Ingeniera de Software 3

TP 2: Introduccion a Docker

4.1: Instalar docker

4.2: Instalamos DockerHub
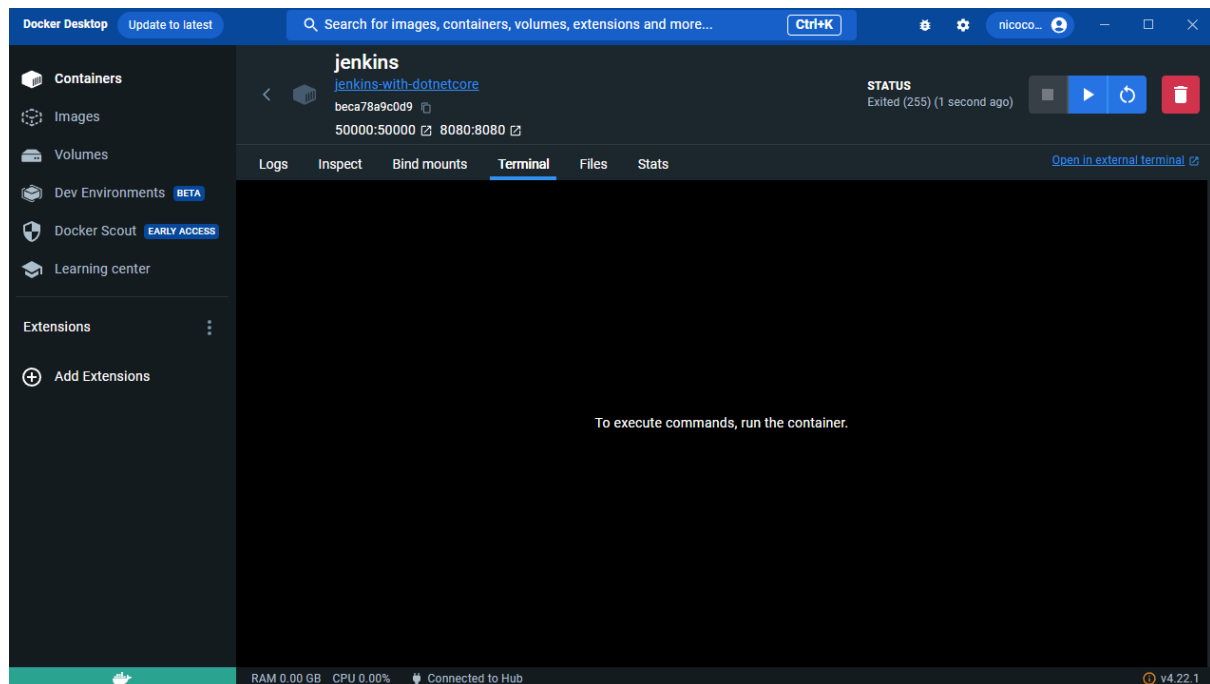
4.3: Obtenemos la imagen busybox

```
C:\Users\Nico>docker images
REPOSITORY    TAG       IMAGE ID       CREATED       SIZE
busybox       latest    a416a98b71e2   6 weeks ago   4.26MB

C:\Users\Nico>docker run busybox
```

4.4: Ejecutamos:

```
C:\Users\Nico>docker run busybox
```

No obtenemos ningun resultado ya que el contenedor esta vacio.

Hacemos un echo con "Hola Mundo" y ahora nos muestra un resultado.

```
C:\Users\Nico>docker run busybox echo "Hola Mundo"
Hola Mundo
```

```
C:\Users\Nico>docker ps
CONTAINER ID    IMAGE      COMMAND    CREATED    STATUS      PORTS      NAMES
```

```
C:\Users\Nico>docker ps -a
CONTAINER ID   IMAGE     COMMAND            CREATED          STATUS                     PORTS     NAMES
ad53d99083f0   busybox   "echo 'Hola Mundo'"  18 seconds ago   Exited (0) 17 seconds ago            elastic_murdock
e2269d7266a8   busybox   "sh"               43 seconds ago   Exited (0) 43 seconds ago            frosty_burnell
```

Estos son todos los contenedores ejecutados.

4.5: Modo iterativo:

```
C:\Users\Nico>docker run -it busybox sh
/ # ps
PID   USER      TIME  COMMAND
    1 root      0:00 sh
    7 root      0:00 ps
/ #
```

```
    7 root      0:00 ps
/ # uptime
 18:08:23 up 9 min,  0 users,  load average: 0.00, 0.02, 0.00
/ #
```

```
            up 9 min,   0 users,   load average: 0.00, 0.02, 0.00
/ # free
              total        used        free      shared  buff/cache   available
Mem:        7830824      907508     6552424        2576      370892     6691952
Swap:       2097152           0     2097152
```

```
/ # ls -l /
total 40
drwxr-xr-x     2 root      root       12288 Jul 17 18:30 bin
drwxr-xr-x     5 root      root         360 Sep  5 18:07 dev
drwxr-xr-x     1 root      root        4096 Sep  5 18:07 etc
drwxr-xr-x     2 nobody    nobody      4096 Jul 17 18:30 home
drwxr-xr-x     2 root      root        4096 Jul 17 18:30 lib
lrwxrwxrwx     1 root      root           3 Jul 17 18:30 lib64 -> lib
dr-xr-xr-x   311 root      root           0 Sep  5 18:07 proc
drwx------     1 root      root        4096 Sep  5 18:07 root
dr-xr-xr-x    11 root      root           0 Sep  5 18:07 sys
drwxrwxrwt     2 root      root        4096 Jul 17 18:30 tmp
drwxr-xr-x     4 root      root        4096 Jul 17 18:30 usr
drwxr-xr-x     4 root      root        4096 Jul 17 18:30 var
/ #
```

4.6: Borramos los contenedores terminados

```
C:\Users\Nico>docker ps -a
CONTAINER ID   IMAGE     COMMAND              CREATED         STATUS                    PORTS     NAMES
c4725da490af   busybox   "sh"                 2 minutes ago   Exited (0) 17 seconds ago           stupefied_euler
ad53d99083f0   busybox   "echo 'Hola Mundo'"  8 minutes ago   Exited (0) 8 minutes ago            elastic_murdock
e2269d7266a8   busybox   "sh"                 9 minutes ago   Exited (0) 9 minutes ago            frosty_burnell
```

```
C:\Users\Nico>docker rm stupefied_euler
stupefied_euler
```

```
C:\Users\Nico>docker container prune
WARNING! This will remove all stopped containers.
Are you sure you want to continue? [y/N] y
Deleted Containers:
ad53d99083f0ad1ceae8a897399011d285199e0b421c9cbb34ac8c86cc00cc4d
e2269d7266a82e0ee5de1bc62d80b80bfc086bbf316b9927d31fb1161c23fde8

Total reclaimed space: 0B
```

7.1: Construimos una imagen a partir de un codigo en Github.

```
Nico@DESKTOP-MDI8QUJ MINGW64 ~/Documents/Ing de Soft 3
$ git clone https://github.com/ingsoft3ucc/SimpleWebAPI
Cloning into 'SimpleWebAPI'...
remote: Enumerating objects: 43, done.
remote: Counting objects: 100% (43/43), done.
remote: Compressing objects: 100% (30/30), done.
remote: Total 43 (delta 17), reused 35 (delta 11), pack-reused 0
Receiving objects: 100% (43/43), 9.47 KiB | 4.74 MiB/s, done.
Resolving deltas: 100% (17/17), done.
```

```
PS C:\Users\Nico\Documents\Ing de Soft 3\SimpleWebAPI> docker build -t mywebapi .
[+] Building 290.4s (18/18) FINISHED                                                       docker:default
 => [internal] load build definition from Dockerfile                                                0.0s
 => => transferring dockerfile: 810B                                                                0.0s
 => [internal] load .dockerignore                                                                   0.0s
 => => transferring context: 2B                                                                     0.0s
 => [internal] load metadata for mcr.microsoft.com/dotnet/sdk:7.0                                   1.0s
 => [internal] load metadata for mcr.microsoft.com/dotnet/aspnet:7.0                                0.9s
 => [base 1/2] FROM mcr.microsoft.com/dotnet/aspnet:7.0@sha256:54a3864f1c7dbb232982f61105aa18a59b976382a4e720fe1  93.9s
 => => resolve mcr.microsoft.com/dotnet/aspnet:7.0@sha256:54a3864f1c7dbb232982f61105aa18a59b976382a4e720fe18b4af9  0.0s
 => => sha256:54a3864f1c7dbb232982f61105aa18a59b976382a4e720fe18b4af98fcd663a6 1.82kB / 1.82kB      0.0s
 => => sha256:6261b1c7c11786341b6fa46f8fee00f52643a2a799f989eaf406064163b015be 1.37kB / 1.37kB      0.0s
 => => sha256:63837b17b8fbc4a19c269fbf17abdb0cabb0205c352f115510b316cecb794cba 3.26kB / 3.26kB      0.0s
 => => sha256:14726c8f78342865030f97a8d3492e2d1a68fbd22778f9a31dc6be4b4f12a9bc 31.42MB / 31.42MB   65.3s
 => => sha256:6c3981608c2b6b3825fe63b9c920101d7bf4ca80ceda7ddc0dc09fdc93a5797d 14.97MB / 14.97MB   39.3s
 => => sha256:a2e354eccee4ad7336b4692c2662de61a68677cec551530c6d7045bcbb3cd6d7 32.45MB / 32.45MB   92.9s
 => => sha256:cbca4ad4689cc576651411faa2b98dff307dc8f6de9adf08a5a2d6ae2239d233 154B / 154B         39.7s
 => => sha256:31e2a952779c41cf5cd187ae1ae276f3de3965a43d41a0250b56c58e1206db8a 10.12MB / 10.12MB   67.8s
 => => extracting sha256:14726c8f78342865030f97a8d3492e2d1a68fbd22778f9a31dc6be4b4f12a9bc          224.0s
 => => extracting sha256:6c3981608c2b6b3825fe63b9c920101d7bf4ca80ceda7ddc0dc09fdc93a5797d          222.0s
 => => extracting sha256:a2e354eccee4ad7336b4692c2662de61a68677cec551530c6d7045bcbb3cd6d7          196.4s
 => => extracting sha256:cbca4ad4689cc576651411faa2b98dff307dc8f6de9adf08a5a2d6ae2239d233          195.8s
 => => extracting sha256:31e2a952779c41cf5cd187ae1ae276f3de3965a43d41a0250b56c58e1206db8a            0.2s
 => [build 1/7] FROM mcr.microsoft.com/dotnet/sdk:7.0@sha256:926b9337622ccc594ed051d3559f1c4544686c9fd5f4656f09  267.4s
 => => resolve mcr.microsoft.com/dotnet/sdk:7.0@sha256:926b9337622ccc594ed051d3559f1c4544686c9fd5f4656f09d1fe37d9  0.0s
 => => sha256:93b7191fd296e4befb35c4440ab640b7322c0536f5aa9331229b77ac955b7c19 7.17kB / 7.17kB      0.0s
 => => sha256:09b55190ab5c2a00d676678bdd1d702afc1aed8b505681fb4bd5979c62971e4a 2.01kB / 2.01kB      0.0s
 => => sha256:14726c8f78342865030f97a8d3492e2d1a68fbd22778f9a31dc6be4b4f12a9bc 31.42MB / 31.42MB   65.3s
 => => sha256:6c3981608c2b6b3825fe63b9c920101d7bf4ca80ceda7ddc0dc09fdc93a5797d 14.97MB / 14.97MB   39.3s
 => => sha256:a2e354eccee4ad7336b4692c2662de61a68677cec551530c6d7045bcbb3cd6d7 32.45MB / 32.45MB   92.9s
 => => sha256:926b9337622ccc594ed051d3559f1c4544686c9fd5f4656f09d1fe37d9a5ace2 1.82kB / 1.82kB      0.0s
 => => sha256:cbca4ad4689cc576651411faa2b98dff307dc8f6de9adf08a5a2d6ae2239d233 154B / 154B         39.7s
 => => sha256:31e2a952779c41cf5cd187ae1ae276f3de3965a43d41a0250b56c58e1206db8a 10.12MB / 10.12MB   67.8s
 => => extracting sha256:14726c8f78342865030f97a8d3492e2d1a68fbd22778f9a31dc6be4b4f12a9bc            2.0s
 => => sha256:6ecbf1fb37e17325876ded9c592f691f875695c9df5a16bd4c5ecd7fc2526533 25.37MB / 25.37MB  121.3s
 => => extracting sha256:6c3981608c2b6b3825fe63b9c920101d7bf4ca80ceda7ddc0dc09fdc93a5797d            0.4s
 => => sha256:6d8c48eeb4a8d2aa20e975cbaa5a6160df0baeb2acccebcc32a23a5e6ed6a9b2 181.16MB / 181.16MB 261.8s
 => => extracting sha256:a2e354eccee4ad7336b4692c2662de61a68677cec551530c6d7045bcbb3cd6d7            0.6s
 => => sha256:2e9754864261a7fb02d063e91161ddc95ad63e2e75e0ba8fb4d604b0ebfa8ba1 13.90MB / 13.90MB  125.5s
 => => extracting sha256:cbca4ad4689cc576651411faa2b98dff307dc8f6de9adf08a5a2d6ae2239d233            0.0s
 => => extracting sha256:31e2a952779c41cf5cd187ae1ae276f3de3965a43d41a0250b56c58e1206db8a            0.2s
 => => extracting sha256:6ecbf1fb37e17325876ded9c592f691f875695c9df5a16bd4c5ecd7fc2526533            1.2s
 => => extracting sha256:6d8c48eeb4a8d2aa20e975cbaa5a6160df0baeb2acccebcc32a23a5e6ed6a9b2            5.0s
 => => extracting sha256:2e9754864261a7fb02d063e91161ddc95ad63e2e75e0ba8fb4d604b0ebfa8ba1            0.4s
 => [internal] load build context                                                                  0.0s
 => => transferring context: 2.59kB                                                                 0.0s
 => [base 2/2] WORKDIR /app                                                                         0.1s
 => [final 1/2] WORKDIR /app                                                                        0.0s
 => [build 2/7] WORKDIR /src                                                                        0.4s
```

```
PS C:\Users\Nico\Documents\Ing de Soft 3\SimpleWebAPI> docker images
REPOSITORY      TAG          IMAGE ID        CREATED          SIZE
mywebapi        latest       b639d7e8b0b6    3 minutes ago    216MB
```

```
PS C:\Users\Nico\Documents\Ing de Soft 3\SimpleWebAPI> docker run mywebapi
info: Microsoft.Hosting.Lifetime[14]
      Now listening on: http://[::]:80
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Production
info: Microsoft.Hosting.Lifetime[0]
      Content root path: /app
info: Microsoft.Hosting.Lifetime[0]
      Application is shutting down...
PS C:\Users\Nico\Documents\Ing de Soft 3\SimpleWebAPI> docker run mywebapi
info: Microsoft.Hosting.Lifetime[14]
      Now listening on: http://[::]:80
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Production
info: Microsoft.Hosting.Lifetime[0]
      Content root path: /app
```

```
C: > Users > Nico > Documents > Ing de Soft 3 > SimpleWebAPI > 🐳 Dockerfile
1    #See https://aka.ms/containerfastmode to understand how Visual Studio uses this Dockerfile to build your images for faster debugging.
2
3    FROM mcr.microsoft.com/dotnet/aspnet:7.0 AS base
4    WORKDIR /app
5    EXPOSE 80
6    EXPOSE 443
7    EXPOSE 5254
8
9    FROM mcr.microsoft.com/dotnet/sdk:7.0 AS build
10   WORKDIR /src
11   COPY ["SimpleWebAPI/SimpleWebAPI.csproj", "SimpleWebAPI/"]
12   RUN dotnet restore "SimpleWebAPI/SimpleWebAPI.csproj"
13   COPY . .
14   WORKDIR "/src/SimpleWebAPI"
15   RUN dotnet build "SimpleWebAPI.csproj" -c Release -o /app/build
16
17   FROM build AS publish
18   RUN dotnet publish "SimpleWebAPI.csproj" -c Release -o /app/publish /p:UseAppHost=false
19
20   FROM base AS final
21   WORKDIR /app
22   COPY --from=publish /app/publish .
23   ENTRYPOINT ["dotnet", "SimpleWebAPI.dll"]
24   #CMD ["/bin/bash"]
25
26
27
```

Explicacion de cada linea:

FROM: Especifica la imagen base que se va a utilizar.

WORKDIR: Especifica el directorio.

EXPOSE: Especifica los puertos de escucha.

COPY: Para copiar archivos o directorios del sistema.

RUN: Ejecutar comandos durante la construccion.

ENTRYPOINT: Para configurar el comando principal que se ejecutara al iniciar el contenedor.

```
PS C:\Users\Nico\Documents\Ing de Soft 3\SimpleWebAPI> docker push nicocolman3/mywebapi1
Using default tag: latest
The push refers to repository [docker.io/nicocolman3/mywebapi1]
db6863be10b6: Pushed
5f70bf18a086: Pushed
43484fbeb766: Pushed
b260d977135d: Pushed
c65e011b79f6: Pushed
7af65d5c9e2a: Pushed
a5511d7cb706: Pushed
63290f9c9e52: Pushed
latest: digest: sha256:5f0bfcd79d60a3608aa3e11a99f645b6c0d88ccfdaf591242a72a635f29af87b size: 1996
PS C:\Users\Nico\Documents\Ing de Soft 3\SimpleWebAPI>
```

4.8: Ahora publicamos puertos para poder acceder desde el exterior

```
PS C:\Users\Nico\Documents\Ing de Soft 3\SimpleWebAPI> docker run --name myapi -d mywebapi
26515a60f5e674f1047b8d0418bf09cef1dea56d8e0a17530170da0568172f48
PS C:\Users\Nico\Documents\Ing de Soft 3\SimpleWebAPI> docker ps
CONTAINER ID   IMAGE      COMMAND             CREATED        STATUS         PORTS                       NAMES
26515a60f5e6   mywebapi   "dotnet SimpleWebAPI…"   33 seconds ago   Up 32 seconds   80/tcp, 443/tcp, 5254/tcp   myapi
PS C:\Users\Nico\Documents\Ing de Soft 3\SimpleWebAPI>
```

```
PS C:\Users\Nico\Documents\Ing de Soft 3\SimpleWebAPI> docker kill myapi
myapi
PS C:\Users\Nico\Documents\Ing de Soft 3\SimpleWebAPI> docker rm myapi
myapi
```

```
PS C:\Users\Nico\Documents\Ing de Soft 3\SimpleWebAPI> docker run --name myapi -d -p 80:80 -p 5254:5254 mywebapi
230ac03d9ea2485e357221f123d3688a93b106a230f7aa7b4300efb4fb30397f
```

```
← → C  ⓘ localhost/WeatherForecast
[{"date":"2023-09-06","temperatureC":44,"temperatureF":111,"summary":"Cool"},{"date":"2023-09-07","temperatureC":37,"temperatureF":98,"summary":"Balmy"},{"date":"2023-09-08","temperatureC":50,"temperatureF":121,"summary":"Cool"},{"date":"2023-09-09","temperatureC":44,"temperatureF":111,"summary":"Warm"},{"date":"2023-09-10","temperatureC":38,"temperatureF":100,"summary":"Warm"}]
```

4.9: Modificamos Dockerfile para que soporte bash

```
#ENTRYPOINT ["dotnet", "SimpleWebAPI.dll"]
CMD ["/bin/bash"]
```

Lo volvemos a correr y no se ejecuta.

Esta página no funciona

localhost no envió ningún dato.

ERR_EMPTY_RESPONSE

Volver a cargar