
Python and Vue.js integration Documentation

Release 1.0.9

Christian Strappazzon

Feb 23, 2018

Contents

1	Python and Vue.js gule	3
1.1	UPDATE	3
1.2	The aim	3
1.3	Requirements	3
1.4	Commands reference	4
1.5	Installation	4
1.6	Features	5
2	Installation	7
2.1	Stable release	7
2.2	From sources	7
3	Usage	9
3.1	Django quickstart	9
4	Contributing	11
4.1	Types of Contributions	11
4.2	Get Started!	12
4.3	Pull Request Guidelines	13
4.4	Tips	13
5	Credits	15
5.1	Development Lead	15
5.2	Contributors	15
6	History	17
6.1	1.0.9 (2017-10-03)	17
6.2	1.0.7 (2017-10-03)	17
6.3	1.0.6 (2017-05-30)	17
6.4	1.0.5 (2017-05-30)	17
6.5	1.0.4 (2017-05-30)	17
6.6	1.0.3 (2017-05-30)	18
6.7	1.0.2 (2017-05-30)	18
6.8	1.0.1 (2017-05-30)	18
6.9	1.0.0 (2017-05-29)	18
6.10	0.0.7 (2017-05-24)	18
6.11	0.0.4 (2017-05-24)	18

6.12	0.0.3 (2017-05-24)	18
6.13	0.0.2 (2017-05-24)	19
6.14	0.0.1 (2017-05-23)	19
7	Indices and tables	21

Contents:

CHAPTER 1

Python and Vue.js gule

1.1 UPDATE

At the moment project is not supported anymore, due to the rapid evolving of the JavaScript ecosystem. It was born like a POF to allow me to study a bit of JavaScript and VueJS and try some hacks with *click*. Maybe in the future, the project could reborn. Thanks for using it!

1.2 The aim

Gluing Python and [Vue.js](#) with a set of scripts that automate the dev and build process.

Projects aims to be agnostic, just use it in order to automate the boring stuff to setup a Vue.js project.

The point is: you start with SPA app inside your current project and then extract it without having the dependency with backend framework, simply changing the `npm` build scripts.

Feel free to contribute with PRs and opening issues.

Thanks! Cheers!

1.3 Requirements

- Python 2.7+ or 3.4+
- nodejs > 5 and npm > 3 (could not work with new versions)
- *nix like SO, Windows not supported

1.4 Commands reference

python-vuejs is shipped with a nice cli built on top of click.:

```
(env) $ pyvue --help
Usage: pyvue [OPTIONS] COMMAND [ARGS]...

Options:
  --help  Show this message and exit.

Commands:
 .djangofy      Convert Vue.js webpack project into a django...
  djbuild       Called inside `package.json`
  djstartvueapp Run click commands on bash.
  installvuecli Install vue-cli
  startvueapp   Init vue project via vue-cli
  vuebuild      Build Vue.js project via npm
  vuecheck      Check if node > 5 and npm > 3 are installed
  vuedev        Run frontend dev server via npm
```

1.5 Installation

To install python-vuejs, simply::

```
$ pip install python-vuejs
```

1.5.1 Vue.js - A quick overview

Wrappers around npm and vue. These commands automate the boring stuff of setup vue via vue-cli.:

```
$ pyvue startvueapp myapp
$ cd myapp
$ pyvue vuedev
...
```

Before you go on production run.:

```
$ cd myapp
$ pyvue vuebuild
...
```

1.5.2 Django - A quick overview

Run `pyvue djstartvueapp` into your django project directory:

```
(env) $ pyvue djstartvueapp myapp
...
Enjoy!
```

Command installs all dependencies and make the `myapp` a django app.

1.6 Features

- Vue.js starter via vue-cli
- Django integrations with no dependencies
- Detach backend and frontend with less effort

1.6.1 TODO

- Supporting Flask, eg <https://github.com/taogeT/flask-vue> (without js inside package)
- Supporting other frameworks
- Supporting windows

1.6.2 Credits

This package was created with [Cookiecutter](#) and the [audreyr/cookiecutter-pypackage](#) project template.

2.1 Stable release

To install Python and Vue.js integration, run this command in your terminal:

```
$ pip install python_vuejs
```

This is the preferred method to install Python and Vue.js integration, as it will always install the most recent stable release.

If you don't have `pip` installed, this [Python installation guide](#) can guide you through the process.

2.2 From sources

The sources for Python and Vue.js integration can be downloaded from the [Github repo](#).

You can either clone the public repository:

```
$ git clone git://github.com/cstrap/python_vuejs
```

Or download the [tarball](#):

```
$ curl -OL https://github.com/cstrap/python_vuejs/tarball/master
```

Once you have a copy of the source, you can install it with:

```
$ python setup.py install
```


To use Python and Vue.js integration in a project:

```
pip install python-vue
```

3.1 Django quickstart

Assuming you work inside a virtualenv environment:

```
(env) $ pip install django
...
(env) $ pip install python-vuejs
...
(env) $ django-admin startproject djvue
...
(env) $ cd djvue
(env) $ ./manage.py startapp backend
...
(env) $ pyvue djstartvueapp frontend
...
```

That's it.

`frontend app` is a vue project created via `vue-cli` behind the scene that is converted into a django app.

`djstartvueapp` script adds the `__init__.py` and `urls.py` scripts, then it edits the vue-project build files in order to have the correct django-app directory structure.

3.1.1 Develop process

Working on `frontend app`:

```
(env) $ cd frontend
(env) $ npm run dev # or pyvue vuedev
```

This is decupled from django, used only for develop.

3.1.2 Build process

The build process is:

```
(env) $ cd frontend
(env) $ npm run build # or pyvue vuebuild
(env) $ cd ..
(env) $ ./manage.py collectstatic --noinput
```

You can add the “frontend” build to your pipeline (eg if you use fabric, ansible or others)

Refer to the ‘[Vue.js https://vuejs.org/](https://vuejs.org/)’_ documentation for other stuff not related to python-vuejs wrappers.

The cool thing is that you can detach `frontend` app and deploy it wherever you want. Also, you can move it into another repo.

Happy coding!

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given. You can contribute in many ways:

4.1 Types of Contributions

4.1.1 Report Bugs

Report bugs at https://github.com/cstrap/python_vuejs/issues.

If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

4.1.2 Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with “bug” and “help wanted” is open to whoever wants to implement it.

4.1.3 Implement Features

Look through the GitHub issues for features. Anything tagged with “enhancement” and “help wanted” is open to whoever wants to implement it.

4.1.4 Write Documentation

Python and Vue.js integration could always use more documentation, whether as part of the official Python and Vue.js integration docs, in docstrings, or even on the web in blog posts, articles, and such.

4.1.5 Submit Feedback

The best way to send feedback is to file an issue at https://github.com/cstrap/python_vuejs/issues.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that contributions are welcome :)

4.2 Get Started!

Ready to contribute? Here's how to set up *python_vuejs* for local development.

1. Fork the *python_vuejs* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/python_vuejs.git
```

3. Install your local copy into a virtualenv. Assuming you have virtualenvwrapper installed, this is how you set up your fork for local development:

```
$ mkvirtualenv python_vuejs
$ cd python_vuejs/
$ python setup.py develop
```

4. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally.

5. When you're done making changes, check that your changes pass flake8 and the tests, including testing other Python versions with tox:

```
$ flake8 python_vuejs tests
$ python setup.py test # or make test
$ tox # not ready
```

To get flake8 and tox, just pip install them into your virtualenv.

6. Commit your changes and push your branch to GitHub:

```
$ git add .
$ git commit -m "Your detailed description of your changes."
$ git push origin name-of-your-bugfix-or-feature
```

7. Submit a pull request through the GitHub website.

4.3 Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests.
2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.
3. The pull request should work for Python 2.7, 3.3, 3.4 and 3.5. Check https://travis-ci.org/cstrap/python_vuejs/pull_requests and make sure that the tests pass for all supported Python versions.

4.4 Tips

To run a subset of tests:

```
$ python -m unittest tests.test_python_vuejs
```


5.1 Development Lead

- Christian Strappazzon <lab@strap.it>

5.2 Contributors

None yet. Why not be the first?

6.1 1.0.9 (2017-10-03)

- Fix slash issue #21

6.2 1.0.7 (2017-10-03)

- Better error message
- Logging
- Passing context - Click issue on calling function command

6.3 1.0.6 (2017-05-30)

- Fix broken docs link

6.4 1.0.5 (2017-05-30)

- Docs on readthedocs.org

6.5 1.0.4 (2017-05-30)

- punch.py troubleshooting...

6.6 1.0.3 (2017-05-30)

- Housekeeping

6.7 1.0.2 (2017-05-30)

- Better coverage
- Make commands
- Dropped bumpversion in favour of punch.py

6.8 1.0.1 (2017-05-30)

- Better coverage
- Fix function name after reworking

6.9 1.0.0 (2017-05-29)

Beta release

- Grouping all commands under `pyvue`
- Added unit tests
- Added travisCI configuration
- Added coveralls and coverage configuration
- Docs updated
- Better semver

6.10 0.0.7 (2017-05-24)

- Documentation
- Fix `djstartvueapp`

6.11 0.0.4 (2017-05-24)

- Alpha version

6.12 0.0.3 (2017-05-24)

- Added full django support with single command `djstartvueapp`

6.13 0.0.2 (2017-05-24)

- Added some docs and fix broken link

6.14 0.0.1 (2017-05-23)

- First release on PyPI.
- Provide a set of scripts in order to automate Vue.js project creation.

CHAPTER 7

Indices and tables

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