Alberto Bolta Sanmateu
1º DAW

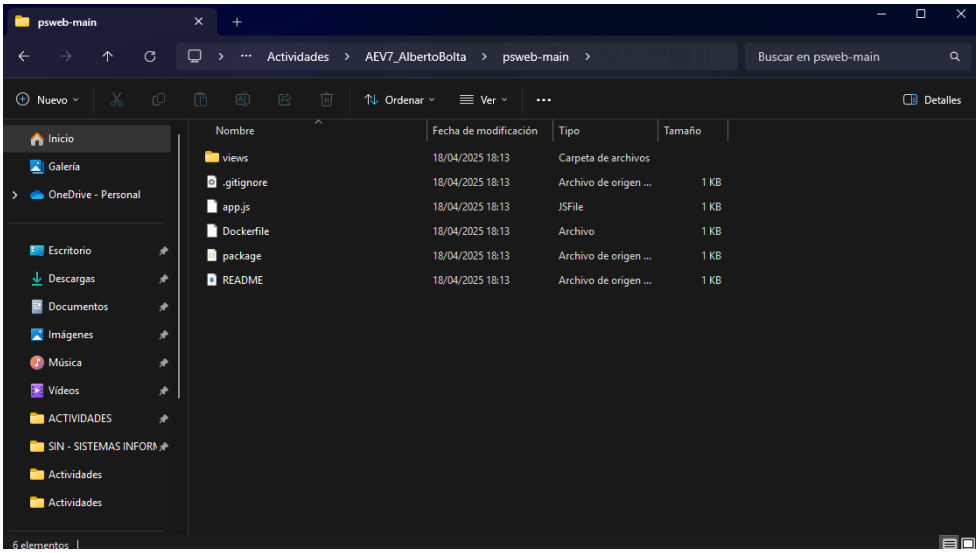
AEV7 – EDE

1.- **Obtención de la aplicación. (1 punto)**

Vamos a basar la ejecución de esta actividad en una pequeña demo, propuesta por un especialista en Cloud Computing, particularmente en entornos de contenedores. Su nombre es Nigel Poulton, ha escrito varios libros sobre el tema y es instructor de cursos online de la plataforma "Pluralsight", además de haber participado de forma activa en el desarrollo y difusión global de este tipo de tecnologías.

Usaremos como base, los ficheros del repositorio psweb de la cuenta NigelPoulton en GitHub, es decir: https://github.com/nigelpoulton/psweb

Accede al repositorio y descarga el contenido en una carpeta local. También podríamos clonarlo, pero como posteriormente no vamos a integrar nuestros cambios en el repositorio, con descargar los ficheros es suficiente en este caso.



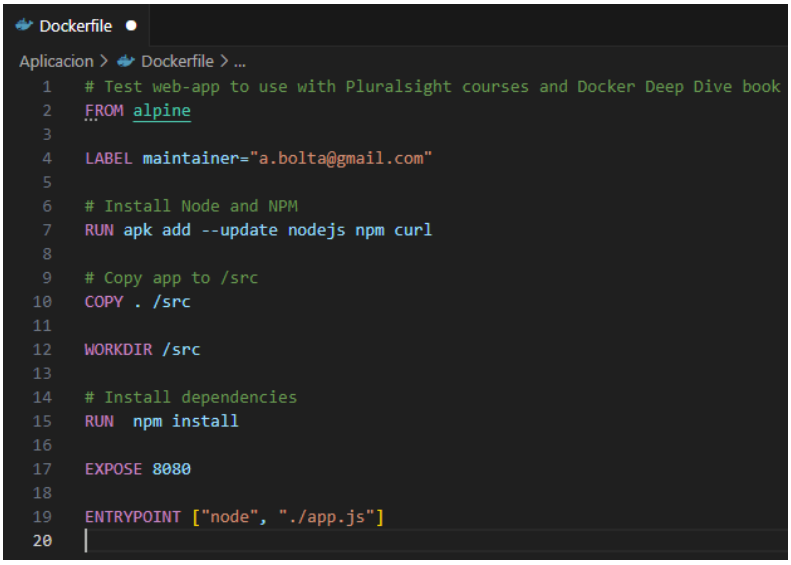
2.- **Personalización de la aplicación. (1 punto)**

Una vez descomprimida la descarga, hacemos una copia de trabajo del contenido completo de "psweb-master" en otra carpeta local. Observamos que disponemos de un conjunto de directorios y ficheros que compondrán una pequeña aplicación. Será la aplicación que integraremos en un contenedor.

En el directorio raíz de la aplicación dispondremos, entre otros, de un fichero Dockerfile. Para poder visualizar el contenido del fichero, puedes usar el editor "Visual Studio Code". El fichero Dockerfile es un poco especial, puesto que no tiene extensión.

En este caso, el fichero Dockerfile nos viene dado (se explica su estructura en el recurso didáctico correspondiente, en el curso de Florida Oberta).

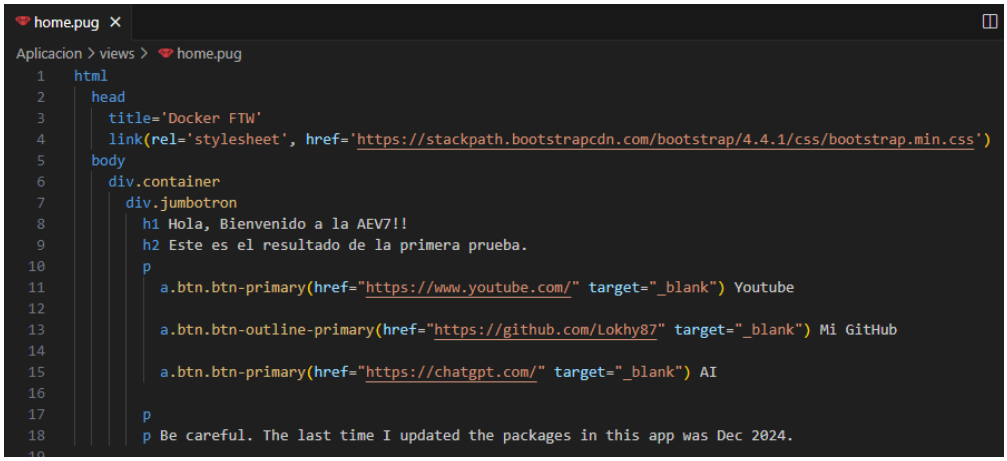
Modifica el fichero Dockerfile. Concretamente, debes indicar tu correo electrónico de contacto en la etiqueta **LABEL maintener**.



Respecto al resto de los ficheros descargados en nuestra carpeta local y que formarán nuestra aplicación web, accede al fichero "**home.pug**" de la carpeta "**views**". La aplicación es un desarrollo del tipo "hola mundo", que muestra una sencilla página como resultado. Este fichero define el contenido de la página principal que se mostrará cuando arranque el contenedor.

Edita el fichero, puedes hacerlo también con "Visual Studio Code", y personalízalo. Modifica los textos y los enlaces, poniendo unos personalizados por ti, los que tú quieras. Esto hará que tu contenedor sea único.

Llegados a este punto y habiendo guardado los cambios, ya tenemos preparada nuestra aplicación personalizada, para poder "hacerle una foto" y meterla en un contenedor.



Alberto Bolta Sanmateu
1º DAW

3.- **Generación de la imagen. (2 puntos)**

Genera una imagen Docker que contenga nuestra aplicación, a partir de las especificaciones descritas en el fichero Dockerfile. Confirma mediante la línea de comandos que la imagen se ha generado (opcionalmente, también mediante Docker Desktop).

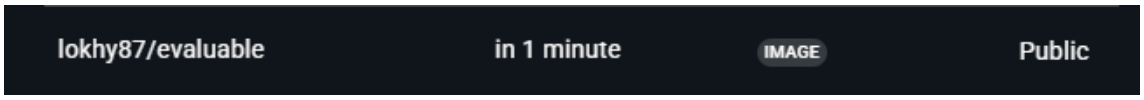
```
PS C:\Users\abolt\Documents\GitHub\DAW.FLORIDA\EDE - ENTORNOS DE DESARROLLO\Actividades\AEV7_AlbertoBolta\Aplicacion
> docker image build -t lokhy87/evaluable:v0 .
[+] Building 13.5s (11/11) FINISHED                                    docker:desktop-linux
 => [internal] load build definition from Dockerfile                                  0.1s
 => => transferring dockerfile: 356B                                                  0.1s
 => [internal] load metadata for docker.io/library/alpine:latest                      2.6s
 => [auth] library/alpine:pull token for registry-1.docker.io                         0.0s
 => [internal] load .dockerignore                                                     0.1s
 => => transferring context: 2B                                                       0.0s
 => [1/5] FROM docker.io/library/alpine:latest@sha256:a8560b36e8b8210634f77d9f7f9efd7ffa463e380  0.8s
 => => resolve docker.io/library/alpine:latest@sha256:a8560b36e8b8210634f77d9f7f9efd7ffa463e380  0.0s
 => => sha256:a8560b36e8b8210634f77d9f7f9efd7ffa463e380b75e2e74aff4511df3ef88c 9.22kB / 9.22kB  0.0s
 => => sha256:1c4eef651f65e2f7daee7ee785882ac164b02b78fb74503052a26dc061c90474 1.02kB / 1.02kB  0.0s
 => => sha256:aded1e1a5b3705116fa0a92ba074a5e0b00316f7d9c315983ccba2ee5428ec8b 581B / 581B      0.0s
 => => sha256:f18232174bc91741fdf3da96d85011092101a032a93a388b79e99e69c2d5c870 3.64MB / 3.64MB  0.4s
 => => extracting sha256:f18232174bc91741fdf3da96d85011092101a032a93a388b79e99e69c2d5c870       0.2s
 => [internal] load build context                                                     0.1s
 => => transferring context: 2.29kB                                                   0.0s
 => [2/5] RUN apk add --update nodejs npm curl                                        3.3s
 => [3/5] COPY . /src                                                                 0.1s
 => [4/5] WORKDIR /src                                                                0.1s
 => [5/5] RUN  npm install                                                            5.8s
 => exporting to image                                                                0.5s
 => => exporting layers                                                               0.5s
 => => writing image sha256:dba9b498eda25d4739850c790c2018d7c19197848fa72fc4f37d06bd5caecbb4  0.0s
 => => naming to docker.io/lokhy87/evaluable:v0                                       0.0s

What's next:
    View a summary of image vulnerabilities and recommendations → docker scout quickview
PS C:\Users\abol                                S DE DESARROLLO\Actividades\AEV7_AlbertoBolta\Aplicacion
PS C:\Users\abolt\Documents\GitHub\DAW.FLORIDA\EDE - ENTORNOS DE DESARROLLO\Actividades\AEV7_AlbertoBolta\Aplicacion
> docker image ls
REPOSITORY                          TAG       IMAGE ID       CREATED          SIZE
lokhy87/evaluable                   v0        dba9b498eda2   49 seconds ago   107MB
baseprogramacionxdebug1daw-main-web   latest    c13f955a593b   3 months ago     504MB
```

4.- **Integración de la imagen en Docker Hub. (1 puntos)**

Guarda la imagen generada en Docker Hub, mediante el uso de comandos. Confirma que existe en el repositorio.

```
PS C:\Users\abolt\Documents\GitHub\DAW.FLORIDA\EDE - ENTORNOS DE DESARROLLO\Actividades\AEV7_AlbertoBolta\Aplicacion
> docker image push lokhy87/evaluable:v0
The push refers to repository [docker.io/lokhy87/evaluable]
ca304e2ccb01: Pushed
5f70bf18a086: Mounted from library/php
b5666e0bec18: Pushed
60eb09b128f6: Pushed
08000c18d16d: Mounted from library/alpine
v0: digest: sha256:853b3c031ead224ab1c399d962809995a06bc8b2668ebb9951a7b46b29d92062 size: 1365
```

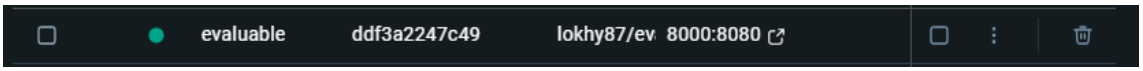


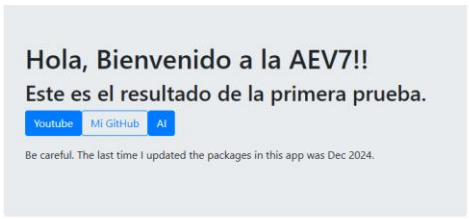
5.- **Ejecución de un contenedor en base a la imagen generada. (3 puntos)**

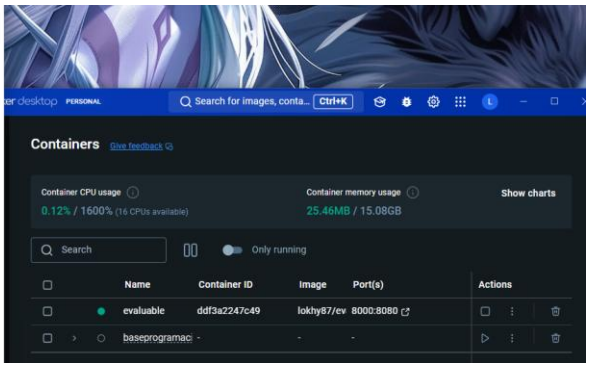
Crea un contenedor basado en la imagen generada. El contenedor escuchará las peticiones por el puerto 8080, que a su vez hayan entrado previamente a través del host anfitrión (máquina física) mediante el puerto 8000. Confirma que se muestra la página web correspondiente. Detén el contenedor, confirma que ahora no se muestra la web y posteriormente vuelve a arrancarlo. Todo mediante el uso de

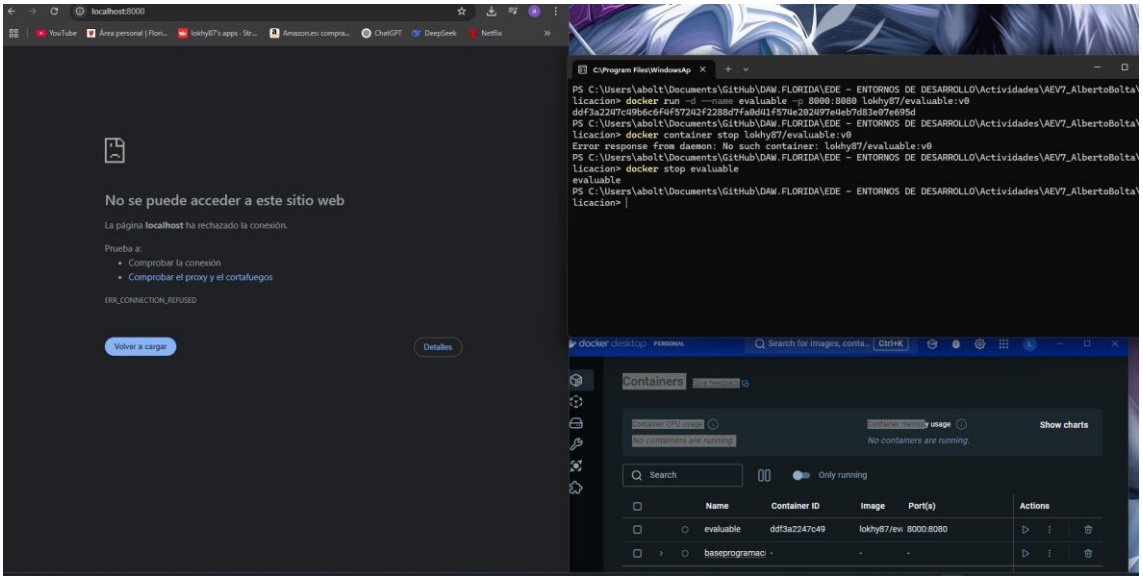
comandos (opcional y adicionalmente, puedes hacerlo también mediante Docker Desktop).

```
PS C:\Users\abolt\Documents\GitHub\DAW.FLORIDA\EDE - ENTORNOS DE DESARROLLO\Actividades\AEV7_AlbertoBolta\Aplicacion
> docker run -d --name evaluable -p 8000:8080 lokhy87/evaluable:v0
ddf3a2247c49b6c6f4f57242f2288d7fa0d41f574e202497e4eb7d83e07e695d
```









6.- **Ejecución de un contenedor con volumen en base a la imagen generada. (2 puntos)**

Crea un nuevo contenedor basado en la imagen generada, pero en este caso con un volumen. De forma que queden mapeadas las carpetas denominadas “views” del contenedor y del sistema de archivos del anfitrión o máquina física. Confirma que si modificas el fichero “home.pug” desde el anfitrión, éste se modifica en el contenedor y, en consecuencia, los cambios se verán reflejados en la página web correspondiente. Confirma también que, si lo modificas desde el contenedor, se modifica en el sistema de archivos del anfitrión.

```
PS C:\Users\abolt\Documents\GitHub\DAW.FLORIDA\EDE - ENTORNOS DE DESARROLLO\Actividades\AEV7_AlbertoBolta\Aplicacion> docker run -d --n
ame volumen-evaluable -p 8000:8080 -v "/C/Users/abolt/Documents/GitHub/DAW.FLORIDA/EDE - ENTORNOS DE DESARROLLO/Actividades/AEV7_Albert
oBolta/Aplicacion/views:/src/views" lokhy87/evaluable:v0
b22d54edc60d22b5ead96984827c4b99118f78c769c183c85025142f62b85edc
```