
Partoo API documentation

Documentation

Release 0.0.0

Quentin Francois

Aug 27, 2019

Contents

1	Table of contents	3
1.1	Getting started	3
1.2	Rest API	4
1.3	Javascript API	8
1.4	Need help?	19
1.5	Releases history	20

Welcome on Partoo API documentation website! Partoo works hard on **gathering rich and up-to-date information** about **local businesses** in a structured format. We collect rich contents such as restaurants' menus, special offers, photographs and pictures, opening hours and so on. Our database keeps growing every day!

Our goal is to share all these information with you and continue **helping local businesses** to be present all over the internet.

By using our API, **you agree with our terms of use** : <https://www.partoo.fr/cgu-api>.

1.1 Getting started

1.1.1 Overview

Partoo APIs are currently available for partners and clients only. There are 2 of them:

- a REST API maint for querying and updating data in our database using JSON format
- a JS API maint to ease the integration of part of our application in third party solutions

1.1.2 Authentication

For both APIs, authentication is achieved thanks to an API key.

The Rest API documentation can be found here: https://developers.partoo.co/rest_api/v2/

For the JS API, you should provide an API key authenticating a user to the *login* function.

```
var partooPage = Partoo.init('the-id-of-the-container-where-you-want-to-insert-partoo  
↪');  
  
partooPage.login(apiKey);
```

1.2 Rest API

1.2.1 Authorizations

Warning: Warning : This documentation is now outdated. For current doc, please see: https://developers.partoo.co/rest_api/v2/#tag/ApiKeys

1.2.2 Businesses

Warning: Warning : This documentation is now outdated. For current doc, please see: https://developers.partoo.co/rest_api/v2/#tag/Users

1.2.3 Organisations

Warning: Warning : This documentation is now outdated. For current doc, please see: https://developers.partoo.co/rest_api/v2/#tag/Organisations

1.2.4 Users

Warning: Warning : This documentation is now outdated. For current doc, please see: https://developers.partoo.co/rest_api/v2/#tag/Users

1.2.5 Categories

Warning: Warning : This documentation is now outdated. For current doc, please see: https://developers.partoo.co/rest_api/v2/#tag/Categories

1.2.6 Reviews

Warning: Warning : This documentation is now outdated. For current doc, please see: https://developers.partoo.co/rest_api/v2/#tag/Reviews

1.2.7 Roles

Warning : This documentation is now outdated. For current doc, please see: https://developers.partoo.co/rest_api/v2/#section/Resources-access

1.2.8 Errors

Warning: Warning : This documentation is now outdated. For current doc, please see: https://developers.partoo.co/rest_api/v2/

1.2.9 Change Log

You can find below the changes made on the API V2 order by descending date.

2019-06-07 : Minor changes

Update the documentation for the user endpoints:

- add sidebar_products
- update the documentation which was not up to date

Endpoint documented:

- User getting details user_get
- User search user_search
- User creation user_creation
- User update user_update

2019-04-19 : Minor change

Add the documentation for the category details endpoint.

Endpoint documented:

- Category getting details get_category_details

2019-03-11 : Minor change

The documentation was added for integration status endpoint.

Endpoint documented:

- Business integration status get_integration_status

2019-03-07 : Major change

Change #1:

2 endpoints were added to delete org and business.

Endpoints created:

- Create org org_delete
- Create business business_delete

Change #2:

The sandbox environments changed. The 2 new sandbox base urls are:

- <https://sandbox.partoo.co>
- <https://sandbox.localoo.es>

Change #3:

Add documentation for search reviews endpoint review_search

2018-20-11 : Minor change

Added *lang* param in user create and update endpoints:

Endpoint updated:

- Create user user_creation
- Update user user_update

2018-19-11 : Major changes

Change #1: Added an endpoint to add, and access additional data.

Endpoint created:

- Get additional data get_additional_data
- Set additional data set_additional_data

Change #2: Added an endpoint to access partner urls.

Endpoint created:

- Get partner urls get_partner_urls

Change #3: Added a 'client_offer_url' field in promo.

Endpoint affected:

- Promotional offers *Businesses*

2018-09-26 : Minor change

Added two search options to business/search : subscribed_to_rm & subscribed_to_pm.

Endpoint created:

- Business search business_search

2018-09-26 : Minor change

Added an endpoint to re-invite users.

Endpoint created:

- User re-invite user_re_invite

2018-09-04 : Minor change

It is now possible to update user password.

Endpoint affected:

- User update user_update

2018-09-03 : Major changes

Change #1: Added an endpoint to delete user.

Endpoint created:

- User delete user_delete

Change #2: It is now possible to invite a user by mail at creation by setting `send_invitation` parameter to `True`.

Endpoint affected:

- User create user_creation

2018-07-24 : Minor change

Added the `api/v2/user/me` endpoint, where you can get all the information about your user (the one your authenticated as through the API key).

2018-07-24 : Minor change

Added the `api/v2/userstats` endpoint, where you can get the count of : - All users. - Active users. - Invited users.

2018-07-20 : Minor changes

Change #1: Added a `display_business_ids` param in user get and search endpoints. This parameter can take 2 values: `True` or `False`. It is not required and by default it is set to `True`. When `display_business_ids` set to `False`, the field `businesses_ids` is not returned by the endpoint. Setting `display_business_ids` will dramatically decrease the response time of the search user endpoint.

Endpoints affected:

- User get user_get
- User search user_search

Change #2: Added a `query` param in user search endpoint. This parameter is expecting a string. It is not required. It is meant to perform a fuzzy search on user. The fuzzy search is performed on user's `email`, `first_name` and `last_name` params.

Endpoints affected:

- User search user_search

Change #3: Added a `consumable` param in authorize endpoint. This parameter can take 2 values: `True` or `False`. It is not required. By default, it is set to `False`. When `consumable` is set to `True`, the token will be valid for one request only (it will be consumed).

Endpoints affected:

- Authorize *Authorizations*

2018-07-19 : Minor change

Added a “status” field for users. An “invited” status means that a user is not activated, until he sets his password. An “active” status means that a user is fully configured and has logged in at least once in the application.

Endpoints affected:

- User get user_get
- User search user_search

2018-07-11 : Minor change

The accepted statuses now authorized for businesses are **open, closed and definitely_closed**. The **will_open** status is not authorized anymore.

Endpoints affected:

- Business creation business_creation
- Business update business_update

1.3 Javascript API

1.3.1 Getting started with the JS API

In order to embed Partoo inside your own webpage, you can use our javascript API.

You can import our Javascript API by adding this script in your `<head>` :

```
<script src="https://www.partoo.fr/partoo.js" type="text/javascript"></script>
```

It gives you access to our global `Partoo` namespace.

Please note that if you are using our test environment for our development, you must change `https://www.partoo.fr` by `https://test.partoo.fr`.

Instanciating Partoo on your webpage

To add the partoo interface on a part of your website, do the following:

(for token generation please see the section dedicated to the authorization process through the REST API `authorize_authorizing`)

```
var userToken = 'userTokenGeneratedServerSide'; // User Token you generated server_
↳side thanks to the authorize endpoint

var options = { // this is a list of options you can configure to adapt page display/
↳behaviour
  'startPage': 'presenceManagement', // here the user will be redirected to the_
↳presenceManagement view after logging
  'displayIntercom': false, // true by default
  'displayUserParams': false, // true by default
```

(continues on next page)

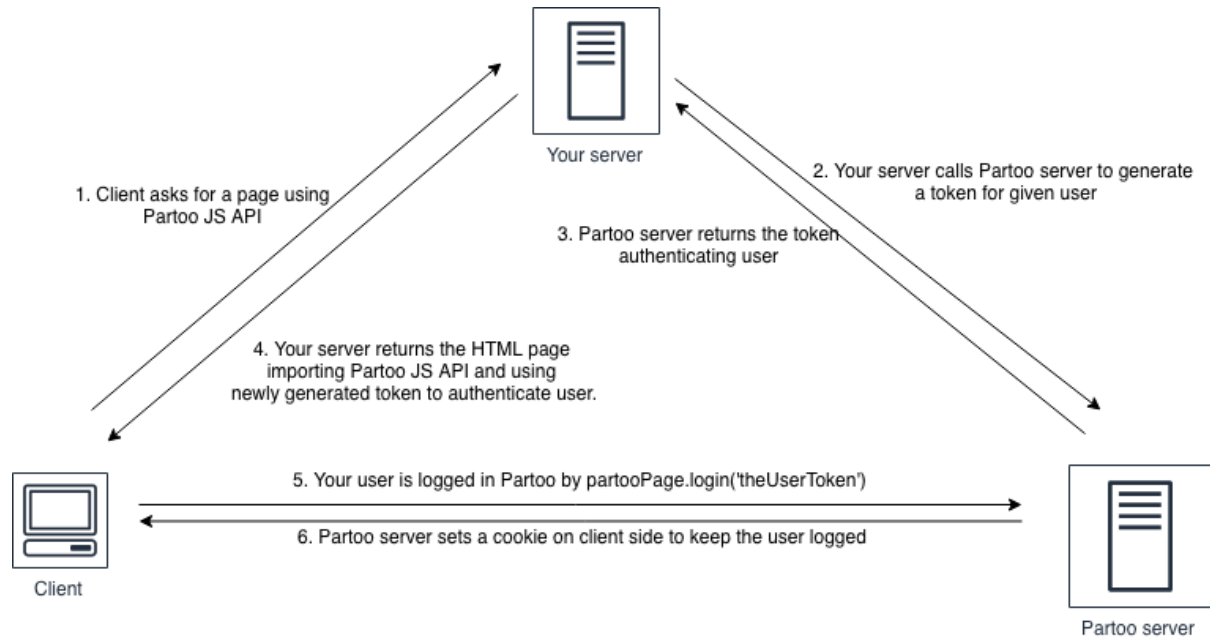
(continued from previous page)

```

    'displayAddButton': false, // true by default
  };
  var partooPage = Partoo.init(
    'the-id-of-the-container-where-you-want-to-insert-partoo',
    options
  ); // instantiates the Partoo page (A loader appears)

  partooPage.login(userToken) // logs the user and redirects him to the `startPage`

```

Schema of a possible integration:**Destroying Partoo page after use**

To remove the partoo interface, just do the following (it will automatically logout the user).

```
partooPage.destroy();
```

1.3.2 Integration of Partoo Pages**Navigation**

To navigate through Partoo App pages, you can use 3 methods:

```

// To navigate to a specific page
// The 2nd and 3rd arguments are optional
partooPage.navigate('reviewManagement', seedData, pageOptions);

// To go back to previous page
partooPage.back();

```

(continues on next page)

(continued from previous page)

```
// To go forward to next page
partooPage.forward();
```

Available Pages

In the table below:

- Route name is the string value you should as the first argument of the *partooPage.navigate* function
- seedData is the seed object you can indicate as the second argument of the *partooPage.navigate* function
- pageOptions is the options object you can indicate as the third argument of the *partooPage.navigate* function

Route name	seedData argument	pageOptions argument	View	Authentication required
lab	LabSeedData null	null	the lab view	No
businesses	null	null	the list of a user's businesses	Yes
business	null	{businessId: string}	the business edit view of a given business	Yes
add	null	null	the add business view	Yes
analytics	null	null	the analytics page	Yes
presenceManagement	null	null	the presence management view	Yes
reviewManagement	null	null	the review management view	Yes
reviewAnalytics	null	null	the review analytics view	Yes
reviewBooster	null	null	the review booster view	Yes
partnerConnections	null	null	the google & facebook partner connexion page	Yes

LabSeedData is a JS object that should look like this:

```
var seedData = {
  name: 'string',
  address: 'string',
  zipcode: 'string',
  city: 'string',
  country: 'string',
  phone: 'string',
};
```

Lab Page

Integrating

The Lab Page is the only page that does not require the user to be authenticated.

You can integrate this page by doing:

```
var partooPage = Partoo('partoo-container', {});

// if you want to pre-fill some inputs in the Lab form
var seedData = {
  name: 'Joe Doe',
  address: '404, rue foobar',
  zipcode: '75017',
  city: 'Paris',
  country: 'France',
  phone: '',
};

// go to lab view
partooPage.navigate('lab', seedData);
```

Available callbacks

The are 4 available callbacks:

- Lab create button callback *Lab create callback*
- Lab sign up button callback *Lab sign up button callback*
- Lab login button callback *Lab login button callback*
- Lab results received callback *Lab results received callback*

Business list Page

Integrating

You can integrate this page by doing:

```
var partooPage = Partoo('partoo-container', {});
var userToken = 'serverSideGeneratedToken';

partooPage.login(userToken); // authentication is mandatory access add page
partooPage.navigate('businesses');
```

Available callbacks

There are 2 available callbacks:

- Subscribe business callback *Subscribe business callback*
- Open business callback *Open business callback*

Business Edit Page

Integrating

You can integrate this page by doing:

```
var partooPage = Partoo('partoo-container', {});
var userToken = 'serverSideGeneratedToken';

partooPage.login(userToken); // authentication is mandatory access add page
partooPage.navigate('business', null, { businessId: 'theIDOfTheBusinesYouWantToEdit' }
↔);
```

Available callbacks

No available callback.

Add Business Page

Integrating

You can integrate this page by doing:

```
var partooPage = Partoo('partoo-container', {});
var userToken = 'serverSideGeneratedToken';

partooPage.login(userToken); // authentication is mandatory access add page
partooPage.navigate('add');
```

Available callbacks

The is 1 available callback:

- Business created callback *Business created callback*

Analytics Page

Integrating

You can integrate this page by doing:

```
var partooPage = Partoo('partoo-container', {});
var userToken = 'serverSideGeneratedToken';

partooPage.login(userToken); // authentication is mandatory access add page
partooPage.navigate('analytics');
```

Available callbacks

No available callback.

Presence Management Page

Integrating

You can integrate this page by doing:


```
var partooPage = Partoo('partoo-container', {});
var userToken = 'serverSideGeneratedToken';

partooPage.login(userToken); // authentication is mandatory access add page
partooPage.navigate('presenceManagement');
```

Available callbacks

The are 2 available callbacks:

- Go to edit click callback *Presence Managment go to edit click callback*
- Go to partner connection click callback *Presence Management go to partner connection click callback*

Review Management Page

Integrating

You can integrate this page by doing:

```
var partooPage = Partoo('partoo-container', {});
var userToken = 'serverSideGeneratedToken';

partooPage.login(userToken); // authentication is mandatory access add page
partooPage.navigate('reviewManagement');
```

Available callbacks

No available callback.

Review Analytics Page

Integrating

You can integrate this page by doing:

```
var partooPage = Partoo('partoo-container', {});
var userToken = 'serverSideGeneratedToken';

partooPage.login(userToken); // authentication is mandatory access add page
partooPage.navigate('reviewAnalytics');
```

Available callbacks

No available callback.

Review Booster Page

Integrating

You can integrate this page by doing:

```
var partooPage = Partoo('partoo-container', {});
var userToken = 'serverSideGeneratedToken';

partooPage.login(userToken); // authentication is mandatory access add page
partooPage.navigate('reviewBooster');
```

Available callbacks

No available callback.

Partner Connections Page

Integrating

You can integrate this page by doing:

```
var partooPage = Partoo('partoo-container', {});
var userToken = 'serverSideGeneratedToken';

partooPage.login(userToken); // authentication is mandatory access add page
partooPage.navigate('partnerConnections');
```

Available callbacks

No available callback.

1.3.3 Available Options

The options can be provided to both the Partoo global object (being default options) or to a page instance, result of Partoo.init() using the following syntax

```
var partooPage = Partoo.init(elemId, userToken, options);
partooPage.setOptions(options);
```

options is a dict with the following keys

Already implemented:

Parameter	Type	Default	Description
startPage	string	businesses	The navigation start page when doing login()
startPageAnchorParam	string	null	Add an anchor param in the login url
displayIntercom	boolean	true	Display the intercom help button in the lower right corner
displayUserParams	boolean	true	Display the user's parameter dialog
displayAddButton	boolean	true	Display the add business button on the top banner
displayPartnerConnexionBanner	boolean	true	Display the partner connexion (Google My Business) banner on edit
displayBusinessListSearch	boolean	true	Display the search dialog on the business list
displayMultibusinessSelection	boolean	true	Allow selection of multiple business on the business list
displayPresenceManagement-Download	boolean	true	Display the download data button on the presence management view
displayLabButtons	boolean	true	Display the Create Account button and Connection button in lab view
displayBusinessAdmin	boolean	true	Display the menu item to close a business in edit view
displayLabConnectionForm	boolean	true	Display the connection form when the business is found after lab search
displayEditBusinessSelector	boolean	true	Display the business selector on the business edit view
displayLabFoundView	boolean	true	Display the result of the lab when the business is already in db
businessFoundBannerText	string	''	Change the text of the lab business found banner

1.3.4 Callbacks

- All callbacks prevent the default action of taking place
- To setup your callbacks on user actions you can use:

```
// implemented
partooPage.on('subscribe', function() {});
partooPage.on('open_business', function() {});
partooPage.on('lab_create', function() {});
partooPage.on('business_created', function() {});
partooPage.on('business_additional_info_updated', function() {});
partooPage.on('business_address_updated', function() {});
partooPage.on('business_contact_updated', function() {});
partooPage.on('business_description_updated', function() {});
partooPage.on('business_open_hours_updated', function() {});
partooPage.on('lab_results_received', function() {});
partooPage.on('no_eligible_business_click', function() {});
partooPage.on('pm_view_go_to_edit_click', function() {});
partooPage.on('pm_view_go_to_partner_connection_click', function() {});
partooPage.on('lab_results_received', function() {});
partooPage.on('no_business_click', function() {});
partooPage.on('lab_login_button', function() {});
partooPage.on('lab_sign_up_button', function() {});
```

Subscribe business callback

Happens when a user clicks on subscribe

```
partooPage.on('subscribe', function(subscribeData) {
    // use business_id
})
```

subscribeData has the following structure, with presence_management being either presence_management or review_management

```
{
  "productName": "presence_management",
  "businesses": [
    {
      "id": "5a2ab4edb12ff67cba1b7e1b",
      "org_id": 709,
      "name": "SCEP du Caire",
    }
  ]
}
```

Open business callback

Happens when a user click a link to open a business page.

```
partooPage.on('open_business', function(business_id) {
    // use business_id
});
```

business_id is the identifier of the business the user wants to open

Business created callback

Happens when then user has created a business, has clicked on save and the business has been saved in the Partoo DB
Warning : this callback does not prevent redirection to the business edit page

```
partooPage.on('business_created', function(business_id) {  
    // use business_id  
})
```

Business additional info updated callback

Happens when then user has successfully updated the additionnal info of his/her business (in the *additionnal info tab*, he/she clicked on save). **Warning** : this is a none blocking callback

```
partooPage.on('business_additional_info_updated', function(data) {  
    // use data  
})
```

Business address updated callback

Happens when then user has successfully updated the address of his/her business (in the *address tab*, he/she clicked on save). **Warning** : this is a none blocking callback

```
partooPage.on('business_address_updated', function(data) {  
    // use data  
})
```

Business contact updated callback

Happens when then user has successfully updated contact of his/her business (in the *contact tab*, he/she clicked on save). **Warning** : this is a none blocking callback

```
partooPage.on('business_contact_updated', function(data) {  
    // use data  
})
```

Business description updated callback

Happens when then user has successfully updated the description of his/her business (in the *description tab*, he/she clicked on save). **Warning** : this is a none blocking callback

```
partooPage.on('business_description_updated', function(data) {  
    // use data  
})
```

Business open hours updated callback

Happens when then user has successfully updated the open hours of his/her business (in the *open hours tab*, he/she clicked on save). **Warning** : this is a none blocking callback

```
partooPage.on('business_open_hours_updated', function(data) {
    // use data
})
```

Lab create callback

Happens when the user click on create business in the lab view

```
partooPage.on('lab_create', function(labDetails) {
    // use labDetails
});
```

labDetails structure is the following :

```
{
    "civlity": "Mr",
    "email": "deLaBatte@gmail.com",
    "first_name": "Hubert",
    "last_name": "de la Batte",
    "password": "zzzzzzzz",
    "phone": "0102030405"
}
```

Lab results received callback

Happens when the lab has returned results

```
partooPage.on('lab_results_received', function(labResults) {
    // use labResults
})
```

labResults structure is the following :

```
{
    "annuaire": {
        "stats": {
            "detailed": {},
            "summary": {
                "accurate": 0,
                "not_found": 1,
                "error": 0
            }
        },
        "business": {}
    },
    "le118000": {
        "stats": {
            "detailed": {},
            "summary": {
                "accurate": 0,
                "not_found": 1,
                "error": 0
            }
        },
    },
}
```

(continues on next page)

(continued from previous page)

```
        "business": {}
    },
}
```

With a key per integration displayed in the view (google_my_business, annuaire, le118000)

Lab sign up button callback

Happens when a user click on the sign up button in the Lab View.

```
partooPage.on('lab_sign_up_button', function() {
    // ...
})
```

Lab login button callback

Happens when a user click on the login button in the Lab View.

```
partooPage.on('lab_login_button', function() {
    // ...
})
```

Error callback

Happens when an error 400, 403, 404, 500 happens during page rendering

```
partooPage.on('error', function(errorCode) {
    // use errorCode
})
```

errorCode is an int (400 (client error), 403 (not authorized), 404 (not found), 500 (server error))

No business button callback

Happens when a use click on a *No Business* button.

```
partooPage.on('no_business_click', function() { /* the code you want to be executed_
↳on click */});
```

No business eligible button callback

Happens when a use click on a *No Business* button.

```
partooPage.on('no_eligible_business_click', function() { /* the code you want to be_
↳executed on click */});
```

Presence Management go to edit click callback

Happens when a user clicks on a *Edit* link in the presence management view that should redirect him to business edit view.

```
partooPage.on('pm_view_go_to_edit_click', function() { /* the code you want to be_
↳executed on click */});
```

Presence Management go to partner connection click callback

Happens when a user clicks on a *Edit* link in the presence management view that should redirect him to connection view.

```
partooPage.on('pm_view_go_to_edit_click', function() { /* the code you want to be_
↳executed on click */});
```

1.3.5 Change Log

You can find below the changes made on the JS API order by descending date.

2019-03-20: Minor changes

Added 4 callbacks:

- **business_additional_info_updated** *Business additional info updated callback*
 - business_address_updated *Business address updated callback*
 - business_contact_updated *Business contact updated callback*
 - business_description_updated *Business description updated callback*
 - business_open_hours_updated *Business open hours updated callback*

2018-08-23 : Minor changes

Added 2 callbacks:

- pm_view_go_to_edit_click
- pm_view_go_to_partner_connection_click

2018-08-10 : Minor changes

Fixed bug that was preventing *startPage* param to work when set to *business*.

Added *startPageAnchorParam* to add an anchor param in the login URL.

1.4 Need help?

Contact our support team on intercom just in the low right corner of this window.

1.5 Releases history

1.5.1 v1.0.0

- Added post & patch endpoint
- Added doc on organisations & users

1.5.2 v0.0.0

First version of the Partoo API