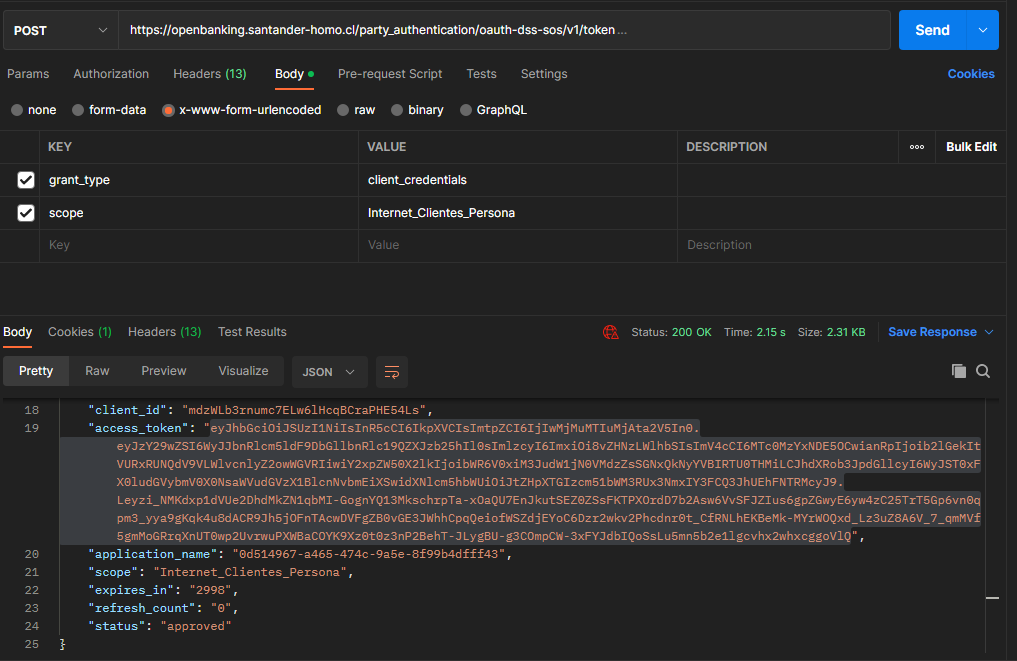


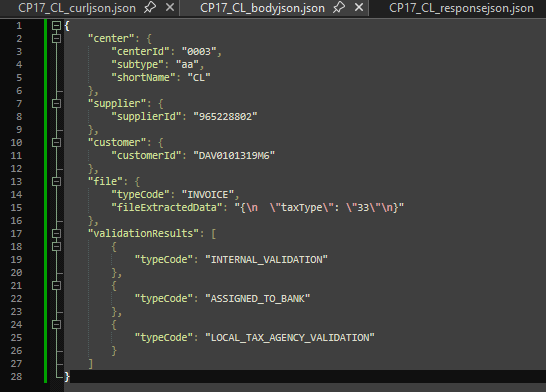
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | CP 17 |  | | ¿Despliegue? | No |   Caso de prueba correcto: shortName CL subType inválido |
| **Descripción:**  A través de postman, haremos una petición correcta, con token válido, “x\_santander\_client\_id” válido, content-type como application-json válido, un bearer token válido y un subType inválido en el body, al endpoint del ms para obtener la información relacionada al contexto shortName CL |
| **Prerrequisitos:**   1. Generar un token válido. 2. Tener el dato “x\_santander\_client\_id” correcto. 3. Tener un body valido. |
| **Pasos:**   1. Generar nuevo token válido. 2. Armar el “request” en postman. 3. Ejecutar. |
| **Resultado esperado:**  Código de respuesta http 400 “bad request” con la data consistente según el “id” de cuenta para ms trading and working capital documents |
| **Resultado obtenido:**  Código de respuesta http 400 “bad request” con la data consistente según el “id” de cuenta para ms trading and working capital documents |

**Archivos relacionados:**

* CP17\_CL\_curljson.json
* CP17\_CL\_bodyjson.json
* CP17\_CL\_responsejson.json



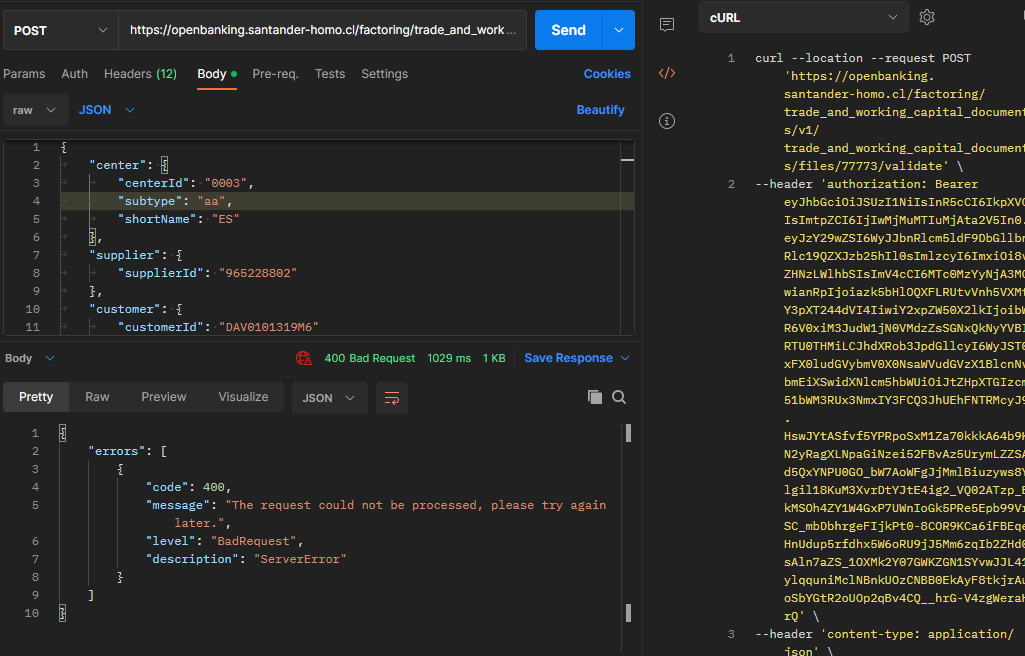


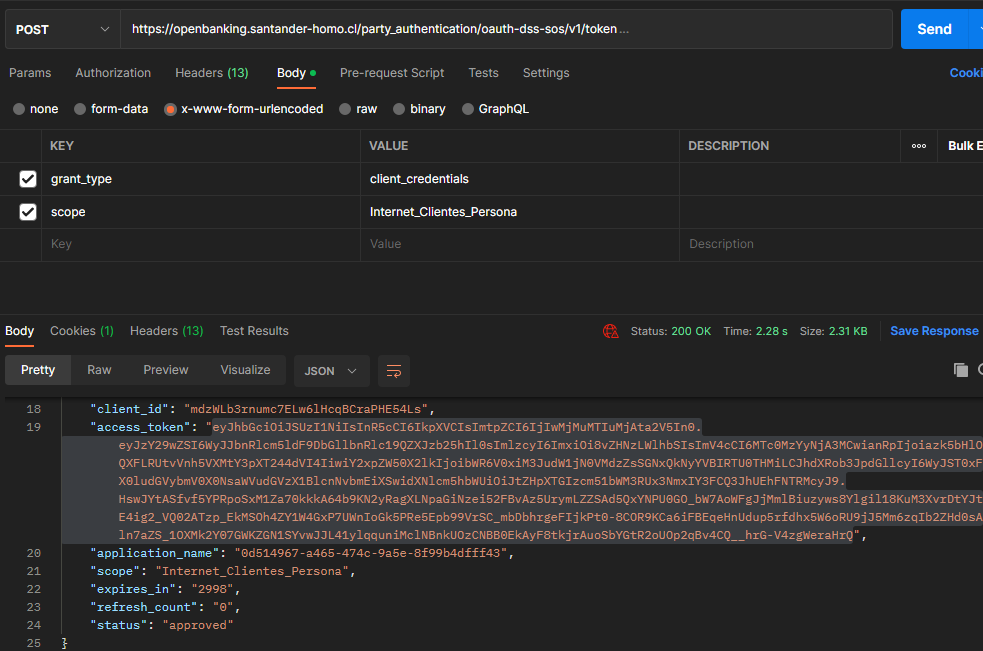


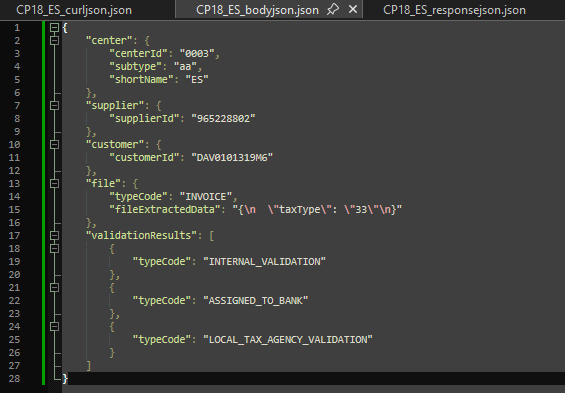
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | CP 18 |  | | ¿Despliegue? | No |   Caso de prueba correcto: shortName ES subType inválido |
| **Descripción:**  A través de postman, haremos una petición correcta, con token válido, “x\_santander\_client\_id” válido, content-type como application-json válido, un bearer token válido y un subType inválido en el body, al endpoint del ms para obtener la información relacionada al contexto shortName ES |
| **Prerrequisitos:**   1. Generar un token válido. 2. Tener el dato “x\_santander\_client\_id” correcto. 3. Tener un body valido. |
| **Pasos:**   1. Generar nuevo token válido. 2. Armar el “request” en postman. 3. Ejecutar. |
| **Resultado esperado:**  Código de respuesta http 400 “bad request” con la data consistente según el “id” de cuenta para ms trading and working capital documents |
| **Resultado obtenido:**  Código de respuesta http 400 “bad request” con la data consistente según el “id” de cuenta para ms trading and working capital documents |

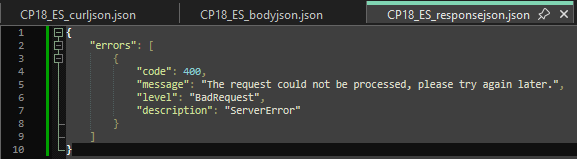
**Archivos relacionados:**

* CP18\_ES\_curljson.json
* CP18\_ES\_bodyjson.json
* CP18\_ES\_responsejson.json





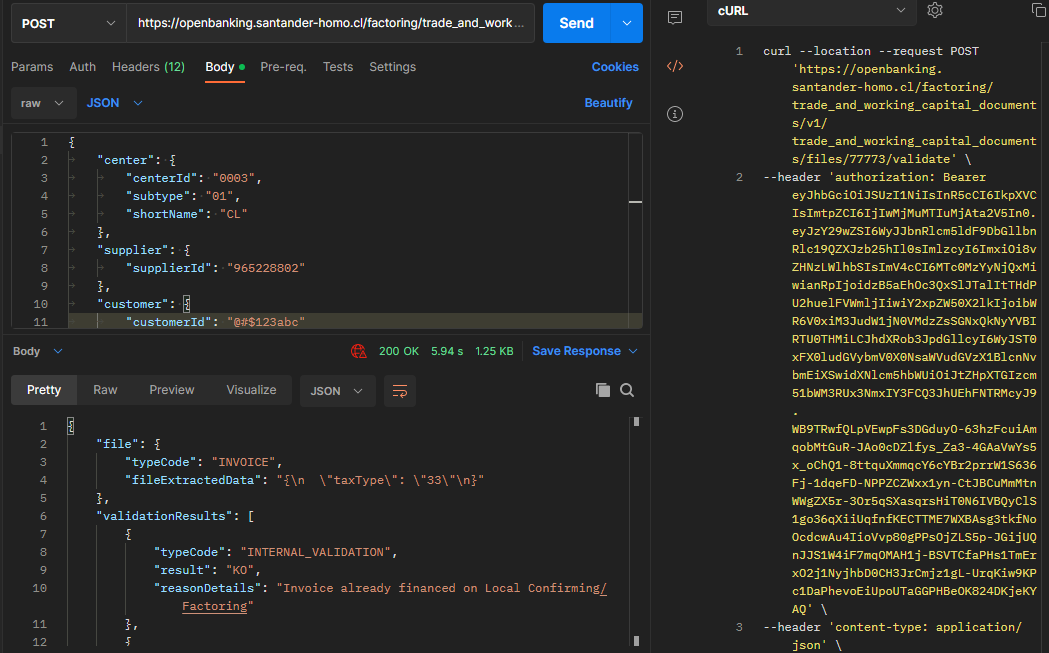


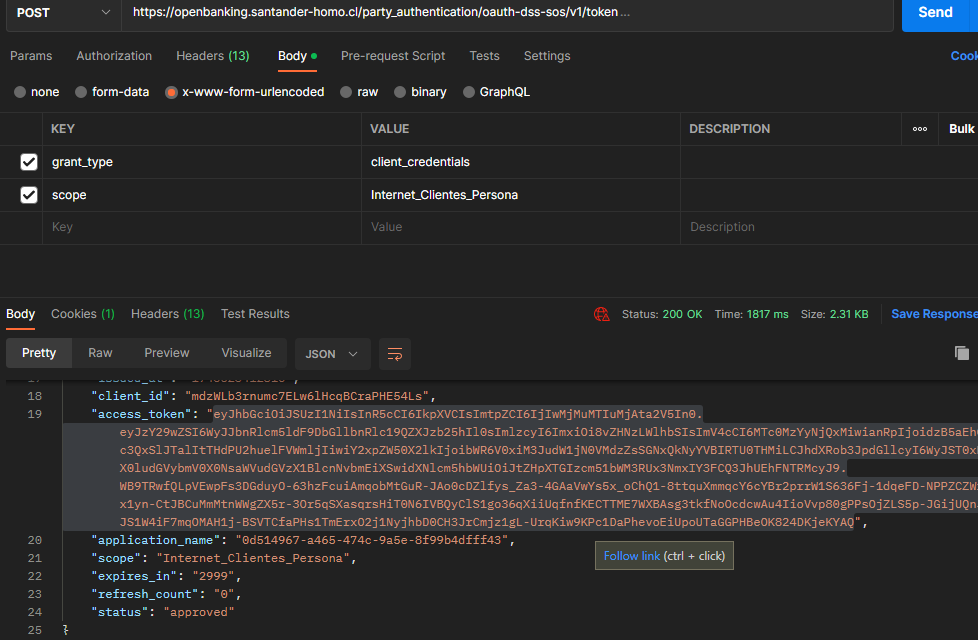


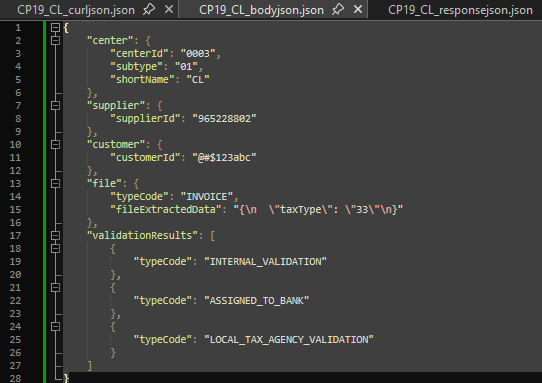
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | CP 19 |  | | ¿Despliegue? | No |   Caso de prueba correcto: shortName CL customerId inválido |
| **Descripción:**  A través de postman, haremos una petición correcta, con token válido, “x\_santander\_client\_id” válido, content-type como application-json válido, un bearer token válido y un customerId inválido en el body, al endpoint del ms para obtener la información relacionada al contexto shortName CL |
| **Prerrequisitos:**   1. Generar un token válido. 2. Tener el dato “x\_santander\_client\_id” correcto. 3. Tener un body valido. |
| **Pasos:**   1. Generar nuevo token válido. 2. Armar el “request” en postman. 3. Ejecutar. |
| **Resultado esperado:**  Código de respuesta http 200 “ok con la data consistente según el “id” de cuenta para ms trading and working capital documents |
| **Resultado obtenido:**  Código de respuesta http 200 “ok" con la data consistente según el “id” de cuenta para ms trading and working capital documents |

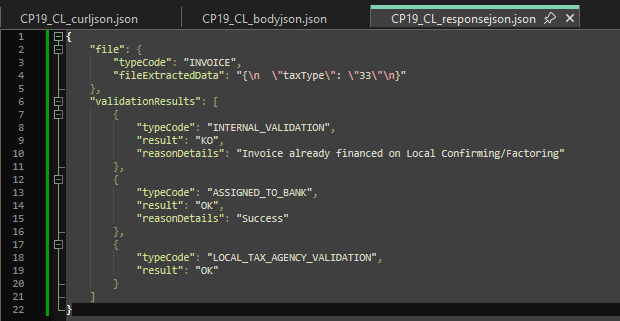
**Archivos relacionados:**

* CP19\_CL\_curljson.json
* CP19\_CL\_bodyjson.json
* CP19\_CL\_responsejson.json









|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | CP 20 |  | | ¿Despliegue? | No |   Caso de prueba correcto: shortName ES customerId inválido |
| **Descripción:**  A través de postman, haremos una petición correcta, con token válido, “x\_santander\_client\_id” válido, content-type como application-json válido, un bearer token válido y un customerId inválido en el body, al endpoint del ms para obtener la información relacionada al contexto shortName ES |
| **Prerrequisitos:**   1. Generar un token válido. 2. Tener el dato “x\_santander\_client\_id” correcto. 3. Tener un body valido. |
| **Pasos:**   1. Generar nuevo token válido. 2. Armar el “request” en postman. 3. Ejecutar. |
| **Resultado esperado:**  Código de respuesta http 200 “ok con la data consistente según el “id” de cuenta para ms trading and working capital documents |
| **Resultado obtenido:**  Código de respuesta http 200 “ok" con la data consistente según el “id” de cuenta para ms trading and working capital documents |

**Archivos relacionados:**

* CP20\_ES\_curljson.json
* CP20\_ES\_bodyjson.json
* CP20\_ES\_responsejson.json

