
Django Crawfish

Crawford Leeds

May 13, 2022

CONTENTS

1	Overview	1
1.1	Installation	1
1.2	Basic Usage	1
2	Models	3
2.1	Models	3
2.2	Fields	3
3	Middleware	5
3.1	ErrorPagesMiddleware	5
3.2	Future Plans	5
4	Mixins	7
4.1	ApiErrorsMixin	7
4.2	DataclassMappingMixin	7
5	Crawfish API	9
6	Indices and tables	11

OVERVIEW

1.1 Installation

Using pip:

```
$ pip install django-crawfish
```

1.2 Basic Usage

Add *crawfish* to `INSTALLED_APPS` in your settings. Crawfish does not require installation in any particular order.

```
INSTALLED_APPS = [  
    # Other apps  
    "crawfish"  
]
```


MODELS

Crawfish includes base models and fields for better Django models.

2.1 Models

```
from crawfish.models import BaseModel
```

2.2 Fields

```
from crawfish.models import EmailField
```


MIDDLEWARE

Crawfish includes base models and fields for better Django models.

3.1 ErrorPagesMiddleware

Automatically override response body when requests are made using header *Content-Type: application/json* that generates a response with status code 403, 404, or 500 response.

Add to your middleware list either last or as late as possible. This is to ensure this middleware is run early in the response cycle to avoid suppressing any other middleware in the request cycle.

More information available in the [Django documentation](#).

```
MIDDLEWARE = [  
    # other middleware  
    "crawfish.middleware.ErrorPagesMiddleware"  
]
```

3.2 Future Plans

- Configure static or dynamic response body content in settings.
- Decorator to enable/disable middleware on individual views.

Crawfish includes base models and fields for better Django models.

4.1 ApiErrorsMixin

```
from crawfish.mixins import ApiErrorsMixin
```

usage:

```
class MyModel(ApiErrorsMixin, models.Model):  
    ...
```

ApiErrorsMixin adds the following attributes to the model:

- **errors**: a list of errors
- **error_messages**: a dictionary of error messages

WIP: This ^^ was generated entirely with GitHub Copilot. Come back soon for proper documentation.

4.2 DataclassMappingMixin

```
from crawfish.mixins import DataclassMappingMixin
```

usage:

```
@dataclass  
class MyModel(DataclassMappingMixin):  
    field1: str  
    field2: str  
    ...
```

WIP

CRAWFISH API

Test

INDICES AND TABLES

- `genindex`
- `modindex`