

---

# **Magnificent Messages Documentation**

**Artem Vasin**

**Jan 16, 2020**



---

## Contents

---

<b>1</b>	<b>Badges</b>	<b>3</b>
1.1	PyPI . . . . .	3
1.2	Master . . . . .	3
1.3	Develop . . . . .	3
<b>2</b>	<b>Install</b>	<b>5</b>
<b>3</b>	<b>Readme</b>	<b>7</b>
<b>4</b>	<b>Installation</b>	<b>9</b>
<b>5</b>	<b>Usage</b>	<b>11</b>
<b>6</b>	<b>Contributing</b>	<b>13</b>
6.1	Types of Contributions . . . . .	13
6.2	Get Started! . . . . .	14
6.3	Pull Request Guidelines . . . . .	15
<b>7</b>	<b>Credits</b>	<b>17</b>
7.1	Development Lead . . . . .	17
7.2	Contributors . . . . .	17



Contents:



# CHAPTER 1

---

Badges

---

## 1.1 PyPI

## 1.2 Master

## 1.3 Develop





## CHAPTER 2

---

### Install

---

```
pip install django-magnificent-messages
```



## CHAPTER 3

---

### Readme

---

All-in-one Django message app! Coming soon!



## CHAPTER 4

---

### Installation

---

At the command line:

```
$ easy_install django-magnificent-messages
```

Or, if you have virtualenvwrapper installed:

```
$ mkvirtualenv django-magnificent-messages  
$ pip install django-magnificent-messages
```



To use Magnificent Messages in a project, notifications\_add it to your *INSTALLED\_APPS*:

```
INSTALLED_APPS = (  
    ...  
    'django_magnificent_messages.apps.DjangoMagnificentMessagesConfig',  
    ...  
)
```





Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given. You can contribute in many ways:

## 6.1 Types of Contributions

### 6.1.1 Report Bugs

Report bugs at <https://github.com/NoNameItem/django-magnificent-messages/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.

### 6.1.2 Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with “bug” is open to whoever wants to implement it.

### 6.1.3 Implement Features

Look through the GitHub issues for features. Anything tagged with “feature” is open to whoever wants to implement it.

### 6.1.4 Write Documentation

Magnificent Messages could always use more documentation, whether as part of the official Magnificent Messages docs, in docstrings, or even on the web in blog posts, articles, and such.

### 6.1.5 Submit Feedback

The best way to send feedback is to file an issue at <https://github.com/NoNameItem/django-magnificent-messages/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 :)

## 6.2 Get Started!

Ready to contribute? Here's how to set up *django-magnificent-messages* for local development.

1. Fork the *django-magnificent-messages* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/django-magnificent-messages.git
```

3. Install development requirements:

```
$ pip install requirements_dev.txt
```

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, bandit and the tests, including testing other Python versions with tox:

```
$ flake8 cool_django_auth_ldap tests
$ bandit -r .
$ python manage.py test --settings tests.settings
```

To get flake8 and bandit, just pip install it into your virtualenv (Should be installed if you use requirements\_dev.txt).

6. Commit your changes and push your branch to GitHub:

```
$ git notifications_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.

## 6.3 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 update docs/ accordingly.
3. The pull request should work for all supported python versions. Check [https://travis-ci.org/NoNameItem/django-magnificent-messages/pull\\_requests](https://travis-ci.org/NoNameItem/django-magnificent-messages/pull_requests) and make sure that the tests pass.



## CHAPTER 7

---

### Credits

---

#### 7.1 Development Lead

- Artem Vasin <nonameitem@me.com>

#### 7.2 Contributors

None yet. Why not be the first?