

Implantación de arquitecturas web

Busca en Internet información sobre los siguientes conceptos y contesta en un documento aparte con tus propias palabras:

Ejercicio 1:

◆ *Arquitectura de software y patrones de diseño, definición y diferencias*

La arquitectura del software es la base de un sistema de software, tiene la información de los Componentes y como se comportan entre sí y también con el entorno externo.

Los patrones de diseño son soluciones a problemas comunes en el desarrollo de software a nivel de Componentes, no es código, es como una guía/plantilla que ayuda a solucionar los problemas.

Diferencia: Arquitectura del software es la estructura y los patrones de diseño son soluciones a problemas.

La arquitectura del software tiene un nivel alto de abstracción en cambio los patrones es medio / bajo.

La arquitectura del software afecta a todo el sistema y en cambio los patrones a partes específicas del código.

◆ *Arquitectura web monolítica, definición, características, ventajas y desventajas*

Descripción: Es un modelo de desarrollo de software en el que todo esta construido como un proyecto o unidad.

Características: Toda la app es solo un archivo.

Los módulos comparten el mismo espacio y recursos.

Tiene componentes fuertemente acoplados.

Ventajas: Facilidad de desarrollo inicial y la sencillez del despliegue de la app.

Desventajas: Escalabilidad limitada y complejidad del mantenimiento.

◆ *Arquitectura web por capas, definición, características, tipos, ventajas y desventajas*

Definición: Es un estilo de diseño que se basa por capas jerárquicas, donde cada una tiene una responsabilidad.

Características: Separación de responsabilidades.

Comunicación entre las capas adyacentes.

Fácil mantenimiento y pruebas.

Tipos:

1. Capa de presentación
2. Capa de lógica de negocio
3. Capa de acceso a datos
4. Capa de persistencia o almacenamiento

Ventajas: Claridad y organización, reutilización del código y mantenimiento más sencillo.

Desventajas: Rendimiento bajo, dependencia entre capas y mayor complejidad.

◆ *Arquitectura web de microservicios, definición, características, ventajas y desventajas*

Definición: Es un estilo de diseño en el que una app se construye con servicios pequeños, independientes el uno al otro y despegables de forma autónoma.

Características: División por funcionalidades.

Desacoplamiento.

Comunicación con APIs.

Desarrollo y despliegue autónomo.

Ventajas: Escalabilidad, despliegue continuo, mejor aislamiento de fallos

Desventajas: Mayor complejidad, gestión compleja y seguridad distribuida.

◆ *Arquitectura web serverless, definición, características, ventajas y desventajas*

Definición: Es un modelo de desarrollo en el que los desarrolladores escriben y despliegan código sin tener que gestionar servidores.

Características: Ejecución basada en funciones.

Despliegue por eventos.

Modelo de pago por uso.

Ventajas: Reducción de costos, escalado automático, despliegue rápido.

Desventajas: Cold start, difícil de depurar, limitaciones de tiempo y recursos.

Ejercicio 2:

♦ *Servidores web más utilizados, breve historia, características y precios*

Nginx: Nginx es un servidor web de código abierto y de alto rendimiento, lanzado en 2004. Se ha convertido en uno de los servidores web más utilizados, junto con Apache. Nginx es una solución ideal para gestionar sitios web de alto tráfico. Muchos sitios y aplicaciones web de alta visibilidad, como Netflix o Pinterest, utilizan el servidor web Nginx. A fecha de mayo de 2022, Nginx posee un 33,5 % del mercado según W3Techs y un 30,71 % según Netcraft.

- **Gratuito y de código abierto**
- **Arquitectura asíncrona basada en eventos**
- **Alto rendimiento**
- **Bajo consumo de memoria**
- **Compatibilidad con IPv6**

Apache HTTP Server: El servidor web Apache, oficialmente conocido como Apache HTTP Server, se lanzó en 1995. Apache es un servidor web gratuito y de código abierto; desarrollado y mantenido por la Apache Software Foundation (ASF). Es uno de los servidores web más populares del mundo. A fecha de mayo de 2022, Apache posee un 31,5 % del mercado según W3Techs y un 22,99 % según Netcraft.

- **Gratuito y de código abierto**
- **Arquitectura basada en módulos**
- **Fácil configuración y personalización**
- **Actualizaciones y parches de seguridad regulares**
- **Gran comunidad de desarrolladores**
- **Compatibilidad con IPv6**

LiteSpeed: LiteSpeed es un servidor web propietario, de pago desarrollado por LiteSpeed Technologies; lanzado en 2003. Dispone de tres licencias: Free Starter, Site Owner y Web Host. Es una alternativa de pago para sitios web de alto tráfico, que ofrece un alto rendimiento y gran escalabilidad. A fecha de mayo de 2022, LiteSpeed posee un 12,1 % del mercado según W3Techs y un 4,35 % según Netcraft.

- **Arquitectura basada en eventos**
- **Ligero**
- **Configuración sencilla**
- **Características anti-DDoS integradas**
- **Mantenimiento sin interrupciones**

7,71 \$ al mes.

♦ *Lenguajes de programación web más utilizados, breve historia, características, ventajas y desventajas*

Java:

La historia del lenguaje Java comienza en 1991, cuando fue creado inicialmente como "OAK" por James Gosling y su equipo en Sun Microsystems.

-Orientado a Objetos

-Robustez y Seguridad

-Alto rendimiento

Ventajas: Portabilidad, seguridad , facilidad de uso.

Desventajas: Rendimiento, consumo de memoria.

JavaScript:

La historia del lenguaje JavaScript comienza en 1995, cuando fue creado por Brendan Eich mientras trabajaba en Netscape Communications. Inicialmente fue llamado Mocha, luego LiveScript, y finalmente JavaScript, en una estrategia de marketing para aprovechar la popularidad de Java en ese momento. A pesar del nombre, JavaScript y Java son lenguajes completamente diferentes.

- **Lenguaje interpretado y basado en prototipos**
- **Orientado a eventos y dinámico**
- **Muy utilizado en el desarrollo web (frontend y backend)**

Ventajas:

- **Compatibilidad con todos los navegadores**

- **Interactividad en páginas web**
- **Versatilidad (puede usarse en el cliente y en el servidor con Node.js)**
- **Gran comunidad y abundancia de recursos**

Desventajas:

- **Problemas de seguridad (por ejecución del lado del cliente)**
- **Dependencia del navegador**
- **Puede ser difícil de depurar en proyectos grandes sin buenas prácticas**

PHP:

La historia del lenguaje PHP comienza en 1994, cuando fue creado por Rasmus Lerdorf. Inicialmente era un conjunto de scripts llamados "Personal Home Page Tools", pero con el tiempo evolucionó hasta convertirse en un lenguaje de programación completo, conocido como PHP: Hypertext Preprocessor.

- **Lenguaje de programación del lado del servidor**
- **Orientado a scripts para desarrollo web dinámico**
- **Fácil integración con HTML y bases de datos como MySQL**

Ventajas:

- **Fácil de aprender y usar**
- **Gran compatibilidad con servidores y bases de datos**
- **Comunidad amplia y muchos recursos**
- **Muy utilizado en CMS populares como WordPress, Joomla, Drupal**

Desventajas:

- **Código desorganizado si no se siguen buenas prácticas**
- **Menor rendimiento frente a tecnologías más modernas**
- **Seguridad puede ser un problema si no se configura correctamente**

◆ *Gestores de bases de datos relacionales más utilizados, características, ventajas y desventajas*

MySQL:

- **Características:** Multihilo, multiusuario, compatibilidad con SQL, soporte multiplataforma, escalabilidad limitada con bases grandes.
- **Ventajas:** Rendimiento, facilidad de uso, amplia compatibilidad.
- **Desventajas:** Escalabilidad limitada con bases grandes.

SQL Server:

- **Características:** Soporte para transacciones, visualización de datos, escalabilidad, seguridad.
- **Ventajas:** Integración completa con herramientas de Microsoft, soporte para aplicaciones empresariales.
- **Desventajas:** Escalabilidad limitada con bases grandes.

PostgreSQL:

- **Características:** Orientado a objetos, soporte para transacciones ACID, herramienta pgAdmin para administración.
- **Ventajas:** Robustez, flexibilidad, soporte para grandes volúmenes de datos.
- **Desventajas:** Escalabilidad limitada con bases grandes.

◆ *Gestores de bases de datos no relacionales más utilizados, características, ventajas y desventajas*

- **MongoDB:**
- **Características:** Almacena datos en documentos, permite consultas ad hoc y es escalable.
- **Ventajas:** Alta flexibilidad, rendimiento elevado y fácil integración.
- **Desventajas:** No soporta la atomicidad, consistencia y durabilidad como las bases de datos relacionales.
- **Cassandra:**

- **Características:** *Diseñada para gestionar grandes volúmenes de datos distribuidos.*
- **Ventajas:** *Alta escalabilidad y capacidad para manejar datos grandes.*
- **Desventajas:** *No es tan flexible como MongoDB en términos de esquemas de almacenamiento.*
- **DynamoDB:**
- **Características:** *Parte de Amazon Web Services, ideal para aplicaciones móviles y videojuegos.*
- **Ventajas:** *Tiempo de respuesta bajo y rápida escalabilidad.*
- **Desventajas:** *No soporta la gestión de grandes volúmenes de datos como MongoDB.*

Estos sistemas son ideales para aplicaciones que requieren manejar grandes volúmenes de datos y ofrecen características que superan a las bases de datos relacionales en ciertas áreas.