
cOXtactoe Documentation

Release 0.1.0

Brett Anderson

July 13, 2014

1	Contents	1
1.1	Pre-requisites	1
1.2	Installation & Configuration	1
1.3	0. Set installation/path options	1
1.4	1. Install virtualenv, pyenv, and pyenv-virtualenv	1
1.5	2. Install Python-2.7 with pyenv	2
1.6	3. Create and activate pyenv virtualenv	2
1.7	4. Clone this repo where you keep your project source code	2
1.8	5. Customize settings (optional)	2
1.9	6. Add project to python library path	3
1.10	7. Set the <i>DJANGO_SETTINGS_MODULE</i> environment var	3
1.11	8. Install the basic project requirements	3
1.12	9. syncdb && migrate	3
1.13	Testing	3
1.14	Running	4

1.1 Pre-requisites

- Bash
- Git
- SQLite3

1.2 Installation & Configuration

1.3 0. Set installation/path options

```
$ INSTALL_DIR="$HOME/code"  
$ TTT_PROJ="tictac"
```

1.4 1. Install virtualenv, pyenv, and pyenv-virtualenv

- pyenv: <https://github.com/yyuu/pyenv>
- pyenv-installer: <https://github.com/yyuu/pyenv-installer>
- pyenv-virtualenv: <https://github.com/yyuu/pyenv-virtualenv>

1.4.1 NOTE

If you already have Python 2.7 and virtualenv, you may simply create a new virtual env and skip the pyenv/pyenv-virtualenv setup. If you do skip the pyenv installation make sure to use the appropriate virtualenv path in steps #6 & #7.

1.4.2 Installation

You may use Homebrew if you have it or alternatively, use the installer.

1.4.3 Homebrew

```
$ brew install pyenv pyenv-virtualenv
```

1.4.4 Manual Install

```
$ cd $INSTALL_DIR
$ git clone https://github.com/yyuu/pyenv-installer.git
$ bash ./pyenv-installer/bin/pyenv-installer
$ echo 'export PATH="$HOME/.pyenv/bin:$PATH"' >> ~/.bashrc
$ echo 'eval "$(pyenv init -)"' >> ~/.bashrc
$ echo 'eval "$(pyenv virtualenv-init -)"' >> ~/.bashrc
$ exec "$SHELL"
```

1.5 2. Install Python-2.7 with pyenv

OS X

Use Homebrew or MacPorts if the build process complains of a missing library.

Ubuntu

```
$ sudo apt-get libreadline-dev libbz2-dev zlib1g-dev libsqlite3-dev libssl-dev
```

On Both

```
$ pyenv install 2.7.7
Installing Python-2.7.7...
Installed Python-2.7.7 to /Users/username/.pyenv/versions/2.7.7
$ pyenv rehash
```

1.6 3. Create and activate pyenv virtualenv

```
$ pyenv virtualenv 2.7.7 $TTT_PROJ
$ pyenv activate $TTT_PROJ
```

1.7 4. Clone this repo where you keep your project source code

```
$ cd $INSTALL_DIR
$ git clone https://github.com/bzdzb/Tic-Tac-Toe.git $TTT_PROJ
$ cd $TTT_PROJ
```

1.8 5. Customize settings (optional)

If you would like to update your `TIME_ZONE` or any other options, the settings files are located in `$TTT_PROJ/tictac/settings/`. They are organized into `base.py`, `dev.py`, `test.py`, `stage.py`, and `prod.py`. Only `base.py` and/or `dev.py` are likely to be of interest at this time.

Settings

- Base: `$TTT_PROJ/tictac/settings/base.py` Core configuration. Shared by other settings files.
- Development: `$TTT_PROJ/tictac/settings/dev.py` Development specific settings. E.g. `DEBUG=True`, `django_toolbar`, etc.
- Testing: `$TTT_PROJ/tictac/settings/test.py` Settings specific to running tests.
Set `SELENIUM_WEBDRIVER = "chrome"` if you have `chromedriver` installed and in your `PATH`.
- Stage: `$TTT_PROJ/tictac/settings/stage.py` Settings for stage deployment.
- Production: `$TTT_PROJ/tictac/settings/prod.py` Settings for production deployment. E.g. `DEBUG=False`, `production DB`, etc.

1.9 6. Add project to python library path

```
$ cd $INSTALL_DIR/$TTT_PROJ/tictac
$ echo `pwd` > ~/.pyenv/versions/${TTT_PROJ}/lib/python2.7/site-packages/tictac.pth
```

1.10 7. Set the `DJANGO_SETTINGS_MODULE` environment var

```
$ export DJANGO_SETTINGS_MODULE=tictac.settings.dev
$ echo "!!" >> ~/.pyenv/versions/${TTT_PROJ}/bin/activate
$ exec "$SHELL"
$ pyenv activate $TTT_PROJ
```

1.11 8. Install the basic project requirements

```
$ cd $INSTALL_DIR/$TTT_PROJ
$ pip install -r requirements/bin.txt
$ pip install -r requirements/dev.txt
```

As you edit your `'requirements.txt'` files, you can run those last commands again; `'pip'` will realise which packages you've added and will ignore those already installed.

1.12 9. syncdb && migrate

```
$ cd tictac
$ ./manage.py syncdb && ./manage.py migrate
```

1.13 Testing

```
pyenv activate tictac # or source your activate script if using virtualenv
cd $INSTALL_DIR/$TTT_PROJ/tictac
./run_tests.sh
```

1.14 Running

```
pyenv activate tictac # or source your activate script if using virtualenv
cd $INSTALL_DIR/$TTT_PROJ/tictac
./run.sh
```