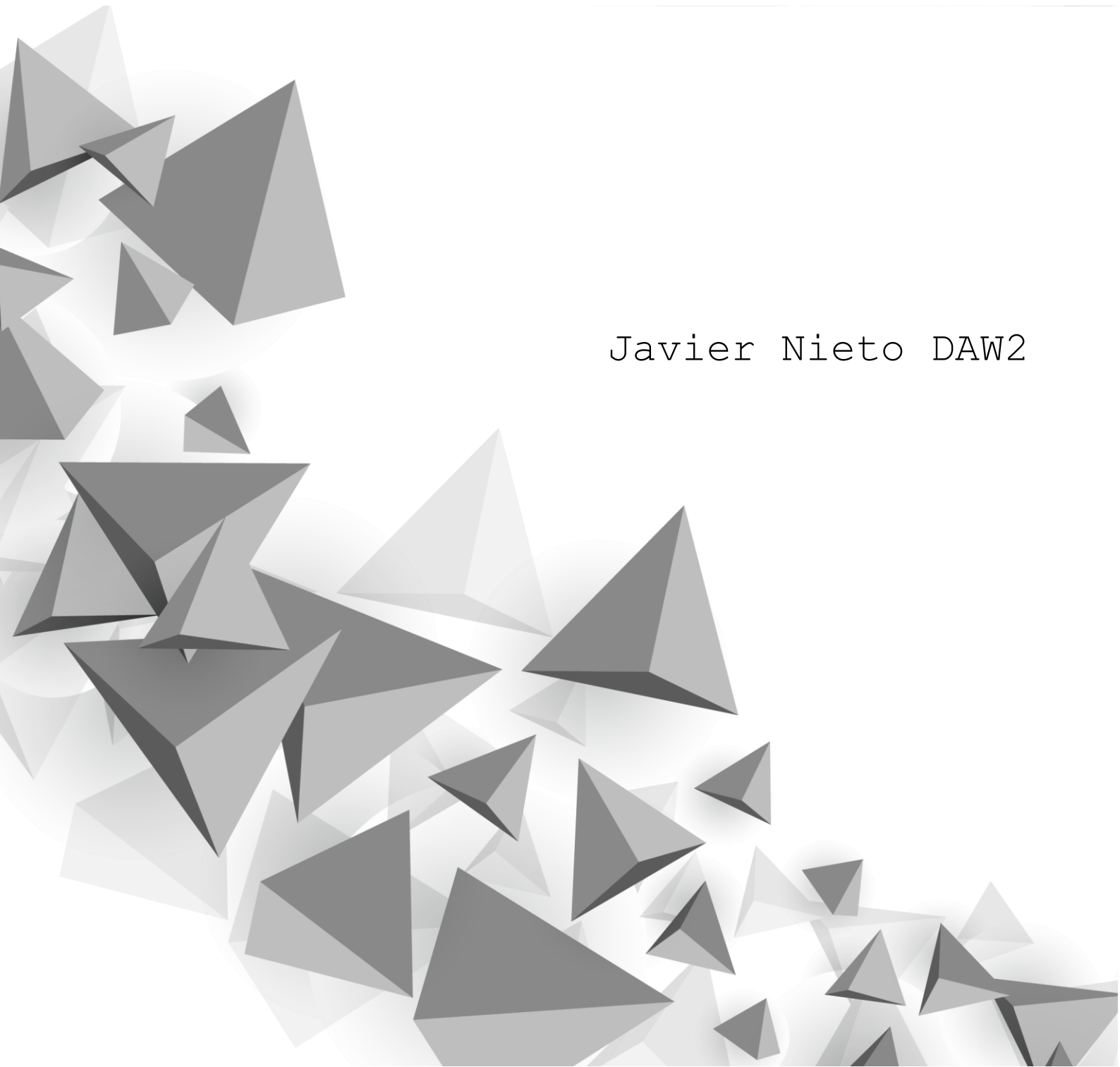


Ejercicios

UD1-Tarea 2

Javier Nieto DAW2



Índice

1. Especificaciones técnicas de tu ordenador personal	3
2. Estudio detallado del protocolo HTTP.....	4
3. Se pide buscar en Internet diferentes aplicaciones web y clasificarlas según su tipo, como mínimo dos páginas de cada tipo.....	5
4. ¿Qué tecnologías se utilizan en cada una de las capas de la arquitectura web?	6
5. Existen multitud de servidores web en el mercado. La empresa Netcraft se encarga de hacer estadísticas y mediciones sobre la utilización de los distintos servidores web. Se pide una gráfica con la utilización de los principales servidores web y las principales características.	7
6. Diferencias y semejanzas entre Saas, PaaS e IaaS.....	9
7. Diferentes versiones disponibles de Apache Tomcat y sus diferentes versiones de las especificaciones	9
8. URI, URN Y URL: definición, sintaxis (estructura) y diferentes ejemplos.	10
9. Escalabilidad horizontal: Ventajas e inconvenientes	11
10. Escalabilidad vertical: Ventajas e inconvenientes.....	12
Bibliografía	13

Responde a las siguientes cuestiones:

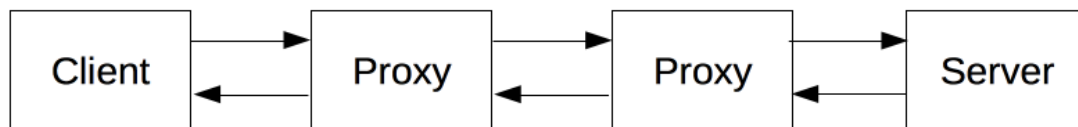
1. Especificaciones técnicas de tu ordenador personal

PARTE	DESCRIPCIÓN
Sistema Operativo	Windows 10 Home, x64
Placa Base	Acer EA50_KV v1.8
CPU	AMD Quad-Core FX-7500 (2.1 GHz, Turbo 3.3 GHz, 4 MB Cache)
Memoria RAM	2x4GB DDR3
Disco Duro	SSD 500GB + 1TB HDD
Tarjeta Grafica	AMD Radeon R7 M265 de 2 GB

2. Estudio detallado del protocolo HTTP.

HTTP es un protocolo basado en el principio de cliente-servidor: las peticiones son enviadas por una entidad: el agente del usuario (o un proxy a petición de uno). La mayoría de las veces el agente del usuario (cliente) es un navegador Web.

Cada petición individual se envía a un servidor, el cuál la gestiona y responde. Entre cada petición y respuesta, hay varios intermediarios, normalmente denominados proxies, los cuales realizan distintas funciones, como: gateways o caches.



Cada vez que un cliente realiza una petición a un servidor, se ejecutan los siguientes pasos:

- Un usuario accede a una URL, seleccionando un enlace de un documento HTML o introduciéndola directamente en el campo Location del cliente Web.
- El cliente Web descodifica la URL, separando sus diferentes partes. Así identifica el protocolo de acceso, la dirección DNS o IP del servidor, el posible puerto opcional (el valor por defecto es 80) y el objeto requerido del servidor.
- Se abre una conexión TCP/IP con el servidor, llamando al puerto TCP correspondiente.
- Se realiza la petición. Para ello, se envía el comando necesario (GET, POST, HEAD,...), la dirección del objeto requerido (el contenido de la URL que sigue a la dirección del servidor), la versión del protocolo HTTP empleada (casi siempre HTTP/1.0) y un conjunto variable de información, que incluye datos sobre las capacidades del browser, datos opcionales para el servidor,...
- El servidor devuelve la respuesta al cliente. Consiste en un código de estado y el tipo de dato MIME de la información de retorno, seguido de la propia información.
- Se cierra la conexión TCP.

3. Se pide buscar en Internet diferentes aplicaciones web y clasificarlas según su tipo, como mínimo dos páginas de cada tipo.

Aplicaciones Web Estáticas

- <https://pizzeriapiyayos.webnode.es/>
- <https://mesonelpicaro.com/>

Aplicaciones Web Dinámicas

- <https://www.netflix.com/es/>
- <https://es.shein.com/>

Aplicaciones Web del Portal

- <https://www.jcyl.es/>
- <https://transparencia.gob.es/>

Aplicaciones de comercio electrónico

- <https://www.amazon.es/>
- <https://www.pccomponentes.com/>

Aplicaciones Web Animadas

- <http://yourpower.panda.org/>
- <http://www.rleonardi.com/interactive-resume/>

Aplicaciones web con gestor de contenidos (CMS)

- <https://blog.ted.com/>
- <https://www.bbc.com/>

4. ¿Qué tecnologías se utilizan en cada una de las capas de la arquitectura web?

Capa de presentación

HTML, CSS, DOM, JavaScript, AJAX.

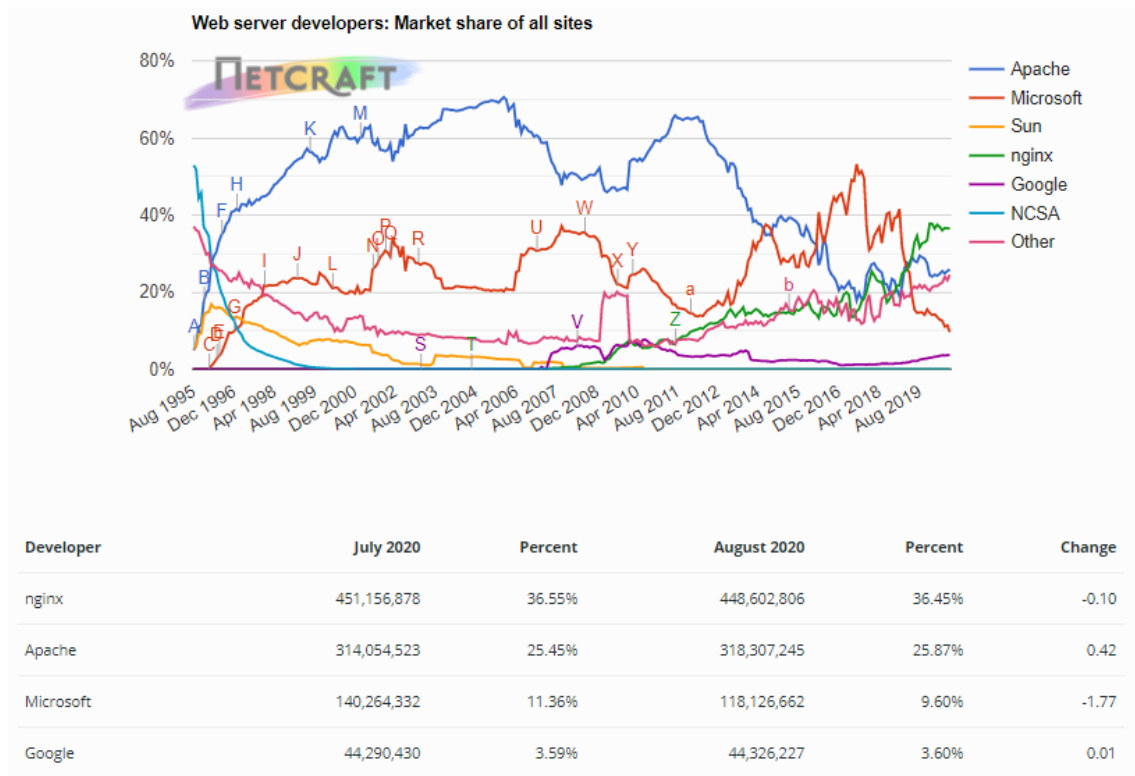
Capa de negocio

PHP, Java con sus tecnologías JavaServlets y JavaServer Pages (JSP), Perl, Ruby, Python, Node.js, C# y Visual Basic con sus tecnologías ASP/ASP.NET.

Capa de acceso a datos:

MySQL, MariaDB

5. Existen multitud de servidores web en el mercado. La empresa Netcraft se encarga de hacer estadísticas y mediciones sobre la utilización de los distintos servidores web. Se pide una gráfica con la utilización de los principales servidores web y las principales características.



Gráfica de los servidores web más usados, teniendo en cuenta todos los sitios web

Nginx

- Servidor de archivos estáticos y dinámicos
- Sirve como proxy inverso + servidor de cache
- Soporte de autenticación HTTP
- Soporta índices y autoindexado
- Balanceo de carga inteligente
- Tolerante ante fallos
- Ofrece alta disponibilidad
- Escalabilidad asegurada
- Compatible con IPv4 e IPv6
- Soporte para HTTPS con certificados SSH

Apache - Apache HTTPS

- Modular
- Extensible
- Código abierto
- Popular (fácil conseguir ayuda/soporte)
- Multi-plataforma

Microsoft - ISS

- Es modular para permitir una gestión ordenada y la posibilidad de agregar funciones adicionales.
- Brinda soporte para los varios protocolos: HTTP/HTTPS, FTP/FTPS, SMTP y NNTP.
- Provee soporte SSL para Socket Layer, adicionalmente provee múltiples alternativas de autenticación como Kerberos entre otros.
- El lenguaje dinámico soportado es ASP.NET sin embargo también es posible configurar para utilizar PHP, VBScript, Perl e incluso Java (Java Server Pages y Servlets) y prácticamente cualquier lenguaje como módulo adicional o a través de la interface CGI

6. Diferencias y semejanzas entre SaaS, PaaS e IaaS

Diferencias

La principal diferencia entre SaaS, PaaS e IaaS es el mantenimiento y soporte ofrecidos por el proveedor. En IaaS es el desarrollador de las aplicaciones el que debe ocuparse de todo, mientras que en SaaS, los usuarios ni siquiera tienen acceso al software. Por su parte en PaaS pueden gestionar la plataforma, pero no el servidor.

Semejanzas

Todos ellos ofrecen servicio en línea, cuyo funcionamiento se da a través de la nube.. Además, ofrecen la posibilidad de pagar únicamente en función de los recursos a consumir y del uso que se les vaya a dar. Todos ellos son sistemas escalables, de modo que permite aumentar la capacidad siempre y cuando sea necesario.

7. Diferentes versiones disponibles de Apache Tomcat y sus diferentes versiones de las especificaciones

Servlet Spec	JSP Spec	EL Spec	WebSocket Spec	Authentication (JASIC) Spec	Apache Tomcat Version	Latest Released Version	Supported Java Versions
5.0	3.0	4.0	2.0	2.0	10.0.x	10.0.0-M8 (alpha)	8 and later
4.0	2.3	3.0	1.1	1.1	9.0.x	9.0.38	8 and later
3.1	2.3	3.0	1.1	1.1	8.5.x	8.5.58	7 and later
3.1	2.3	3.0	1.1	N/A	8.0.x (superseded)	8.0.53 (superseded)	7 and later
3.0	2.2	2.2	1.1	N/A	7.0.x	7.0.106	6 and later (7 and later for WebSocket)
2.5	2.1	2.1	N/A	N/A	6.0.x (archived)	6.0.53 (archived)	5 and later
2.4	2.0	N/A	N/A	N/A	5.5.x (archived)	5.5.36 (archived)	1.4 and later
2.3	1.2	N/A	N/A	N/A	4.1.x (archived)	4.1.40 (archived)	1.3 and later
2.2	1.1	N/A	N/A	N/A	3.3.x (archived)	3.3.2 (archived)	1.1 and later

Tabla de las diferentes versiones de Apache Tomcat disponibles para diferentes versiones de las especificaciones

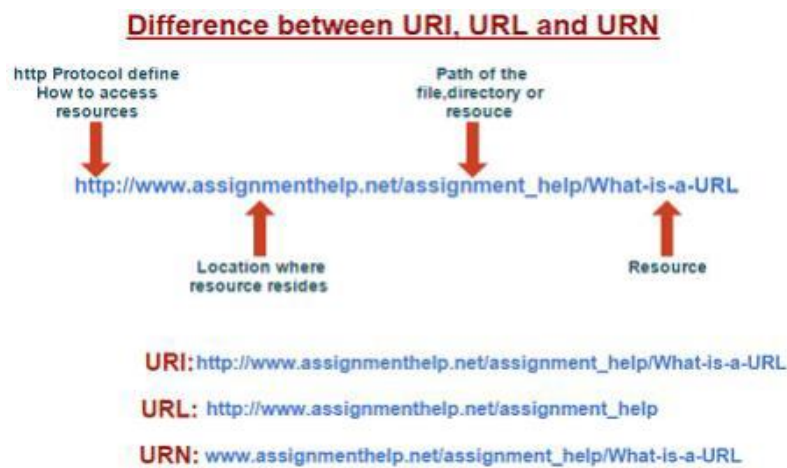
En esta tabla podemos observar como van cambiando las diferentes versiones de los servlet, JSP, WebSocket, Java... a medida que cambian las diferentes versiones de Apache Tomcat.

8. URI, URN Y URL: definición, sintaxis (estructura) y diferentes ejemplos.

Un **URI** (Uniform Resource Identifier) es una cadena de caracteres que identifica los recursos de una red de forma unívoca.

Un **URL** (Uniform Resource Locator) es la dirección específica que se asigna a cada uno de los recursos disponibles en la red para que puedan ser localizados o identificados.

Un **URN** (Uniform Resource Name) indica el nombre de un recurso que debe ser único en todo internet.



Diferencias entre URL, URL and URN



Composición de una URL

9. Escalabilidad horizontal: Ventajas e inconvenientes

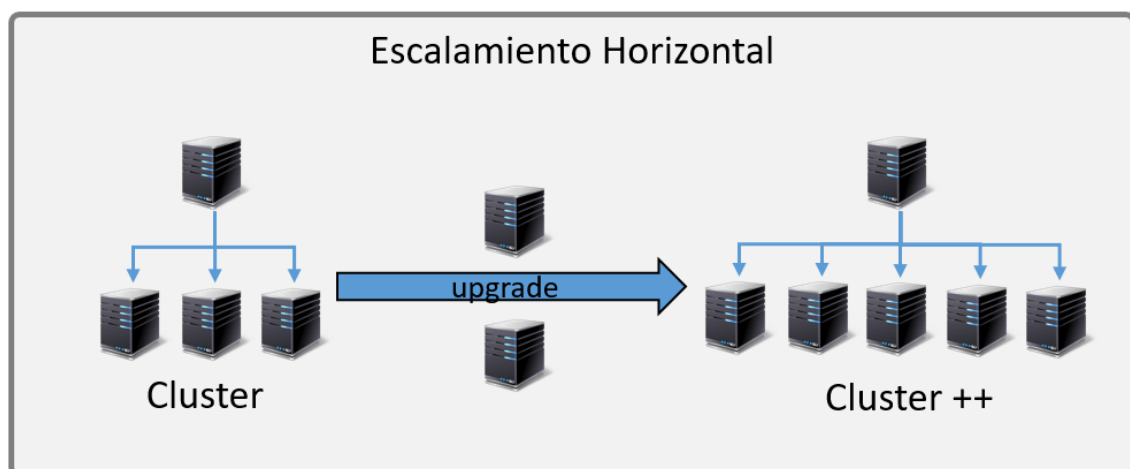
La **escalabilidad horizontal** es el **modelo más potente**, significa agregar más servidores (conocidos como Nodos) trabajando como un solo nodo. Se crea una red de servidores conocida como Clúster, con la finalidad de repartirse el trabajo entre todos los nodos del clúster.

Ventajas

- El crecimiento es prácticamente infinito, podríamos agregar cuantos servidores sean necesarios
- Es posible combinarse con el escalamiento vertical.
- Soporta la alta disponibilidad
- Si un nodo falla, los demás sigue trabajando.
- Soporta el balanceo de cargas.

Desventajas

- Requiere de mucho mantenimiento
- Es difícil de configurar
- Requiere de grandes cambios en las aplicaciones (si no fueron diseñadas para trabajar en cluster)
- Requiere de una infraestructura más grande



10. Escalabilidad vertical: Ventajas e inconvenientes.

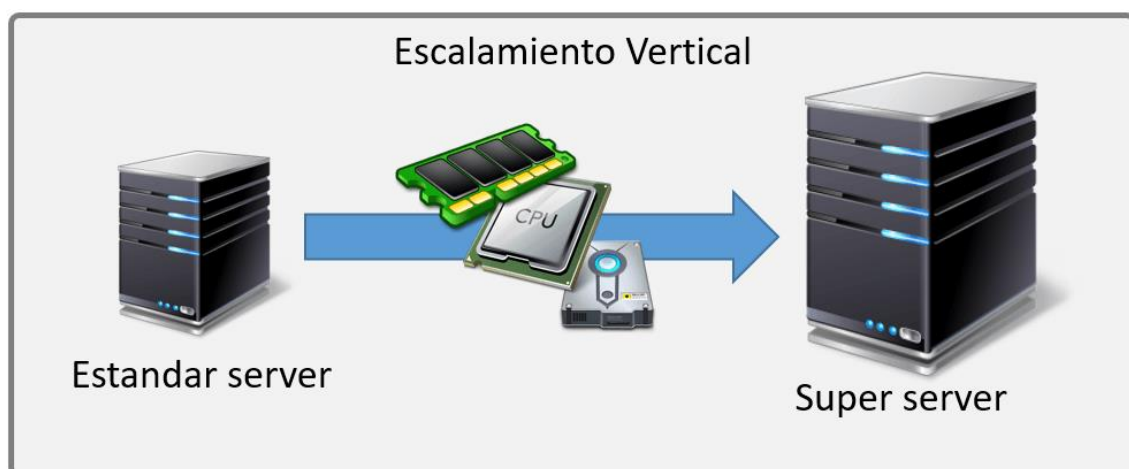
La **escalabilidad vertical** o **hacia arriba**, significa aumentar el hardware por uno más potente, como disco duro, memoria, procesador, etc. pero también puede ser la migración completa del hardware por uno más potente.

Ventajas

- No implica un gran problema para las aplicaciones, pues todo el cambio es sobre el hardware
- Es mucho más fácil de implementar que el escalamiento horizontal.
- Puede ser una solución rápida y económica (compara con modificar el software)

Desventajas

- El crecimiento está limitado por el hardware.
- Una falla en el servidor implica que la aplicación se detenga.
- No soporta la Alta disponibilidad.
- Hacer un upgrade del hardware al máximo pues llegar a ser muy caro, ya que las partes más nuevas suelen ser caras con respecto al rendimiento de un modelo anterior.



Bibliografía

<https://developer.mozilla.org/es/docs/Web/HTTP/Overview>

<https://news.netcraft.com/archives/2020/08/26/august-2020-web-server-survey.html>

<https://blog.infranetworking.com/que-es-nginx/>

[https://blog.infranetworking.com/servidor-iis/#Caracteristicas del Servidor Internet Information Services](https://blog.infranetworking.com/servidor-iis/#Caracteristicas_del_Servidor_Internet_Information_Services)

https://www.ecured.cu/Servidor_HTTP_Apache#Ventajas_y_aceptaci.C3.B3n

<https://neo.lcc.uma.es/evirtual/cdd/tutorial/aplicacion/http.html>

<https://obelearningservices.com/url-uri-y-urn-explicacion-y-diferencias/>