

# Instal·lació d'un servidor Apache amb suport per a aplicacions PHP i nodes virtuals



Nom i Cognom: Carles Molina Espinós

Curs: 2CFGs DAW

## **Index**

Objectius.....	3
Realitzar la instal·lació i configuració bàsica de servidors d'aplicacions.....	3
Realitzar proves de funcionament dels servidors web i d'aplicacions.....	3
Analitzar l'estructura i recursos que componen una aplicació web.....	3
Enunciat.....	3

# Objectius

- Realitzar la instal·lació i configuració bàsica de servidors d'aplicacions
- Realitzar proves de funcionament dels servidors web i d'aplicacions.
- Analitzar l'estructura i recursos que componen una aplicació web.

## Enunciat

### **1. Investiga què és CGI (Common Gateway Interface) i quina relació té en les aplicacions web. ?**

El CGI (Common Gateway Interface) és un estàndard que permet a un servidor web executar programes externs per generar contingut dinàmic en resposta a les peticions dels usuaris. És un mecanisme clau en les primeres aplicacions web dinàmiques, ja que permet la comunicació entre el servidor i scripts escrits en llenguatges com Perl, Python, Bash o C.

#### Relació del CGI amb les aplicacions web

1. Generació de contingut dinàmic → CGI permet que un servidor web execute un script que genera una resposta basada en dades d'entrada (formularis, bases de dades, etc.).
2. Interacció amb l'usuari → Els CGI s'utilitzen per processar formularis web, gestionar autenticacions i recuperar informació d'una base de dades.
3. Independència del llenguatge → Pot usar qualsevol llenguatge de programació capaç d'executar-se al servidor i retornar una resposta HTTP.
4. Funcionament → Quan un usuari accedeix a un URL que apunta a un script CGI, el servidor:
  - Executa l'script i li passa les dades d'entrada (via GET o POST).
  - L'script processa la informació i retorna una resposta en format HTML o un altre tipus MIME.
  - El servidor envia la resposta al navegador.

## 2. Apache és un programari modular. Esbrina quins mòduls són necessaris per executar PHP i instal·la'ls per poder executar aplicacions web basades en PHP

Apache és un servidor web modular, cosa que significa que pot ampliar les seues funcionalitats mitjançant mòduls. Per a executar aplicacions web basades en **PHP**, cal instal·lar i activar els següents mòduls:

1. **php** → Proporciona el suport bàsic per a PHP.
2. **libapache2-mod-php** → Mòdul que permet a Apache interpretar i executar codi PHP.

### Instal·lació dels mòduls en Ubuntu

Executa la següent comanda per instal·lar els mòduls necessaris:

```
sudo apt install php libapache2-mod-php php-cli
```

Després, **reinicia Apache** per aplicar els canvis:

```
sudo systemctl restart apache2
```

## 3. Afegir al DocumentRoot un fitxer anomenat phpinfo.php amb el següent contingut:

Crear el fitxer `phpinfo.php` en el DocumentRoot:

```
sudo nano /var/www/html/phpinfo.php
```

Afegir el contingut següent al fitxer:

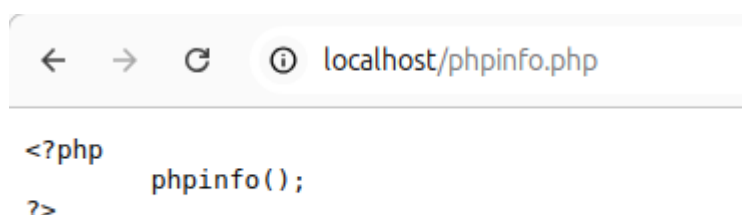
```
<?php
phpinfo();
?>
```

Assegurar permisos correctes:

```
sudo chown www-data:www-data /var/www/html/phpinfo.php
sudo chmod 644 /var/www/html/phpinfo.php
```

Verificar la configuració i obri un navegador:

```
http://localhost/phpinfo.php
```



Si PHP està funcionant correctament, veuràs una pàgina amb informació detallada sobre la configuració de PHP, les extensions carregades i altres detalls del sistema.

#### 4. Accedeix al document anterior i comprova quins mòduls hi ha instal·lats a Apache

Per comprovar quins **mòduls estan instal·lats a Apache**, pots utilitzar la següent comanda en el terminal del servidor:

```
sudo apache2ctl -M
```

#### 5. Què és PHP-FPM? Quina diferència hi ha respecte a usar el mòdul integrat de PHP en Apache?

PHP-FPM (PHP FastCGI Process Manager) és una implementació de FastCGI per executar aplicacions PHP de manera més eficient i flexible. Proporciona un millor rendiment i control sobre els processos PHP que s'executen en un servidor web.

Diferències entre PHP-FPM i el mòdul integrat de PHP en Apache:

##### Rendiment:

- **PHP-FPM:** Permet una millor gestió de la càrrega i el rendiment, ja que PHP s'executa com un procés separat que pot ser gestionat de forma independent del servidor web. Això és útil quan el servidor rep moltes sol·licituds.
- **Mòdul integrat de PHP en Apache:** PHP s'executa dins del mateix procés d'Apache, el que pot ser menys eficient, sobretot en entorns amb altes càrregues.

##### Escalabilitat:

- **PHP-FPM:** Ofereix millor escalabilitat, ja que pot gestionar diversos processos PHP simultàniament i es pot configurar per tenir un nombre variable de treballadors segons les necessitats de càrrega.
- **Mòdul integrat de PHP en Apache:** Tot i que pot gestionar diverses sol·licituds, no té la mateixa flexibilitat per adaptar-se a canvis de càrrega.

##### Gestió de processos:

- **PHP-FPM:** Permet un control més gran sobre els processos PHP, com la configuració del nombre màxim de processos, el reinici automàtic de treballadors i la configuració de temps d'espera i memòria.
- **Mòdul integrat de PHP en Apache:** No ofereix tanta flexibilitat en la gestió dels processos PHP, ja que PHP s'executa com part del servidor Apache.

**6. Configura 2 nodes virtuals que responguen al nom prova-NOM.lan i test-NOM.lan. Els DocumentRoot seran /var/www/prova i /var/www/test respectivament. El DocumentIndex haurà de ser index.php en els dos casos. Els noms del servidor els pots afegir al fitxer /etc/hosts**

Editar el fitxer /etc/hosts:

Afegeix les línies següents al fitxer /etc/hosts per associar els noms de domini amb l'adreça IP del servidor (suposant que és 127.0.0.1, però canvia-ho segons sigui necessari):

```
127.0.0.1 prova-CARLES.lan
127.0.0.1 test-CARLES.lan
```

Crear el fitxer de configuració per als VirtualHosts

Obre o crea un fitxer de configuració per als VirtualHosts. Normalment, les configuracions de VirtualHost estan ubicades en /etc/apache2/sites-available/. Crearem un fitxer per a cada node virtual:

Crea o edita el fitxer prova-CARLES.conf:

```
sudo nano /etc/apache2/sites-available/prova-CARLES.conf
```

Afegim la configuració següent:

```
<VirtualHost *:80>
  ServerAdmin webmaster@prova-CARLES.lan
  ServerName prova-CARLES.lan
  DocumentRoot /var/www/prova
  DirectoryIndex index.php

  ErrorLog /var/log/apache2/prova/error.log
  CustomLog /var/log/apache2/prova/access.log combined

  <Directory /var/www/prova>
    Options Indexes FollowSymLinks
    AllowOverride All
    Require all granted
  </Directory>
</VirtualHost>
```

Crea o edita el fitxer test-CARLES.conf:

```
sudo nano /etc/apache2/sites-available/test-CARLES.conf
```

Afegim la configuració següent:

```
<VirtualHost *:80>
  ServerAdmin webmaster@test-CARLES.lan
  ServerName test-CARLES.lan
  DocumentRoot /var/www/test
  DirectoryIndex index.php

  ErrorLog /var/log/apache2/test/error.log
  CustomLog /var/log/apache2/test/access.log combined

  <Directory /var/www/test>
    Options Indexes FollowSymLinks
    AllowOverride All
    Require all granted
  </Directory>
</VirtualHost>
```

Crear els directoris per als registres i contingut web:

```
sudo mkdir -p /var/www/prova
sudo mkdir -p /var/www/test
sudo mkdir -p /var/log/apache2/prova
sudo mkdir -p /var/log/apache2/test
```

Permisos adequats:

```
sudo chown -R www-data:www-data /var/www/prova /var/www/test
sudo chown -R www-data:www-data /var/log/apache2/prova /var/log/apache2/test
```

Activar les configuracions de VirtualHost:

```
sudo a2ensite prova-NOM.conf
sudo a2ensite test-NOM.conf
```

Reinicar Apache:

```
sudo systemctl restart apache2
```

Comprobació prova-CARLES.lan:

<http://prova-CARLES.lan>

← → ↻ ⚠ No es seguro prova-carles.lan/

```
<?php echo "Benvingut a Prova!" ?>
```

Comprobació test-CARLES.lan:

<http://test-CARLES.lan>

← → ↻ ⚠ No es seguro test-carles.lan/

```
<?php echo "Benvingut a Test!" ?>
```

**7. Configura el fitxer de registre perquè els registres es guarden en el directori /var/log/apache2/prova i /var/log/apache2/test respectivament.**

VirtualHost per a prova-CARLES.lan:

```
<VirtualHost *:80>
    ServerAdmin webmaster@prova-CARLES.lan
    ServerName prova-CARLES.lan
    DocumentRoot /var/www/prova
    DirectoryIndex index.php

    # Registres per aquest VirtualHost
    ErrorLog /var/log/apache2/prova/error.log
    CustomLog /var/log/apache2/prova/access.log combined

    <Directory /var/www/prova>
        Options Indexes FollowSymLinks
        AllowOverride All
        Require all granted
    </Directory>
</VirtualHost>
```



VirtualHost per a test-CARLES.lan:

```
<VirtualHost *:80>
  ServerAdmin webmaster@test-CARLES.lan
  ServerName test-CARLES.lan
  DocumentRoot /var/www/test
  DirectoryIndex index.php

  # Registres per aquest VirtualHost
  ErrorLog /var/log/apache2/test/error.log
  CustomLog /var/log/apache2/test/access.log combined

  <Directory /var/www/test>
    Options Indexes FollowSymLinks
    AllowOverride All
    Require all granted
  </Directory>
</VirtualHost>
```

Mostrar els logs en el temps real:

```
tail -f /var/log/prova-CARLES.lan
```

```
tail -f /var/log/test-CARLES.lan
```