

Actividades

Programación en entorno servidor

Álvaro Cañas González

Índice

1. ¿Qué son los servicios web?.....	3
2. ¿Qué son los estándares web?.....	3
3. ¿Quién forma la W3C?.....	3
4. Haz un breve resumen de algunas de las tecnologías más importantes utilizadas en la web.....	3
5. ¿Qué es un lenguaje de scripts?.....	3
6. Explica las diferencias entre los lenguajes del lado del servidor y los lenguajes del cliente. Ejemplos.....	3
7. Haz un breve análisis de los lenguajes más utilizados en aplicaciones web.....	4
8. ¿Qué es Apache y cuáles son sus características más importantes?.....	4
9. ¿Cuáles son los archivos de configuración más importantes de Apache?. Explica algunos de los parámetros de configuración con algún ejemplo.....	5
10. ¿Para qué sirve el módulo mod_ssl de Apache?.....	7
11. ¿Qué es una entidad certificadora?.....	7
12. Explica las diferencias entre el protocolo http y https.....	7
13. ¿Qué es un servidor de bases de datos relacional?.....	7
14. Haz un estudio de los servidores de bases de datos más utilizados en aplicaciones web.....	7
15. Haz un análisis de los servicios de hosting comerciales.....	8
16. Elige la respuesta correcta justificando brevemente su elección.....	9

1. ¿Qué son los servicios web?

- Son tecnologías que utilizan una serie de protocolos de comunicación para intercambiar datos e información entre aplicaciones.

2. ¿Qué son los estándares web?

- Son especificaciones y protocolos que describen el comportamiento de la WWW y compatibilizan entre si diferentes tecnologías.

3. ¿Quién forma la W3C?

- Fundado por Tim Berners-Lee, está compuesta por bastantes organizaciones de todo tipo.

4. Haz un breve resumen de algunas de las tecnologías más importantes utilizadas en la web

- Podemos mencionar diferentes tecnologías utilizadas en la web, una de estas son los servicios, como pueden ser los servicios **web**, **ftp**, entre otros, junto a estos también podemos encontrar tecnologías muy utilizadas en conjunto, como pueden ser los lenguajes de programación o de maquetación como pueden ser **HTML**, **CSS**, **JavaScript** o **PHP**.

5. ¿Qué es un lenguaje de scripts?

- Es un lenguaje de programación utilizado para manipular, personalizar y automatizar diferentes tareas en un sistema informático, por ejemplo, **PHP** es un lenguaje de scripts que se utiliza mayormente para insertar código dentro de páginas **HTML**.

6. Explica las diferencias entre los lenguajes del lado del servidor y los lenguajes del cliente. Ejemplos

- La mayor diferencia es el entorno de ejecución, mientras que los del lado del servidor se ejecutan en un servidor aligerando la carga del cliente y dando más potencia y seguridad a la ejecución del código, los del lado cliente se ejecutan en el navegador del usuario final.

7. Haz un breve análisis de los lenguajes más utilizados en aplicaciones web

- Los lenguajes más utilizados en aplicaciones web son los siguientes:
- **HTML** → (HypertText Markup Lenguaje), es un lenguaje de marcado utilizado para la creación de páginas web, este es el que se encarga de darle forma a toda la página y puede ir acompañado de otros como CSS, JavaScript, PHP, entre otros.
- **CSS** → (Hoja de estilos en cascada), este lenguaje es utilizado para darle diseño a las páginas web, va acompañando al HTML para darle formato y diseño.
- **JavaScript** → Este lenguaje de programación interpretado, es utilizado principalmente en el lado cliente, aunque también puede ser ejecutado en el servidor, este nos permite mejorar las interfaces y usabilidad de las páginas web.
- **PHP** → Este lenguaje de programación de scripts interpretado, principalmente ejecutado en un servidor, adaptado especialmente para el desarrollo web, este nos permite ampliar en gran medida las funcionalidades de las páginas web.
- **Python** → Este lenguaje de programación interpretado de alto nivel multiparadigma lo hace esencial para nuevo programadores, este tiene una sintaxis de fácil lectura y nos permite elaborar diferentes tipos de aplicaciones, entre ellas en el ámbito web.
- **Java** → Este lenguaje de programación orientado a objetos, cuenta con ser multiplataforma al ejecutarse en una maquina virtual, este es uno de los más populares y puede ser utilizado para diferentes plataformas, como por ejemplo, en las aplicaciones web.

8. ¿Qué es Apache y cuáles son sus características más importantes?

- Apache es un servidor web de código abierto multiplataforma que implementa el protocolo HTTP.

Sus características más importantes son las siguientes:

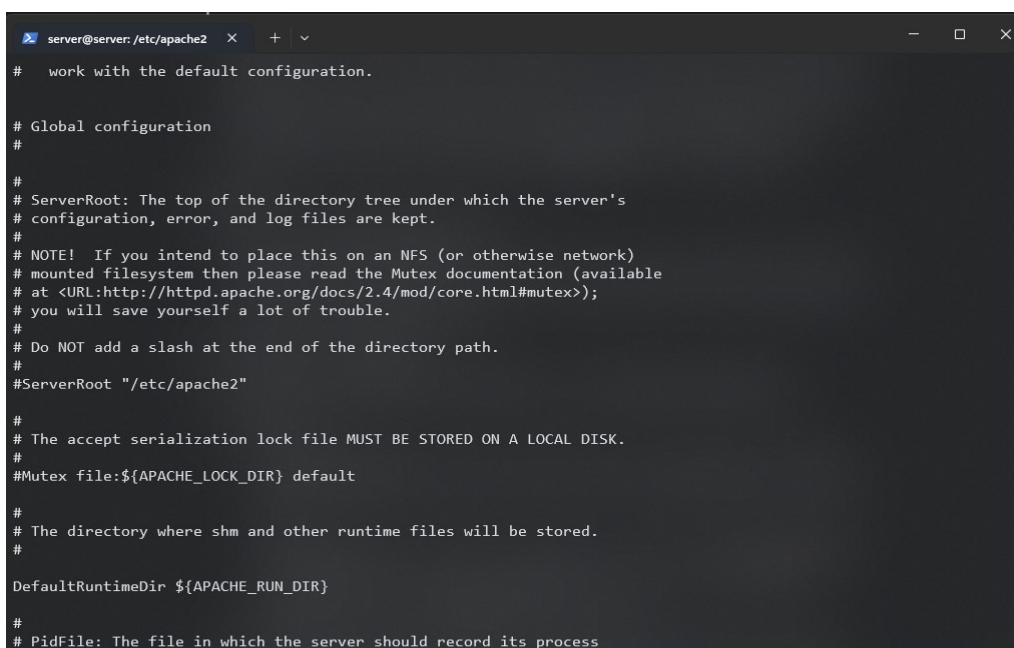
- Código abierto.
- Gratuito.

- Multiplataforma.
- Modular.
- Múltiples lenguajes.
- Soporte virtual host.
- Seguridad.
- Muy configurable.
- Escalable.
- Gran comunidad.
- Logs y estadísticas.
- Etc.

9. ¿Cuáles son los archivos de configuración más importantes de Apache?. Explica algunos de los parámetros de configuración con algún ejemplo

- Algunos de los archivos de configuración más importantes son los siguientes:

- `apache2.conf` → Este es el archivo principal de configuración, en este se definen directivas globales que afectan a todo el servidor, como la configuración de módulos, políticas de seguridad, límite de recursos, entre otros.



```
server@server: /etc/apache2
# work with the default configuration.

# Global configuration
#
#
# ServerRoot: The top of the directory tree under which the server's
# configuration, error, and log files are kept.
#
# NOTE! If you intend to place this on an NFS (or otherwise network)
# mounted filesystem then please read the Mutex documentation (available
# at <URL:http://httpd.apache.org/docs/2.4/mod/core.html#mutex>);
# you will save yourself a lot of trouble.
#
# Do NOT add a slash at the end of the directory path.
#
#ServerRoot "/etc/apache2"

#
# The accept serialization lock file MUST BE STORED ON A LOCAL DISK.
#
#Mutex file:${APACHE_LOCK_DIR} default

#
# The directory where shm and other runtime files will be stored.
#

DefaultRuntimeDir ${APACHE_RUN_DIR}

#
# PidFile: The file in which the server should record its process
```

- sites-available y sites-enabled → En estos directorios se almacenan las configuraciones de los Virtual Hosts de los sitios web y los enlaces simbólicos.

```
server@server: /etc/apache2$ ls
apache2.conf  conf-enabled  magic          mods-enabled  sites-available
conf-available  envvars      mods-available  ports.conf    sites-enabled
server@server: /etc/apache2$ ls sites-available/
000-default.conf  default-ssl.conf
server@server: /etc/apache2$ ls sites-enabled/
000-default.conf
server@server: /etc/apache2$ |
```

- ports.conf → Define los puertos de escucha de Apache, en el siguiente ejemplo solo especifica el puerto 443 para HTTPS.

```
server@server: /etc/apache2$ ls
apache2.conf  conf-enabled  magic          mods-enabled  sites-available
conf-available  envvars      mods-available  ports.conf    sites-enabled
server@server: /etc/apache2$ cat ports.conf
# If you just change the port or add more ports here, you will likely also
# have to change the VirtualHost statement in
# /etc/apache2/sites-enabled/000-default.conf

Listen 80

<IfModule ssl_module>
    Listen 443
</IfModule>

<IfModule mod_gnutls.c>
    Listen 443
</IfModule>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
server@server: /etc/apache2$ |
```

- mods-available y mods-enabled → Este contiene la configuración y enlaces simbólicos para los módulos de apache.

```
server@server: /etc/apache2$ ls mods-available/
access_compat.load  cache_disk.load  http2.load  php7.4.load  rewrite.load
actions.conf        cache.load        ident.load  php8.1.conf  sed.load
actions.load        cache_socache.load  imagemap.load  php8.1.load  session_cookie.load
alias.conf          cern_meta.load   include.load  php8.3.conf  session_crypto.load
allowmethods.load  cgi.conf         info.conf     proxy.ajp.load  session_dbd.load
asis.load           cgi.load         lbmethod.bybusyness.load  proxy_balancer.conf  setenvif.conf
auth_basic.load     charset_lite.load  lbmethod.byrequests.load  proxy_balancer.load  setenvif.load
auth_digest.load    data.load         lbmethod.bytraffic.load   proxy.conf            slotmem_plain.load
auth_form.load      dav_fs.conf       lbmethod.heartbeat.load   proxy_connect.load    slotmem_shm.load
authn_anon.load     dav_fs.load       ldap.conf     proxy_express.load   socache_dbm.load
authn_core.load     dav.load          ldap.load     proxy_fcgi.load      socache_memcache.load
authn_dbd.load      dav_lock.load     log_debug.load  proxy_fdpass.load    socache_redis.load
authn_dbm.load      dbd.load          log_forensic.load  proxy_ftp.conf        socache_shmcb.load
authn_file.load     deflate.conf      lua.load       proxy_ftp.load       spelling.load
authn_socache.load  deflate.load      macro.load     proxy_hcheck.load    ssl.conf
authnz_fcgi.load    dialup.load      md.load       proxy_html.conf      ssl.load
authnz_ldap.load    dir.conf         mime.conf     proxy_html.load     status.conf
authz_core.load     dir.load         mime.load     proxy_http2.load     status.load
authz_dbd.load      dump_io.load     mime_magic.conf  proxy_http.load      substitute.load
authz_dbm.load      echo.load        mime_magic.load  proxy.load           suexec.load
authz_groupfile.load  env.load        mpm.event.conf  proxy_scgi.load      unique_id.load
authz_host.load     expires.load     mpm.event.load  proxy_uwsgi.load     userdir.conf
authz_owner.load    ext_filter.load  mpm_prefork.conf  proxy_wstunnel.load  usertrack.load
authz_user.load     file_cache.load  mpm_prefork.load  ratelimit.load       vhost_alias.load
autoindex.conf      filter.load      mpm_worker.conf  reflector.load       xml2enc.load
autoindex.load      headers.load     mpm_worker.load  remotepip.load
brotli.load         heartbeat.load   negotiation.conf  reqtimeout.conf
buffer.load         heartmonitor.load  negotiation.load  reqtimeout.load
cache_disk.conf     http2.conf       php7.4.conf     request.load
```

```
server@server: /etc/apache2$ |
```

```
server@server: /etc/apache2$ ls mods-enabled/
access_compat.load  authz_core.load  deflate.load  mime.load  php8.3.load  status.load
alias.conf          authz_host.load  dir.conf     mpm_prefork.conf  php8.3.conf  substitute.load
alias.load          authz_user.load  dir.load     mpm_prefork.load  reqtimeout.conf  suexec.load
auth_basic.load     autoindex.conf  env.load     negotiation.conf  setenvif.conf  usertrack.load
authn_core.load     autoindex.load  filter.load  negotiation.load  setenvif.load  vhost_alias.load
authn_file.load     deflate.conf     mime.conf    php8.3.conf      status.conf
```

- Etc.

10. ¿Para qué sirve el módulo mod_ssl de Apache?

- Este módulo proporciona una interfaz a la biblioteca OpenSSL, otorgando un cifrado fuerte utilizando los protocolos de capa de sockets seguros y seguridad de la capa de transporte.

11. ¿Qué es una entidad certificadora?

- Es una organización de confianza que emite y gestiona certificados digitales que permiten asegurar las comunicaciones en Internet mediante protocolos como el HTTPS.

12. Explica las diferencias entre el protocolo http y https

- La mayor diferencia es la capa de cifrado que se aplica sobre el protocolo, HTTPS es el resultado de utilizar la tecnología TLS/SSL que cifra la información que se intercambian entre cliente y servidor de forma que esta no puede ser interceptada ni manipulada.

También encontramos la autenticación que asegura al cliente que la interacción se realiza con el servidor correcto y la integridad de los datos que garantiza que estos no sean modificados durante la transferencia de datos.

13. ¿Qué es un servidor de bases de datos relacional?

- Es una aplicación que proporciona servicios de bases de datos relacional a un programa o aplicación, esta puede ser nativa o web.

14. Haz un estudio de los servidores de bases de datos más utilizados en aplicaciones web

- Los servidores de BBDD más utilizados en la web son los siguientes:
 - **MySQL** → Este SGBD es de los más populares en la web, incluyendo plataformas como WordPress es conocido por su rendimiento y simplicidad, formando parte de la pila LAMP.
 - **PostgreSQL** → Este SGBD es altamente potente y de código abierto, ofrece características avanzadas como el cumplimiento estricto de las

normas SQL, soporte para JSON y extensiones de datos geoespaciales con PostGIS.

- **MariaDB** → Este es un fork (bifurcación) de MySQL desarrollado tras la adquisición de MySQL por Oracle, estos son 100% compatibles, mantenido por la comunidad y con un enfoque a la estabilidad y rendimiento.
- **MongoDB** → Este es un SGBD NoSQL orientado a documentos, muy popular para aplicaciones que manejen grandes volúmenes de datos no estructurados, estos son almacenados en formato JSON, permitiendo así una gran flexibilidad y escalabilidad.

15. Haz un análisis de los servicios de hosting comerciales

- Algunos de los servicios de hosting comerciales son los siguientes:

- **Hostinger** → Este servicio de hosting de pago cuenta con servidores rápidos, incluye dominio, plantillas de creación de sitios web y diferentes recursos. Pero uno de los problemas que tiene este hosting es que no cuenta con soporte técnico telefónico.
- **SiteGround** → Este hosting de pago cuenta con servidores en América, Europa y Asia, ofrece servicios a millones de sitios y es de los más recomendados para almacenar sus páginas con Linux. Todos sus planes incluyen SSL y no tiene límite de velocidad de transferencia. Su gran desventaja es su precio elevado frente a sus competidores.
- **Google Cloud** → Cuando Google unifica todas sus aplicaciones de desarrollo web en un solo espacio añadió el servicio cloud hosting a este, cuenta con facilidad de escalada y te permite utilizar otras herramientas de Google. Sin embargo necesitaras de grandes conocimientos por su complejidad de administración.
- **Dinahosting** → Cuenta con planes de diferentes precios que ofrecen gran calidad, proporciona un soporte técnico inmediata las 24 horas del día el año, incluye por defectos discos NVMe otorgando más velocidad. Su gran desventaja es que no cuenta con dominio gratuito.
- **A2 Hosting** → Este es uno de los hosting más antiguos que cuenta con planes personalizados, ofrece distintos certificados SSL y cuentas mercantiles PayPal, proporciona la posibilidad de escalar o hosting compartido. Su gran desventaja es la falta de seguridad para datos bancarios.

16. Elige la respuesta correcta justificando brevemente su elección

- En las arquitecturas cliente/servidor:
 - a) La parte cliente se conoce como front-end y la servidor como back-end.
 - b) La parte cliente se conoce como back-end y la servidor como front-end.
 - c) La parte cliente se conoce como display-end y la servidor como process.end.
 - d) Las anteriores respuestas no son correctas.
- De las siguientes tecnologías, ¿cuál se emplea para programar un cliente web?:
 - a) CGI.
 - b) ASP.
 - c) HTML.
 - d) ColdFusion.
- De las siguientes tecnologías, ¿cuál no se emplea para programar un servidor web?:
 - a) Servlets.
 - b) Applets.
 - c) SSI.
 - d) ColdFusion.
- El protocolo HTTP fue creado por:
 - a) Tim Berners-Lee.
 - b) Marc Andresseen.
 - c) Linus Torvalds.
 - d) Steve Jobs.
- JavaScript es un lenguaje de programación:
 - a) Orientado a objetos.

- b) Basado en objetos.
- c) Sin objetos.
- d) Las anteriores respuestas no son correctas.
- Quando nos referimos a Internet y a la Web:
 - a) Internet incluye la Web.
 - b) La Web incluye a Internet.
 - c) Nos estamos refiriendo a lo mismo.
 - d) Las anteriores respuestas no son correctas.
- La estandarización de la Web es tarea de:
 - a) Microsoft y sus compañías asociadas.
 - b) Netscape y sus compañías asociadas.
 - c) W3C.
 - d) IEEE.
- De las siguientes tecnologías, ¿cuál no se emplea para programar un cliente web?:
 - a) HTML.
 - b) SSL.
 - c) JavaScript.
 - d) VRML.