
django*languageselectDocumentation*

Release 4.0.1

Mounir Messelmeni

Aug 22, 2023

Contents

1	django_languageselect	3
1.1	Requirements	3
1.2	Usage:	3
1.3	Routes:	3
1.4	Tests	4
1.5	Requirements upgrades	4
1.6	Making a new release	4
2	Installation	5
2.1	Stable release	5
2.2	From sources	5
3	Usage	7
4	Contributing	9
4.1	Types of Contributions	9
4.2	Get Started!	10
4.3	Pull Request Guidelines	11
4.4	Tips	11
5	Credits	13
5.1	Development Lead	13
5.2	Contributors	13
6	History	15
6.1	Unreleased	15
6.2	4.0.1 (2023-01-02)	15
6.3	4.0.0 (2023-01-02)	15
6.4	3.0.0 (2021-05-05)	15
6.5	2.0.2 (2021-05-05)	16
6.6	2.0.1 (2021-03-25)	16
6.7	2.0.0 (2021-03-25)	16
6.8	1.0.0 (2019-11-04)	16
6.9	0.1.6 (2018-01-27)	16
6.10	0.1.5 (2018-01-15)	16
6.11	0.1.4 (2017-04-18)	16
6.12	0.1.3 (2016-06-17)	16

Contents:

Simple language select as custom template tag

1.1 Requirements

- “django.middleware.locale.LocaleMiddleware” in MIDDLEWARE_CLASSES / MIDDLEWARE
- “django.core.context_processors.request” in TEMPLATE_CONTEXT_PROCESSORS / TEMPLATES[‘OPTIONS’][‘context_processors’]
- Add ‘django_languageselect’, to INSTALLED_APPS

1.2 Usage:

```
{% load languageselect %} {% languageselect %}
```

1.3 Routes:

The only url provided by this application is “languageselect_index”. Required GET-parameter is “language”, optional GET-parameter is “next”. Next contains the named url to redirect after the language change. This parameter is pre-filled with the current page url. Customization

Feel free to use your own template, just add languageselect/layer.html

- Free software: MIT license

- Documentation: <https://django-languageselect.readthedocs.io>.

1.4 Tests

Tests will be automatically run by travis on commit to master.

They can also be executed locally using docker-compose by running *docker-compose up*

1.5 Requirements upgrades

Check for upgradeable packages by running *docker-compose run --rm python pip-check*

1.6 Making a new release

`bumpversion` is used to manage releases.

Add your changes to the `HISTORY` and run *docker-compose run --rm python bumpversion <major|minor|patch>*, then push (including tags)

2.1 Stable release

To install `django_languageselect`, run this command in your terminal:

```
$ pip install django_languageselect
```

This is the preferred method to install `django_languageselect`, 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 `django_languageselect` can be downloaded from the [Github repo](#).

You can either clone the public repository:

```
$ git clone git://github.com/RegioHelden GmbH/django_languageselect
```

Or download the [tarball](#):

```
$ curl -OL https://github.com/RegioHelden GmbH/django_languageselect/tarball/master
```

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

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


To use `django_languageselect` in a project, add it to *INSTALLED_APP*

```
'django_languageselect',
```

Add this to your `urls.py`

```
url(r'^languageselect/', include('django_languageselect.urls')),
```

Use the `languageselect` tag where you which to show languages list:

```
{% load languageselect %} {% languageselect %}
```


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/RegioHelden/django-languageselect/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

django_languageselect could always use more documentation, whether as part of the official django_languageselect 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/RegioHelden/django-languageselect/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 *django_languageselect* for local development.

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

```
$ git clone git@github.com:your_name_here/django_languageselect.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 django_languageselect
$ cd django_languageselect/
$ 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 django_languageselect tests
$ python setup.py test or py.test
$ tox
```

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, 3.6, 3.7 and for PyPy. Check https://travis-ci.org/RegioHelden/django-languageselect/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_django_languageselect
```


5.1 Development Lead

- Jens Nistler <jens.nistler@regiohelden.de>
- Mounir Messelmeni <mounir.messelmeni@regiohelden.de>

5.2 Contributors

None yet. Why not be the first?

6.1 Unreleased

- Remove support for Python < 3.9
- Remove support for Django 4.0
- Modernize build and test environment

6.2 4.0.1 (2023-01-02)

- Remove support for Django < 3.2
- Remove support for Python < 3.8
- Modernize build and test environment

6.3 4.0.0 (2023-01-02)

Not released due an issue with the deploy action

6.4 3.0.0 (2021-05-05)

- Remove support for Django 2.x
- Remove utf 8 headers as all Python 3 files must be unicode anyways
- Fix CI badge in readme
- Modernize build and test environment

6.5 2.0.2 (2021-05-05)

- Remove setting language in session as this was deprecated since Django 2.x and is broken with Django 3.x
- Add tests for Django 3.2
- Remove old python 2 and Django compatibility code

6.6 2.0.1 (2021-03-25)

- Fix documentation syntax

6.7 2.0.0 (2021-03-25)

- Update packages
- Update Docker env
- Remove support for Django < 2.2

6.8 1.0.0 (2019-11-04)

- Update packages
- Remove Python 2 support
- Add docker container to run tests locally

6.9 0.1.6 (2018-01-27)

- Add missing folders and modules from package

6.10 0.1.5 (2018-01-15)

- Add support for python 3 and Django 2, implementing tests in travis ci

6.11 0.1.4 (2017-04-18)

- Rename package to django_languageselect (backwards incompatible) to be able to import the module

6.12 0.1.3 (2016-06-17)

- Fix session key to store language, Django 1.9 compatibility

CHAPTER 7

Indices and tables

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