Seamless.Insure

Manual name:

Developer guide

https://seamless.insure/

OpenAPI

This page provides the list of Seamless APIs and best practices for consuming them.

Best practices for consuming JSON APIs

This section describes best practices for consuming Seamless APIs that support the JSON:API specification. The list of JSON:API APIs are given below:

- Agency Configuration
- Compliance Service
- Document Service
- Finance Service
- Motor Registry
- Policy Management
- Pricing Engine
- Quotation Service
- Tenant Management
- User Management

All requests must be authenticated

Every request to Seamless APIs must be authenticated. For details of the authentication process, please see Authentication

Use sparse fieldset selection

Clients should request only fields that they need from an endpoint. This practice is especially true for resource types that have a lot of fields or have fields with large data such as `quotes` in the Quotation Service and `policies` in the Policy Management. For example, the following request retrieves quotes with a sparse fieldset selection specified for the `quotes` resource. It also specifies the page size, page number, and a sort order.

```
GET {{base-url-sales}}/api/v1/quotes?fields[quotes]=reference,description,period,status&page[size]=10&page[number]=1&sort=-createdDate
Authorization: {{access-token}}
Accept: application/vnd.api+json
```

If the client does not specify a sparse fieldset selection for a given resource type, the API may not include fields with large data for that resource type. This behaviour adheres to the specification.

- If a client does not specify the set of fields for a given resource type, the server **MAY** send all fields, a subset of fields, or no fields for that resource type.

For more details of sparse fieldset specification, please refer to https://jsonapi.org/format/1.1/#fetching-sparse-fieldsets

Use paging

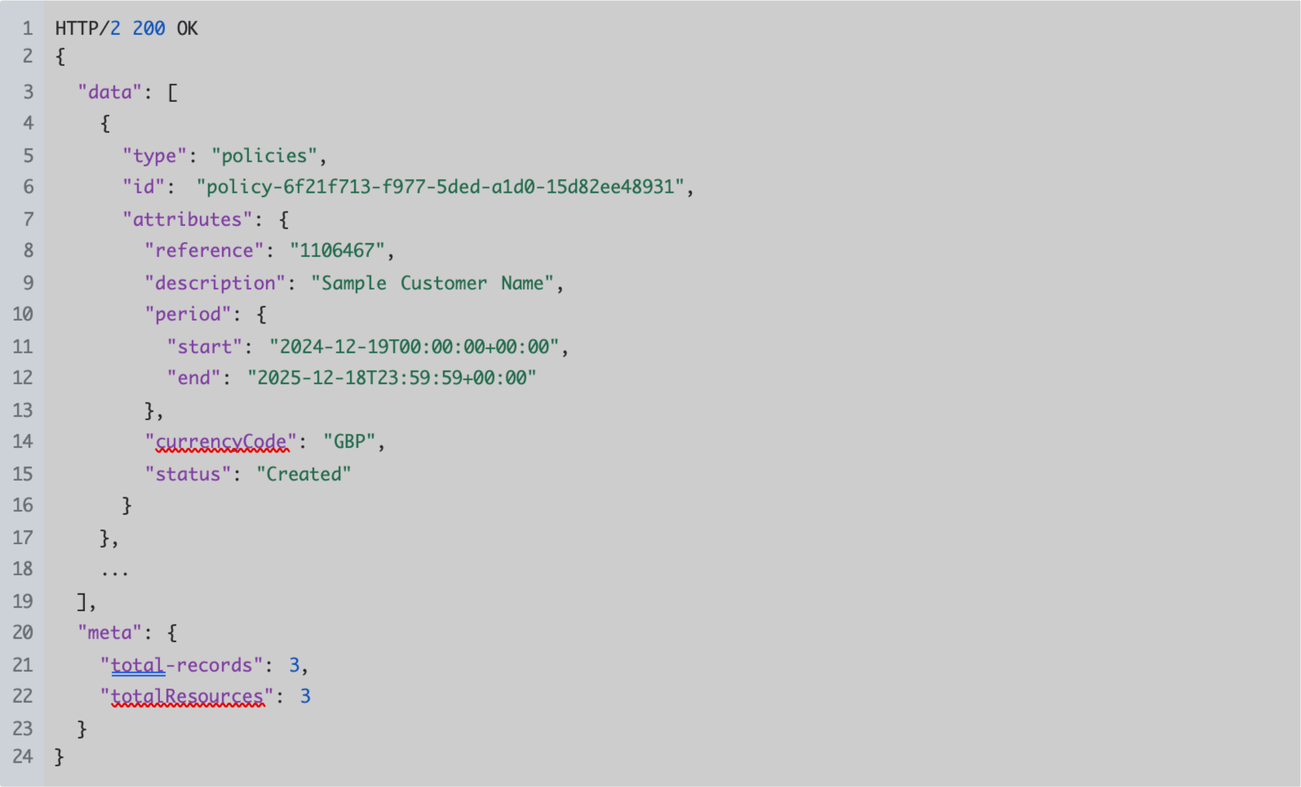
Requests for multiple resource instances should include a page size. If a page size is not specified, the default page size of 10 is used.

Do not use obsolete features

This section describes obsolete features that should not be used and may not be available in future updates to the APIs.

total-records

Responses for a multiple-resources request with pagination may include a `total-records` item in the `meta` property, which is the total of resource items match the query parameters. `total-records` is obsolete and may not be available in future updates of the APIs. Please use the new `totalResources` instead. For example, the following response include both `total-records` and `totalResources` , but again, `totalResources` should be used instead.



```
HTTP/2 200 OK
{
  "data": [
    {
      "type": "policies",
      "id":  "policy-6f21f713-f977-5ded-a1d0-15d82ee48931",
      "attributes": {
        "reference": "1106467",
        "description": "Sample  Customer  Name",
        "period": {
          "start": "2024-12-19T00:00:00+00:00",
          "end": "2025-12-18T23:59:59+00:00"
        },
        "currencyCode": "GBP",
        "status": "Created"
      }
    },
    ...
  ],
  "meta": {
    "total-records": 3,
    "totalResources": 3
  }
}
```

Filter syntax

JSON:API specification v1.1 doesn’t describe a standardized filter syntax. Therefore, the filter syntax varies and depends on the underlying technologies used to develop the APIs. For microservices in the list at the beginning of the Best practices for consuming JSON APIs section, the filter syntax is described in detailed at https://www.jsonapi.net/usage/reading/filtering.html. Please be informed of the following guidelines regarding to the filter syntax:

1. Legacy filters, e.g. `?filter[lastName]=eq:Smith` are still supported, but they may be removed in a future update of the services. Any new code should use the new filter syntax, and existing code using legacy filters should be updated to use the new filters.

2. New filters MUST be prefixed with expr: , for example, ?filter=expr:equals(lastName,'Smith') .

List of Seamless APIs

Microservice	Domain	OpenAPI Documentation
Agency Configuration	Seamless Control	https://api.seamless.insure/configuration/swagger/index.html
Campaign Service	Seamless Service	https://api.seamless.insure/campaign/documentation/
Claims Service	Seamless Claims	https://api.seamless.insure/claim- api/documentation/
Compliance Service	Seamless Compliance	https://api.seamless.insure/compliance/swagger/index.html
Contact Center	Seamless Service	https://api.seamless.insure/contact-center/documentation/#/
Customer Service	Seamless Service	https://api.seamless.insure/customer/documentation/#/
Document Service	Seamless Documents	https://api.seamless.insure/documents/swagger/index.html
Finance Service	Seamless Finance	https://api.seamless.insure/finance/swagger/index.html
Integration	Seamless Service	https://api.seamless.insure/integration/documentation/
Investment	Seamless Service	https://api.seamless.insure/investment/documentation/#/
Loyalty Service	Seamless Service	https://api.seamless.insure/loyalty/documentation/
Mail Service	Seamless Service	https://api.seamless.insure/mail/documentation/#/
Motor Registry	Seamless Plugins	https://api.seamless.insure/motorregistry/swagger/index.html
Notification Service	Seamless Service	https://api.seamless.insure/notification/documentation/#/
Opportunity	Seamless Service	https://api.seamless.insure/opportunity/documentation/#/
Policy Management	Seamless UW	https://api.seamless.insure/underwriting/swagger/index.html
Pricing Engine	Seamless Pricing	https://api.seamless.insure/pricing/swagger/index.html
Process Engine Service	Seamless Flow	https://api.seamless.insure/process-api/documentation/
Quotation Service	Seamless UW	https://api.seamless.insure/sales/swagger/index.html
Sale workflow service	Seamless Grow	
Scoring	Seamless UW	https://api.seamless.insure/scoring- api/documentation/
Search Service	Seamless Service	https://api.seamless.insure/search/documentation/

Task Service	Seamless Service	https://api.seamless.insure/task/documentation/
Tenant Management	Seamless Control	https://api.seamless.insure/tenancy/swagger/index.html
Time Tracking	Seamless Service	https://api.seamless.insure/timetracking/documentation/#/
User Management	Seamless Control	https://api.seamless.insure/users/swagger/index.html

Authentication

- Obtain an access token:

```
curl  https://AUTH_BASE_URL/oauth2/token  \
-X POST \
-H "Content-Type:  application/x-www-form-urlencoded"  \
-d   "grant_type=client_credentials&client_id={CLIENT_ID}&client_secret={CLIENT_SECRET}"
```

You can get the `AUTH_BASE_URL` from the Seamless team.

The `CLIENT_ID` and `CLIENT_SECRET` can be obtained from the Applications page on the Seamless website.

- Sample response:

```
{"access_token":"...","expires_in":3600,"token_type":"Bearer"}
```

- Make API requests using the obtained access token:

```
curl  "https://external-api.staging.contemisaasdev.com/underwriting/api/v1/policies"  \
-H  'Content-Type:application/vnd.api+json'  \
-H  'Accept:application/vnd.api+json'  \
-H  'Authorization:  {ACCESS_TOKEN}'
```

Customer

API documentation: Swagger UI

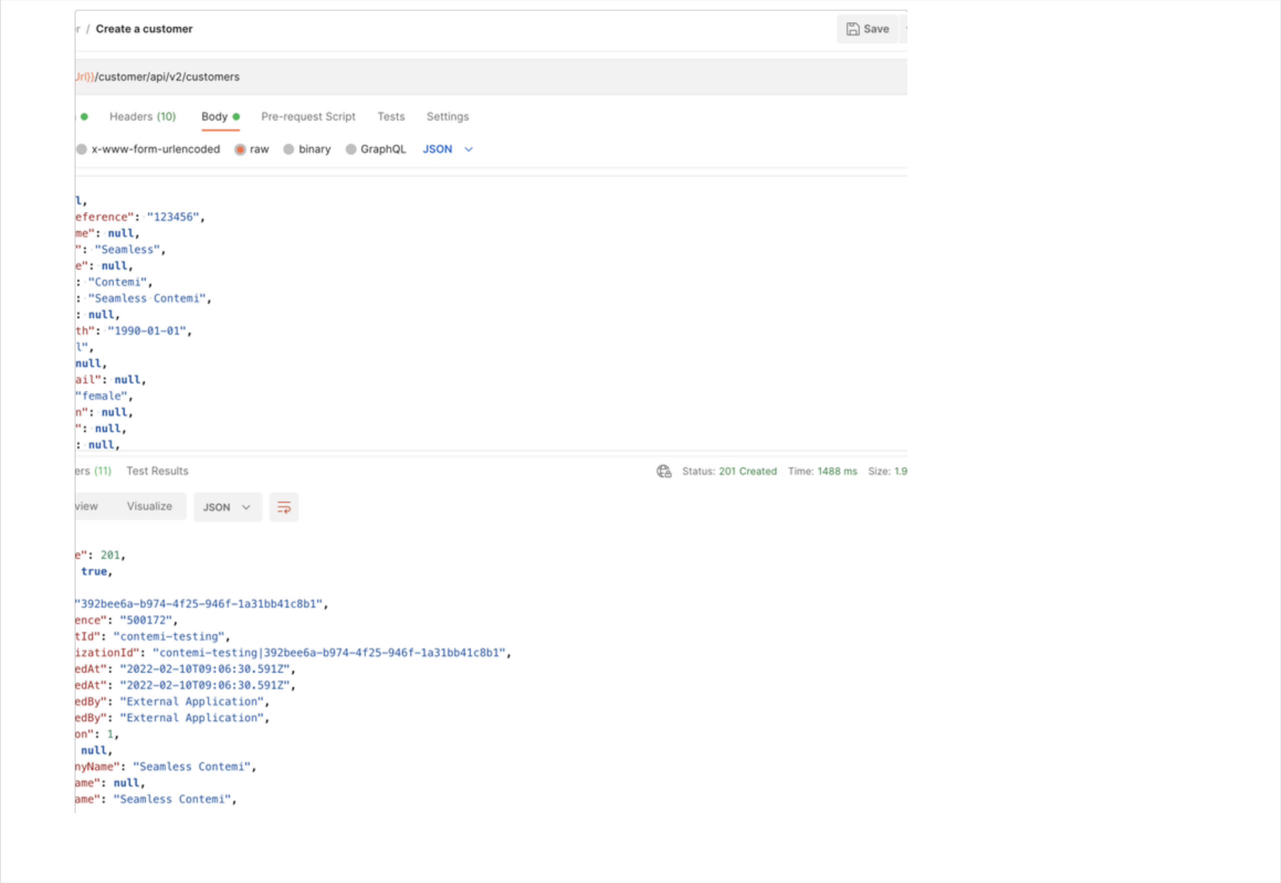
1. Create a customer

- URL: https://external-api.seamless.insure/customer/api/v2/customers
- Method: POST
- Header:

Content-Type: application/json

Authorization: Bearer {{accessToken}}

- Postman example (please see it on the attachments):



r / Create a customer

Jrl}}/customer/api/v2/customers

○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
l,
eference": "123456",
me": null,
": "Seamless",
e": null,
: "Contemi",
: "Seamless Contemi",
: null,
th": "1990-01-01",
l",
null,
ail": null,
"female",
n": null,
": null,
: null,
```

Status: 201 Created Time: 1488 ms Size: 1.9

```
e": 201,
 true,

"392bee6a-b974-4f25-946f-1a31bb41c8b1",
ence": "500172",
tId": "contemi-testing",
izationId": "contemi-testing|392bee6a-b974-4f25-946f-1a31bb41c8b1",
edAt": "2022-02-10T09:06:30.591Z",
edAt": "2022-02-10T09:06:30.591Z",
edBy": "External Application",
edBy": "External Application",
on": 1,
 null,
nyName": "Seamless Contemi",
ame": null,
ame": "Seamless Contemi",
```

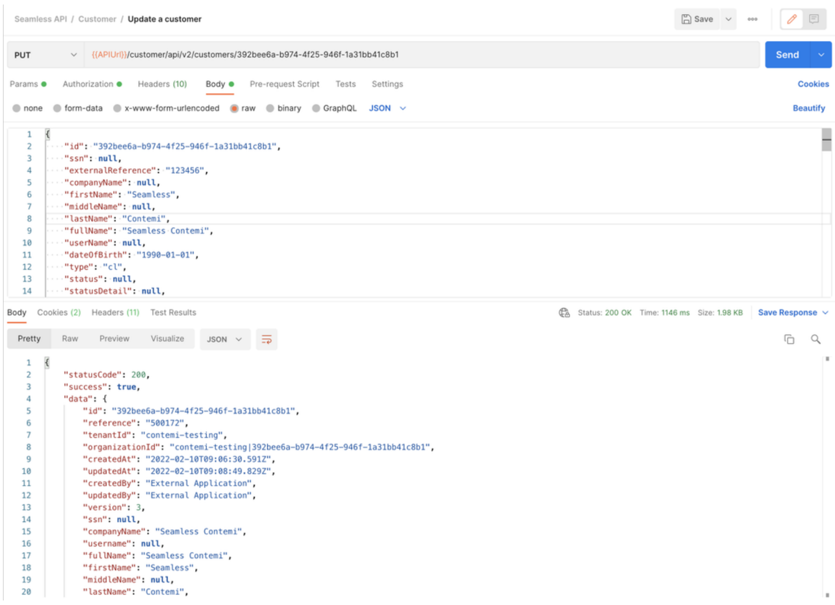
2. Update a customer

- URL: https://external-api.seamless.insure/customer/api/v2/customers/{{customerId}}
- Method: PUT
- Header:

Content-Type: application/json

Authorization: Bearer {{accessToken}}

- Body: (please see it on the Postman example)
- Postman example (please see it on the attachments):



PUT {{APIUrl}}/customer/api/v2/customers/392bee6a-b974-4f25-946f-1a31bb41c8b1

```
{
    "id": "392bee6a-b974-4f25-946f-1a31bb41c8b1",
    "ssn": null,
    "externalReference": "123456",
    "companyName": null,
    "firstName": "Seamless",
    "middleName": null,
    "lastName": "Contemi",
    "fullName": "Seamless Contemi",
    "userName": null,
    "dateOfBirth": "1990-01-01",
    "type": "cl",
    "status": null,
    "statusDetail": null,
```

Status: 200 OK Time: 1146 ms Size: 1.98 KB

```
{
    "statusCode": 200,
    "success": true,
    "data": {
        "id": "392bee6a-b974-4f25-946f-1a31bb41c8b1",
        "reference": "500172",
        "tenantId": "contemi-testing",
        "organizationId": "contemi-testing|392bee6a-b974-4f25-946f-1a31bb41c8b1",
        "createdAt": "2022-02-10T09:06:30.591Z",
        "updatedAt": "2022-02-10T09:08:49.829Z",
        "createdBy": "External Application",
        "updatedBy": "External Application",
        "version": 3,
        "ssn": null,
        "companyName": "Seamless Contemi",
        "username": null,
        "fullName": "Seamless Contemi",
        "firstName": "Seamless",
        "middleName": null,
        "lastName": "Contemi",
```

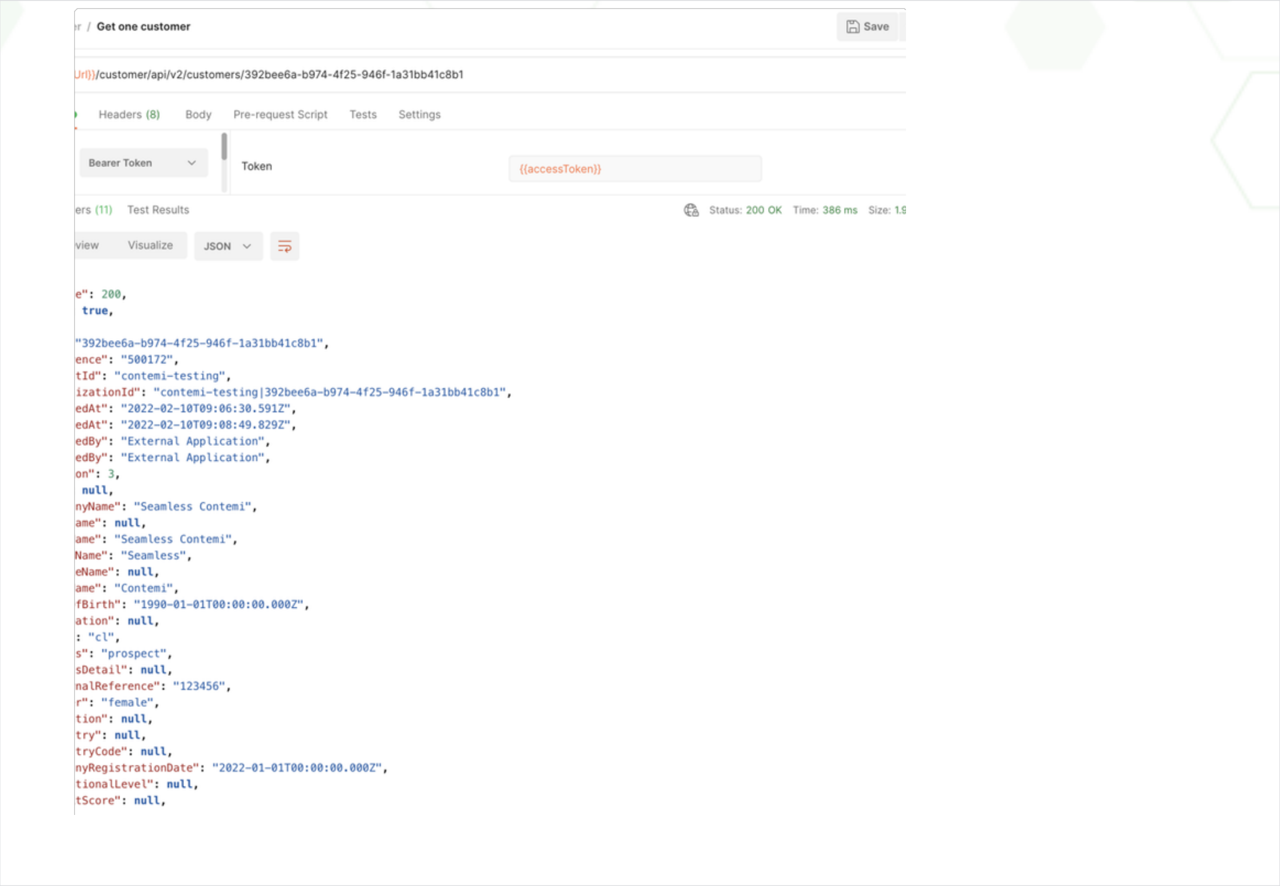
3. Get one customer

- URL: https://external-api.seamless.insure/customer/api/v2/customers/{{customerId}}

  (We can replace {{customerId}} with Id of customer on Seamless)
- Method: GET
- Header:

  Content-Type: application/json

  Authorization: Bearer {{accessToken}}
- Postman example (please see it on the attachments):



Url}}/customer/api/v2/customers/392bee6a-b974-4f25-946f-1a31bb41c8b1

Bearer Token

Token {{accessToken}}

Status: 200 OK Time: 386 ms Size: 1.9

```
e": 200,
 true,

"392bee6a-b974-4f25-946f-1a31bb41c8b1",
ence": "500172",
tId": "contemi-testing",
izationId": "contemi-testing|392bee6a-b974-4f25-946f-1a31bb41c8b1",
edAt": "2022-02-10T09:06:30.591Z",
edAt": "2022-02-10T09:08:49.829Z",
edBy": "External Application",
edBy": "External Application",
on": 3,
 null,
nyName": "Seamless Contemi",
ame": null,
ame": "Seamless Contemi",
Name": "Seamless",
eName": null,
ame": "Contemi",
fBirth": "1990-01-01T00:00:00.000Z",
ation": null,
: "cl",
s": "prospect",
sDetail": null,
nalReference": "123456",
r": "female",
tion": null,
try": null,
tryCode": null,
nyRegistrationDate": "2022-01-01T00:00:00.000Z",
tionalLevel": null,
tScore": null,
```

4. Search customers

- URL: https://external-api.seamless.insure/search/api/v1/customers/search
- Method: POST
- Request Body:

```
"Page": 1,

"ItemsPerPage": 10, "Criteria": { "Profile": [],

"SSN": null, "FullName": null,

"Type": null // "pl" or "cl",

...

}
```

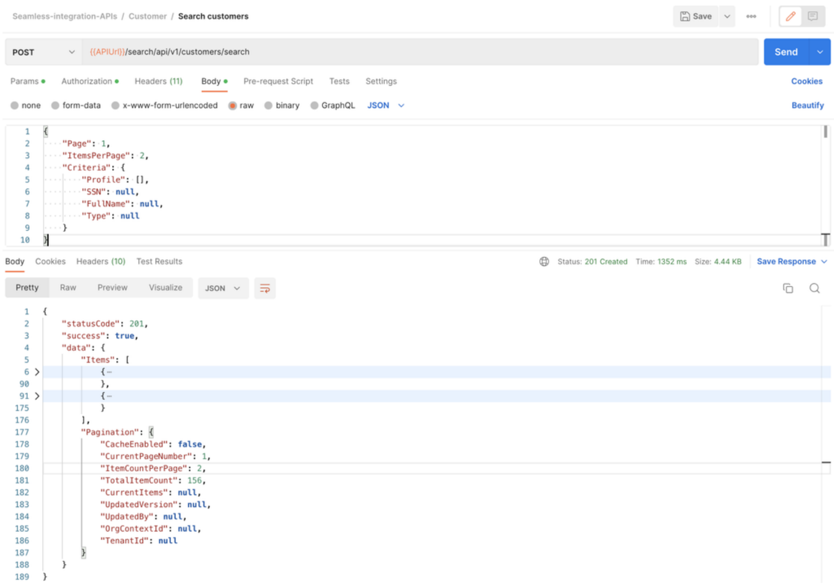
*Note: Type is "pl" or "cl"

- Header:

Content-Type: application/json

Authorization: Bearer {{accessToken}}

- Postman example (please see it on the attachments):



Seamless-integration-APIs / Customer / **Search customers**

POST {{APIUrl}}/search/api/v1/customers/search

Body: ○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
{
    "Page": 1,
    "ItemsPerPage": 2,
    "Criteria": {
        "Profile": [],
        "SSN": null,
        "FullName": null,
        "Type": null
    }
}
```

Body — Status: 201 Created Time: 1352 ms Size: 4.44 KB — JSON

```
{
    "statusCode": 201,
    "success": true,
    "data": {
        "Items": [
            {…
            },
            {…
            }
        ],
        "Pagination": {
            "CacheEnabled": false,
            "CurrentPageNumber": 1,
            "ItemCountPerPage": 2,
            "TotalItemCount": 156,
            "CurrentItems": null,
            "UpdatedVersion": null,
            "UpdatedBy": null,
            "OrgContextId": null,
            "TenantId": null
        }
    }
}
```

5. Get customers

- URL: https://external-api.seamless.insure/customer/api/v2/customers?page=1&itemsPerPage=10&order={"createdAt":"ASC"}&externalReference=12345abc
- Method: GET
- Request Params:

page: 1

itemsPerPage: 10

order: {"createdAt":"ASC"}

externalReference: 12345abc

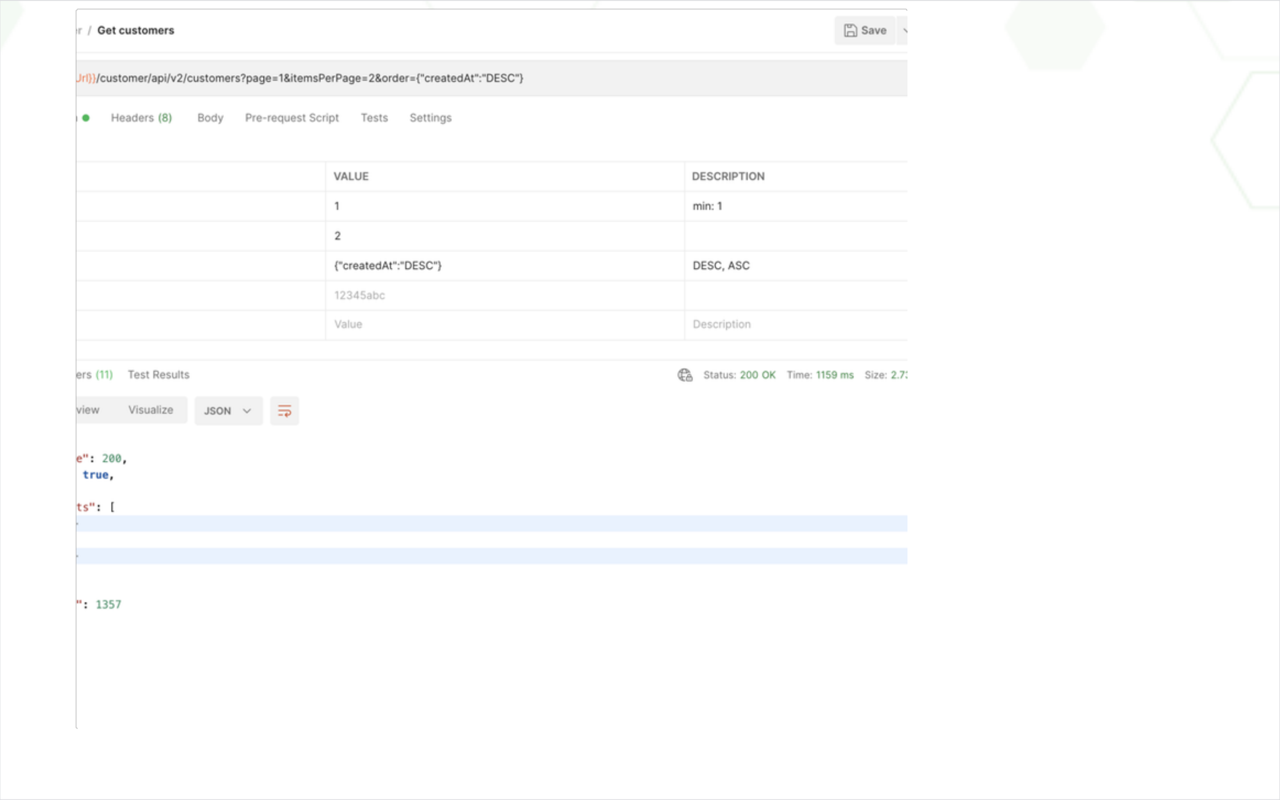
Note: that value is example, we can modify it, order directions: ASC or DESC

- Header:

Content-Type: application/json

Authorization: Bearer {{accessToken}}

- Postman example (please see it on the attachments):



r / **Get customers**

Save

Url}}/customer/api/v2/customers?page=1&itemsPerPage=2&order={"createdAt":"DESC"}

Headers (8) Body Pre-request Script Tests Settings

VALUE	DESCRIPTION
1	min: 1
2	
{"createdAt":"DESC"}	DESC, ASC
12345abc	
Value	Description

ers (11) Test Results Status: 200 OK Time: 1159 ms Size: 2.7

view Visualize JSON

```
e": 200,
 true,

ts": [



": 1357
```

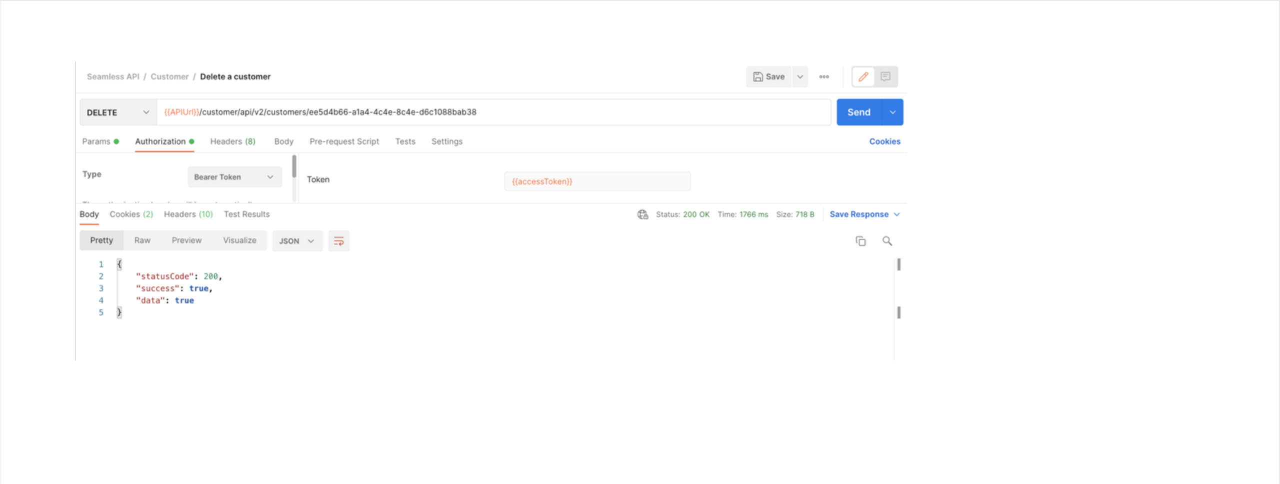
6. Delete a customer

- URL: https://external-api.seamless.insure/customer/api/v2/customers/{{customerId}} (We can replace {{customerId}} with Id of customer on Seamless)
- Method: DELETE
- Header:

    Content-Type: application/json

    Authorization: Bearer {{accessToken}}

- Postman example (please see it on the attachments):



Seamless API / Customer / **Delete a customer**

Save •••

DELETE {{APIUrl}}/customer/api/v2/customers/ee5d4b66-a1a4-4c4e-8c4e-d6c1088bab38 Send

Params ● Authorization ● Headers (8) Body Pre-request Script Tests Settings Cookies

Type Bearer Token Token {{accessToken}}

Body Cookies (2) Headers (10) Test Results Status: 200 OK Time: 1766 ms Size: 718 B Save Response

Pretty Raw Preview Visualize JSON

```
{
    "statusCode": 200,
    "success": true,
    "data": true
}
```

Quotation Service

This page describes basic steps for browsing available products, select one and create a quote, update the quote, specify sales discounts and commissions, and accept and purchase the quote.

Authenticate and retrieve an access token

Every request to Seamless needs to be authenticated. Please refer to the Authentication for more details of how to authenticate and get an access token.

Get available products for sales

To get product versions available for sales, please send an HTTP request to the Agency Configuration API as in the example below.

```
### Get active product versions
GET   {{base-url-configuration}}/api/v1/salesProducts/activeVersions?page[size]=100&page[number]=1
Authorization:  {{access-token}}
Accept: application/vnd.api+json
```

Sample response

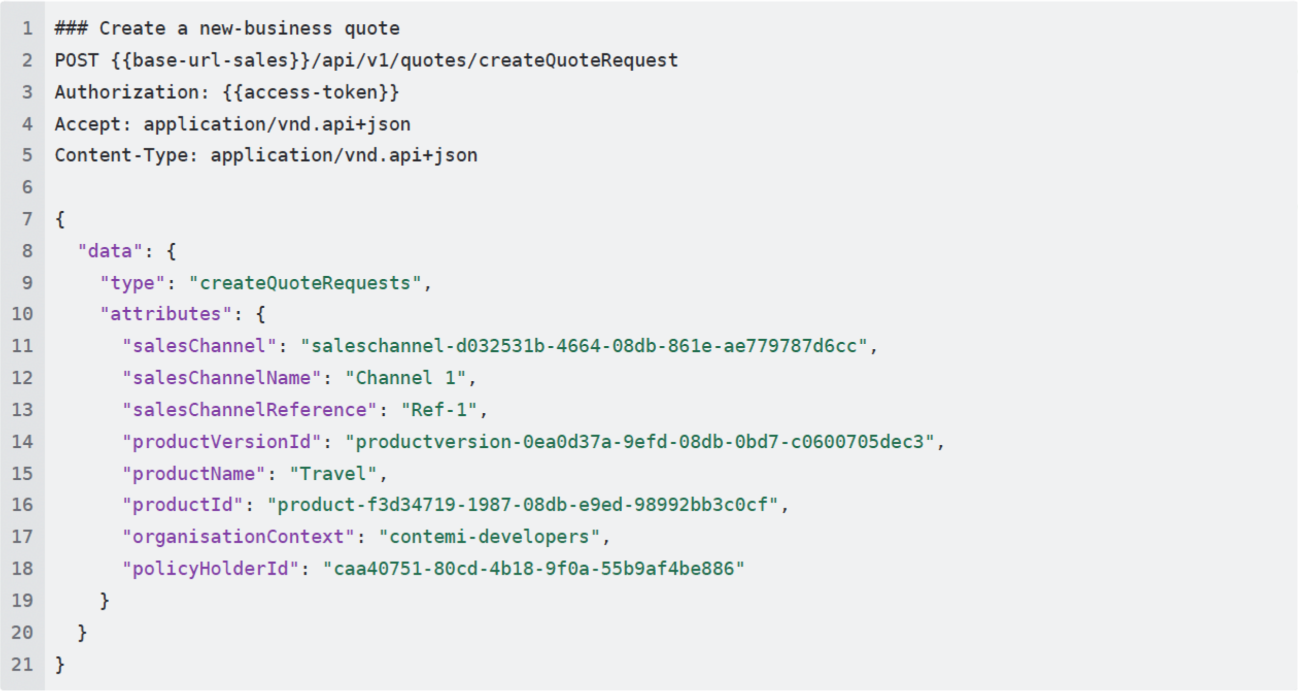
```
HTTP/2 200 OK
{
  "data": [
    {
      "type": "salesProductVersions",
      "id": "4c2fbd66-301f-4885-96cb-d60be50b113c",
      "attributes": {
        "productVersionId": "productversion-0ea0d37a-9efd-08db-0bd7-c0600705dec3",
        "productId": "product-f3d34719-1987-08db-e9ed-98992bb3c0cf",
        "productName": "Travel",
        "salesChannelId": "saleschannel-d032531b-4664-08db-861e-ae779787d6cc",
        "salesChannelName": "Channel 1",
        "salesChannelReference": "Ref-1",
        "productIconId": "2cddad98-7e2c-4d58-b91b-42c1752663d4",
        "iconBackgroundColor": "#fff",
        "paymentOptions": null
      }
    },
    ...
  ]
}
```

Create a new-business quote

Send a request with request payload attributes as below to the Quotation Service.

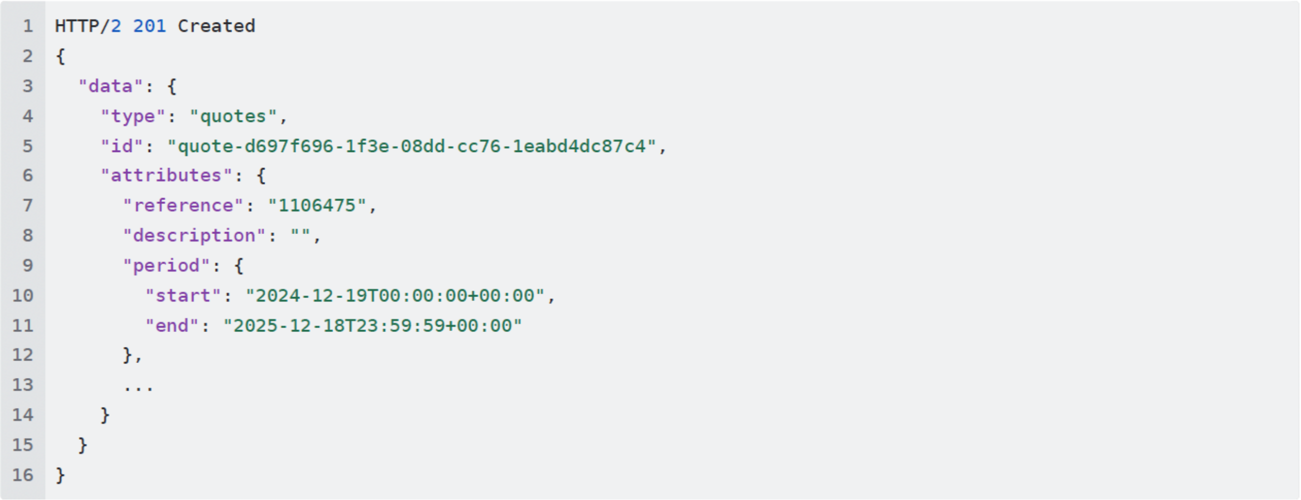
- `policyHolderId` is the customer Id.
- `salesChannel` , `salesChannelName` , `productVersionId` , `productName` , and `productID` are data from the response of step.

  Get available production versions for sales.
- `organisationContext` is the authenticated user's organization context.



```
### Create a new-business quote
POST {{base-url-sales}}/api/v1/quotes/createQuoteRequest
Authorization: {{access-token}}
Accept: application/vnd.api+json
Content-Type: application/vnd.api+json

{
  "data": {
    "type": "createQuoteRequests",
    "attributes": {
      "salesChannel": "saleschannel-d032531b-4664-08db-861e-ae779787d6cc",
      "salesChannelName": "Channel 1",
      "salesChannelReference": "Ref-1",
      "productVersionId": "productversion-0ea0d37a-9efd-08db-0bd7-c0600705dec3",
      "productName": "Travel",
      "productId": "product-f3d34719-1987-08db-e9ed-98992bb3c0cf",
      "organisationContext": "contemi-developers",
      "policyHolderId": "caa40751-80cd-4b18-9f0a-55b9af4be886"
    }
  }
}
```

Sample response

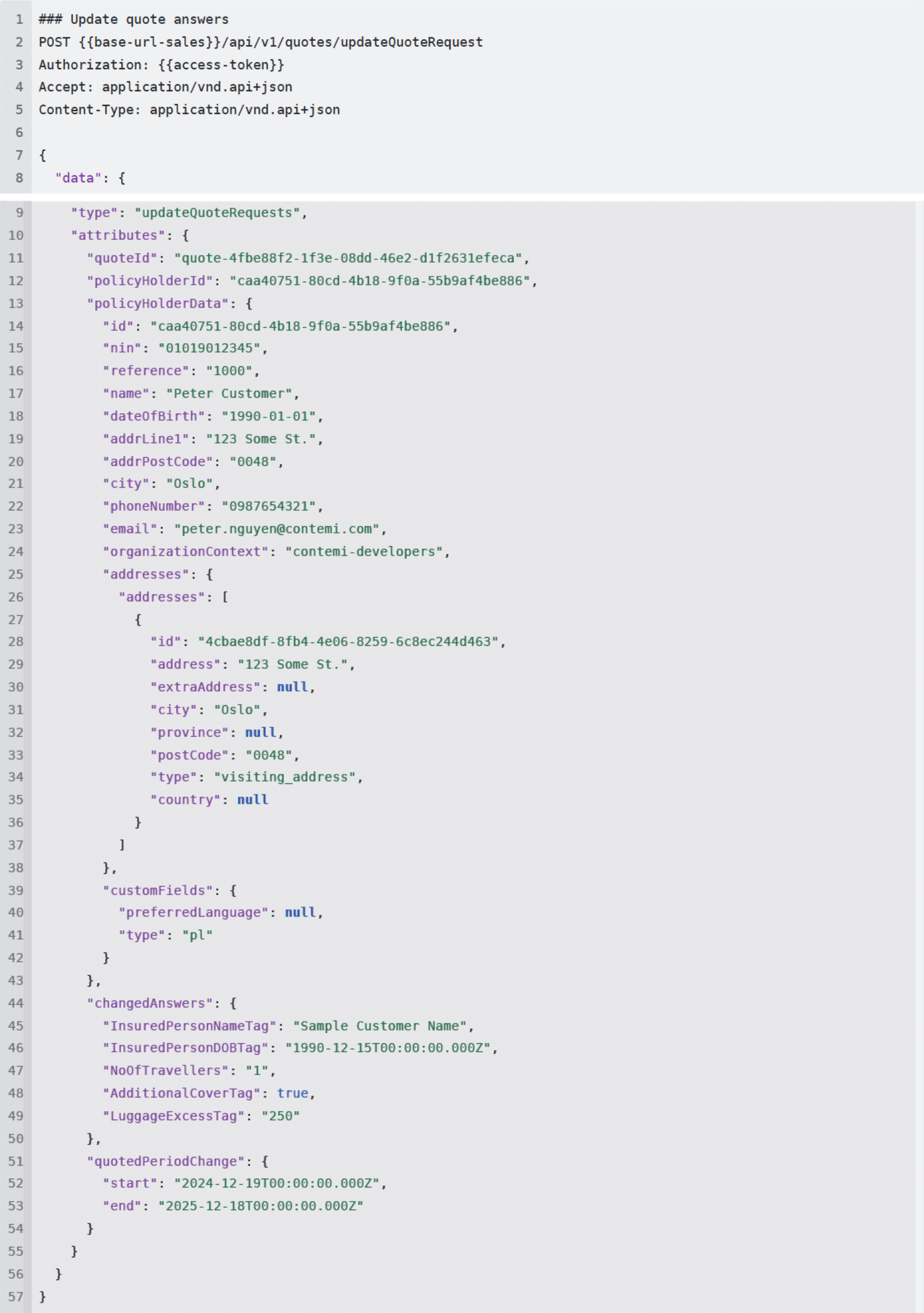


```
HTTP/2 201 Created
{
  "data": {
    "type": "quotes",
    "id": "quote-d697f696-1f3e-08dd-cc76-1eabd4dc87c4",
    "attributes": {
      "reference": "1106475",
      "description": "",
      "period": {
        "start": "2024-12-19T00:00:00+00:00",
        "end": "2025-12-18T23:59:59+00:00"
      },
      ...
    }
  }
}
```

Update quote answers

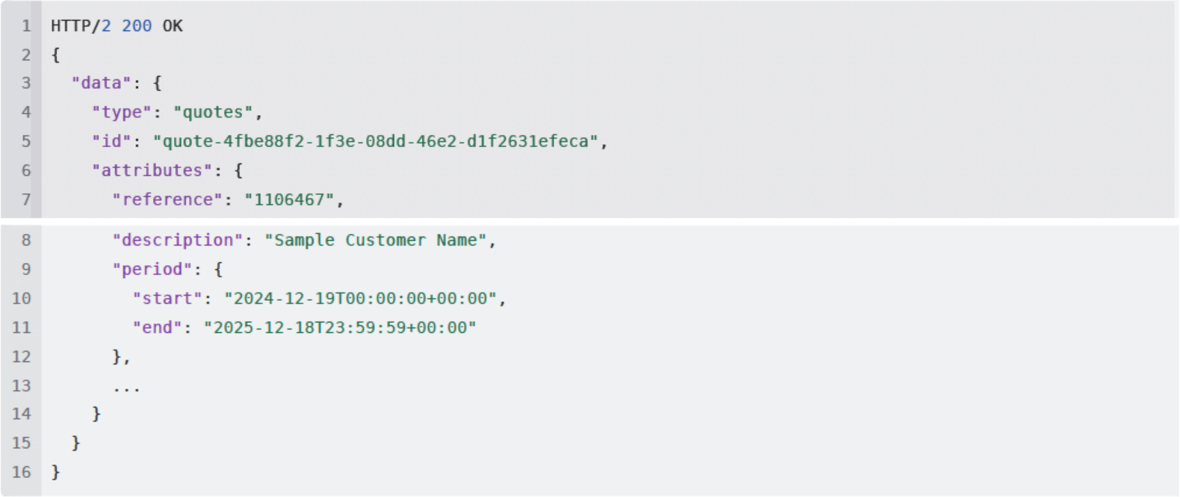
To update the quote answers, send a request with the payload as below to the Quotation Service.

- `quoteId` is the quote Id taken from the previous step.
- `policyHolderId` and `policyHolderData` are customer data.
- The answer tags and values in `changedAnswers` are varied and defined by the product version associated with the quote.



```
### Update quote answers
POST {{base-url-sales}}/api/v1/quotes/updateQuoteRequest
Authorization: {{access-token}}
Accept: application/vnd.api+json
Content-Type: application/vnd.api+json

{
  "data": {
    "type": "updateQuoteRequests",
    "attributes": {
      "quoteId": "quote-4fbe88f2-1f3e-08dd-46e2-d1f2631efeca",
      "policyHolderId": "caa40751-80cd-4b18-9f0a-55b9af4be886",
      "policyHolderData": {
        "id": "caa40751-80cd-4b18-9f0a-55b9af4be886",
        "nin": "01019012345",
        "reference": "1000",
        "name": "Peter Customer",
        "dateOfBirth": "1990-01-01",
        "addrLine1": "123 Some St.",
        "addrPostCode": "0048",
        "city": "Oslo",
        "phoneNumber": "0987654321",
        "email": "peter.nguyen@contemi.com",
        "organizationContext": "contemi-developers",
        "addresses": {
          "addresses": [
            {
              "id": "4cbae8df-8fb4-4e06-8259-6c8ec244d463",
              "address": "123 Some St.",
              "extraAddress": null,
              "city": "Oslo",
              "province": null,
              "postCode": "0048",
              "type": "visiting_address",
              "country": null
            }
          ]
        },
        "customFields": {
          "preferredLanguage": null,
          "type": "pl"
        }
      },
      "changedAnswers": {
        "InsuredPersonNameTag": "Sample Customer Name",
        "InsuredPersonDOBTag": "1990-12-15T00:00:00.000Z",
        "NoOfTravellers": "1",
        "AdditionalCoverTag": true,
        "LuggageExcessTag": "250"
      },
      "quotedPeriodChange": {
        "start": "2024-12-19T00:00:00.000Z",
        "end": "2025-12-18T00:00:00.000Z"
      }
    }
  }
}
```

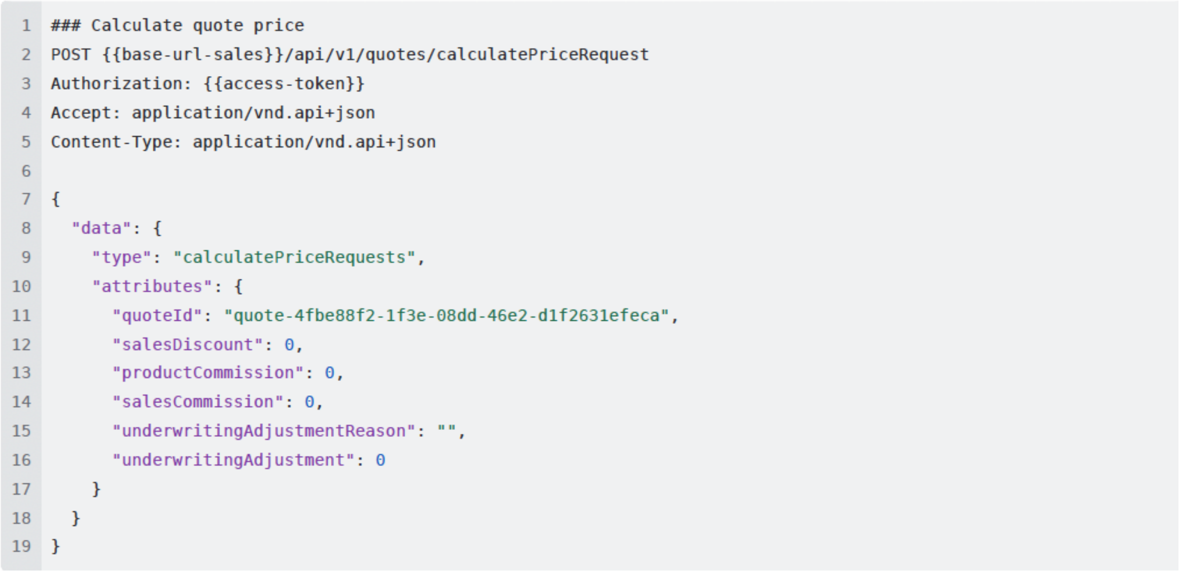
Sample response



```
HTTP/2 200 OK
{
  "data": {
    "type": "quotes",
    "id": "quote-4fbe88f2-1f3e-08dd-46e2-d1f2631efeca",
    "attributes": {
      "reference": "1106467",
      "description": "Sample Customer Name",
      "period": {
        "start": "2024-12-19T00:00:00+00:00",
        "end": "2025-12-18T23:59:59+00:00"
      },
      ...
    }
  }
}
```

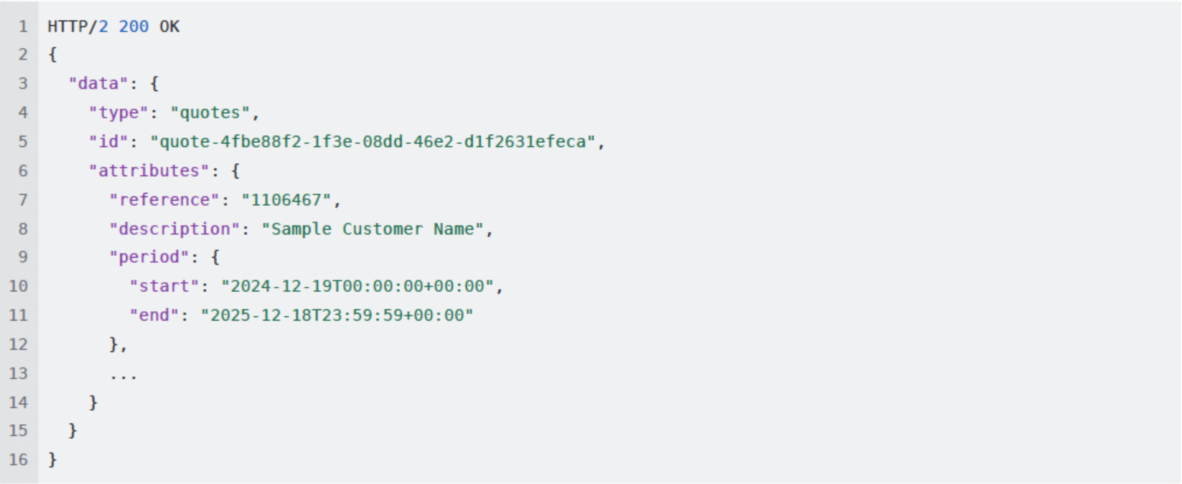
Calculate quote price

To calculate a quote price, send a request to the Quotation Service.



```
### Calculate quote price
POST {{base-url-sales}}/api/v1/quotes/calculatePriceRequest
Authorization: {{access-token}}
Accept: application/vnd.api+json
Content-Type: application/vnd.api+json

{
  "data": {
    "type": "calculatePriceRequests",
    "attributes": {
      "quoteId": "quote-4fbe88f2-1f3e-08dd-46e2-d1f2631efeca",
      "salesDiscount": 0,
      "productCommission": 0,
      "salesCommission": 0,
      "underwritingAdjustmentReason": "",
      "underwritingAdjustment": 0
    }
  }
}
```

Sample response



```
HTTP/2 200 OK
{
  "data": {
    "type": "quotes",
    "id": "quote-4fbe88f2-1f3e-08dd-46e2-d1f2631efeca",
    "attributes": {
      "reference": "1106467",
      "description": "Sample Customer Name",
      "period": {
        "start": "2024-12-19T00:00:00+00:00",
        "end": "2025-12-18T23:59:59+00:00"
      },
      ...
    }
  }
}
```

Sales discount, product commission, sales commission, and underwriting adjustment can be specified in the request payload.

Poll quote price

Since quote price is calculated asynchronously, clients may need to poll for the quote price calculation complete.

To poll quote price, send a request like the example below, replace the `:id` path variable with the quote Id. The request can be repeated until reaching the max polling time or receiving a response with `hasBeenCalculated` attribute as `true` .

```
### Poll quote price
GET {{base-url-sales}}/api/v1/quotes/:id?fields[quotes]=premium
Authorization: {{access-token}}
Accept: application/vnd.api+json
```

Sample response

```
HTTP/2 200 OK
{
  "data": {
    "type": "quotes",
    "id": "quote-4fbe88f2-1f3e-08dd-46e2-d1f2631efeca",
    "attributes": {
      "premium": {
        "isBeingCalculated": false,
        "hasBeenCalculated": true,
        "isConstrained": false,
        "canBeAccessed": true
      }
    }
  }
}
```

Specify a payment option

A payment option may need to be specified for the quote, depending on the agency configuration. To specify the payment option, send a request like below.

```
### Specify payment option
POST {{base-url-sales}}/api/v1/quotes/updateQuoteRequest
Authorization: {{access-token}}
Accept: application/vnd.api+json
Content-Type: application/vnd.api+json

{
  "data": {
    "type": "updateQuoteRequests",
    "attributes": {
      "quoteId": "quote-4fbe88f2-1f3e-08dd-46e2-d1f2631efeca",
      "paymentOptionName": "Cash Monthly",
      "paymentOptionValue": "cash_monthly",
      "paymentMethod": "External",
      "paymentFrequency": "Monthly"
    }
  }
}
```

Sample response

```
HTTP/2 200 OK
{
  "data": {
```

```
    "type": "quotes",
    "id": "quote-4fbe88f2-1f3e-08dd-46e2-d1f2631efeca",
    "attributes": {
      "reference": "1106467",
      "description": "Sample Customer Name",
      "period": {
        "start": "2024-12-19T00:00:00+00:00",
        "end": "2025-12-18T23:59:59+00:00"
      },
      ...
    }
  }
}
```

Complete quote

Send a request with a `quoteId` specified in the request payload. Take note of the `version` and the `policyHolderId` in the response payload.

```
### Complete quote
POST {{base-url-sales}}/api/v1/quotes/completeQuoteRequest
Authorization: {{access-token}}
Accept: application/vnd.api+json
Content-Type: application/vnd.api+json

{
  "data": {
    "type": "completeQuoteRequests",
    "attributes": {
      "quoteId": "quote-4fbe88f2-1f3e-08dd-46e2-d1f2631efeca"
    }
  }
}
```

Sample response

```
HTTP/2 200 OK
{
  "data": {
    "type": "quotes",
    "id": "quote-4fbe88f2-1f3e-08dd-46e2-d1f2631efeca",
    "attributes": {
      "reference": "1106467",
      "description": "Sample Customer Name",
      "period": {
        "start": "2024-12-19T00:00:00+00:00",
        "end": "2025-12-18T23:59:59+00:00"
      },
      "version": 28,
      "policyHolderId": "caa40751-80cd-4b18-9f0a-55b9af4be886",
      ...
    }
  }
}
```

Accept quote

Send a request with following information.

`quoteId` : Id of the quote to accept.

`acceptedVersion` : The quote version noted down in the previous step.

`policyHolderId` : Id of the policy holder noted down in the previous step.

`paymentMethod` : Can be one of `Invoice` , `IntegratedCheckout` , or `External` .

`paymentFrequency` : Choose one of these available options `Annually` , `Semi-Annually` , `Quarterly` , `Monthly` , `Single` or `One-Off`

```
### Accept quote
POST {{base-url-sales}}/api/v1/quotes/acceptanceRequest
Authorization: {{access-token}}
Accept: application/vnd.api+json
Content-Type: application/vnd.api+json

{
  "data": {
    "type": "acceptanceRequests",
    "attributes": {
      "quoteId": "quote-4fbe88f2-1f3e-08dd-46e2-d1f2631efeca",
      "acceptedVersion": 28,
      "policyHolderId": "caa40751-80cd-4b18-9f0a-55b9af4be886",
      "preferredCollectionDay": 1
    }
  }
}
```

Sample response

```
HTTP/2 200 OK
{
  "data": {
    "type": "quotes",
    "id": "quote-4fbe88f2-1f3e-08dd-46e2-d1f2631efeca",
    "attributes": {
      "reference": "1106467",
      "description": "Sample Customer Name",
      "period": {
        "start": "2024-12-19T00:00:00+00:00",
        "end": "2025-12-18T23:59:59+00:00"
      },
      ...
    }
  }
}
```

Poll issued policy Id

Since the policy is issued asynchronously, the client should poll for the policy issuance result by sending the following request. Please replace the `:id` path variable with the actual quote Id.

```
### Poll policy issuance
GET {{base-url-sales}}/api/v1/quotes/:id?fields[quotes]=status,underwrittenByPolicyId,acceptanceFailureReason
Authorization: {{access-token}}
Accept: application/vnd.api+json
```

Sample response

```
HTTP/2 200 OK
```

```
{
  "data": {
    "type": "quotes",
    "id": "quote-4fbe88f2-1f3e-08dd-46e2-d1f2631efeca",
    "attributes": {
      "status": "Underwritten",
      "underwrittenByPolicyId": "policy-6f21f713-f977-5ded-a1d0-15d82ee48931",
      "acceptanceFailureReason": null
    }
  }
}
```

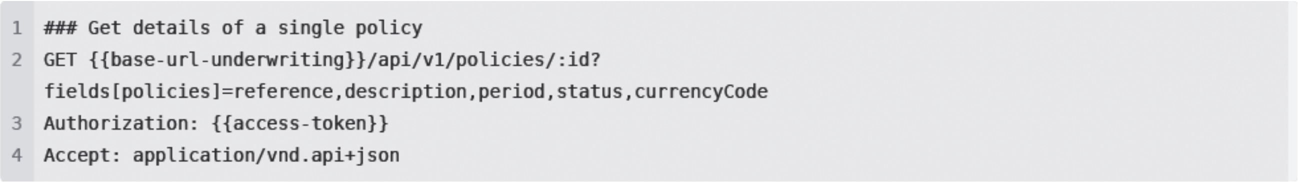
Please take note of the `underwrittenByPolicyId` if you need to retrieve the policy details.

Policy Management

This page provides supplementary information for the Policy Management API.

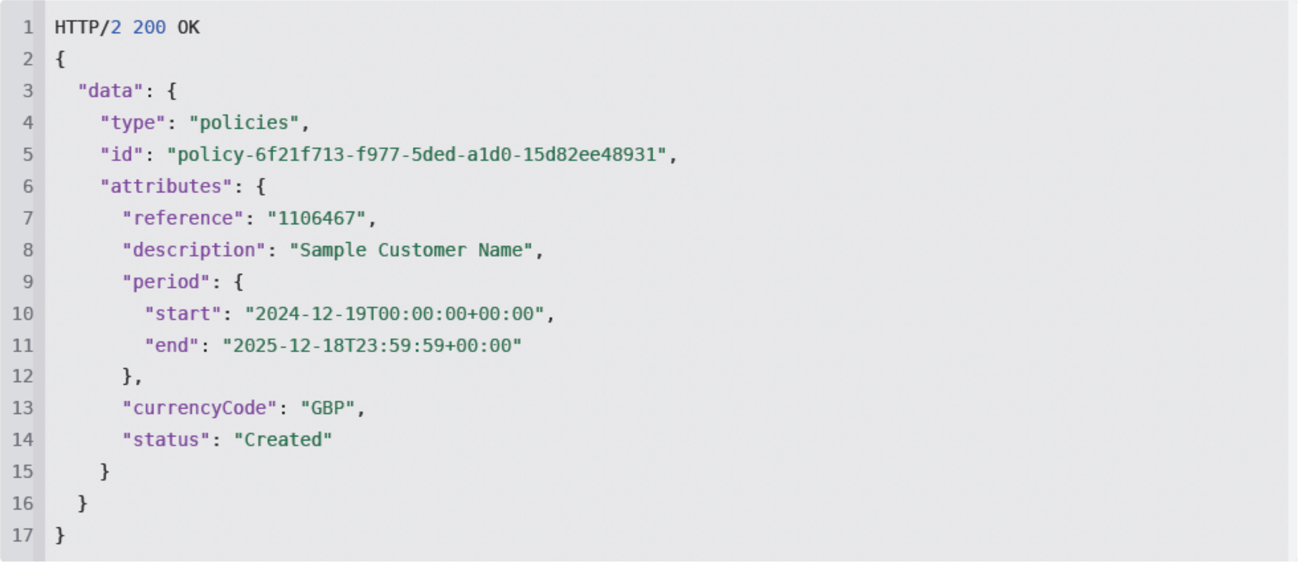
Get policy details

Send a request that looks like below to retrieve details of a single policy. Please replace the `:id` path variable with the actual policy Id.



```
### Get details of a single policy
GET {{base-url-underwriting}}/api/v1/policies/:id?fields[policies]=reference,description,period,status,currencyCode
Authorization: {{access-token}}
Accept: application/vnd.api+json
```

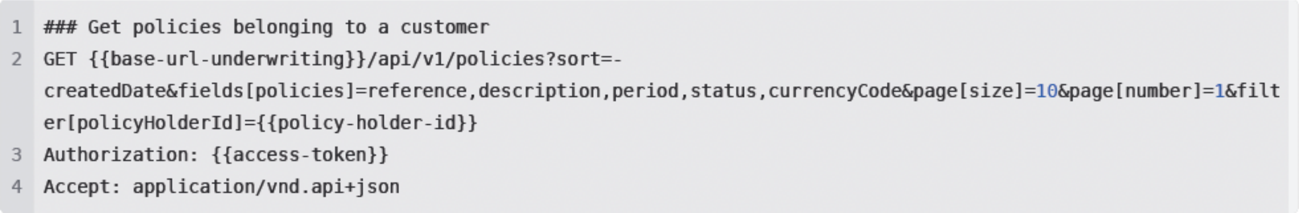
Sample response



```
HTTP/2 200 OK
{
  "data": {
    "type": "policies",
    "id": "policy-6f21f713-f977-5ded-a1d0-15d82ee48931",
    "attributes": {
      "reference": "1106467",
      "description": "Sample Customer Name",
      "period": {
        "start": "2024-12-19T00:00:00+00:00",
        "end": "2025-12-18T23:59:59+00:00"
      },
      "currencyCode": "GBP",
      "status": "Created"
    }
  }
}
```

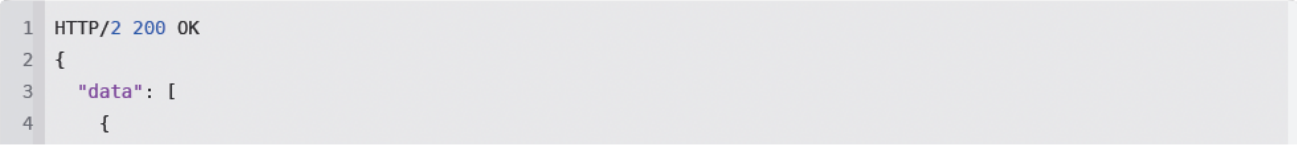
Get policies belonging to a customer

To get policies belonging to a customer, send a request with parameters for paging, sorting, filtering, and a sparse fieldset selection. Please replace the `{{policy-holder-id}}` query parameter with the actual policy holder Id (aka the customer Id). For example,

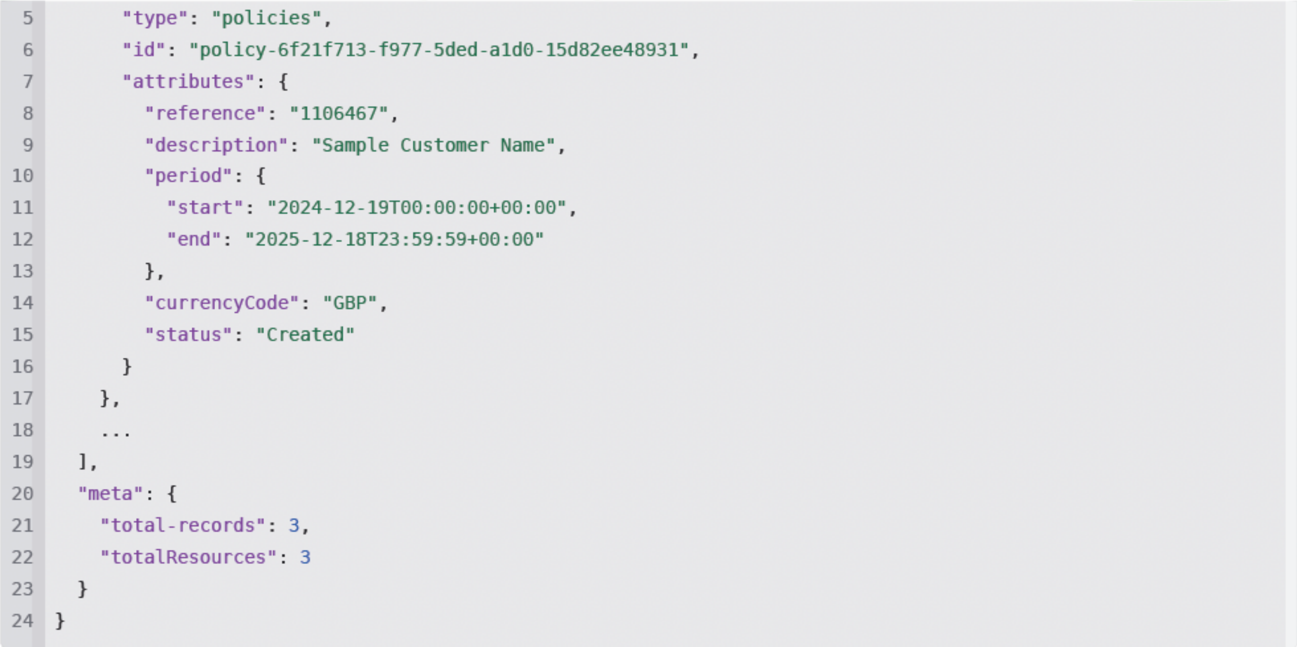


```
### Get policies belonging to a customer
GET {{base-url-underwriting}}/api/v1/policies?sort=-createdDate&fields[policies]=reference,description,period,status,currencyCode&page[size]=10&page[number]=1&filter[policyHolderId]={{policy-holder-id}}
Authorization: {{access-token}}
Accept: application/vnd.api+json
```

Sample response



```
HTTP/2 200 OK
{
  "data": [
    {
```



```
      "type": "policies",
      "id": "policy-6f21f713-f977-5ded-a1d0-15d82ee48931",
      "attributes": {
        "reference": "1106467",
        "description": "Sample Customer Name",
        "period": {
          "start": "2024-12-19T00:00:00+00:00",
          "end": "2025-12-18T23:59:59+00:00"
        },
        "currencyCode": "GBP",
        "status": "Created"
      }
    },
    ...
  ],
  "meta": {
    "total-records": 3,
    "totalResources": 3
  }
}
```

Customer Credit check

1. Credit checks for consumer (personal customer)

- URL:

https://external-api.seamless.insure/connector-api/api/credit/consumer/{{ssn}}

(We can replace *{{ssn}}* with SSN (Org. No) of customer on Seamless)

- Params:

forceNew=true: always call credit API to check

forceNew=false (or undefined):

+ If this customer hasn’t been previously checked, API will proceed a call to CREDITSAFE.

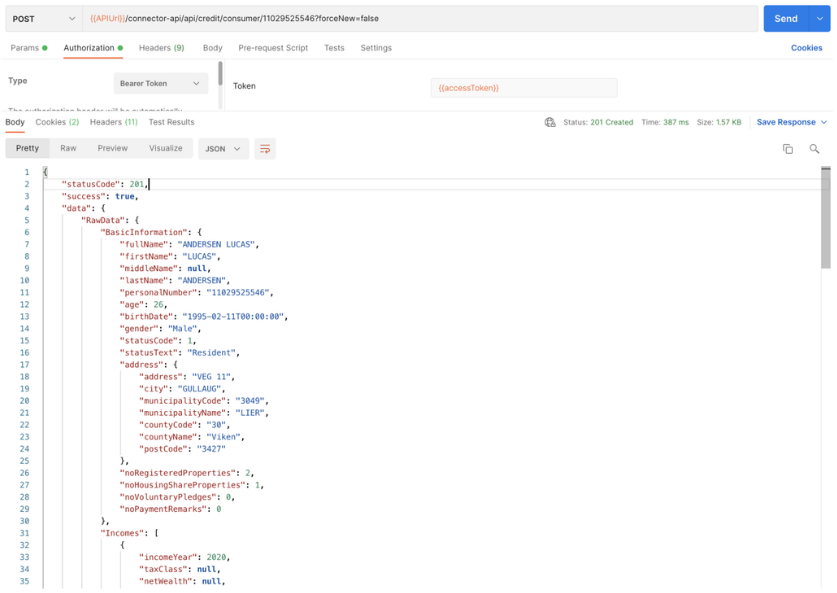
+ If this customer has been previously checked, API will proceed a call to CREDITSAFE if and only if the period is expired.

- Method: POST
- Header:

Content-Type: application/json

Authorization: Bearer {{accessToken}}

Postman example (please see it on the attachments):



POST {{APIUrl}}/connector-api/api/credit/consumer/11029525546?forceNew=false

Type: Bearer Token

Token: {{accessToken}}

Status: 201 Created Time: 387 ms Size: 1.57 KB

```
{
    "statusCode": 201,
    "success": true,
    "data": {
        "RawData": {
            "BasicInformation": {
                "fullName": "ANDERSEN LUCAS",
                "firstName": "LUCAS",
                "middleName": null,
                "lastName": "ANDERSEN",
                "personalNumber": "11029525546",
                "age": 26,
                "birthDate": "1995-02-11T00:00:00",
                "gender": "Male",
                "statusCode": 1,
                "statusText": "Resident",
                "address": {
                    "address": "VEG 11",
                    "city": "GULLAUG",
                    "municipalityCode": "3049",
                    "municipalityName": "LIER",
                    "countyCode": "30",
                    "countyName": "Viken",
                    "postCode": "3427"
                },
                "noRegisteredProperties": 2,
                "noHousingShareProperties": 1,
                "noVoluntaryPledges": 0,
                "noPaymentRemarks": 0
            },
            "Incomes": [
                {
                    "incomeYear": 2020,
                    "taxClass": null,
                    "netWealth": null,
```

1. Credit checks for company

- URL:

https://external-api.seamless.insure/connector-api/api/credit/company/{{ssn}}

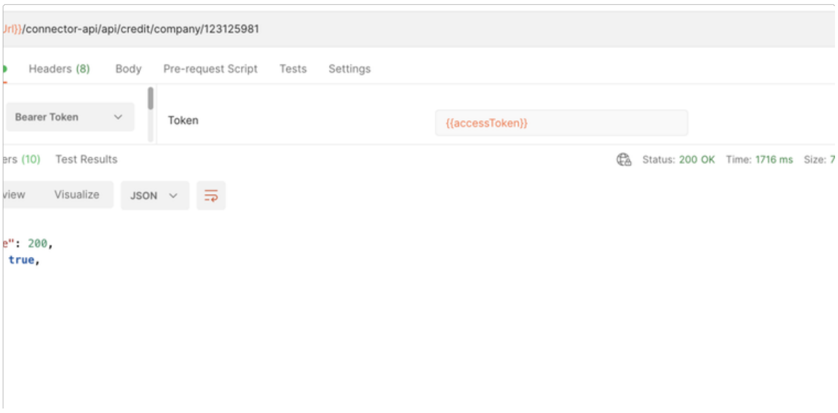
(We can replace *{{ssn}}* with SSN (Org. No) of customer on Seamless)

- Method: POST
- Header:

  Content-Type: application/json

  Authorization: Bearer {{accessToken}}

- Postman example (please see it on the attachments):



Jrl}}/connector-api/api/credit/company/123125981

Bearer Token

Token {{accessToken}}

Status: 200 OK Time: 1716 ms Size: 7

```
e": 200,
 true,
```

Credit checks for company

Opportunity

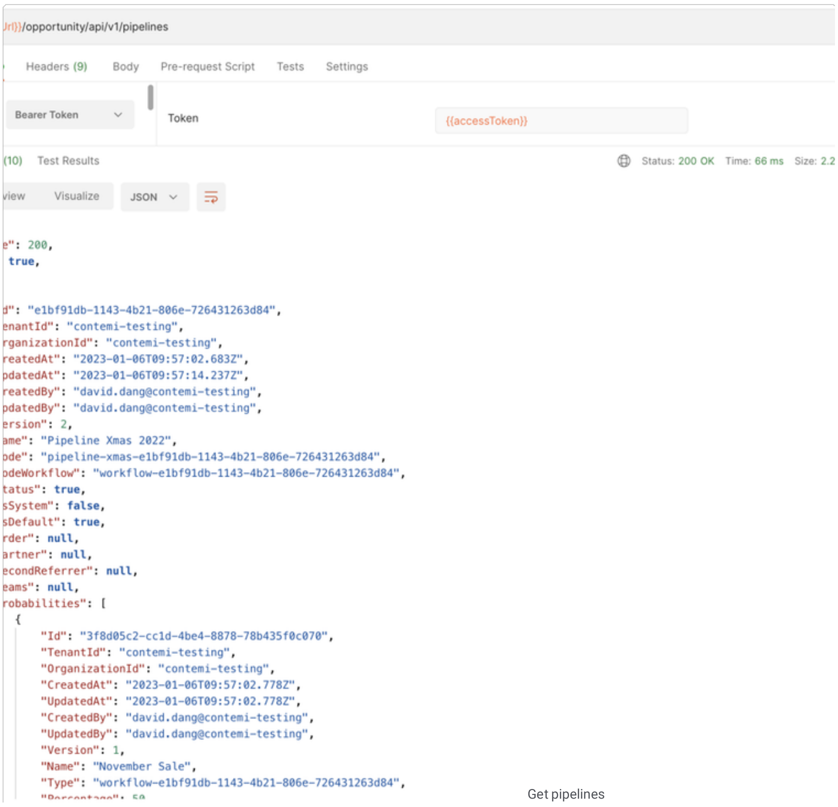
API documentation: https://external-api.seamless.insure/opportunity/documentation/

1. Get pipelines and probabilities

- URL: https://external-api.seamless.insure/opportunity/api/v1/pipelines
- Method: GET
- Header:

  Authorization: Bearer {{accessToken}}

- Postman example (please see it on the attachments):



Token {{accessToken}}

Status: 200 OK Time: 66 ms Size: 2.2

```
e": 200,
 true,

d": "e1bf91db-1143-4b21-806e-726431263d84",
enantId": "contemi-testing",
rganizationId": "contemi-testing",
reatedAt": "2023-01-06T09:57:02.683Z",
pdatedAt": "2023-01-06T09:57:14.237Z",
reatedBy": "david.dang@contemi-testing",
pdatedBy": "david.dang@contemi-testing",
ersion": 2,
ame": "Pipeline Xmas 2022",
ode": "pipeline-xmas-e1bf91db-1143-4b21-806e-726431263d84",
odeWorkflow": "workflow-e1bf91db-1143-4b21-806e-726431263d84",
tatus": true,
sSystem": false,
sDefault": true,
rder": null,
artner": null,
econdReferrer": null,
eams": null,
robabilities": [
  {
      "Id": "3f8d05c2-cc1d-4be4-8878-78b435f0c070",
      "TenantId": "contemi-testing",
      "OrganizationId": "contemi-testing",
      "CreatedAt": "2023-01-06T09:57:02.778Z",
      "UpdatedAt": "2023-01-06T09:57:02.778Z",
      "CreatedBy": "david.dang@contemi-testing",
      "UpdatedBy": "david.dang@contemi-testing",
      "Version": 1,
      "Name": "November Sale",
      "Type": "workflow-e1bf91db-1143-4b21-806e-726431263d84",
      "Percentage": 50
```

Get pipelines

2. Create a sale

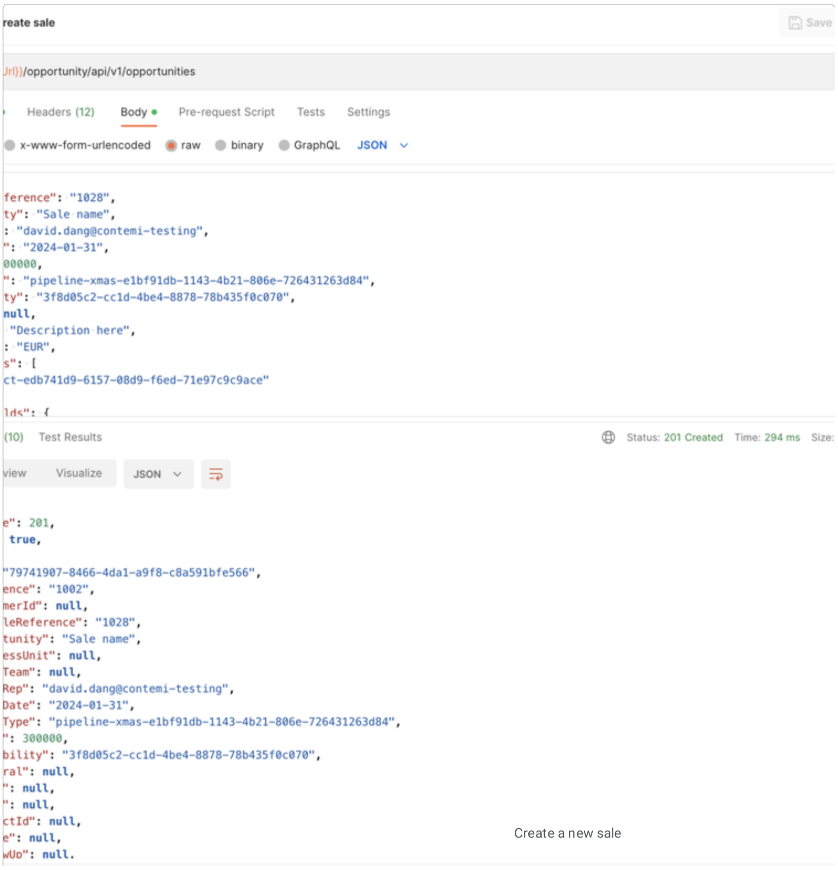
- URL: https://external-api.seamless.insure/opportunity/api/v1/opportunities
- Method: POST
- Header:

  Content-Type: application/json
  Authorization: Bearer {{accessToken}}
  seamless-as-user: (Seamless action username)

- Body: (please see it on the Postman example)
- Fields explanation:

SalesRep: Seamless username (example: user.one@contemi-testing)
SalesType: pipeline code (Use pipeline API to get it, see section 1)
Probability: probability id of selected pipeline
ProfileReference: customer profile reference (Use customer API to get it)
ProductIds: product id list (see Postman example)
SalesRelatedRecords: quote/policy data (see it on the Postman example)

- Postman example (please see it on the attachments):



```
ference": "1028",
ty": "Sale name",
: "david.dang@contemi-testing",
": "2024-01-31",
00000,
": "pipeline-xmas-e1bf91db-1143-4b21-806e-726431263d84",
ty": "3f8d05c2-cc1d-4be4-8878-78b435f0c070",
null,
 "Description here",
: "EUR",
s": [
ct-edb741d9-6157-08d9-f6ed-71e97c9c9ace"

lds": {
```

```
e": 201,
 true,

"79741907-8466-4da1-a9f8-c8a591bfe566",
ence": "1002",
merId": null,
leReference": "1028",
tunity": "Sale name",
essUnit": null,
Team": null,
Rep": "david.dang@contemi-testing",
Date": "2024-01-31",
Type": "pipeline-xmas-e1bf91db-1143-4b21-806e-726431263d84",
": 300000,
bility": "3f8d05c2-cc1d-4be4-8878-78b435f0c070",
ral": null,
": null,
": null,
ctId": null,
e": null,
wUp": null.
```

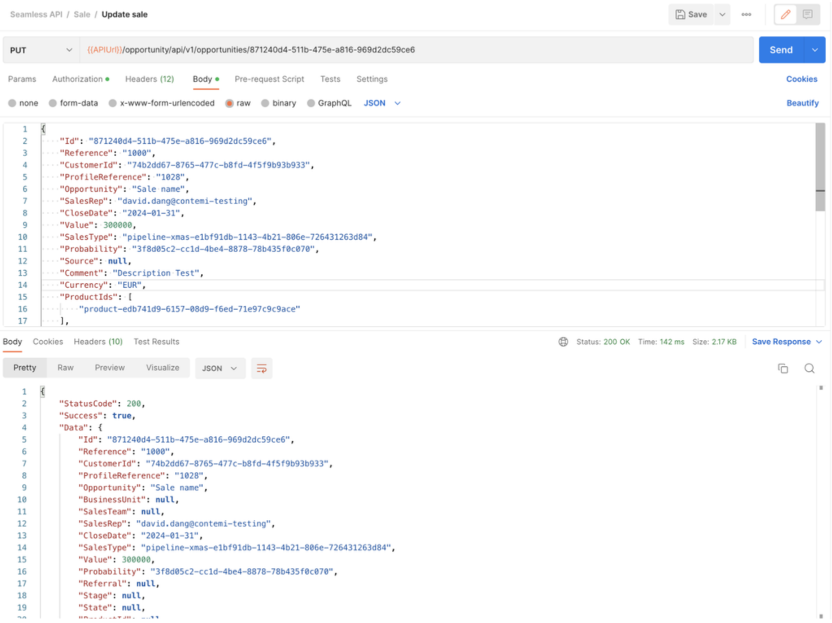
Create a new sale

3. Update a sale

- URL: https://external-api.seamless.insure/opportunity/api/v1/opportunities/{{saleId}} (We can replace *{{saleId}}* with Id of a sale on Seamless)
- Method: PUT
- Header:

  Content-Type: application/json
  Authorization: Bearer {{accessToken}}
  seamless-as-user: (Seamless action username)

- Body: (please see it on the Postman example)
- Postman example (please see it on the attachments):



PUT {{APIUrl}}/opportunity/api/v1/opportunities/871240d4-511b-475e-a816-969d2dc59ce6

○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
{
    "Id": "871240d4-511b-475e-a816-969d2dc59ce6",
    "Reference": "1000",
    "CustomerId": "74b2dd67-8765-477c-b8fd-4f5f9b93b933",
    "ProfileReference": "1028",
    "Opportunity": "Sale name",
    "SalesRep": "david.dang@contemi-testing",
    "CloseDate": "2024-01-31",
    "Value": 300000,
    "SalesType": "pipeline-xmas-e1bf91db-1143-4b21-806e-726431263d84",
    "Probability": "3f8d05c2-cc1d-4be4-8878-78b435f0c070",
    "Source": null,
    "Comment": "Description Test",
    "Currency": "EUR",
    "ProductIds": [
        "product-edb741d9-6157-08d9-f6ed-71e97c9c9ace"
    ],
```

Status: 200 OK Time: 142 ms Size: 2.17 KB

JSON

```
{
    "StatusCode": 200,
    "Success": true,
    "Data": {
        "Id": "871240d4-511b-475e-a816-969d2dc59ce6",
        "Reference": "1000",
        "CustomerId": "74b2dd67-8765-477c-b8fd-4f5f9b93b933",
        "ProfileReference": "1028",
        "Opportunity": "Sale name",
        "BusinessUnit": null,
        "SalesTeam": null,
        "SalesRep": "david.dang@contemi-testing",
        "CloseDate": "2024-01-31",
        "SalesType": "pipeline-xmas-e1bf91db-1143-4b21-806e-726431263d84",
        "Value": 300000,
        "Probability": "3f8d05c2-cc1d-4be4-8878-78b435f0c070",
        "Referral": null,
        "Stage": null,
        "State": null,
```

Update a sale

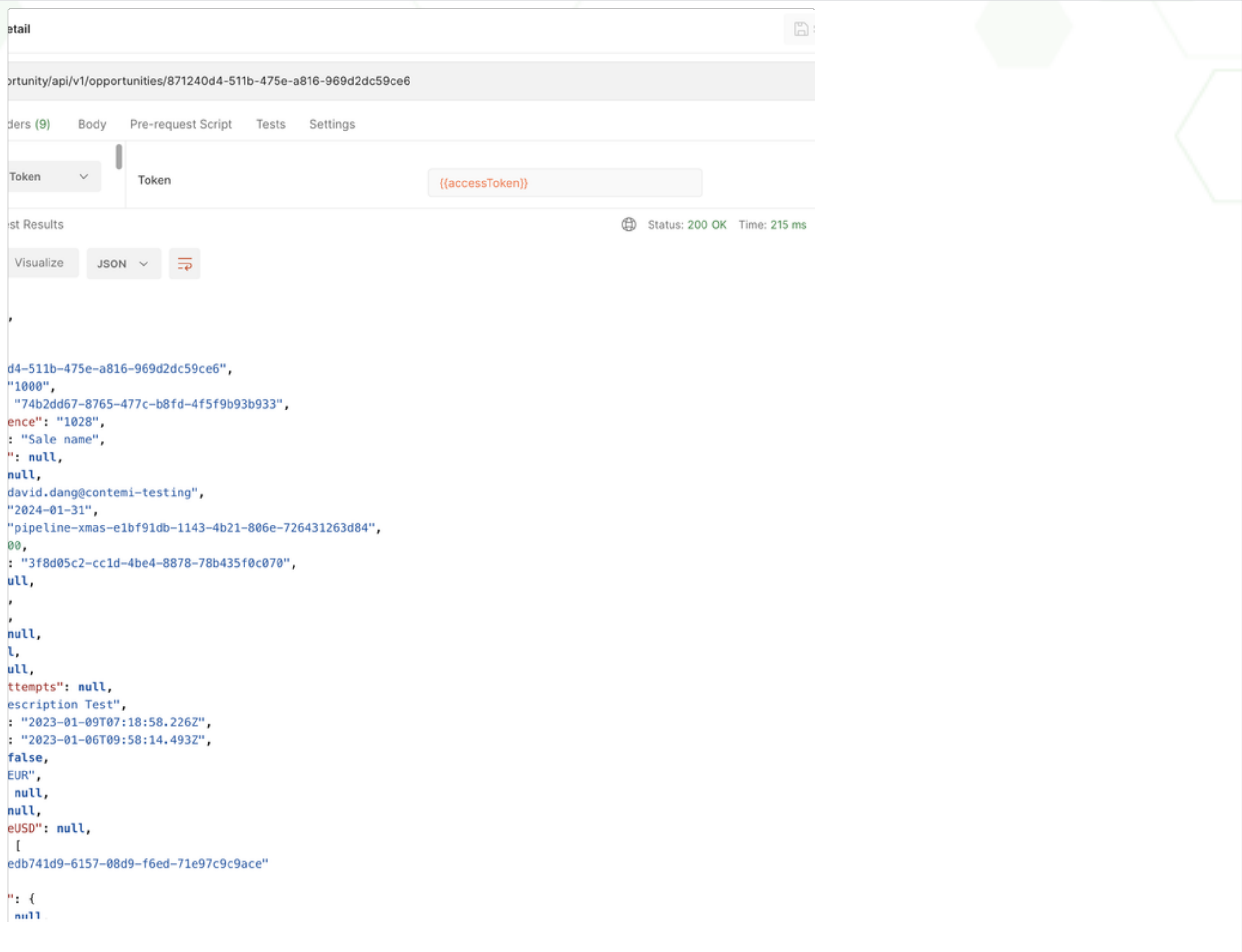
4. Get sale detail
    - URL:

    https://external-api.seamless.insure/opportunity/api/v1/opportunities/{{saleId}}

    - Method: GET
    - Header:

        Authorization: Bearer {{accessToken}}

- Postman example (please see it on the attachments):



```
,


d4-511b-475e-a816-969d2dc59ce6",
"1000",
 "74b2dd67-8765-477c-b8fd-4f5f9b93b933",
ence": "1028",
: "Sale name",
": null,
null,
david.dang@contemi-testing",
"2024-01-31",
"pipeline-xmas-e1bf91db-1143-4b21-806e-726431263d84",
00,
: "3f8d05c2-cc1d-4be4-8878-78b435f0c070",
ull,
,
,
null,
l,
ull,
ttempts": null,
escription Test",
: "2023-01-09T07:18:58.226Z",
: "2023-01-06T09:58:14.493Z",
false,
EUR",
 null,
null,
eUSD": null,
 [
edb741d9-6157-08d9-f6ed-71e97c9c9ace"

": {
 null
```

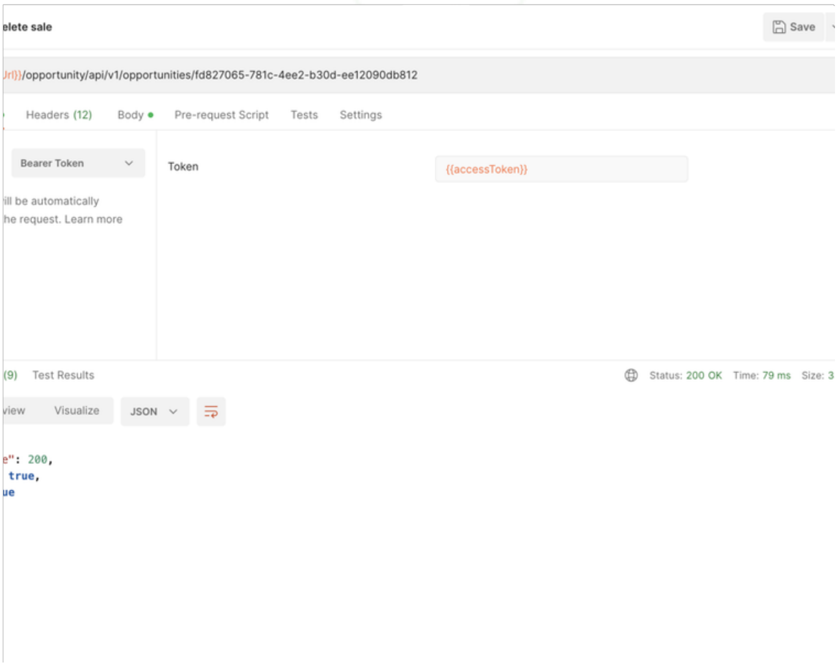
Get sale detail

5. Delete a sale

- URL: https://external-api.seamless.insure/opportunity/api/v1/opportunities/{{saleId}}
- Method: DELETE
- Header:

  Content-Type: application/json
  Authorization: Bearer {{accessToken}}

- Postman example (please see it on the attachments):



```
Jrl}}/opportunity/api/v1/opportunities/fd827065-781c-4ee2-b30d-ee12090db812
```

Token {{accessToken}}

ill be automatically
he request. Learn more

(9)

Status: 200 OK Time: 79 ms Size: 3

```
e": 200,
 true,
ue
```

Delete a sale

Case

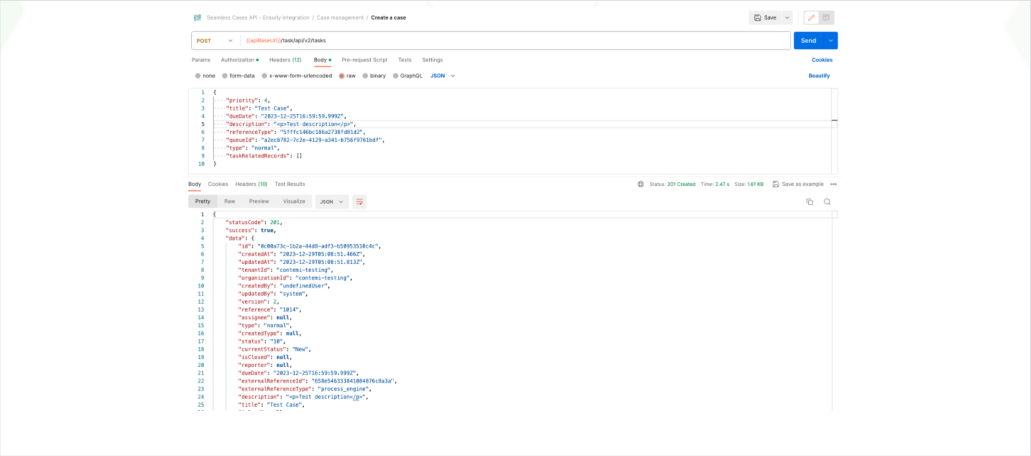
- Postman: Postman_example
- Fields specs: Case_fields.specs.xlsx
- Swagger: Swagger UI

1. Create a case

URL	https://external-api.seamless.insure/task/api/v2/tasks
Method	POST
Headers	**Content-Type**: application/json **Authorization**: Bearer {{*accessToken*}}

Body	{
	"priority": 4,
	"title": "Test Case",
	"dueDate": "2023-12-25T16:59:59.999Z",
	"description": "<p>Test description</p>",
	"referenceType": {{*workflow-id*}},
	"queueId": {{*queue-id*}},
	"type": {{*case-type-value*}},
	"taskRelatedRecords": []
	}
	• These values are example, please see the sample request in Postman file and fields specs in the sheet file for more details.
	• See priority options in the specs file.
	• We must make a request to **Get case types** API, we have response data ($ is at root):
	Replace {{*workflow-id*}} by $.data.results[].workflowId of the response data.
	Replace {{*queue-id*}} by $.data.results[].queueId of the response data.
	Replace {{*case-type-value*}} by $.data.results[].value of the response data.

Response data	Please see the sample response in Postman file and fields specs in the sheet file



POST {{apiBaseUrl}}/task/api/v2/tasks

○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

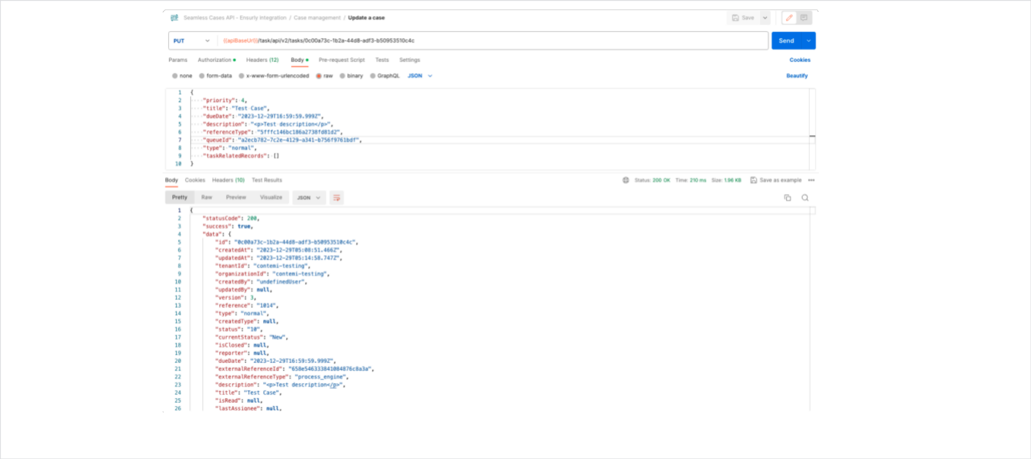
```
{
    "priority": 4,
    "title": "Test Case",
    "dueDate": "2023-12-25T16:59:59.999Z",
    "description": "<p>Test description</p>",
    "referenceType": "5fffc146bc186a2738fd81d2",
    "queueId": "a2ecb782-7c2e-4129-a341-b756f9761bdf",
    "type": "normal",
    "taskRelatedRecords": []
}
```

Status: 201 Created Time: 2.47 s Size: 1.61 KB

```
{
    "statusCode": 201,
    "success": true,
    "data": {
        "id": "0c00a73c-1b2a-44d8-adf3-b50953510c4c",
        "createdAt": "2023-12-29T05:08:51.466Z",
        "updatedAt": "2023-12-29T05:08:51.813Z",
        "tenantId": "contemi-testing",
        "organizationId": "contemi-testing",
        "createdBy": "undefinedUser",
        "updatedBy": "system",
        "version": 2,
        "reference": "1014",
        "assignee": null,
        "type": "normal",
        "createdType": null,
        "status": "10",
        "currentStatus": "New",
        "isClosed": null,
        "reporter": null,
        "dueDate": "2023-12-25T16:59:59.999Z",
        "externalReferenceId": "658e546333841084876c8a3a",
        "externalReferenceType": "process_engine",
        "description": "<p>Test description</p>",
        "title": "Test Case",
```

2. Update a case

URL	https://external-api.seamless.insure/task/api/v2/tasks/{_{id}} (Please replace *{{id}}* with guid Id of Case on Seamless)
Method	PUT
Headers	**Content-Type**: application/json **Authorization**: Bearer {{*accessToken*}}
Body	Same with Body of Create a case section
Response data	Please see the sample response in Postman file and fields specs in the sheet file



PUT {{apiBaseUrl}}/task/api/v2/tasks/0c00a73c-1b2a-44d8-adf3-b50953510c4c

○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
{
    "priority": 4,
    "title": "Test Case",
    "dueDate": "2023-12-29T16:59:59.999Z",
    "description": "<p>Test description</p>",
    "referenceType": "5fffc146bc186a2738fd81d2",
    "queueId": "a2ecb782-7c2e-4129-a341-b756f9761bdf",
    "type": "normal",
    "taskRelatedRecords": []
}
```

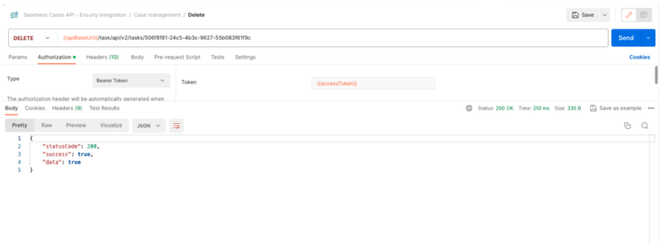
Status: 200 OK Time: 210 ms Size: 1.96 KB

```
{
    "statusCode": 200,
    "success": true,
    "data": {
        "id": "0c00a73c-1b2a-44d8-adf3-b50953510c4c",
        "createdAt": "2023-12-29T05:08:51.466Z",
        "updatedAt": "2023-12-29T05:14:58.747Z",
        "tenantId": "contemi-testing",
        "organizationId": "contemi-testing",
        "createdBy": "undefinedUser",
        "updatedBy": null,
        "version": 3,
        "reference": "1014",
        "type": "normal",
        "createdType": null,
        "status": "10",
        "currentStatus": "New",
        "isClosed": null,
        "reporter": null,
        "dueDate": "2023-12-29T16:59:59.999Z",
        "externalReferenceId": "658e546333841084876c8a3a",
        "externalReferenceType": "process_engine",
        "description": "<p>Test description</p>",
        "title": "Test Case",
        "isRead": null,
        "lastAssignee": null,
```

3. Delete case

URL	https://external-api.seamless.insure/task/api/v2/tasks/{_{id}} (Please replace *{{id}}* with guid Id of Case on Seamless)
Method	DELETE

Headers	Content-Type: application/json Authorization: Bearer {{accessToken}}
Response data	Success of failure status code

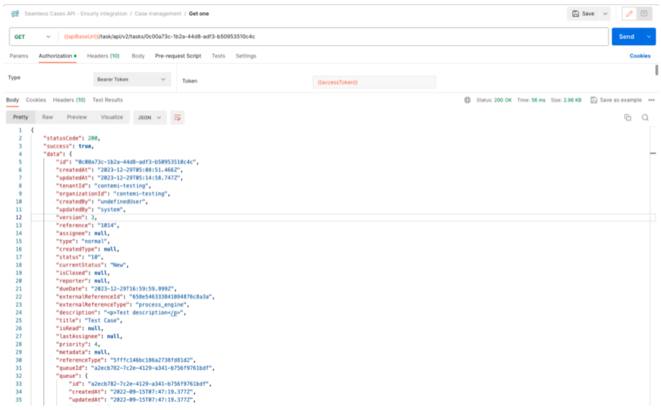


```
DELETE {{apiBaseUrl}}/task/api/v2/tasks/506f8f81-24c5-4b3c-9627-55b083f61f9c
```

```
{
    "statusCode": 200,
    "success": true,
    "data": true
}
```

4. Get case details

URL	https://external-api.seamless.insure/task/api/v2/tasks/{_{id}} _ _ (Please replace *{{id}}* with Guid Id of Case on Seamless)
Method	GET
Headers	**Content-Type**: application/json **Authorization**: Bearer {{*accessToken*}}
Response data	Please see the sample response in Postman file and fields specs in the sheet file



```
GET {{apiBaseUrl}}/task/api/v2/tasks/0c00a73c-1b2a-44d8-adf3-b50953510c4c
```

```
{
    "statusCode": 200,
    "success": true,
    "data": {
        "id": "0c00a73c-1b2a-44d8-adf3-b50953510c4c",
        "createdAt": "2023-12-29T05:08:51.466Z",
        "updatedAt": "2023-12-29T05:14:58.747Z",
        "tenantId": "contemi-testing",
        "organizationId": "contemi-testing",
        "createdBy": "undefinedUser",
        "updatedBy": "system",
        "version": 3,
        "reference": "1014",
        "assignee": null,
        "type": "normal",
        "createdType": null,
        "status": "10",
        "currentStatus": "New",
        "isClosed": null,
        "reporter": null,
        "dueDate": "2023-12-29T16:59:59.999Z",
        "externalReferenceId": "658e546333841084876c8a3a",
        "externalReferenceType": "process_engine",
        "description": "<p>Test description</p>",
        "title": "Test Case",
        "isRead": null,
        "lastAssignee": null,
        "priority": 4,
        "metadata": null,
        "referenceType": "5fffc146bc186a2738fd81d2",
        "queueId": "a2ecb782-7c2e-4129-a341-b756f9761bdf",
        "queue": {
            "id": "a2ecb782-7c2e-4129-a341-b756f9761bdf",
            "createdAt": "2022-09-15T07:47:19.377Z",
            "updatedAt": "2022-09-15T07:47:19.377Z",
```

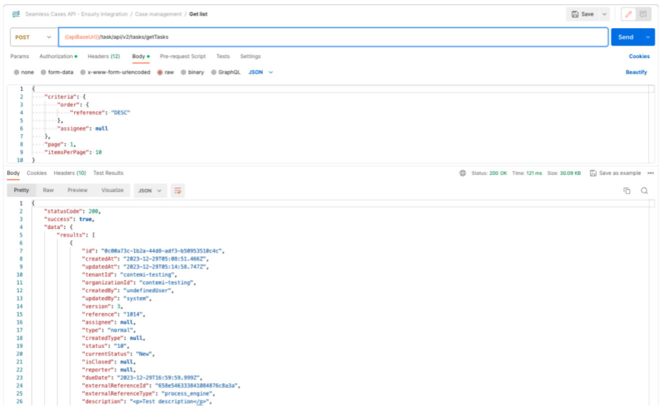
5. Get case list

URL	https://external-api.seamless.insure/task/api/v2/tasks/g_etTasks_
Method	POST

Headers	**Content-Type**: application/json **Authorization**: Bearer {{*accessToken*}}

Request Body	{
	"page": 1,
	"itemsPerPage": 10,
	"criteria": {
	"order": {"reference": "DESC"},
	"assignee": *{{username}}*
	}
	}
	
	- Note: these values are example, we can modify it.
	Order direction value: ASC or DESC
	Replace *{{username}}* with an *username* or *null* (default value is *null*)

Response data	Please see the sample response in Postman file and fields specs in the sheet file



POST {{apiBaseUrl}}/task/api/v2/tasks/getTasks

```
{
    "criteria": {
        "order": {
            "reference": "DESC"
        },
        "assignee": null
    },
    "page": 1,
    "itemsPerPage": 10
}
```

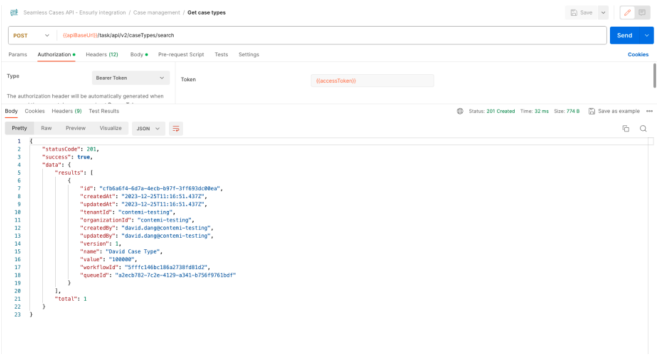
Status: 200 OK Time: 121 ms Size: 30.09 KB

```
{
    "statusCode": 200,
    "success": true,
    "data": {
        "results": [
            {
                "id": "0c00a73c-1b2a-44d8-adf3-b50953510c4c",
                "createdAt": "2023-12-29T05:08:51.466Z",
                "updatedAt": "2023-12-29T05:14:58.747Z",
                "tenantId": "contemi-testing",
                "organizationId": "contemi-testing",
                "createdBy": "undefinedUser",
                "updatedBy": "system",
                "version": 3,
                "reference": "1014",
                "assignee": null,
                "type": "normal",
                "createdType": null,
                "status": "10",
                "currentStatus": "New",
                "isClosed": null,
                "reporter": null,
                "dueDate": "2023-12-29T16:59:59.999Z",
                "externalReferenceId": "658e546333841084876c8a3a",
                "externalReferenceType": "process_engine",
                "description": "<p>Test description</p>",
```

6. Get case types

URL	https://external-api.seamless.insure/task/api/v2/caseTy pes/search
Method	POST
Headers	Content-Type: application/json Authorization: Bearer {{accessToken}}
Request Body	{ "page": 1,

	"itemsPerPage": 10,
	"criteria": {
	"order": {"name": "DESC"}
	}
	}
	Note: these values is example, we can modify it.
	Order direction value: ASC or DESC
Response data	Please see the sample response in Postman file and fields specs in the sheet file



POST {{apiBaseUrl}}/task/api/v2/caseTypes/search

Type Bearer Token

Token {{accessToken}}

The authorization header will be automatically generated when

Status: 201 Created Time: 32 ms Size: 774 B

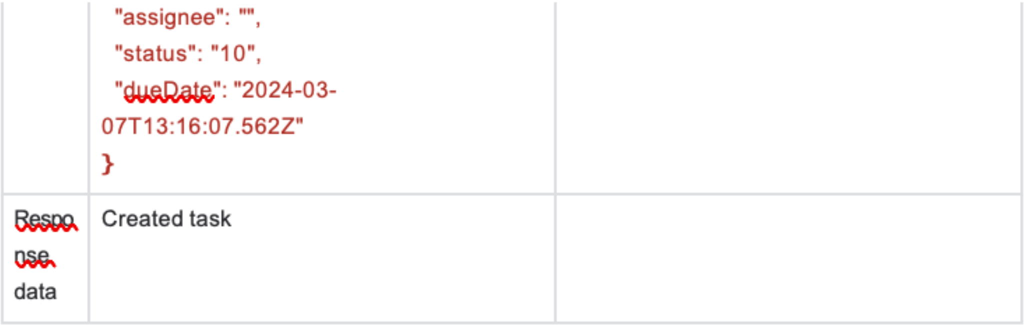
```
{
    "statusCode": 201,
    "success": true,
    "data": {
        "results": [
            {
                "id": "cfb6a6f4-6d7a-4ecb-b97f-3ff693dc00ea",
                "createdAt": "2023-12-25T11:16:51.437Z",
                "updatedAt": "2023-12-25T11:16:51.437Z",
                "tenantId": "contemi-testing",
                "organizationId": "contemi-testing",
                "createdBy": "david.dang@contemi-testing",
                "updatedBy": "david.dang@contemi-testing",
                "version": 1,
                "name": "David Case Type",
                "value": "100000",
                "workflowId": "5fffc146bc186a2738fd81d2",
                "queueId": "a2ecb782-7c2e-4129-a341-b756f9761bdf"
            }
        ],
        "total": 1
    }
}
```

7. Add task to the case

URL	https://external-api.seamless.insure/task/api/v2/tasks/{{caseID}}/subtasks/	
Method	POST	
Headers	**Content-Type**: application/json **Authorization**: Bearer {{*accessToken*}}	

Body	{
	  "title": "Checklist Item 1",
	  "description": "Important part of handling the case",
	  "priority": "Low/Normal/High/Critical",
	  "assignee": "",
	  "status": "10",
	  "dueDate": "2024-03-07T13:16:07.562Z"
	}	

Response data	Created task	



Note

Postman: Postman_example

Swagger: Swagger UI

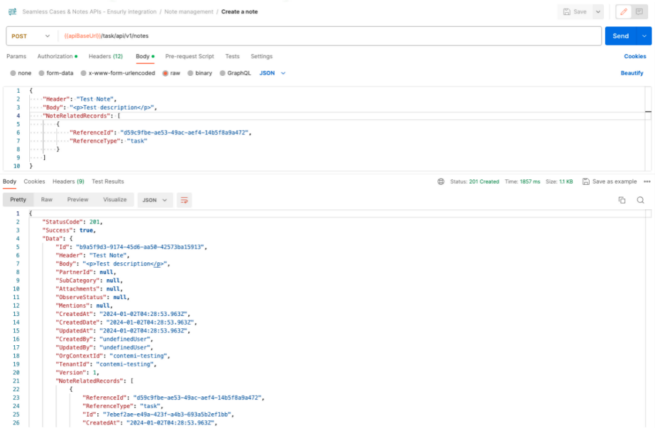
1. Create a note

URL	https://external-api.seamless.insure/task/api/v1/notes
Method	POST

Headers	**Content-Type**: application/json
	Authorization: Bearer {{*accessToken*}}
	seamless-as-user: {{*username*}}

Body	{
	"Header": "Test 1",
	"Body": "<p>Test description</p>",
	"NoteRelatedRecords": [
	{
	"ReferenceId": {{*case-id*}},
	"ReferenceType": "task"
	}
	]
	}
	• This is a example request body, please see the sample request in Postman file.
	• Note:
	Replace {{*case-id*}} by the **Id** of **Case** data (see **Case_API_Guide** PDF file) in order to create note for a case.
	"ReferenceType": default value is "task".

Response data	Please see the sample response in Postman file.



{{apiBaseUrl}}/task/api/v1/notes

```
{
    "Header": "Test Note",
    "Body": "<p>Test description</p>",
    "NoteRelatedRecords": [
        {
            "ReferenceId": "d59c9fbe-ae53-49ac-aef4-14b5f8a9a472",
            "ReferenceType": "task"
        }
    ]
}
```

Status: 201 Created Time: 1857 ms Size: 1.1 KB

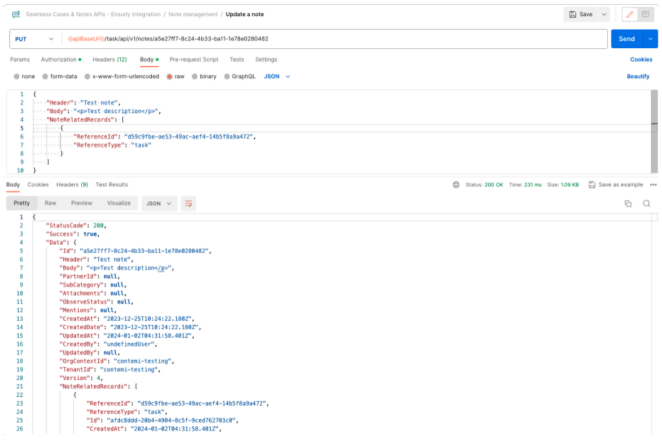
```
{
    "StatusCode": 201,
    "Success": true,
    "Data": {
        "Id": "b9a5f9d3-9174-45d6-aa50-42573ba15913",
        "Header": "Test Note",
        "Body": "<p>Test description</p>",
        "PartnerId": null,
        "SubCategory": null,
        "Attachments": null,
        "ObserveStatus": null,
        "Mentions": null,
        "CreatedAt": "2024-01-02T04:28:53.963Z",
        "CreatedDate": "2024-01-02T04:28:53.963Z",
        "UpdatedAt": "2024-01-02T04:28:53.963Z",
        "CreatedBy": "undefinedUser",
        "UpdatedBy": "undefinedUser",
        "OrgContextId": "contemi-testing",
        "TenantId": "contemi-testing",
        "Version": 1,
        "NoteRelatedRecords": [
            {
                "ReferenceId": "d59c9fbe-ae53-49ac-aef4-14b5f8a9a472",
                "ReferenceType": "task",
                "Id": "7ebef2ae-e49a-423f-a4b3-693a5b2ef1bb",
                "CreatedAt": "2024-01-02T04:28:53.963Z",
```

2. Update a note

URL	https://external-api.seamless.insure/task/api/v1/notes/{{id}} (Please replace *{{id}}* with guid Id of Note on Seamless)
Method	PUT

Headers	**Content-Type**: application/json
	Authorization: Bearer {{*accessToken*}}
	seamless-as-user: {{*username*}}

Body	Same with Body of Create a note section
Response data	Please see the sample response in Postman file.



{{apiBaseUrl}}/task/api/v1/notes/a5e27ff7-8c24-4b33-ba11-1e78e0280482

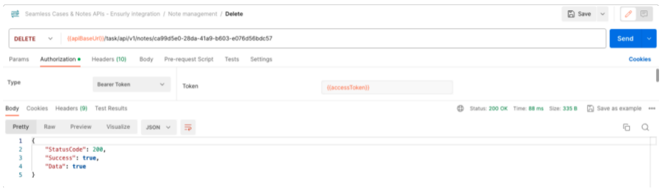
```
{
    "Header": "Test note",
    "Body": "<p>Test description</p>",
    "NoteRelatedRecords": [
        {
            "ReferenceId": "d59c9fbe-ae53-49ac-aef4-14b5f8a9a472",
            "ReferenceType": "task"
        }
    ]
}
```

Status: 200 OK Time: 231 ms Size: 1.09 KB

```
{
    "StatusCode": 200,
    "Success": true,
    "Data": {
        "Id": "a5e27ff7-8c24-4b33-ba11-1e78e0280482",
        "Header": "Test note",
        "Body": "<p>Test description</p>",
        "PartnerId": null,
        "SubCategory": null,
        "Attachments": null,
        "ObserveStatus": null,
        "Mentions": null,
        "CreatedAt": "2023-12-25T10:24:22.180Z",
        "CreatedDate": "2023-12-25T10:24:22.180Z",
        "UpdatedAt": "2024-01-02T04:31:58.401Z",
        "CreatedBy": "undefinedUser",
        "UpdatedBy": null,
        "OrgContextId": "contemi-testing",
        "TenantId": "contemi-testing",
        "Version": 4,
        "NoteRelatedRecords": [
            {
                "ReferenceId": "d59c9fbe-ae53-49ac-aef4-14b5f8a9a472",
                "ReferenceType": "task",
                "Id": "afdc8ddd-20b4-4904-8c5f-9ced762703c0",
                "CreatedAt": "2024-01-02T04:31:58.401Z",
```

3. Delete a note

URL	https://external-api.seamless.insure/task/api/v1/notes/{{id}} (Please replace *{{id}}* with guid Id of Note on Seamless)
Method	DELETE
Headers	**Content-Type**: application/json **Authorization**: Bearer {{*accessToken*}}
Response data	Success of failure status code



DELETE {{apiBaseUrl}}/task/api/v1/notes/ca99d5e0-28da-41a9-b603-e076d56bdc57

Type: Bearer Token

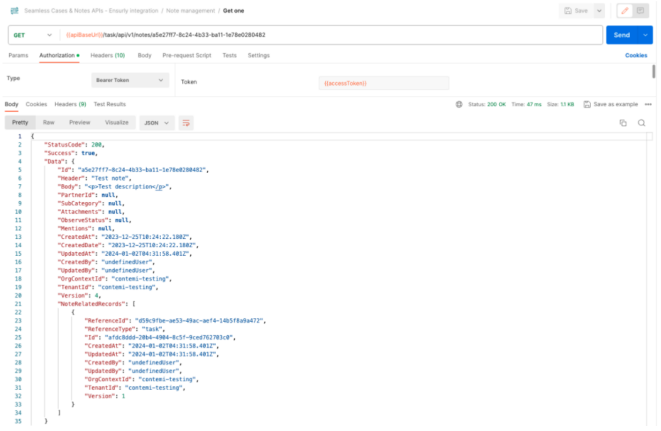
Token: {{accessToken}}

Status: 200 OK Time: 88 ms Size: 335 B

```
{
    "StatusCode": 200,
    "Success": true,
    "Data": true
}
```

4. Get note details

URL	https://external-api.seamless.insure/task/api/v1/notes/{{id}} (Please replace *{{id}}* with guid Id of Note on Seamless)
Method	GET
Headers	**Content-Type**: application/json **Authorization**: Bearer {{*accessToken*}}
Response data	Please see the sample response in Postman file



GET {{apiBaseUrl}}/task/api/v1/notes/a5e27ff7-8c24-4b33-ba11-1e78e0280482

Type: Bearer Token

Token: {{accessToken}}

Status: 200 OK Time: 47 ms Size: 1.1 KB

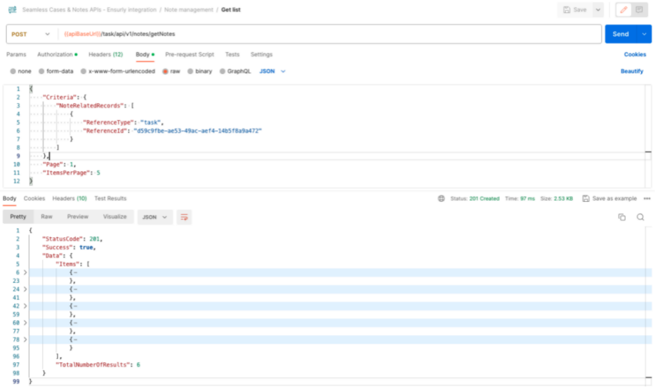
```
{
    "StatusCode": 200,
    "Success": true,
    "Data": {
        "Id": "a5e27ff7-8c24-4b33-ba11-1e78e0280482",
        "Header": "Test note",
        "Body": "<p>Test description</p>",
        "PartnerId": null,
        "SubCategory": null,
        "Attachments": null,
        "ObserveStatus": null,
        "Mentions": null,
        "CreatedAt": "2023-12-25T10:24:22.180Z",
        "CreatedDate": "2023-12-25T10:24:22.180Z",
        "UpdatedAt": "2024-01-02T04:31:58.401Z",
        "CreatedBy": "undefinedUser",
        "UpdatedBy": "undefinedUser",
        "OrgContextId": "contemi-testing",
        "TenantId": "contemi-testing",
        "Version": 4,
        "NoteRelatedRecords": [
            {
                "ReferenceId": "d59c9fbe-ae53-49ac-aef4-14b5f8a9a472",
                "ReferenceType": "task",
                "Id": "afdc8ddd-20b4-4904-8c5f-9ced762703c0",
                "CreatedAt": "2024-01-02T04:31:58.401Z",
                "UpdatedAt": "2024-01-02T04:31:58.401Z",
                "CreatedBy": "undefinedUser",
                "UpdatedBy": "undefinedUser",
                "OrgContextId": "contemi-testing",
                "TenantId": "contemi-testing",
                "Version": 1
            }
        ]
    }
}
```

5. Get note list

URL	https://external-api.seamless.insure/task/api/v1/notes/getNotes
Method	POST
Headers	**Content-Type**: application/json **Authorization**: Bearer {{*accessToken*}}

Request Body	{
	"Criteria": {
	"NoteRelatedRecords": [
	{
	"ReferenceType": "task",
	"ReferenceId": {{*case-id*}}
	}
	]
	},
	"Page": 1,
	"ItemsPerPage": 10
	}
	- Please see the sample request in Postman file.
	- Note:
	Replace {{*case-id*}} by the **Id** of **Case** data (see **Case_API_Guide** PDF file) in order to create note for a case

Response data	Please see the sample response in Postman file



POST {{apiBaseUrl}}/task/api/v1/notes/getNotes

```
{
    "Criteria": {
        "NoteRelatedRecords": [
            {
                "ReferenceType": "task",
                "ReferenceId": "d59c9fbe-ae53-49ac-aef4-14b5f8a9a472"
            }
        ]
    },
    "Page": 1,
    "ItemsPerPage": 5
}
```

Status: 201 Created Time: 97 ms Size: 2.53 KB

```
{
    "StatusCode": 201,
    "Success": true,
    "Data": {
        "Items": [
            {…
            },
            {…
            },
            {…
            },
            {…
            },
            {…
            }
        ],
        "TotalNumberOfResults": 6
    }
}
```

Documents Service

This page is a supplement for the Documents service Open API documentation. It provides supplementary information that is not included in the endpoint documentation on Swagger UI. It also provides documentation for endpoints that are not documented on the Swagger UI.

Atomic Operations

Documents service supports JSON:API Atomic Operations. The current implementation limits to operations on `documents` and `documentMetadataItems` resource types only. For details of authorization requirements for using atomic operations on supporting resource types, please refer to the following table.

Resource Type	Required Permission
`documents`	Documents_ManageDocuments
`documentMetadataItems`	Documents_ManageDocuments

For samples of atomic operations that are currently supported by this service, please see the attached file. sample-documents-atomic-operations.http

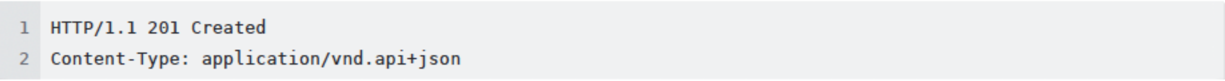
Upload Documents

Upload documents to Seamless

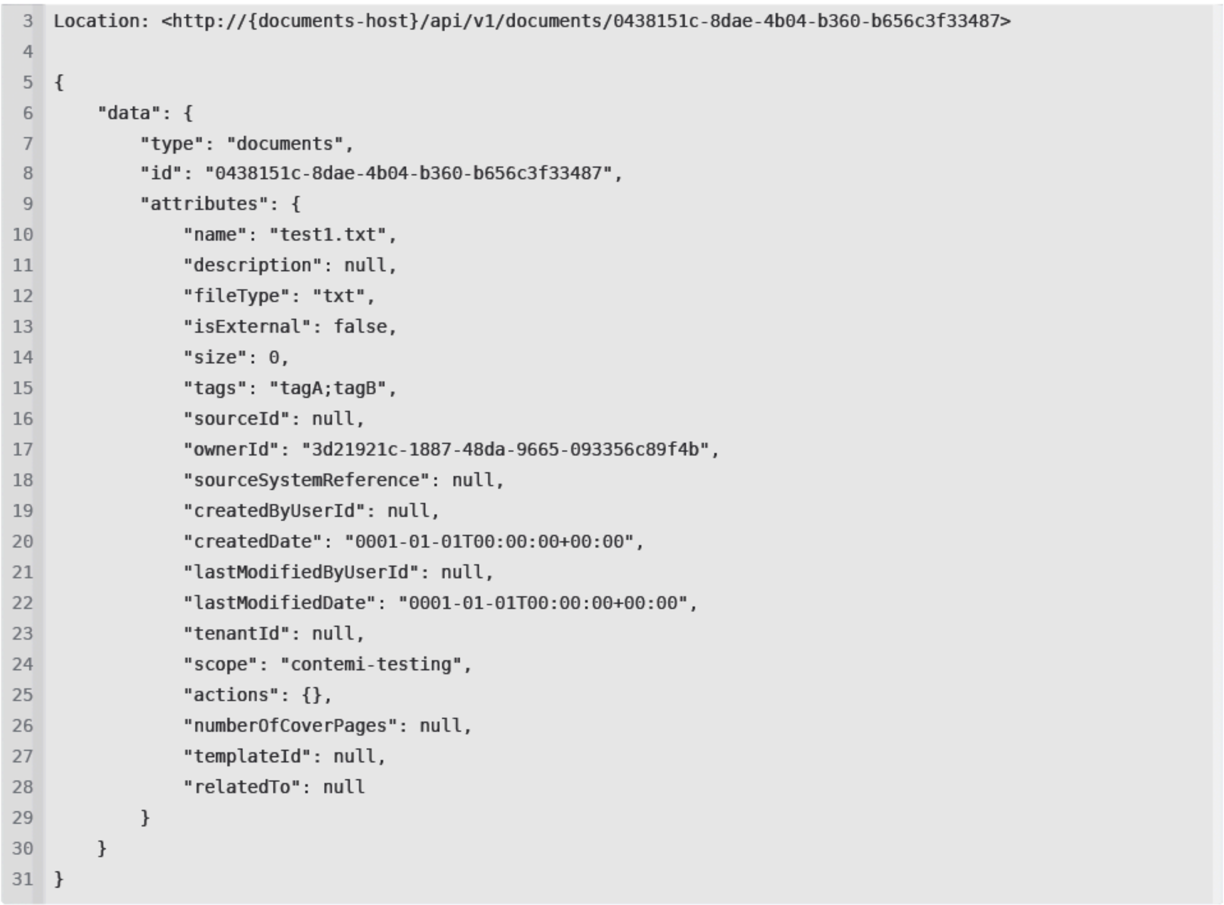
- Steps
    - Create a document record:
      Request:

```
POST '<https://{{host-documents}}/api/v1/documents>'
Content-Type: application/vnd.api+json

{
    "data": {
      "type": "documents",
      "attributes": {
        "name": "test1.txt",
        "description": "My test file",
        "fileType": "txt",
        "tags": [
          "tagA",
          "tagB"
        ]
      }
    }
}
```

Response:

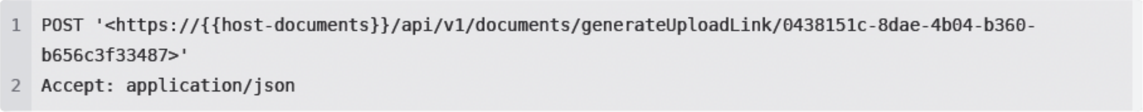


```
HTTP/1.1 201 Created
Content-Type: application/vnd.api+json
```



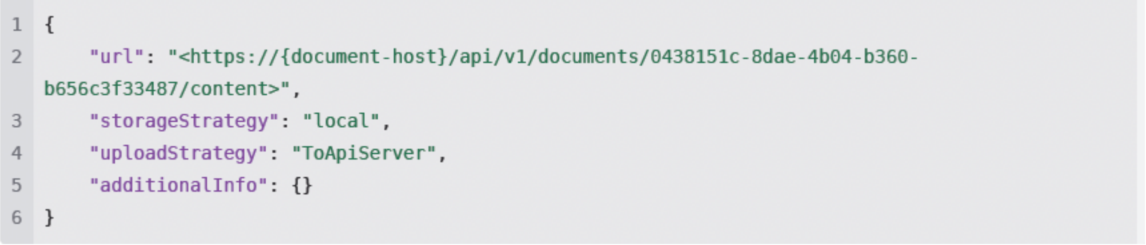
```
Location: <http://{documents-host}/api/v1/documents/0438151c-8dae-4b04-b360-b656c3f33487>

{
    "data": {
        "type": "documents",
        "id": "0438151c-8dae-4b04-b360-b656c3f33487",
        "attributes": {
            "name": "test1.txt",
            "description": null,
            "fileType": "txt",
            "isExternal": false,
            "size": 0,
            "tags": "tagA;tagB",
            "sourceId": null,
            "ownerId": "3d21921c-1887-48da-9665-093356c89f4b",
            "sourceSystemReference": null,
            "createdByUserId": null,
            "createdDate": "0001-01-01T00:00:00+00:00",
            "lastModifiedByUserId": null,
            "lastModifiedDate": "0001-01-01T00:00:00+00:00",
            "tenantId": null,
            "scope": "contemi-testing",
            "actions": {},
            "numberOfCoverPages": null,
            "templateId": null,
            "relatedTo": null
        }
    }
}
```

    - Upload file content to the document record:
- Generate an upload URL:



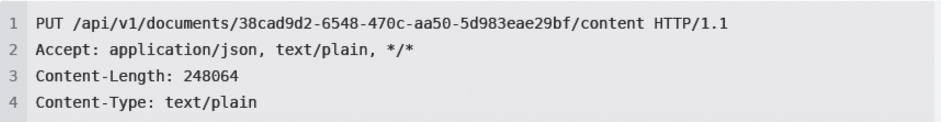
```
POST '<https://{{host-documents}}/api/v1/documents/generateUploadLink/0438151c-8dae-4b04-b360-b656c3f33487>'
Accept: application/json
```

- If the server responds with a `ToApiServer` upload strategy, the client should upload the file content to the server
    - Response



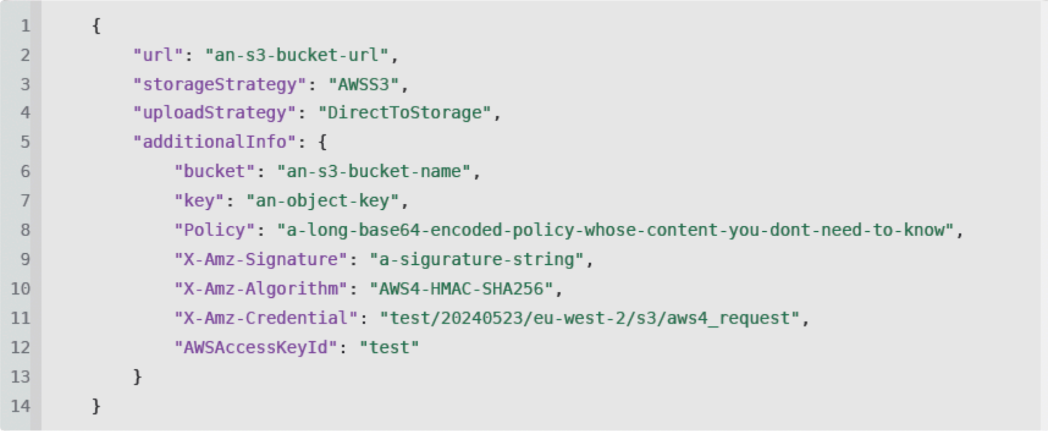
```
{
    "url": "<https://{document-host}/api/v1/documents/0438151c-8dae-4b04-b360-b656c3f33487/content>",
    "storageStrategy": "local",
    "uploadStrategy": "ToApiServer",
    "additionalInfo": {}
}
```

    - Upload the file content to the server



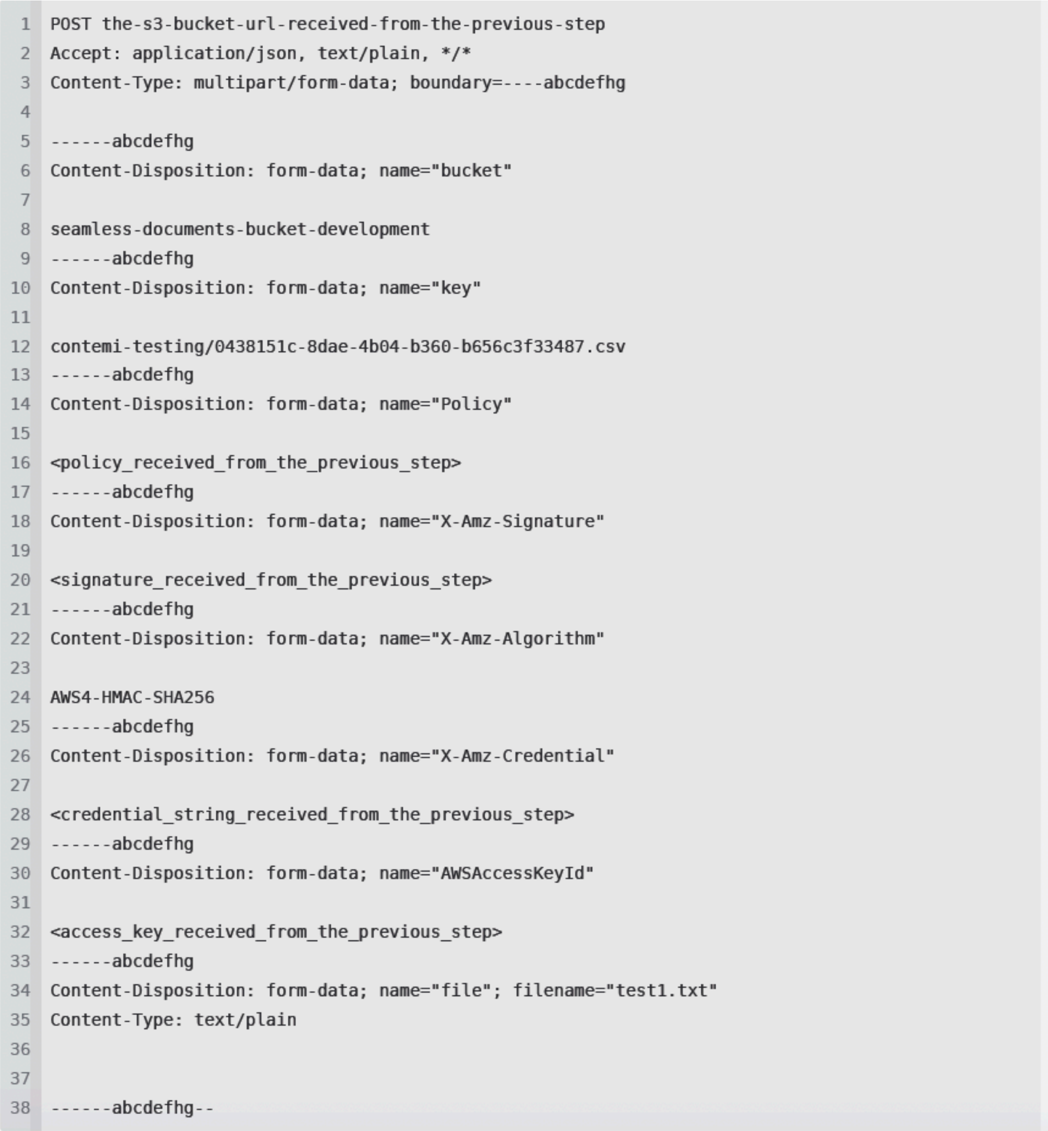
```
PUT /api/v1/documents/38cad9d2-6548-470c-aa50-5d983eae29bf/content HTTP/1.1
Accept: application/json, text/plain, */*
Content-Length: 248064
Content-Type: text/plain
```

- If the server responds with a `DirectToStorage` upload strategy, the client should upload the file content to the storage location (at the time of writing this, there is only AWSS3 storage) directly:
    - Response



```
{
    "url": "an-s3-bucket-url",
    "storageStrategy": "AWSS3",
    "uploadStrategy": "DirectToStorage",
    "additionalInfo": {
        "bucket": "an-s3-bucket-name",
        "key": "an-object-key",
        "Policy": "a-long-base64-encoded-policy-whose-content-you-dont-need-to-know",
        "X-Amz-Signature": "a-sigurature-string",
        "X-Amz-Algorithm": "AWS4-HMAC-SHA256",
        "X-Amz-Credential": "test/20240523/eu-west-2/s3/aws4_request",
        "AWSAccessKeyId": "test"
    }
}
```

    - Upload directly to the S3 bucket



```
POST the-s3-bucket-url-received-from-the-previous-step
Accept: application/json, text/plain, */*
Content-Type: multipart/form-data; boundary=----abcdefhg

------abcdefhg
Content-Disposition: form-data; name="bucket"

seamless-documents-bucket-development
------abcdefhg
Content-Disposition: form-data; name="key"

contemi-testing/0438151c-8dae-4b04-b360-b656c3f33487.csv
------abcdefhg
Content-Disposition: form-data; name="Policy"

<policy_received_from_the_previous_step>
------abcdefhg
Content-Disposition: form-data; name="X-Amz-Signature"

<signature_received_from_the_previous_step>
------abcdefhg
Content-Disposition: form-data; name="X-Amz-Algorithm"

AWS4-HMAC-SHA256
------abcdefhg
Content-Disposition: form-data; name="X-Amz-Credential"

<credential_string_received_from_the_previous_step>
------abcdefhg
Content-Disposition: form-data; name="AWSAccessKeyId"

<access_key_received_from_the_previous_step>
------abcdefhg
Content-Disposition: form-data; name="file"; filename="test1.txt"
Content-Type: text/plain


------abcdefhg--
```

- It is not mandatory to generate an upload URL if you decide to upload the file content to the API server. Howerver, uploading large files is only possible by generating an upload URL and uploading the file content directly to the storage location

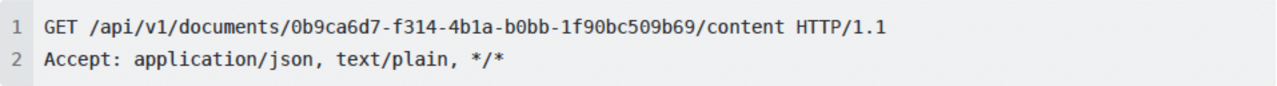
Link a Seameless document record to a real document in an external system

Use: `POST /api/v1/documents/externalLinkDocument`

Download Documents

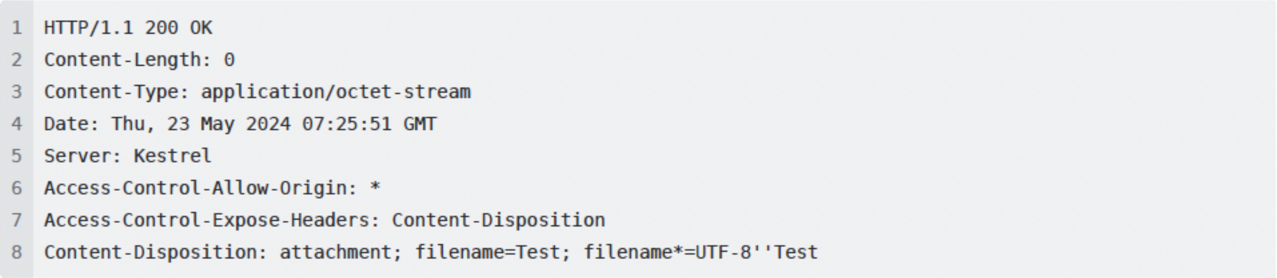
- Download from API server:

Request:



```
GET /api/v1/documents/0b9ca6d7-f314-4b1a-b0bb-1f90bc509b69/content HTTP/1.1
Accept: application/json, text/plain, */*
```

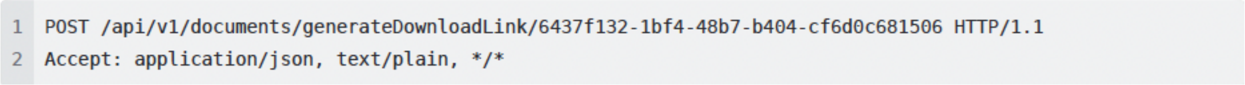
Response:



```
HTTP/1.1 200 OK
Content-Length: 0
Content-Type: application/octet-stream
Date: Thu, 23 May 2024 07:25:51 GMT
Server: Kestrel
Access-Control-Allow-Origin: *
Access-Control-Expose-Headers: Content-Disposition
Content-Disposition: attachment; filename=Test; filename*=UTF-8''Test
```

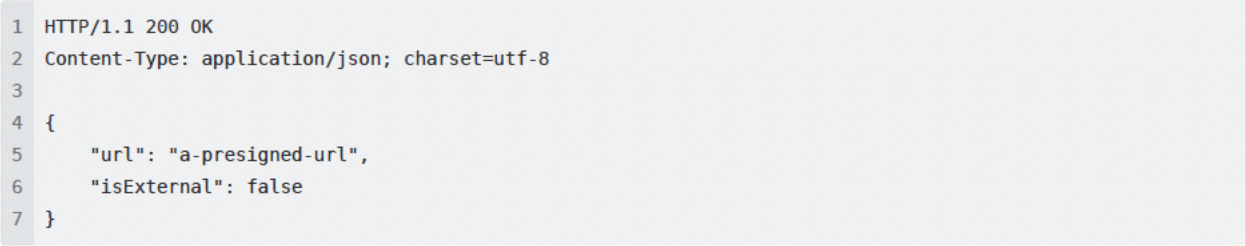
- Download directly from storage location (Neccessary if the file is large in size)
  - Generate download URL

Request:



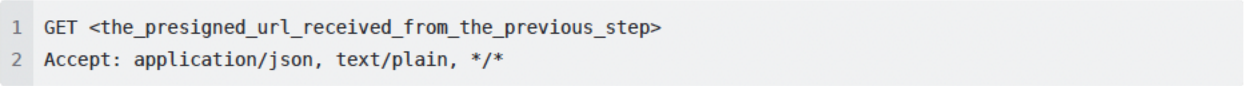
```
POST /api/v1/documents/generateDownloadLink/6437f132-1bf4-48b7-b404-cf6d0c681506 HTTP/1.1
Accept: application/json, text/plain, */*
```

Response:



```
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8

{
    "url": "a-presigned-url",
    "isExternal": false
}
```

  - Download the file from the presigned URL



```
GET <the_presigned_url_received_from_the_previous_step>
Accept: application/json, text/plain, */*
```

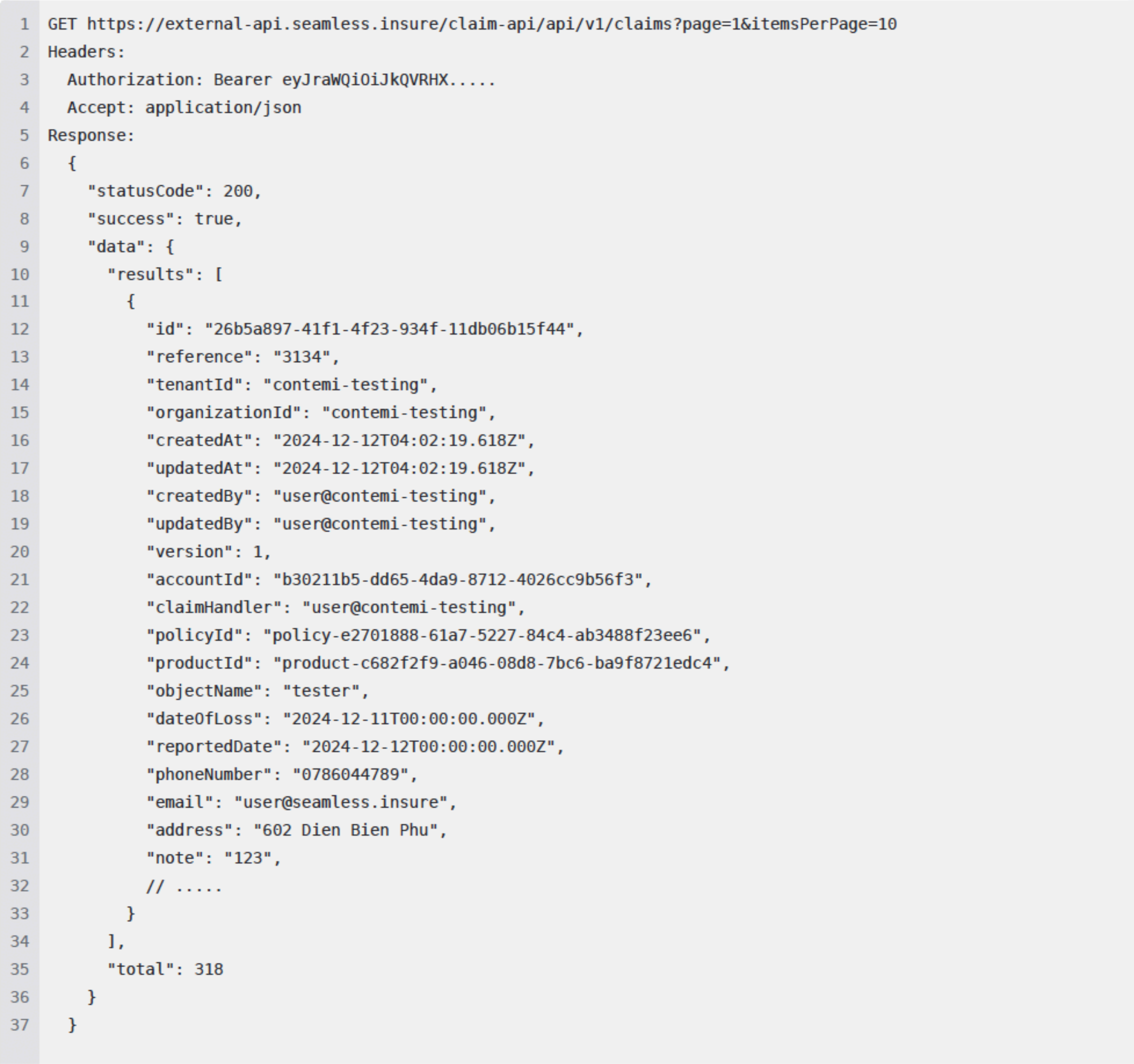
Claim

This page describes how to get a Claims list, Claim details, and FONL.

BaseUrl: external-api.seamless.insure

Get Claims

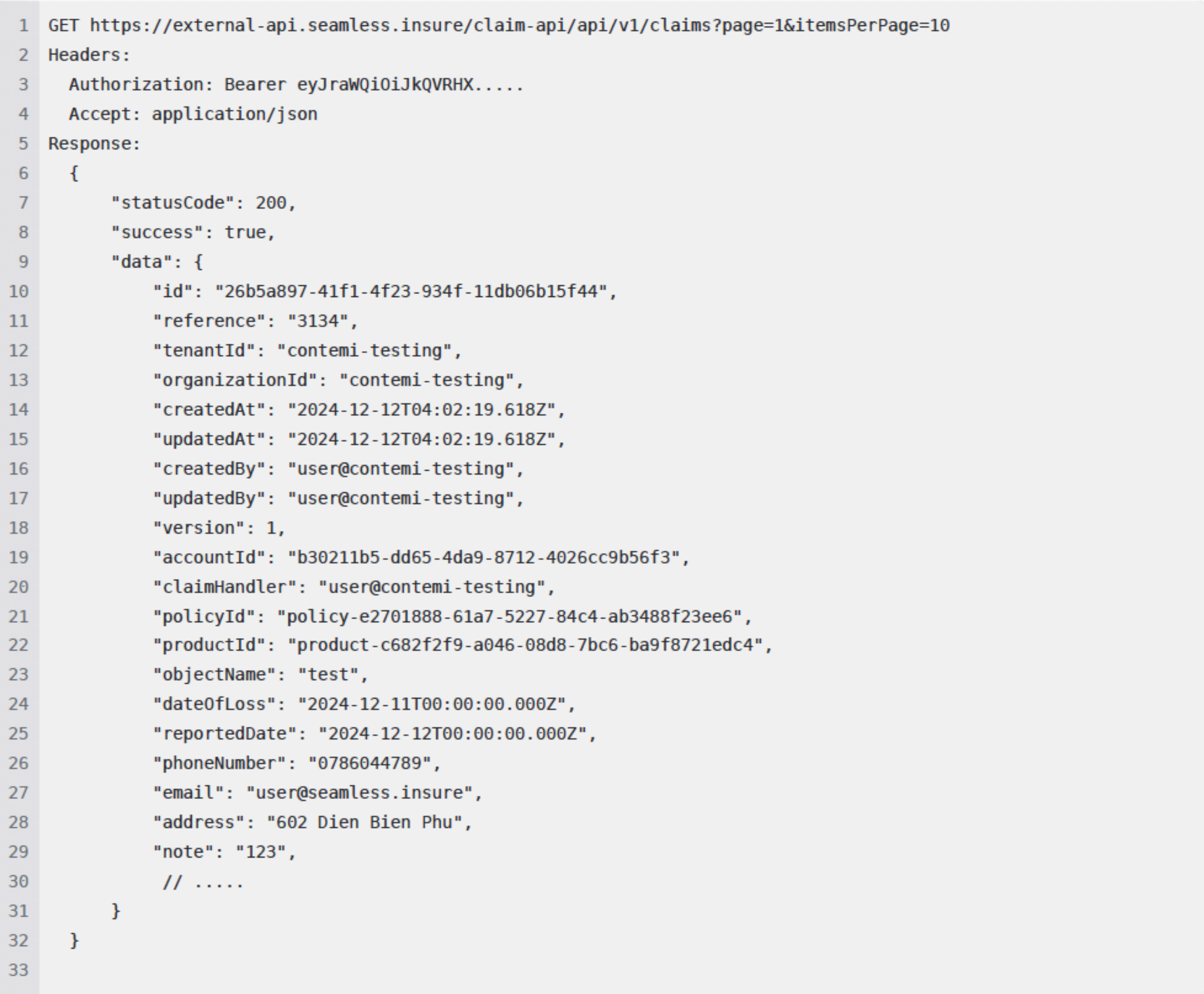
- URL: {BaseUrl}/claim-api/api/v1/claims
- Method: GET
- Query params:
  - page: number
  - itemsPerPage: number
- Response: JSON data
- Sample:



```
GET https://external-api.seamless.insure/claim-api/api/v1/claims?page=1&itemsPerPage=10
Headers:
  Authorization: Bearer eyJraWQiOiJkQVRHX.....
  Accept: application/json
Response:
  {
    "statusCode": 200,
    "success": true,
    "data": {
      "results": [
        {
          "id": "26b5a897-41f1-4f23-934f-11db06b15f44",
          "reference": "3134",
          "tenantId": "contemi-testing",
          "organizationId": "contemi-testing",
          "createdAt": "2024-12-12T04:02:19.618Z",
          "updatedAt": "2024-12-12T04:02:19.618Z",
          "createdBy": "user@contemi-testing",
          "updatedBy": "user@contemi-testing",
          "version": 1,
          "accountId": "b30211b5-dd65-4da9-8712-4026cc9b56f3",
          "claimHandler": "user@contemi-testing",
          "policyId": "policy-e2701888-61a7-5227-84c4-ab3488f23ee6",
          "productId": "product-c682f2f9-a046-08d8-7bc6-ba9f8721edc4",
          "objectName": "tester",
          "dateOfLoss": "2024-12-11T00:00:00.000Z",
          "reportedDate": "2024-12-12T00:00:00.000Z",
          "phoneNumber": "0786044789",
          "email": "user@seamless.insure",
          "address": "602 Dien Bien Phu",
          "note": "123",
          // .....
        }
      ],
      "total": 318
    }
  }
```

Get Claim

- URL: {BaseUrl}/claim-api/api/v1/claims/{uuid}
- Response: JSON data
- Sample:



```
GET https://external-api.seamless.insure/claim-api/api/v1/claims?page=1&itemsPerPage=10
Headers:
  Authorization: Bearer eyJraWQiOiJkQVRHX.....
  Accept: application/json
Response:
  {
      "statusCode": 200,
      "success": true,
      "data": {
          "id": "26b5a897-41f1-4f23-934f-11db06b15f44",
          "reference": "3134",
          "tenantId": "contemi-testing",
          "organizationId": "contemi-testing",
          "createdAt": "2024-12-12T04:02:19.618Z",
          "updatedAt": "2024-12-12T04:02:19.618Z",
          "createdBy": "user@contemi-testing",
          "updatedBy": "user@contemi-testing",
          "version": 1,
          "accountId": "b30211b5-dd65-4da9-8712-4026cc9b56f3",
          "claimHandler": "user@contemi-testing",
          "policyId": "policy-e2701888-61a7-5227-84c4-ab3488f23ee6",
          "productId": "product-c682f2f9-a046-08d8-7bc6-ba9f8721edc4",
          "objectName": "test",
          "dateOfLoss": "2024-12-11T00:00:00.000Z",
          "reportedDate": "2024-12-12T00:00:00.000Z",
          "phoneNumber": "0786044789",
          "email": "user@seamless.insure",
          "address": "602 Dien Bien Phu",
          "note": "123",
           // .....
      }
  }

```

FONL