pytest-ngrok Documentation

Release 0.0.4

Apkawa

Jan 20, 2022

CONTENTS

1 Installation **3**

2 Usage **5**

3 Contributing **7**
- 3.1 Initialize . . . 7
- 3.2 Run tests . . . 7
- 3.3 Update version . . . 7
- 3.4 Publish pypi . . . 7

4 Fixtures **9**
- 4.1 `ngrok` . . . 9
- 4.2 `live_server_ngrok` . . . 10
- 4.3 `ngrok_bin` . . . 10
- 4.4 `ngrok_install_url` . . . 10
- 4.5 `ngrok_allow_install` . . . 11

5 Indices and tables **13**

Pytest integration for ngrok.io

</br>

pytest integration for ngrok.io

CHAPTER

ONE

INSTALLATION

from PyPi

```
pip install pytest-ngrok
```

or from git

```
pip install -e git+https://githib.com/Apkawa/pytest-ngrok.git#egg=pytest-ngrok
```

CHAPTER

TWO

USAGE

```
import pytest

from urllib.error import HTTPError
from urllib.request import urlopen

def test_ngrok(ngrok, httpserver):
    httpserver.expect_request("/foobar").respond_with_data("ok")
    remote_url = ngrok(httpserver.port)
    assert urlopen(remote_url + "/foobar").read() == b'ok'


def test_ngrok_context_manager(ngrok, httpserver):
    # example local server
    httpserver.expect_request("/foobar").respond_with_data("ok")
    with ngrok(httpserver.port) as remote_url:
        _test_url = str(remote_url) + '/foobar'
        assert urlopen(_test_url).read() == b'ok'

    # Connection closes
    pytest.raises(HTTPError, urlopen, _test_url)
```

With `pytest-django` can use fixture `live_server_ngrok`

```
def test_server(live_server_ngrok):
    assert live_server_ngrok.url.endswith('ngrok.io')
```

CHAPTER

THREE

CONTRIBUTING

3.1 Initialize

```
python -m venv /tmp/.venv/pytest-ngrok
source /tmp/.venv/pytest-ngrok/activate
pip install -r requirements-dev.txt
```

3.2 Run tests

```
pytest
tox
```

3.3 Update version

```
python setup.py bumpversion
```

3.4 Publish pypi

```
python setup.py publish
```

CHAPTER

FOUR

FIXTURES

4.1 ngrok

An wrapper for run `ngrok` to port

4.1.1 Example:

Use as callable

```
import pytest

from urllib.request import urlopen

def test_ngrok(ngrok, httpserver):
    httpserver.expect_request("/foobar").respond_with_data("ok")
    remote_url = ngrok(httpserver.port)
    assert urlopen(remote_url + "/foobar").read() == b'ok'
```

Use as context manager

```
import pytest

from urllib.error import HTTPError
from urllib.request import urlopen

def test_ngrok_context_manager(ngrok, httpserver):
    # example local server
    httpserver.expect_request("/foobar").respond_with_data("ok")
    with ngrok(httpserver.port) as remote_url:
        _test_url = str(remote_url) + '/foobar'
        assert urlopen(_test_url).read() == b'ok'

    # Connection closes
    pytest.raises(HTTPError, urlopen, _test_url)
```

4.2 live_server_ngrok

If installed `pytest-django` - you can use `live_server_ngrok`

4.2.1 Example

```
def test_server(live_server_ngrok):
    assert live_server_ngrok.url.endswith('ngrok.io')
```

4.3 ngrok_bin

Trying find `ngrok` binary in PATH or set default to `$HOME/.local/bin/ngrok` Also use `--ngrok-bin=/path/to/ngrok`

4.3.1 Example

```
from pytest import fixture

@fixture(scope='session')
def ngrok_bin(request):
    ngrok_path = request.config.getoption('--ngrok-bin')
    if not ngrok_path:
        ngrok_path = '/tmp/ngrok'
    return ngrok_path
```

4.4 ngrok_install_url

Url for fetch binary if `ngrok_bin` not exists. Default is `https://bin.equinox.io/c/4VmDzA7iaHb/ngrok-stable-linux-amd64.zip`

4.4.1 Example:

```
from pytest import fixture

@fixture(scope='session')
def ngrok_install_url():
    return 'https://bin.equinox.io/c/4VmDzA7iaHb/ngrok-stable-linux-amd64.zip'
```

4.5 ngrok_allow_install

Allow to fetch and install before test. Default is `true`. You can set to `false` for forbid download binaries but m Also use `--ngrok-no-install`

4.5.1 Example

```
from pytest import fixture

@fixture(scope='session')
def ngrok_allow_install(request):
    return true
```

CHAPTER

FIVE

INDICES AND TABLES

- genindex
- modindex
- search