
latest
Release 0.2

Sep 27, 2017

Contents

1	Installation	3
2	Shotgun Loading	5
3	Shotgun Firing	7
4	Configuration	9

Created by [Stephen McDonald](#)

`django-shotgun` is a Django application that provides the ability to validate HTTP status codes across your entire Django application. `django-shotgun` consists of two parts, a [management command](#) called `load_shotgun` that prepares `django-shotgun` with a snapshot of your database to use in the [Django test case](#), and the test case itself which spiders your Django site testing for valid HTTP status codes for every URL it finds.

CHAPTER 1

Installation

Assuming you have [pip](#) installed, the easiest method is to install directly from pypi by running the following command:

```
$ pip install -U django-shotgun
```

Otherwise you can check out the source directly and install it via:

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

Once installed you can then add `django_shotgun` to your `INSTALLED_APPS`.

CHAPTER 2

Shotgun Loading

`django-shotgun` is most effective when used in conjunction with real data. This is in contrast to the approach a Django test case will take where it purposely does not use the project's real database to run tests. `django-shotgun` deals with this by providing a management command called `load_shotgun` which dumps the contents of your database to a fixture called `shotgun.json`. Under the hood, `load_shotgun` extends the built-in `dumppdata` command and supports the same options, with one extra option called `path` which allows you to define where the fixture file should be saved to:

```
$ python manage.py load_shotgun --path=/path/to/fixtures/
```

If the `path` option is omitted, the fixture file will be saved to the root of the current project.

CHAPTER 3

Shotgun Firing

The `django-shotgun` test case is a standard Django test case and can therefore be run by calling the built-in test management command:

```
$ python manage.py test
```

The `django-shotgun` test case will first test the URL `/` and parse URLs found in `<a>` tags from the resulting response. It then continues this process for each URL it finds until all URLs have been tested. `<form>` tags and their field elements are also parsed and each form is tested with both a `GET` and `POST`, first without any field data and then with test values for each form field. The test case will fail if any response has a status code other than `200`.

CHAPTER 4

Configuration

The following settings can be defined in your project's `settings` module to control the behaviour of `django-shotgun`.

- `SHOTGUN_EXCLUDE_APPS` - A list of app names to exclude when generating the shotgun fixtures file. Defaults to `[]`.
- `SHOTGUN_EXCLUDE_URLS` - A list of URLs to exclude from checking. Defaults to `[]`.
- `SHOTGUN_FIXTURE_NAME` - The name of the fixture file that will be saved and loaded from. Defaults to `shotgun.json`.
- `SHOTGUN_FIXTURE_PATH` - The path to where the fixture file will be saved and loaded from. Defaults to the project's root directory.
- `SHOTGUN_ROOT_URL` - The first URL that the test case will request. Defaults to `/`.