django-skeleton-app Documentation

Release 0.1.1

Trey Hunner

Sep 27, 2017

Contents

1 Descriptions of the Files **3**
- 1.1 README.rst . . . 3
- 1.2 CHANGES.rst . . . 3
- 1.3 LICENSE . . . 3
- 1.4 setup.py . . . 3
- 1.5 MANIFEST.in . . . 4
- 1.6 Test files . . . 4
- 1.7 Empty Package Files . . . 4
- 1.8 .editorconfig . . . 4
- 1.9 AUTHORS.rst . . . 5
- 1.10 .gitignore . . . 5
- 1.11 CONTRIBUTING.rst . . . 5
- 1.12 Sphinx Documentation . . . 5
- 1.13 Makefile . . . 5

2 License **7**

3 Indices and tables **9**

This project documents tips for creating an open source Django app.

This project's file structure should not be copied verbatim. Instead start building your repository from the ground up, using the files provided by django-skeleton-app as a guideline. Remember that you can always add more to your repository later.

Contents:

CHAPTER 1

Descriptions of the Files

The Django skeleton app repository contains example files which you can use for inspiration when creating a repository for your own open source Django app.

README.rst

The README file is displayed on Github and should be used to generate the long description for the package which is displayed on PyPI.

CHANGES.rst

The CHANGES file will record version numbers and dates and the changes between each version.

LICENSE

If you don't specify an open source license for your code, all rights will be reserved by default and no one can legally use your code without explicit permission.

The example LICENSE file uses an MIT license which is less strict than the BSD license and more well-known than the Simplified 2-Clause BSD license.

setup.py

The setup.py file is used to install the Python package, upload it to PyPI, and is generally used for running commands related to your Python project.

Some things that must be changed in the example file:

- PyPI package name (often hyphenated)
- Python package name
- Author name and email
- The short description
- The project URL
- Installation requirements (remove if there are none)
- Test requirements (remove if there are none)
- Classifier troves (Python versions, license, and environment)

MANIFEST.in

The MANIFEST.in file specifies the files that should be included when creating and installing a Python package. The example MANIFEST.in file includes all reStructuredText files. This will ensure the README and CHANGES files so the setup.py file works correctly. The rst files in the docs directory will also be included.

Test files

You need test files. Even if you have no tests for your package initially you at least need to create the files that will store the tests. This will make creating tests much easier once you do want to add some.

The tests files you should create:

- runtests.py: stores a function that runs your tests. Change the setup.py file as needed to reflect the name of this file and the function it contains.
- tox.ini: notes your tox settings as well as your pep8/flake8 settings. Tox is used to run your tests against multiple Python environments.
- .coveragerc: stores your settings for coverage.py, the test coverage analyzer
- .travis.yml: configuration file for Travis CI (runs your tests each time you push to your git repository)

Empty Package Files

- packagename/init.py: This should contain the __version__ variable
- packagename/tests/init.py: This should be empty
- packagename/tests/models.py: A models.py file is needed for Django apps
- packagename/tests/tests.py: This will contain the tests eventually

.editorconfig

PEP8 checkers usually handle enforcing Python file formatting, but I like including an .editorconfig file to specify the indentation I use for my INI and YAML files.

See http://editorconfig.org for more details on EditorConfig.

AUTHORS.rst

You may be the only author now, but eventually someone may discover your project and submit a pull request. The least you could do is honor that person with a listing in the authors file.

.gitignore

This is a requirement for nearly all Git repositories.

CONTRIBUTING.rst

The CONTRIBUTING file should explain to new contributors the steps they should take while submitting an issue or pull request. Github shows a link to this file when creating a new issue or pull request on your repository.

Sphinx Documentation

Incomplete documentation is usually better than missing documentation. Use Sphinx to create a skeleton documentation directory so it will be easy to add documentation later. You can host the documentation for free on ReadTheDocs.org.

To generate Sphinx documentation:

```
sphinx-quickstart docs/
```

Makefile

It's nice to have shortcut commands for installing tox and coverage, building documentation, and running tests while generating coverage reports.

CHAPTER 2

License

To the extent possible under law, the author has dedicated all copyright and related and neighboring rights to this software to the public domain worldwide. This software is distributed without any warranty. The CC0 Public Domain Dedication license is available at http://creativecommons.org/publicdomain/zero/1.0/.

CHAPTER 3

Indices and tables

- genindex
- modindex
- search