# live API (1.6.1000-161)

Download OpenAPI specification: Download

HTTP interface of the live server.

**Only the `public` section is released for integrations:** sign in with an API key and create, read, list and change assignments. This part is stable – it is only ever extended; nothing goes away and nothing changes its meaning. Everything under `internal` serves the web app and the tools and may change without notice.

## Conventions

- All payloads are JSON (`content-type: application/json`).
- Points in time are milliseconds since 1970 (UTC).
- Errors come as JSON with a fixed `code` (e.g. `policy.missing_right`), `category`, `status` and `message`. Decide on `code`, never on `message`. `401`: key missing or invalid. `403`: the account lacks the right.
- Responses may carry fields not described here; ignore unknown fields.

## 1. Sign in

Every request carries the API key in a header: `x-api-key: live_api_…`. An admin issues the key for an account; it is shown only once and acts with that account's rights.

## 2. Create an assignment

`POST /api/dto/fsg3` with `order`, `subject` and `creditor`. The executing partner is assigned by postal code. The answer names the `assignmentId` and the reference number `ref`.

### 3. Read an assignment

`GET /api/dto/fsg3/{ref}` – complete, with status, dates and results (`results.<service>.<field>`).

### 4. List assignments

`GET /api/dto/fsg3` with a filter (`filter=status=pending`), search (`search=`), order (`sort=subjectDisplayName&dir=asc`) and page size (`limit=`). While the answer carries `nextCursor`, the same request plus `cursor=<nextCursor>` returns the next page.

### 5. Change an assignment

`PATCH /api/dto/fsg3/{ref}` with the fields to change: fields not named stay, `null` clears a field, `{ "status": { "status": "cancelled" } }` changes the status. There is no delete; cancelling is a status change.

The field names are English; each field names its column in the fsg3 format.

This page shows a complete example for each of these endpoints.

## public

The released, stable interface: sign in with an API key and create, read, list and change assignments. It is only ever extended.

## List assignments (FSG, fsg3)

Schritt 4 des Standardablaufs: Aufträge auflisten, jede Zeile in derselben Form wie beim Lesen.

Filter, Sortierung und Seiten lassen sich kombinieren. Die erste Seite liefert `nextCursor`; dieselbe Anfrage mit `cursor=<nextCursor>` liefert die Folgeseite, bis `nextCursor` fehlt.

Beispiel erste Seite: `GET /api/dto/fsg3?filter=status%3Dto_confirm&sort=subjectDisplayName&limit=2` mit dem Kopf `x-api-key`.

AUTHORIZATIONS: › *apiKeyAuth*

QUERY PARAMETERS

**filter**
string
Example: `filter=status=to_confirm`
Filter, wiederholbar: `status=<stand>`, `ref=<aktenzeichen>` oder ein Datumsbereich wie `modified>=<ms>`.

**search**
string
Freitextsuche (Name, Aktenzeichen, Adresse).

**sort**
string
Example: `sort=subjectDisplayName`
Sortierfeld, z. B. `ref`, `subjectDisplayName`, `status`, `editedAt`, `dueDate`. Ohne Angabe nach Auftrags-ID.

**dir**
string
Enum: `"asc"` `"desc"`
`asc` (Vorgabe) oder `desc`.

**limit**
integer `[ 1 .. 1000 ]`
Example: `limit=2`
Seitengröße (Vorgabe 100, höchstens 1000).

**cursor**
string
`nextCursor` der vorigen Seite, unverändert.

**offset**
integer `>= 0`
Alternative zum Cursor: Position der Seite.

**lifecycle**
string
`active` (Vorgabe), `inactive` oder `all`.

## Responses

› **200** Eine Seite Aufträge

> **400** Structured application error

> **401** Structured application error

> **403** Structured application error

> **404** Structured application error

> **409** Structured application error

> **500** Structured application error

GET /api/dto/fsg3

## Response samples

200 | 400 | 401 | 403 | 404 | 409 | 500

**Content type**
application/json

**Example**
Erste Seite (filter=status%3Dto_confirm&sort=subjectDisplayName&limit=2)

Copy Expand all Collapse all


```
{
  - "filter": [
        "status=to_confirm"
    ],
    "total": 3,
    "offset": 0,
    "limit": 2,
    "returned": 2,
    "nextCursor": "eyJrIjoiZXlKemIzSjBWbUZzZFdVaU9pSndaWFJ5WVNCaVpXbHpjR2xsYkl
  - "rows": [
     + { … },
     + { … }
    ]
}
```

# Create an assignment (FSG, fsg3)

Schritt 2 des Standardablaufs: einen FSG-Auftrag anlegen. Der Ausführende wird wie beim Import über die PLZ des Schuldners zugeordnet. Die Antwort nennt das Aktenzeichen, unter dem der Auftrag gelesen und geändert wird.

AUTHORIZATIONS: › *apiKeyAuth*

REQUEST BODY SCHEMA: application/json
required

| | |
|---|---|
| `id` | string<br>Eigene Auftrags-ID (optional). Ein zweites Anlegen mit derselben ID wird abgewiesen. |
| `tenantId` | string<br>Mandant; entfällt, wenn der Key genau einem Mandanten gehört. |
| `clientId` | string<br>Einspeisender; entfällt, wenn der Key zu einem Einspeisenden gehört. |
| `order` ›<br>required | object |
| `subject` ›<br>required | object |
| `creditor` › | object |
| `status` › | object |
| `property name*`<br>additional property | any |

## Responses

› **200** Der Auftrag ist angelegt

› **400** Structured application error

› **401** Structured application error

› **403** Structured application error

› **404** Structured application error

› **409** Structured application error

› **500** Structured application error

POST /api/dto/fsg3

## Request samples

**Payload**

**Content type**
application/json

Copy Expand all Collapse all


```
{
    "id": "3f6d2c1e-7b4a-4c1e-9a53-2b8e0c4d7f10",
    "tenantId": "tenant-demo1",
    "clientId": "a403fc5a-0213-485a-95f1-091a5daf51f7",
  - "order": {
        "clientReference": "ERP-4711",
        "creditorReference": "GL-2026-0815",
        "typeLabel": "std",
      + "serviceIds": [ … ],
        "claimDescription": "Adressermittlung",
        "principalClaim": 100,
        "costs": 0,
        "interest": 0,
        "payments": 0,
        "totalClaim": 100,
        "dailyInterest": 0,
        "showVatToSubject": false,
        "bankAccountHolder": "DEBICO AG",
        "bankBic": "GENODEF1M04",
        "bankIban": "DE02120300000000202051"
    },
```

```
  - "subject": {
        "firstName": "Petra",
        "name1": "Beispiel",
        "street": "Musterweg 1",
        "postalCode": "33106",
        "city": "Paderborn"
    },
  - "creditor": {
        "name1": "Gläubiger GmbH",
        "street": "Gläubigerweg 1",
        "postalCode": "33613",
        "city": "Bielefeld"
    }
}
```

### Response samples

200 | 400 | 401 | 403 | 404 | 409 | 500

**Content type**
application/json

Copy

```
{
    "assignmentId": "3f6d2c1e-7b4a-4c1e-9a53-2b8e0c4d7f10",
    "ref": "100036",
    "dataVersion": 1790222260000
}
```

## Read an assignment with results (FSG, fsg3)

Schritt 3 des Standardablaufs: einen Auftrag vollständig lesen – Auftragsdaten, Schuldner, Gläubiger, Status mit Datumswerten und die Ergebnisse je Leistung. Es erscheint nur, was das Konto auch in der App sehen darf.

AUTHORIZATIONS: › *apiKeyAuth*

PATH PARAMETERS

ref
required

string
Example: `100036`
Aktenzeichen des Auftrags; die Auftrags-ID geht ebenso.

## Responses

- **200** Der Auftrag
- **400** Structured application error
- **401** Structured application error
- **403** Structured application error
- **404** Structured application error
- **409** Structured application error
- **500** Structured application error

GET /api/dto/fsg3/{ref}

### Response samples

200 | 400 | 401 | 403 | 404 | 409 | 500

Content type
application/json

Copy Expand all Collapse all

```
{
  - "assignment": {
        "id": "3f6d2c1e-7b4a-4c1e-9a53-2b8e0c4d7f10",
        "ref": "100036",
        "tenantId": "tenant-demo1",
        "moduleId": "module/fsg",
        "lifecycleState": "active",
        "clientId": "a403fc5a-0213-485a-95f1-091a5daf51f7",
        "clientRef": "IADB-4201",
        "clientDisplayName": "DEBICO AG",
        "partnerId": "b1c2d3e4-0000-4000-8000-000000000001",
        "partnerRef": "IADB-5001",
        "partnerDisplayName": "Anton Aus GmbH",
        "subjectDisplayName": "Petra Beispiel",
      + "assignee": { … },
      + "status": { … },
      + "order": { … },
      + "subject": { … },
      + "creditor": { … },
      + "results": { … },
        "modified": 1790913460000
    }
}
```


## Change an assignment (FSG, fsg3)

Schritt 5 des Standardablaufs: einen Auftrag ändern.

Nur die genannten Felder ändern sich, `null` leert ein Feld, `status.status` wechselt den Status – sofern das Konto das in der App darf. Der Aufruf gilt ganz oder gar nicht: ein unbekanntes, nicht änderbares oder nicht erlaubtes Feld weist ihn ab, bevor etwas geschrieben wird. Löschen gibt es nicht; storniert wird per Statuswechsel. Die Antwort ist der Auftrag nach der Änderung.

AUTHORIZATIONS: › *apiKeyAuth*

PATH PARAMETERS

ref
required

string
Example: `100036`
Aktenzeichen des Auftrags; die Auftrags-ID geht ebenso.

REQUEST BODY SCHEMA: application/json

required

| | |
|---|---|
| order > | object |
| subject > | object |
| status > | object |
| results > | object |
| property name*<br>additional property | any |

## Responses

> **200** Der geänderte Auftrag

> **400** Structured application error

> **401** Structured application error

> **403** Structured application error

> **404** Structured application error

> **409** Structured application error

> **500** Structured application error

PATCH /api/dto/fsg3/{ref}

### Request samples

**Payload**

**Content type**
application/json

**Example**
Ein Feld ändern

Copy Expand all Collapse all


```
{
  - "order": {
        "claimDescription": "Adressermittlung, dringend"
    }
}
```


## Response samples

200 400 401 403 404 409 500

**Content type**
application/json

Copy Expand all Collapse all


```
{
  - "assignment": {
        "id": "3f6d2c1e-7b4a-4c1e-9a53-2b8e0c4d7f10",
        "ref": "100036",
        "tenantId": "tenant-demo1",
        "moduleId": "module/fsg",
        "lifecycleState": "active",
        "clientId": "a403fc5a-0213-485a-95f1-091a5daf51f7",
        "clientRef": "IADB-4201",
        "clientDisplayName": "DEBICO AG",
        "partnerId": "b1c2d3e4-0000-4000-8000-000000000001",
        "partnerRef": "IADB-5001",
        "partnerDisplayName": "Anton Aus GmbH",
        "subjectDisplayName": "Petra Beispiel",
        "assignee": null,
      + "status": { … },
      + "order": { … },
      + "subject": { … },
      + "creditor": { … },
        "results": { },
        "modified": 1790222290000
    },
    "dataVersion": 1790222290000
}
```