



Recibe una cálida:

¡Bienvenida!

Te estábamos esperando 😊 

➤ JEE y su entorno

Plan formativo: Desarrollo de Aplicaciones Móviles Android Trainee V2.0

HOJA DE RUTA

¿Cuáles **skill** conforman el programa?



REPASO CLASE ANTERIOR

En la clase anterior trabajamos :

- ✓ El entorno JEE y sus componentes
- ✓ Arquitectura en capas

LEARNING PATHWAY

¿Sobre qué temas trabajaremos?

5.1

Start! 🏁

JEE y su entorno

Servlets

Servidores de aplicaciones

Ciclo de vida

OBJETIVOS DE APRENDIZAJE

¿Qué aprenderemos?



Reconocer qué es un servlet y para qué se utiliza



Comprender el concepto de servidor de aplicaciones



› Servlet



Servlet



¿Qué es?:



Es una tecnología que nos permite crear aplicaciones web interactivas (dinámicas), es decir, le permite al usuario interactuar con la aplicación (hacer consultas, insertar datos, modificarlos, etc). Un Servlet es un objeto Java que pertenece a una clase que extiende de `javax.servlet.http.HttpServlet`.



Son pequeños programas escritos en Java que admiten peticiones a través del protocolo HTTP. Los servlets reciben peticiones desde un navegador web, las procesan y devuelven una respuesta al navegador, normalmente en HTML.

Estos programas son los intermediarios entre el cliente (casi siempre un navegador web) y los datos (BBDD).



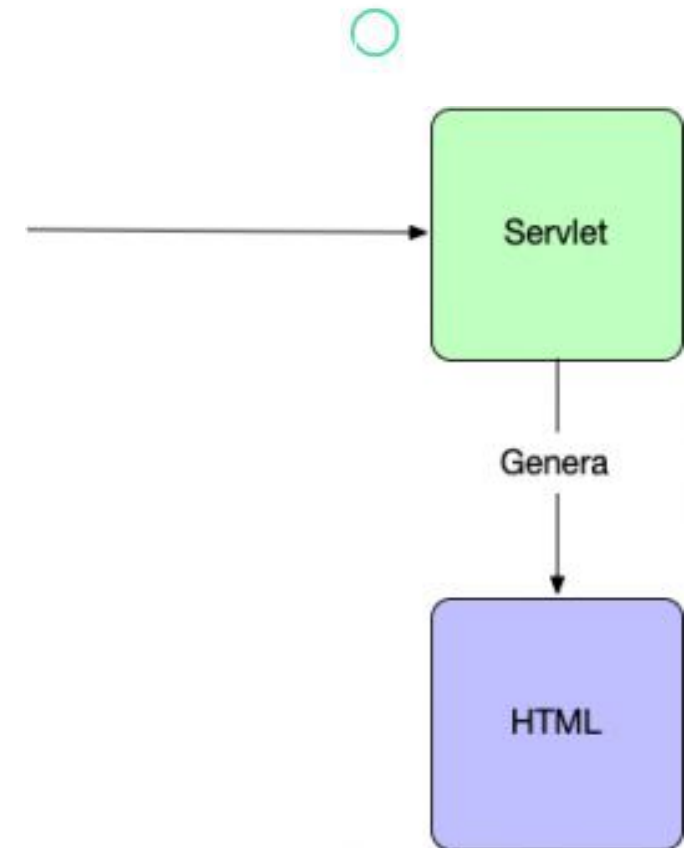


Servlet

Ciclo de vida:

Desde el momento que inicializamos el servlet hasta que el servlet es destruido, este pasará por una serie de estados dependiendo de cada una de las situaciones ante las que se encuentre. Muy por encima podríamos decir que la secuencia de acciones que se producen en un servlet son las siguientes:

- La primera vez que realicemos una petición sobre el servlet se ejecutará un método de inicio.
- Luego estaremos a la espera de peticiones. Dependiendo de como llegen los datos al servlet se ejecutará un método específico.
- Por último el servlet tendrá un estado de finalización en el cual eliminará las variables creadas en su inicialización, conexiones a bases de datos, etc. Finalmente se destruirá.



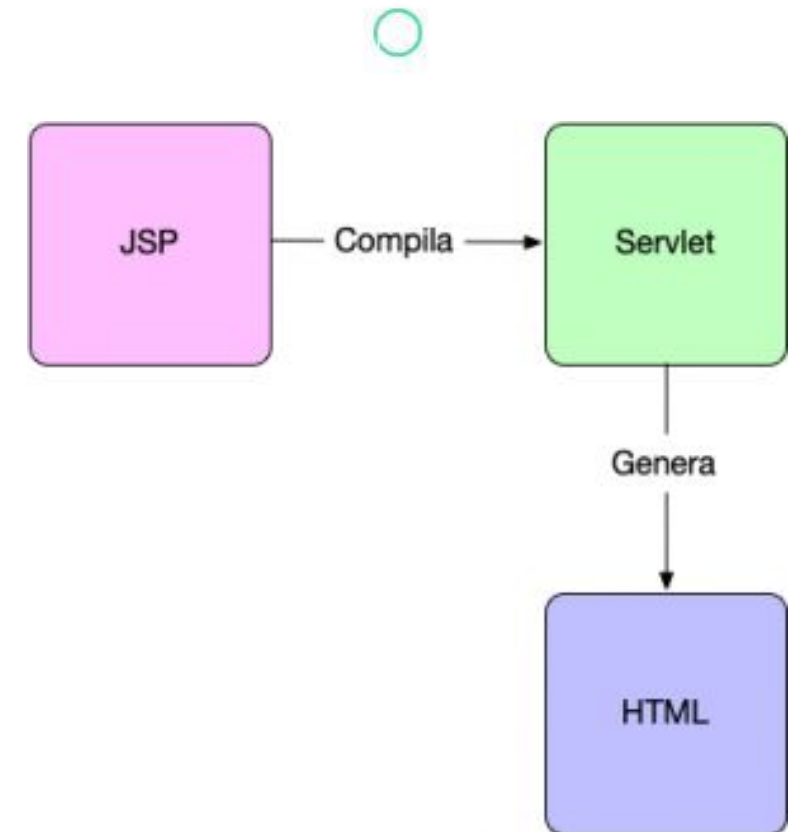


Servlet

¿Cómo funciona?

1. El navegador (cliente) pide una página al servidor HTTP que es un contenedor de Servlets.
2. El servlet procesa los argumentos de la petición, es decir, el contenedor de Servlets delega la petición a un Servlet en particular elegido de entre los Servlets que contiene.
3. El Servlet, que es una objeto java, se encarga de generar el texto de la página web que se entrega al contenedor.
4. El contenedor devuelve la página web al navegador (cliente) que la solicitó, normalmente en HTML.

Por lo tanto nos encontramos en una arquitectura Cliente-Servidor. En nuestro caso vamos a utilizar Apache Tomcat como contenedor de servlets.



› Servidor de aplicaciones



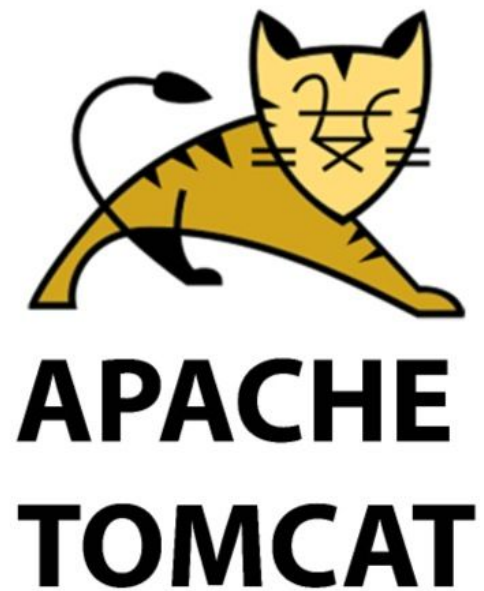
Servidor de aplicaciones

¿Qué es?:

Los servidores de aplicaciones **son programas de servidor en una red distribuida que proporciona el entorno de ejecución para un programa de aplicación**. Más específicamente, el servidor de aplicaciones es el componente de tiempo de ejecución principal en todas las configuraciones y donde una aplicación se ejecuta realmente.

El servidor de aplicaciones colabora con el servidor web para ofrecer una respuesta dinámica y personalizada a una solicitud de cliente.

Algunos ejemplos populares de servidores de aplicaciones JEE son *Apache Tomcat*, *JBoss/WildFly*, *IBM WebSphere Application Server* y *Oracle WebLogic Server*.

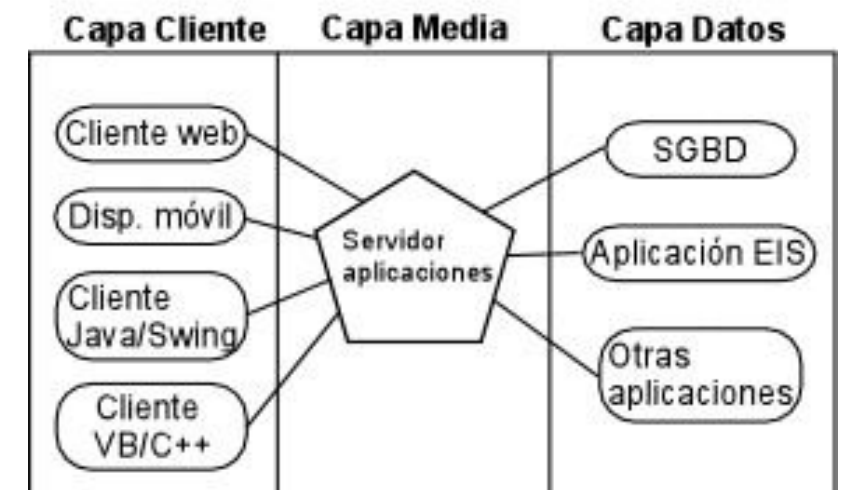
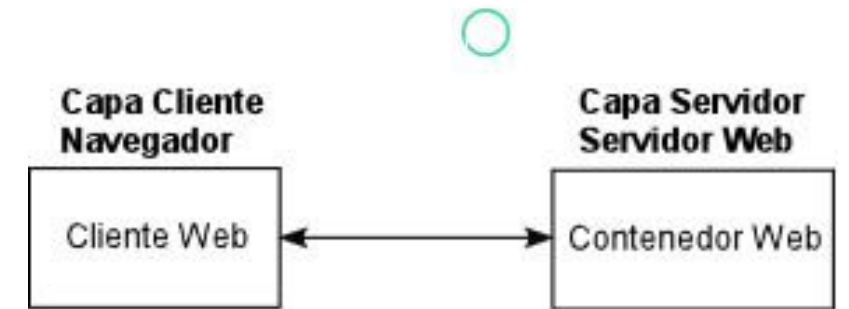




Servidor de aplicaciones

El concepto de servidor de aplicaciones está relacionado con el concepto de sistema distribuido. Un sistema distribuido, en oposición a un sistema monolítico, permite mejorar tres aspectos fundamentales en una aplicación: la alta disponibilidad, la escalabilidad y el mantenimiento.

- La alta disponibilidad hace referencia a que un sistema debe estar funcionando las 24 horas del día los 365 días al año.
- La escalabilidad es la capacidad de hacer crecer un sistema cuando se incrementa la carga de trabajo (el número de peticiones).
- El mantenimiento tiene que ver con la versatilidad a la hora de actualizar, depurar fallos y mantener un sistema. La solución al mantenimiento es la construcción de la lógica de negocio en unidades reusables y modulares.



LIVE CODING

Ejemplo en vivo

Subtítulo:

Breve descripción

1. *Descripción*
2. *Descripción*
3. *Descripción*
4. *Descripción*

5. *Descripción*
6. *Descripción*
7. *Descripción*
8. *Descripción*

 **Tiempo:**



Ejercicio N°

Nombre del ejercicio

○

¿Alguna consulta?

+



RESUMEN

¿Qué logramos en esta clase?

- ✓ Reconocer el concepto y funcionalidad de los servlets
- ✓ Comprender la función de un servidor de aplicaciones



#WorkingTime

Continuemos ejercitando

¡Antes de cerrar la clase! Te invitamos a: 🙌 🙌 🙌

1. Repasar nuevamente la grabación de esta clase
2. Revisar el material compartido en la plataforma de Moodle (lo que se vio en clase y algún ejercicio adicional)
 - a. *Lectura Modulo 5, Lección 1: página 3*
3. Traer al próximo encuentro, todas tus dudas y consultas para verlas antes de iniciar nuevo tema.

¡Muchas Gracias!

Nos vemos en la próxima clase 🙌



Momento: ✚

Time-out!

🕒 5 min.

