Pl@ntNet-API
System & User Guide

Pl@ntNet-API

Training & capacity building resources

Cos4Cloud Services

About the Pl@ntNet-API system and user guide

This is a system and user guide for the Pl@ntNet-API, an AI-based plant identification service that uses Pl@ntNet's visual identification engine that can be used to improve the user experience in citizen observatories.

Description

This is a system and user guide for the **Pl@ntNet-API**, an AI-based plant identification service that uses Pl@ntNet's visual identification engine that can be used to improve the user experience in citizen observatories (COs).

The **Pl@ntNet-API** has been developed by project partner Inria as part of Cos4Cloud a European Horizon 2020 funded project. The Cos4Cloud project has developed thirteen services boosting citizen science technological services to help increase and improve the quantity and quality of observations.

The **Pl@ntNet-API** is a plant identification service, based on Pl@ntNet's extensive database, that uses Artificial Intelligence (AI) to help people identify and better understand plants. Pl@ntNet is also one of the citizen observatories, focused on biodiversity involved in Cos4Cloud, see more about the Cos4Cloud Citzen Observatories here.

Cos4Cloud project partner Inria leads Pl@ntNet and this API is one of three different web services developed for existing and new citizen observatories and other citizen science initiatives wanting to integrate AI-based Species Photo ID and AI-based Biodiversity Prediction as a tool and resource for plant biodiversity into their applications.

Service Coordinator

© Inria Pl@ntNet

Cos4Cloud Coordinator

© CSIC Consejo Superior de Investigaciones Científicas Institut de Ciències del Mar Cos4Cloud

The following content has been provided to help guide users of this service.

WHAT IS PL@NTNET-API?

The **Pl@ntNet Application Programming Interface** (**Pl@ntNet-API**) is a plant identification service, based on the Pl@ntNet citizen observatory extensive database, that uses Artificial Intelligence (AI) to help people identify and better understand plants. It has been developed to be integrated into citizen observatories, and other initiatives, to enhance the user experience when using these platforms, apps etc.

WHO IS THIS SERVICE FOR?

The **Pl@ntNet-API** is available to developers, researchers and others, with some expertise in web technologies, interested in integrating automatic plant identification features into citizen science, apps, platforms or citizen observatories (COs).

For example, a citizen observatory app focusing on plant biodiversity is under development and would like to integrate an automatic plant identification feature without managing it itself - the **Pl@ntNet-API** is a solution. Or, an existing citizen science platform or citizen observatory, operating for a few years would like to trial using AI to enhance the capabilities of its plant identification for its online community - this API can provide this service.

WHAT ARE THE BENEFITS OF THIS SERVICE?

By integrating the **Pl@ntNet-API**:

- A citizen observatory team developing a new app or an existing platform with some focus on plant biodiversity can:
  - Integrate automatic plant identification features without the additional challenge of developing and managing this themselves.
  - Have access to a service which gets the most similar photo: the **Pl@ntnet-API** includes a tool to find the most similar photo (species) from the Pl@ntNet database, compared to an uploaded observation. This provides quality control of identifications.

  - Build in a user-friendly personalised dashboard tool: if used this can enable users to manage their account and see statistics about their use of the API (e.g. graph of the number of queries per day).

- If integrated, citizen observatory users can:

  - Identify plant species easily: when a user uploads a plant photo, the service returns the list of species and the likelihood of the image being species ‘x’ or ‘y’.

  - Post multiple photos in the same observation: users can send several pictures of different parts of the same plant i.e. the flower, the fruit or the leaf. Through this the visual recognition system will learn to recognise them all as part of the same plant.

- Developers can benefit from integrating a service which is:

  - Customisable to the flora they want users to monitor: for example, if the focus is on monitoring flora in the Mediterranean mountains a filter can be applied which allows the visual identification engine to facilitate identification of plants living in this area only.

  - Highly scalable: to the way your system wants to use it. This way, developers of downstream applications do not have to manage the workload. See the details of a record, make comments.

HOW DOES IT WORK?

Developed as a RESTful JSON based API, it uses Pl@ntNet’s visual identification engine as a web service. Pl@ntNet is a CO dedicated to plant biodiversity. It is available for anyone to use as a mobile and web application and identifies different species of plants from observations submitted with pictures. These are automatically compared to the thousands of images in the Pl@ntNet database, and a list of options is proposed.

The service works by querying Pl@ntNet’s database of plant species. The **Pl@ntNet-API** can detect if the image is a plant; this means that any non-plant pictures are automatically tagged REJECT. It also monitors quality based on a range of criteria i.e. status, response time, scalability, etc. It also includes an in-built alert system that allows developers to react quickly.

The system is regularly updated with new flora images posted by the Pl@ntNet user community as well as from other expert databases which are integrated. All of this helps to improve the options for visual recognition.

Identification performance can vary this can depend on the number of images available of the species in the database. Pl@ntNet's system is based on image recognition and the quality of images uploaded can also impact on the quality of the identification provided so this needs to be as informative and clear as possible. For assistance and information about this see Pl@ntNet's how and why tips about this.

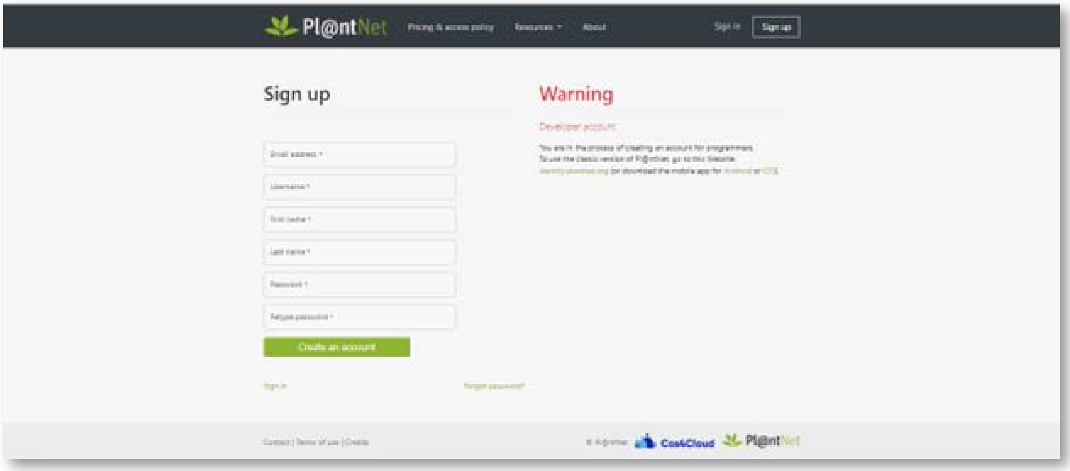
HOW TO USE PL@NTNET-API – A STEP BY STEP GUIDE

The **Pl@ntNet-API** is a service that provides access to the plant identification engine used in Pl@ntNet as a RESTful Web service. The service allows submission of 1 to 5 images of the same plant, simultaneously, and the search returns a list of the most likely species with a confidence score for each. The identification engine is based on most advanced deep learning technologies and is regularly updated thanks to the contributions of the Pl@ntNet user community and users from other COs which have integrated this i.e. iSpot (www.iSpotnature.org), which is also a Cos4Cloud CO.

This system and user guide is for leaders and developers of existing or new COs interested in integrating this service. Citizen scientists and others interested in using this AI recognition service to identify plant species can do so directly on Pl@ntNet itself or by using one of COs that has integrated the Pl@ntNet API as a service. To see how iSpot and other COs have used the **Pl@ntNet-API** see the *Case Studies* and *Best Practice Guidelines* sections of the Toolbox and Evidence Hub.

01 CREATE AN ACCOUNT:

Create a *myPl@ntNet* account: Go to: https://my.plantnet.org, and follow the instructions to register. Please note this requires setting up a developer account. Remember to check email and validate the account for access.



Sign up

Email address *

Username *

First name *

Last name *

Password *

Retype password *

Create an account

Sign in

Forgot password?

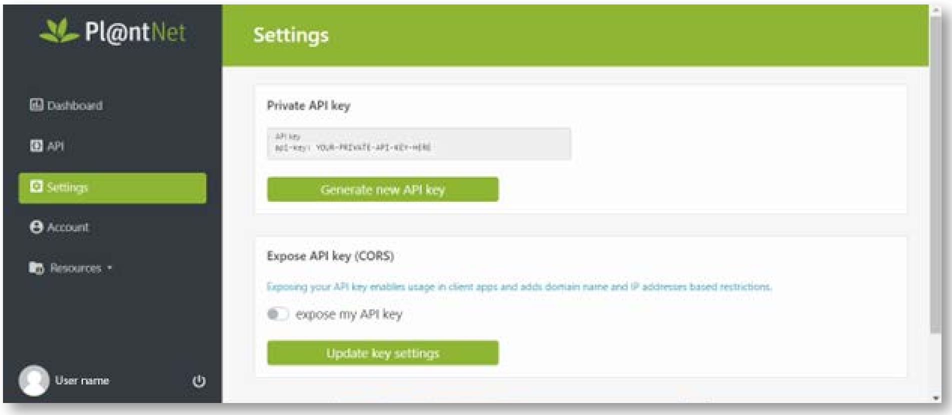
Warning

Developer account

You are in the process of creating an account for programmers.
To use the classic version of Pl@ntNet, go to this website:
identify.plantnet.org (or download the mobile app for Android or iOS).

02 LOGIN AND GET AN API KEY:

Log in and from the account Settings, generate a ‘private API key‘. Please note: the service is only accessible with an API key. See example view below:



Settings

Private API key

API key
api-key: YOUR-PRIVATE-API-KEY-HERE

Generate new API key

Expose API key (CORS)

Exposing your API key enables usage in client apps and adds domain name and IP addresses based restrictions.

○ expose my API key

Update key settings

03 SET UP HTTP REQUESTS TO ADD PLANT IMAGES:

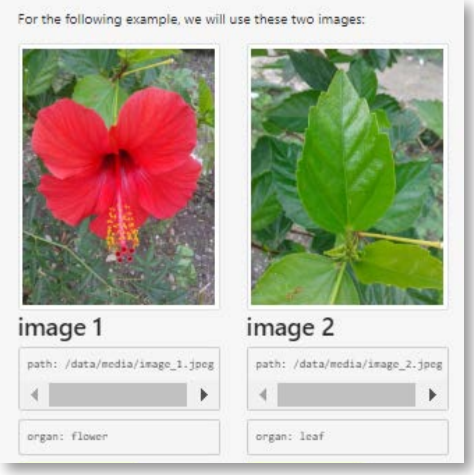
Requests can be made using either one of the following options:

- GET remote images - using the images URLs: This method requests images that are hosted online (i.e. in a citizen observatory) where URLs are available.
- POST local images – path based on local files: This method requests images that are accessible locally i.e. on a computer.

It is important to have the URLs or images of plants ready for integration to facilitate the request. Each identification request must represent a single plant and up to 5 images can be added in a single identification request.

For advice on image size, format, compression etc. go to the FAQs. For example, in the POST local images request example below, two images, a flower and a leaf, are used for one observation and the system recognises them as the same plant:

- Get plant pictures: Add the paths of the images as demonstrated below:



For the following example, we will use these two images:

image 1

path: /data/media/image_1.jpeg

organ: flower

image 2

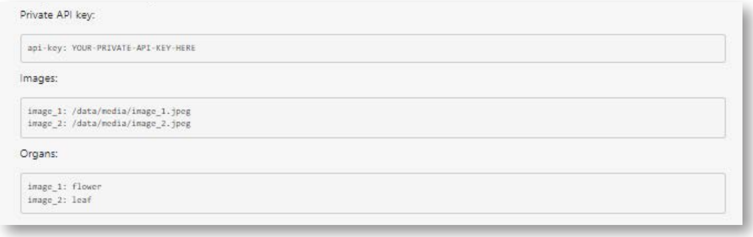
path: /data/media/image_2.jpeg

organ: leaf

- Image 1 (*https://my.plantnet.org/images/image_1.jpeg*) and
- Image 2 (*https://my.plantnet.org/images/image_2.jpeg*).

04 FULL DATA SUMMARY:

It is important view and check the data generated including the assigned private API key, the paths of the images of encoded URLs and the Organs i.e. images demonstrating different parts of the plant. See POST example below:



Private API key:

```
api-key: YOUR-PRIVATE-API-KEY-HERE
```

Images:

```
image_1: /data/media/image_1.jpeg
image_2: /data/media/image_2.jpeg
```

Organs:

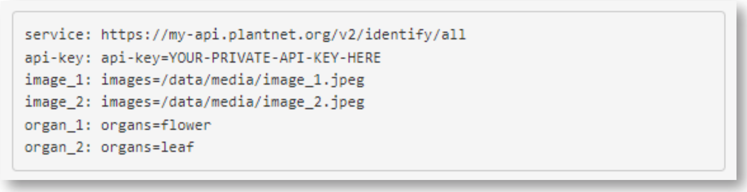
```
image_1: flower
image_2: leaf
```

05 RUN IDENTIFICATION REQUESTS:

The service will run identification requests based on the selected HTTP parameters i.e.urls or images:

- HTTP GET - remote images: URLs of the images to be identified
- HTTP POST - local images: Paths of the images to be identified

See POST Identification request parameters example below:



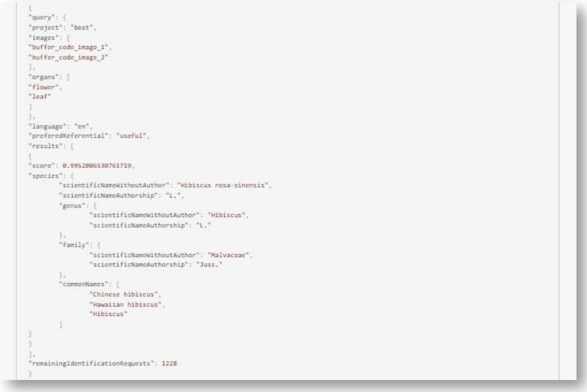
```
service: https://my-api.plantnet.org/v2/identify/all
api-key: api-key=YOUR-PRIVATE-API-KEY-HERE
image_1: images=/data/media/image_1.jpeg
image_2: images=/data/media/image_2.jpeg
organ_1: organs=flower
organ_2: organs=leaf
```

Other POST demonstration examples using JavaScript, Java, Python, PHP and R are available in the **Pl@ntNet-API** documentation here.

06 GENERATE IDENTIFICATION RESULTS:

The **Pl@ntNet-API** includes a tool that finds the most similar photo (species) from the Pl@ntNet database and compares this to an uploaded observation. Once actioned the service returns a list of species and the likelihood of the image being species ‘x’ or ‘y’. A list of probable species names is generated. This is sorted by score with the most likely ID at the top.

See POST identification results example below:



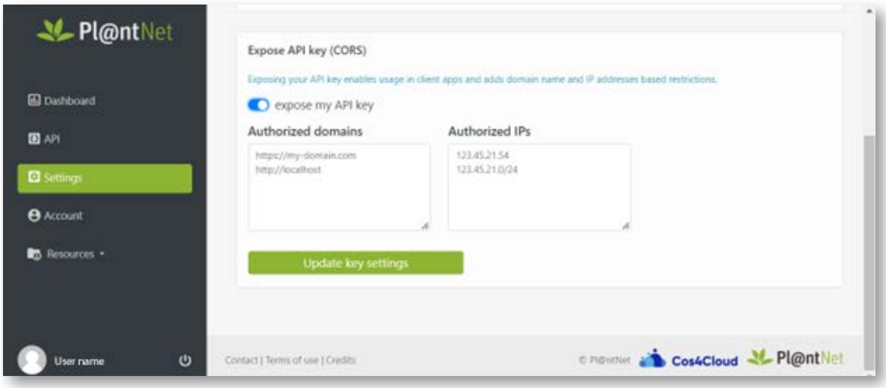
```
{
"query": {
"project": "best",
"images": [
"buffer_code_image_1",
"buffer_code_image_2"
],
"organs": [
"flower",
"leaf"
]
},
"language": "en",
"preferedReferential": "useful",
"results": [
{
"score": 0.9952006530761719,
"species": {
        "scientificNameWithoutAuthor": "Hibiscus rosa-sinensis",
        "scientificNameAuthorship": "L.",
        "genus": {
                "scientificNameWithoutAuthor": "Hibiscus",
                "scientificNameAuthorship": "L."
        },
        "family": {
                "scientificNameWithoutAuthor": "Malvaceae",
                "scientificNameAuthorship": "Juss."
        },
        "commonNames": [
                "Chinese hibiscus",
                "Hawaiian hibiscus",
                "Hibiscus"
        ]
}
}
],
"remainingIdentificationRequests": 1228
}
```

07 API KEY AUTHORISATION:

Exposing the API key enables usage in client apps and adds domain name and IP addresses-based controls This supports integration into COs and visibility of the service within these platforms.

i. Enable client access to the API key (CORS): To enable usage of the API key in browser-side client apps (AJAX), the domain names that the apps are hosted on need to be authorised. Go to the account Settings page and check "expose my API key"

The API key can now be used in a client-side browser application, ie. in a CORS-enabled environment.



Expose API key (CORS)

Exposing your API key enables usage in client apps and adds domain name and IP addresses based restrictions.

● expose my API key

Authorized domains

https://my-domain.com
http://localhost

Authorized IPs

123.45.21.54
123.45.21.0/24

Update key settings

ii. Authorise domain: In the Authorize domains box , add one domain per line. Note: CORS requests will be rejected if the domain present in Origin header is not in this list. (CORS considers *http://domain.com* and *https://domain.com* as different origins).

iii. Authorise IP addresses: once "expose my API key" is checked, any non-CORS request will be rejected if not coming from a server with an IP address matching the "Authorized IPs" list. This prevents the exposed API key from being stolen and used server-side without permission.

However, if access to the API key is required both client-side and server-side, server IP addresses need to be explicitly authorised. Under "*Authorized IPs*", add one IPv4 or IPv4 range (CIDR format) per line.

INFORMATION AND FURTHER SUPPORT

Introduction to this Cos4Cloud service: https://cos4cloud-eosc.eu/services/plantnet-as-a-service/

Register to use this service here: https://my.plantnet.org/signup

Getting started support: https://my.plantnet.org/doc

API documentation can be found here: https://my.plantnet.org/doc/openapi

Full documentation of the Pl@ntNet API service for developers: https://my-api.plantnet.org/?tags=my-api#/cos4cloud

API FAQs: https://my.plantnet.org/doc/faq

Terms of use: https://my.plantnet.org/terms_of_use

Contact: api@plantnet-project.org

About Pl@ntNet: https://plantnet.org/en/

Access this service from the EOSC Marketplace: https://marketplace.eosc-portal.eu/services/pl-ntnet-identification-service

Get the Cos4Cloud infographic for this service: Pl@ntNet-API infographic

Additional resources: About the Cos4Cloud Toolbox and Evidence Hub

This system and user guide is a training resource for the Pl@ntNet-API, one of the technological services co-designed by the Cos4Cloud project (https://cos4cloud-eosc.eu/). **Title: Pl@ntNet-API System and User Guide**. Service Coordinator: Inria. This guide is one of the Training and Capacity Building resources in the Cos4Cloud Toolbox & Evidence Hub developed by the Open University in collaboration with project partners. Contact: cos4cloud-toolbox@open.ac.uk.

This project is part of: EUROPEAN OPEN SCIENCE CLOUD

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no 863463.