

# FAQ: Zabun Api

- [FAQ: Zabun Api](#)
  - [Guidelines](#)
  - [General](#)
    - [Where can I find the JSON-to-screen mapping?](#)
    - [Which endpoint should I use?](#)
    - [Where can I find the swagger then?](#)
    - [What values do I need?](#)
    - [How can I test the connection to the zabun api/server?](#)
    - [What is curl, and what should I do with it?](#)
    - [How do I do curl on windows then?](#)
    - [What are HTTP status codes then?](#)
    - [Authentication](#)
      - [API Key](#)
    - [How do I sort, filter, and paging?](#)
    - [How to retrieve specific or all languages?](#)
  - [Buildings](#)
    - [How do I synchronize my properties?](#)
    - [An alternative way of retrieving properties](#)
    - [Retrieve all data from a property](#)
      - [1. Via the parameter '?extended=true'](#)
      - [2. Request the municipality ID via another endpoint](#)
    - [Price visible](#)
    - [Mark property as ONLINE](#)
    - [Manage and assign media categories to properties for your website](#)
    - [How do I retrieve photos of properties?](#)
    - [How do I apply the photo and file type filtering set in my Zabun admin?](#)
    - [What are documents and why don't I want them on my website?](#)
    - [How do layouts work in the API](#)
    - [How do I create a new property?](#)
    - [How can I find out how long a property has been in a specific transaction \(for sale, for rent, etc.\)?](#)
  - [Accounts](#)
    - [How do I find my accounts?](#)
  - [Contacts](#)
    - [How do I search for my contacts?](#)
    - [How do I create a new contact?](#)
    - [How do I make a contact who is also a candidate for a property?](#)

- [How do I create a new contact + associated search form?](#)
- [How do I add a file to a contact?](#)
- [Tasks](#)
- [Search cards](#)
  - [How do I search my search cards?](#)
  - [How can I add a municipality to a search request?](#)
  - [The difference between main types and types](#)
  - [Adding extra filters to the search form](#)
- [Selection lists](#)
- [Extra Fields](#)
  - [Step 1: Retrieving additional fields](#)
  - [Step 1a: Retrieve dropdown options.](#)
  - [Step 3. Retrieving the additional field values that were entered for the object](#)
  - [Step 4. Enter or edit an additional field value](#)
- [Short questions with concise answers](#)
  - [Q: I get a validation error with the message "Error converting value xxxx to type 'System.Int64\[\]'.](#)
  - [Q: Land area, habitable area, cadastral income](#)
  - [Q: Construction technical information](#)
  - [Q: LEGAL INFO](#)
  - [Q: I receive EPC Certificate 1 via the API, how do I know which value represents which Label or is this separate?](#)
  - [Q: What's the difference between the standard EPC fields and the Shared fields? Why is there an epb\\_reference but no epb\\_value or custom\\_epb\\_label?](#)
  - [Q: Flood sensitivity values](#)
  - [Q: Environmental permit for urban development activities](#)
  - [Q: Sales Law, Enforcement](#)
  - [Q: Environmental permit for the subdivision of land](#)
  - [Q: Where can I find my representatives, and do I need account rights?](#)
  - [Q: What are the possible flood sensitivity values?](#)
  - [Q: Where can I find sales rights, development permits, judicial or administrative measures?](#)
  - [Q: I have 15 records in zabun, for example, but I only see 5 in the API response?](#)
  - [Q: If a property is added to a project, will the project's modified date be updated?](#)
  - [Q: What is the purpose of the sales\\_rep variable \(=representative/agent\) in a search query, and what happens if it is not filled in during the request?](#)
  - [Q: What additional things do we need to take care of after going live with our API integration?](#)
  - [Q: Do files and photos \(=Media\) need to be downloaded and can the direct URL be used?](#)

- Q: Can properties be retrieved on the fly? Or is it intended that all data be retrieved from a separate database?
- Q: How do I retrieve only projects? How do I retrieve only "new construction" properties?

# Guidelines

- Take advantage of the (<https://gateway-cmsapi.v2.zabun.be>) version of the API. We have already internally set a rate limit of 5 requests per second. (The current public version of the API will be deactivated in the near future.)
- Cache data yourself to minimize the number of requests. A useful guideline is, for example, one request per hour per property.
- The same goes for photos and files: make a copy of them to avoid broken links or invisible photos
- The option lists can be retrieved separately, but it's best to retrieve them via the /xxxxx/option\_items calls (e.g., transaction and type via /property/option\_items instead of /property/transaction). The option\_items calls contain all the option lists for the requested item. This data is best cached.
- The list of geocity data (for example, all cities in Belgium) is cached for longer than 1 day, as this is a very intensive request, and cities hardly change.
- Keep the authentication keys (client\_id, server\_id, api\_key) secret. Therefore, avoid using JavaScript clients that pass these headers in plain text.

# General

Where can I find the JSON-to-screen mapping?

Mapping of fields (</files/mapping>).

Which endpoint should I use?

## URL

New <https://gateway-cmsapi.v2.zabun.be> (<https://gateway-cmsapi.v2.zabun.be>).  
Old <https://public.api-cms.zabun.be> (<https://public.api-cms.zabun.be>).

The API on the old platform will be taken offline in the near future. If you haven't yet migrated to the new platform, please do so. There are no differences in the basic functionality of the API between the new and the old,

except for the URL.

## Where can I find the swagger then?

	<b>URL</b>
New	<a href="https://gateway-cmsapi.v2.zabun.be/swagger/index.html">https://gateway-cmsapi.v2.zabun.be/swagger/index.html</a> ( <a href="https://gateway-cmsapi.v2.zabun.be/swagger/index.html">https://gateway-cmsapi.v2.zabun.be/swagger/index.html</a> )
Old	<a href="https://public.api-cms.zabun.be/swagger/ui/index">https://public.api-cms.zabun.be/swagger/ui/index</a> ( <a href="https://public.api-cms.zabun.be/swagger/ui/index">https://public.api-cms.zabun.be/swagger/ui/index</a> )

## What values do I need?

<b>Header name</b>	<b>Value</b>
api_key	request via support
client_id	request via support
server_id	request via support
X-CLIENT-ID	Your company ID can be found by hovering over your username in zabun at the top right.
X-USER-ID	Your user ID

Tip: Everything is case-sensitive. If a header is capitalized, you must enter it as such. Otherwise, errors may occur.

## How can I test the connection to the zabun api/server?

In the curl below, replace the #...# values with the ones above.

```
curl -X GET 'https://gateway-cmsapi.v2.zabun.be/auth/v1/heartbeat' \
--header 'Accept: text/plain' \
--header 'api_key: #api_key#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'X-USER-ID: #X-USER-ID#'
```

This is the answer you should get

```

HTTP/1.1 200 OK
Cache-Control: no-cache
Pragma: no-cache
Content-Type: application/json
Expires: -1
Server: Microsoft-IIS/10.0
X-AspNet-Version: 4.0.30319
Request-Context: appId=cid-v1:863e6ef5-006b-4414-af65-e59561f372b2
Access-Control-Expose-Headers: Request-Context
X-Powered-By: ASP.NET
Date: Tue, 23 Jan 2024 06:55:23 GMT
Connection: close
Content-Length: 15

"V1 xx/xx/yyyy"

```

## What is curl, and what should I do with it?

The `curl` command is a widely used command-line tool in Unix and Unix-like operating systems for transferring data using URL syntax. It supports various protocols such as HTTP, HTTPS, FTP, and SFTP, making it widely applicable for downloading or sending files to and from servers. It is also `curl` frequently used in scripting and programming to make API requests and interact with web services.

## How do I do curl on windows then?

([https://www.vulongtran.com/how-to-use-visual-studio-to-run-curl-commands#google\\_vignette](https://www.vulongtran.com/how-to-use-visual-studio-to-run-curl-commands#google_vignette))

1. Install Visual Studio Code
2. Add the REST Client extension
3. Create a file with the extension .crl
4. Press F1 and choose command: REST Client: Send Request
5. Result: Status code 200

```

HTTP/1.1 200 OK
Cache-Control: private
Content-Type: text/html; charset=utf-8
Content-Encoding: gzip
Vary: Accept-Encoding
Date: Tue, 23 Jan 2024 06:50:26 GMT
Connection: close
Content-Length: 10967

```

# What are HTTP status codes then?

HTTP status codes are standardized numbers returned by a web server to indicate the status of a request, such as whether it was successful, contained an error, or requires further action. They are divided into five categories, each representing a specific class of responses: informational (1xx), success (2xx), redirect (3xx), client error (4xx), and server error (5xx). The five most common HTTP status codes are:

- 200 (OK)
- 401 (Invalid)
- 403 (Forbidden)
- 404 (Not Found)
- 500 (Internal Server Error)

## Authentication

### API Key

Three additional API key headers—`client_id`, `server_id`, and `api_key`—allow direct requests to be made without using the OAuth endpoints. API key data is activated and issued by support to external partners or web developers who want to set up synchronization via the API.

```
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'api_Key: #api_key#' \
```

**Important: You cannot call OAuth endpoints with the API key headers. Consequently, if you try to include an OAuth Bearer token in the Authentication header, you will always receive a 403 authentication failed error.** This authentication takes precedence over API Key authentication. Therefore, you must not add an authentication header.

## How do I sort, filter, and paging?

basic setup for all search endpoints:

```
{
  "paging": {
    "page": 0,
    "size": 100
  },
  "filtering": {
    "full_text": "xxxx"
  },
  "sorting": {
    "sort": "MOST_RECENT",
    "order": "ASC"
  }
}
```

*Paging :*

- page: current page, starts at 0
- size: number of items shown

The page field under paging, which is sent in search, for example, starts with index 0. So if you have 21 items with size 10, you'll see:

- page 0: the first 10 items
- page 1: the 2nd set of 10 items
- page 2: 1 item

*Filtering :*

- full\_text: shows only the records that contain this text
- updated\_since: shows only records updated since that date

*Sorting :*

- order: ASC or DESC
- sort: MOST\_RECENT or CREATED

## How to retrieve specific or all languages?

- Accept-Language with value \*: then you will see everything
- Accept-Language with value nl: then you will only see nl
- Accept-Language with value nl,fr: then you will see nl and fr

Accept-Language is a header parameter

# Buildings

## How do I synchronize my properties?

The following endpoint is suggested if you want to retrieve the properties that will be activated in Zabun for the media api.

```
curl --location 'https://gateway-cmsapi.v2.zabun.be/api/v1/property/media-syn
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'api_Key: #api_key#' \
```

De 14 in de URL staat voor media met ID "14", dat "Website via API" noemt in Zabun. De vertegenwoordiger zal deze media gebruiken om aan te duiden welke panden online mogen op de website die u zal connecteren met de API.

Via deze endpoint worden enkel panden opgehaald die zijn ingesteld als publiek (= groen huisje, status actief, niet verwijderd en niet in het archief) en geactiveerd voor de media binnen de opgegeven start en eventueel (geplande) einddatum

Deze endpoint zal ook automatisch panden deactiveren voor de media als de publicatie van het pand buiten de ingestelde datums valt, of het pand niet meer ingesteld is als publiek (= een rood huisje of status inactief, verwijderd of gearchiveerd)

Panden die u terug krijgt in de verkregen lijst, mogen online weergegeven worden. Panden die voorheen in de lijst voorkwamen, maar bij de laatste request niet meer, moeten offline gehaald worden.

U kan zelf nog filters toevoegen aan deze request, om bijvoorbeeld panden te verkrijgen met een bepaalde transactie type, pand type, stad ... zie hiervoor de swagger. De filters moeten worden toegevoegd aan de uri als query-parameters

## Een alternatieve manier van panden ophalen

Als u echter alle panden wil ophalen of met uw eigen filters en zonder gebruik te maken van een media, dan kan de standaard search endpoint gebruikt worden.

```
curl --location --request POST 'https://https://gateway-cmsapi.v2.zabun.be/api/v1/search' \
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'api_key: #api_key#' \
--data-raw ''
```

Hier vult u zelf de filters in, te vinden op de swagger, in de body van de request. Dit is een verschil met de **GET** endpoint hierboven sinds dat deze een **POST** is. Onthoud: hier kan u dus ook panden terug krijgen die eventueel niet online mogen weergegeven worden op de website, als u niet de correcte filters gebruikt. Daarom stellen we voor om de media-sync endpoint, hierboven, te gebruiken als u de vertegenwoordiger vanuit Zabun de controle wil geven over de beschikbare panden.

**always\_show\_all\_children: zal altijd alle kind objecten in de lijst stoppen o**

Indien u de sync request wil nabootsen met deze search endpoint, moet u ten minste volgende filters meegeven

```
public: true,
active_media_ids: [#media_id#], //Dit zal 14 zijn
```

## Alle data van een pand ophalen

Bovenstaande endpoints halen een lijst van panden op met een beperkte collectie van data(=variabelen). Als u echter alle data van een pand wilt, moet u dit pand rechtstreeks opvragen

```
curl --location --request GET 'https://gateway-cmsapi.v2.zabun.be/api/v1/property' \
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'api_key: #api_key#' \
--data-raw ''
```

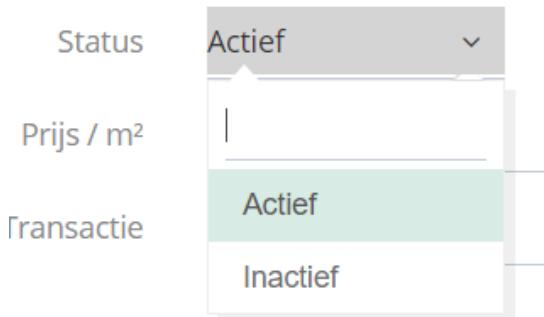
Als je wil dat de geo informatie uitgebreid wordt terug gegeven kan dat op 2 manieren:

1. Via de parameter '?extended=true'

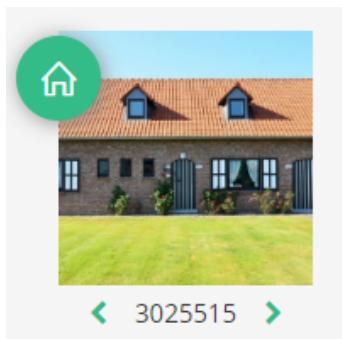
```
curl --location --request GET 'https://gateway-cmsapi.v2.zabun.be/api/v1/prope
' \
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'api_key: #api_key#' \
--data-raw ''
```

Speciale velden van een pand

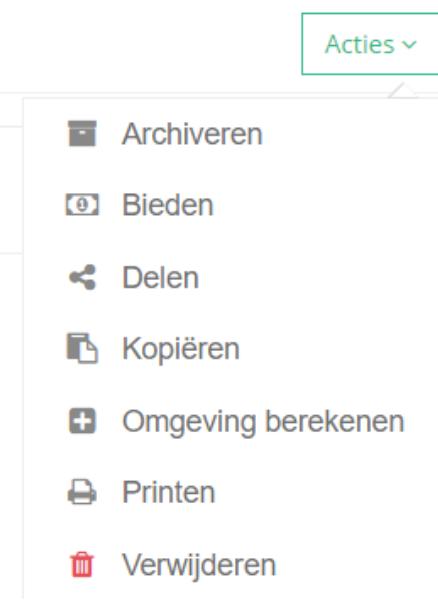
- **publish** = deze vlag bepaalt of het pand online (true) of offline (false) is, deze vlag gebruikt de onderstaande velden om dat te bepalen
- **status\_id** = 1 = actief, 2 = inactief



- **show**: true = groen huisje, show = false = rood huisje



- **deleted** = is het pand gewist of niet
- **archived** = is het pand gearchiveerd of niet



- **history\_transaction** = is een historiek van transacties (te koop, te huur,...). deze lijst is oplopend gesorteerd op datum. hier kan je zien wanneer de transactie is aangepast.

```
"history_transaction": [
  {
    "transaction_id": 1,
    "property_id": 2091991,
    "date_history": "2020-02-28T14:02:46+01:00",
    "responsible_salesrep_person_id": 18911,
    "responsible_salesrep_bedrijf_id": 3270,
    "company_id": 3270
  },
  {
    "transaction_id": 5,
    "property_id": 2091991,
    "date_history": "2021-07-09T10:58:39+02:00",
    "responsible_salesrep_person_id": 18911,
    "responsible_salesrep_bedrijf_id": 3270,
    "company_id": 3270
  },
]
```

2. De id van de gemeente opvragen via een ander endpoint

```
curl --location --request GET 'https://gateway-cmsapi.v2.zabun.be/api/v1/geo/c
' \
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'api_key: #api_key#' \
--data-raw ''
```

## Prijs zichtbaar

Ongelijk een portaal, halen we niet automatisch de prijs weg uit de pand data in de API. De logica voor het tonen/onzichtbaar maken van de prijs moet door de website zelf toegepast worden. Bij de pand data, in beide de GET en de SEARCH, vindt u variable **price\_visible**. Indien deze "false" is, mag u de prijs niet tonen (behalve als anders afgesproken met de makelaar).

## Pand markeren als ONLINE

Wanneer u een pand synchroniseert en online brengt op uw website, is het interessant om dit door te geven aan de CMS Api. Deze gegevens worden gebruikt om aan de vertegenwoordiger te tonen dat het pand succesvol is gesynchroniseerd en online staat, waarbij een link naar het pand op de website wordt toegevoegd aan Zabun.

```
curl -X POST 'https://gateway-cmsapi.v2.zabun.be/api/v1/property/#PAND_ID#/me
--header 'Content-Type: application/json' \
--header 'Accept: application/json' \
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'api_key: #api_key#' \
-d '{ "start_media": true}'
```

U kan ook aangeven wanneer een pand op uw website wordt bijgewerkt, of offline gehaald van de website, via deze endpoint. Deze drie JSON body's zijn mogelijke waarden die u kan gebruiken. U kan ze niet door elkaar gebruiken

```
{"start_media": true} => "Het pand is aangemaakt op de website en online te
>{"update_media": true} => "Het pand is bijgewerkt
>{"stop_media": true} => "Het pand is offline gehaald en niet meer te vinden"
```

## Media categorie beheren en toewijzen aan panden voor op uw website

In Zabun is het mogelijk om een categorie te kiezen bij het activeren van een pand voor de geselecteerde media. Om een categorie beschikbaar te maken voor de panden voor uw media, moet u deze aanmaken via de API voordat ze geselecteerd kunnen worden op Zabun. Hieronder staan de verschillende stappen beschreven.

1. Ophalen van beschikbare categorieën Dit kan via endpoint **GET /api/v1/media/14/categories**.

```
curl --location --request GET 'https://gateway-cmsapi.v2.zabun.be/api/v1/media/14/categories' \
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'api_key: #api_key#' \
--header 'Content-Type: application/json'
```

2. Aanmaken van categorieën Dit kan via endpoint 'POST /api/v1/media/14/categories'. Enkel de 'name' is verplicht. U krijgt een object terug met daarin **id**. Dit is de 'category\_id'. Deze variabele wordt gebruikt om de categorie te koppelen aan het pand.

```
curl --location --request POST 'https://gateway-cmsapi.v2.zabun.be/api/v1/media/14/categories' \
--header 'Content-Type: application/json' \
--header 'Accept: application/json' \
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'api_key: #api_key#' \
--data '{
  "name": {
    "NL": "Mijn nieuwe categorie",
    "FR": "Ma nouvelle catégorie"
  }
}'
```

Voorbeeld antwoord

```
{
  "name": {
    "nl": "Mijn nieuwe categorie",
    "fr": "Ma nouvelle catégorie"
  },
  "id": 2,
  "media_id": 14,
  "bedrijf_id": ####
}
```

3. Verwijderen van categorieën Dit kan via endpoint **DELETE /api/v1/media/14/categories/{category\_id}**. Category\_id vindt u in de antwoorden van de GET list of POST new in variabele "ID". Gelieve op te passen met het verwijderen van categorieën. Dit heeft een direct resultaat en invloed op de panden in uw media syncs

```
curl --location --request DELETE 'https://gateway-cmsapi.v2.zabun.be/api/v1/media/14/categories/2' \
--header 'Content-Type: application/json' \
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'api_key: #api_key#' \
```

U krijgt een 200 OK met body "true" als antwoord indien het verwijderen is gelukt

- Een categorie aan een pand toewijzen Dit kan via Zabun als er categorieën bestaan voor de media

The screenshot shows a modal window titled "Website via API". It contains three input fields: "Startdatum" with the value "06-11-2024", "Einddatum" which is empty, and "Categorie" with the value "Mijn nieuwe categorie". At the bottom right are two buttons: a green "Opslaan" button and a white "Sluit venster" button.

- Verkrijgen van de categorie voor een pand in de API De variabele **active\_category\_id**, een lijst van intger ids = de category ids, wordt gegeven in de response van endpoint **POST /api/v1/property/media-sync/14** = "media sync", indien er minstens 1 categorie geselecteerd is. Je kan de variabele voor de panden ook terug vinden in endpoint **GET /api/v1/media/{media\_id}/properties**

## Hoe haal ik foto's bij panden op?

Dit kan via endpoint <https://gateway-cmsapi.v2.zabun.be/api/v1/property/{ID}/photos> (<https://gateway-cmsapi.v2.zabun.be/api/v1/property/%7BID%7D/photos>) je krijgt een url die u kan gebruiken om een kopie te downloaden, of rechtstreeks te gebruiken op uw website. Standaard zitten foto's ook bij de GET Property endpoint

```
[  
  {  
    "property_id": 4129931,  
    "type_id": 0,  
    "url": "https://files.zabun.be/upload/2791/images/c51b4ce53616061458a379e  
    "autoid": "374600279100001048",  
    "file_autoid": "374600279100001919",  
    "url_thumbnail": "https://files.zabun.be/upload/2791/images/c51b4ce536160  
    "reference": "pexels-asad-photo-maldives-1268871.jpg",  
    "file_size": 494047,  
    "index": 1,  
    "creation": "2024-04-08T13:21:12+02:00",  
    "creation_person_id": 18454,  
    "changed": "2024-04-08T13:21:12+02:00",  
    "changed_person_id": 18454,  
    "company_id": 2791,  
    "comment": {}  
  }  
]  
]
```

Gelieve volgende puntjes in acht te nemen

- URLs mogen gebruikt worden op eigen risico
- Indien een foto/bestand offline gaat op ons systeem, dan is het ook direct offline op uw website nadat de cache deze ook kwijt is
- Dit kan gebeuren bij het verwijderen van de foto/het bestand door de makelaar, maar ook door aanpassingen in ons bestanden systeem
- Het wordt sterk aangeraden om een kopie te downloaden van de foto/het bestand om dit te vermijden

Zie ook: <https://gateway-cmsapi.v2.zabun.be/wiki#q-moeten-bestanden-en-fotos-media-gedownload-worden-en-mogen-de-directe-url-gebruikt-worden>

## Hoe pas ik de foto en bestand filtering op type toe die ingesteld is in de admin van mijn Zabun?

####Wat zijn de filters? Bij het syncen van foto's en bestanden is het mogelijk om een filter toe te passen op hun type. De volgende drie regels worden vaak toegepast.

- Foto's en bestanden zonder een gekozen type worden uit hun lijsten gefilterd. M.a.w. ze moeten een type hebben voordat ze zichtbaar mogen zijn op de website/portaal
- Voor foto's wordt standaard alle types doorgegeven. Men kan in de Zabun admin ervoor kiezen om specifieke foto's types uit te filteren. M.a.w.

types kunnen aangevinkt worden om aan te geven dat de foto's met deze type niet zichtbaar mogen zijn op de website/portaal. => all excluding X's

- Voor bestanden wordt standaard geen enkele type toegelaten/doorgegeven. Men kan in de Zabun admin ervoor kiezen om specifieke bestand types toe te laten. M.a.w. types worden standaard uitgefiterd, behalve deze die toegelaten zijn en dus zichtbaar mogen zijn op de website/portaal. => only including X's

####Hoe pas ik de filters toe op mijn website?

1. **Automatisch** Via de GET property endpoint geeft u de media\_id mee als query\_parameter. Hierbij worden de filters, die hierboven zijn beschreven, automatisch toegepast op de foto en bestand lijsten die u meekrijgt in de pand data. Indien u geen media\_id meegeeft, dan wordt er geen filter toegepast en krijgt u alle foto's en bestanden terug, ongeacht hun type (dus ook zonder type). De aparte endpoints om foto's en bestanden op te halen van een pand, worden nooit automatisch gefilterd. Bijvoorbeeld: /api/v1/property/?media\_id=14.

## 2. Manueel

- U kan zelf de foto's en bestanden filteren op hun type. Hierbij kan u checken ofdat "type\_id" is ingevuld met een waarde hoger dan 0 en niet null. U kan ook zelf kiezen welke type\_ids u toe laat of uitfiltert. De namen van de types vindt u onder
  - /api/v1/property/photos/types
  - /api/v1/property/files/types
- Via '/api/v1/media//picture\_settings' krijgt u een lijst van foto types die uitgefiterd moeten worden voor de opgegeven media. Staat een bepaalde Id in de lijst, dan is er in de Zabun admin ingesteld dat fotos met deze type niet getoond mogen worden.
- Via '/api/v1/media//file\_settings' krijgt u een lijst van bestand types die zichtbaar mogen zijn op de opgegeven media. Staat een bepaalde Id in de lijst, dan is er in de Zabun admin ingesteld dat bestanden met deze type getoond mogen worden. Bestand type ids die niet in de lijst staan, mogen niet getoond worden.

## Wat zijn documenten en waarom wil ik ze niet op mijn website?

Documenten zijn realtime gegenereerde bestanden op basis van een vooringestelde template. Deze bestanden kunnen affiches zijn met pand data, maar kunnen ook statistieken van contacten en hun gegevens zijn, zoals maand rapporten. Documenten zijn dus vooral voor intern gebruik bedoeld, en worden

best niet zomaar op de website aangeboden. Het is uiteraard mogelijk om een document te genereren in Zabun, en deze op te laden naar de bestanden van een pand, om ze dan via /files op te halen.

Als er toch documenten (en de generatie ervan) willen getoond worden op de website, dan zal u voor permissie moeten vragen aan Zabun. Deze zijn namelijk niet standaard inbegrepen in het webbouwer profiel.

## Hoe werken indelingen in de API

Om de beschikbare indelingen te verkrijgen, kan men de endpoint '/property/layouts' oproepen. Deze data bevindt zich ook in '/property/options\_items' onder "layouts".

Het resultaat is een lijst van IDs met benamingen en andere data over de indelingen die het bedrijf heeft. Hieronder zitten standaard indelingen, zoals slaapkamer: ID = 1 en badkamer: ID = 5, en ook custom indelingen met hun eigen ID.

```
JSON
[{"id": 0}, {"id": 1}, {"id": 2}, {"id": 3}, {"id": 4}, {"id": 5}, {"id": 6}
  [
    {
      "company_id": 3149,
      "creation": "2023-12-08T14:19:26+01:00",
      "creation_person_id": -22464,
      "changed": "2023-12-08T14:19:26+01:00",
      "changed_person_id": -22464,
      "id": 7,
      "name": {
        "en": "Living room",
        "nl": "Woonkamer"
      }
    }
  ]
  {"id": 7}
```

Foto toont een voorbeeld van woonkamer, met ID = 7

Deze IDs moeten gekoppeld worden met de layout\_id die men vindt in de pand data onder "layouts". Pand data via endpoint /property/

The screenshot shows a software interface with a tree view of layout data. The root node is 'layouts'. It has two children: '0' and '1'. Node '0' contains several properties: autoid, property\_id, layout\_id, count, surface, floor, creation, creation\_person\_id, changed, changed\_person\_id, company\_id, and sort. Node '1' also contains properties: autoid and property\_id. At the bottom of the interface, there is a navigation bar with icons for back, forward, and search.

- layouts
  - 0
    - autoid : "374600314900005830"
    - property\_id : 3643509
    - layout\_id : 7
    - count : 1
    - surface : 0
    - floor : 0
    - creation : "2023-12-19T11:59:46+01:00"
    - creation\_person\_id : 23226
    - changed : "2023-12-19T11:59:46+01:00"
    - changed\_person\_id : 23226
    - company\_id : 3149
    - sort : 2
  - comment
  - 1
    - autoid : "374600314900005831"
    - property\_id : 3643509

In deze foto gaat het dus over een woonkamer, doordat layout\_id gelijk is aan 7. Andere data die men vindt over een indeling van een pand is opp., verdiep, aantal, sortering en commentaar in alle actieve talen als ze iets ingevuld hebben.

Om het aantal indelingen te tellen, moet men de volgende som doen: Tel alle "counts" op van de items in de indeling lijst die voor een layout\_id voorkomen. Het is namelijk mogelijk dat een bepaalde layout meerdere keren voorkomt in de layouts lijst, sinds men in Zabun dezelfde indelingen kan opsplitsen in meerdere regels, om zo een onderscheid te maken tussen elkaar.

Een voorbeeld: Men maakt een regel voor slaapkamer aan met als commentaar "gelijksvloer" en vult het aantal "2" in. Men maakt nog een regel voor slaapkamer aan met als commentaar "bovenverdiep" en vult het aantal "3" in. Het resultaat is een "layouts" lijst met 2 items. Beide items hebben "layout\_id" = 1, waarvan de ene count = 2 en de andere count = 3 heeft. In totaal zijn er 5 slaapkamers in het pand.

# Hoe maak ik een nieuw pand?

```
curl --location "https://gateway-cmsapi.v2.zabun.be/api/v1/property" \
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'api_key: #api_key#' \
--header "Content-Type: application/json" \
--data "{
    \"office_autoid\": 0,
    \"transaction_id\": 1,
    \"type_id\": 26,
    \"status_id\": 1,
    \"show\": 1,
    \"mandate_type_id\": 1,
    \"mandate_start\": \"2024-05-03T09:39:48.423Z\",
    \"price\": 250000,
    \"responsible_salesrep_person_id\": #X-USER-ID#,
    \"address\": {
        \"city_geo_id\": 1001082,
        \"number\": \"180\",
        \"country_geo_id\": 23,
        \"street_translated\": {
            \"nl\": \"vijfseweg\"
        }
    }
}"
```

- Deze payload toont enkel de verplichte velden, zie swagger voor de volledige payload. Opgelet! Sommige type\_id, transacties, ... kunnen specifieke velden verplicht maken, deze eisen zijn vermeld in swagger en worden duidelijk gemaakt in de response body (400 Bad Request).
- transactie : <https://gateway-cmsapi.v2.zabun.be/api/v1/property/transactions> (<https://gateway-cmsapi.v2.zabun.be/api/v1/property/transactions>)
- type\_id : <https://gateway-cmsapi.v2.zabun.be/api/v1/property/types> (<https://gateway-cmsapi.v2.zabun.be/api/v1/property/types>)
- mandate\_type\_id : <https://gateway-cmsapi.v2.zabun.be/api/v1/property/status> (<https://gateway-cmsapi.v2.zabun.be/api/v1/property/status>)
- type\_id : <https://gateway-cmsapi.v2.zabun.be/api/v1/property/status> (<https://gateway-cmsapi.v2.zabun.be/api/v1/property/status>)
- mandate\_type\_id : [\(https://gateway-cmsapi.v2.zabun.be/api/v1/property/mandate\\_types\)](https://gateway-cmsapi.v2.zabun.be/api/v1/property/mandate_types)
- city\_geo\_id : <https://gateway-cmsapi.v2.zabun.be/api/v1/geo/cities/search> (<https://gateway-cmsapi.v2.zabun.be/api/v1/geo/cities/search>) of  
[https://gateway-cmsapi.v2.zabun.be/api/v1/geo/cities?country\\_geo\\_id={country\\_geo\\_id}](https://gateway-cmsapi.v2.zabun.be/api/v1/geo/cities?country_geo_id={country_geo_id}) ([https://gateway-cmsapi.v2.zabun.be/api/v1/geo/cities?country\\_geo\\_id={country\\_geo\\_id}](https://gateway-cmsapi.v2.zabun.be/api/v1/geo/cities?country_geo_id={country_geo_id}))

- country\_geo\_id=%7Bcountry\_geo\_id%7D) (volledige lijst van geocities per land)
- country\_geo\_id : <https://gateway-cmsapi.v2.zabun.be/api/v1/geo/countries> (<https://gateway-cmsapi.v2.zabun.be/api/v1/geo/countries>)
  - status 1 = actief
  - show 1 = groen huisje (mag doorgestuurd worden naar media's)
  - responsible\_salesrep\_person\_id = je gebruiker id (#X-USER-ID#)

Hoe kan ik weten hoe lang een pand een bepaalde transactie(te koop, te huur, . . . ) heeft?

Onder endpoint <https://gateway-cmsapi.v2.zabun.be/api/v1/property/{ID}> (<https://gateway-cmsapi.v2.zabun.be/api/v1/property/%7BID%7D>) heb je een veld

```

"history_transaction": [
  {
    "transaction_id": 1,
    "property_id": 2091991,
    "date_history": "2020-02-28T14:02:46+01:00",
    "responsible_salesrep_person_id": 18911,
    "responsible_salesrep_bedrijf_id": 3270,
    "company_id": 3270
  },
  {
    "transaction_id": 5,
    "property_id": 2091991,
    "date_history": "2021-07-09T10:58:39+02:00",
    "responsible_salesrep_person_id": 18911,
    "responsible_salesrep_bedrijf_id": 3270,
    "company_id": 3270
  }
]

```

history\_transaction.

hier kan je zien wanneer een pand naar deze transactie is veranderd. de onderste in de lijst is de laatste aangepaste transactie en op datum kan je zien hoelang het pand al deze transactie heeft. Op deze manier kan je de laatste verkocht datum opzoeken. Endpoint <https://gateway-cmsapi.v2.zabun.be/api/v1/property/{ID}/history/transaction> (<https://gateway-cmsapi.v2.zabun.be/api/v1/property/%7BID%7D/history/transaction>) toont dezelfde data als in deze variabelen, waarbij je ook kan filteren en sorteren. Voor de verkocht datum bijvoorbeeld, filter je op transactie ID 5 en sorteert je op descending

# Accounts

Hoe zoek ik mijn accounts op?

```
curl --location --request POST 'https://gateway-cmsapi.v2.zabun.be/api/v1/accounts' \
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'api_key: #api_key#' \
--data-binary ''
```

# Contacten

Hoe zoek ik mijn contacten op?

```
curl --location --request POST 'https://gateway-cmsapi.v2.zabun.be/api/v1/contacts' \
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'api_key: #api_key#' \
--data-binary ''
```

Hoe maak ik een nieuw contact

```
curl --location --request POST 'https://gateway-cmsapi.v2.zabun.be/api/v1/contacts' \
--header 'Content-Type: application/json' \
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'api_key: #api_key#' \
--data-binary '{"last_name": "CROUGHS", "title_id": 1, "status_id": 1, "responsible_salesrep_person_id": 18911, "last_name": "Familie"}'
```

Deze payload toont enkel de verplichte velden, zie swagger voor de volledige payload

- Mogelijk statussen: /api/v1/contact/status
  - 1 = actief
  - 2 = inactief

- . . .
- Mogelijke aansprekingen: **api/v1/contact/titles**
  - 1 = Dhr
  - 2 = Mevr.
- Vertegenwoordiger = responsible\_salesrep\_person\_id = je gebruiker id
- Familie naam = last\_name

Ook nuttige zijn

- first\_name
- email
- mobile + mobile\_cc
- can\_be\_mailed: Zet dit op "true" om de contact aan te melden als "mag gemaild worden"
- is\_opted\_in": Zet dit op "true" om aan te geven dat de contact akkoord gaat met het privacy beleid
- ... zie swagger voor meer

## Hoe maak ik een contact die ook kandidaat is bij een pand

```
curl --location --request POST 'https://gateway-cmsapi.v2.zabun.be/api/v1/contact' \
--header 'Content-Type: application/json' \
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'api_key: #api_key#' \
--d '{
    "contact": {
        "email": "ben@zabun.be",
        "first_name": "Ben",
        "last_name": "Croughs",
        "marketing_opt_in": true, //Is contact akkoord met privacy beleid?
        "mailing_opt_in": true, //Mag Contact Gemailed worden?
        "phone": "496999632",
        "phone_cc": "32",
        "language": "NL"
    },
    "message": {
        "text": "Kopen",
        "property_id": 3309172,
        "info": [
            "info1",
            "info2",
            "info3"
        ]
    }
}'
```

## Belangrijke of verplichte velden

- **contact** = eigenschappen van het contact
  - **email** of **phone** + **phone\_cc** (één van de twee is verplicht)
  - **last\_name** (verplicht)
  - **first\_name** (niet verplicht maar handig)
  - **language** (verplicht)
  - **marketing\_opt\_in** = Privacybeleid. Geeft aan of dat contact akkoord is hiermee of niet. Mag je niet zomaar op true zetten, zonder dat contact ervan op de hoogte is via een disclaimer of het zelf kan aangeven via een vinkje
  - **mailing\_opt\_in** = Mag Contact Gemaild worden. Laat de contact toe gemailed te mogen worden mariages/marketing emails? Mag je niet zomaar op true zetten, zonder dat contact ervan op de hoogte is via een disclaimer of het zelf kan aangeven via een vinkje
- **Message** = het formulier gedeelte.
  - **text** = Wat de contact heeft ingevuld als vraag of beschrijving bij het contactformulier. M.a.w. het bericht
  - **property\_id** = de autoid van het pand waarvoor de contact een aanvraag uitstuurt
- ... zie swagger

Deze endpoint kan ook een standaard zoekfiche aanmaken op basis van het opgegeven pand (budget, type, transactie, ...), indien deze automatische zoekfiche optie is ingeschakeld in uw Zabun Admin.

**Contact > Question** is een oude variabele die enkel nog maar als backup wordt gebruikt voor als **Message > text** leeg is. Gelieve dus **text** te gebruiken. **Message > info** kan gebruikt worden om extra informatie of vragen toe te voegen. Dit is handig voor als uw keuze opties (=checkboxes) toevoegt aan uw contactformulier. Deze worden na de vraag in de taak gezet in volgende formaat: Vraag + "Vraagt info over:" + elke item in message.info, gesplitst met een ','.

Voorbeeld: **Message.text** = "Ik wil graag een huis kopen", **message.info:** ["Ik wil A weten", "Ik wil B zien"] Resultaat beschrijving taak (contactformulier) = Ik wil graag een huis kopen. Vraagt info over: Ik wil A weten, Ik wil B zien

# Hoe maak ik een nieuw contact + bijhorende zoekfiche

```
curl --location --request POST 'https://gateway-cmsapi.v2.zabun.be/api/v1/contact'
--header 'Content-Type: application/json' \
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'api_key: #api_key#' \
--d '{
    "contact": {
        "last_name": "CROUGHS1",
        "email": "ben@zabun.be",
        "language": "NL",
        "mailing_opt_in": true, //Mag Contact Gemailed worden?
        "marketing_opt_in": false, //Is contact akkoord met privacy beleid?
    },
    "request": {
        "price": {
            "min" : 250000,
            "max" : 750000
        },
        "sales_rep": 18911,
        "bedrooms": {
            "min": 1,
            "max": 3
        },
        "bathrooms": {
            "min": 1,
            "max": 3
        },
        "price": {
            "min": 1000,
            "max": 500000
        },
        "cities": [
            1001082 //Waregem
        ],
        "surface": {
            "min": 500,
            "max": 9000
        },
        "transactions": [
            1 //te koop
        ],
        "types": [
            1 //appartement
        ]
    }
}'
```

Deze payload toont enkel de verplichte velden, zie swagger voor de volledige payload. Hieronder nog extra uitleg voor de verplichte of belangrijke velden

- **contact** = eigenschappen van het contact
  - **email** of **phone** + **phone\_cc** (één van de twee is verplicht)
  - **last\_name** (verplicht)
    - **first\_name** (niet verplicht maar handig)
  - **language** (verplicht)
  - **marketing\_opt\_in** = Privacybeleid. Geeft aan of dat contact akkoord is hiermee of niet. Mag je niet zomaar op true zetten, zonder dat contact ervan op de hoogte is via een disclaimer of het zelf kan aangeven via een vinkje
  - **mailing\_opt\_in** = Mag Contact Gemaild worden. Laat de contact toe gemailed te mogen worden mariages/marketings emails? Mag je niet zomaar op true zetten, zonder dat contact ervan op de hoogte is via een disclaimer of het zelf kan aangeven via een vinkje
- **request** = eigenschappen de zoekfiche
  - **sales\_rep** = je gebruiker id
  - **price > min + max** = budget
  - **surface > min + max** = gewenste oppervlakte
  - **bedrooms > min + max** = aantal slaapkamers
  - **cities > lijst van geo\_city\_ids**, waarin de contact wilt zoeken. De geo\_city\_ids vind je in [/api/v1/geo/cities](#)
  - **transaction > lijst van transaction\_ids**, zoals te koop of te huur. De transacties vind je in [/api/v1/property/transactions](#)
  - **types > lijst van type\_ids**, zoals "woning", "villa", "duplex", "bouwgrond", ... De types vind je in [/api/v1/property/types](#)
  - **headtypes > lijst van head\_type\_ids**, zoals "appartement", "huis", "grond", waarvoor de contact wilt zoeken. Dit zijn de overkoepelende "hoofd" types die de gewone types samen groepeert. De hoofdtypes vind je in [/api/v1/property/headtypes](#)
  - ... zie swagger

Voor beide de contact en de zoekfiche is de status altijd default "actief". Je kan dit niet meegeven in deze request. Met de nodige permissies, kan u deze wel achteraf patchen.

## Hoe voeg ik een bestand toe bij een contact

Via [/api/v1/contact/{contact\\_autoid}/files](#) kan een bestand opladen via de API. Als **data/body** van de request, geeft u het effectieve bestand door met key **file** als **form-data**

Hieronder een voorbeeld van in POSTMAN

```
curl --location 'https://gateway-cmsapi.v2.zabun.be/api/v1/contact/xxxxxxxxxx'
\--header 'X-CLIENT-ID: 3210'
\--header 'CLIENT_ID: f21a2c33-6d9e-401f-89fc-6c991dacc504'
\--header 'SERVER_ID: 6b5535a6-c71b-4c8f-ba77-8ef58cc29f7f'
\--header 'API_KEY: 264b23e7-c812-40ae-a2d1-fb978eeb9b34'
\--header 'Content-Type: application/json'
\--form 'file=@"/C:/Uw/Mappen/Folder/bestand_naam.extensie"'
```

Bij het gebruik van uw eigen client in een programmeer taal, zal u uiteraard het bestand moeten toevoegen aan de request volgens de standaarden van de gekozen taal. (Bijvoorbeeld een multipart/form-data request en het gebruik van een filestream)

## Taken

Bij het updaten van de vertegenwoordiger van een taak moet u opletten voor de nieuwe **task\_autoid**. Er wordt namelijk een nieuwe taak met een nieuwe **task\_autoid** gegenereerd, waarbij de oude taak met de meegegeven **task\_autoid** in de patch URL verwijderd wordt. Dit is uitzonderlijke logica die enkel bij taken en voor veld **responsible\_salesrep\_person\_id** uitgevoerd wordt. Andere velden patchen verandert de **task\_autoid** niet.

## Zoekfiches

Hoe zoek ik mijn zoekfiches ?

```
curl --location --request POST 'https://gateway-cmsapi.v2.zabun.be/api/v1/req
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'api_key: #api_key#' \
--data-raw ''
```

Hoe kan ik een gemeente toevoegen aan een zoekfiche verzoek ?

om een zoekfiche aan te maken, kan je een lijst van cityids meegeven, als je een postcode hebt, dan zal je deze moeten vertalen naar cityids, dat kan je door eerst onderstaande request te doen, country\_geo\_ids = id van België, vast = 23, en dan kan je in zipcode een lijst van postcode zetten

```
curl --location --request POST 'https://gateway-cmsapi.v2.zabun.be/api/v1/ge
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'Client_ID: #client_id#' \
--header 'Server_ID: #server_id#' \
--header 'API_Key: #api_key#' \
--header 'Content-Type: application/json' \
--data-raw '{ filtering: {
    "zip_codes": [ 3800, 3840 ],
    "country_geo_ids": [23],
},
"paging": {
    "page": 0,
    "size": 10
},
"sorting": {
    "order": "ASC"
},
}'
```

en dan via onderstaande call kan je de fiche zelf aanmaken

```
curl --location --request POST 'https://gateway-cmsapi.v2.zabun.be/api/v1/con
--header 'Content-Type: application/json' \
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'api_key: #api_key#' \
--d '{

    "contact": {
        "last_name": "CROUGHS1",
        "title": 1,
    },
    "request": {
        "price": {
            "min" : 250000,
            "max" : 750000
        },
        "sales_rep": 18911,
        "cities": [
            "#city-ids#"
        ],
    },
}

}'
```

dan heb je een lijst van gemeenten en deelgemeenten, die moet je dan doorgeven aan de contactrequest api functie

# Het verschil tussen hoofdtyperen en types

Zoekfiches werken op basis van pand types. Als een hoofdtyper wordt aangeduid in Zabun, dan worden alle types onder die hoofdtyper opgeslagen in het zoekfiche. Dezelfde logica wordt toegepast op het aanmaken van een zoekfiche via de API.

Bij het aanmaken van een zoekfiche kan je een lijst van pand hoofdtyper "headtypes" en/of pand types "types" doorgeven. Indien je een hoofdtyper id invult in de headtypes lijst, dan worden automatisch alle types van die hoofdtyper toegevoegd aan de zoekfiche. Als je een type id invult in de types lijst, dan wordt enkel die ene type toegevoegd aan de zoekfiche. Terwijl wordt ook de hoofdtyper zichtbaar in de zoekfiche in Zabun, omdat je een type onder deze hoofdtyper hebt gekozen. De hoofdtyper zelf wordt niet opgeslagen in de zoekfiche.

Dit betekent dat, als je maar 1 of meerdere types van een hoofdtyper wilt toevoegen aan een zoekfiche, maar ze niet allemaal, dat je dan al de nodige type\_ids moet toevoegen aan "types", zonder de headtype\_id toe te voegen aan "headtypes". Je laat dan "headtypes" leeg.

Je kan uiteraard mengelen door bijvoorbeeld de headtype\_id van Huis toe te voegen aan de headtypes lijst voor alle huis types toe te voegen, en dan een specifieke grond type\_id in te vullen in de "types" lijst, zodat enkel die ene grond type wordt toegevoegd aan de zoekfiches i.p.v. ze allemaal.

```
curl --location --request POST 'https://gateway-cmsapi.v2.zabun.be/api/v1/contactrequest' \
--header 'Content-Type: application/json' \
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'api_key: #api_key#' \
--d '{
    "contact": {
        "last_name": "CR0UGHHS1",
        "title": 1,
    },
    "request": {
        "price": {
            "min" : 250000,
            "max" : 750000
        },
        "sales_rep": 18911,
        "cities": [
            "#city-ids#"
        ],
        "headtypes": [3], //House = alle types onder hoofdtype Huis
        "types": [19] //Landbouwgrond. Hoofdtype grond zal zichtbaar zijn in
    }
}'
```

## Toevoegen van extra filters op de zoekfiche

Mits de nodige permissies, is het mogelijk om een zoekfiche aan te passen en extra filters/parameters toe te voegen aan de zoekfiche, zodat de gevonden panden gerichter kunnen gezocht worden voor de contact. Hieronder een voorbeeld hoe je een indeling filter kan toevoegen op een zoekfiche die gemaakt is via de basis /contactrequest endpoint

### *PreSetup*

1. Via `/request option_items > parameters object`, of `/request parameters` kunnen alle mogelijke filters voor een zoekfiche opgehaald worden. In geval van indeling, zoek je achter de objecten met type: **LAYOUT**. Deze hebben allemaal **ID = 13**. Je kan ook enkel de layout parameter filters ophalen via `/request parameters?parameter_id= 13`. Voor de **PATCH request** hebben we de autoid nodig van deze objecten. Het is hier van belang dat je bijhoudt welke hoofdtypes je doorgegeven hebt bij de creatie van de zoekfiche, sinds dat de indeling filters bestaat op basis van hoofdtype, en je deze **head\_type\_ids** zal nodig hebben voor de correcte autoids te verkrijgen. Headtypes van panden vind je via `/property/head_types`  
Voorbeeld

- Hoofdtype Appartement = ID 1 => Dit komt overeen met request filter parameter autoid 97.
- Hoofdtype Huis = ID 3 => Dit komt overeen met request filter parameter autoid 98.
- Hoofdtype grond heeft geen indeling filter en vind je dus niet terug in de parameter lijst

2. De beschikbare indelingen voor uw bedrijf, vind je via **/property/layouts**. De "ID" is hiervan belang.

*Zoekfiche aanmaken en aanpassen voor layout*

1. Maak een basis contact + zoekfiche aan via de **/contactrequest**. Als resultaat krijg je de **request\_autoid**.
2. Voor de zoekfiche aan te passen, gebruik je **/request/{request\_autoid}** van uit puntje 1}
3. In de body vul je volgende variabele toe om de indeling filters aan te passen

```
{"detail_layouts": [{hier komen de layout objecten waarvoor je de filter wil
```

4. Per layout en per hoofdtype, moet je hier een layout filter object toevoegen. Het object ziet er als volgt uit

```
{"count": x, = aantal van deze indeling waarvoor gezocht wordt = exact nummer  
"layout_Id": y, = ID te vinden in /property/layouts  
"parameter_autoid": z = autoid te vinden in /request/parameters  
}
```

Bijvoorbeeld:

- 1 zolder voor hoofdtype huis
- 2 kelders voor hoofdtype appartement
- 3 parkeerplaatsen voor hoofdtype appartement, en 1 voor vakantiewoning.
- Nodige request body

```
{"detail_layouts": [
  {"count": 1,
   "layout_Id": 11, //zolder
   "parameter_autoid": 98 //autoid van layout parameter voor huis
  },
  {
    "count": 2,
    "layout_Id": 3, //kelder
    "parameter_autoid": 97 //autoid van layout parameter voor appartement
  },
  {
    "count": 3,
    "layout_Id": 14, //parkeerplaats
    "parameter_autoid": 97 //autoid van layout parameter voor appartement
  },
  {
    "count": 1,
    "layout_Id": 14, //parkeerplaats
    "parameter_autoid": 101 //autoid van layout parameter voor vakantiewoning
  }
]
```

5. Gelieve geen andere objecten mee te geven, zoals `detail_min_max`, `price_min`, `description...` of andere variabelen, sinds deze ook zullen gewijzigd worden door de patch mits dat ze meegegeven zijn. Ze worden genegeerd als ze niet in de body staan. Indien je niet anders kan dan ze meegeven, door een generated client bijvoorbeeld, vul je ze best volledig in zoals ze nu al waren. Hiervoor zal je best de zoekfiche nog eens apart ophalen via GET `/request/`, waarvoor je permissie de nodig zal hebben.

## Keuzelijsten

Alle keuzelijsten hebben hun eigen endpoints, maar je kan ook alle keuzelijsten voor een sectie ophalen via de `/option_items` endpoint

Voorbeeld: [\(https://gateway-cmsapi.v2.zabun.be/api/v1/property/options\\_items\)](https://gateway-cmsapi.v2.zabun.be/api/v1/property/options_items)

Met deze endpoint kan u het aantal requests naar de API verminderen.

## Extra Velden

Het ophalen en invullen van extra velden op pand, contact of account niveau kan op verschillende manieren. Hieronder een stappenplan met een contact dropdown extra veld als voorbeeld

## Stap 1: Ophalen van extra velden

De `/extra_fields` endpoints halen de mogelijke extra velden op voor hun respectievelijke object: `/property/extra_fields`, `/contact/extra_fields`, `/account/extra_fields` of `/extra_fields` voor ze allemaal.

```
curl --location 'https://gateway-cmsapi.v2.zabun.be/api/v1/contact/extra_fields'
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'api_key: #api_key#' \
--header 'Content-Type: application/json'
```

```
[
  {
    "autoid": "352300279100000004",
    "name": "Nieuwsbrief",
    "page": "CONTACT_GENERAL",
    "object": "CONTACT",
    "type": "DROPDOWNLIST",
    "changed_id": 2791,
    "tooltip": ""
  },
  {
    "autoid": "374600279100000023",
    "name": "Zapier Test Veld",
    "page": "CONTACT_GENERAL",
    "object": "CONTACT",
    "type": "TEXT",
    "changed_id": 2791,
    "tooltip": ""
  }
]
```

## Stap 1 bis: Ophalen van dropdown opties.

Er zijn drie mogelijkheden om de optie lijst op te halen van een extra veld met type "DROPDOWNLIST"

1. Je kan de `/extra_field` endpoint extenderen om de lijst als extra data binnen te krijgen. Dit zorgt ervoor dat de request mogelijk wat langer duurt

```
curl --location 'https://gateway-cmsapi.v2.zabun.be/api/v1/contact/extra_field'
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'api_key: #api_key#' \
--header 'Content-Type: application/json'
```

```
{
  "autoid": "352300279100000004",
  "name": "Nieuwsbrief",
  "page": "CONTACT_GENERAL",
  "object": "CONTACT",
  "type": "DROPODOWNLIST",
  "changed_id": 2791,
  "tooltip": "",
  "dropdown_options": [
    {
      "extra_field": 352300279100000004,
      "active": true,
      "company_id": 2791,
      "creation": "2019-12-04T15:07:00+01:00",
      "creation_person_id": 16639,
      "changed": "2019-12-04T15:07:00+01:00",
      "changed_person_id": 16639,
      "id": 4,
      "name": {
        "nl": "Events",
        "fr": "Events",
        "en": "Events"
      }
    },
    {
      "extra_field": 352300279100000004,
      "active": true,
      "company_id": 2791,
      "creation": "2019-12-04T15:07:32+01:00",
      "creation_person_id": 16639,
      "changed": "2019-12-04T15:07:32+01:00",
      "changed_person_id": 16639,
      "id": 5,
      "name": {
        "nl": "Kantoren",
        "fr": "Kantoren",
        "en": "Kantoren"
      }
    }
  ],
  {
    "autoid": "374600279100000023",
    "name": "Zanier Test Veld".
  }
}
```

([https://gateway-cmsapi.v2.zabun.be/api/v1/extra\\_fields?extended=true](https://gateway-cmsapi.v2.zabun.be/api/v1/extra_fields?extended=true)) haalt dan alle extra velden op met hun dropdown opties.

- Via volgende endpoint kan je de dropdown opties apart ophalen per extra veld

```
curl --location 'https://gateway-cmsapi.v2.zabun.be/api/v1/extra_fields/35230
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'api_key: #api_key#' \
--header 'Content-Type: application/json'
```

```
[ {
  {
    "extra_field": 3523002791000000004,
    "active": true,
    "company_id": 2791,
    "creation": "2019-12-04T15:07:00+01:00",
    "creation_person_id": 16639,
    "changed": "2019-12-04T15:07:00+01:00",
    "changed_person_id": 16639,
    "id": 4,
    "name": {
      "nl": "Events",
      "fr": "Events",
      "en": "Events"
    }
  },
  {
    "extra_field": 3523002791000000004,
    "active": true,
    "company_id": 2791,
    "creation": "2019-12-04T15:07:32+01:00",
    "creation_person_id": 16639,
    "changed": "2019-12-04T15:07:32+01:00",
    "changed_person_id": 16639,
    "id": 5,
    "name": {
      "nl": "Kantoren",
      "fr": "Kantoren",
      "en": "Kantoren"
    }
  }
]
```

- Je kan ook via ([https://gateway-cmsapi.v2.zabun.be/api/v1/extra\\_fields/dropdowns](https://gateway-cmsapi.v2.zabun.be/api/v1/extra_fields/dropdowns)) alle dropdown opties ophalen voor alle extra velden. Het resultaat is dezelfde als hierboven. Je moet zelf de dropdown opties groeperen op basis van de **extra\_field** variabele.

## Stap 3. Ophalen van de extra veld waardes die ingevuld zijn bij het object

Dit kan via de `/extra_field_values` endpoints bij de respectievelijke controllers

```
curl --location 'https://gateway-cmsapi.v2.zabun.be/api/v1/contact/3709002791
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'api_key: #api_key#' \
--header 'Content-Type: application/json'
```

```
[  
  {  
    "object_key_autoid": "3709002791000000010",  
    "extra_field_autoid": "3523002791000000004",  
    "value_ddl": 5,  
    "value_string": "",  
    "value_number": 0.0,  
    "value_boolean": false,  
    "value_date": null,  
    "creation": "2024-09-11T17:19:02+02:00",  
    "creation_person_id": 9993,  
    "changed": "2024-09-11T17:22:22+02:00",  
    "changed_person_id": 9993,  
    "company_id": 2791,  
    "type": "DROPDOWNLIST",  
    "value": "5"  
  }]  
]
```

Voor de waarde op te halen, gebruik je ofwel de respectievelijke "value\_xxxx", of kan je de generieke `value` variabele gebruiken. In het voorbeeld is de `type` "DROPDOWNLIST" en kan je `value_ddl` gebruiken, maar ook `value` die dezelfde waarde heeft

Hieronder een lijst van de koppeling tussen type en waarde

- TEXT => value\_lang\_string (of value\_string indien taal niet van belang is)
- INTEGER => value\_number
- BOOLEAN => value\_boolean
- DATETIME => value\_date
- BIGTEXT => value\_lang\_string

- `NULLABLE_BOOLEAN` => `value_ddl` (uitzonderlijk!)
- `DATE` => `value_date`
- `DROPODOWNLIST` => `value_ddl`
- `DOUBLE` => `value_number`
- `MULTIDROPODOWNLIST` => `value_multi_ddl`
- `AUTOCOMPLETE_ACCOUNTCONTACT` => `value_string`
- `AUTOCOMPLETE_ACCOUNT` => `value_string`
- `AUTOCOMPLETE_CONTACT` => `value_string`
- `AUTOCOMPLETE_PROPERTY` => `value_string`
- `AUTOCOMPLETE_PROJECT` => `value_string`

## Stap 4. Invullen of aanpassen van een extra veld waarde

1. Dit kan via de `PUT .../extra_field_values` endpoints bij de respectievelijke controllers. De belangrijke velden in de body zijn `extra_field_autoid` en `value_xxxx`.
  - `TEXT` of `BIGTEXT` = vul `value_lang_string` in (geen `value_string`, deze wordt nog niet ondersteund)
  - `INTEGER` of `DOUBLE` = vul `value_number` in
  - `DROPODOWNLIST` = vul `value_ddl` in

```
curl --location --request PUT 'https://gateway-cmsapi.v2.zabun.be/api/v1/cont
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'api_key: #api_key#' \
--header 'Content-Type: application/json'
--data '{
    "extra_field_autoid": 352300279100000004,
    "value_ddl": 5
}'
```

U hoeft geen `object_key_autoid` in te vullen. Indien u dit toch wilt doen, moet u zorgen dat deze gelijk is aan de ID die u gebruikt in de URL van de endpoint. In het voorbeeld moet `object_key_autoid` = 3709002791000000010

Op dit moment kan je de generieke `value + type` variabele niet gebruiken om de waarde in te vullen, je moet de specifieke gebruiken. Volgende waarden werken maar zijn nog niet uitvoerig getest. Indien u een van deze wilt gebruiken, neem contact op met de support zodat uw gevraagde waarde kan afgecheckt worden, of gebruik het op eigen risico.

- `DATE` of `DATETIME` = vul `value_date` in
- `MULTIDROPODOWNLIST` of `NULLABLE_BOOLEAN` = vul `value_ddl` in
- `AUTOCOMPLETE_xxxx` = vul `value_string` in

2. Je kan ook via de `extra_field` controller een waarde invullen voor een extra veld van eender welk object. Bij deze endpoint moet u wel de `object_key_autoid` variabele invullen met de id van het pand, de contact of het account. De validatie controleert of deze bestaat

```
curl --location --request PUT 'https://gateway-cmsapi.v2.zabun.be/api/v1/extr
--header 'X-CLIENT-ID: #X-CLIENT-ID#' \
--header 'client_id: #client_id#' \
--header 'server_id: #server_id#' \
--header 'api_key: #api_key#' \
--header 'Content-Type: application/json'
--data '{
  "extra_field_autoid": 352300279100000004,
  "object_key_autoid": 370900279100000010,
  "value_ddl": 5
}'
```

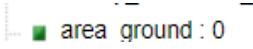
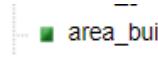
## Korte vragen met bondige antwoorden

**Q:** Ik krijg een validatie error met als melding "Error converting value xxxx to type 'System.Int64 []'.

**A:** U stuurt een waarde, zoals een id, als apart door terwijl de variabele/filter een lijst verwacht. Zet het ID tussen []. Bijvoorbeeld `office_autoids => "office_autoids: [xxxxxxxxxx]"` (foutief: `"office_autoids: xxxxxxxxxx"`)

**Q:** Grondoppervlakte, bewoonbare oppervlakte, kadastraal inkomen

**A:** op te vragen via url: [\(https://gateway-cmsapi.v2.zabun.be/api/v1/property/xxxxx\)](https://gateway-cmsapi.v2.zabun.be/api/v1/property/xxxxx)

- Grondoppervlakte  area `ground: 0`
  
- bewoonbare oppervlakte  area `build: 121`

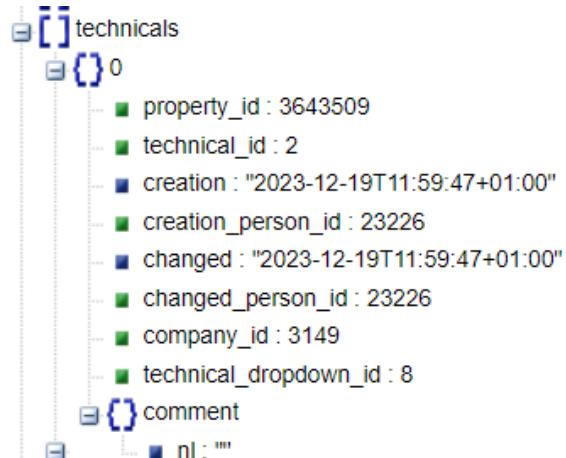
- kadastraal inkomen

## Q: Bouwtechnische informatie

Te vinden onder de tab technisch in Zabun. Waar via de API?

A: op te vragen via url: <https://gateway-cmsapi.v2.zabun.be/api/v1/property/xxxxxx>

(<https://gateway-cmsapi.v2.zabun.be/api/v1/property/xxxxxx>)



<https://cmsapi.v2.zabun.be/api/v1/property/xxxxxx>)

- Nieuw: Enkel ingevulde technicals worden getoond. Technicals zonder type, evaluatie en commentaar worden weg gefilterd
- Technical\_id is de identificatie van de bouwtechnische item. De benaming kan je opvraging via url: (<https://gateway-cmsapi.v2.zabun.be/api/v1/property/technical>)

```

      "fr": "Toit"
    }
  },
  {
    "company_id": ,
    "creation": "2017-10-19T16:47:19+02:00",
    "creation_person_id": -2673,
    "changed": "2017-10-19T16:47:19+02:00",
    "changed_person_id": -2673,
    "id": 2,
    "name": {
      "fr": "Châssis",
      "nl": "Raamwerk",
      "cz": "Raamwerk"
    }
  },
  {
    "company_id": ,
    "creation": "2020-01-28T14:42:23+01:00",
    "creation_person_id": -2673,
    "changed": "2020-01-28T14:42:23+01:00",
    "changed_person_id": -2673,
    "id": 3,
    "name": {
      "fr": "Porte d'entrée",
      "nl": "Entreepoort",
      "cz": "Vchod"
    }
  }
]
  
```

De mapping gebeurt tussen technical\_id op pand niveau en id uit deze option call. In dit voorbeeld komt de bouwtechnische item dus overeen met "Raamwerk".

- De gekozen waarde/type voor de technical = technical\_dropdown\_id. Dit is bijvoorbeeld het ID voor "PVC" bij "Raamwerk". De benamingen kan opgehaald worden via

- voor alle bouwtechnische items: <https://gateway-cmsapi.v2.zabun.be/api/v1/property/technicals/dropdowns>. U moet hier dan ook mappen met de technical\_id property
- voor 1 specifiek bouwtechnisch item: [https://gateway-cmsapi.v2.zabun.be/api/v1/property/technicals/{technical\\_id}/dropdowns](https://gateway-cmsapi.v2.zabun.be/api/v1/property/technicals/{technical_id}/dropdowns).

```
[  
 {  
   "technical_id": 2,  
   "company_id": ,  
   "creation": "2021-10-28T14:09:16+02:00",  
   "creation_person_id": -2673,  
   "changed": "2021-10-28T14:09:16+02:00",  
   "changed_person_id": -2673,  
   "id": 8,  
   "name": {  
     "cz": "PVC",  
     "nl": "PVC",  
     "fr": "PVC"  
   }  
 },
```

- De gekozen evaluatie = evaluation\_id
  - benamingen via <https://gateway-cmsapi.v2.zabun.be/api/v1/property/technicals/evaluations>
- Commentaar is ingevuld voor alle actieve talen van het bedrijf

Technicals, technical\_dropdowns en evaluation lijsten zijn ook terug te vinden onder [https://gateway-cmsapi.v2.zabun.be/api/v1/property/option\\_items](https://gateway-cmsapi.v2.zabun.be/api/v1/property/option_items).

## Q: WETTELIJKE INFO

EPC Label Alle gewenste EPC info kan ik vinden via de API, behalve de label. Op Zabun zie ik wel een label staan.

**A:** op te vragen via url: <https://gateway-cmsapi.v2.zabun.be/api/v1/property/xxxxxx> (<https://gateway-cmsapi.v2.zabun.be/api/v1/property/xxxxxx>). Indien de vertegenwoordiger manueel een label heeft gekozen voor het pand, dan bevat variabele **custom\_epc\_label** deze label als pure tekst. (Bijvoorbeeld A, A+, A plus, D, E, ...)

```
custom_epc_label : ""
```

Indien bovenstaande veld leeg is, dan werd de label berekend. U zal zelf een

label moeten berekenen voor uw website op basis van de waarde die te vinden is  
 in variabele **epc\_value**

- **K\_label**: 0
- **epc\_value**: 325

**Q:** Via de API krijg ik EPC Certificate 1 terug, hoe weet ik welke waarde voor welke Label staat of staat dit er los van?

EPC-Certificaat	Ja
<b>A:</b> dit veld is gemapped met dit veld	

De referentie van het epc document is te vinden in volgende variabele

- **epc\_reference**: "20221108-0002717126-RES-1"

**Q:** Wat is het verschil tussen de gewone EPC velden en de \_Shared velden? Waarom bestaat er een **epb\_reference** maar geen **epb\_value** of **custom\_epb\_label**?

**A:** De gewone epc en epb zijn voor volledige huizen of uw eigen appartement, de \_shared versies zijn voor het gehele gebouw bij een appartement of andere properties waar er gedeelde layout is.

Alle epc velden worden gebruikt voor de epb waarden, behalve de referentie. Dit komt omdat deze andere regels heeft qua formaat die niet samen kan geprogrammeerd worden met het epc referentie veld. De aparte regels voor de waarde en de label van de epb is wel geprogrammeerd in de epc velden.

## Q: Waardes van overstromingsgevoeligheid

A:	Overstromingsgevoeligh d	Niet overstromingsgevoelig
----	-----------------------------	----------------------------

o\_level\_flooding\_sensitivity\_id : 3

de tekst kan je opvragen via url: [https://gateway-cmsapi.v2.zabun.be/api/v1/property/flooding\\_sensitivities](https://gateway-cmsapi.v2.zabun.be/api/v1/property/flooding_sensitivities) ([https://gateway-cmsapi.v2.zabun.be/api/v1/property/flooding\\_sensitivities](https://gateway-cmsapi.v2.zabun.be/api/v1/property/flooding_sensitivities))

```

[{"id": 1, "name": {"nl": "Effectief overstromingsgevoelig", "en": "Recognized flood zone"}}, {"id": 3, "name": {"nl": "Niet overstromingsgevoelig", "en": "Non-flood zone"}}
]
  
```

in dit geval betekend 3 niet overstromingsgevoelig.

Afgebakende oeverzone	Niet van toepassing
-----------------------	---------------------

o\_level\_flooding\_zone\_id : 3

de tekst kan je opvragen via url: [https://gateway-cmsapi.v2.zabun.be/api/v1/property/flooding\\_zones](https://gateway-cmsapi.v2.zabun.be/api/v1/property/flooding_zones) ([https://gateway-cmsapi.v2.zabun.be/api/v1/property/flooding\\_zones](https://gateway-cmsapi.v2.zabun.be/api/v1/property/flooding_zones))

```

[{"id": 3, "name": {"nl": "Niet van toepassing", "en": "Not applicable"}}
]
  
```

in dit geval betekend 3 niet van toepassing.

## Q: Omgevingsvergunning voor stedenbouwkundige handelingen

Zou dit hetzelfde zijn als Dagvaarding voor stedenbouwkundige overtreding? Hoe kan ik hier dan via de API juist aan?

A: Dagvaarding voor stedenbouwkundige overtreding Geen rechterlijke herstelmaatregel of bestuurlijke maatregel opgelegd

is gemapped met town\_planningViolationId : 0

om te weten wat de id betekend kan je dit opvragen via url: [https://gateway-cmsapi.v2.zabun.be/api/v1/property/town\\_planning\\_violations](https://gateway-cmsapi.v2.zabun.be/api/v1/property/town_planning_violations) ([https://gateway-cmsapi.v2.zabun.be/api/v1/property/town\\_planning\\_violations](https://gateway-cmsapi.v2.zabun.be/api/v1/property/town_planning_violations))

```

[{"id": 0, "name": {"nl": "Geen rechterlijke herstelmaatregel of bestuurlijke maatregel opgelegd", "en": "No judicial remedial or administrative measure imposed"}, "abbreviation": "Gmo"}, {"id": 1}, {"id": 2}, {"id": 3}, {"id": 4}, {"id": 5}]
  
```

in dit geval betekend 0 Geen rechterlijke herstelmaatregel of bestuurlijke maatregel opgelegd.

## Q: Verkooprechtdienst, Handhaving

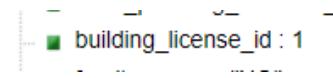
Kan dit precies ook niet in Zabun terugvinden. Onder wat kan ik dit moet dit toch ergens staan, terugvinden via de API?

A: Recht van voorverkoop Neen

deze is gemappt met presale\_right\_ynu : NO

**Bouwvergunning****Bouw vergund**

deze is gemaped met



om te weten wat deze id betekend kan je dit opvragen via de url:

[https://gateway-cmsapi.v2.zabun.be/api/v1/property/building\\_licenses](https://gateway-cmsapi.v2.zabun.be/api/v1/property/building_licenses)  
[https://gateway-cmsapi.v2.zabun.be/api/v1/property/building\\_licenses](https://gateway-cmsapi.v2.zabun.be/api/v1/property/building_licenses))

**JSON**

0

id : 1

name

nl : "Bouw vergund"

en : "Building licensed"

1

2

3

in dit geval betekend 1 bouw vergund

**Q:** Omgevingsvergunning voor het verkavelen van gronden

Zou dit dan gaan om verkavelingsvergunning? Hoe kan ik hier dan aan via de API?"

**Verkavelingsvergunning**

selecteer

**A:**

presale\_ynu : NO

allocation\_license\_ynu : "UNKNOWN"

is gemapped met

**Q:** Waar vind ik mijn vertegenwoordigers, en heb ik rechten tot accounts nodig ?

**A** Omtrent accounts/gebruikers: ze moeten </api/v1/person> gebruiken voor de vertegenwoordigers. Rechten tot accounts is dus niet nodig

**Q:** Wat zijn de mogelijk waardes van overstroomingsgevoeligheid ?

**A** op pand niveau bestaat **flooding\_building\_score** (g score) en **flooding\_parcel\_score** (p score). Deze geven de effectieve waarde terug, A, B, C, D of 0 voor onbekend .

**Q:** Waar vind ik verkoopsrecht, verkavelingsvergunning, gerechterlijke of bestuurlijk maatregel

**A** Voorkooprecht **presale\_right\_ynu** (string, optional): Recht van voorkoop = ['YES', 'NO', 'UNKNOWN']

Verkavelingsvergunning **allocation\_license\_ynu** (string, optional): Verkavelingsvergunning = ['YES', 'NO', 'UNKNOWN']

Tip: Alle **\_ynu** velden zijn variabelen met als waarde "Ja", "Nee", of "Onbekend", in het engels

Gerechterlijke en bestuurlijke maatregel ik neem aan dat hiermee "Dagvaarding voor stedenbouwkundige overtreding" wordt bedoelt **town\_planningViolation\_id** (integer, optional)

Deze bevat slechts de ID, de mapping kan opgehaald worden onder /api/v1/property/town\_planning\_violations

**Q:** Ik heb bv. 15 records in zabun maar zie er maar bv. 5 in api response?

kijk of je page parameter start met index 0. Zie 'Hoe sorteer, filter en doe ik aan paging?'

**Q:** Indien een pand wordt toegevoegd aan een project, wordt de gewijzigd datum van het project aangepast?

**A:** Ja

**Q:** Wat is het nut van de sales\_rep variabele (=vertegenwoordiger/makelaar) in een zoekfiche en wat gebeurt er als deze niet is ingevuld bij de request?

**A:** De sales\_rep bij zoekfiche is handig voor als je een specifieke vertegenwoordiger hebt die binnenkomende zoekfiches behandeld, of als men met regios werkt. Het is niet verplicht, maar betekent wel dat de zoekfiche in Zabun terecht komt zonder vertegenwoordiger. Die kan uiteraard in Zabun of via de API alsnog ingevuld worden via een update.

**Q:** Welke zaken moeten wij extra in orde brengen na het live brengen van onze integratie van de Api?

**A:** U hoeft verder niets door te geven om uw implementatie te voltooien. Wij hebben geen extra configuratie nodig. U kan wel het URL formaat van de pand pagina op uw website doorgeven voor de functionaliteit van het oogje in de media grid bij een pand op Zabun. Zorg hier wel voor een url formaat die gebruik maakt van de Zabun Property Ids. Dit is echter niet nodig, sinds u bij het doorgeven van de publicatie status van het pand ook de URL kan doorgeven, volledig in uw eigen formaat.

**Q:** Moeten bestanden en foto's (=Media) gedownload worden en mogen de directe URL gebruikt worden.

**A:** Wij raden ten sterkste aan om kopieën te maken van de publieke foto's en bestanden die weergegeven moeten worden bij het pand op uw website. U mag de directe URLs gebruiken. Echter, als de makelaar de foto/het bestand verwijdert uit Zabun, of wij doen een aanpassing aan ons bestanden systeem, dan zal de foto/het bestand niet meer zichtbaar zijn (indien deze ook verdwenen is uit de cache) en is dit direct zichtbaar op uw website, waarbij panden zonder foto's een mogelijkheid kan zijn. U kan dit verhelpen door de data opnieuw te ophalen en de geldige urls te verkrijgen.

**Q:** Mogen panden 'on the fly' opgevraagd worden? Of is het de bedoeling dat alle data opgehaald wordt in een eigen databank?

**A:** Panden kunnen 'on the fly' opgehaald worden maar we raden dit ten strengste af door volgende redenen

1. De API is ingesteld met een rate-limit, waardoor u maar een beperkt aantal requesten kan uitsturen per minuut. Indien uw veel verkeer ontvangt op uw website, kan dit betekenen dat u dit limit bereikt en tijdelijk geen requests meer mag uitsturen.
2. Indien er een fout ontstaat op onze API of deze is gaat offline, bijvoorbeeld door een update of interne problemen, dan hebt u tijdelijk geen data meer en zullen de panden niet meer zichtbaar zijn op uw website
3. De snelheid van uw website wordt bepaald door de snelheid van de requests naar de API. Indien er netwerk problemen zijn of onze API heeft vertraging tijdens piekmomenten, zal dit direct zichtbaar zijn op uw website. U voorziet ofwel een database of ten minste een cache die de panden data bijhoudt. Beide de "Property Search" en de "Property Media Sync" accepteren een "updated\_since" filter om de laatste gewijzigde panden op te halen

Dit is ook van toepassing op de foto's en bestanden van het pand, zoals we al vermelden in de vraag hierboven.

**Q:** Hoe haal ik enkel projecten op? Hoe haal ik enkel "nieuwbouw" panden op?

Om enkel projecten op te halen, kan je op twee manieren werken.

1. Je kan aan de "Property Search" de filteren "**kind": "true**" meegeven om enkel projecten binnen te halen.
2. Je kan zelf in de lijst van panden filteren op variabele **is\_project = true**

Om nieuwbuw panden op te halen, m.a.w. panden met eigenschap "nieuwbouw", moet je zelf in de panden lijst filteren op variabele **development = true**.

Nieuwbouw panden kunnen beide gewone panden en projecten zijn. Als u enkel nieuwbuw projecten wil, m.a.w. een pand van type "project" met eigenschap "nieuwbouw", dan doet u een combinatie van bovenstaande opties

1. "Property Search" met de project filter + zelf filteren op **development = true**
2. Zelf filteren op **is\_project = true** en **development = true**