

---

# WAVES Documentation

*Release 1.1.1*

**[Read the Docs](#)**

Oct 03, 2017



<b>1</b>	<b>ReadME</b>	<b>1</b>
<b>2</b>	<b>Authors</b>	<b>3</b>
<b>3</b>	<b>License (GPLv3)</b>	<b>5</b>
<b>4</b>	<b>Installation</b>	<b>7</b>
<b>5</b>	<b>Contributing</b>	<b>11</b>
<b>6</b>	<b>Version 1.2 - 2017-09-30</b>	<b>13</b>
<b>7</b>	<b>Version 1.1 - 2017-07-07</b>	<b>15</b>
<b>8</b>	<b>Source Documentation</b>	<b>17</b>
<b>9</b>	<b>Indices and tables</b>	<b>19</b>



---

**Note:** What WAVES stands for ?

WAVES stands for “Web Application for Versatile and Easy online bioinformatic Services.”

It is aimed to help you easily present, publish and give access on the web of any bioinformatic tools developed in your teams.

Based on Python and Django framework, it is easy to maintain and make evolve for your specific requirements.

---

## Features

- Create and manage you services execution over platform such as Galaxy, DRMAA clusters (SGE), Direct script execution, API calls to other service, remote calls to other platforms via ssh, etc.
- Easily presents these tools in a nice frontend based on Bootstrap3
- Follow and manage remote REST API access to your service platform
- Manage user’s access to your services

## Installation

- WAVES web app comply to standard reusable project layout for Django. You can use it as a stand alone platform.
- Simply clone or download sources, follow the few configurations steps, and her you are, you can put your tools online !
- See “Details Installation” for more info.

## Contribute

- Git source code: <https://github.com/lirmm/waves-demo>
- Issue tracker: <https://github.com/lirmm/waves-demo/issues>
- Mailing list: [waves-webapp@googlegroups.com](mailto:waves-webapp@googlegroups.com)

## Support

If you are having issues, (or just want to say hello) : we have a mailing list located at: [waves-webapp@googlegroups.com](mailto:waves-webapp@googlegroups.com)

## CHAPTER 2

---

### Authors

---

- Marc Chakiachvili (LIRMM - UMR 5506 CNRS / UM - France),
- Floréal Cabanettes (LIRMM - UMR 5506 CNRS / UM - France),
- Vincent Berry (LIRMM - UMR 5506 CNRS / UM - France)
- Anne-Muriel Arigon Chiffolleau (LIRMM - UMR 5506 CNRS / UM - France),
- Vincent Lefort (LIRMM - UMR 5506 CNRS / UM - France),

### Version

1.1.1 (1.1.1) of 2016/09/25





## CHAPTER 3

---

### License (GPLv3)

---

WAVES are free software: you can redistribute it and/or modify it under the terms of the GNU General Public License version 3 as published by the [Free Software Foundation](#).

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

For more specific details see <http://www.gnu.org/licenses>, the [Quick Guide to GPLv3](#). in the codebase.

The GNU operating system which is under the same license has an informative [FAQ here](#).

### Note to developers

We very much appreciate you using our code to do fun and interesting things with. We hope that while doing this you may find and fix bugs or make enhancements that could be useful for the greater community and will makes the developers aware of them by emailing to [waves-webapp@googlegroups.com](mailto:waves-webapp@googlegroups.com) so they can be considered to be added to the original code base.



GET a WAVES demo website following the next few steps. WAVES can run with Apache or Nginx with uWSGI

**Warning:** To run WAVES, it is strongly recommended to setup a dedicated user. Indeed, WAVES uses `saga-python`. This module will try to create directories (`.radical` and `.saga`) for which another user (such as `www-data`) might not have the rights to do so.

## 0. Prerequisites

---

### Note:

#### In order to install WAVES you will need:

- python 2.7.X (WAVES is not yet compatible with python 3.5)
  - pip package manager
  - A web server: [Apache](#) or [NGINX](#)
  - A database backend (Mysql or Postgres) but by default WAVES runs with sqlite
- 

#### 0.1 Get the sources:

- **Clone WAVES repository:** `git clone https://github.com/lirmm/waves-demo/ [waves_dir]`
- **Download archive:** Download archive at <https://github.com/lirmm/waves-demo/> and uncompress the archive in your destination directory ([waves\_dir])

#### 0.2 Create python virtual environment:

- `$ cd [waves_dir]`
- `[waves_dir]$ virtualenv .venv`

- `[waves_dir]$ source .venv/bin/activate`
- `(.venv)[waves_dir]$ pip install -r requirements.txt`

## 1. Install WAVES

### 1.1 Configuration files:

- **WAVES env configuration file:**

- `(.venv)[waves_dir]$ cd src/waves_demo/settings/`
- `(.venv)[waves_dir]/src/waves_demo/settings$ cp local.sample.env local.env`
- **minimal setup requires these parameters:**
  - \* `SECRET_KEY=your-secret-key-to-keep-secret`
  - \* `REGISTRATION_SALT=generate-your-key`
  - \* `ALLOWED_HOSTS=your-host-name` (Ex: localhost, 127.0.0.1 for testing purpose)
- you can set up as well your db connection params here (or in classical “DJANGO way” settings if you want)

### 1.2 Set up database:

- Check parameters with: `(.venv)[waves_dir]/src/$ ./manage.py check` (pip install any missing dependencies)
- See your configuration with: `(.venv)[waves_dir]/src/$ ./manage.py waves config`
- If no setup, default database is used `~/[waves_dir]/waves.sample.sqlite3`

#### 1.2.1: Setup your database connection string (if not using sqlite default)

- Create local.env (copy from local.env.sample located in waves\_demo/settings/)
- Setup line corresponding to your needs (DATABASE\_URL)
- Create your database: `(.venv)[waves_dir]/src/$ ./manage.py migrate`
- Create Superadmin user: `(.venv)[waves_dir]/src/$ ./manage.py createsuperuser`

#### 12.2.: Load sample data into your database:

- Create your database: `(.venv)[waves_dir]/src/$ ./manage.py loaddata demo/fixtures/samples/sample_data.json`

### 1.3 Test your server:

- `(.venv)[waves_dir]/src/$ ./manage.py runserver [ServerIP:ServerPort] --insecure`

#### 1.2.1 If you use default database configuration:

- You may login into backoffice (127.0.0.1:8000) with credentials : `demo@demo.fr / demodemo`
- A sample ‘cp’ service is already given as a starter service template. Running on local server.

**1.4 Start WAVES daemons:**

- `(.venv) [waves_dir]/src/$ ./manage.py waves queue start`
- `(.venv) [waves_dir]/src/$ ./manage.py waves purge start`

## 2. Configure the web server:

**2.1 Production settings:**

- Create staticfiles: `(.venv) [waves_dir]/src/$ ./manage.py collectstatic`
- Setup your server: [Django Docs](#)

**2.2 UWSGI:**

- Init sample script is available in `waves_uwsgi.ini`
- Rename/Edit according to your settings

**See also:**

<http://uwsgi-docs.readthedocs.io/>

**2.3 APACHE:****See also:**

<http://uwsgi-docs.readthedocs.io/en/latest/Apache.html>

**2.4 NGINX:****See also:**

[http://uwsgi-docs.readthedocs.io/en/latest/tutorials/Django\\_and\\_nginx.html](http://uwsgi-docs.readthedocs.io/en/latest/tutorials/Django_and_nginx.html)



## CHAPTER 5

---

### Contributing

---

You can contribute to WAVES project with following repositories:

- Git source code: <https://github.com/lirmm/waves-demo>
- Issue tracker: <https://github.com/lirmm/waves-demo/issues>
- Mailing list: [waves-webapp@googlegroups.com](mailto:waves-webapp@googlegroups.com)





## CHAPTER 6

---

Version 1.2 - 2017-09-30

---

- Integrated modification issued from waves-core 1.1.2
- Added changelog



## CHAPTER 7

---

Version 1.1 - 2017-07-07

---

First Ready to play version - detached from waves-core



### Waves Demo classes

WAVES Demo extends base Waves-Core functionality

**See also:**

Waves-Core Documentation <http://waves-core.readthedocs.io/en/latest/modules/source.html>

**See also:**

WAVES CORE documentation : <http://waves-core.readthedocs.io/en/latest/>



## CHAPTER 9

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`