

## Fase 4 - Activitat 8.2: Docker Hub

### 0- Identificació del grup i activitat:

**Curs:** ASIX2

**Projecte:** PJ9 DevOps i Cloud Computing

**Fase:** 4

**Activitat:** 8.2

**Grup/Individual:** Individual

**Membres/Alumne:**

### 1- Introducció i objectius de l'activitat 8.2

- Creació d'un compte de Docker Hub.
- Compartint imatge de contenidor Docker via Docker Hub
- Accedint a imatges de contenidors Docker via Docker Hub.

### 2- Canviant el nom de la màquina virtual i el seu nom de sistema.

- Dins de la màquina física:
  - Accedeix a la carpeta **pj9** → **f4** → **a8** i canvia el nom de la carpeta **pj9f4a8.1** a **pj9f4a8.2**
  - A continuació, accedeix a la carpeta **vm** i canvia el valor de les variables **NOM\_MAUQUINA** i **HOSTNAME** del fitxer **Vagrantfile**:
    - NOM\_MAUQUINA = pj9f4a8.2\_xxyyzz**
    - HOSTNAME = pj9f4a82-xxyyzz.fjeclot.net**Recorda que **xxyyzz** són les 2 primeres lletres del nom i cognoms.
  - Executa la següent ordre per iniciar la màquina virtual treballant amb la nova configuració:  
**vagrant reload**
  - Comprova que els canvis de nom de sistema i nom de la màquina han tingut efecte.

### 3- Crea un dipòsit local i remot per l'aplicació i el seu entorn

- Dins de la màquina física:
  - Accedeix a la carpeta **pj9** → **f4** → **a8** → **pj9f4a8.2**.
  - Crea dins **pj9f4a8.2** a un fitxer de nom **.gitignore** amb el següent contingut:  
**.gitignore**  
**vm/.vagrant**
  - Inicia un dipòsit de **Git** dins de **pj9f4a8.2**.
  - Fes un **add** del continguts del directori **pj9f4a8.2** (excepte el indicats a **.gitignore**).
  - Fes un **commit** del continguts del directori **pj9f4a8.2** (excepte el indicats a **.gitignore**) dins del dipòsit local. El missatge del commit serà: "**Versió 1.0 del projecte pizzeria**".
  - Comprova que s'han pujat al dipòsit local només els fitxer **pizza.html**, **pizza.php**, **Dockerfile** dins de la carpeta **codi** i **Vagrantfile** dins de **vm**. Executa:  
**git show --pretty="" --name-only**
- Crea dins del teu compte de **Github** un dipòsit de nom **pj9f4a8.2** de tipus **Privat**.
- Sincronitza el dipòsit local amb el dipòsit remot amb la informació que trobarà a la secció **...or push an existing repository from the command line**. Recorda que les instruccions s'han d'executar en el directori a on tens el dipòsit, o sigui, dins de **pj9f4a8.2**.
- Refresca la web del dipòsit per assegurar-te que els fitxers del projecte s'han pujat dins de les seves carpetes.

#### 4- Creant una nova versió de l'aplicació i una nova imatge de contenidor Docker per desplegar l'aplicació.

a) Dins de la màquina física:

- Accedeix a la carpeta **pj9 → f4 → a8 → pj9f4a8.2 → codi → app**
  - Modifica la línia **9** de **pizza.html** i indica que la pizza bàsica val **10€**.
  - Modifica la línia **2** de **pizza.php** i indica que el nou valor inicial de la variable **\$vpv** és **10**.
- Accedeix a la carpeta **pj9 → f4 → a8 → pj9f4a8.2 → codi**
  - Modifica la línia **1** de **Dockerfile** i fes que valgui **FROM php:8.2-apache**

b) Accedeix a la màquina virtual:

- Accedeix a la carpeta **/home/vagrant/pizzeria**
- Crea una nova imatge de contenidor **ipizza** versió **1.1**
- Comprova que s'ha creat la nova imatge.
- Crea un nou contenidor de nom **cpizza1.1** a partir de la imatge **ipizza** versió **1.1**
- Comprova que s'ha creat el nou contenidor.
- Comprova l'adreça IP de la màquina virtual.

c) Des de la màquina física comprova amb el navegador i l'adreça IP de la màquina virtual que pots accedir a la nova versió de l'aplicació i que funciona correctament.

d) Si funciona correctament, des de la màquina física torna al directori **pj9 → f4 → a8 → pj9f4a8.2** i a continuació:

- Fes un **add** del continguts del directori **pj9f4a8.2** (excepte el indicats a **.gitignore**).
- Fes un **commit** del continguts del directori **pj9f4a8.2** (excepte el indicats a **.gitignore**) dins del dipòsit local. El missatge del commit serà: "**Versió 1.1 del projecte pizzeria**".
- Sincronitza el dipòsit local amb el dipòsit remot (recorda que ara només cal fer el **push**).
- Comprova que s'ha pujat al dipòsit remot la nova versió del projecte.

#### 5- Creant un compte de Docker Hub. Pujant les imatges creades a Docker Hub

a) Creu-vos un compte de **Docker Hub** a <https://hub.docker.com/>.

b) Crea un repository (dipòsit) per les teves imatges de contenidors amb les següents caracterísiques:

- Namespace → El valor per defecte és correcte.
- Repository name: **ipizza**
- Short description: **App pizza activitat pj9f4a8.2**
- Visibility: **Public**

c) Dins de la màquina virtual (en aquest cas no importa a quin directori us trobeu) executa les següent ordres per associar les imatges locals **ipizza** versió **1.0** i **ipizza** versió **1.1** al seu corresponent dins del repository **ipizza** del teu compte de **Docker Hub**:

```
docker tag ipizza:1.0 <nom_usuari_dockerhub>/ipizza:1.0
docker tag ipizza:1.1 <nom_usuari_dockerhub>/ipizza:1.1
```

d) Fes un **login** al teu compte de Docker Hub:

```
docker login -u <nom_usuari_dockerhub>
```

i si et pots validar dins de Docker Hub sortirà el missatge: **Login Succeeded**

e) Ara puja les 2 imatges creades:

```
docker push <nom_usuari_dockerhub>/ipizza:1.0
docker push <nom_usuari_dockerhub>/ipizza:1.1
```

f) Ara fes un **logout** del teu compte de Docker Hub:

```
docker logout
```

g) Refresqueu el vostre dipòsit de Docker Hub i comproveu que s'han pujat les imatges **ipizza:1.0** i **ipizza:1.1** dins del dipòsit **ipizza** del teu compte.

## **6- Baixant imatges d'un dipòsit d'un usuari de Docker Hub. Creació d'un contenidor a partir de la imatge descarregada.**

a) Descarrega la imatge de contenidor **preus** versió **1.0** del compte de l'usuari **globproj2** de Docker Hub. Executa:

```
docker pull globproj2/preus:1.0
```

b) Comprova que la imatge ha estat descarregada.

c) Assegura't que no hi ha cap altre contenidor en marxa que utilitzi el port **80/tcp** de la màquina virtual. Si en trobes cap, atura'l.

d) Crea un contenidor de nom **cpreus** a partir de la imatge **globproj2/preus:1.0** exportant el port **80/tcp** del contenidor al **80/tcp** de la màquina virtual. Assegura't que el contenidor treballa en 2n terme amb l'opció **-d** i que pot accedir-se i treballar al seu interior per mitjà del terminal i bash afegint les opcions **-i -t**.

e) Comprova que el nou contenidor està en marxa.

f) Comprova l'adreça IP de la màquina virtual

g) Des de la màquina física comprova amb el navegador i l'adreça IP de la màquina virtual que pots accedir a l'aplicació dins del contenidor preu i que funciona correctament.

h) Atura el contenidor i comprova que l'aplicació deixa de funcionar.

## **Lliurament de l'activitat**

a) Comprovacions:

- Mostra dins del teu compte de **GitHub** el dipòsit **pj9f4a8.2** i el seu contingut
- Mostra totes les imatges que tens a la teva màquina virtual.
- Comprova els contenidors de la teva màquina virtual.
- Mostra les imatges pujades al **dipòsit ipizza** del **teu compte de Docker Hub**.
- Posa en marxa el contenidor **cpizza1.1**. Accedeix a l'aplicació. Fes una prova de funcionament.
- Atura **cpizza1.1**. Inicia **cpreus**. Accedeix a l'aplicació. Fes una prova de funcionament.

b) Data límit per obtenir el 100% de la nota: **dijous 14-11-22** a les **18.15**. Posteriorment la nota serà menor.