GC-F-004v.08

Análisis y desarrollo de software

FASE 3 Ejecución – GA7-AA2

Octubre 08 de 2025

@SENAComunica

www.sena.edu.co

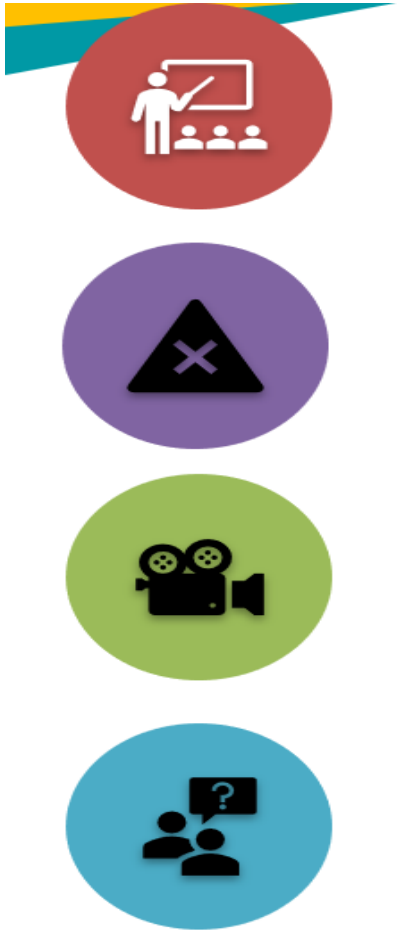
Agenda del día

1. Saludo de Bienvenida.
2. Recomendaciones Generales
3. Evidencias a desarrollar

   ✓**Evidencia de producto: GA7-220501096-AA2-EV02 módulos de software codificados y probados.**

4. Resolución de Dudas e Inquietudes
5. Despedida Equipo de Instructores

Recomendaciones generales

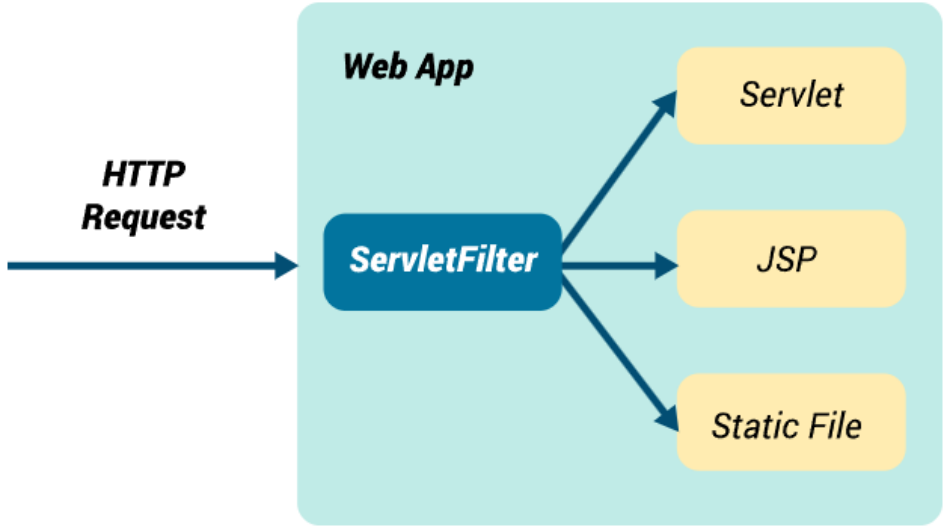


1. Estudiar el material de la semana antes de la videoconferencia.
2. Evita distractores durante la sesión.
3. Toma nota de las dudas y preguntas que surjan durante la sesión y ten presente las que te hayan surgido durante el estudio del material.
4. Al final de la sesión, se asignarán 10 minutos para resolver todas las inquietudes que se hayan presentado.

Servlets y JSP

Los *Servlets* y los JSP (*Java Server Pages*), son tecnologías que se utilizan para realizar ejecuciones en servidores Java, permitiendo extender los servicios de un servidor *web*.

Un ejemplo:
por medio de una página JSP se consigue hacer un formulario que contenga una serie de datos como: primer nombre, segundo nombre, edad, correo y, a través de un servlet.

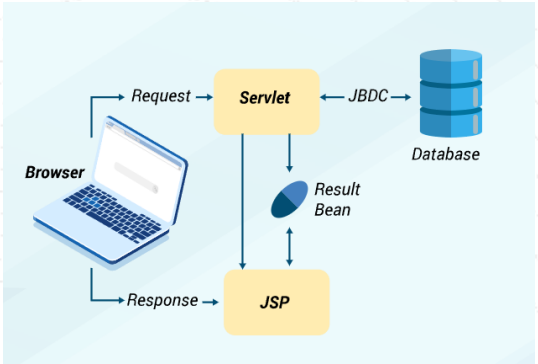


Procesos Principales

Los *servlets*
Adecuan el proceso de las peticiones que entran al servidor, procesan los datos de formularios, generan contenidos de varios tipos y redirecciona las peticiones.



JSP (*Java Server Pages*)
Es un código Java que se ejecuta en un servidor remoto, que lee las acciones del usuario desde formularios html, donde el servidor realiza consultas, usualmente a bases de datos, para obtener la información solicitada por el usuario y devolviendo una página *html* dinámica y en tiempo real.



Los archivos JSP
Están formados por código Java dentro de etiquetas **<html>** con extensión **.jsp** Estos archivos deben ser guardados en la carpeta del contenido *web* del servidor.
El código JSP inicia con **<% %>.**



Procesos Principales

Ejemplo
Este es un ejemplo sencillo de código JSP.

```
<%@ page language='java' contentType='text/html;
charset=iso-8859-1'%>
<%@ page import='java.util.Date' %>
<html>
 <head>
  <title>Hola Planeta</title>
 </head>
 <body>
  <p>Bienvenido a la página JSP. </p>
  <p>La hora del servidor es <%= new Date() %> </p>
          <% out.println("HOLA PLANETA"); %>
 </body>
</html>
```

***Software* requerido**
El *software* que se necesita para trabajar con JSP y *Servlets* es: **Java JEE IDE** (*NetBeans*, *Eclipse*), y el Servidor *Web* (*Tomcat*).

Otros requerimientos
Para crear ejecutar y probar *servlets* y páginas JSP, se debe configurar un entorno de desarrollo; las herramientas y pasos de configuración son: instalar el **JDK** (Kit de desarrollo De Java), instalar **IDE**
Netbeans en https://netbeans.apache.org/download/index.html, instalar Tomcat: Servidor de Aplicaciones, que es un *software* de código abierto; una vez terminada la configuración de las herramientas se inicia la introducción al tema de los *servlets*.





Tecnologías *Servlet* y JSP

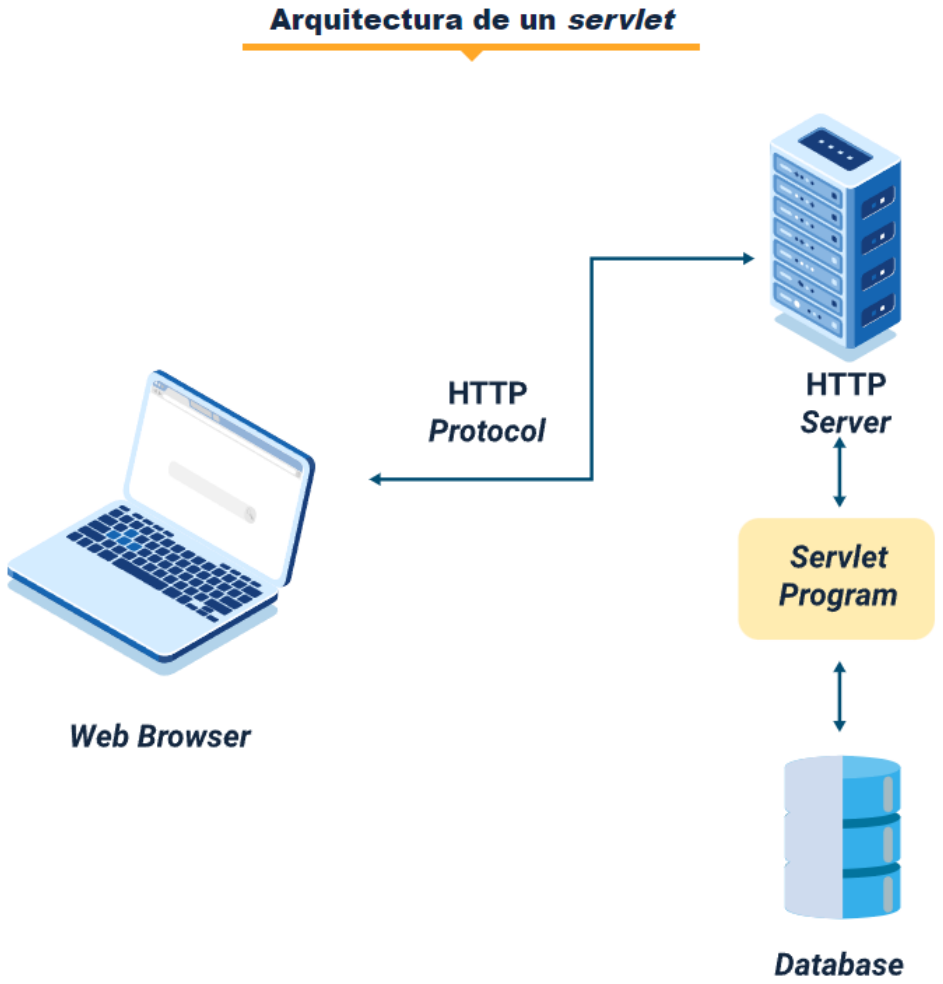
Que es Tomcat: Servidor de Aplicaciones



http://localhost:8080

Ventajas de utilizar *servlet*:

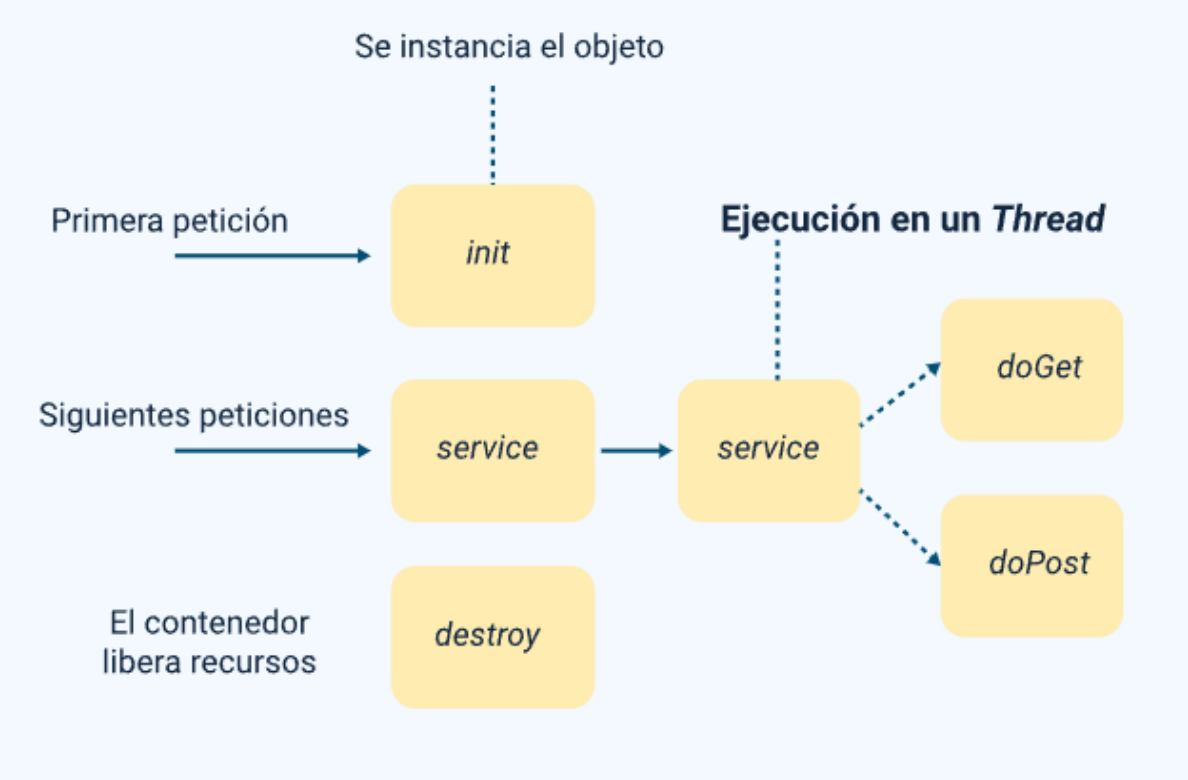
- Rendimiento: a diferencia de CGI ***(Common Gateway Interface)*** el rendimiento de un *servlet* es mucho mejor.
- No necesita procesos independientes para operar las solicitudes del cliente.
- Ya que los *servlet* están escritos en Java los hace una plataforma independiente.
- Creíble: resguarda los recursos del servidor debido a el servidor de seguridad de Java.



Las tareas principales de un *servlet* son:

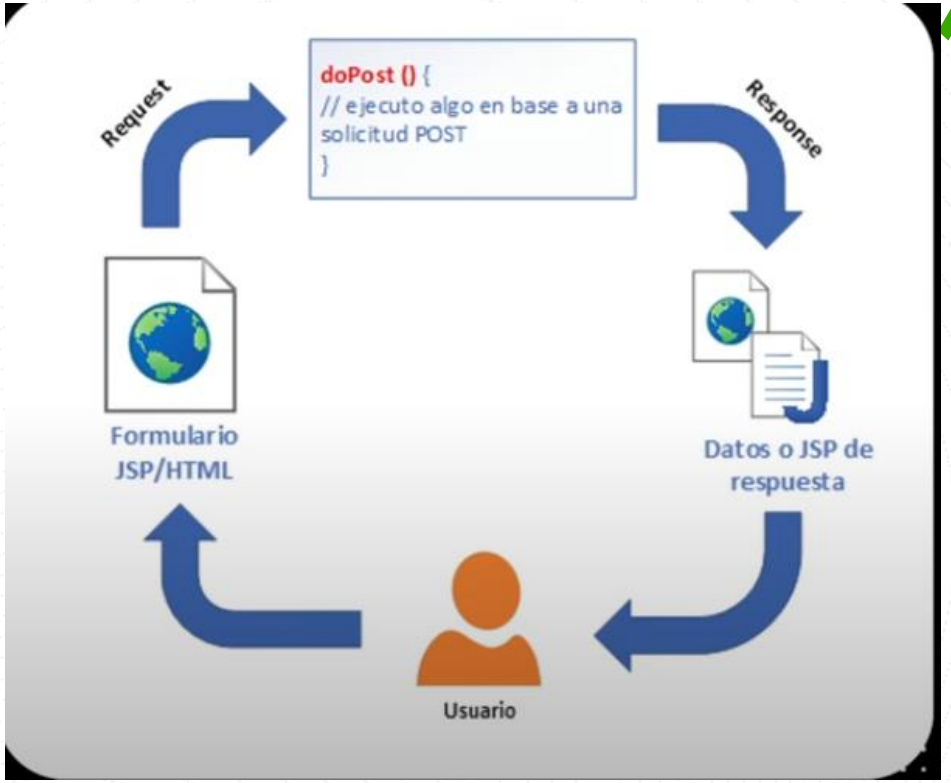
- **Datos de transmisión.**
- **Remisión de peticiones http**
- **Tratamiento de datos**
- **Envío de documentos**
- **Envíos de respuesta**

Ciclo de vida de un *Servlet*
A diferencia de las clases en Java, que inician con un método ***main (),*** en los *servlets* son "llamados" unos métodos, al recibir peticiones. A esta metodología se le llama ciclo de vida de un *servlet* y se compone por tres métodos: ***init, service, destroy*.**



Formularios *HTML* con *Servlets*

- Las páginas *web* contienen formularios *html* para la solicitud de información al usuario. El proceso de recolección de datos y respuesta, de este formulario, se debe hacer por medio de un programa.



La declaración de un formulario HTML se hace por medio de la etiqueta **<form>,** donde es primordial que se especifique el método por el que se realizará el envío de los datos, los cuales pueden ser *get* o *post*.

doGet(): Es el método encargado de recibir las solicitudes que provienen mediante GET.
doPost(): Es el método encargado de recibir las solicitudes que provienen mediante POST.

Apache Tomcat y Formulario Web con JSP

https://www.youtube.com/watch?v=_MLNWEd9GHQ https://www.youtube.com/watch?v=yCCjG4yGw7I

Evidencia de producto: GA7-220501096-AA2-EV02 módulos de software codificados y probados

Teniendo en cuenta las características del software a desarrollar realizar la codificación del módulo del proyecto enfocándose a web con servlets según lo visto en el componente formativo “Construcción de aplicaciones con JAVA”.

Elementos para tener en cuenta:

● Para la codificación del módulo debe tener en cuenta los artefactos del ciclo del software realizados con anterioridad: diagrama de clases, diagramas de casos de uso, historias de usuario, diseños, prototipos, Informe técnico de plan de trabajo para construcción de software con tecnologías seleccionadas etc**.**

● Se debe crear el proyecto utilizando herramientas de versionamiento.

● El código debe contener: formularios HTML con servlets

● Utilizar métodos get y pos

● Utilizar elementos de JSP

Lineamientos generales para la entrega de la evidencia:

● **Productos para entregar:** carpeta comprimida que debe tener los siguientes archivos: ● Archivos del proyecto,

● Archivo con enlace del repositorio, la carpeta comprimida debe tener el nombre del aprendiz y número de la evidencia así: NOMBREAPELLIDO_AA2_EV02

● **Extensión:** ZIP, RAR.

● Para hacer el envío de la evidencia remítase al área de la actividad correspondiente y acceda al espacio:

módulos de software codificados y probados GA7-220501096-AA2-EV02.

Criterios de evaluación

• Codifica los módulos del software Stand alone, web y móvil, de acuerdo con las especificaciones del diseño y el estándar de codificación.

2. LISTA DE CHEQUEO

No.	VARIABLES/INDICADORES DE LOGRO	CUMPLE		Observaciones
		SI	NO	
1.	El código contiene formularios HTML con servlets			30%
2.	Utiliza los métodos get y pos para uso de parámetros			30%
3.	Implementa elementos de JSP			30%
4.	Utiliza herramientas para versionamiento del código			10%

Plazo Actividades:

08 al 19 de Octubre



G R A C I A S

Línea de atención al ciudadano: 01 8000 910270
Línea de atención al empresario: 01 8000 910682

@SENAComunica

www.sena.edu.co