
django-vote Documentation

Release 1.0

shellfly

Dec 17, 2022

Contents

1	Getting Started	3
2	The API	5
2.1	Aggregation	6
3	REST API	7
4	Changelog	9
4.1	2.4.0 (2022.12.17)	9
4.2	2.3.0 (2022.02.06)	9
4.3	2.2.0 (2020.05.20)	9
4.4	2.1.7(2018.05.08)	9
4.5	2.1.6(2017.12.20)	9
4.6	2.1.5(2017.10.08)	10
4.7	2.1.4(2017.02.09)	10
4.8	2.1.2(2016.11.02)	10
4.9	2.0.0(2016.07.15)	10
4.10	1.1.3(2016.03.17)	10
4.11	1.1.1(2015.12.11)	10
4.12	1.1.0(2015.11.12)	10
4.13	1.0.9(2015.09.24)	10
4.14	1.0.5(2014.07.09)	10
4.15	1.0.4(2014.07.09)	11
4.16	1.0.3 (2014.07.08)	11
4.17	1.0.2 (2014.07.08)	11
4.18	1.0 (2014.07.07)	11
5	Indices and tables	13
Index		15

`django-vote` is a reusable Django application designed to voting your django model easy and fun.

CHAPTER 1

Getting Started

To get started using django-vote simply install it with pip:

```
$ pip install django-vote
```

Add "vote" to your project's INSTALLED_APPS setting.

And then to any model you want vote on do the following:

```
from django.db import models

from vote.models import VoteModel

class Comment(VoteModel, models.Model):
    # ... fields here
```

Run:

```
./manage.py makemigrations
./manage.py migrate
```


CHAPTER 2

The API

After you've got your `VoteModel` added to your model you can start playing around with the API.

```
class VotableManager([through=None, verbose_name="Votes", field_name='votes', extra_field=None])
```

Parameters

- **verbose_name** – The verbose_name for this field.
- **through** – The through model
- **field_name** – The field name added to the query
- **extra_field** – The field on your model. It will be updated when up or down

up (user_id)

This adds a vote to an object by the user. `IntegrityError` will be raised if the user has voted before:

```
>>> comments.votes.up(user)
```

down (user_id)

Removes the vote from an object. No exception is raised if the user doesn't have voted the object.

delete (user_id)

Removes the vote from an object. No exception is raised if the user doesn't have voted the object.

exists (user_id, action=UP)

Check if user has voted the instance before.

all (user_id, action=UP)

Get all instances voted by the specify user.

user_ids (action=UP)

Get all user_ids voted the instance

count (action=UP)

The count of all votes for an object.

get (*user_id*)

Get the whole Vote object for the user. Returns None if no vote present.

annotate (*queryset=None*, *user_id=None*, *reverse=True*, *sort=True*)

Add annotation data to the queryset

2.1 Aggregation

Django does not support aggregation with GenericRelation [currently](#) but you still can use annotate:

```
>>> Comment.objects.filter(article__id=article_id).annotate(num_votes=Count('votes__user'))
```

CHAPTER 3

REST API

There is a *VoteMixin* for you to easily add rest api to conduct vote:

```
from vote.views import VoteMixin

class CommentViewSet(VoteMixin, ModelViewSet):
    # ... fields here
```


CHAPTER 4

Changelog

4.1 2.4.0 (2022.12.17)

- Add template tag to get vote count

4.2 2.3.0 (2022.02.06)

- Add support for Django 4.0

4.3 2.2.0 (2020.05.20)

- Add post_vote signal
- Add VoteMixin for easily write vote api
- Drop support for Django < 2.0
- Add Django 3.0 test

4.4 2.1.7(2018.05.08)

- fix template tag error on Django 2.0

4.5 2.1.6(2017.12.20)

- fix error on Django 2.0

4.6 2.1.5(2017.10.08)

- rename user_vote to get

4.7 2.1.4(2017.02.09)

- Support vote down

4.8 2.1.2(2016.11.02)

- Add missing migration files
- add more tests

4.9 2.0.0(2016.07.15)

- use user_id on vote model, Drop support for Django < 1.7

4.10 1.1.3(2016.03.17)

- fix decrease to negative bug

4.11 1.1.1(2015.12.11)

- Python 3 support

4.12 1.1.0(2015.11.12)

- add api *all*, return all instances voted by specify user.

4.13 1.0.9(2015.09.24)

- add extra field. When up and down, the extra field on parasite model will be updated

4.14 1.0.5(2014.07.09)

- change default order_by to ‘-id’

4.15 1.0.4(2014.07.09)

- enable using custom field name for VotableManager
- fix empty queryset bug

4.16 1.0.3 (2014.07.08)

- add compat code

4.17 1.0.2 (2014.07.08)

- add unit test

4.18 1.0 (2014.07.07)

CHAPTER 5

Indices and tables

- genindex
- modindex
- search

Index

A

`all()` (*VotableManager method*), 5
`annotate()` (*VotableManager method*), 6

C

`count()` (*VotableManager method*), 5

D

`delete()` (*VotableManager method*), 5
`down()` (*VotableManager method*), 5

E

`exists()` (*VotableManager method*), 5

G

`get()` (*VotableManager method*), 5

U

`up()` (*VotableManager method*), 5
`user_ids()` (*VotableManager method*), 5

V

`VotableManager` (*built-in class*), 5