

Fase 4 - Activitat 8: Desplegament d'aplicacions sobre una màquina virtual en el núvol d'Azure

0- Identificació del grup i activitat:

Curs: ASIX2

Projecte: PJ10 Cloud Computing

Fase: 4

Activitat: 8

Grup/Individual: Grupal

Membres/Alumne:

Introducció i objectius de l'activitat 8

Desplegament d'una aplicacions a partir de:

- Una màquina virtual creades dins del cloud d'Azure.
- Els codis html i php de l'aplicació descarregats des de Github
- Instal·lació del programari de contenidors Docker
- Desplegament de l'aplicació utilitzant contenidors utilitzant fitxers Dockerfile i docker-compose.yml

L'activitat requereix:

- Entendre les especificacions de l'activitat
- Realització d'una planificació de l'activitat utilitzant les eines adequades.
- Desplegament de l'aplicació demanada.
- Creació de documentació:
 - Documentació amb instruccions precises i comentades de la creació de la màquina en el núvol necessària pel desplegament de l'aplicació orientada a ser llegida per un altre administrador que hagi de fer la mateixa feina
 - Documentació amb instruccions precises i comentades del desplegament de l'aplicació orientada a ser llegida per un altre administrador que hagi de fer la mateixa feina
- Treball en equip.

El lliurament de l'activitat té 2 parts:

- Lliurament de la planificació i de la documentació → Hores de mentoria (professor mentor).
- Comprovació del funcionament de les aplicacions → Hores de projecte tècnic (professor tècnic).

Part 1 – Planificació de l'activitat

a) Creació del diagrama de Gantt

b) Creació de la planificació de les activitats amb Trello

c) Creació de la taula (o quadre) de dependències

Part 2 - Creació d'una màquina en el núvol i instal·lació de programari

a) Accedeix a la web d'administració del compte d'Azure i crea una màquina en el núvol que ha de tenir les següents característiques:

- Nom de la màquina: **pj10f4a7-grupXX** a on XX és el vostre número de grup.
- Ports oberts: **22, 80 i 443**

La resta de característiques de la màquina són a l'elecció del grup i d'acord amb els criteris utilitzats a l'activitat **pj10f4a7**.

b) Un cop creada, troba la seva adreça IP pública i accedeix a la màquina via SSH.

- c) Un cop creada la màquina virtual en el núvol, instal·la:
- El programari per treballar amb control de versions **Git**.
 - El programari per treballar amb contenidors **Docker**.

Part 2- Estructura de la carpeta de l'aplicació

Dins d'aquesta activitat es desplegarà una aplicacions utilitzant contenidors Docker sobre la màquina creada en el núvol d'Azure, de manera que haureu de crear previament una estructura de carpetes com aquesta:

- La carpeta de l'aplicació de nom **pj10f4a8**.
- Dins de la carpeta **pj10f4a8** s'haurà de crear:
 - Una carpeta de nom **app** pels fitxers PHP o HTML.
 - Una carpeta de nom **sql** pels scripts sql.
 - Un fitxer **Dockerfile**
 - Un fitxer **docker-compose.yml**
 - Un fitxer **.dockerignore**.
 - Una carpeta de nom **documents** per desar la documentació.
 - El dipòsit local **.git** que s'haurà crear a l'inicialitzar **git** dins de **pj10f4a8** un cop creats els anteriors fitxers i carpetes.

Part 3 - Desplegament d'una aplicació d'accés a bases de dades utilitzant Apache i MySQL

a) Hem de desplegar una aplicació que necessitarà:

- Un contenidor Docker amb:
 - Apache + PHP 8.3
 - L'extensió PDO de PHP per poder fer aplicacions PHP amb accés a bases de dades MySQL.
 - El codi PHP (HTML, CSS, etc) d'una aplicació per mostrar dades d'una taula d'una base de dades MySQL.
- Un contenidor Docker amb:
 - El servidor servidor MySQL.
 - Una base de dades i una taula a la qual s'accedirà des del docker amb l'aplicació PHP.
 - Dades dins de la taula.
- Una xarxa de contenidors dockers.
- Volumes de 2 tipus:
 - Carpetes compartides
 - Volumes creats.

b) Dins de la màquina virtual en el cloud d'Azure, accediu a la carpeta a **app** i des de dins de la carpeta descarregueu els següents fitxers:

- <https://raw.githubusercontent.com/asix2pj9/pj9f4a87apl2/refs/heads/main/app/app.css>
- <https://raw.githubusercontent.com/asix2pj9/pj9f4a87apl2/refs/heads/main/app/app.php>
- <https://raw.githubusercontent.com/asix2pj9/pj9f4a87apl2/refs/heads/main/app/biblioteca.php>
- <https://raw.githubusercontent.com/asix2pj9/pj9f4a87apl2/refs/heads/main/app/config.php>
- <https://raw.githubusercontent.com/asix2pj9/pj9f4a87apl2/refs/heads/main/app/index.html>

i comprova que ara, dins d'**app** tens els fitxers **app.css**, **app.php**, **biblioteca.php**, **config.php** i **index.html**.

c) Dins de la màquina virtual en el cloud d'Azure, accediu a la carpeta a **sql** i des de dins de la carpeta descarregueu els següents fitxers:

- <https://raw.githubusercontent.com/asix2pj9/pj9f4a87apl2/refs/heads/main/sql/bd.sql>

i comprova que ara, dins de **sql** tens el fitxer **bd.sql**.

d) Entreu dins del directori **sql**, i modifiqueu el script **bd.sql** perquè quan s'executi pugui crear dins d'un servidor MySQL:

- Una base de dades de nom **bdPj10f4a8**.
- Una taula dins de la base de dades **bdPj10f4a8** de nom **usuaris**.

e) Desenvolpeu un fitxer **Dockerfile** de manera que pugui ser utilitzat per crear una imatge de contenidor que tingui:

- Apache i PHP8.3.
- La biblioteca amb l'extensió PDO de PHP necessària per poder desenvolupar aplicacions PHP amb accés a bases de dades MySQL.

Per tant, aquest fitxer només té 2 línies:

- La primera línia per tenir Apache + PHP 8.3 → Això ja ho hem fet en una altra activitat.
- La segona línia per afegir a biblioteca de l'extensió PDO de PHP a la imatge. Hauràs d'escriure la línia: `RUN docker-php-ext-install pdo pdo_mysql`

f) Desenvolpeu un fitxer **docker-compose.yml** per desplegar l'aplicació d'accés a base de dades que hi ha dins de la carpeta **app** utilitzant contenidors. El fitxer **docker-compose.yml** ha de fer aquestes tasques:

- Ha de crear un servei de nom **sMysql** que:
 - Utilitzi darrera imatge de contenidor oficial dins Docker Hub de MySQL que té el nom **mysql** i la versió **9.1**.
 - Crei a partir de la imatge descarregada un contenidor de nom **cMysql**.
 - Que exporti el port **3306** del contenidor al port **33306** de la màquina virtual.
 - Que utilitzant l'opció dels volums per compartir carpetes, comparteixi la carpeta ja existent **sql** de la màquina virtual amb un nou directori de nom **/sql** dins del contenidor. Recordeu que dins del màquina virtual la carpeta **sql** està en el mateix directori que **docker-compose.yml**. La carpeta **/sql** de contenidor la crearà automàticament el programa **docker compose**.
 - Que crearà automàticament un volum dins de la màquina virtual de nom **vDades** que estarà associat a la carpeta **/var/lib/mysql** del contenidor. Així podem accedir i fer backups de les bases de dades.
 - Ha de crear les següents variables d'entorn (environment):
 - **MYSQL_DATABASE** que val el mateix que el nom de la base de dades que es crearà amb el script **bd.sql**, i per tant és **'bdPj10f4a8'**. Les cometes són importants.
 - **MYSQL_USER** que valdrà **'asix2'** (això crea un usuari **asix2** dins del servidor MySQL del contenidor amb tots els permisos necessaris).
 - **MYSQL_PASSWORD** que valdrà **'FjeClot2425#'** (aquest serà el password d'**asix2**).
 - **MYSQL_ROOT_PASSWORD** que valdrà **'FjeClot2425@'**
 - Forma part d'una xarxa identificada dins de **docker-compose.yml** amb el nom **xPj10f4a8**.
- Ha de crear un servei de nom **sWebApp** que:
 - Generi una imatge de contenidor a partir del **Dockerfile** creat a l'apartat e). El nom de la imatge serà **iwebapp** i la versió **1.0**.
 - Crei a partir de la imatge generada un contenidor de nom **cWebApp**.
 - Que exporti el port **80** del contenidor al port **80** de la màquina virtual.
 - Que utilitzant volums comparteixi la carpeta **app** de la màquina virtual amb el directori **/var/www/html** del contenidor. Recordeu que dins del màquina virtual la carpeta **app** està en el mateix directori que **docker-compose.yml**.
 - Depèn del servei **sMysql**. Si el servei **sMysql** no s'inicia tampoc s'inicia **sWebApp**.
 - Forma part d'una xarxa identificada dins de **docker-compose.yml** amb el nom **xPj10f4a8**.

- Ha de generar una xarxa identificada dins de **docker-compose.yml** amb el nom **xPj10f4a8** que:
 - dins de la màquina virtual s'identificarà amb el nom **xarxa**.
 - utilitzarà un driver de tipus **bridge**.
- Ha de generar una volum identificat dins de **docker-compose.yml** amb el nom **xvDades** que dins de la màquina virtual s'identificarà amb el nom **dades**.

g) Assegureu-vos que la imatge **iwebapp:1.0** que es crearà amb el fitxer **docker-compose.yml** no tingui dins carpetes i fitxers innecessaris. Feu que el contingut de **.dockerignore** sigui aquest:

```
.dockerignore
.git
app
sql
docker-compose.yml
```

h) Ara, entreu dins del directori **app** de la màquina física, i modifiqueu el fitxer **config.php** tenint en compte que:

- **\$host** ha de ser igual al nom del contenidor amb MySQL. El nom és el mateix que l'indicat a **docker-compose.yml**.
- **\$user** és l'usuari de la base de dades l'indicat a **docker-compose.yml** dins del paràmetre **MYSQL_USER**.
- **\$passwd** és la contrasenya indicada a **docker-compose.yml** per l'usuari definit dins del paràmetre **MYSQL_USER**.
- **\$bd** és la base de dades indicada a **bd.sql** i també a **docker-compose.yml**.
- **\$taula** és la taula indicada a **bd.sql**.

i) Poseu en marxa l'aplicació amb **docker compose** executant-se en **2n terme** i després comprova que els 2 cotenidors funcionen correctament.

k) Afegeix la taula **usuaris** i les seves dades a la base de dades **bdPj10f4a8** de l'aplicació executant el script **bd.sql**. Realitza els següents passos:

- Accedeix al contenidor de la **cMySQL** executant: **docker exec -it cMysql bash**
- Dins del contenidor accedeix a la carpeta **/sql** executant: **cd /sql**
- Executa el script **bd.sql** que ara també es troba dins **/sql**: **mysql -u asix2 -p -e './bd.sql'**
- Escric la contrasenya de l'usuari **asix2** indicada a **docker-compose.yml**.
- Comprova que la taula **usuaris** i les seves dades s'han creat. Executa: **mysql -u asix2 -p -e 'use bdPj10f4a8;select * from/usuaris;'**
- Surt del contenidor. Executa: **exit**

l) A partir de l'adreça IP pública de la màquina en el núvol d'Azure, accedeix a l'aplicació amb el navegador des de la màquina física. Comprova que pots veure i llegir la taula d'**usuaris** de la base de dades de l'aplicació.

m) Si l'aplicació funciona, fes un commit dels fitxers i carpetes dins de la carpeta **pj10f4a8** amb el comentari "**Commit 1 de l'activitat pjf104a8**".

n) Pujeu a un compte de Github de qualsevol membre del vostre grup tots els fitxers de l'activitat.

Part 4 – Creació de la documentació

- Documentació en format PDF amb instruccions precises i comentades de la creació de la màquina en el núvol orientada a ser llegida per un altre administrador que hagi de fer la mateixa feina

- Documentació amb instruccions precises i comentades del desplegament de l'aplicació orientada a ser llegida per un altre administrador que hagi de fer la mateixa feina

Lliurament de l'activitat

a) Lliurament de la part de desenvolupament de l'activitat:

- Demostració que:
 - Heu creat una màquina virtual en el núvol amb el nom indicat.
 - L'adreça IP de la màquina virtual en el núvol.
 - Els 2 contenidors CMySQL i CWebApp estan en marxa dins de la màquina virtual en el núvol.
 - Podeu posar accedir a l'aplicació i podeu visualitzar les dades de la taula **usuaris**.
 - Que podeu aturar els contenidor amb l'aplicació d'accés a bases de dades i que l'aplicació deixa de funcionar.
 - Els fitxers del dipòsit de Github.
- Data límit per obtenir el 100% de la nota: **dimarts 1-4-25 a les 19.10**. (posteriorment és el 70%)

b) Lliurament de la planificació i documentació → D'acord amb les indicacions del professor mentor.