

SERVIDOR DE APLICACIONES

DESPLIEGUE DE APLICACIONES WEB



JOEL CAPAPE HERNÁNDEZ
2º DESARROLLO DE APLICACIONES WEB

ÍNDICE

| | |
|------------------------|---|
| RESUMEN | 2 |
| FUNDAMENTACIÓN TEÓRICA | 3 |
| REFLEXIÓN PERSONAL | 3 |

PORTFOLIO 4 – SERVIDOR DE APLICACIONES

RESUMEN

1. ¿Qué es un servidor de aplicaciones? Busca algún ejemplo y describe sus características

Los servidores de aplicaciones funcionan como intermediarios que alojan aplicaciones web (programas que se ejecutan en un navegador web) para que los usuarios de la red ejecuten y usen estas aplicaciones sin tener que instalarlas en sus ordenadores.

Un servidor de aplicaciones está diseñado para instalar, operar y alojar aplicaciones y otros servicios asociados para usuarios finales, servicios y empresas de TI, y facilita el alojamiento y la distribución de aplicaciones de consumo o empresariales de gama alta. Dependiendo de lo que tenga instalado, un servidor de aplicaciones puede clasificarse de varias maneras: como un servidor web, como un servidor de aplicaciones de bases de datos, como un servidor de aplicaciones de propósito general o como un servidor de aplicaciones empresariales. Por ejemplo:

- Apache Geronimo

Geronimo se caracteriza por un diseño arquitectónico que se basa en el concepto de Inversión de control (IoC) (a veces llamado Inyección de dependencias), lo que significa que el núcleo no tiene dependencia directa de cualquiera de sus componentes. El núcleo es un framework para los servicios que controla el ciclo de vida y el registro. El núcleo está basado en Java EE. Funciona con los servicios y componentes de Java EE para construir configuraciones específicas, una de las cuales es una pila de soluciones de Java EE completa.

La mayoría de los servicios de Geronimo son añadidos y configurados a través de GBeans para convertirse en una parte del servidor de aplicaciones en conjunto. Un GBean es la interfaz que conecta el componente con el núcleo. Cada GBean puede mantener estado, depender de e interrelacionarse con otros GBeans, y operar en los eventos del núcleo y de otros GBeans. La interfaz de GBeans permite alternar entre dos contenedores de servlets, por ejemplo, Jetty o Tomcat, sin afectar a toda la arquitectura usando una interfaz de GBeans. Esta arquitectura flexible permite a los desarrolladores de Geronimo integrar varios proyectos de software de código abierto

2. ¿Qué es Apache Tomcat?

Tomcat es un contenedor de código abierto de servlets que se puede usar para compilar y ejecutar aplicaciones web realizadas en Java. Implementa y da soporte tanto a servlets como a páginas realizadas en JavaServer Pages, Java Expression Language y Java WebSocket, del ecosistema de Java.

FUNDAMENTACIÓN TEÓRICA

1. ¿Qué es un Servlet?

Un servlet es un programa Java que se ejecuta en un servidor Web y construye o sirve páginas web. De esta forma se pueden construir páginas dinámicas, basadas en diferentes fuentes variables: datos proporcionados por el usuario, fuentes de información variable (páginas de noticias, por ejemplo), o programas que extraigan información de bases de datos.

Comparado con un CGI, un servlet es más sencillo de utilizar, más eficiente (se arranca un hilo por cada petición y no un proceso entero), más potente y portable. Con los servlets podemos, entre otras muchas cosas, procesar, sincronizar y coordinar múltiples peticiones de clientes, reenviar peticiones a otros servlets u otros servidores, etc.

2. ¿Qué es un fichero WAR y para qué se utiliza?

Un archivo WAR es un archivo JAR especial que se utiliza para empaquetar una aplicación web para facilitar su implementación en un servidor de aplicaciones.

Un archivo WAR se utiliza para distribuir una colección de archivos Java (JAR), archivos comprimidos que contienen recursos de Java como clases, texto, imágenes, XML y HTML, y otros recursos que se utilizan para entregar una aplicación web.

REFLEXIÓN PERSONAL

1. ¿Cuáles son las diferencias entre apache y tomcat?

Las diferencias que existen entre apache y tomcat son las siguientes:

- El objetivo que tiene Tomcat es desplegar aplicaciones Java, mientras que Apache es un servidor HTTP de propósito general.
- Apache está escrito en C, mientras que Tomcat está escrito en Java
- Apache te proporciona más opciones de configuración que Tomcat
- Puedes utilizar un servidor Tomcat para hacer páginas web estáticas, es menos eficiente para realizar páginas estáticas puedes utilizar mejor Apache. Por ejemplo, Tomcat precarga Java Virtual Machine y otras bibliotecas relacionadas con Java que no necesitarás en la mayoría de los sitios web.