Cookiecutter for Birdhouse Documentation

Release 0.2.0

Birdhouse

May 25, 2018

Contents:

Todo:

- Review all this README.
- Add Cookiecutter logo with a Bird.

A Cookiecutter template for a Birdhouse bird package (PyWPS server).

- GitHub repo: https://github.com/bird-house/cookiecutter-birdhouse/
- Documentation: http://cookiecutter-birdhouse.readthedocs.io/en/latest/
- Free software: BSD license

CHAPTER 1

Features

- Ready-made PyWPS service (a bird).
- Pre-configured `.travis.yml` for Travis-CI
- Sphinx docs: Documentation ready for generation with, for example, ReadTheDocs

CHAPTER 2

Installation

Prior to installing cookiecutter-birdhouse, the cookiecutter package must be installed in your environment. This is achieved via the following command:

```
$ conda install -c conda-forge cookiecutter
```

With cookiecutter installed, the cookiecutter-birdhouse template can be installed with:

```
$ cookiecutter https://github.com/bird-house/cookiecutter-birdhouse.git
```

Once cookiecutter clones the template, you will be asked a series of questions related to your project:

```
$ full_name [Full Name]: Enter your full name.

$ email [Email Address]: Enter your email address.

$ github_username [bird-house]: Accept the default or enter your github username.

$ project_name [Babybird]: The name of your new bird.

$ project_slug [babybird]: The name of your bird used as Python package.

$ project_short_description [Short description]: Enter a short description about your␣
↪project.

$ version [0.1.0]: Enter the version number for your application.

$ http_port [5000]: The HTTP port on which your service will be accessible.

$ https_port [25000]: The HTTPS port on which your service will be accessible.

$ output_port [8090]: The HTTP port on which your service outputs will be accessible.
```

CHAPTER 3

Usage

After answering the questions asked during installation, a *bird* Python package will be created in your current working directory. This package will contain a configurable PyWPS service with some initial test processes.

Then:

- Create a repo and put it there.
- Add the repo to your Travis-CI account.
- Add the repo to your ReadTheDocs account + turn on the ReadTheDocs service hook.

For more details, see the cookiecutter-pypackage tutorial.

CHAPTER 4

Development

If you want to extend the cookiecutter template then prepare your development environment as follows:

```
# clone repo
$ git clone git@github.com:bird-house/cookiecutter-birdhouse.git

# change into repo
$ cd cookiecutter-birdhouse

# create conda environment
$ conda env create -f environment.yml

# activate conda environment
$ source activate cookiecutter-birdhouse

# run tests
$ make test

# bake a new bird with default settings
$ make bake

# the new "baked" bird is created in the cookies folder
$ ls -l cookies/
babybird

# well ... you know what to do with a bird :)

# finally you may clean it all up
$ make clean
```

CHAPTER 5

Bump a new version

Make a new version of this Cookiecutter in the following steps:

```
* Make sure everything is commit to GitHub.
* Update `CHANGES.rst` with the next version.
* Dry Run: `bumpversion --dry-run --verbose --new-version 0.2.1 patch`
* Do it: `bumpversion --new-version 0.2.1 patch`
```

See the bumpversion documentation for details.