



facultade de
informática
da coruña

FACULTADE DE INFORMÁTICA

**Práctica de AISI.
Clústering y Docker**

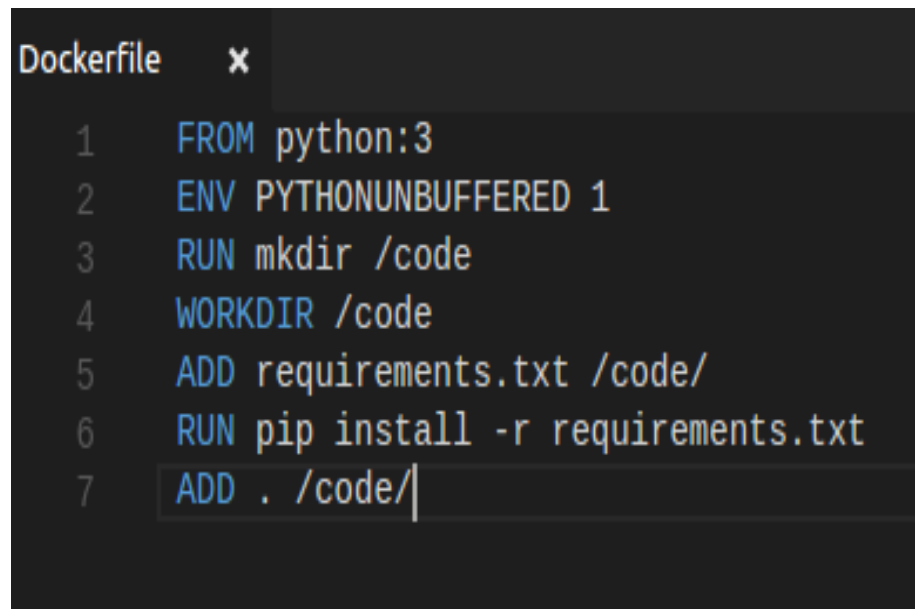
Capítulo 1

Proyecto personal con Docker

A intención e crear un contenedor donde se ejecute un proyecto personal dunha páxina desenvolada con django.

Para empezar temos que crear un Dockerfile, un arquivo coas dependencias de Python e un arquivo docker-compose.yml

O arquivo do Dockerfile terá a seguinte configuración:

A screenshot of a code editor showing a Dockerfile. The editor has a dark background with light-colored text. The title bar of the editor shows 'Dockerfile' and a close button 'x'. The code is as follows:

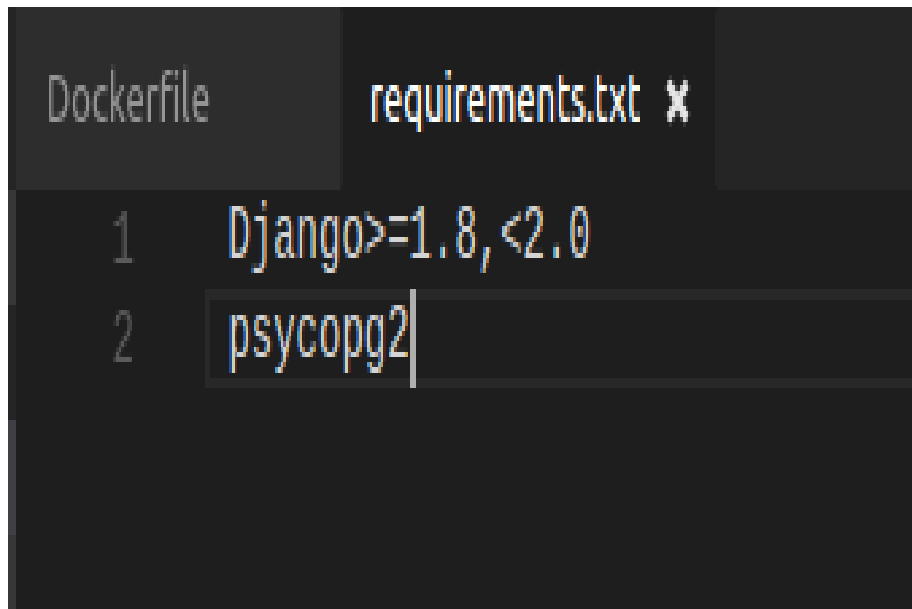
```
1 FROM python:3
2 ENV PYTHONUNBUFFERED 1
3 RUN mkdir /code
4 WORKDIR /code
5 ADD requirements.txt /code/
6 RUN pip install -r requirements.txt
7 ADD . /code/
```

Figura 1.1: Dockerfile



As dependencias de Python a instalar inclúranse nun ficheiro chamado requirements.txt, este arquivo executarase co comando RUN pip install -r requirements.txt que se encontra no Dockerfile.

O arquivo requirements.txt será o seguinte:



```
Dockerfile requirements.txt X
1 Django>=1.8,<2.0
2 psycopg2
```

Figura 1.2: requirements.txt

A continuación crease o contenedor, utilízase a seguinte orde:

docker build -t proyectopagina .



CAPÍTULO 1. PROYECTO PERSONAL CON DOCKER

```
pablito@pablitoBar:~/uni/aisi/practica4/dockerpersonal$ docker build -t proyectopagina .
Sending build context to Docker daemon  51.2kB
Step 1/7 : FROM python:3
3: Pulling from library/python
cd0a524342ef: Already exists
e39c3ffe4133: Already exists
85334a7c2001: Already exists
4c546d9d6a84: Already exists
a2eb12d55dae: Already exists
c315b5d973a6: Pull complete
7ada7ee91ec1: Pull complete
1954cbaa0cd4: Pull complete
Digest: sha256:ff4ead5fb37ce95c8468e1ca728a29e9db607cd19f1226d555fcl1a0d38cd38ea
Status: Downloaded newer image for python:3
--> 21289e3715bd
Step 2/7 : ENV PYTHONUNBUFFERED 1
--> Running in b5e8203a04a6
--> 07f9e6af10
Removing intermediate container b5e8203a04a6
Step 3/7 : RUN mkdir /code
--> Running in 6ae126730d3c
--> 1d2b4bfc4f08
Removing intermediate container 6ae126730d3c
Step 4/7 : WORKDIR /code
--> 337b9d1b6b70
Removing intermediate container fe9ba7f47ca5
Step 5/7 : ADD requirements.txt /code/
--> 64cc9a89552e
Removing intermediate container 84b9e50d9665
Step 6/7 : RUN pip install -r requirements.txt
--> Running in 20cc9f52c418
Collecting Django<2.0,>=1.8 (from -r requirements.txt (line 1))
  Downloading Django-1.11.1-py2.py3-none-any.whl (6.9MB)
Collecting psycopg2 (from -r requirements.txt (line 2))
  Downloading psycopg2-2.7.1-cp36-cp36m-manylinux1 x86_64.whl (2.7MB)
Collecting pytz (from Django<2.0,>=1.8->-r requirements.txt (line 1))
  Downloading pytz-2017.2-py2.py3-none-any.whl (484kB)
Installing collected packages: pytz, Django, psycopg2
Successfully installed Django-1.11.1 psycopg2-2.7.1 pytz-2017.2
--> 396ca95a5439
Removing intermediate container 20cc9f52c418
Step 7/7 : ADD . /code/
--> 5ea67473bf70
Removing intermediate container b39781b12ee7
Successfully built 5ea67473bf70
```

Figura 1.3: requirements.txt

Tamen crearse o docker-compose.yml, a configuración será a seguinte:

```
Dockerfile  requirements.txt  docker-compose.yml x
1  version: '2'
2  services:
3    web:
4      build: .
5      command: python3 manage.py runserver 0.0.0.0:8000
6      volumes:
7        - ./code
8      ports:
9        - "8000:8000"
```

Figura 1.4: docker-compose.yml



Configurase un servicio web.

A continuación copíase os arquivos do proxecto no directorio, cando se execute o docker-compose iniciárase o servidore na dirección 0.0.0.0:8000

1.1. Referencias:

<https://github.com/mmorejon/docker-django>

<https://docs.docker.com/compose/django/# connect-the-database>