
django-testrecorder Documentation

Release 0.1

PyDevUA

June 05, 2013

CONTENTS

1	Overview	3
2	Installation	5
3	Usage	7
4	Settings	9
4.1	RECORDER_FIXTURES	9
4.2	RECORDER_IGNORE	9
4.3	RECORDER_AUTOLOGIN	9
4.4	RECORDER_AUTH	9
4.5	RECORDER_FILES_PATH	10
4.6	RECORDER_TEST_FORM_VALIDATION	10
4.7	RECORDER_TEST_EMAIL_SENDING	10
5	Indices and tables	11

Contents:

OVERVIEW

Django-testrecorder is the Django application that allow you just use your project and get ready tests. It can be useful if you are don't know and don't want to know nothing about how write tests, or if you need some simple tests just to test response status, or easy get a skeleton for future tests.

INSTALLATION

For installation get source code from [repo](#), and use `python setup.py install`. Or you can install with `pip install django-testrecorder`, but it can contain not up-to-date version.

Edit `settings.py` to add `'testrecorder'` to `INSTALLED_APPS`.

USAGE

To start server use command `python manage.py recordserver`. This is just wrapper for django `testserver` command, so any data in current database will not be affected.

Now you can visit `http://127.0.0.1:8000/` and all your actions will be recorded and then you can get tests for this actions.

SETTINGS

4.1 RECORDER_FIXTURES

You can add fixtures that will be loaded automatically.

```
RECORDER_FIXTURES = [  
    'test1.json',  
    'test2.json',  
    'test3.json',  
    'test4.json'  
]
```

4.2 RECORDER_IGNORE

You can set URLs that will not be recorded.

```
RECORDER_IGNORE = (  
    '^/favicon.ico',  
    '^/admin.*',  
)
```

4.3 RECORDER_AUTOLOGIN

4.4 RECORDER_AUTH

Also you can set auto-login. For default auth-backend this looks like this:

```
RECORDER_AUTOLOGIN = True  
RECORDER_AUTH = {  
    'username': 'admin',  
    'password': 'admin'  
}
```

4.5 RECORDER_FILES_PATH

If you upload some files during test recording they will be saved in `MEDIA_ROOT/test/` and used in tests. You can change path with `RECORDER_FILES_PATH` setting:

```
RECORDER_FILES_PATH = 'files_to_test/'
```

4.6 RECORDER_TEST_FORM_VALIDATION

You can turn on testing form validation with `RECORDER_TEST_FORM_VALIDATION`. It is turned off by default, because is experimental feature. This does not call any lazy objects in context or form validation before request finished and just check in test if form have same validation state that was recorded.

4.7 RECORDER_TEST_EMAIL_SENDING

You can turn on testing of emails sending with `RECORDER_TEST_EMAIL_SENDING`. In this case `django.core.mail.backends.locmem.EmailBackend` will be set as `EMAIL_BACKEND` and `testrecorder` will save number of sent emails.

INDICES AND TABLES

- *genindex*
- *modindex*
- *search*