

Orange County Lettings

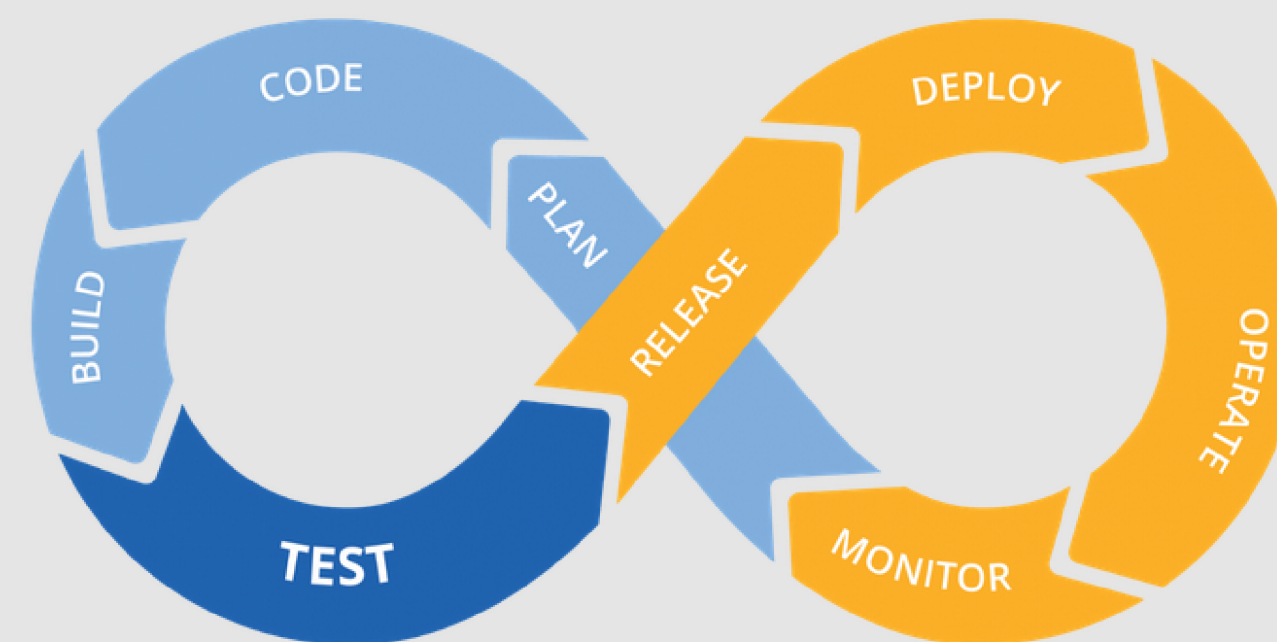


Emmanuel Letremble



Objectifs

- Mettre à jours le site actuel pour réduire la dette technique
- Mettre en place un système d'Intégration Continue et de Déploiement Continu (CI / CD)



Les contraintes

- ◉ réorganiser l'app sans perdre les données
- ◉ ne rien ajouter pour le moment, juste refactorer donc conserver Django et SQLite
- ◉ écrire des tests pytest pour vérifier les routes
- ◉ veiller au respect de la PEP 8 avec Flake8
- ◉ publier le code sur GitHub
- ◉ stocker les conteneurs sur Docker Hub
- ◉ déployer le site sur Heroku
- ◉ surveiller l'application avec Sentry
- ◉ centraliser le processus CI/CD sur CircleCI

Refactoring & tests

Code & Structure

Tests

```
>>> pytest
>>> python manage.py test . apps
```

Coverage

```
>>> coverage run
      --source='apps'
manage.py test . apps
>>> coverage report
```

```
apps
├── lettings
│   ├── admin.py
│   ├── apps.py
│   ├── __init__.py
│   ├── models.py
│   └── templates
│       ├── lettings
│       │   ├── index.html
│       │   └── letting.html
│       ├── tests.py
│       ├── urls.py
│       └── views.py
├── profiles
│   ├── admin.py
│   ├── apps.py
│   ├── __init__.py
│   ├── models.py
│   └── templates
│       └── profiles
│           ├── index.html
│           └── profile.html
├── tests.py
├── urls.py
└── views.py
Dockerfile
manage.py
oc_lettings_site
├── apps.py
├── asgi.py
├── __init__.py
├── settings.py
├── tests.py
├── urls.py
├── views.py
├── wsgi.py
├── oc-lettings-site.sqlite3
├── README.md
├── requirements.txt
├── setup.cfg
├── static
├── templates
│   └── index.html
```


Démo en local



◎ Directement depuis les fichiers sources

```
>>> python manage.py runserver  
>>> gunicorn oc_lettings_site.wsgi
```



◎ Depuis un conteneur Docker crée localement

```
>>> docker build -t oc-lettings .  
>>> docker run -d -e SENTRY_DSN -p 80:8000 oc-lettings
```

◎ Depuis un conteneur stocké sur Docker-Hub

```
>>> docker run -d -e SENTRY_DSN -p 80:8000  
--pull always valkea/oc-lettings
```


Démo sur Heroku



1. présentation du site sur
<https://oc-lettings.herokuapp.com>

2. suppression de l'application sur
<https://dashboard.heroku.com/apps/oc-lettings>

Pipeline CI/CD



1. commit du code sur **GitHub**
2. lancement des process **CircleCI**
3. envoi et stockage sur **Docker-Hub**
4. envoi et déploiement sur **Heroku**
5. surveillance de l'app avec **Sentry**
(<https://oc-lettings.herokuapp.com/sentry-debug>)



Bonnes pratiques



- **docstrings** (fake pour cette fois)
- **PEP8** (black & vim-flake8 & flake8)
- **README.md**
- **Tests** (pytest)
- **Coverage** (94%)

Et après ?



1. passer de **SQLite** à **PostgreSQL** afin de permettre la persistance des données sur **Heroku**
2. améliorer le **config.py** de **CircleCI** pour réduire le temps de traitement du cycle **CI / CD**
3. optimiser la taille des images **Docker** (au delà du simple **.dockerignore**) pour réduire les temps d'envoi et de collecte

Merci

de m'avoir écouté,
évalué et conseillé

