

---

# **rewardify-base Documentation**

***Release 0.0.0.0***

**Jonas Teufel**

**Apr 07, 2020**



---

## Contents:

---

<b>1</b>	<b>rewardify-base</b>	<b>1</b>
1.1	Features . . . . .	1
1.2	Credits . . . . .	1
<b>2</b>	<b>Installation</b>	<b>3</b>
2.1	Stable release . . . . .	3
2.2	From sources . . . . .	3
<b>3</b>	<b>Usage</b>	<b>5</b>
<b>4</b>	<b>Contributing</b>	<b>7</b>
4.1	Types of Contributions . . . . .	7
4.2	Get Started! . . . . .	8
4.3	Pull Request Guidelines . . . . .	9
4.4	Tips . . . . .	9
4.5	Deploying . . . . .	9
<b>5</b>	<b>Credits</b>	<b>11</b>
5.1	Development Lead . . . . .	11
5.2	Contributors . . . . .	11
<b>6</b>	<b>History</b>	<b>13</b>
6.1	0.0.0 (2019-06-08) . . . . .	13
6.2	0.1.0 (2019-06-14) . . . . .	13
6.3	0.1.1 (2019-06-14) . . . . .	13
6.4	0.2.0 (2019-06-15) . . . . .	13
6.5	0.2.1 (2019-06-16) . . . . .	14
6.6	0.2.2 (2019-06-16) . . . . .	14
6.7	0.2.3 (2019-06-16) . . . . .	14
6.8	0.2.4 (2019-06-16) . . . . .	14
6.9	0.2.5 (2019-06-16) . . . . .	14
6.10	0.2.6 (2019-06-16) . . . . .	14
6.11	0.2.10 (2019-06-16) . . . . .	14
6.12	0.2.11 (2019-06-16) . . . . .	14
6.13	0.2.12 (2019-06-16) . . . . .	14
6.14	0.2.13 (2020-04-06) . . . . .	15
6.15	0.2.14 (2020-04-07) . . . . .	15



Base frame reward system

- Free software: BSD license
- Documentation: <https://rewardify-base.readthedocs.io>.

## 1.1 Features

- TODO

## 1.2 Credits

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



### 2.1 Stable release

To install rewardify-base, run this command in your terminal:

```
$ pip install rewardify_base
```

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

You can either clone the public repository:

```
$ git clone git://github.com/thel6thpythonist/rewardify_base
```

Or download the [tarball](#):

```
$ curl -OL https://github.com/thel6thpythonist/rewardify_base/tarball/master
```

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

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





## CHAPTER 3

---

### Usage

---

To use rewardify-base in a project:

```
import rewardify_base
```



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/the16thpythonist/rewardify\\_base/issues](https://github.com/the16thpythonist/rewardify_base/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

rewardify-base could always use more documentation, whether as part of the official rewardify-base 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/the16thpythonist/rewardify\\_base/issues](https://github.com/the16thpythonist/rewardify_base/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 *rewardify\_base* for local development.

1. Fork the *rewardify\_base* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/rewardify_base.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 rewardify_base
$ cd rewardify_base/
$ 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 rewardify_base 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, 3.4, 3.5 and 3.6, and for PyPy. Check [https://travis-ci.org/the16thpythonist/rewardify\\_base/pull\\_requests](https://travis-ci.org/the16thpythonist/rewardify_base/pull_requests) and make sure that the tests pass for all supported Python versions.

## 4.4 Tips

To run a subset of tests:

```
$ py.test tests.test_rewardify_base
```

## 4.5 Deploying

A reminder for the maintainers on how to deploy. Make sure all your changes are committed (including an entry in HISTORY.rst). Then run:

```
$ bumpversion patch # possible: major / minor / patch
$ git push
$ git push --tags
```

Travis will then deploy to PyPI if tests pass.



### 5.1 Development Lead

- Jonas Teufel <jonseb1998@gmail.com>

### 5.2 Contributors

None yet. Why not be the first?





#### 6.1 0.0.0 (2019-06-08)

- First release on PyPI.

#### 6.2 0.1.0 (2019-06-14)

- Initial version

#### 6.3 0.1.1 (2019-06-14)

- Fixed bug in setup.py, which caused the “backends” package to be excluded

#### 6.4 0.2.0 (2019-06-15)

- Internal changes in the recycle method for rewards
- Added a “user\_exists” method to the facade, which will return, whether or not the given user name exists in the database or not
- **Added “effect” field to Reward class**
  - An optional string item “effect” can be added to the config of any reward. This string should contain special syntax, for what effect the reward usage should have on the rewardify system
  - Currently granting the user gold and dust are supported effects
- Added more methods for handling rewards to the facade, which include buying, using and recycling rewards

## 6.5 0.2.1 (2019-06-16)

- Fixed, that the facade method for buying a pack internally called the method for adding a pack, thus not actually spending gold on it

## 6.6 0.2.2 (2019-06-16)

- Added “user.save()” calls to all the methods of the facade, that modify the models, so that the modifications actually get taken over into the database

## 6.7 0.2.3 (2019-06-16)

- Extended the clean method for the Environment setup of the test cases to also remove sub folders

## 6.8 0.2.4 (2019-06-16)

- Fixed a bug, which causes the “open\_pack” method of the User model not to work properly

## 6.9 0.2.5 (2019-06-16)

- Rethought the effect system: The “use” method of Reward now returns a list with functions, which contain the effects.
- Fixed a bug, where the facade method for using a reward did not execute the effects of the reward properly

## 6.10 0.2.6 (2019-06-16)

- Changed the install directory for linux from “~/rewardify” to “/opt/rewardify” as the previous one isnt working with python path handling.

## 6.11 0.2.10 (2019-06-16)

- Added a method “init” to the EnvironmentConfig singleton, which will create the connection to the database

## 6.12 0.2.11 (2019-06-16)

- Fixed, that the pointer to the last accounted action was not being saved within the ForestBackend

## 6.13 0.2.12 (2019-06-16)

- Fixed the gold evaluation for the reward effects, it was accidentally set to constant value

## 6.14 0.2.13 (2020-04-06)

- Added the new method “rewardify.backends.combine” which offers the function “combine\_backends” which can be used to use multiple backend classes at the same time.

## 6.15 0.2.14 (2020-04-07)

- Added the GoalBackend



## CHAPTER 7

---

### Indices and tables

---

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