

TECHNISCHE SPECIFICATIE

EKB — BPV

Technische documentatie voor developers

VERTROUWELIJK





Inhoudsopgave

Overzicht & scope	3
Aan de slag & dev-omgeving	3
Beveiliging	3
Transport (TLS)	3
Authenticatie & autorisatie	3
Per endpoint	4
Conventies & ontwerpprincipes	4
Formaat & naamgeving	4
Paginering	4
Versionering	5
Wijzigingen ophalen	5
Do's & don'ts	5
Do's	5
Don'ts	5
Foutafhandeling	6
Voorbeeld (401)	6
Veelvoorkomende statuscodes	6
Throttling & rate limiting	7
Aanbevolen client-gedrag	7
Beschikbaarheid & SLA	7
Uitgangspunten	7
Endpoints op hoofdlijnen	7
Versiebeheer & changelog	9



Overzicht & scope

EKB (Eduarte Koppeling BPV) is een **lezen-en-schrijven** REST-API voor het BPV-proces: matching, leerbedrijven, BPV-inschrijvingen, documenten en logboek. De API levert JSON en volgt de gemeenschappelijke Connect-conventies.

- **Versie:** 1.0
- **Pad-prefix:** /connect/ekb/1_0
- **Volledige OpenAPI-specificatie:** meegeleverd als **EKB_v1.0.yml** (zelfstandig, importeerbaar in tooling zoals Swagger UI, Postman of een codegenerator).
- **Interactieve referentie online:** [API-referentie van EKB](#)

Deze specificatie beschrijft de technische uitgangspunten en geeft een overzicht op hoofdlijnen; de volledige endpoint- en schemadefinities staan in de meegeleverde **EKB_v1.0.yml** en in de online referentie.

Aan de slag & dev-omgeving

1. **Toegang aanvragen** — vraag via Eduarte (connect@eduarde.nl) OAuth2-clientgegevens aan voor EKB.
2. **Scopes** — EKB kent twee scopes:
 - **ekbb** — BPV Begeleiding (inschrijvingen, documenten, logboek beheren);
 - **ekbm** — BPV Matchen (groepen, studenten en bedrijven t.b.v. matching). De meeste endpoints accepteren één van beide; de exact vereiste scope per endpoint staat in **EKB_v1.0.yml**.
3. **Token ophalen** — haal een access-token op via de OAuth2 Client Credentials-flow met de benodigde scope(s).
4. **Aanroepen** — stuur het token mee als **Authorization: Bearer <token>** en roep endpoints aan onder /connect/ekb/1_0.
5. **Testen** — gebruik de beschikbare ontwikkel-/testomgeving voordat je in productie gaat.

Beveiliging

Alle Connect API's worden uniform beveiligd. De maatregelen hieronder gelden voor elke connector; afwijkingen staan in de connector-specifieke secties.

Transport (TLS)

Alle communicatie verloopt verplicht via **HTTPS**. Minimaal **TLS 1.2**, bij voorkeur TLS 1.3. Er zijn geen HTTP-endpoints, ook niet voor redirects. Een geldig servercertificaat is vereist. Hierin volgen we de aanbeveling van de [NSCT](#)

Authenticatie & autorisatie

Authenticatie verloopt via **OAuth 2.0 met de Client Credentials-flow** (server-to-server):

- Token-endpoint: <https://login.educus.nl/oauth2/token?organisatieuuid=<uuid>>



- De client wisselt zijn `client_id` en `client_secret` in voor een access-token.
- Het verkregen token wordt bij elke aanroep meegestuurd in de `Authorization: Bearer <token>` -header.

Scopes bepalen de toegangsrechten volgens het principe van *least privilege*. Elke API kent één of meer scopes; een token bevat alleen de scopes die de client nodig heeft. De vereiste scope(s) per connector staan in de connector-specifieke sectie.

Waar OAuth 2.0 niet mogelijk is (bijvoorbeeld bij legacy-integraties), kan als terugvaloptie een **Bearer-token (JWT)** worden gebruikt.

Per endpoint

Elk endpoint heeft een expliciete `security` -definitie. Onjuiste of ontbrekende autorisatie leidt tot een `401 Unauthorized` (niet geauthenticeerd) of `403 Forbidden` (onvoldoende rechten); zie *Foutafhandeling*.

Conventies & ontwerpprincipes

De Connect API's volgen gemeenschappelijke conventies, zodat integraties voorspelbaar en consistent zijn.

Formaat & naamgeving

- **JSON** is het representatieformaat voor requests en responses.
- **Resources** zijn meervoud en in `camelCase`: `/fases`, `/organisatieeenheden`.
- **Path-parameters** zijn `camelCase`: `/fases/{id}`.
- **Query-parameters** zijn `lowercase kebab-case` (uitzondering: het historische `pageSize`).
- **Schema-namen** zijn `PascalCase`; **properties** zijn `camelCase`.

Paginerig

Lijst-endpoints zijn gepagineerd (offset-based):

Parameter	Type	Default	Beschrijving
<code>offset</code>	integer	0	Startpositie in de resultatenlijst
<code>pageSize</code>	integer	100	Aantal items per pagina (standaard maximum 200)

De response bevat een `meta` -object met `offset`, `pageSize` en `totalItems`, plus een `items` -array. Een leeg resultaat geeft een lege `items` -array terug — geen `404`.

```
{
  "meta": { "offset": 20, "pageSize": 10, "totalItems": 156 },
  "items": [ { "id": "..." } ]
}
```



Versionering

Versies volgen **SemVer** in de vorm `MAJOR.MINOR`. De major-versie staat in het pad (bv. `/connect/eks/v1_0/...`). Non-breaking wijzigingen (nieuw optioneel veld, nieuw endpoint, nieuwe enum-waarde) verschijnen als minor-versie binnen dezelfde major; breaking wijzigingen leiden tot een nieuwe major-versie die parallel draait. Bij uitfasering gelden een deprecation-notice en een sunset-datum (minimaal 6 maanden), gecommuniceerd via `Deprecation` - en `Sunset` -headers.

Wijzigingen ophalen

Voor incrementeel synchroniseren ondersteunen de relevante zoek-endpoints (zoals `/bpvinschrijvingen` en `/bedrijven`) de query-parameter `changed-since` (datum-tijd, ISO-8601). Daarmee haal je alleen op wat sinds dat moment is gewijzigd, in plaats van steeds de volledige set.

In de toekomst signaleert de notificatiekoppeling **EKN** wijzigingen via **cloudevents**, zodat een afnemer direct de gewijzigde stand kan ophalen zonder te pollen. Deze notificaties zijn **nog niet beschikbaar**; tot die tijd gebruik je `changed-since` (polling).

Do's & don'ts

Praktische richtlijnen voor een robuuste integratie met Connect.

Do's

- **Gebruik paginering** bij het ophalen van lijsten; doorloop pagina's via `offset / pageSize`.
- **Hergebruik access-tokens** binnen hun geldigheidsduur in plaats van bij elke aanroep een nieuw token op te halen.
- **Handel fouten af op basis van de HTTP-statuscode** en het `problem+json` -foutobject, niet op tekst.
- **Implementeer backoff** bij `429 Too Many Requests` en respecteer de `Retry-After` -header.
- **Pin op een major-versie** in het pad en test tegen de testomgeving vóór productie.
- **Cache referentie-/stamdata** waar passend, omdat deze weinig verandert.

Don'ts

- **Geen polling zonder backoff** — vermijd onnodige herhaalde aanroepen in korte tijd.
- **Geen hardcoded URLs zonder versie** — verwijst altijd naar het versie-pad.
- **Vertrouw niet op niet-gedocumenteerde velden** of op de volgorde van items zonder expliciete sortering.
- **Stuur geen gevoelige gegevens** in query-parameters of logregels.
- **Ga niet uit van directe consistentie** tussen los opgehaalde resources.



Bron: API-CONVENTIONS.md (o.a. "Wanneer niet suppressen") en de Connect REST-principes.

Foutafhandeling

Foutmeldingen zijn gestandaardiseerd op basis van **RFC 7807** (*Problem Details for HTTP APIs*). Hierdoor zien fouten er over alle Connect API's hetzelfde uit.

- Alle fouten hebben mediatype `application/problem+json`.
- Elke fout bevat een compacte JSON-body met een omschrijving en details.
- Elke operatie definieert ten minste de onderstaande fouten.

Voorbeeld (401)

```
{
  "type": "https://tools.ietf.org/html/rfc6750#section-3.1",
  "title": "Unauthorized",
  "status": "401",
  "detail": "The request requires user authentication."
}
```

Veelvoorkomende statuscodes

Status	Betekenis	Wanneer
200 / 201	OK / Created	Verzoek geslaagd
400	Bad Request	Ongeldig verzoek (validatie)
401	Unauthorized	Niet (geldig) geauthenticeerd
403	Forbidden	Geauthenticeerd, maar onvoldoende rechten
404	Not Found	Resource bestaat niet
405	Method Not Allowed	HTTP-methode niet toegestaan op deze resource
429	Too Many Requests	Rate limit overschreden (zie <i>Throttling</i>)
500	Internal Server Error	Onverwachte fout aan serverzijde

Let op: API's uit de keten (OKxx) volgen de foutafhandeling van hun eigen standaard (bv. OOAPI) en kunnen hiervan afwijken.



Throttling & rate limiting

Alle endpoints kennen **rate limiting** om misbruik en overbelasting te voorkomen. Dit wordt op infrastructuurniveau afgedwongen via de Topicus API-gateway.

Bij overschrijding van de limiet antwoordt de API met **429 Too Many Requests**. De response kan de volgende headers bevatten:

Header	Betekenis
Retry-After	Aantal seconden tot een volgende aanroep is toegestaan
X-RateLimit-Limit	Maximum aantal aanroepen per tijdsperiode
X-RateLimit-Remaining	Resterend aantal aanroepen in de huidige periode

Aanbevolen client-gedrag

- Respecteer **Retry-After** en wacht ten minste het opgegeven aantal seconden.
- Gebruik **exponential backoff** met jitter bij herhaalde 429-responses.
- Verspreid bulk-aanroepen over de tijd in plaats van ze gelijktijdig te versturen.

Beschikbaarheid & SLA

Eduarte Connect wordt aangeboden als beheerde dienst. Beschikbaarheid, onderhoud en support zijn vastgelegd in de partnerovereenkomst en bijbehorende SLA.

Uitgangspunten

- De API's zijn ontworpen voor continue beschikbaarheid tijdens kantoor- en uren.
- Gepland onderhoud wordt vooraf aangekondigd via de gangbare communicatiekanalen en zo veel mogelijk buiten piekuren uitgevoerd.
- Storingen en vragen kunnen worden gemeld bij Eduarte via de beklende servicedesk meldingen
- Partner vragen kunnen worden gemeld bij **info@eduarte.nl**.

Let op: concrete SLA-cijfers (beschikbaarheidspercentage, reactietijden, onderhoudsvensters) worden vastgelegd in de partnerovereenkomst en zijn **nog te bevestigen** voor deze documentatie.

Endpoints op hoofdlijnen

EKB groepeert de endpoints in drie clusters. Alle lijst-endpoints zijn gepagineerd (zie *Conventies*).

BPV-inschrijvingen



Endpoint	Methoden	Toelichting
/bpvinschrijvingen	GET, POST	Zoeken/aanmaken
/bpvinschrijvingen/{id}	GET, PUT	Opvragen/bijwerken (status wijzigt hier níét)
/bpvinschrijvingen/{id}/_status	POST	Statusovergang uitvoeren
/bpvinschrijvingen/{id}/documents	GET	Documenten opvragen
/bpvinschrijvingen/{id}/documents/{kind}	POST	Document toevoegen (per soort)
/bpvinschrijvingen/{id}/logentries	GET, POST	Logboek opvragen/aanvullen
/bpvinschrijvingen/{id}/logentries/{datum}	PATCH	Logboekregel bijwerken

Matching (lezen)

Endpoint	Methoden	Toelichting
/groepen · /groepen/{id}	GET	Groepen opvragen
/studenten/{id}	GET	Student opvragen
/onderwijsbegeleiders	GET	Onderwijsbegeleiders opvragen

Leerbedrijven

Endpoint	Methoden	Toelichting
/bedrijven · /bedrijven/{id}	GET, POST / GET, PUT	Bedrijven beheren
/bedrijven/{id}/contactpersonen	GET, POST	Contactpersonen opvragen/aanmaken
/bedrijven/{id}/contactpersonen/{contactpersoonid}	GET, PUT, DELETE	Contactpersoon beheren

Het zoeken op /bpvinschrijvingen ondersteunt onder meer filtering op externe-id , changed-since , organisatieeenheid-id , locatie-id , highlevelstatus (MATCHING/ACTIEF/BEEÏNDIGD) en created-since . Voor de volledige parameter-, response- en schemadefinities: zie de meegeleverde **EKB_v1.0.yml** of de [API-referentie](#).



Versiebeheer & changelog

EKB volgt de Connect-versioneringsstrategie (SemVer, major-versie in het pad). De huidige versie is 1.0. Non-breaking wijzigingen verschijnen als minor-versie binnen dezelfde major; breaking wijzigingen leiden tot een nieuwe major-versie die parallel draait, met een deprecation-notice en sunset-datum.

Versie	Datum	Wijzigingen
1.0	2026	Eerste versie: matching, leerbedrijven, BPV-inschrijvingen (incl. status, documenten en logboek).

VERTROUWELIJK