

---

# **Django REST Framework Read Write Serializers Documentation**

*Release 1.0.4*

**Vinta Software**

**Sep 02, 2020**



---

## Contents

---

<b>1</b>	<b>Django REST Framework Read &amp; Write Serializers</b>	<b>3</b>
1.1	Documentation . . . . .	3
1.2	License . . . . .	3
1.3	How To Contribute . . . . .	3
1.4	Reporting Security Issues . . . . .	4
1.5	Getting Help . . . . .	4
1.6	Commercial Support . . . . .	4
<b>2</b>	<b>Installation</b>	<b>5</b>
<b>3</b>	<b>Usage</b>	<b>7</b>
<b>4</b>	<b>Credits</b>	<b>9</b>
4.1	Development Lead . . . . .	9
4.2	Contributors . . . . .	9
<b>5</b>	<b>Change Log</b>	<b>11</b>
5.1	Unreleased . . . . .	11
5.2	[1.0.4] - 2020-09-02 . . . . .	11
5.3	[1.0.2] - 2019-08-30 . . . . .	11
5.4	[1.0.1] - 2019-01-16 . . . . .	12
5.5	[0.1.1] - 2018-11-12 . . . . .	12
5.6	[0.1.0] - 2018-02-21 . . . . .	12



Contents:



---

## Django REST Framework Read & Write Serializers

---

Generic views, viewsets and mixins that extend the Django REST Framework ones adding separated serializers for read and write operations.

### 1.1 Documentation

The full documentation is at <https://drf-rw-serializers.readthedocs.org>.

### 1.2 License

The code in this repository is licensed under the MIT unless otherwise noted.

Please see `LICENSE.txt` for details.

### 1.3 How To Contribute

Contributions are very welcome.

Please read [How To Contribute](#) for details.

PR description template should be automatically applied if you are sending PR from github interface; otherwise you can find it at [PULL\\_REQUEST\\_TEMPLATE.md](#)

Issue report template should be automatically applied if you are sending it from github UI as well; otherwise you can find it at [ISSUE\\_TEMPLATE.md](#)

## 1.4 Reporting Security Issues

Please do not report security issues in public. Please email [contact@vinta.com.br](mailto:contact@vinta.com.br).

## 1.5 Getting Help

Have a question about this repository? Please create an issue!

## 1.6 Commercial Support

This project, as other Vinta open-source projects, is used in products of Vinta clients. We are always looking for exciting work, so if you need any commercial support, feel free to get in touch: [contact@vinta.com.br](mailto:contact@vinta.com.br)

Copyright (c) 2020 Vinta Serviços e Soluções Tecnológicas Ltda



## CHAPTER 2

---

### Installation

---

Django REST Framework is a dependency for this package, so before installing it you need to have Django REST Framework installed as a dependency. You can see how to install it by following [these instructions](#)

After installing Django REST Framework you can now follow these steps:

At the command line:

```
$ easy_install drf-rw-serializers
```

Or, if you have virtualenvwrapper installed:

```
$ mkvirtualenv drf-rw-serializers
$ pip install drf-rw-serializers
```



To use Django REST Framework Read & Write Serializers in a project, you just need to import some of the generic classes and use them to build your own endpoints the same way you do with Django REST Framework generic views, viewsets and mixins. The only difference is that instead of defining only one serializer class for the view, now you can define a `read_serializer_class` and a `write_serializer_class` that be used according to the method you are implementing.

Eg.:

```
from drf_rw_serializers import generics
from .models import MyModel
from .serializers import MyReadSerializer, MyWriteSerializer

class MyModelListCreateView(generics.ListCreateAPIView):
    queryset = MyModel.objects.all()
    read_serializer_class = MyModelReadSerializer
    write_serializer_class = MyModelWriteSerializer
```

If you need to dynamically override the serializers you can override the following methods the same way you do with `get_serializer_class` and `get_serializer` from Django REST Framework generic classes:

- `get_read_serializer_class`
- `get_write_serializer_class`
- `get_read_serializer`
- `get_write_serializer`

The `drf_rw_serializers` classes implementation doesn't break any of the features Django REST Framework implements, so you can use the same way you use DRF classes, but with the read and write serializers extra feature.

Eg.:

```
from drf_rw_serializers import generics
from .models import MyModel
from .serializers import MyWriteSerializer
```

(continues on next page)

(continued from previous page)

```
class MyModelListCreateView(generics.ListCreateAPIView):
    queryset = MyModel.objects.all()
    # this still works the way it works with DRF ListCreateAPIView
    serializer_class = MyModelWriteSerializer
```

### 4.1 Development Lead

- Hugo Bessa <hugo@vinta.com.br>

### 4.2 Contributors

- Amit Garu <amitgaru2@gmail.com>
- Hebert Júlio <hebertjulio@yahoo.com.br>



### 5.1 Unreleased

### 5.2 [1.0.4] - 2020-09-02

#### 5.2.1 Added

- Move from model-mommy to model-bakery
- Support Django 3.1

#### 5.2.2 Removed

- Drop support to Django 1.10
- Drop support to Python 2.7
- Drop support to Python 3.6

### 5.3 [1.0.2] - 2019-08-30

#### 5.3.1 Added

- Support Django 2.2

## 5.4 [1.0.1] - 2019-01-16

### 5.4.1 Added

- Fixed bug in retrieval of success headers in create endpoints (breaking change)
- Updated dependencies

## 5.5 [0.1.1] - 2018-11-12

### 5.5.1 Added

- Support Django 2.1

## 5.6 [0.1.0] - 2018-02-21

### 5.6.1 Added

- First release on PyPI.