



SMARTLIFE PROPERTIES · DOCUMENTATION

API developer path

Connect your tools to the platform

Get an OAuth2 token, read your projects and rentals, change the status of a project and fix the most common API errors.

Who is it for

Developers and integrators

Steps

6 articles

Duration

About 30 minutes

What you will learn**Authenticate****Read data****Write and troubleshoot**

1. Get an OAuth2 token
2. List your projects
3. Read a project's data
4. Get the rentals summary
5. Change a project's status
6. Understand common errors

SmartLife Properties — We turn your ideas into reality

<https://construction.my-smartlife.fr> · contact@my-smartlife.fr · +23771494855

Biteng - Yaoundé - Cameroun · Document generated on 09 Oct 2026

Contents

Authenticate	3
1. Get an OAuth2 token	3
Read data	6
2. List your projects	6
3. Read a project's data	8
4. Get the rentals summary	10
Write and troubleshoot	12
5. Change a project's status	12
6. Understand common errors	14

STEP 1 OF 6 · AUTHENTICATE

API — Authentication and OAuth2 tokens

Overview of the API and of the two OAuth2 authentication flows available to get an access token.

Online version: <https://construction.my-smartlife.fr/documentation/article-4-api-authentification-et-jetons-oauth2.html?lang=1>

Overview

The platform exposes an **HTTP API** that lets a third-party application read the data of your projects and rentals. Every endpoint returns **JSON**; all of them are read-only (**GET** method) except the one documented as a write endpoint.

Endpoint base URL (production): <https://construction.my-smartlife.fr/api/>

Every endpoint requires OAuth2 authentication. A call without a token, or with an invalid/expired token, returns a 401 error (the API messages are in French):

```
curl https://construction.my-smartlife.fr/api/projects/list

{"detail":"Jeton OAuth manquant"}
```

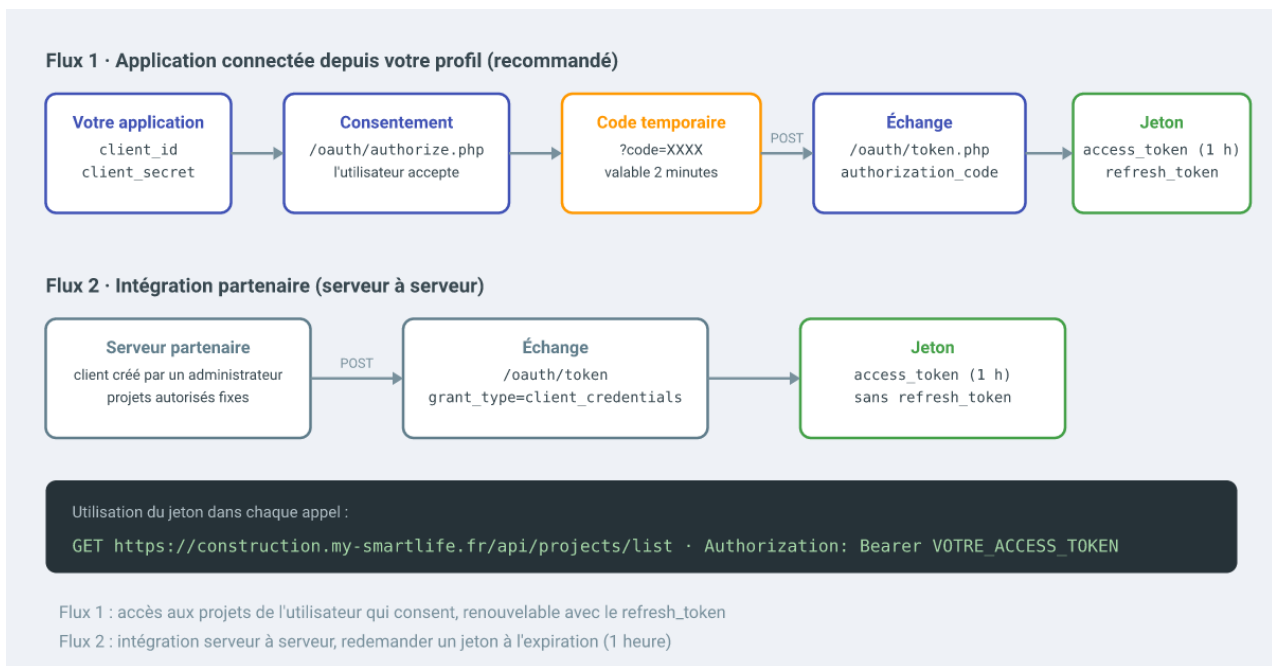
This documentation spans several pages

This page covers authentication. The following pages detail each available endpoint, with a built-in tester to try it directly from your browser.

Authenticate: two possible OAuth2 flows

Depending on your situation, two ways of getting an access token (*access_token*) are available. Both produce a token used the same way, in the HTTP header:

```
Authorization: Bearer YOUR_ACCESS_TOKEN
```



The two ways to get a token, then its use in the Authorization header.

Flow 1: application connected from your profile (recommended)

This is the flow to use if you are building an application that must access **your own projects** (or those of a user who gives their consent).

1. Register your application

From your profile, **“OAuth application”** tab, create an application: you get a `client_id` and a `client_secret` (shown only once), as well as a redirect URL (`redirect_uri`) and allowed `scopes`.

2. Ask for the user's authorisation

Redirect the user to the consent screen:

```
GET https://construction.my-smartlife.fr/oauth/authorize.php
?client_id=YOUR_CLIENT_ID
&redirect_uri=YOUR_REDIRECT_URI
&scope=projects:read rentals:read
&state=a_random_value
```

Once accepted, the user is redirected to your `redirect_uri` with a temporary code (valid for 2 minutes): `?code=XXXX&state=...`

3. Exchange the code for a token

```
curl -X POST https://construction.my-smartlife.fr/oauth/token.php \
-d 'grant_type=authorization_code' \
-d 'code=XXXX' \
-d 'client_id=YOUR_CLIENT_ID' \
-d 'client_secret=YOUR_CLIENT_SECRET' \
-d 'redirect_uri=YOUR_REDIRECT_URI'
```

Response:

```
{
  "access_token": "...",
  "refresh_token": "...",
  "token_type": "Bearer",
  "expires_in": 3600,
  "scope": "projects:read rentals:read"
}
```

Data scope

A token obtained through this flow only gives access to **the projects belonging to the account** that accepted the connection — never to those of another user.

4. Renew the token

The token expires after **1 hour**. Use the refresh_token (whose value does not change as long as it is not used) to get a new one without going through the consent screen again:

```
curl -X POST https://construction.my-smartlife.fr/oauth/token.php \
  -d 'grant_type=refresh_token' \
  -d 'refresh_token=YOUR_REFRESH_TOKEN' \
  -d 'client_id=YOUR_CLIENT_ID' \
  -d 'client_secret=YOUR_CLIENT_SECRET'
```

You can revoke access at any time from the “OAuth application” tab of the profile: the current token then stops working immediately.

Flow 2: partner integration (client_credentials)

This flow is reserved for server-to-server integrations for which a Smartlife administrator has created a dedicated client, linked to a fixed list of projects (useful for an internal dashboard or a trusted partner, for example).

An administrator creates the client:

```
python app/create_oauth_client.py partenaire-1 \
  --projects 12,18 \
  --scopes project:read,rental:read
```

Your application then exchanges its credentials for a token:

```
curl -u 'partenaire-1:SECRET' -X POST https://construction.my-smartlife.fr/oauth/token \
  -H 'Content-Type: application/x-www-form-urlencoded' \
  --data 'grant_type=client_credentials&scope=project:read rental:read'
```

This flow has neither a consent screen nor a refresh_token: simply request a new token with the same credentials once it has expired (1 hour).

Ready to try?

Once you have your token (through either flow), go to the next page: you can paste it into the built-in tester and call the real endpoints directly from this documentation.

STEP 2 OF 6 · READ DATA

API — projects/list endpoint

Documentation of the GET /api/projects/list endpoint: required scope, response structure, example call and built-in tester.

Online version: <https://construction.my-smartlife.fr/documentation/article-6-api-endpoint-projects-list.html?lang=1>

GET /api/projects/list

Required scope: projects:read ("OAuth application" flow) or project:read (partner flow).

Returns, for each project accessible to your token, its general information and the **status of each step and sub-step** of the project (current, completed, disabled, ...). Each step and sub-step also carries its name in French, English and Spanish (name_fr, name_en, name_es) — empty for a step outside the standard process.

The **possession** field gives the ownership share of the token's user in this project (useful for a property with several owners). It is null for a partner token (client_credentials, not linked to a user) or if this user is not one of the project's owners.

The country is provided in two forms: country (legacy numeric id) and country_iso / country_name_fr / country_name_en / country_name_es — empty if the id matches none of the 3 countries currently installed (Cameroon, France, Senegal).

```
curl https://construction.my-smartlife.fr/api/projects/list \
-H 'Authorization: Bearer YOUR_ACCESS_TOKEN'
```

Example response:

```
{
  "projects": [
    {
      "id": "12",
      "name": "Résidence Les Palmiers",
      "city": "Abidjan",
      "country": 1,
      "country_iso": "FR",
      "country_name_fr": "France",
      "country_name_en": "France",
      "country_name_es": "Francia",
      "currency": "XOF",
      "possession": 50.0,
      "status": { "label": "en_cours", "legacy_value": 40 },
      "start_date": "2024-01-01",
      "steps": {
        "3": {
          "name_fr": "Avant construction",
          "name_en": "Pre-construction",
          "name_es": "Preconstrucción",
          "status": "current",
          "date": { "begin": "2024-01-01", "end": "" },

```

```

    "sub_steps": {
      "0": { "name_fr": "Définir le projet de construction", "name_en": "Define the construction project", "name_es": "Definir el proyecto de construcción", "status": "completed", "date": { "begin": "", "end": "" } },
      "1": { "name_fr": "Estimer le budget global prévisionnel", "name_en": "Estimate the overall provisional budget", "name_es": "Estimar el presupuesto global previsto", "status": "current", "date": { "begin": "", "end": "" } }
    }
  }
}
]
}

```

Requête

GET `/api/projects/list`

En-têtes

Authorization: Bearer VOTRE_ACCESS_TOKEN

Portée requise

projects:read

Réponse - 200 OK - JSON

```

{
  "projects": [
    {
      "id": "12",
      "name": "Résidence Les Palmiers",
      "currency": "XOF",
      "status": { "label": "en_cours" },
      "steps": { ... }
    }
  ]
}

```

Liste les projets accessibles avec le jeton : identifiant, nom, ville, devise, statut, avancement des étapes
 Portée projects:read (ou project:read selon le type de jeton)

Request and response of the projects/list endpoint.

Test this endpoint

An interactive tester for this endpoint is available in the online version: <https://construction.my-smartlife.fr/documentation/article-6-api-endpoint-projects-list.html?lang=1>

STEP 3 OF 6 · READ DATA

API — project/{id}/data endpoint

Documentation of the GET /api/project/{project_id}/data endpoint: status and aggregated rental data of a single project, with its access rights.

Online version: <https://construction.my-smartlife.fr/documentation/article-9-api-endpoint-project-id-data.html?lang=1>

GET /api/project/{project_id}/data

Required scopes: project:read **and** rental:read.

Unlike /projects/list and /rentals/summary (which go through every project accessible to the token), this endpoint targets **a single project** and combines in one call: its general status, its aggregated rental indicators (contracts, occupancy, unit by unit) and the token's effective rights on this project.

If the project does not exist: 404. If the project is outside the token's scope: 403 (see [Common errors](#)).

```
curl https://construction.my-smartlife.fr/api/project/12/data \
-H 'Authorization: Bearer YOUR_ACCESS_TOKEN'
```

Example response:

```
{
  "project": {
    "id": "12",
    "name": "Résidence Les Palmiers",
    "status": { "label": "en_cours", "legacy_value": 40 },
    "start_date": "2024-01-01",
    "city": "Abidjan",
    "country": 1,
    "currency": "XOF",
    "progress": { "current": ["6"], "completed": ["3"], "disabled": ["9", "12"] }
  },
  "rental": {
    "global": {
      "contracts": 8, "active_contracts": 6,
      "rent_expected": 4860000.0, "rent_paid": 405000.0,
      "charges_expected": 120000.0, "charges_paid": 10000.0,
      "lots": 10, "occupied_lots": 6, "available_lots": 4
    },
    "monthly": {
      "2025-01": { "contracts": 6, "rent_expected": 405000.0, "rent_paid": 405000.0 }
    },
    "lots": [
      {
        "architecture_id": "0", "lot_id": "3", "title": "Appartement B3",
        "floor": "2", "current_type": "long_terme", "public": true,
        "capacity": 1, "occupancy_status": "occupe",
        "global": { "contracts": 1, "active_contracts": 1, "rent_expected": 810000.0, "rent_paid": 67500.0 },
        "monthly": { "2025-01": { "contracts": 1, "rent_expected": 67500.0, "rent_paid": 67500.0 } }
      }
    ]
  }
}
```

```

    }
  ],
  "rights": {
    "read": { "project": true, "rental": true },
    "write": { "project": false },
    "scopes": ["project:read", "rental:read"]
  }
}

```

The **occupancy_status** of a unit is disponible (available, no active contract), occupe (occupied, capacity reached) or partiellement_occupe (partly occupied).

Requête

GET /api/project/{id}/data

En-têtes

Authorization: Bearer VOTRE_ACCESS_TOKEN

Portée requise

project:read

Réponse · 200 OK · JSON

```

{
  "project": {
    "id": "12",
    "status": { "label": "en_cours" },
    "progress": { "current": ["6"] }
  },
  "rental": { "global": {
    "contracts": 8,
    "active_contracts": 6,
    "rent_expected": 4860000.0
  } }
}

```

État et avancement d'un projet, et données locatives agrégées (contrats, loyers, charges)
 404 si le projet n'existe pas, 403 s'il est hors du périmètre du jeton

Request and response of the project/{id}/data endpoint.

Test this endpoint

An interactive tester for this endpoint is available in the online version: <https://construction.my-smartlife.fr/documentation/article-9-api-endpoint-project-id-data.html?lang=1>

STEP 4 OF 6 · READ DATA

API — rentals/summary endpoint

Documentation of the GET /api/rentals/summary endpoint: required scope, response structure, example call and built-in tester.

Online version: <https://construction.my-smartlife.fr/documentation/article-7-api-endpoint-rentals-summary.html?lang=1>

GET /api/rentals/summary

Required scope: rentals:read ("OAuth application" flow) or rental:read (partner flow).

Returns, for each rental project accessible to your token, a financial summary **per year** (field names are in French):

- **attendu** (expected): rent + charges + fees expected over the year, all rental types combined;
- **depenses** (expenses): expenses recorded over the year (overall);
- **encaisse** (collected): amounts actually collected over the year, all types combined;
- **benefices** (profit): encaisse - depenses;
- **taux_occupation** (occupancy rate): average of the year's monthly occupancy rates;
- **taux_recouvrement** (collection rate): encaisse / attendu * 100;
- **par_mois** (per month): month-by-month detail (YYYY-MM key) of attendu, encaisse, depenses and taux_occupation;
- **benefices_par_type** (profit per type): the same detail (attendu, encaisse, depenses_allouees, benefices, taux_recouvrement) separately for long_term (long term) and court_term (short term, holiday rental). As expenses are not linked to a specific type in the data, they are allocated in proportion to the amount collected for each type.

Requête

GET /api/rentals/summary

En-têtes

Authorization: Bearer VOTRE_ACCESS_TOKEN

Portée requise

rentals:read

→

Réponse · 200 OK · JSON

```
{
  "rentals": [ {
    "id": "12",
    "resume": { "2025": {
      "attendu": 4860000.0,
      "encaisse": 405000.0,
      "depenses": 40000.0,
      "taux_occupation": 100.0,
      "taux_recouvrement": 8.33,
      "par_mois": { ... }
    } } ]
}
```

Synthèse financière par location et par année : attendu, encaissé, dépenses, bénéfices
Taux d'occupation et de recouvrement, détail par mois (par_mois)

Request and response of the rentals/summary endpoint.

Holiday rentals now included

Short-term (holiday) contracts are now counted in the totals above, whereas they used to be excluded. A project with holiday rentals will therefore show higher amounts than before this update.

```
curl https://construction.my-smartlife.fr/api/rentals/summary \
-H 'Authorization: Bearer YOUR_ACCESS_TOKEN'
```

Example response:

```
{
  "rentals": [
    {
      "id": "12",
      "name": "Résidence Les Palmiers",
      "currency": "XOF",
      "resume": {
        "2025": {
          "attendu": 4860000.0,
          "depenses": 40000.0,
          "encaisse": 405000.0,
          "benefices": 365000.0,
          "taux_occupation": 100.0,
          "taux_recouvrement": 8.33,
          "par_mois": {
            "2025-01": { "attendu": 405000.0, "encaisse": 405000.0, "depenses": 40000.0, "taux_occupation": 100.0 }
          },
          "benefices_par_type": {
            "long_terme": { "attendu": 1260000.0, "encaisse": 105000.0, "depenses_allouees": 10370.37, "benefices": 94629.63, "taux_recouvrement": 8.33 },
            "court_terme": { "attendu": 3600000.0, "encaisse": 300000.0, "depenses_allouees": 29629.63, "benefices": 270370.37, "taux_recouvrement": 8.33 }
          }
        }
      }
    }
  ]
}
```

Test this endpoint

An interactive tester for this endpoint is available in the online version: <https://construction.my-smartlife.fr/documentation/article-7-api-endpoint-rentals-summary.html?lang=1>

STEP 5 OF 6 · WRITE AND TROUBLESHOOT

API — project/{id}/state endpoint (write)

Documentation of the PATCH /api/project/{project_id}/state endpoint: change the status of a project through the OAuth2 API.

Online version: <https://construction.my-smartlife.fr/documentation/article-10-api-endpoint-project-id-state-ecriture.html?lang=1>

PATCH /api/project/{project_id}/state

Required scope: project:write.



Requête

PATCH /api/project/{id}/state

En-têtes

Authorization: Bearer VOTRE_ACCESS_TOKEN

Portée requise

project:write

Corps (JSON)

```
{ "status": "termine" }
en_cours · termine · suspendu · archive
```

Réponse · 200 OK · JSON

```
{
  "project_id": "12",
  "status": {
    "label": "termine",
    "legacy_value": 100
  }
}
```

Seul endpoint d'écriture : modifie réellement le statut du projet (confirmation demandée dans le testeur)
Toute autre valeur de status renvoie 422

Request and response of the project/{id}/state write endpoint.

The only write endpoint of this API

All the other documented endpoints are read-only. This one really changes the status of the targeted project — use it with a token that has been explicitly granted the project:write scope.

Request body (JSON):

```
{
  "status": "en_cours"
}
```

status must be one of these values: en_cours (in progress), termine (completed), suspendu (suspended) or archive (archived). Any other value returns 422.

```
curl -X PATCH https://construction.my-smartlife.fr/api/project/12/state \
-H 'Authorization: Bearer YOUR_ACCESS_TOKEN' \
-H 'Content-Type: application/json' \
-d '{"status": "termine"}'
```

Example response:

```
{  
  "project_id": "12",  
  "status": { "label": "termine", "legacy_value": 100 }  
}
```

Test this endpoint

An interactive tester for this endpoint is available in the online version: <https://construction.my-smartlife.fr/documentation/article-10-api-endpoint-project-id-state-ecriture.html?lang=1>

STEP 6 OF 6 · WRITE AND TROUBLESHOOT

API — Common errors

Table of the error codes returned by the API (401, 403, 404, 422) and how to fix them.

Online version: <https://construction.my-smartlife.fr/documentation/article-8-api-erreurs-courantes.html?lang=1>

Common errors

Code	Cause	Fix
401	Missing or malformed Authorization header ("Jeton OAuth manquant")	Add Authorization: Bearer YOUR_ACCESS_TOKEN to the request.
401	Invalid or expired token ("Jeton OAuth invalide ou expiré")	The token expires after 1 hour: request a new one (with the refresh_token or your client credentials).
403	Missing scope ("Droits insuffisants")	Request a token with the scope required by the endpoint (see the previous pages).
403	Project outside the authorised scope	A token from the "OAuth application" flow only sees its owner's projects; a partner token only sees the projects listed when it was created.
404	The requested project (project/{id}/data or /state) does not exist	Check the project ID.
422	Invalid request body (e.g. status outside the allowed values on PATCH /project/{id}/state)	Check the format and accepted values documented for the endpoint.

401 Non authentifié	403 Accès refusé	404 Introuvable	422 Requête invalide
Cause Jeton absent ou mal formé Jeton invalide ou expiré (1 h)	Cause Portée (scope) manquante Projet hors périmètre du jeton	Cause Le projet demandé n'existe pas	Cause Corps JSON incorrect (ex. status non autorisé)
Solution Ajouter l'en-tête Authorization ou redemander un jeton	Solution Redemander un jeton avec le scope requis	Solution Vérifier l'identifiant du projet	Solution Respecter le format et les valeurs documentés

Chaque erreur renvoie un JSON avec un message explicite, par exemple « Jeton OAuth invalide ou expiré »

The API errors at a glance.



SmartLife Properties

We turn your ideas into reality

Thank you for following the “API developer path”

Find this learning path, its latest version and the whole documentation online.

A question? Our team is here to help.



Scan to open the learning path online

Contact

construction.my-smartlife.fr · contact@my-smartlife.fr · +23771494855

Biteng - Yaoundé - Cameroun

© 2026 SmartLife Properties · Document generated on 09 Oct 2026