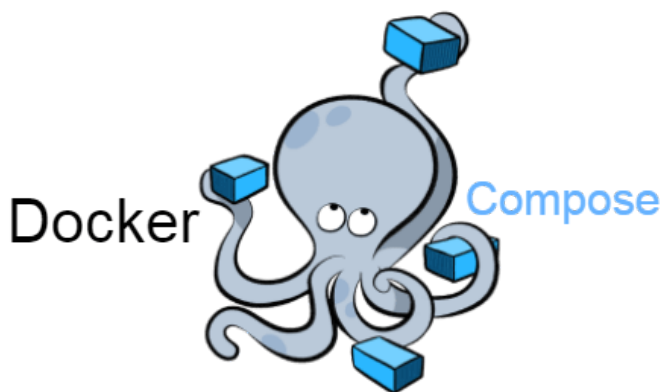


Ins la Guineueta

Proyecto 3

Módulo M14



Sergi Espuch Escobar
Bernat Torrent Pinart

Curso: Administración de
Sistemas Informáticos Y Redes

Tutores: Michael Ragel,
Sergio Garrido i Pere Arnau

11-05-2024

Contenido:

INTRODUCCION.....	2
COMPARATIVA.....	2
COSTES DE PROYECTO.....	3
GANTT.....	4
Inicial:.....	5
Final:.....	5
SERVICIOS.....	6
CONTENEDORES.....	7
WEB.....	24
Base de datos:.....	24
Diagrama ER:.....	24
MR:.....	24
PRUEBAS DEL PROYECTO.....	28
RAÍZ DEL PROYECTO:.....	40
DOCKER HUB Y GITHUB.....	41
CONCLUSIÓN.....	41
WEBGRAFÍA.....	42

INTRODUCCION

El objetivo de nuestra empresa es la creación de la infraestructura de empresas o centros educativos. Nos contratan porque necesitan una estructura con varios servicios, por ejemplo una página web, una base de datos, un DHCP, un DNS para el dominio de la empresa, etc.

En este caso documentamos para el "Institut Vila de Gràcia" un centro de Educación Secundaria Obligatoria que se encuentra en el barrio de "Gràcia" Barcelona. Entre los servicios que hemos comentado hace un momento en esta ocasión también hemos hecho un sistema de carpetas compartidas para el profesorado, los secretarios y los alumnos, así también como un sistema de backup diario para que los servicios puestos en marcha tengan un respaldo. También hemos implementado un servicio para que el TI del centro pueda gestionar los servicios una vez hecha la instalación. Para ello hemos puesto un servicio "Portainer" que se encarga de gestionar gráficamente los contenedores de Docker.

COMPARATIVA

ISC-DHCP-SERVER: Hemos decidido usar "isc-dhcp-server" en un contenedor separado porque una alternativa es una imagen "dnsmasq" que es DHCP y DNS en una, pero su funcionamiento es muy poco claro y su documentación nos ha parecido pobre. Así que nos hemos decantado por usar el servicio propio de DHCP de Linux. A lo mismo nos pasa con DNS, hemos usado "bind9" en un contenedor aparte por la misma razón.

MySQL vs. alternativas: Bases de datos hay muchas (MariaDB, PostgreSQL, MongoDB...) pero con la que estamos más familiarizados es con MySQL además de una integración perfecta con "phpMyAdmin", aparte tiene un rendimiento superior a muchas de las bases de datos comentadas anteriormente.

PhpMyAdmin vs. alternativas: Gestor de bases de datos también hay muchas (Adminer, PgAdmin, DBeaver) pero entre que algunas son de pago y también estamos familiarizados con "phpMyAdmin" por eso la hemos escogido.

Apache vs. Alternativas: Apache es un servidor web para Linux hay varias alternativas como (Nginx, Apache Tomcat, H2O web server, etc.) pero el mayor rendimiento de estos servidores web los tiene Nginx y apache hemos decidido decantarnos por Apache, ya que sabemos más como funciona.

Samba vs. Alternativas: Samba es un servidor de impresión y compartición de carpetas, también hay otras opciones como (Artica, Linux Network File Share, etc.) pero con integración con el protocolo SMB/CIFS samba es una muy buena opción para la comunicación entre Linux y Windows, así que hemos decidido agregar este servicio.

Duplicati vs. Alternativas: "Duplicati" es un servidor de copias de seguridad, con encriptación a optar, hay diferentes alternativas como (Rsync, Rclone, Duplicacy, etc.) pero según hemos indagado Duplicati tiene muy buena integración con Docker, no es así con algunas de las alternativas comentadas anteriormente, por eso lo hemos escogido.

Portainer vs. Alternativas: "Portainer" es un gestor gráfico de contenedores, las principales alternativas son (Kubernetes, Rancher y Nanobox) pero Portainer tiene una configuración sencilla y unas posibilidades muy grandes, por eso hemos optado por ella.

COSTES DE PROYECTO

Los costos del proyecto se componen principalmente de dos aspectos: recursos humanos y tecnológicos. Necesitamos un programador y un administrador de redes, ambos cobran 45€ por hora. Además, se requiere un dominio web por 300€ y un servidor para contenedores Docker por 5700€. Estos elementos son esenciales para el desarrollo y la implementación exitosa del proyecto.

Este proyecto se ha desarrollado considerando un margen de tiempo de 132 horas. Esto significa que se ha estimado que el trabajo necesario para completar todas las tareas y objetivos del proyecto requerirá un total de 132 horas

Recurso	
Personal	Cantidad
Administrador de sistemas	1
Programadores	1
Materials	
Servidor	1
DominioWeb	1

Recurso	
Personal	Precio
Administrador de sistemas	45€/h
Programadores	45€/h
Materials	
Servidor	5.700€
DominioWeb	300€

servicio	Precio	Cantidad/horas	Total
Implementación	45,00 €	132	5.940,00 €
Hardware	6.000,00 €	1	6.000,00 €
Mantenimiento	35,00 €	24	840,00 €
Mantenimiento Urgente	34,20 €	4	136,80 €
TOTAL			12.916,80 €

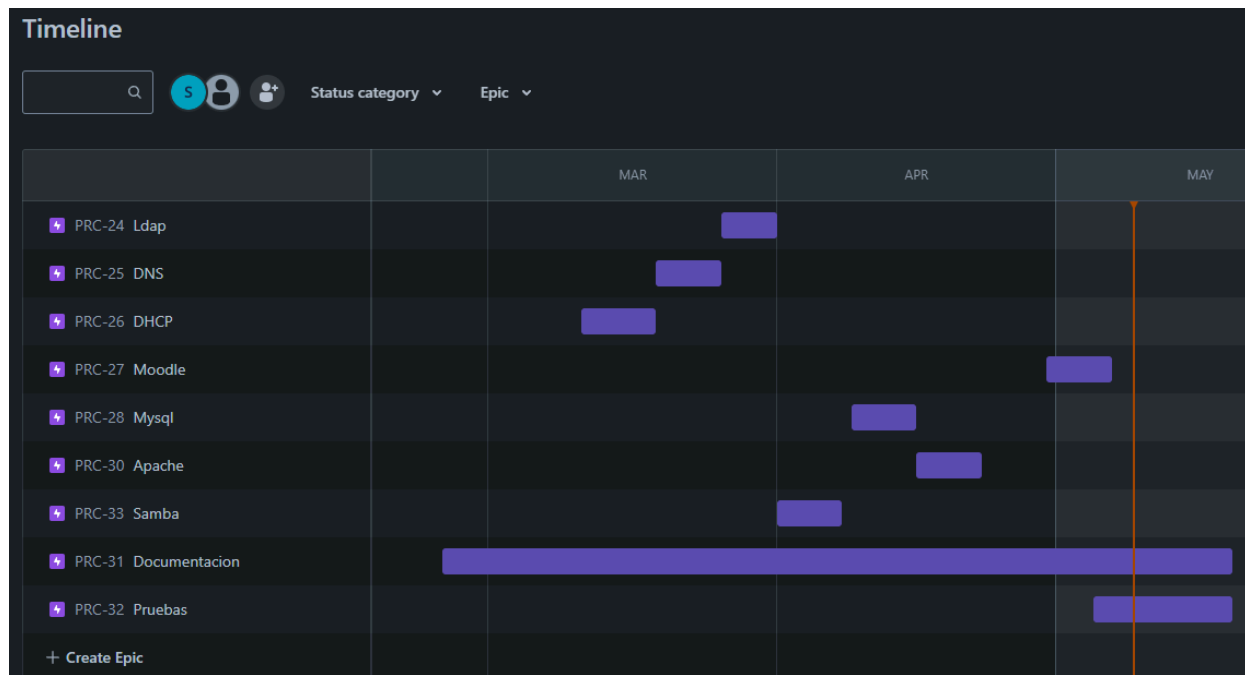
GANTT

La planificación que he estructurado ha sido la siguiente, empezamos el proyecto el 27 febrero, poco a poco hemos ido dando paso por paso, empezando por el Apache, PhpMyAdmin, MySQL e ISC-DHCP-SERVER, bind9, Portainer, Duplicati, Samba.

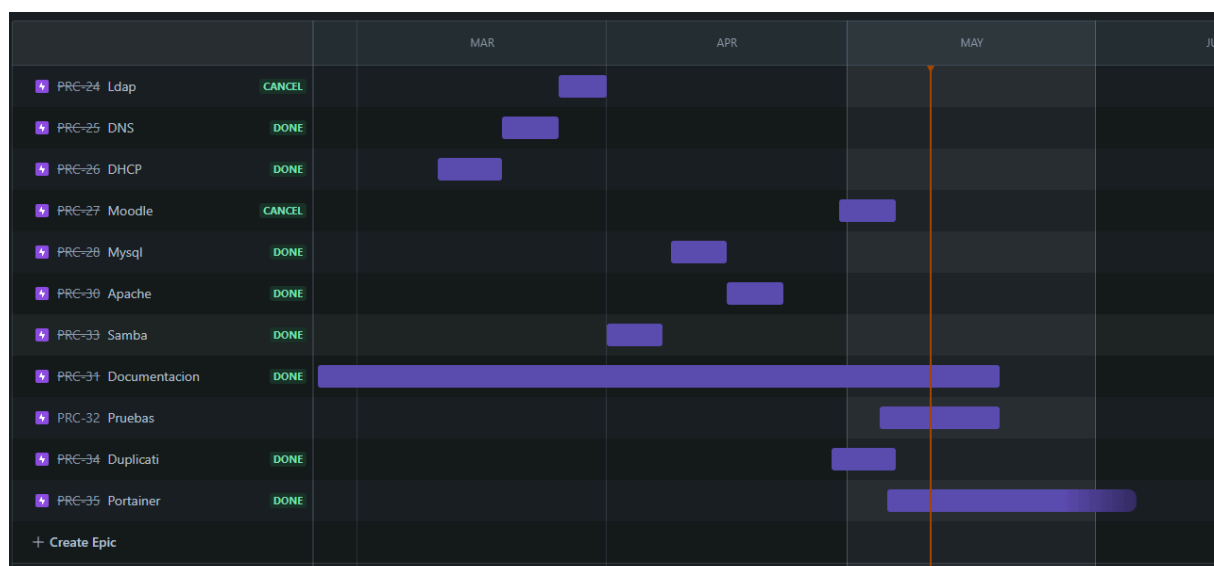
La configuración tanto de LDAP como de OpenLDAP resultaba bastante difícil y nos generaba problemas de forma recurrente. A pesar de múltiples intentos, no logramos solventar los obstáculos.

Con respecto a Moodle, identificamos el problema, pero nos encontrábamos perdidos en cuanto a la solución. Básicamente, el inconveniente radicaba en que Moodle intentaba acceder a una base de datos preexistente sin poder establecer la conexión adecuada con nuestra base de datos existente.

Inicial:



Final:



SERVICIOS

Los siguientes servicios se implementarán usando “docker-compose” una herramienta de gestión de contenedores con pequeñas imágenes de distintos servicios así haciendo mucho más eficiente la virtualización al usar pocos recursos, para un centro educativo va perfecto.

1. isc-dhcp-server: El contenedor que tendrá este servicio se encargará de gestionar las IP de la red del instituto.
2. ubuntu/bind: El contenedor que tendrá este servicio se encargará de gestionar las consultas DNS locales de la red.
3. mysql-server: El contenedor que tendrá este servicio se encargará de gestionar la o las bases de datos que pueda contener el centro educativo.
4. phpmyadmin: El contenedor que tendrá este servicio se encargará de gestionar la base de datos gráficamente para un uso más fácil para el encargado TI del centro.
5. Apache: El contenedor que tendrá este servicio se encargará de la página web del centro educativo, como así también el “IEduca” que está hecho para el centro.
6. dperson/samba: El contenedor que tendrá este servicio se encargará de gestionar las carpetas compartidas en red para el centro.
7. linuxserver/duplicati: El contenedor que tendrá este servicio se encargará de gestionar diariamente las copias de seguridad de todos los contenedores, así como también las carpetas que no solo estén dentro de los servicios.
8. portainer/portainer-ce: El contenedor que tendrá este servicio se encargará de de que los contenedores y volúmenes se puedan gestionar de una manera mucho más sencilla haciendo más autónomos a las empresas que nos contraten y no dependan de aprender difíciles comandos con probabilidades altas para tener un error y perderlo todo.

CONTENEDORES

```
bernat@srvdocker:~/PROJECTE_3_ACAVAT$ cat docker-compose.yaml
version: '3'
services:
  # Servidor DHCP #
  dhcpd:
    build: ./dhcp/
    container_name: isc-dhcp-server
    hostname: dhcp
    deploy:
      replicas: 1
      placement:
        constraints: [node.role == manager]
    image: isc-dhcp-server
    ports:
      - "67/udp"
    volumes:
      - ./dhcp/dhcpd/dhcpd.conf:/data/dhcpd/dhcpd.conf
      - ./dhcp/dhcpd/ubuntu.conf:/etc/dhcp/ubuntu.conf
      - ./dhcp/dhcpd/windows.conf:/etc/dhcp/windows.conf
    restart: unless-stopped
    networks:
      macvlan1:
        ipv4_address: 192.168.10.200

  # Servidor DNS #
  bind:
    image: ubuntu/bind9
    container_name: bind9
    hostname: bind
    volumes:
      - ./bind-conf/named.conf:/etc/bind/named.conf
      - ./bind-conf/named.conf.options:/etc/bind/named.conf.options
      - ./bind-conf/named.conf.default-zones:/etc/bind/named.conf.default-zones
      - ./bind-conf/named.conf.local:/etc/bind/named.conf.local
      - ./bind-conf/resolv.conf:/etc/resolv.conf
      - ./bind-zones/db.insviladegracia.cat:/var/lib/bind/db.insviladegracia.cat
      - ./bind-zones/db.10.168.192.in-addr.arpa:/var/lib/bind/db.10.168.192.in-addr.arpa
    ports:
      - "53:53/tcp"
      - "53:53/udp"
    command: named -g -c /etc/bind/named.conf
    networks:
      macvlan1:
        ipv4_address: 192.168.10.201
    restart: unless-stopped
```



```
# Servidor DATABASE MYSQL #
db:
  image: mysql
  container_name: mysql-server
  hostname: db
  environment:
    MYSQL_ROOT_PASSWORD: 12345aA
    MYSQL_DATABASE: insviladegracia
    MYSQL_USER: admin
    MYSQL_PASSWORD: 12345aA
  ports:
    - "3307:3307"
  command: ["--init-file", "/docker-entrypoint-initdb.d/insviladegracia.sql"]
  volumes:
    - ./sql_database/insviladegracia.sql:/docker-entrypoint-initdb.d/insviladegracia.sql
    - dbdata:/var/lib/mysql
  networks:
    macvlan1:
      ipv4_address: 192.168.10.202
  restart: unless-stopped

# Gestor DATABASE PhpMyAdmin #
phpmyadmin:
  image: phpmyadmin
  container_name: phpmyadmin
  hostname: phpmyadmin
  links:
    - db
  environment:
    PMA_HOST: db
    PMA_PORT: 3306
    PMA_ARBITRARY: 1
  depends_on:
    - db
  ports:
    - 8081:80
  volumes:
    - phpmyadmin_data:/hola
  networks:
    macvlan1:
      ipv4_address: 192.168.10.203
  restart: unless-stopped

# Servidor WEB Apache #
apache:
  build: './apache_docker/'
  hostname: apache
  #depends_on:
    #- php
  ports:
    - "443:443"
    - "80:80"
  volumes:
    - ./www:/var/www/html/
  networks:
    macvlan1:
      ipv4_address: 192.168.10.204
  restart: unless-stopped
```

```
# Servidor SAMBA comparticion de carpetas y impresoras #
samba:
  image: dperson/samba
  hostname: SAMBA
  volumes:
    - ./SAMBA-Professors:/Professors
    - ./SAMBA-Alumnes:/Alumnes
    - ./SAMBA-Secretaria:/Secretaria
    - ./SAMBA-CONFIG/smb.conf:/etc/samba/smb.conf
    - ./SAMBA-CONFIG/usuarios.sh:/usuarios.sh
  networks:
    macvlan1:
      ipv4_address: 192.168.10.205
  restart: unless-stopped

# Servidor DUPLICATI BACKUPS #
duplicati:
  image: linuxserver/duplicati
  container_name: duplicati
  hostname: duplicati
  environment:
    - PUID=0
    - PGID=0
    - TZ=EUROPE/MADRID
    - DB_TYPE=mysql
    - DB_CONNECTION_STRING=Server=db;Port=3306;Database=duplicati;Uid=admin;Pwd=12345aA;
  volumes:
    - ./duplicati:/config
    - ./duplicati_backups:/backups
    - /home/bernat/proyecto3/proyecto3-v1/proyecto3:/source
  ports:
    - "8200:8200"
  restart: unless-stopped
  networks:
    macvlan1:
      ipv4_address: 192.168.10.206

# Servidor PORTAINER gestor de contenedores WEB #
portainer:
  image: portainer/portainer-ce
  hostname: portainer
  container_name: portainer-ce
  ports:
    - 9000:9000
  volumes:
    - /var/run/docker.sock:/var/run/docker.sock
    - portainer_data:/data
  networks:
    macvlan1:
      ipv4_address: 192.168.10.207
  restart: always
```

```
# VOLUMENES LOCALES DE DOCKER #
volumes:
  dbdata:
  phpmyadmin_data:
  duplicati:
  duplicati_backups:
  portainer_data:

# RED PERSONALIZADA PARA TODOS LOS CONTENEDORES #
networks:
  macvlan1:
    driver: macvlan
    driver_opts:
      parent: enp2s0
    ipam:
      config:
        - subnet: 192.168.10.0/24
bernat@srvdocker:~/PROJECTE_3_ACAVAT$
```

DHCP:

En el contenedor dhcp lo que hemos hecho es crear desde una build de debian instalar el paquete isc-dhcp-server, luego le ponemos una carpeta que será la de volumen y copiamos en entrypoint.sh y usa el puerto 67 UDP, usamos el despliegue de Docker Swarm, le decimos que solo se cree una réplica, y que solo se despliegue en entorno de nodo manager.

```
bernat@srvdocker:~/PROJECTE_3_ACAVAT$ cat dhcp/dockerfile
FROM debian
MAINTAINER Bernat Torrent, Sergi Espuch

# Install the dhcpd server.
RUN apt update
RUN apt install -y isc-dhcp-server

# Copy init script
RUN mkdir -p /data
COPY entrypoint.sh /data/entrypoint.sh
RUN chmod +x /data/entrypoint.sh

# Client requests
EXPOSE 67/udp

ENTRYPOINT ["/data/entrypoint.sh"]
bernat@srvdocker:~/PROJECTE_3_ACAVAT$
```

que es el que se encarga de ejecutar el dhcp.leases si no existe y ejecutar el servicio para la puesta en marcha.

```
bernat@srvdocker:~/PROJECTE_3_ACAVAT$ cat dhcp/entrypoint.sh
#!/bin/sh

# Initialize the lease file if it doesn't exist.
touch /data/dhcpd/dhcpd.leases

# Start devpi-server.
dhcpd -cf /data/dhcpd/dhcpd.conf -lf /data/dhcpd/dhcpd.leases --no-pid -4 -f
bernat@srvdocker:~/PROJECTE_3_ACAVAT$
```

BIND9:

Este servicio utiliza la versión de Ubuntu/bind9 una imagen creada para DNS, para ello lo que hacemos es que en el contenedor ejecutamos una imagen de bind9 basada en Ubuntu. Le copiamos unos ficheros de configuración con las zonas directas e inversas, para su resolución de nombres. Y los tres ficheros principales para su funcionamiento, le indicamos que usa el puerto 53 por TCP y UDP. Tiene dos volúmenes, uno de configuración general y la otra para las zonas.

La configuración general más importante sobre todo es el `named.conf.options`, se encargará de saber por qué puerto escucha y por la IP, los reenviadores si él no sabe lo que estás buscando y la ruta donde se encuentran los ficheros de zonas.

```
bernat@srvdocker:~/PROJECTE_3_ACAVAT$ cat bind-conf/named.conf.options
options {
    directory "/var/lib/bind";
    listen-on port 53 { 127.0.0.1; 192.168.10.201; };
    allow-query { 192.168.10.0/24; localhost; };
    forwarders { 8.8.8.8; 8.8.4.4; };
    forward only;
    dnssec-validation auto;
    listen-on-v6 { any; };
};
bernat@srvdocker:~/PROJECTE_3_ACAVAT$
```

Zona Directa

La zona directa es donde le decimos el nombre de la máquina a que IP tiene que ir.

```
bernat@srvdocker:~/PROJECTE_3_ACAVAT$ cat bind-zones/db.insviladegracia.cat
$TTL 604800 ; 1 week
@ IN SOA bind.insviladegracia.cat. root.insviladegracia.cat. (
    205 ; serial
    604800 ; refresh (1 week)
    86400 ; retry (1 day)
    2419200 ; expire (4 weeks)
    604800 ; minimum (1 week)
)
@
dhcp.insviladegracia.cat.      IN      NS      bind.insviladegracia.cat.
bind.insviladegracia.cat.      IN      A        192.168.10.200
bind.insviladegracia.cat.      IN      A        192.168.10.201
db.insviladegracia.cat.        IN      A        192.168.10.202
phpmyadmin.insviladegracia.cat. IN      A        192.168.10.203
apache.insviladegracia.cat.    IN      A        192.168.10.204
SAMBA.insviladegracia.cat.     IN      A        192.168.10.205
duplicati.insviladegracia.cat. IN      A        192.168.10.206
portainer.insviladegracia.cat. IN      A        192.168.10.207
www.insviladegracia.cat.       IN      CNAME    apache.insviladegracia.cat.
bernat@srvdocker:~/PROJECTE_3_ACAVAT$
```

Zona Indirecta

La zona indirecta hace lo mismo, pero al revés, con una IP redirige al nombre de la máquina

```

bernat@srvdocker:~/PROJECTE_3_ACAVAT$ cat bind-zones/db.10.168.192.in-addr.arpa
$TTL 604800 ; 1 week
@ IN SOA bind.insviladegracia.cat. root.insviladegracia.cat.(
    150 ; serial
    604800 ; refresh (1 week)
    86400 ; retry (1 day)
    2419200 ; expire (4 weeks)
    604800 ; minimum (1 week)
)
@      IN      NS      bind.insviladegracia.cat.
200    IN      PTR      dhcp.insviladegracia.cat.
201    IN      PTR      bind.insviladegracia.cat.
202    IN      PTR      db.insviladegracia.cat.
203    IN      PTR      phpmyadmin.insviladegracia.cat.
204    IN      PTR      apache.insviladegracia.cat.
205    IN      PTR      SAMBA.insviladegracia.cat.
206    IN      PTR      duplicati.insviladegracia.cat.
207    IN      PTR      portainer.insviladegracia.cat.
12     IN      PTR      DESKTOP-S3SRC91.insviladegracia.cat.
bernat@srvdocker:~/PROJECTE_3_ACAVAT$

```

Docker-compose DNS.

```

# Servidor DNS #
bind:
  image: ubuntu/bind9
  container_name: bind9
  hostname: bind
  volumes:
    - ./bind-conf/named.conf:/etc/bind/named.conf
    - ./bind-conf/named.conf.options:/etc/bind/named.conf.options
    - ./bind-conf/named.conf.default-zones:/etc/bind/named.conf.default-zones
    - ./bind-conf/named.conf.local:/etc/bind/named.conf.local
    - ./bind-conf/resolv.conf:/etc/resolv.conf
    - ./bind-zones/db.insviladegracia.cat:/var/lib/bind/db.insviladegracia.cat
    - ./bind-zones/db.10.168.192.in-addr.arpa:/var/lib/bind/db.10.168.192.in-addr.arpa
  ports:
    - "53:53/tcp"
    - "53:53/udp"
  command: named -g -c /etc/bind/named.conf
  networks:
    macvlan1:
      ipv4_address: 192.168.10.201
  restart: unless-stopped

```

SAMBA:



dperson/samba ☆660

By [dperson](#) · Updated almost 3 years ago

[IMAGE](#)

Este servicio está montado sobre la imagen dperson/samba una imagen modificada de samba para la compartición de impresoras y directorios entre los protocolos SMB/CIFS, como lo hemos configurado, hemos puesto tres volúmenes persistentes, eso quiere decir que si se modifica en el host también se modifica en el contenedor.

Hay dos ficheros de configuración, el smb.conf que ahí es donde tenemos que poner los recursos compartidos, el WORKSPACE de trabajo, etc. y el de la creación de usuarios de samba, como sabemos los usuarios de samba tiene que existir en el sistema para poder iniciar sesión.

```
bernat@srvdocker:~/PROJECTE_3_ACAVAT$ cat SAMBA-CONFIG/smb.conf
[global]
    workgroup = insviladegracia
    server string = SAMBA
    netbios name = SAMBA
    security = user
    map to guest = bad user
    dns proxy = no

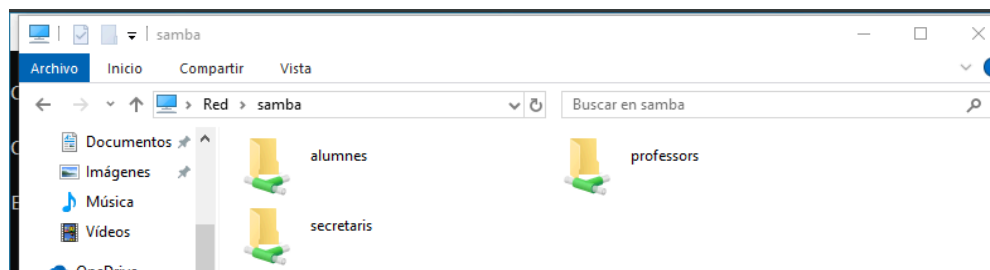
[homes]
    comment = Home Directories
    browseable = no
    read only = no
    create mask = 0700
    directory mask = 0700
    valid users = %S

[professors]
    path = /Professors
    public = yes
    browseable = yes
    writable = yes
    comment = Carpeta de professors
    printable = no
    valid users = @profesores, @administradores

[secretaris]
    path = /Secretaria
    public = yes
    browseable = yes
    writable = yes
    comment = Carpeta de secretaris
    printable = no
    valid users = @secretarios, @administradores

[alumnes]
    path = /Alumnes
    public = yes
    browseable = yes
    writable = no
    comment = Carpeta publica d'alumnes
    printable = no
    guest ok = yes
    write list = @administradores
bernat@srvdocker:~/PROJECTE_3_ACAVAT$
```

Vista de las carpetas compartidas



Como vemos hay 3 carpetas, una es de alumnos que no pueden escribir dentro de la carpeta solo copiar recursos que haya para el uso académico, el de profesores que solo pueden entrar los usuarios que sean del grupo profesores y el de secretaria que solo pueden entrar los usuarios de secretaria para todo el tema de trámites, documentación, etc.

La creación de usuarios la hacemos con un script en bash, dentro de un volumen persistente, como ya hemos comentado el volumen persistente cuando editas algo en el host, también se edita en el contenedor, así que la creación de nuevos usuarios es bastante fácil solo tienes que editar el fichero, entrar en el contenedor y ejecutarlo, si existen no lo ejecutará y si no existen ejecutará la línea del nuevo usuario.

```
bernat@srvdocker:~/PROYECTE_3_ACAVAT$ cat usuarios.sh
#!/bin/bash

# Crear grupos
echo "Creando grupos..."
groupadd profesores
groupadd administradores
groupadd secretarios
groupadd alumnos

# Crear usuarios
echo "Creando usuarios..."
useradd -m -g profesores profesor
useradd -m -g administradores administrador
useradd -m -g secretarios secretario
useradd -m -g alumnos alumno


# Asignar contraseñas a los usuarios
echo "Asignando contraseñas a los usuarios..."
echo "profesor:profesor" | chpasswd
echo "administrador:administrador" | chpasswd
echo "secretario:secretario" | chpasswd
echo "alumno:alumno" | chpasswd

# Crear usuarios en Samba
echo "Creando usuarios en Samba..."
echo -e "profesor\nprofesor" | smbpasswd -a -s profesor
echo -e "administrador\nadministrador" | smbpasswd -a -s administrador
echo -e "secretario\nsecretario" | smbpasswd -a -s secretario
echo -e "alumno\nalumno" | smbpasswd -a -s alumno

echo "Usuarios y contraseñas creados."
echo "Usuarios y contraseñas SAMBA creados."
bernat@srvdocker:~/PROYECTE_3_ACAVAT$
```

DUPLICATI:



linuxserver/duplicati  Sponsored OSS ☆ 389

By linuxserver.io • Updated about 13 hours ago

A Duplicati container, brought to you by LinuxServer.io.

[IMAGE](#)

Overview Tags

Este servicio es lo hemos montado sobre una imagen de linuxserver/duplicati, su función es crear copias de seguridad para los contenedores y sus volúmenes persistentes, la gracia es que fuera no en local la copia de seguridad y fuera en la nube, y se puede con esta herramienta pero como lo hemos estructurado en un entorno sin internet no se pueden hacer las copias fuera del local.

Su única configuración es por variables de entorno, le ponemos el “PUID” y el “PGID” del usuario que en este caso el 0 es el “root”. La zona horaria donde trabaja, el tipo de conexión a la base de datos y los datos para la conexión.

Se puede acceder usando la URL: <http://duplicati.insviladegracia.cat:8200>


```
environment:
  - PUID=0
  - PGID=0
  - TZ=EUROPE/MADRID
  - DB_TYPE=mysql
  - DB_CONNECTION_STRING=Server=db;Port=3306;Database=duplicati;Uid=admin;Pwd=12345aA;
```

Los volúmenes que tiene son para ficheros de configuración de copias de seguridad, la carpeta de los backups y la carpeta de donde las hará, en este caso toda la carpeta del proyecto, así se hará una copia de todos los contenedores.

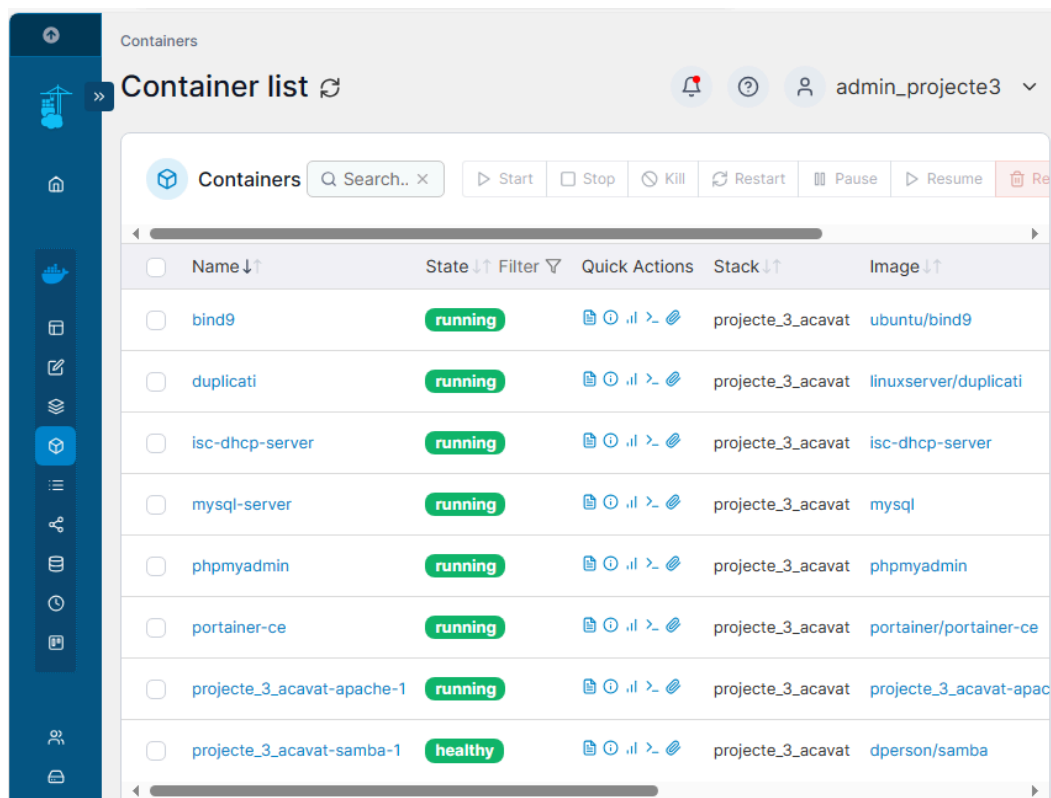
```
volumes:
  - ./duplicati:/config
  - ./duplicati_backups:/backups
  - /home/bernat/PROJECTE_3_ACAVAT:/source
```


PORTAINER:



portainer/portainer-ce  Verified Publisher ☆ 2.3K
By [Portainer.io](#) · Updated 11 days ago
Portainer CE - a lightweight service delivery platform for containerized applications
[IMAGE](#)

Este servicio está compuesto por la imagen de portainer, portainer es una herramienta grafica de contenedores de docker.



Hemos configurado para que vaya por el puerto 9000, tanto en host como en contenedor, hemos configurado también dos volúmenes persistentes para sus datos, y que si se eliminase el contenedor no se eliminen todos los datos.

```
ports:
  - 9000:9000
volumes:
  - /var/run/docker.sock:/var/run/docker.sock
  - portainer_data:/data
```

Accedemos al portal web con la URL: <http://portainer.insviladegracia.cat:9000>

BASE DE DATOS:

Dentro del docker-compose.yml hemos especificado el usuario y password de la base de datos, en archivo verás esta sección “db” aquí podremos encontrar estas líneas de código, las cuales especifican el usuario y password:

```
environment:
  MYSQL_ROOT_PASSWORD: 12345aA
  MYSQL_DATABASE: insviladegracia
  MYSQL_USER: admin
  MYSQL_PASSWORD: 12345aA
```

En el script de la base de Datos ya hemos añadido el pedido con el que damos permisos al usuario en la base de datos.

```
CREATE USER 'admin'@'%' IDENTIFIED BY '12345aA';
GRANT ALL PRIVILEGES ON *.* TO 'admin'@'%';
ALTER USER 'admin'@'%' IDENTIFIED BY '12345aA';
COMMIT;
```

Aquí podemos ver cómo el usuario “admin” adquiere todos los privilegios sobre todas las bases de datos, incluida la nuestra con el nombre “insviladegracia”. Es importante tener claro esto ya después a la hora de realizar la conexión entre apache2 con php será obligatorio saber el usuario y password.

Dentro del Docker-compose.yml podemos localizar la siguiente línea:

```
volumes:
  - ./sql_database/insviladegracia.sql:/docker-entrypoint-initdb.d/insviladegracia.sql
  - dbdata:/var/lib/mysql
```

Aquí subimos el archivo insviladegracia.sql a la carpeta “/docker-entrypoint-initdb.d”, esto hace que en cargar el contenedor, Mysql lee el contenido de esta carpeta y se encuentre el script “insviladegracia.sql” y lo ejecute por la creación de las tablas, insertos y permisos usuarios.

phpMyAdmin:

Para poder administrar la base de datos de forma fácil e intuitiva, hemos seleccionado instalar el servicio de “phpmyadmin”, por eso hemos utilizado la imagen ya creada y oficial que se puede encontrar a Docker Hub:

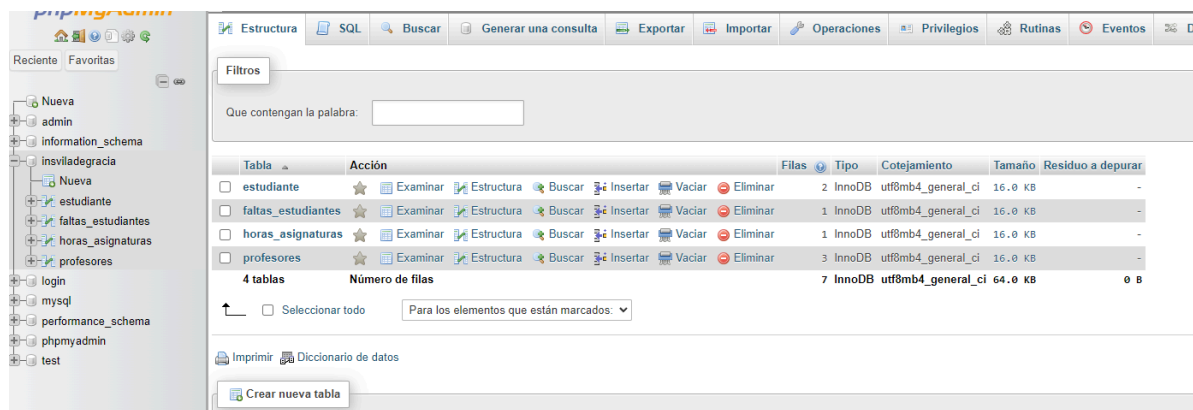


Dentro de docker-compose.yml hemos utilizado esta imagen para crear el contenedor con el que después podemos conectarnos vía página web con interfaz gráfica. Es importante remarcar que como vemos la etiqueta "links" hace referencia a que tiene un link con el contenedor "db" que es la base de datos, ya que así podremos comunicar el contenedor phpMyAdmin con la base de datos, y también remarcar que la etiqueta "depends_on" lo que hace es que hasta que no haya creado el contenedor "db" el phpMyAdmin no se creará, así evitamos errores.

```
phpmyadmin:
  image: phpmyadmin
  container_name: phpmyadmin
  hostname: phpmyadmin
  links:
    - db
  environment:
    PMA_HOST: db
    PMA_PORT: 3306
    PMA_ARBITRARY: 1
  depends_on:
    - db
  ports:
    - 8081:80
  volumes:
    - phpmyadmin_data:/hola
  networks:
    macvlan1:
      ipv4_address: 192.168.10.203
  restart: unless-stopped
```

Si los contenedores se han cargado correctamente verá la siguiente página en el navegador.

<http://phpmyadmin.insviladegracia.cat>







Aquí podemos administrar de forma “administrador” la base de datos, tablas y datos que se encuentren, así como “insertos” que desee hacer, “updates” y “delete”.

Apache:

Apache lo hemos construido manualmente, ya que con la imagen de Dockerhub teníamos problemas, entonces lo que hicimos fue crear un Dockerfile con las órdenes para crear un contenedor con el apache.

```
apache:
  build: './apache_docker/'
  hostname: apache
  #depends_on:
  #- php
  ports:
    - "443:443"
    - "80:80"
  volumes:
    - ./www:/var/www/html/
  networks:
    macvlan1:
      ipv4_address: 192.168.10.204
  restart: unless-stopped
```

Como vemos el trozo de docker-compose de apache lo que hemos hecho es hacer una referencia a una carpeta llamada “apache_docker” que dentro tiene un Dockerfile con la imagen a medida de todo lo que queremos integrar. Dentro de esta carpeta tenemos los siguientes archivos:

	hola.css	09/05/2024 12:51	Documento de ho...	228 KB
	index.html	09/05/2024 12:51	Chrome HTML Do...	5 KB
	Assistencia	09/05/2024 12:51	Carpeta de archivos	
	img	09/05/2024 12:51	Carpeta de archivos	

DockerFile:

```
C: > Users > sergi > Downloads > www correcto > proyecto3 > proyecto3 > apache_docker > Dockerfile
1  # Utilizamos una imagen base con soporte para Apache y PHP
2  FROM php:7.4-apache
3
4  # Actualizamos la lista de paquetes e instalamos herramientas necesarias
5  RUN apt-get update
6  RUN apt-get install openssl -y
7  RUN docker-php-ext-install mysqli
8  #RUN apt-get install php8.1 -y
9  #RUN apt-get install php-mysql -y
10 #RUN apt-get install php-mysqli -y
11
12 # Creamos el directorio para almacenar los archivos SSL
13 RUN mkdir -p /etc/apache2/ssl
14
15 # Generamos el certificado y la clave
16 RUN openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/apache2/ssl/apache.key -out /etc/apache2/ssl/apache.crt -subj "/C=ES/ST=Bar
17
18 # Habilitamos los módulos de Apache necesarios
19 RUN a2enmod ssl
20 RUN a2enmod headers
21
22 # Copiamos la configuración del sitio SSL
23 COPY apache_ssl.conf /etc/apache2/sites-available/default-ssl.conf
24 RUN a2ensite default-ssl
25
26 # Copiamos la configuración del sitio HTTP
27 COPY apache_http.conf /etc/apache2/sites-available/000-default.conf
28
```

Aquí lo que podemos ver es que hemos cogido la imagen de Apache, los "RUN" que tenemos es para actualizar las dependencias y después instalar el OpenSSL y una extensión que necesitamos para poder comunicarse con la base de datos. Después creamos unas carpetas o irá el certificado y la clave privada. Activamos los módulos de SSL con el comando "a2enmod ssl" y los headers con el comando "a2enmod headers". En continuación copiamos los archivos que tenemos de configuración de HTTP y de HTTPS y activamos el "site" de SSL porque por defecto no viene habilitada, y también copiar el archivo de configuración de HTTP en la ruta especificada.

A continuación los dos archivos de configuración de HTTP y HTTPS:

Apache_ssl.conf:

```
C: > Users > sergi > Downloads > www correcto > proyecto3 > proyecto3 > apache_docker > ⚙️ apache_ssl.conf
1  <IfModule mod_ssl.c>
2      <VirtualHost *:443>
3          ServerAdmin btpsee@localhost
4          DocumentRoot /var/www/html
5
6          ErrorLog ${APACHE_LOG_DIR}/error.log
7          CustomLog ${APACHE_LOG_DIR}/access.log combined
8
9          SSLEngine on
10         SSLCertificateFile /etc/apache2/ssl/apache.crt
11         SSLCertificateKeyFile /etc/apache2/ssl/apache.key
12
13         <FilesMatch "\.(cgi|shtml|phtml|php)$">
14             SSLOptions +StdEnvVars
15         </FilesMatch>
16
17         <Directory /var/www/html>
18             Options Indexes FollowSymLinks
19             AllowOverride All
20             Require all granted
21         </Directory>
22
23     </VirtualHost>
24 </IfModule>
25
```

Aquí lo que vemos es el archivo de configuración de HTTPS, lo que cambia de HTTP es el puerto que en vez de 80 es el 443 y también la línea "SSLEngine on", "SSCertificateFile" y la "SSLCertificateKeyFile" que aquí debemos poner la ruta donde tenemos almacenada estos dos archivos, que en Dockerfile hemos visto que hemos creado el certificado y la clave privada en la carpeta "/etc/apache2/ssl".

Y el archivo de configuración de HTTP es el mismo que el de arriba, pero quitando las líneas citadas en HTTPS.

apache_http.conf:

```
C: > Users > sergi > Downloads > www correcto > proyecto3 > proyecto3 > apache_docker > ⚙️ apache_http.conf
1  <VirtualHost *:80>
2      ServerAdmin btpsee@localhost
3      DocumentRoot /var/www/html
4
5      ErrorLog ${APACHE_LOG_DIR}/error.log
6      CustomLog ${APACHE_LOG_DIR}/access.log combined
7
8      <Directory /var/www/html>
9          Options Indexes FollowSymLinks
10         AllowOverride All
11         Require all granted
12     </Directory>
13 </VirtualHost>
14
```

Aquí sólo hacemos referencia a "ServerAdmin" que es el contacto para decirlo de algún modo, el "DocumentRoot" que está en qué carpeta trabaja este archivo de configuración y el puerto con el que trabaja que es el 80 (HTTP).

RED DOCKER:

Por último en el docker-compose tenemos una sección de red, para hacer que este proyecto se asemeje lo más posible a la realidad lo hemos adaptado para que cada contenedor tenga una IP propia, como si fuera un host más dentro de nuestra red, para poder hacer esto hemos tenido que investigar cómo se podía hacer. Entre muchas búsquedas y consultas encontramos los "drivers" de Docker, que es el tipo de enlace de red que usaras, por defecto en Docker todo se ejecuta con "bridge" que es puente usa la misma IP que el host que ha arrancado los contenedores y se conecta a través de puertos, pero nuestro caso no era la mejor opción para asemejarse a la realidad, descubrimos el "driver macvlan" este driver hace que tengas que poner una subred donde trabajaran los contenedores de Docker, y así poder tener una IP fija todos los contenedores.

```
# RED PERSONALIZADA PARA TODOS LOS CONTENEDORES #
networks:
  macvlan1:
    driver: macvlan
    driver_opts:
      parent: enp2s0
    ipam:
      config:
        - subnet: 192.168.10.0/24
```

Para configurarlo tenemos que especificar el nombre de la subred, el "driver", porque interficie pasará, y la configuración de la subred.

Puesta en marcha de el docker-compose

```

bernat@srvdocker:~/PROJECTE_3_ACAVAT$ sudo docker-compose up -d
[sudo] password for bernat:
[+] Running 8/8
✓ Container projecte_3_acavat-samba-1    Running      0.05s
✓ Container duplicati                    Started     3.35s
✓ Container mysql-server                 Running      0.05s
✓ Container isc-dhcp-server              Running      0.05s
✓ Container portainer-ce                 Running      0.05s
✓ Container projecte_3_acavat-apache-1   Running      0.05s
✓ Container bind9                         Running      0.05s
✓ Container phpmyadmin                   Running      0.05s
bernat@srvdocker:~/PROJECTE_3_ACAVAT$

```

Verificación del “status” de los contenedores

```

bernat@srvdocker:~/PROJECTE_3_ACAVAT$ docker ps -a
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS              PORTS          NAMES
b1c795d68615   linuxserver/duplicati              "/init"                 24 seconds ago Up 20 seconds      0.0.0.0:80->80   duplicati
aaf661ad69ae   phpmyadmin                         "/docker-entrypoint..." 2 hours ago    Up 2 hours          0.0.0.0:3306->3306 phpmyadmin
2b0c030ee9ee   ubuntu/bind9                       "/docker-entrypoint.s..." 2 hours ago    Up 2 hours          0.0.0.0:53->53   bind9
148ac813c4ed   projecte_3_acavat-apache           "/docker-php-entrypoi..." 2 hours ago    Up 2 hours          0.0.0.0:80->80   projecte_3_acavat-apache-1
c85f30089d83   dperson/samba                     "/sbin/tini -- /usr/_..." 2 hours ago    Up 2 hours (healthy) 0.0.0.0:139->139 projecte_3_acavat-samba-1
64dcde34c434   isc-dhcp-server                    "/data/entrypoint.sh"    2 hours ago    Up 2 hours          0.0.0.0:67->67   isc-dhcp-server
e5c21909e00a   portainer/portainer-ce            "/portainer"            2 hours ago    Up 2 hours          9000-9001/tcp    portainer-ce
6a7ebfcd805    mysql                               "/docker-entrypoint.s..." 2 hours ago    Up 2 hours          0.0.0.0:3306->3306 mysql-server
bernat@srvdocker:~/PROJECTE_3_ACAVAT$

```

Volúmenes de nuestro docker-compose

```

bernat@srvdocker:~/PROJECTE_3_ACAVAT$ docker volume ls
DRIVER      VOLUME NAME
local       1f36789c69478d39e2ec0285c762aaed4f0d683f6c5e5535708bf65a7cba9af2
local       01abb4ec31ee5ed2d19f9e51c7b49dd3cead31f34d8cc8f584daa1782c9081b4
local       3fcbe2a2e51d441f6fc398c4736eb019457e78c75c3b3bc0e7ada95692f7b1be
local       24e28ae35d7bd7192a519008f2dcae1381a3961ecd21ae1e6b0a1bfacadfc6fa
local       99141feaa30270d3dd69b03d6b0d41ae7a40093c7c7e3701da6c437c3ea3c058
local       a2f047872d58b46balcce4a1ba16f47c729adc5a25bd6439d950a8456d0c09c9
local       d303f5430de143eedf1e5e854c08026b030d0b555bfbed2f9ca5109e9649a147
local       portainer_data
local       projecte_3_acavat_dbdata
local       projecte_3_acavat_phpmyadmin_data
local       projecte_3_acavat_portainer_data
local       proyecto3_dbdata
local       proyecto3_phpmyadmin_data

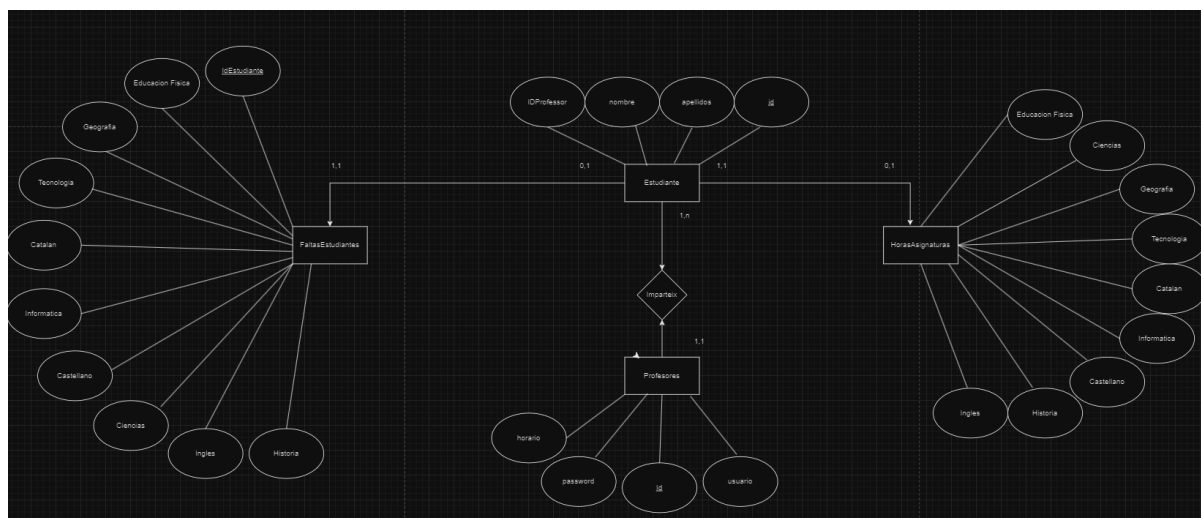
```


WEB

Base de datos:

Para el correcto funcionamiento de nuestra página web, necesitamos vincular el contenedor de mySQL, para poder almacenar la información tanto para el login o datos de usuarios, En nuestra página tenemos la página web oficial, y después una página de asistencia con la cual los profesores se podrán iniciar sesión para poder llevar a cabo un control de la asistencia de los alumnos en las diferentes asignatura.

Diagrama ER:



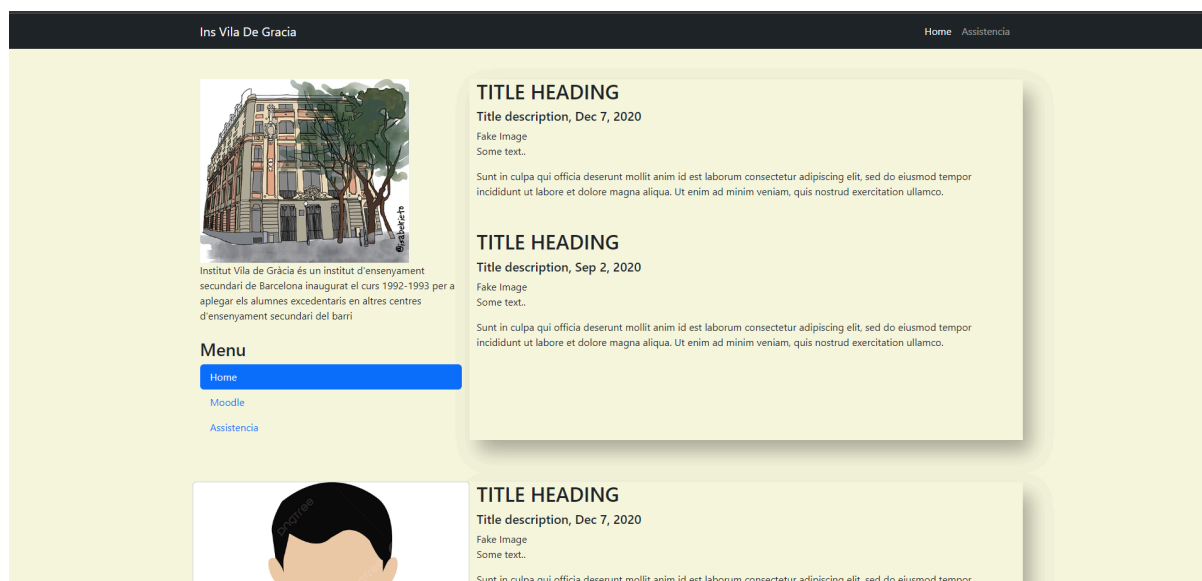
MR:

- FaltaEstudiantes (idEstudiante, educacion_fisica, geografia, tecnologia, catalan, informatica, castellano, ciencias, inglés, historia)
- Profesores (usuario, ID, password, horario)
- HorasAsignaturas (educacion_fisica, geografia, tecnologia, catalan, informatica, castellano, ciencias, inglés, historia)
- Estudiantes(IDProfessor, nombre, apellido, ID)

Pàgina web:

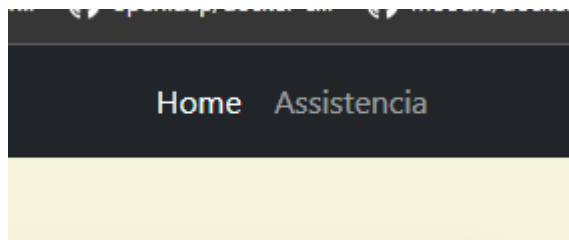
Web Principal Ins vila de gràcia:

Esta página está desarrollada en lenguaje de etiquetas html y css junto a bootstrap, y también utiliza un lenguaje servidor php, para poder realizar operaciones junto a la base de datos.



Assistència:

Para la página asistencia nos hemos basado en iEduca, una aplicación web para llevar el control y asistencia de los alumnos, en la barra de navegación podrás encontrar un link desde la propia pagina principal que te enviará a la página login de nuestro “iEduca”.



****Dentro primero nos pedirá iniciar sesión con un usuario profesor.**

Una vez iniciada la sesión nos mostrará una tabla con el horario y las respectivas asignaturas, toda esta información se carga a partir de archivos .json, para realizar más fácil la integración de asignaturas y horarios.

Web:

Horario

Hora	Clase
15:00 AM	Matematicas
16:00 AM	Ciencias
17:00 AM	Historia
18:00 AM	Ingles
19:00 PM	Catalan
20:00 PM	EE

Archivo .json

```
{
  "15:00 AM": "Matematicas",
  "16:00 AM": "Ciencias",
  "17:00 AM": "Historia",
  "18:00 AM": "Ingles",
  "19:00 PM": "Catalan",
  "20:00 PM": "EE"
}
```

Si entramos dentro de alguna asignatura podremos encontrar otra tabla pero esta vez de alumnos, y a su lado una casilla seleccionable “check-box”.

Asistencia para Matematicas

Nombre del Estudiante	Asistencia
sergi	<input checked="" type="checkbox"/>
sergi	<input checked="" type="checkbox"/>

**Si la casilla está marcada significa que el estudiante no ha venido.

Una vez realizada la asistencia nos podemos dirigir a la página “media.php” donde podremos ver una tabla con los estudiantes de cada profesor y sus faltas en cada asignatura, en caso de que supere el 20% de las faltas el recuadro se pintara de rojo para poder identificar de forma mas rapida que el estudiante no ha logrado el porcentaje de asistencia requerido.

Valores de Materias y Faltas de Estudiantes

Valores de Materias

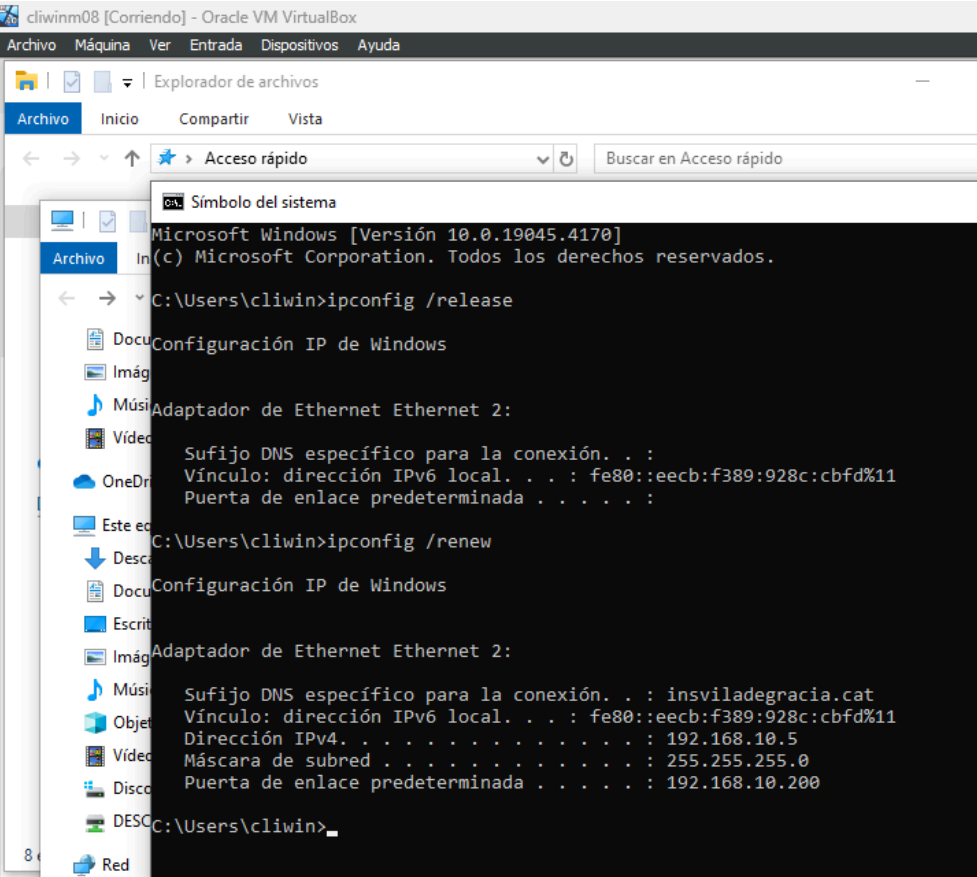
Materia	Valor
Historia	10
Ingles	10
Ciencias	10
Castellano	10
Informatica	10
Catalan	10
Tecnologia	10
Geografia	10
EF	10

Faltas de Estudiantes en Asignaturas

ID Estudiante	Nombre Estudiante	Historia	Ingles	Ciencias	Castellano	Informatica	Catalan	Tecnologia	Geografia	EF
1	sergi	14	2	15	0	0	4	0	0	4
2	bernat	0	0	0	0	0	0	0	0	0
3	sergi	1	1	0	0	0	1	0	0	0

PRUEBAS DEL PROYECTO

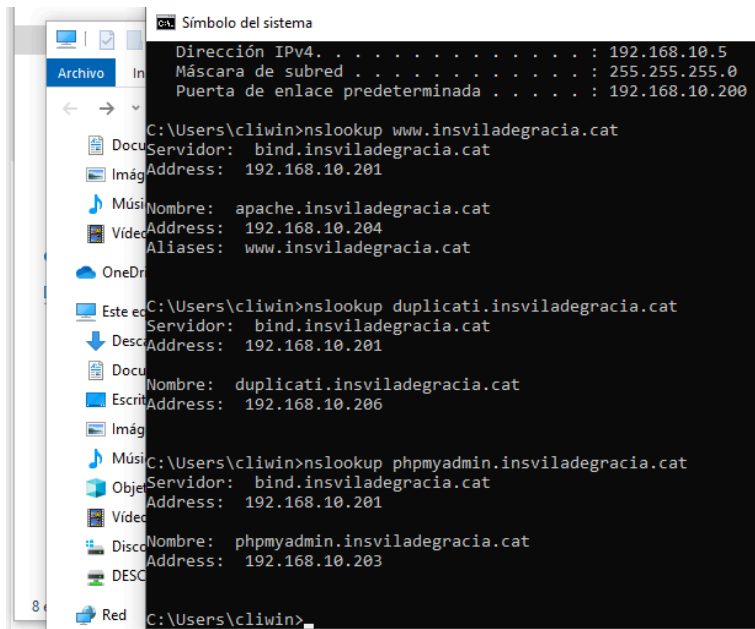
DHCP:



```
cliwinm08 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Explorador de archivos
Archivo Inicio Compartir Vista
Acceso rápido Buscar en Acceso rápido
Símbolo del sistema
Microsoft Windows [Versión 10.0.19045.4170]
(c) Microsoft Corporation. Todos los derechos reservados.
C:\Users\cliwin>ipconfig /release
Configuración IP de Windows
Adaptador de Ethernet Ethernet 2:
    Sufijo DNS específico para la conexión. . . :
    Vínculo: dirección IPv6 local. . . . : fe80::eeb:f389:928c:cbfd%11
    Puerta de enlace predeterminada . . . . . :
C:\Users\cliwin>ipconfig /renew
Configuración IP de Windows
Adaptador de Ethernet Ethernet 2:
    Sufijo DNS específico para la conexión. . . : insviladegracia.cat
    Vínculo: dirección IPv6 local. . . . : fe80::eeb:f389:928c:cbfd%11
    Dirección IPv4. . . . . : 192.168.10.5
    Máscara de subred . . . . . : 255.255.255.0
    Puerta de enlace predeterminada . . . . . : 192.168.10.200
C:\Users\cliwin>
```

Como vemos cuando hacemos un `ipconfig /release` y un `ipconfig /renew`, se nos pone una ip asignada por el dhcp y también nos muestra el DNS de la red que es `insviladegracia.cat`.

DNS:



```

C:\Users\cliwin>nslookup www.insviladegracia.cat
Servidor: bind.insviladegracia.cat
Address: 192.168.10.201

Nombre: apache.insviladegracia.cat
Address: 192.168.10.204
Alias: www.insviladegracia.cat

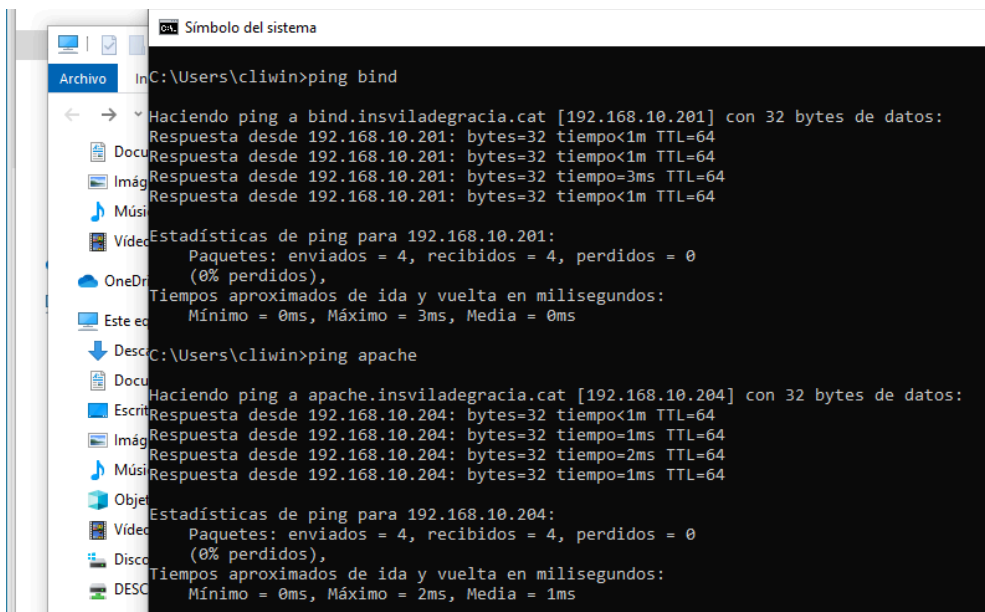
C:\Users\cliwin>nslookup duplicati.insviladegracia.cat
Servidor: bind.insviladegracia.cat
Address: 192.168.10.201

Nombre: duplicati.insviladegracia.cat
Address: 192.168.10.206

C:\Users\cliwin>nslookup phpmyadmin.insviladegracia.cat
Servidor: bind.insviladegracia.cat
Address: 192.168.10.201

Nombre: phpmyadmin.insviladegracia.cat
Address: 192.168.10.203

C:\Users\cliwin>
  
```



```

C:\Users\cliwin>ping bind

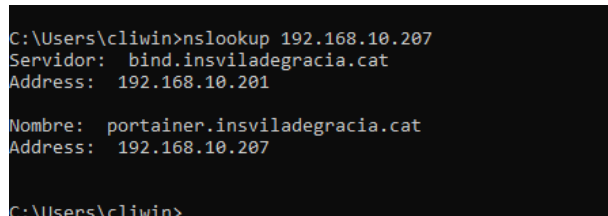
Haciendo ping a bind.insviladegracia.cat [192.168.10.201] con 32 bytes de datos:
Respuesta desde 192.168.10.201: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.10.201: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.10.201: bytes=32 tiempo=3ms TTL=64
Respuesta desde 192.168.10.201: bytes=32 tiempo<1m TTL=64

Estadísticas de ping para 192.168.10.201:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 3ms, Media = 0ms

C:\Users\cliwin>ping apache

Haciendo ping a apache.insviladegracia.cat [192.168.10.204] con 32 bytes de datos:
Respuesta desde 192.168.10.204: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.10.204: bytes=32 tiempo=1ms TTL=64
Respuesta desde 192.168.10.204: bytes=32 tiempo=2ms TTL=64
Respuesta desde 192.168.10.204: bytes=32 tiempo=1ms TTL=64

Estadísticas de ping para 192.168.10.204:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 2ms, Media = 1ms
  
```



```

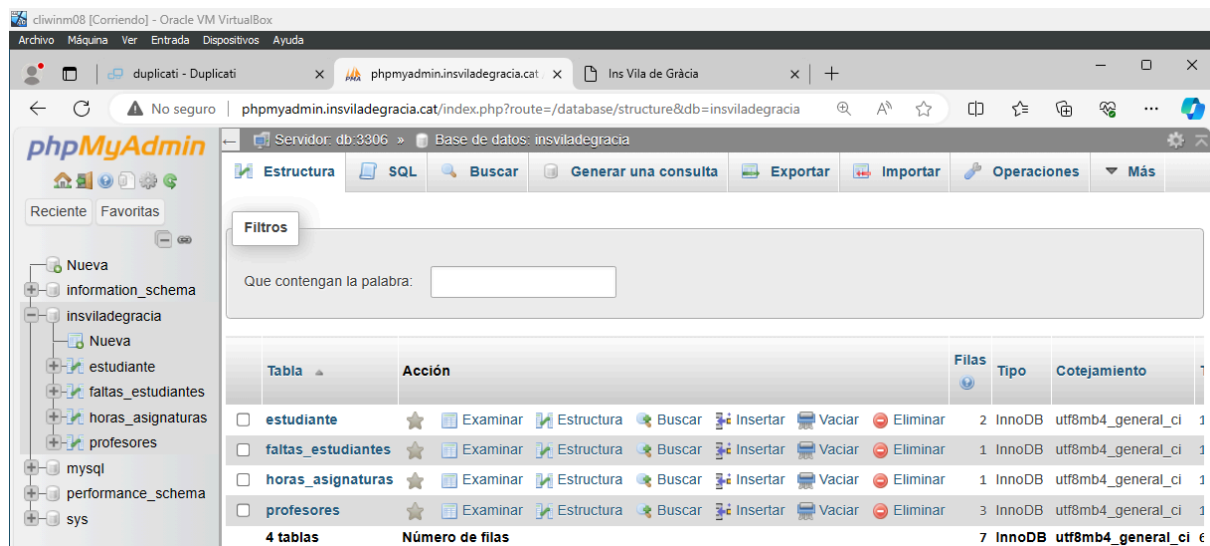
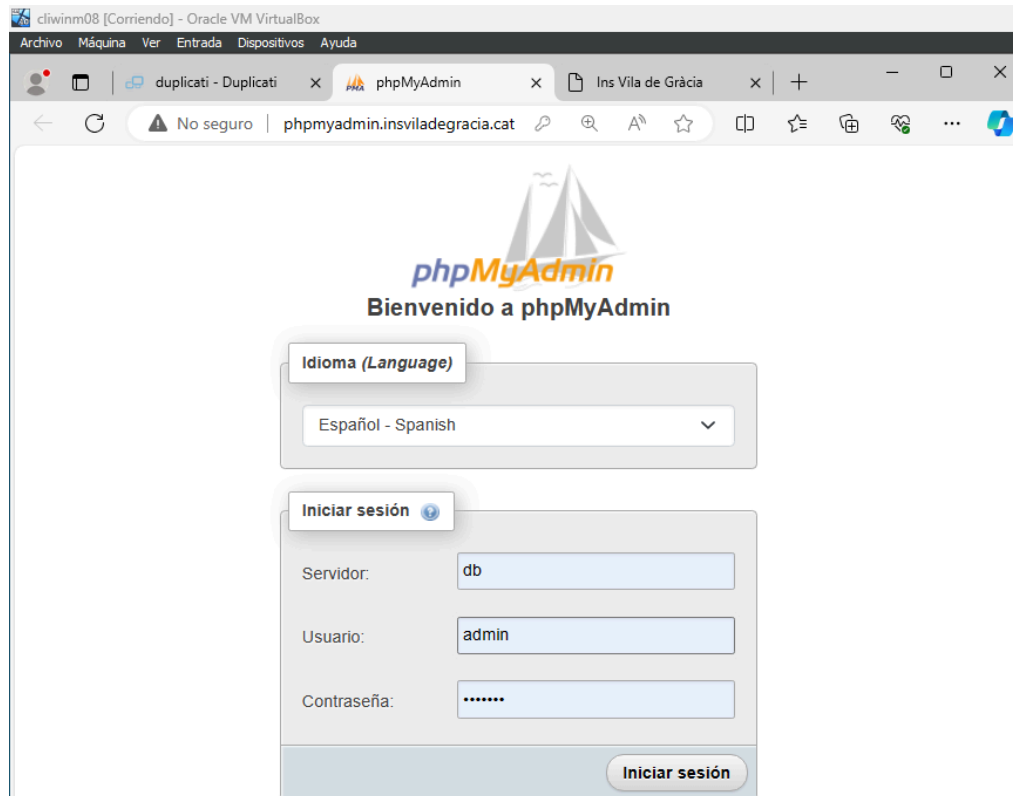
C:\Users\cliwin>nslookup 192.168.10.207
Servidor: bind.insviladegracia.cat
Address: 192.168.10.201

Nombre: portainer.insviladegracia.cat
Address: 192.168.10.207

C:\Users\cliwin>
  
```

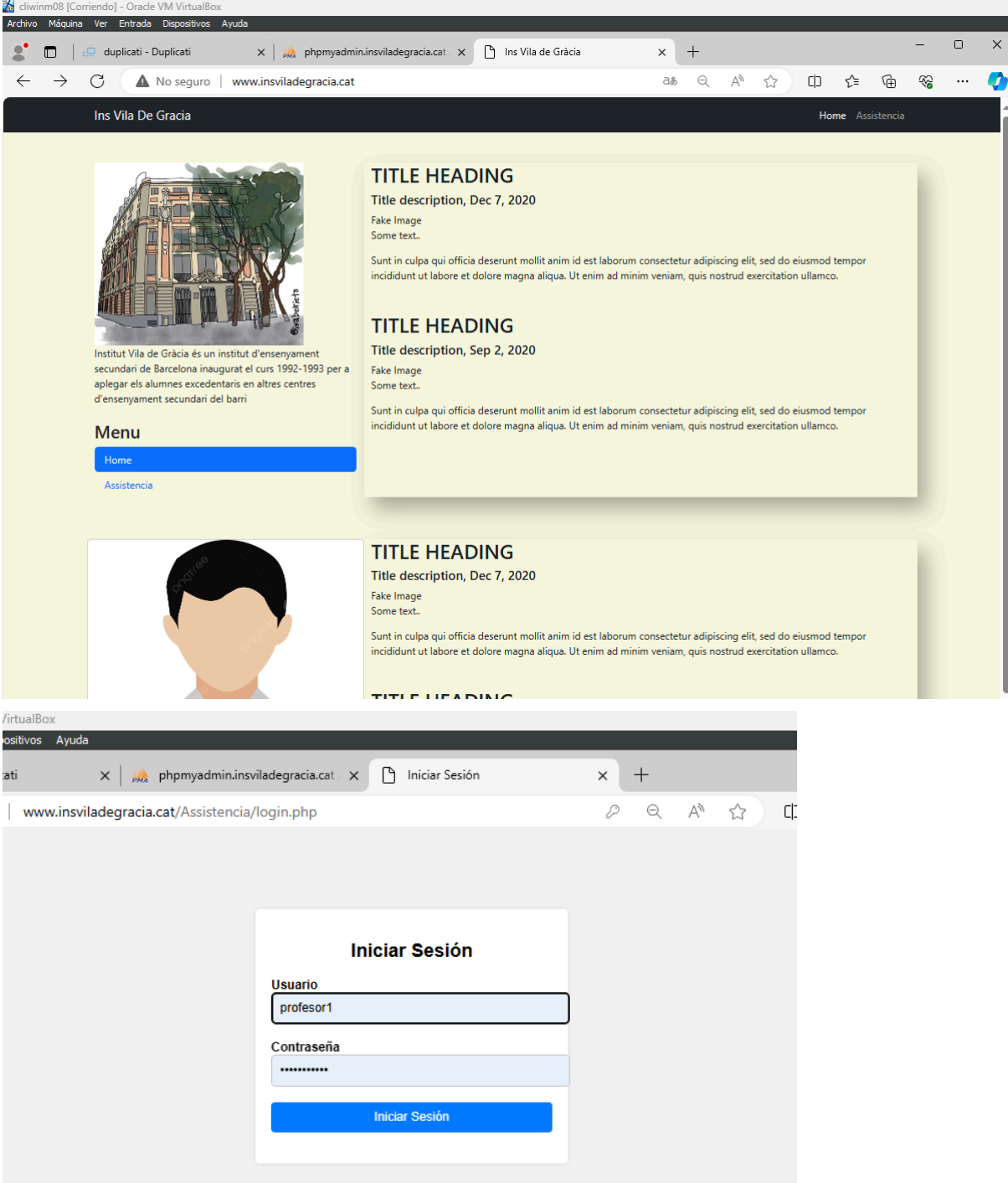
Como vemos la resolución de nombres es resolutive y haciendo ping a cada servidor también.

PHPMYADMIN:



La base de datos también nos carga perfectamente con los datos pasados.

APACHE:



The image shows two screenshots of a web application running in a virtual machine (Oracle VM VirtualBox).

The top screenshot displays the main page of the "Ins Vila De Gràcia" website. The browser address bar shows "www.insviladegracia.cat". The page features a header with "Home" and "Assistència" links. The main content area includes a large image of a building, a "Menu" section with "Home" and "Assistència" links, and three content blocks, each with a "TITLE HEADING", a "Title description", and a "Fake Image".

The bottom screenshot shows the login page of the website, accessed via "www.insviladegracia.cat/Assistència/login.php". The page has a "Iniciar Sesión" button and a form with fields for "Usuario" (username) and "Contraseña" (password).

La web del centro está activa y la página de “IEduca” que hemos creado también nos carga perfectamente.

Vamos a probar que la asistencia del alumnado también se almacene en la base de datos.

Horario del profesor profesor1: horario.json: 1

Horario

Hora	Clase
15:00 AM	Ciencias
16:00 AM	Ciencias
17:00 AM	Historia
18:00 AM	Ingles
19:00 PM	Catalan
20:00 PM	EF

[Faltas](#)

Valores de Materias y Faltas de Estudiantes

Valores de Materias

Materia	Valor
Historia	10
Ingles	10
Ciencias	10
Castellano	10
Informatica	10
Catalan	10
Tecnologia	10
Geografia	10
EF	10

Faltas de Estudiantes en Asignaturas

ID Estudiante	Nombre Estudiante	Historia	Ingles	Ciencias	Castellano	Informatica	Catalan	Tecnologia	Geografia	EF
1	sergi	0	0	0	0	0	2	0	0	1
2	bernat	0	0	0	0	0	0	0	0	0

[Faltas](#)

Como vemos, Sergi tiene 1 falta en "EF".

Si le damos a la hora de las “20:00 PM”.

Asistencia para EF

Nombre del Estudiante	Asistencia
sergi	<input checked="" type="checkbox"/>
<input type="button" value="Registrar Asistencia"/>	

Si le damos a que ha faltado otra vez, veremos como a sergi le queda suspendida la asignatura.

Faltas de Estudiantes en Asignaturas

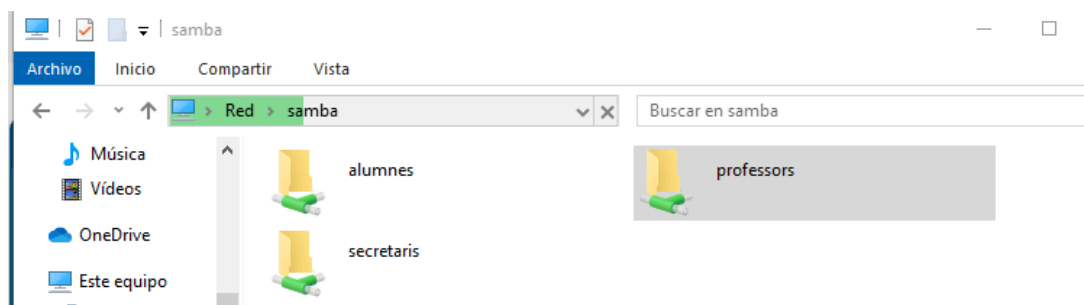
ID Estudiante	Nombre Estudiante	Historia	Inglés	Ciencias	Castellano	Informática	Catalán	Tecnología	Geografía	EF
1	sergi	0	0	0	0	0	2	0	0	2
2	bernat	0	0	0	0	0	0	0	0	0

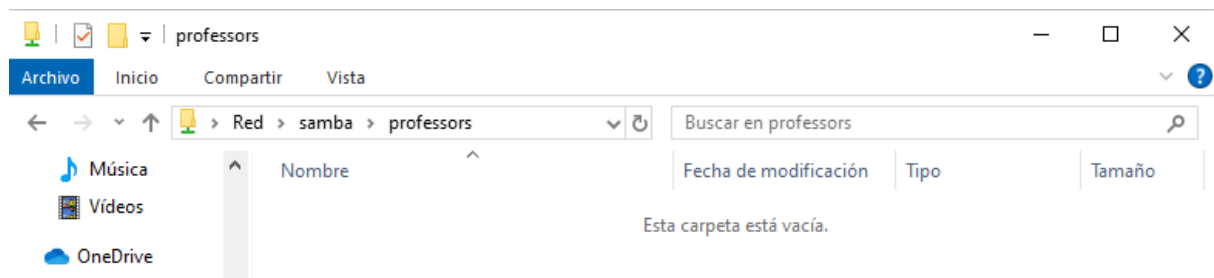
[Faltas](#)

SAMBA:

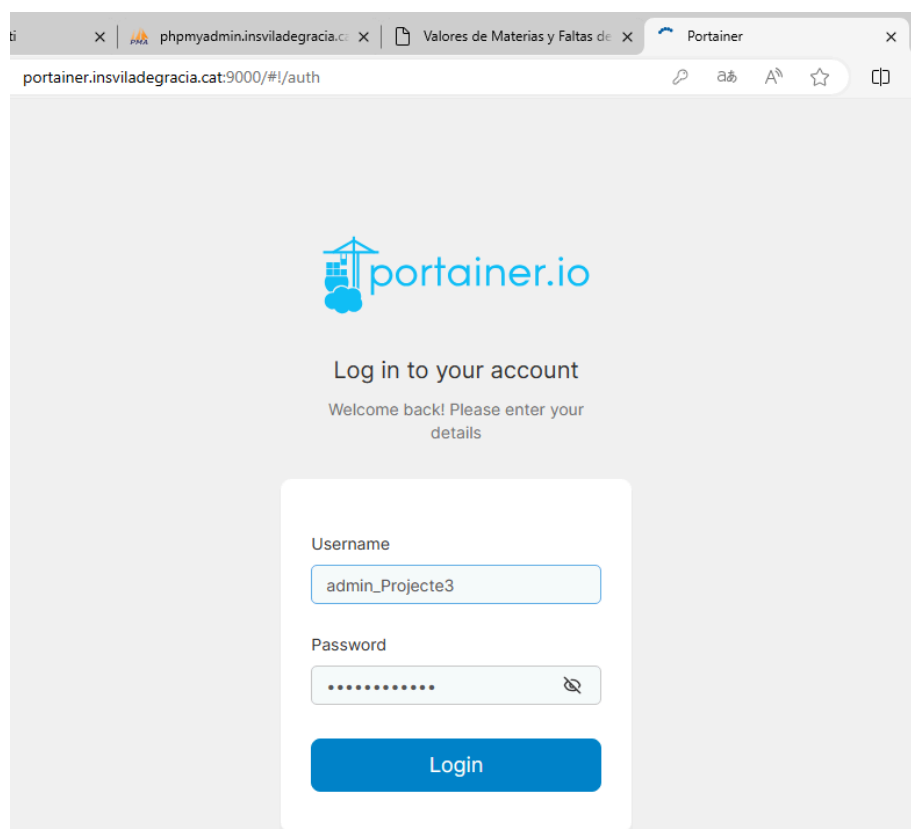
Ejecutamos el comando para iniciar dentro del contenedor, y ejecutamos el script de creación de usuarios, grupos y usuario de samba

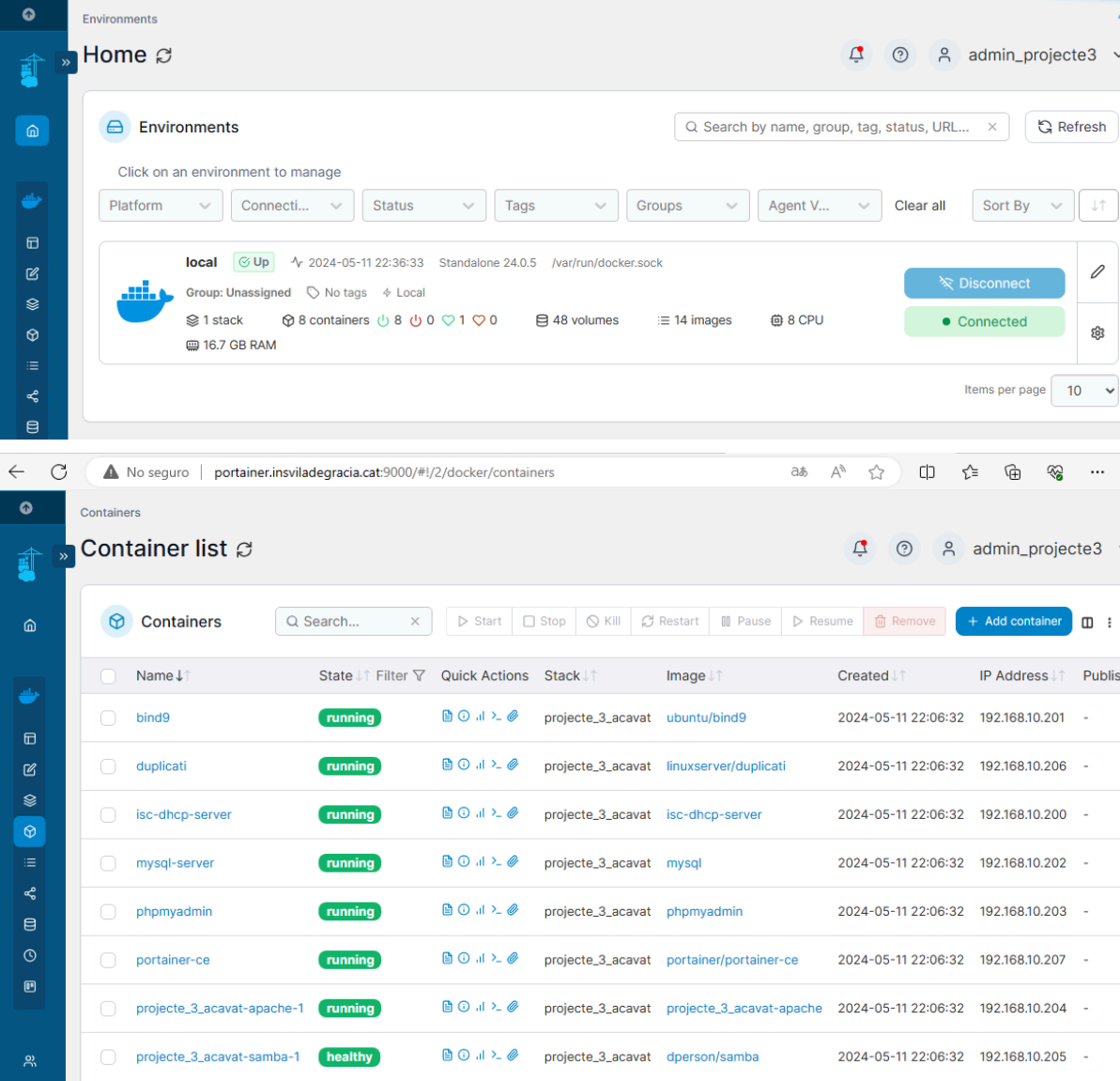
```
bernat@srvdocker:~/PROJECTE_3_ACAVAT$ sudo docker exec -it projecte_3_acavat-samba-1 bash
bash-5.0# smbpasswd -a secretario -s < "secretario"
bash: secretario: No such file or directory
bash-5.0# bash usuarios.sh
Comenzando script de usuarios...
Verificando grupos...
Grupos creados!
Verificando usuarios...
Usuarios y contraseñas creados.
Usuarios y contraseñas SAMBA creados.
bash-5.0#
```





PORTAINER:



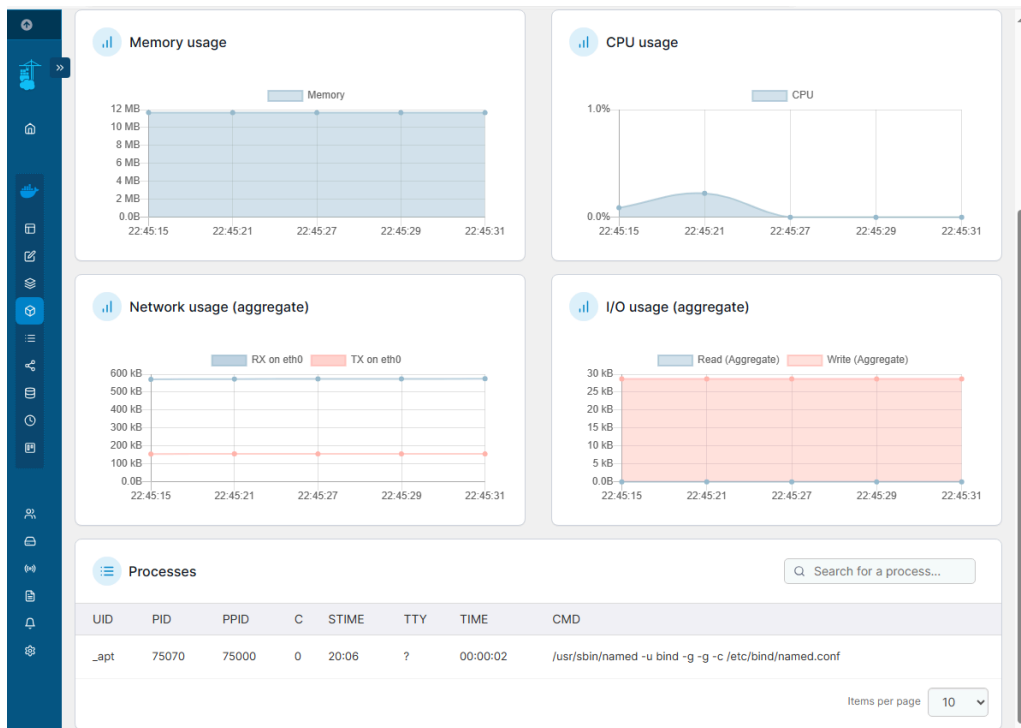


The screenshot displays the Portainer web interface. The top section, 'Environments', shows a single environment named 'local' with a status of 'Up'. It provides details such as the platform (Standalone 24.0.5), connection path (/var/run/docker.sock), and resource usage (1 stack, 8 containers, 8 CPU, 16.7 GB RAM). The bottom section, 'Container list', shows a table of running containers.

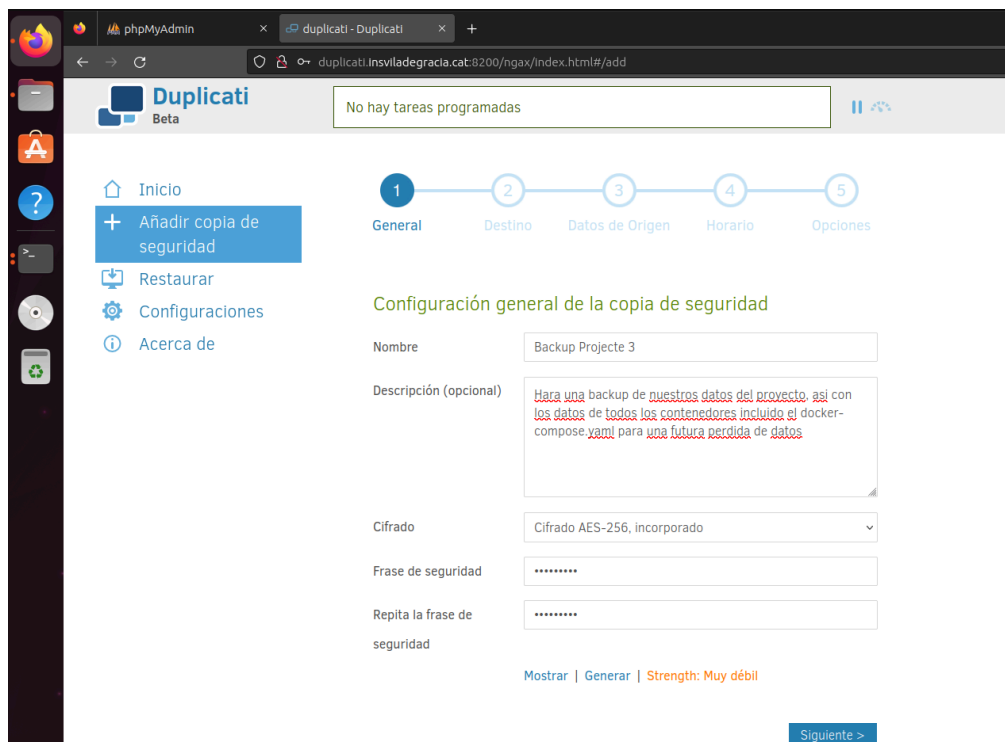
Name	State	Quick Actions	Stack	Image	Created	IP Address	Publish
bind9	running	[Icons]	projecte_3_acavat	ubuntu/bind9	2024-05-11 22:06:32	192.168.10.201	-
duplicati	running	[Icons]	projecte_3_acavat	linuxserver/duplicati	2024-05-11 22:06:32	192.168.10.206	-
isc-dhcp-server	running	[Icons]	projecte_3_acavat	isc-dhcp-server	2024-05-11 22:06:32	192.168.10.200	-
mysql-server	running	[Icons]	projecte_3_acavat	mysql	2024-05-11 22:06:32	192.168.10.202	-
phpmyadmin	running	[Icons]	projecte_3_acavat	phpmyadmin	2024-05-11 22:06:32	192.168.10.203	-
portainer-ce	running	[Icons]	projecte_3_acavat	portainer/portainer-ce	2024-05-11 22:06:32	192.168.10.207	-
projecte_3_acavat-apache-1	running	[Icons]	projecte_3_acavat	projecte_3_acavat-apache	2024-05-11 22:06:32	192.168.10.204	-
projecte_3_acavat-samba-1	healthy	[Icons]	projecte_3_acavat	dperson/samba	2024-05-11 22:06:32	192.168.10.205	-

Como vemos tenemos todos los contenedores corriendo, como vemos también vemos las IP de los contenedores y las imágenes que se han usado.

Lo bueno y el porqué hemos escogido esta herramienta de gestión de contenedores web es por las estadísticas también, cada 1 segundo si quieres se refrescan y puedes ver tanto la memoria usada, el uso de CPU, el uso de red y el uso de I/O.



DUPLICATI:



The Duplicati interface shows the following configuration steps:

- General
- Destino
- Datos de Origen
- Horario
- Opciones

Configuración general de la copia de seguridad

Nombre: Backup Projecte 3

Descripción (opcional): Hara una backup de nuestros datos del proyecto, así con los datos de todos los contenedores incluido el docker-compose.xami para una futura pérdida de datos

Cifrado: Cifrado AES-256, incorporado

Frase de seguridad:

Repita la frase de seguridad:

Mostrar | Generar | Strength: Muy débil

Siguiente >

Cliente_Proyecto3_UBU [Comiendo] - Oracle VM VirtualBox

Archivo Máquina Ver Entrada Dispositivos Ayuda

Actividades Navegador web Firefox 9 de may 20:27

phpMyAdmin duplicati - Duplicati

duplicati.insviladegracia.cat:8200/nginx/index.html#/add

Duplicati
Beta

No hay tareas programadas

Inicio

+ Añadir copia de seguridad

Restaurar

Configuraciones

Acerca de

1 General 2 Destino 3 Datos de Origen 4 Horario 5 Opciones

Destino de la copia de seguridad

Tipo de Almacenamiento Unidad o carpeta local

Ruta de la carpeta

- Computer
 - app
 - backups
 - bin
 - boot
 - command
 - config
 - defaults
 - dev

Nombre de usuario Nombre de usuario para autenticación opcional

Contraseña Contraseña de autenticación opcional

Conexión de prueba

Opciones avanzadas

Cliente_Proyecto3_UBU [Comiendo] - Oracle VM VirtualBox

Archivo Máquina Ver Entrada Dispositivos Ayuda

Actividades Navegador web Firefox 9 de may 20:29

phpMyAdmin duplicati - Duplicati

duplicati.insviladegracia.cat:8200/nginx/index.html#/add

Duplicati
Beta

No hay tareas programadas

Inicio

+ Añadir copia de seguridad

Restaurar

Configuraciones

Acerca de

1 General 2 Destino 3 Datos de Origen 4 Horario 5 Opciones

Datos de origen

☐ Mostrar carpetas ocultas

- source
 - apache_docker
 - bind-conf
 - bind-zones
 - data
 - dhcp
 - duplicati
 - duplicati_backups
 - home
 - php_docker
 - SAMBA-Alumnos
 - SAMBA-CONFIG
 - SAMBA-Professors
 - SAMBA-Secretaria
 - sql_database
 - www
 - docker-compose.yaml

Agregar el path directamente Añadir ruta

Cliente_Proyecto3_UBU [Corriendo] - Oracle VM VirtualBox

Archivo Máquina Ver Entrada Dispositivos Ayuda

Actividades Navegador web Firefox 9 de may 20:28

Navegador web Firefox duplicati - Duplicati

duplicati.insviladegracia.cat:8200/ngax/index.html#/add

Duplicati
Beta

No hay tareas programadas

Inicio
+ Añadir copia de seguridad
Restaurar
Configuraciones
Acerca de

1 General 2 Destino 3 Datos de Origen 4 **Horario** 5 Opciones

Horario

☒ Ejecutar automáticamente las copias de seguridad.

Si la fecha se paso, se ejecutará el trabajo tan pronto como sea posible.

La próxima vez 22:00 10 / 05 / 2024

Volver a ejecutar cada 1 Días

Días permitidos

☒ Lun
☒ Mar
☒ Mié
☒ Jue
☒ Vie
☒ Sab
☒ Dom

< Anterior Siguiente >

Cliente_Proyecto3_UBU [Corriendo] - Oracle VM VirtualBox

Archivo Máquina Ver Entrada Dispositivos Ayuda

Actividades phpMyAdmin 9 de may 20:29

Navegador web Firefox duplicati - Duplicati

duplicati.insviladegracia.cat:8200/ngax/index.html#/add

Duplicati
Beta

No hay tareas programadas

Inicio
+ Añadir copia de seguridad
Restaurar
Configuraciones
Acerca de

1 General 2 Destino 3 Datos de Origen 4 Horario 5 **Opciones**

Opciones generales

Tamaño de volumen remoto 50 GByte

The backups will be split up into multiple files called volumes. Here you can set the maximum size of the individual volume files. See this page for more information.

Conservación de copia de respaldo Mantener todas las copias de seguridad

Nada será borrado. El tamaño de la copia de seguridad aumentará con cada cambio.

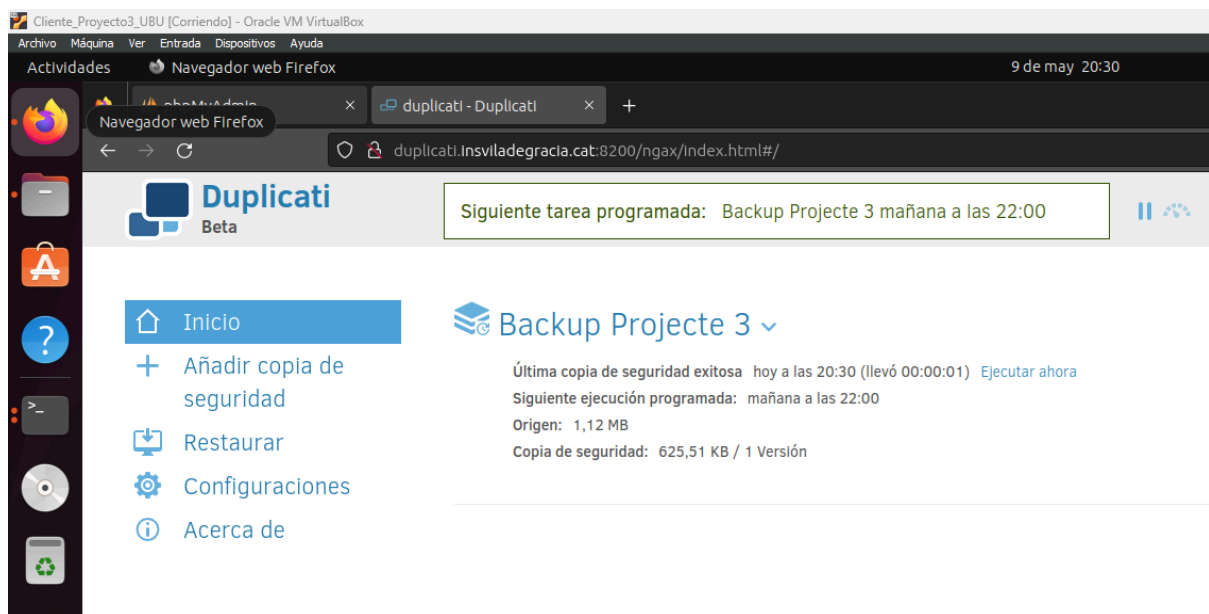
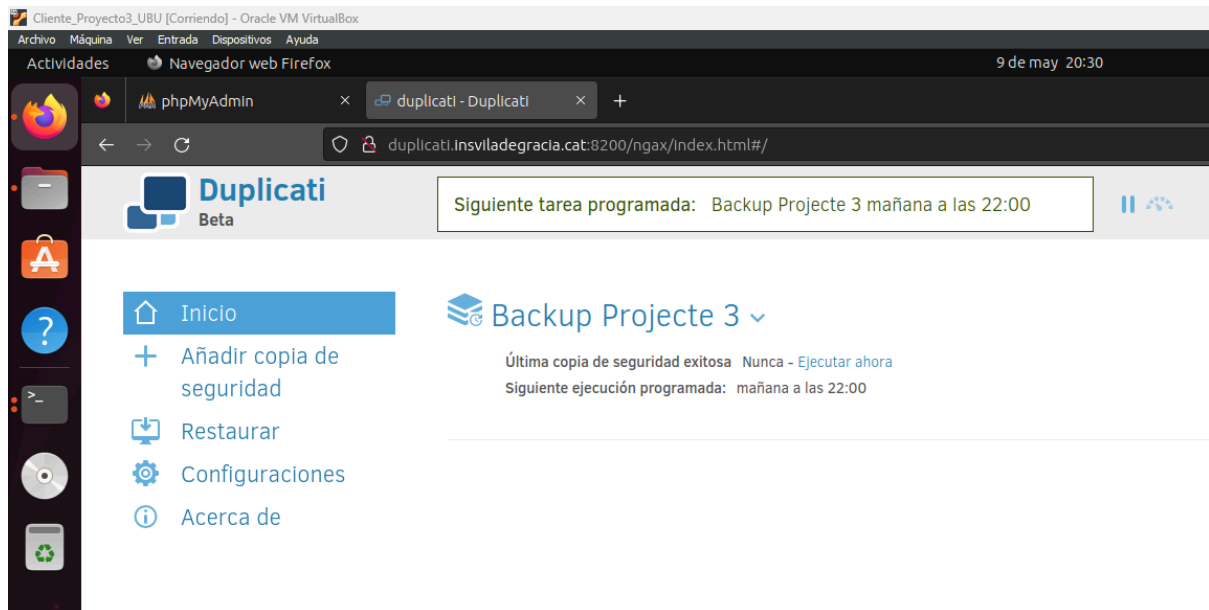
Opciones avanzadas

prefix Backup_Proyecto3V_

A string used to prefix the filenames of the remote volumes, can be used to store multiple backups in the same remote folder. The prefix cannot contain a hyphen (-), but can contain all other characters allowed by the remote storage. Default value: "duplicati"

Añadir opción avanzada - escoja una opción -

< Anterior Guardar



```

bernat@srvdocker:~/proyecto3/proyecto3-v1/proyecto3$ ls -l duplicati_backups/
total 632
-rw-r--r-- 1 root root 5565 may 9 18:30 Backup_Projecte3V_-20240509T183029Z.dlist.zip.aes
-rw-r--r-- 1 root root 628621 may 9 18:30 Backup_Projecte3V_-bfc60e78aa3d54269a77c155d7304fc2d.dblock.zip.aes
-rw-r--r-- 1 root root 6333 may 9 18:30 Backup_Projecte3V_-i33caaccdc06749c0ad993531834494c9.dindex.zip.aes
bernat@srvdocker:~/proyecto3/proyecto3-v1/proyecto3$

```


RAÍZ DEL PROYECTO:

```

bernat@srvdocker:~/PROYECTE_3_ACAVAT$ tree
.
├── apache_docker
│   ├── apache_http.conf
│   ├── apache_ssl.conf
│   └── Dockerfile
├── bind-conf
│   ├── named.conf
│   ├── named.conf.default-zones
│   ├── named.conf.local
│   ├── named.conf.options
│   └── resolv.conf
├── bind-zones
│   ├── db.10.168.192.in-addr.arpa
│   └── db.insviladegracia.cat
├── data
├── dhcp
│   ├── data
│   ├── dhcpd
│   │   ├── dhcpd.conf
│   │   ├── dhcpd.leases
│   │   ├── dhcpd.leases~
│   │   ├── ubuntu.conf
│   │   └── windows.conf
│   ├── dockerrfile
│   └── entrypoint.sh
├── docker-compose.yaml
├── duplicati
│   ├── control_dir_v2
│   │   └── lock_v2
│   ├── Duplicati-server.sqlite
│   ├── JXDQKTXFDV.backup
│   └── JXDQKTXFDV.sqlite
├── duplicati_backups
│   ├── Backup_Projecte3V_-20240509T183029Z.dlist.zip.aes
│   ├── Backup_Projecte3V_-20240509T183126Z.dlist.zip.aes
│   ├── Backup_Projecte3V_-b31af7146177c49b2b6a52d758cd37726.dblock.zip.aes
│   ├── Backup_Projecte3V_-bfc60e78aa3d54269a77c155d7384fc2d.dblock.zip.aes
│   ├── Backup_Projecte3V_-i33caaccdc06749c0ad993531834494c9.dindex.zip.aes
│   └── Backup_Projecte3V_-ib12d9bf18ab644459388e42dd62ee40c.dindex.zip.aes
├── home
│   └── bernat
│       ├── proyecto3
│       │   └── proyecto3-v1
│       │   └── proyecto3
├── php_docker
│   └── Dockerfile
├── registrar.php
├── SAMBA-CONFIG
│   ├── smb.conf
│   └── usuarios.sh
├── 
├── sql_database
│   ├── insviladegracia.sql
│   └── usuarios.sh
├── www
│   ├── Asistencia
│   │   ├── asistencia.php
│   │   ├── horario2.json
│   │   ├── horario3.json
│   │   ├── horario.json
│   │   ├── index.php
│   │   ├── login.php
│   │   ├── media.php
│   │   ├── media.php.bk
│   │   ├── registrar.php
│   │   ├── registrar.php.bk
│   │   ├── settingsbd.php
│   │   └── style.css
│   ├── hola.css
│   ├── img
│   │   ├── avatar.png
│   │   └── vilaGraciadibuix.jpg
│   └── index.html
└── 24 directories, 50 files
bernat@srvdocker:~/PROYECTE_3_ACAVAT$

```

DOCKER HUB Y GITHUB

CONCLUSIÓN

Al concluir este proyecto de Docker Compose, hemos consolidado un profundo entendimiento sobre la creación y gestión de archivos docker-compose. A lo largo de este proceso, hemos adquirido habilidades para conectar varios contenedores entre sí a través de una red privada, lo que ha demostrado ser crucial para el funcionamiento fluido de nuestras aplicaciones. Además, hemos explorado y dominado la interconexión entre contenedores dependientes, asegurando una integración sin problemas de servicios como bind9, Portainer, Duplicati, Samba, Apache, PhpMyAdmin, MySQL e ISC-DHCP-SERVER.

Este proyecto nos ha permitido no solo comprender la arquitectura y la dinámica de Docker Compose, sino también poner en práctica nuestras habilidades mediante la creación y configuración de contenedores para una variedad de servicios críticos. La capacidad de orquestar estos servicios de manera eficiente y escalable dentro de un entorno de contenedores ha demostrado ser invaluable para nuestras futuras implementaciones y despliegues.

Además, hemos fortalecido nuestra capacidad para resolver problemas y optimizar la configuración de Docker Compose, garantizando un rendimiento óptimo de nuestros servicios en todo momento.

WEBGRAFÍA

1. **bind9:**
 - Sitio web oficial: [ISC BIND](#)
 - Documentación: [BIND 9 Documentation](#)
2. **Portainer:**
 - Sitio web oficial: [Portainer](#)
 - Documentación: [Portainer Documentation](#)
3. **Duplicati:**
 - Sitio web oficial: [Duplicati](#)
 - Documentación: [Duplicati Documentation](#)
4. **Samba:**
 - Sitio web oficial: [Samba](#)
 - Documentación: [Samba Documentation](#)
5. **Apache:**
 - Sitio web oficial: [Apache HTTP Server](#)
 - Documentación: [Apache HTTP Server Documentation](#)
6. **PhpMyAdmin:**
 - Sitio web oficial: [phpMyAdmin](#)
 - Documentación: [phpMyAdmin Documentation](#)
7. **MySQL:**
 - Sitio web oficial: [MySQL](#)
 - Documentación: [MySQL Documentation](#)
8. **ISC-DHCP-SERVER:**
 - Sitio web oficial: [ISC DHCP](#)
 - Documentación: [ISC DHCP Documentation](#)
9. **Docker Compose:**
 - Sitio web oficial: [Docker Compose](#)
 - Documentación: [Docker Compose Documentation](#)
10. **Dockerfile:**
 - Documentación oficial: [Best practices for writing Dockerfiles](#)
 - Guía de referencia de Docker: [Dockerfile reference](#)
11. **SSL (Secure Sockets Layer):**
 - Documentación de Let's Encrypt (popular CA para SSL): [Let's Encrypt Documentation](#)
 - Documentación de OpenSSL (utilizado para generar y gestionar certificados SSL): [OpenSSL Documentation](#)