
pyscaffoldext-dsproject Documentation

Release unknown

Florian Wilhelm

Sep 22, 2022

CONTENTS

1	Contents	3
1.1	pyscaffoldext-dsproject	3
1.2	License	4
1.3	Contributors	5
1.4	Changelog	5
1.5	pyscaffoldext	7
2	Indices and tables	9
	Python Module Index	11
	Index	13

PyScaffold extension that adds a perfect structure for Data Science projects.

CONTENTS

Build Status ReadTheDocs Coveralls PyPI-Server Conda-Forge Downloads Sponsor

1.1 pyscaffoldext-dsproject

[PyScaffold](#) extension tailored for *Data Science* projects. This extension is inspired by [cookiecutter-data-science](#) and enhanced in many ways. The main differences are that it

1. advocates a proper Python package structure that can be shipped and distributed,
2. uses a [conda](#) environment instead of something [virtualenv](#)-based and is thus more suitable for data science projects,
3. more default configurations for [Sphinx](#), [pytest](#), [pre-commit](#), etc. to foster clean coding and best practices.

Also consider using [dvc](#) to version control and share your data within your team. Read [this blogpost](#) to learn how to work with JupyterLab notebooks efficiently by using a data science project structure like this.

The final directory structure looks like:

— AUTHORS.md	<- List of developers and maintainers.
— CHANGELOG.md	<- Changelog to keep track of new features and fixes.
— CONTRIBUTING.md	<- Guidelines for contributing to this project.
— Dockerfile	<- Build a docker container with `docker build .`.
— LICENSE.txt	<- License as chosen on the command-line.
— README.md	<- The top-level README for developers.
— configs	<- Directory for configurations of model & application.
— data	
— external	<- Data from third party sources.
— interim	<- Intermediate data that has been transformed.
— processed	<- The final, canonical data sets for modeling.
— raw	<- The original, immutable data dump.
— docs	<- Directory for Sphinx documentation in rst or md.
— environment.yml	<- The conda environment file for reproducibility.
— models	<- Trained and serialized models, model predictions, or model summaries.
— notebooks	<- Jupyter notebooks. Naming convention is a number (for ordering), the creator's initials and a description, e.g. `1.0-fw-initial-data-exploration`.
— pyproject.toml	<- Build configuration. Don't change! Use `pip install -e .` to install for development or to build `tox -e build`.
— references	<- Data dictionaries, manuals, and all other materials.

(continues on next page)

(continued from previous page)

— reports	<- Generated analysis as HTML, PDF, LaTeX, etc.
└─ figures	<- Generated plots and figures for reports.
— scripts	<- Analysis and production scripts which import the actual PYTHON_PKG, e.g. train_model.
— setup.cfg	<- Declarative configuration of your project.
— setup.py	<- [DEPRECATED] Use `python setup.py develop` to install for development or `python setup.py bdist_wheel` to build.
— src	
└─ PYTHON_PKG	<- Actual Python package where the main functionality goes.
— tests	<- Unit tests which can be run with `pytest`.
— .coveragerc	<- Configuration for coverage reports of unit tests.
— .isort.cfg	<- Configuration for git hook that sorts imports.
— .pre-commit-config.yaml	<- Configuration of pre-commit git hooks.

See a demonstration of the initial project structure under [dsproject-demo](#) and also check out the documentation of [PyScaffold](#) for more information.

1.1.1 Usage

Just install this package with `conda install -c conda-forge pyscaffoldext-dsproject` and note that `putup -h` shows a new option `--dsproject`. Creating a data science project is then as easy as:

```
putup --dsproject my_ds_project
```

The flag `--dsproject` comprises additionally the flags `--markdown`, `--pre-commit` and `--no-skeleton` for convenience.

1.1.2 Making Changes & Contributing

This project uses [pre-commit](#), please make sure to install it before making any changes:

```
conda install pre-commit
cd pyscaffoldext-dsproject
pre-commit install
```

It is a good idea to update the hooks to the latest version:

```
pre-commit autoupdate
```

Please also check PyScaffold's [contribution guidelines](#).

1.2 License

The MIT License (MIT)

Copyright (c) 2019 Florian Wilhelm

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.3 Contributors

- Florian Wilhelm <florian.wilhelm@gmail.com>
- Anderson Bravalheri <andersonbravalheri@gmail.com>
- Sterling Baird <sterling.baird@icloud.com>

1.4 Changelog

1.4.1 Version 0.7.2

- Fix inconsistency by renaming data/preprocessed to data/processed.

1.4.2 Version 0.7.1

- Remove separate Description section in README.md
- Remove old wheels in Dockerfile and have a comment about EXPOSE.

1.4.3 Version 0.7

- Added a Dockerfile to the project structure

1.4.4 Version 0.6.1

- Latest updates from PyScaffold
- Fix KeyError: 'long-description', issue #20

1.4.5 Version 0.6

- Curated the list of packages in `environment.yml`
- Editable install of package directly in `environment.yml`

1.4.6 Version 0.5.1

- Fix `isort` when namespaces are used
- Fix wrongs docs, i.e. `rst` -> `md`

1.4.7 Version 0.5

- Updated to use PyScaffold v4 API
- Updated Cirrus CI config file
- Added new useful tasks to `tox.ini`
- Renamed `environment.yaml` to `environment.yml`
- Added [Yellowbrick](#) to `environment.yml`

1.4.8 Version 0.4.1

- Fix problem with additional new lines in Windows

1.4.9 Version 0.4

- Add a `.gitignore` also in the data folder
- Changed wording in `README.md`

1.4.10 Version 0.3.1

- Fixed wrong folder name in `README.md` template

1.4.11 Version 0.3

- Added a `configs` folder

1.4.12 Version 0.2.1

- Fixed some docs
- Adjusted line lengths
- Some cosmetic changes

1.4.13 Version 0.2

- Reworked the general layout some more

1.4.14 Version 0.1

- Initial commit

1.5 pyscaffoldext

1.5.1 pyscaffoldext namespace

Subpackages

pyscaffoldext.dsproject package

Subpackages

pyscaffoldext.dsproject.templates package

Module contents

`pyscaffoldext.dsproject.templates.readme_md(opts: Dict[str, Any]) → str`

Adds a basic README.md

Parameters

opts – given options, see `create_project` for an extensive list.

Returns

file content as string

Submodules

pyscaffoldext.dsproject.extension module

Module contents

INDICES AND TABLES

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

PYTHON MODULE INDEX

p

`pyscaffoldext`, [7](#)

`pyscaffoldext.dsproject`, [7](#)

`pyscaffoldext.dsproject.templates`, [7](#)

INDEX

M

module

 pyscaffoldext, [7](#)

 pyscaffoldext.dsproject, [7](#)

 pyscaffoldext.dsproject.templates, [7](#)

P

pyscaffoldext

 module, [7](#)

pyscaffoldext.dsproject

 module, [7](#)

pyscaffoldext.dsproject.templates

 module, [7](#)

R

readme_md() (in module *pyscaffold-
ext.dsproject.templates*), [7](#)