
MetaGenSense Documentation

Release 1

Olivia Doppelt-Azeroual, Damien Correia

November 22, 2016

1	MetaGenSense	1
1.1	Citation	1
1.2	Requirements	1
1.3	Installation	1
2	Connection	3
2.1	Connection	3
2.2	Configuration	3
3	Project documentation	7
3.1	Project creation	7
4	Launch Workflows	9
4.1	Launch workflow	9
4.2	Create Workflow	10
5	Import documentation	11
5.1	Input data	11
5.2	Load data	12
6	Export documentation	15
6.1	Download / Share / Export	15

MetaGenSense

A web application framework for analysis and visualization of High throughput Sequencing metagenomic data.

1.1 Citation

Correia D, Doppelt-Azeroual O, Denis JB et al. MetaGenSense : A web application for analysis and visualization of high throughput sequencing metagenomic data [v1; ref status: approved with reservations 2, not approved 1, <http://f1000research.com/articles/4-86/v1>] F1000Research 2015.

1.2 Requirements

```
python==2.7
Django==1.6.2
bioblend==0.5.3
```

1.3 Installation

1.3.1 1. virtualenv

You may use virtualenv to isolate your django project workspace [virtualenv](<http://www.virtualenv.org/>), [virtualenvwrapper](<http://www.doughellmann.com/projects/virtualenvwrapper/>)

1.3.2 2. Download

You need the *MetaGenSense* project files in your workspace:

- `cd /path/to/your/workspace`
- `git clone git://github.com/pgp-pasteur-fr/MetaGenSense.git`

1.3.3 3. Requirements installation

```
pip install -r requirements.txt
```

1.3.4 4. Set the settings:

a. wsgi.py

MetaGenSense/wsgi.py file is necessary for WSGI gateways (such as uWSGI) to run your Django application and is also required from Django itself. You definitely want to change `{{ project_name }}` value in this file to the name of your application (e.g. *metagensense.settings*).

b. SECRET_KEY setting

Create your own secret key. Open your *MetaGenSense/settings.py*, find *SECRET_KEY* line, paste your secret key. [SECRET_KEY](cf. <https://docs.djangoproject.com/en/1.8/ref/settings/#secret-key>)

c. Paths setting

Set up this paths for you personal configuration/architecture .

Paths to access to Galaxy personal import directory form server:

```
- GALAXY_SERVER_URL = http://.... #url to your Galaxy server
- GALAXY_IMPORT_DIR = 'links'      #path to import data into Galaxy with import tool
- MGS_GALAXY_FOLDER = '/MGS' #absolute path to Galaxy library (directory inside links)
```

Path where to store downloaded files:

```
MGS_DOWNLOAD_FILE_DIR = os.path.join(os.path.sep, 'opt', 'metagensense', 'limsfiles')
```

Folder to save analyse results on the server:

```
ANALYSE_FOLDER = 'analyse'
```

Folder use to export big data from Galaxy (>2Gb):

```
GALAXY_EXPORT_DIR = '/opt/MGS/outputs'
```

d. Initialize the database

First you must have an database engine installed on your machine, Set the database engine (PostgreSQL, MySQL, etc..) in your settings files; *MetaGenSense/settings.py* at *DATABASES* [DATABASES](<https://docs.djangoproject.com/en/1.8/ref/settings/#databases>)

Create django models with the command:

```
./manage.py migrate
```

e. Test settings

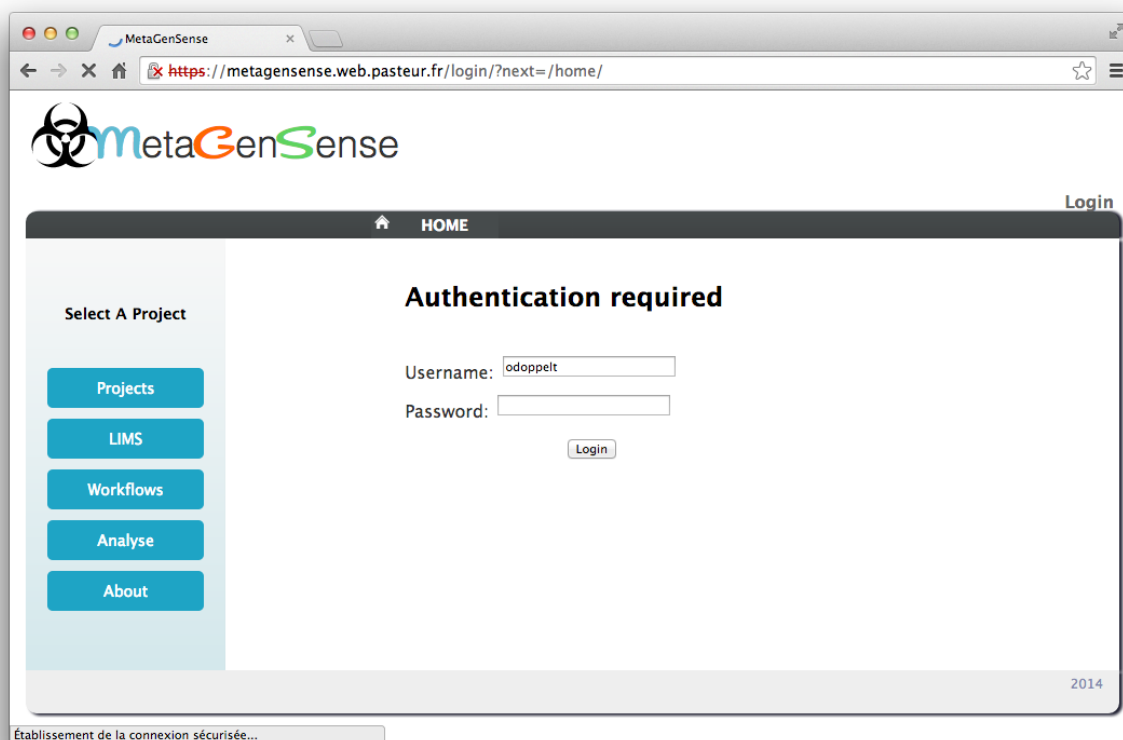
Starts a lightweight development Web server on the local machine. By default, the server runs on port 8000 on the IP address 127.0.0.1. You can pass in an IP address and port number explicitly.

```
./manage.py runserver
```

Connection

2.1 Connection

For access to all features, user connects to the web interface.

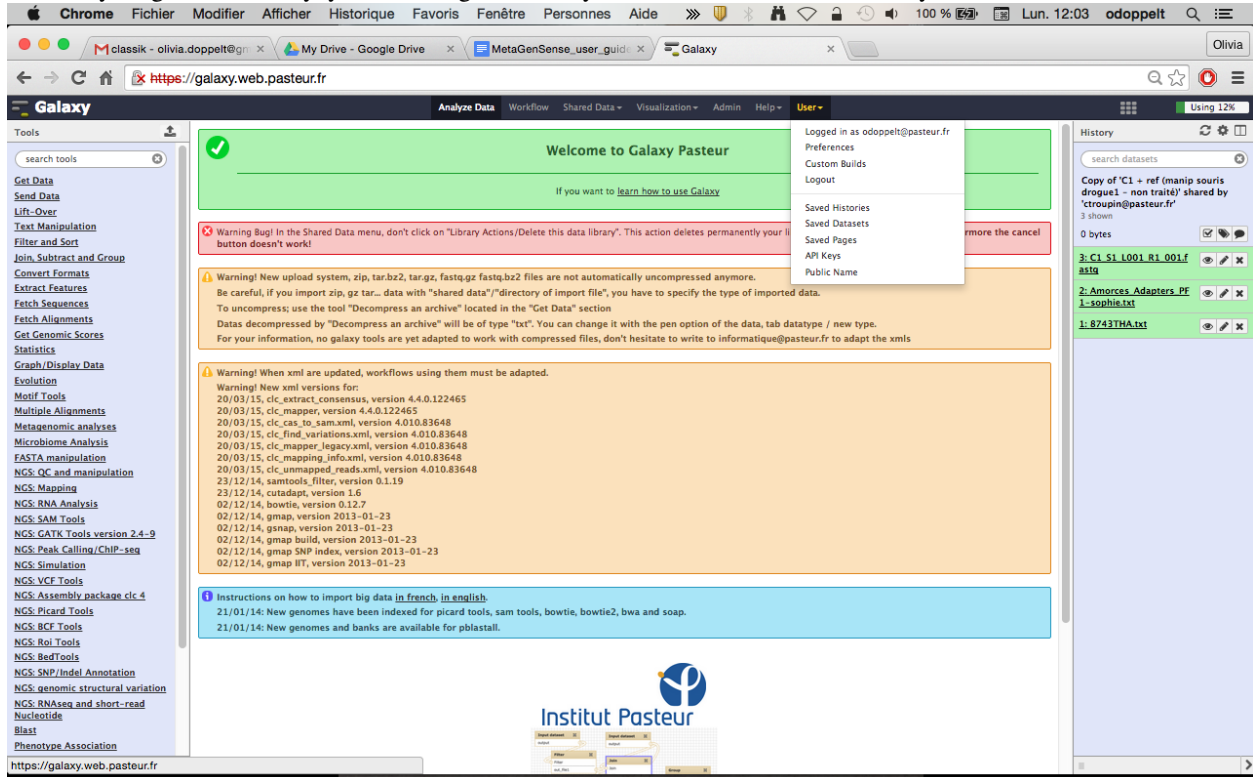


2.2 Configuration

In order to give access to Galaxy from MetaGenSense each user need to:

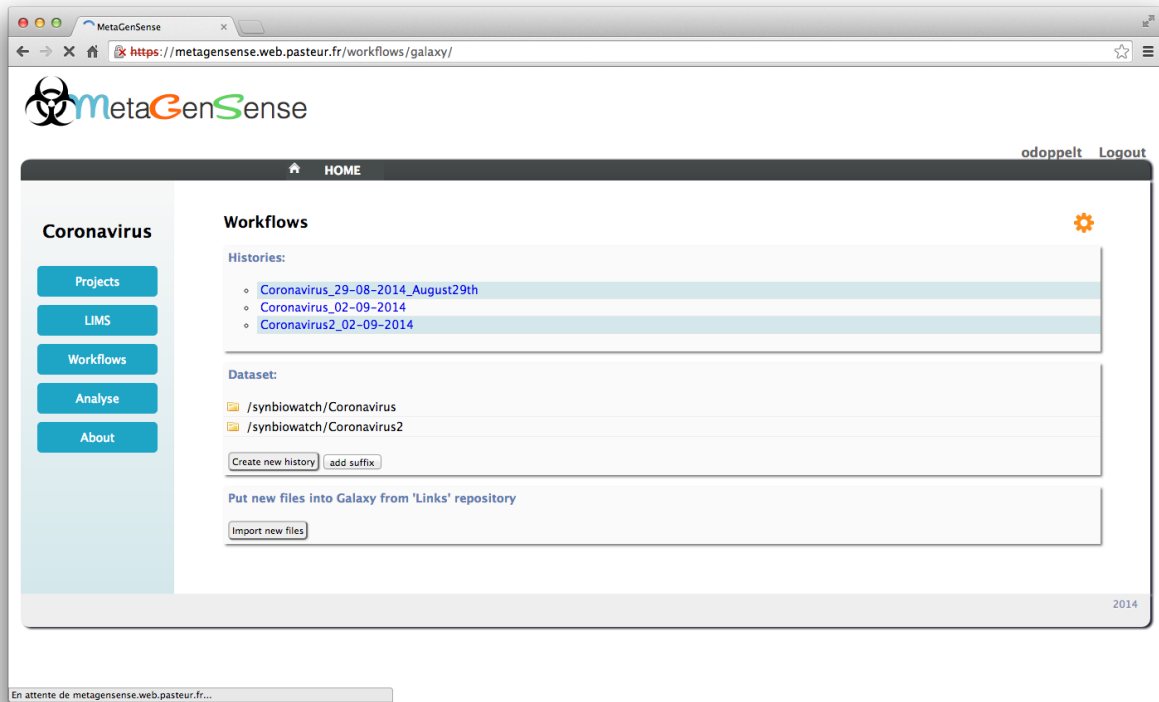
2.2.1 1. Get the API Key

In Galaxy, to get the API key, you need to go to Galaxy, click on User, and on API Keys



2.2.2 2. Set the API Key

In the “Workflows” menu, click on the orange wheel to set or update the galaxy API key.



Project documentation

3.1 Project creation

At his first connection, he needs to create a new project by filling a form which will record data in the LIMS.

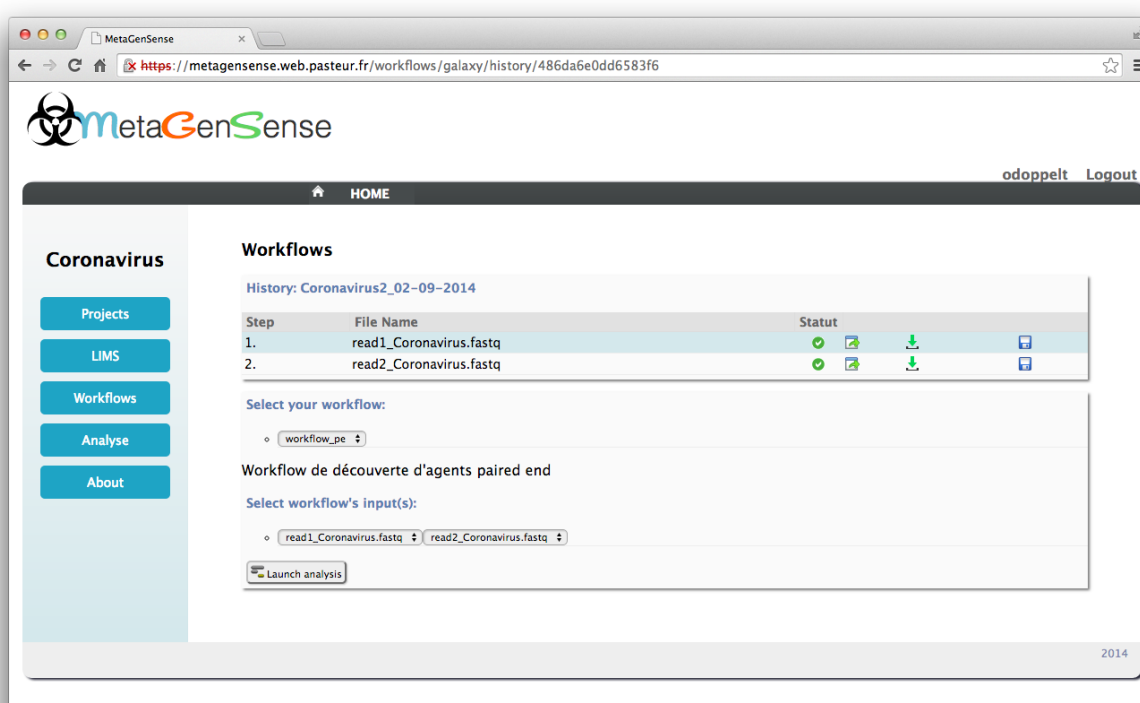
1. Enter a short description
2. Specify who else will have access to the project (don't forget to select yourself)

The screenshot shows a web browser window with the URL <https://metagensense.web.pasteur.fr/projects/add>. The page features the MetaGenSense logo and a navigation bar with a 'HOME' link and a user profile 'odoppelt' with a 'Logout' button. On the left, a sidebar titled 'Select A Project' contains buttons for 'Projects', 'LIMS', 'Workflows', 'Analyse', and 'About'. The main content area is titled 'Project' and contains a form with the following fields: 'Name' (text input), 'Context' (text input), and 'Comments' (text area). Below these is a subscriber selection section. It includes a note: 'Hold down "Control", or "Command" on a Mac, to select more than one.' There are two panels: 'Available Subscribers' with a search filter and a list of users (admin, CORREIA Damien, DIANCOURT Laure, DOPPELT- AZEROUAL Olivia, HERBERT CHARLES, VANDENOEGART Mathias), and 'Chosen Subscribers' (currently empty). A 'Choose all' link is at the bottom of the available list, and a 'Remove all' link is at the bottom of the chosen list. At the bottom of the form, there is a 'Date' input field and a 'Save' button.

Once the project is created, all concerned user will see the project appear when they connect to the interface

Launch Workflows

4.1 Launch workflow



4.1.1 1. Select Workflow

Choose a specific workflow.

Remark: the workflow list is filled by the MetaGenSense (Django) administrators, .

4.1.2 2. Select Input

Select inputs available in the history according to the chosen workflow input format.

4.1.3 3. Launch analyse

After that, click on “launch analysis” to execute the workflow.

4.2 Create Workflow

To set new workflow inside MetaGenSense application you must to have admin right.

1. Create an new workflow in the Galaxy server
2. Retrieve the workflow id
3. Inside MetaGenSense/admin, add a new workflow entry with this id

Import documentation

5.1 Input data

5.1.1 Add input data in the Exchange project directory

At first, MetaGenSense directory (called MGS) needs to be created in the user's Galaxy import directory ("links/'yourLogin'/MGS"). When created, the user must create in it, a folder named as the project to store input raw data that will be analyzed. This way, Galaxy will be able to copy those files within the user's library and then to a history, just before the workflow execution.

This step is the only one, not automated at all. Indeed, the user will need to do those steps manually. To be clear, here are the steps to follow:

1. Go to your Galaxy transfert directory: /links/'yourLogin'
2. Create a directory called MGS, if it doesn't exist.
3. Create in MGS, a sub-directory named like the project (respect even the case).
4. Copy the file(s) to analyse in the sub-directory

5.1.2 Example with user Jake Sully and the project named PanDora:

```
# go in the galaxy transfer directory:
cd galaxy/links/jsully/
#Create the MGS directory:
mkdir MGS
#Create the project directory:
mkdir MGS/PanDora
cd MGS/PanDora
```

Copy a fastq file(s) to the project directory:

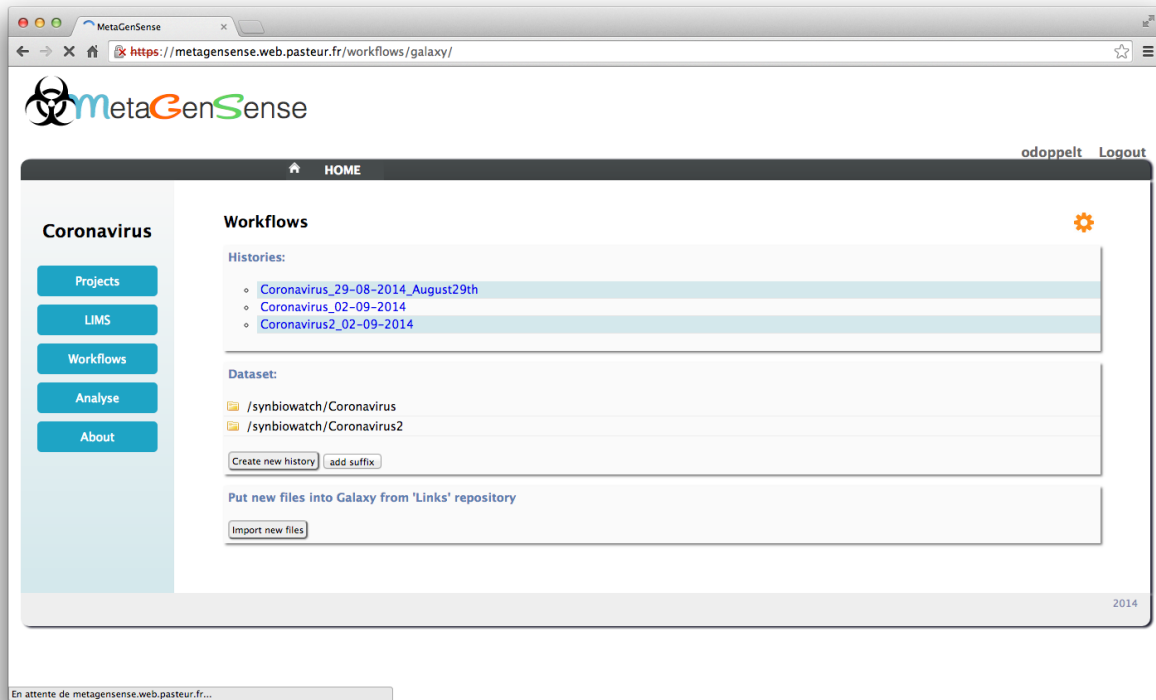
```
cp path/to/myfile.fastq .
```

Remark: This task is the only one which needs to be made outside MetaGenSense.

5.2 Load data

5.2.1 Import data in the galaxy library

New data to analyse must stored in the project directory (cf [Input data](#)) Data are copied within the user's Galaxy library when he clicks on the “import new files” button



Files loaded are now available to create analyze histories

5.2.2 Create new history

- The files to analyze become datasets which will be copied within a galaxy history:
- Select files from Dataset
- Click on the “create new history” button.

By default project name and date are used to create the new history name, user can add a suffix to custom this name. If a new history is created, user is redirected to an other page where he will be invited to select a workflow.

- Another way to create a new history is to upload file with following form, but option is disabled by default because of web-browser limitation

Create new history from Upload files into Galaxy (file < 1Go)

Choose Files

No file chosen

Create

Export documentation

6.1 Download / Share / Export

For each step successfully finished, the user has 3 choices: 1-Download, 2-Share the result files or 3-Export

1. Download : Directly download the result file using the web browser (It uses HTTP protocol, can't exceed 2Gbits)
2. Share: Store the result in MetaGenSense database and share it to other members of the project
3. Export: copy a big file in the user's transfer directory (by default at: `/opt/MGS/outputs/'yourLogin'`)