

Implementación de una app en una red local y comparación de su arquitectura en dos y tres capas

David Espíndola, Denis Fernández, Lidia Ojeda, Alfredo Ramírez, Santiago Pioli¹

¹ Dpto. Informática, Facultad Cs. Exactas, Univeridad Nacional del Nordeste, Corrientes, República Argentina

davidgabrielespindola@gmail.com, dafer1982@outlook.com.ar, mariann1042@gmail.com, ramirezalfredoagustin80@gmail.com, santiago.pioli@comunidad.unne.edu.ar

Abstract. El trabajo que se presenta sugiere dos maneras de afrontar el desarrollo de un sistema de información. Se comparan los modelos de dos y tres capas para la arquitectura cliente-servidor, donde la aplicación y la base de datos se alojarán en un solo servidor como en servidores separados, midiendo sus rendimientos y contrastando ventajas y desventajas de ambas topologías. Nuestra propuesta trata de software de código abierto que se suelen instalar juntas para que un servidor pueda alojar aplicaciones y sitios web dinámicos escritos en PHP. Estos softwares se tratan del sistema operativo Linux con el servidor web Apache. Los datos del sitio se almacenan en una base de datos MySQL y el contenido dinámico se procesa mediante PHP.

Keywords: Aplicación PHP, Servidor Linux, Servidor Apache, Red Local

1 Introducción

Este trabajo de investigación se centra en el proceso integral de alojar una aplicación PHP en una red local y analizar su funcionamiento. Además, se considerarán las herramientas de análisis y evaluación pertinentes para obtener una comprensión completa del rendimiento y la eficiencia de la aplicación alojada localmente.

Una arquitectura multicapa es una arquitectura cliente-servidor en la que las funciones de presentación, lógica de negocio y gestión de datos están separadas físicamente. Una aplicación Web es proporcionada por un servidor Web y utilizada por usuarios que se conectan desde cualquier punto vía clientes Web (browsers o navegadores). La arquitectura de un Sitio Web tiene tres componentes principales:

- Un servidor Web
- Una conexión de red
- Uno o más clientes

El servidor Web distribuye páginas de información formateada a los clientes que las solicitan. Los requerimientos son hechos a través de una conexión de red, y para ello se usa el protocolo HTTP. Una vez que se solicita esta petición mediante el protocolo HTTP y la recibe el servidor Web, éste localiza la página Web en su sistema de archivos y la envía de vuelta al navegador que la solicitó.

Las aplicaciones web se modelan mediante lo que se conoce como modelo de capas, Una capa representa un elemento que procesa o trata información. Los tipos son:

- Modelo de dos capas: La información atraviesa dos capas entre la interfaz y la administración de los datos.
- Modelo de n-capas: La información atraviesa varias capas, el más habitual es el modelo de tres capas.

A través de esta investigación, se espera brindar una comprensión sólida de los pasos necesarios para alojar una aplicación PHP en una red local, así como una evaluación completa de su funcionamiento.

La aplicación en cuestión se encarga de registrar la distribución de suministros informáticos desde el departamento de sistemas hacia los diferentes sectores del organismo. Su desarrollo se basó en la premisa de mantener un control preciso del stock del depósito, facilitar la realización de pedidos de suministros, además, llevar un registro detallado de la cantidad y la frecuencia con la que cada sector recibe los insumos necesarios. Este sistema está alojado en un servidor interno, al cual tienen acceso las diferentes jefaturas que forman parte de la subgerencia de sistemas. Cada una de estas jefaturas desempeña un rol específico en el proceso, donde algunos realizan pedidos, otros se encargan de las entregas, y otros tienen funciones de administración.

2 Instrumentación

Para el despliegue del estudio en cuestión se dispuso de los siguientes paquetes de software y hardware:

- Notebook Intel Core i3 1005G1 y 8Gb de memoria RAM
- Sistema operativo Linux en su distribución Ubuntu 22.04 (con firewall desactivado durante las pruebas)
- MySQL Workbench
- NMap – Zenmap (GUI)
- Apache Jmeter (5.6.2)

3 Desarrollo

Arquitecturas de las Aplicaciones Web

El acceso a las páginas está agrupado también en el tiempo (sesión). Los componentes de una aplicación Web son:

1. Lógica de negocio.

Parte más importante de la aplicación.

Define los procesos que involucran a la aplicación.

Conjunto de operaciones requeridas para proveer el servicio.

2. Administración de los datos.

Manipulación de BD y archivos.

3. Interfaz

Los usuarios acceden a través de navegadores, móviles, PDAs, etc.

Funcionalidad accesible a través del navegador.

Limitada y dirigida por la aplicación.

Las aplicaciones web se modelan mediante lo que se conoce como modelo de capas. Una capa representa un elemento que procesa o trata información. Los tipos que vemos son modelo de dos y tres capas.

Modelo de dos capas

La arquitectura cliente/servidor también es conocida como arquitectura de dos capas. Requiere una interfaz de usuario que se instala y se ejecuta en una PC o estación de trabajo y envía solicitudes a un servidor para ejecutar operaciones complejas.

Los componentes de las capas son:

- Cliente: La lógica de negocio está inmersa dentro de la aplicación que realiza el interfaz de usuario, en el lado del cliente.
- Servidor: Administra los datos.

Modelo de 3 capas

Está diseñada para superar las limitaciones de las arquitecturas ajustadas al modelo de dos capas, introduce una capa intermedia entre presentación y los datos, los procesos pueden ser manejados de forma separada a la interfaz de usuario y a los datos, esta capa intermedia centraliza la lógica de negocio, haciendo la administración más sencilla, los datos se pueden integrar de múltiples fuentes, las aplicaciones web actuales se ajustan a este modelo.

Componentes:

- Un cliente, es decir, el equipo que solicita los recursos, equipado con una interfaz de usuario (generalmente un navegador web) para la presentación.
- El servidor de aplicaciones (también denominado software intermedio), cuya tarea es proporcionar los recursos solicitados, pero que requiere de otro servidor para hacerlo.
- El servidor de datos, que proporciona al servidor de aplicaciones los datos que éste le solicitó.

En el siguiente cuadro se resumen las ventajas y desventajas de ambos modelos:

	Ventajas	Desventajas
Modelo en dos capas	El control centralizado en el servidor garantiza la seguridad y la integridad de los datos, evitando que programas clientes no autorizados afecten el sistema.	Es difícilmente escalable.
	El modelo cliente-servidor permite escalar clientes y servidores de forma independiente para manejar aumentos de capacidad.	Número de conexiones reducido.
	Hay tecnologías bien desarrolladas diseñadas para el modelo cliente-servidor que garantizan seguridad en transacciones, amigabilidad en la interfaz y facilidad de uso.	Alta carga de la red.
La flexibilidad es restringida.		
Modelo en tres capas	Un mayor grado de flexibilidad.	Pueden incrementar el tráfico en la red cuando muchos clientes envían peticiones a un solo servidor.
	Mayor seguridad, ya que la seguridad se puede definir independientemente para cada servicio y en cada nivel.	Requiere más balance de carga y tolerancia a las fallas.
	Mejor rendimiento, ya que las tareas se comparten entre servidores.	Los exploradores actuales no son todos iguales.

Por último, para evidenciar las redes creadas se utilizó la herramienta Nmap de escaneo de puertos y descubrimiento de hosts que existe actualmente. Nmap nos permitirá obtener una gran cantidad de información sobre los equipos de nuestra red, es capaz de escanear qué hosts están levantados, e incluso comprobar si tienen algún puerto abierto.

4 Comparativa de rendimientos

Las pruebas de rendimiento son un aspecto extremadamente importante del ciclo de vida del desarrollo de software y aplicaciones. En un nivel básico, JMeter funciona simulando visitantes a la aplicación o servicio permitiendo a los usuarios crear y enviar solicitudes HTTP (Protocolo de transferencia de hipertexto) al servidor.

El plan de prueba se dividió en dos etapas, las pruebas en dos capas y en tres capas, simulando 200 peticiones de usuarios de listar los productos en periodos de 20 segundos

dando como resultado 10 peticiones por segundo. Los resultados obtenidos fueron los siguientes:

		Prueba en dos capas	Prueba en tres capas
Promedio	Tiempo promedio que tarda una solicitud HTTP.	12	18
Desviación estándar	Evalúa la consistencia del rendimiento de un sistema bajo carga simulada. Una desviación baja indica que los puntos de datos están muy cerca de la media.	14,88	7,55
Rendimiento	Cantidad de pedidos por segundos/minutos/hora de la prueba.	10,2 /sec	10,0 /sec
KB/sec recibidos	Tasa de transferencia de datos medida en kilobytes por segundo (KB/sec) durante una prueba de carga	24,53	24,18
KB/sec enviados	Tasa de transferencia de datos en kilobytes por segundo que se envían desde el cliente (la máquina donde se ejecuta JMeter) al servidor durante la ejecución de una prueba de carga	2,68	2,64
Latencia (Promedio)	Tiempo que transcurre entre un estímulo y la respuesta que produce, en base al promedio de latencia entre todas las peticiones.	8,26	10,28
Tiempo conectado (Promedio)	Tiempo total durante el cual la prueba de carga está ejecutando conexiones al servidor	2,74 ms	2,02 ms
Tiempo de muestreo (Promedio)	Tiempo entre cada medición de datos durante la ejecución de la prueba.	12,47 ms	18,19 ms

En resumen, podemos decir que el modelo de dos capas arroja mejores resultados en cuanto a su rendimiento. Esto lo vemos, por ejemplo, en que la latencia que pasa de 8.26 a 10.28 en la prueba de tres capas y que el tiempo promedio que tarda cada solicitud es menor. Por otro lado, la cantidad de kilobytes recibidos y enviados que puede procesar por segundo es mayor en dicho modelo. El tiempo promedio que tarda una solicitud http también es menor en este modelo siendo de 12ms y 18ms para el modelo de tres capas. El rendimiento también arrojó un resultado levemente mejor ya que la cantidad de pedidos por segundos es 0.2 segundos más.

Estos resultados eran esperados ya que al tener la base de datos en un host diferentes y tener más saltos en la red, los tiempos de respuestas serán mayores. Sin embargo, algunos indicadores son favorables al modelo de 3 capas, como el tiempo conectado siendo el tiempo que ejecuta conexiones al servidor disminuye 72ms.

5 Conclusiones

La aplicación, es su versión inicial en red local, se beneficia de un modelo de dos capas ya que arroja ventajas significativas de rendimiento en cuanto a las variables evaluadas y por lo tanto el desarrollo puede ser más rápido, ya que hay menos componentes y conexiones que gestionar. Esto puede ser beneficioso en proyectos pequeños, como es el caso de estudio. La adopción de este modelo también favorece el control centralizado en el servidor como también la integridad de los datos. Otro aspecto a tener en cuenta es que un modelo de dos capas puede requerir menos recursos y, por lo tanto, puede ser más económico en términos de infraestructura y costos asociados.

References

1. I. T. d. Matehuala, «Programacion Web,» [En línea]. Available: <https://programacion-webisc.wordpress.com/2-1-arquitectura-de-las-aplicaciones-web/>.
2. S. D. Luz, «Realiza escaneos de puertos con Nmap a cualquier servidor o sistema,» Redes Zone, 2023 Agosto 2023. [En línea]. Available: <https://www.redeszone.net/tutoriales/configuracion-puertos/nmap-escanear-puertos-comandos/>.
3. E. Heidi, «Cómo instalar la pila Linux, Apache, MySQL y PHP (LAMP) en Ubuntu 20.04,» Digital Ocean, 19 Mayo 2020. [En línea]. Available: <https://www.digitalocean.com/community/tutorials/how-to-install-linux-apache-mysql-php-lamp-stack-on-ubuntu-20-04-es>.
4. U. d. Murcia, «Alojamiento web, Visión General,» 2018. [En línea]. Available: <https://www.um.es/docencia/barzana/DAWEB/2017-18/daweb-practica-alojamiento-web.html>.
5. E. B. A. Pulgar, «Guía básica para utilizar JMeter,» Academia Pragma, 22 Abril 2021. [En línea]. Available: <https://www.pragma.com.co/academia/lecciones/guia-basica-para-utilizar-jmeter>.
6. Vargas, «ARQUITECTURAS DE APLICACIONES WEB DE 2, 3 Y N CAPAS,» Mundo Android Web, [En línea]. Available: <https://tec755.wordpress.com/infografia/>.