

Aplicaciones Web

JAVA EE: JAVA PLATFORM, ENTERPRISE EDITION

Extensión de la Java SE con especificaciones y APIs para la gestión de transacciones, acceso a la base de datos, desarrollo de aplicaciones web...

El objetivo de esta extensión es el de ejecutar componentes Java en el servidor.

SERVLETS

Especificación de más bajo nivel que extiende la funcionalidad de los servidores web con una arquitectura basada en componentes "ejecutables".

Es una aplicación Java que se ejecuta en el servidor gestionando y procesando peticiones HTTP.

Se ejecutan totalmente en el servidor bajo petición de un cliente. Remplazan a los programas de interfaz CGI.

Las ventajas es que tienen una mejor portabilidad y flexibilidad debido a que la conexión se basa en una API independiente de la plataforma definida en el estándar de Java. Es seguro ya que se ejecutan bajo un único proceso. Tienen un buen rendimiento ya que tienen su propio entorno de ejecución, se ejecutan y permanecen en memoria y se pueden precargar o cargar bajo demanda.

JSP: JAVASERVER PAGES

Tecnología Java de server side scripting que genera de manera dinámica contenidos. Un archivo JSP tiene código HTML nativo, elementos JSP, código funcional java y mecanismos de extensión de etiquetas.

Necesita un servidor web que soporte JSP, se solicitan igual que un documento HTML desde un navegador.

ARQUITECTURA .NET

Arquitectura de Microsoft para aplicaciones web. Consta de 3 componentes principales:

- Un entorno de aplicaciones independiente del lenguaje y optimizado para el entorno distribuido.
- Un entorno de desarrollo para la programación de aplicaciones: Visual Studio .NET
- Un sistema operativo que soporta entornos distribuidos y framework .NET:

Es una plataforma única compartida por múltiples lenguajes compilados a Microsoft Intermediate Language (MSIL)

MODELO VISTA CONTROLADOR (MVC)

- **Modelo**: Representa los datos de la aplicación y las funciones de negocio, los encapsula para hacer transparente su manejo al resto de la aplicación. Permite tener la funcionalidad de la aplicación independiente, y en cualquier tipo de soporte, aunque sean sistemas heredados.
- **Vista**: Realiza la presentación de los datos del modelo. Accede al modelo y adapta los datos presentados al cliente final. Permite múltiples vistas dependiendo del medio de salida de los datos.
- **Controlador**: Recibe las interacciones del usuario y las traslada en acciones sobre el modelo. Decide la salida que es necesaria presentar al usuario y permite realizar la navegación que se necesite para el usuario sin necesidad de alterar la vista ni el modelo.