



SOFTWARESILOS

Customer Documents Extension

PDF Manual

MODULE

customerdocuments

LAST UPDATED

2026-03-29

REST API Reference

Customer Documents Extension · /customerdocuments/addons/api/rest-api

This API requires the optional add-on `mageb2b/customerdocuments-api` (`MageB2B_CustomerDocumentsApi`).

Authentication

Use an Admin/Integration Bearer token.

```
curl -X POST "https://your-store.com/rest/V1/integration/admin/token" \  
-H "Content-Type: application/json" \  
-d '{"username":"admin","password":"***"}'
```

Then call endpoints with:

```
-H "Authorization: Bearer <ADMIN_TOKEN>"
```

Endpoint Overview

Documents

Method	URL
GET	/V1/customerdocuments/:id
GET	/V1/customerdocuments/search
POST	/V1/customerdocuments
PUT	/V1/customerdocuments/:id
DELETE	/V1/customerdocuments/:id

Customer Assignments

Method	URL
GET	/V1/customerdocuments/customer/:id
GET	/V1/customerdocuments/customer/search
POST	/V1/customerdocuments/customer
PUT	/V1/customerdocuments/customer/:id
DELETE	/V1/customerdocuments/customer/:id

Customer Group Assignments

Method	URL
GET	/V1/customerdocuments/customergroup/:id

Method	URL
GET	/V1/customerdocuments/customergroup/search
POST	/V1/customerdocuments/customergroup
PUT	/V1/customerdocuments/customergroup/:id
DELETE	/V1/customerdocuments/customergroup/:id

Product Assignments

Method	URL
GET	/V1/customerdocuments/product/:id
GET	/V1/customerdocuments/product/search
POST	/V1/customerdocuments/product
PUT	/V1/customerdocuments/product/:id
DELETE	/V1/customerdocuments/product/:id
DELETE	/V1/customerdocuments/product/by-product/:productId
DELETE	/V1/customerdocuments/product/by-document/:documentId

Categories

Method	URL
GET	/V1/customerdocuments/category/:id
GET	/V1/customerdocuments/category/search
POST	/V1/customerdocuments/category
PUT	/V1/customerdocuments/category/:id
DELETE	/V1/customerdocuments/category/:id

Links

Method	URL
GET	/V1/customerdocuments/links
POST	/V1/customerdocuments/links
DELETE	/V1/customerdocuments/links/:id

Request Payloads

General Notes

- GET and DELETE routes do not require a request body.
- .../search routes use standard Magento `searchCriteria` query parameters.

Example:

```
/V1/customerdocuments/product/search?searchCriteria[filter_groups][0][filters][0][field]=document_id&searchCriteria[filter_groups][0][filters][0][value]=1777&searchCriteria[filter_groups][0][filters][0][condition_type]=eq&searchCriteria[currentPage]=1&searchCriteria[pageSize]=20
```

POST/PUT /V1/customerdocuments (wrapper: document)

```
{
  "document": {
    "id": 1777,
    "name": "Invoice 1001",
    "description": "January invoice",
    "is_external": 1,
    "external_url": "https://example.com/invoice-1001.pdf",
    "path": "customerdocuments/invoice-1001.pdf",
    "is_public": 0,
    "category_id": 829,
    "website_id": 0,
    "show_in_sidebar": 1,
    "valid_from": "2026-01-01",
    "valid_to": "2026-12-31",
    "customer_ids": [4801],
    "customer_group_ids": [1, 3],
    "document_data": {
      "base_name": "invoice-1001.pdf",
      "base64_data": "data:application/pdf;base64,JVBERi0xLjQK..."
    }
  }
}
```

POST/PUT /V1/customerdocuments/customer (wrapper: documentCustomer)

```
{
  "documentCustomer": {
    "id": 4099,
    "document_id": 1777,
    "customer_id": 4801
  }
}
```

POST/PUT /V1/customerdocuments/customergroup (wrapper: customerGroup)

```
{
  "customerGroup": {
    "id": 3196,
    "document_id": 1777,
  }
}
```

```
"customer_group_id": 3
}
}
```

POST/PUT /V1/customerdocuments/product (wrapper: documentProduct)

```
{
  "documentProduct": {
    "id": 385,
    "document_id": 1777,
    "product_id": 43432,
    "store_id": 0,
    "position": 10
  }
}
```

POST/PUT /V1/customerdocuments/category (wrapper: documentCategory)

```
{
  "documentCategory": {
    "id": 829,
    "name": "Security Policies"
  }
}
```

POST /V1/customerdocuments/links (wrapper: documentLink)

```
{
  "documentLink": {
    "id": 1,
    "customer_document_id": 1777,
    "link_entity_type": "customer",
    "link_entity_id": 4801
  }
}
```

Error Codes

Code	Meaning
200	Request accepted/successful
400	Payload parsing/validation error
401	Unauthorized
403	ACL denied
404	Entity does not exist

Code	Meaning
500	Internal API/service-contract/runtime error