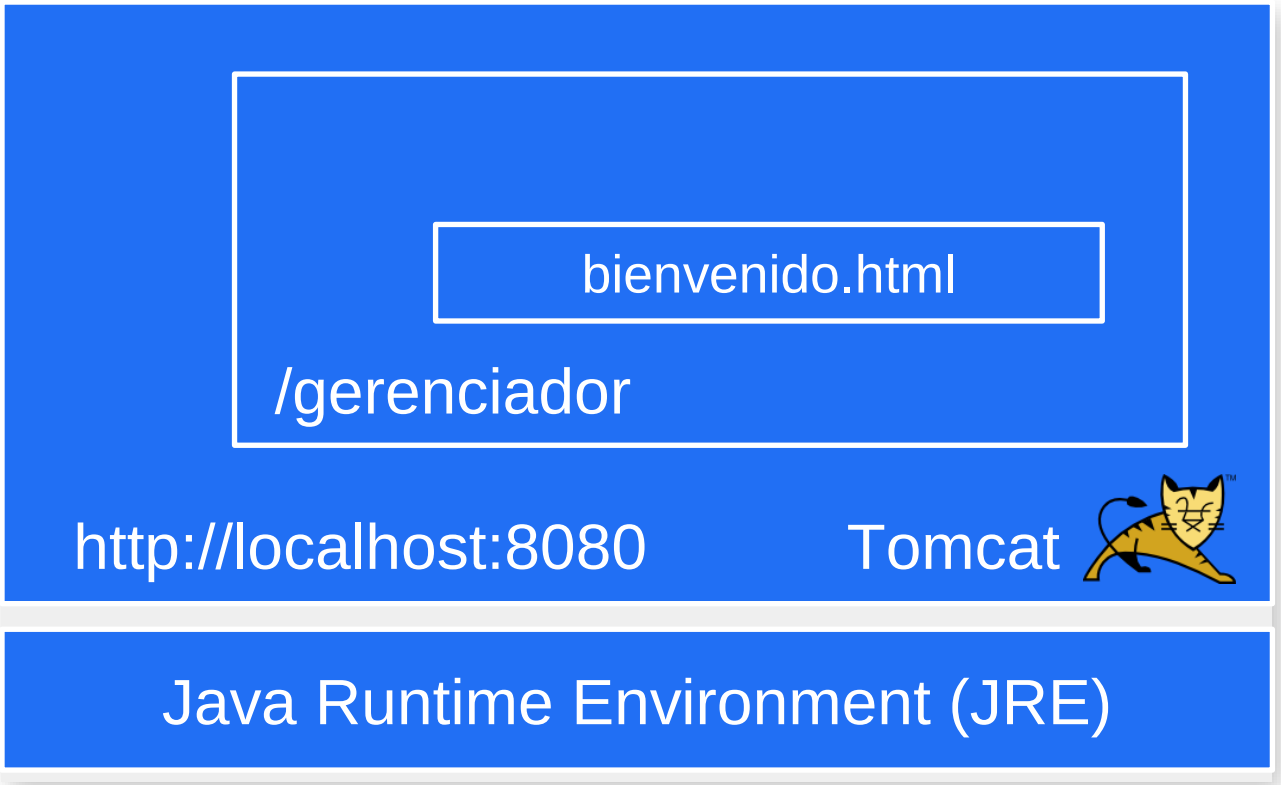
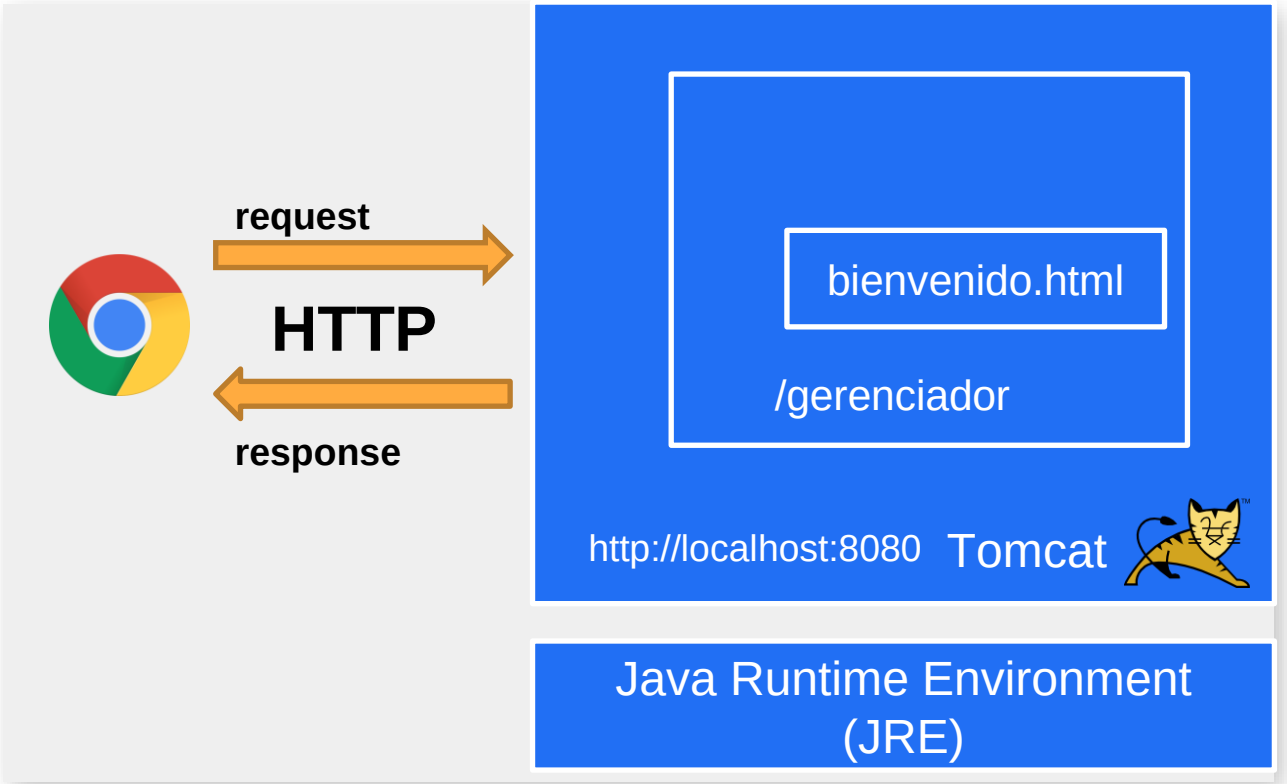
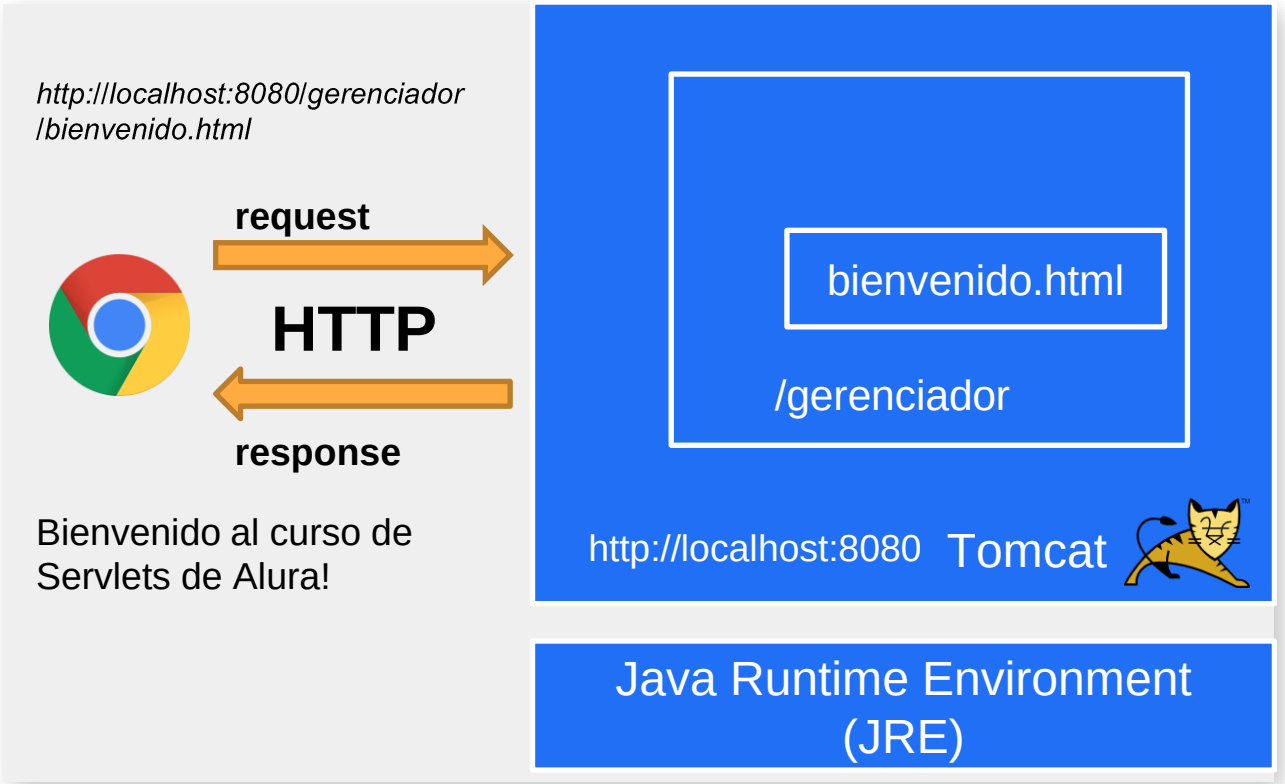
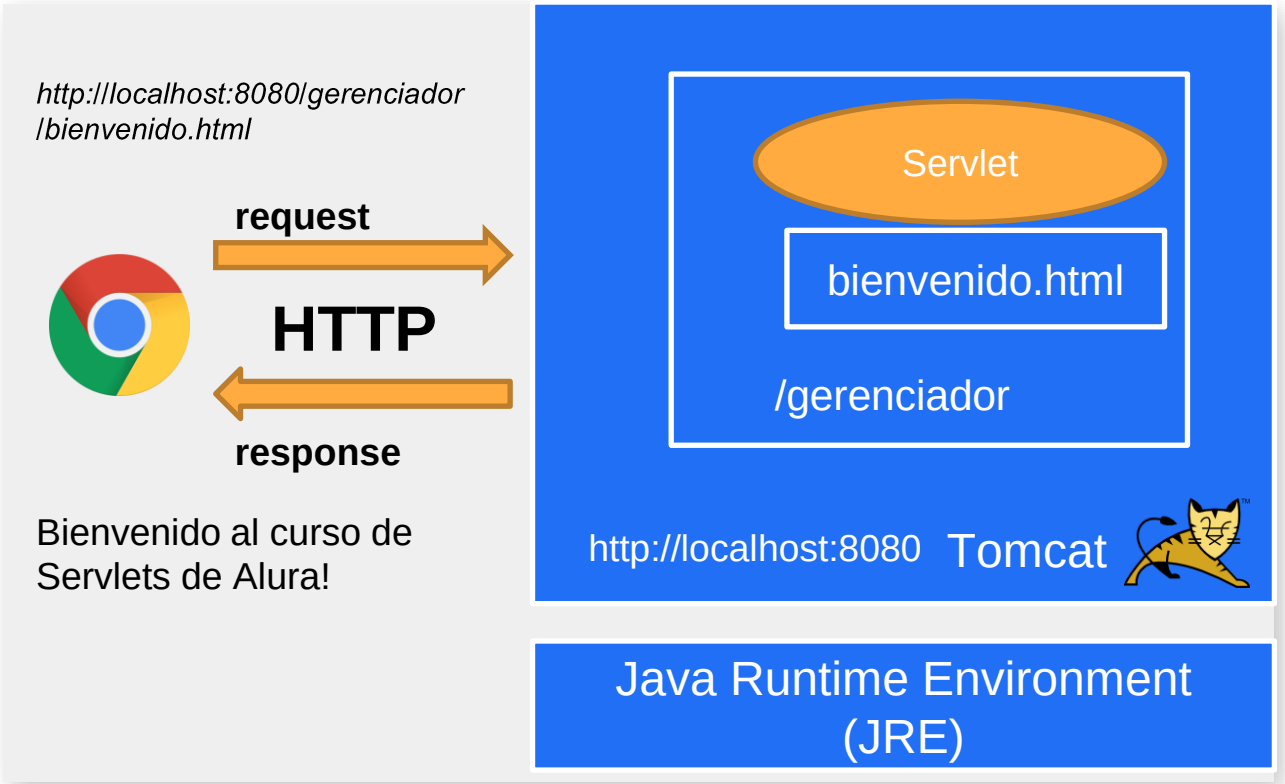
# Servlets.

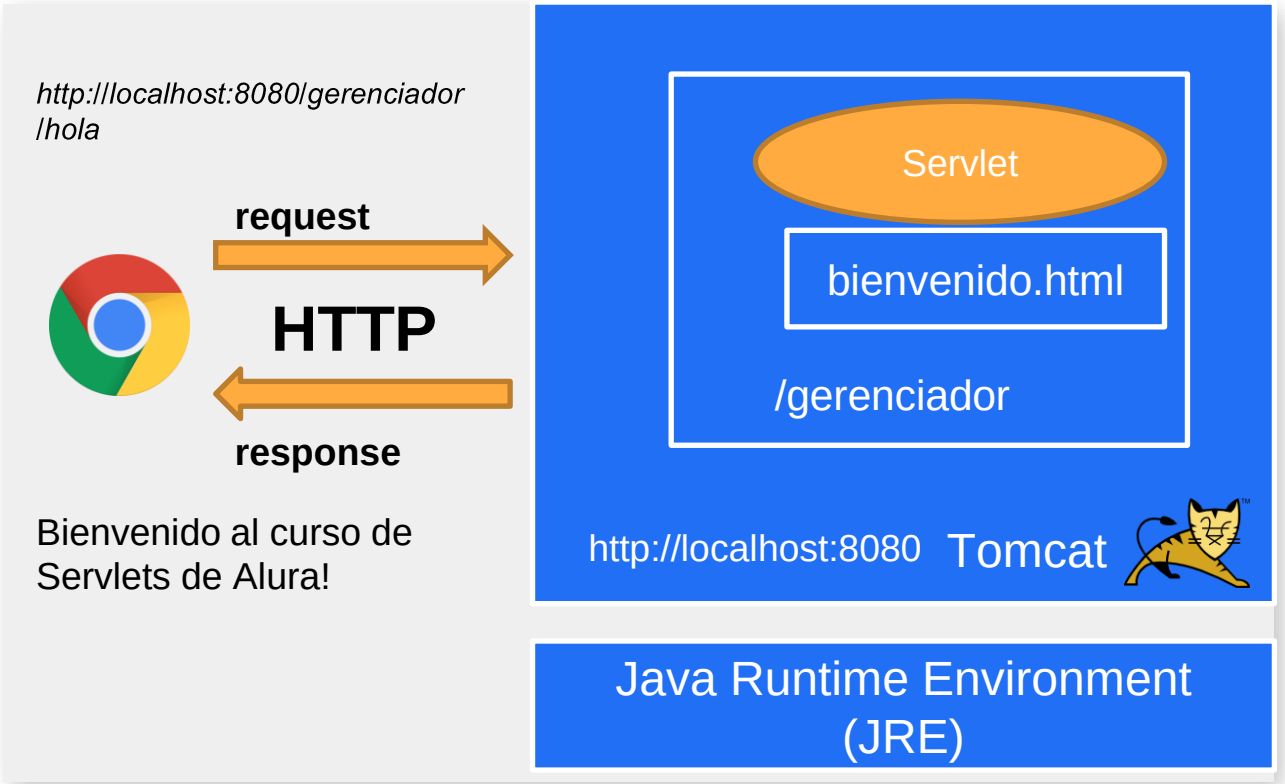

HTTP
HTML
JAVA


bienvenido.html
/gerenciador
http://localhost:8080
Tomcat
Java Runtime Environment (JRE)


request
HTTP
response
bienvenido.html
/gerenciador
http://localhost:8080
Tomcat
Java Runtime Environment
(JRE)

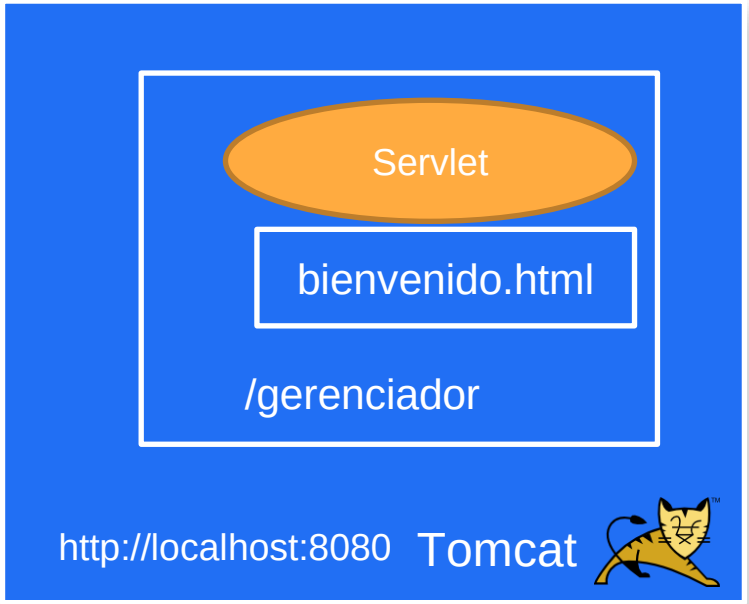

http:///localhost:8080/gerenciador
/bienvenido.html
request
HTTP
response
Bienvenido al curso de
Servlets de Alura!
bienvenido.html
/gerenciador
http://localhost:8080
Tomcat
Java Runtime Environment
(JRE)

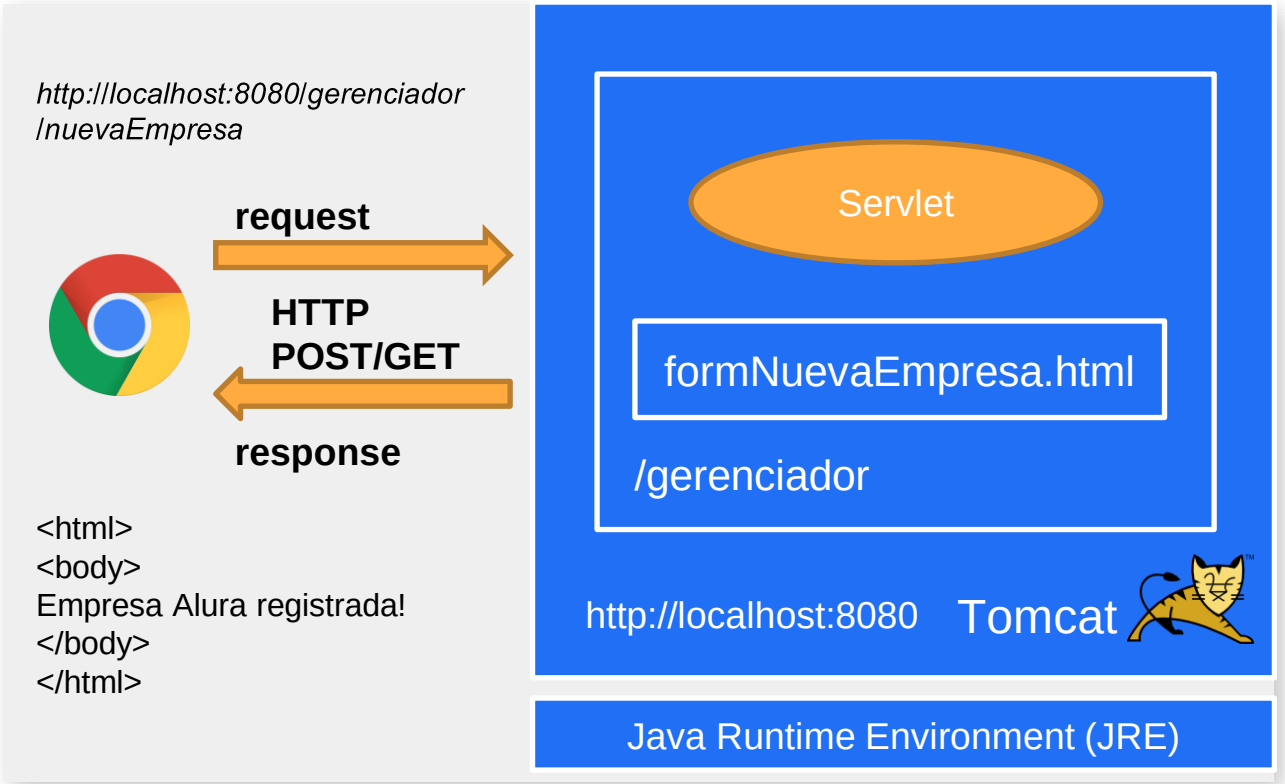

http:///localhost:8080/gerenciador
/bienvenido.html
request
HTTP
response
Bienvenido al curso de
Servlets de Alura!
Servlet
bienvenido.html
/gerenciador
http://localhost:8080
Tomcat
Java Runtime Environment
(JRE)

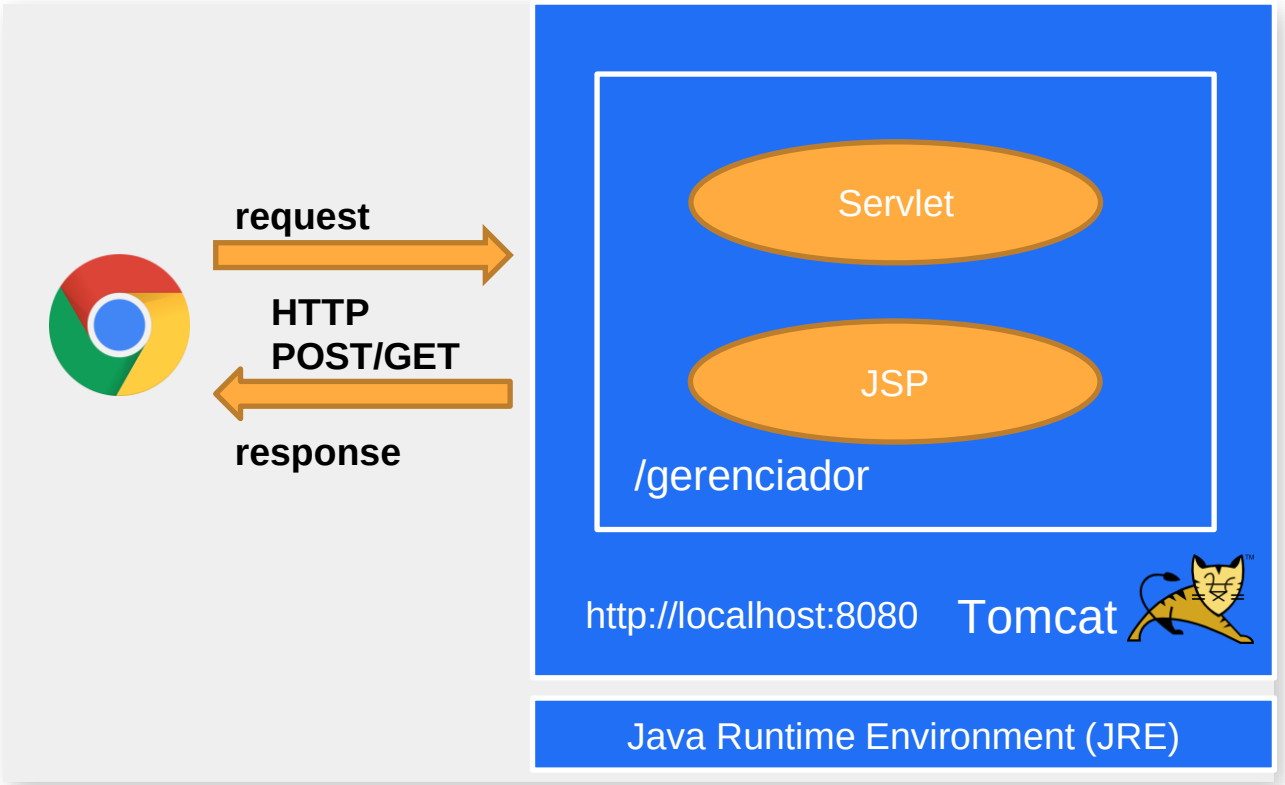

http:///localhost:8080/gerenciador/hola
request
HTTP
response
Bienvenido al curso de Servlets de Alura!
Servlet
bienvenido.html
/gerenciador
http://localhost:8080
Tomcat
Java Runtime Environment (JRE)

*http:///localhost:8080/gerenciador/hola*

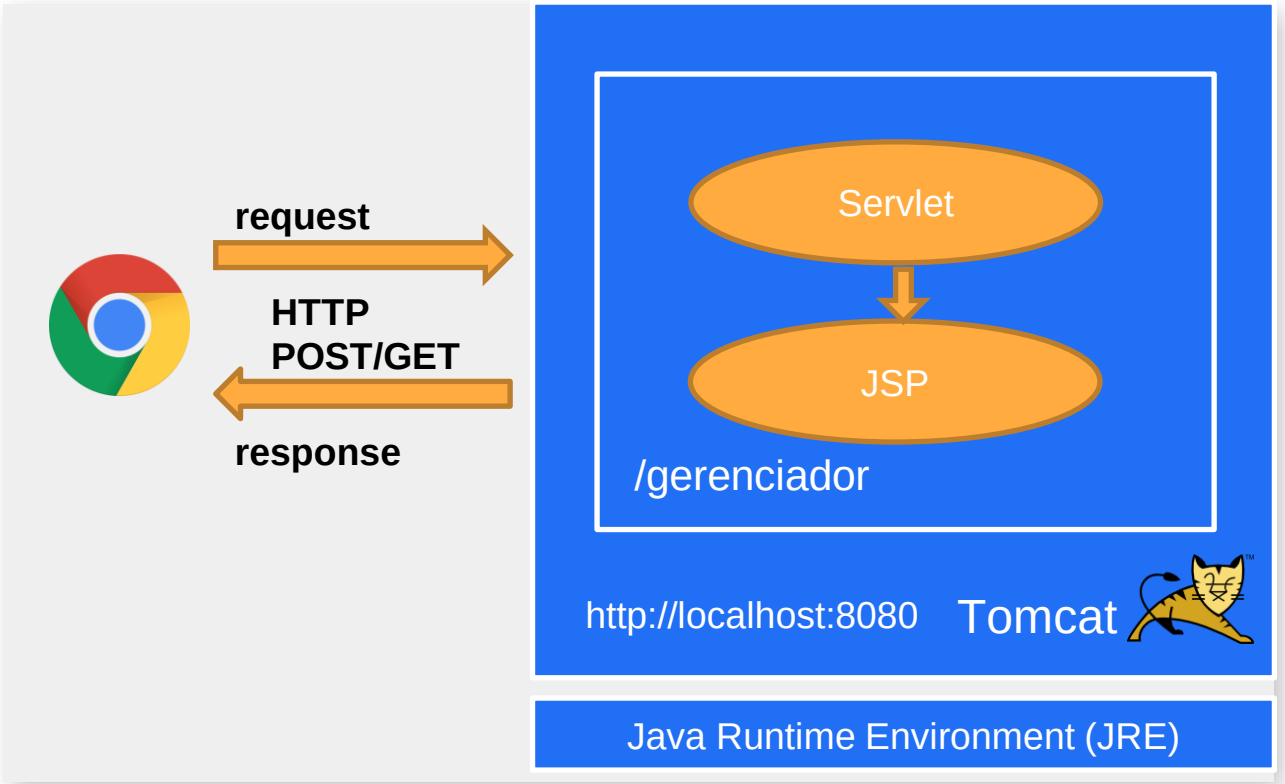

```
<html>
<body>
Hola Mundo! Felicitaciones por crear tu primer Servlet!
</body>
</html>
```



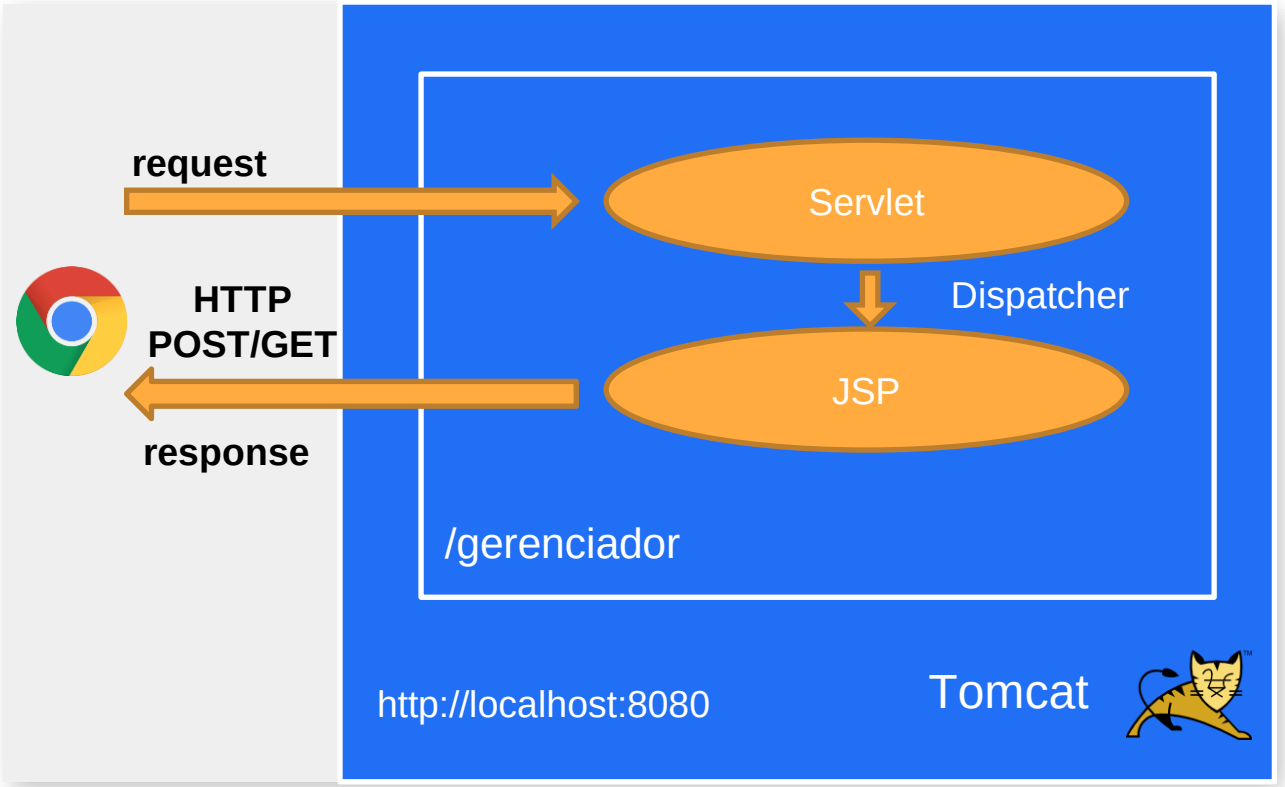

http:///localhost:8080/gerenciador
/nuevaEmpresa
request
HTTP
POST/GET
response
<html>
<body>
Empresa Alura registrada!
</body>
</html>
Servlet
formNuevaEmpresa.html
/gerenciador
http://localhost:8080
Tomcat
Java Runtime Environment (JRE)

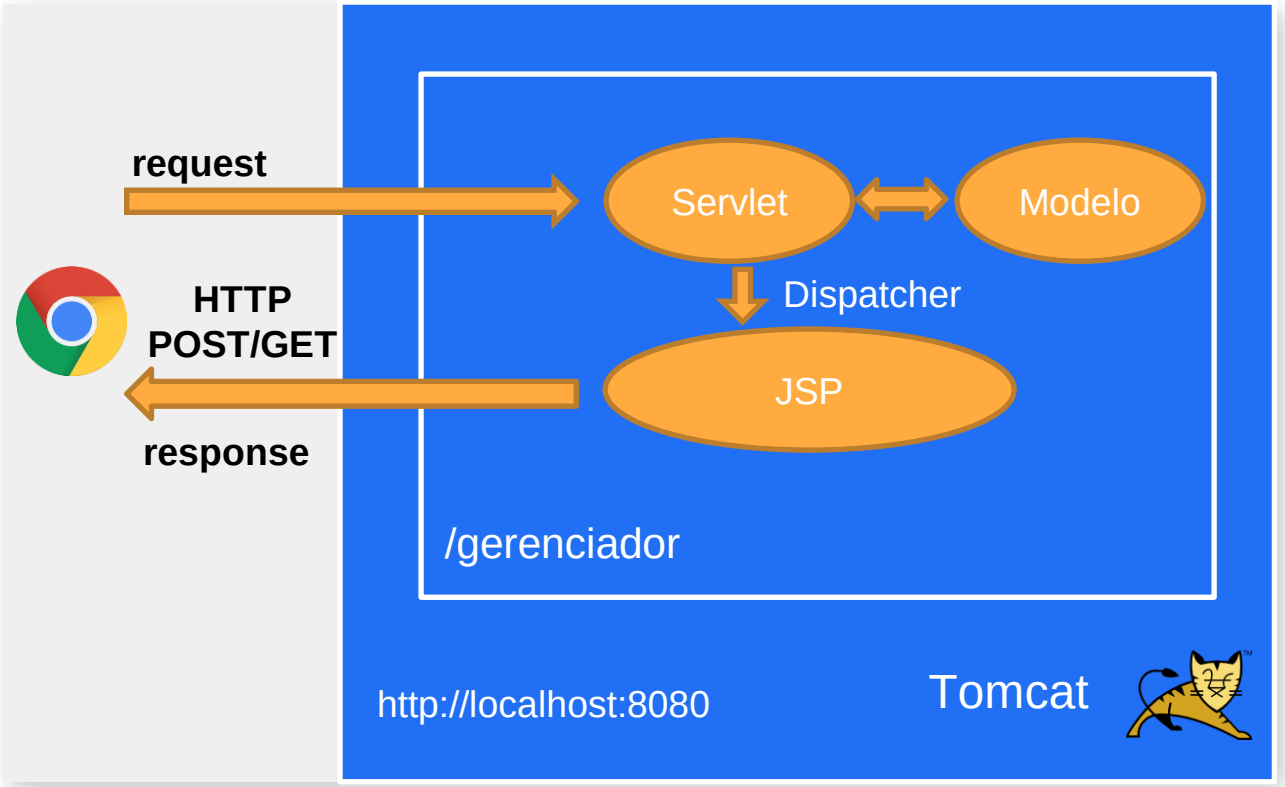


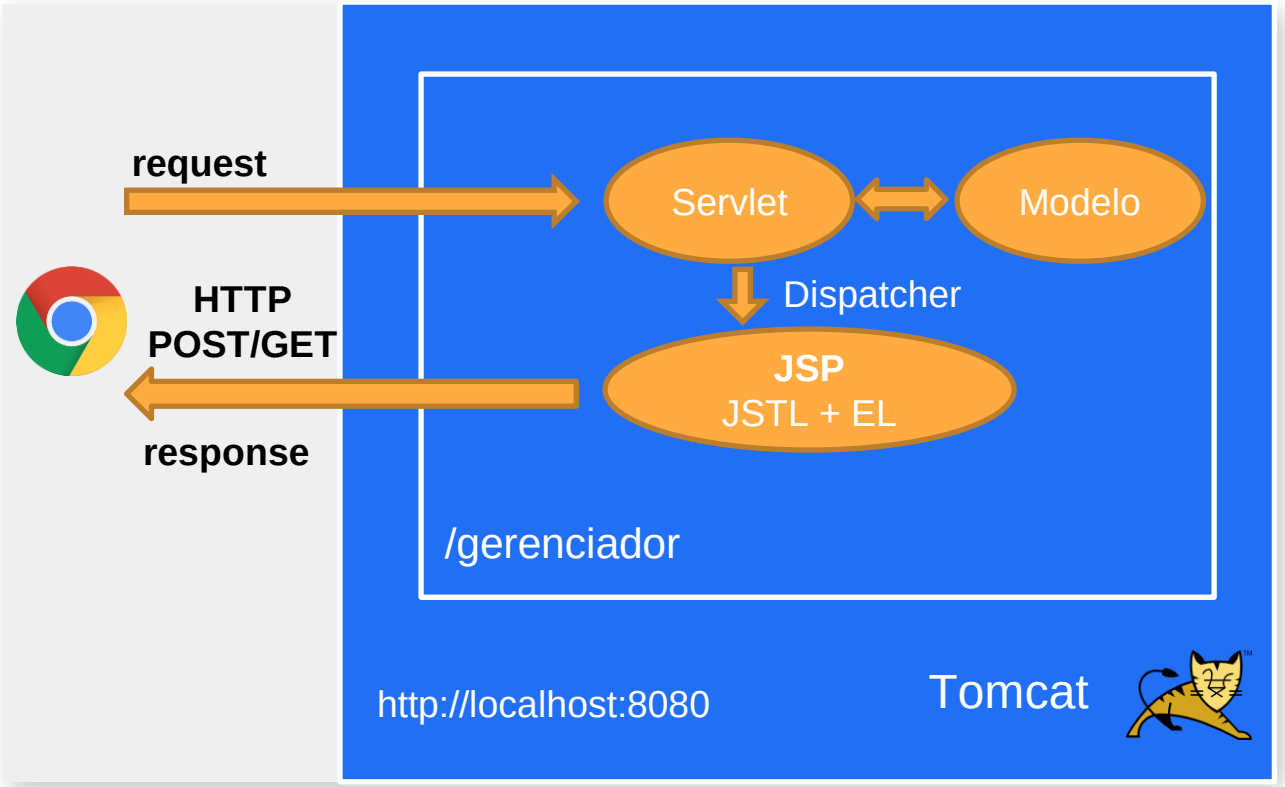

# JSP – Java Server Page




# JSP – Java Server Page


request
HTTP
POST/GET
response
Servlet
Dispatcher
JSP
/gerenciador
http://localhost:8080
Tomcat


request
HTTP
POST/GET
response
Servlet
Modelo
Dispatcher
JSP
/gerenciador
http://localhost:8080
Tomcat


request
HTTP
POST/GET
response
Servlet
Modelo
Dispatcher
JSP
JSTL + EL
/gerenciador
http://localhost:8080
Tomcat

# JSTL (Java Standard Tag Library).

- core – control de flujo
- fmt – formato /i18n (internacionalización)
- sql – ejecutar SQL
- xml – generar XML

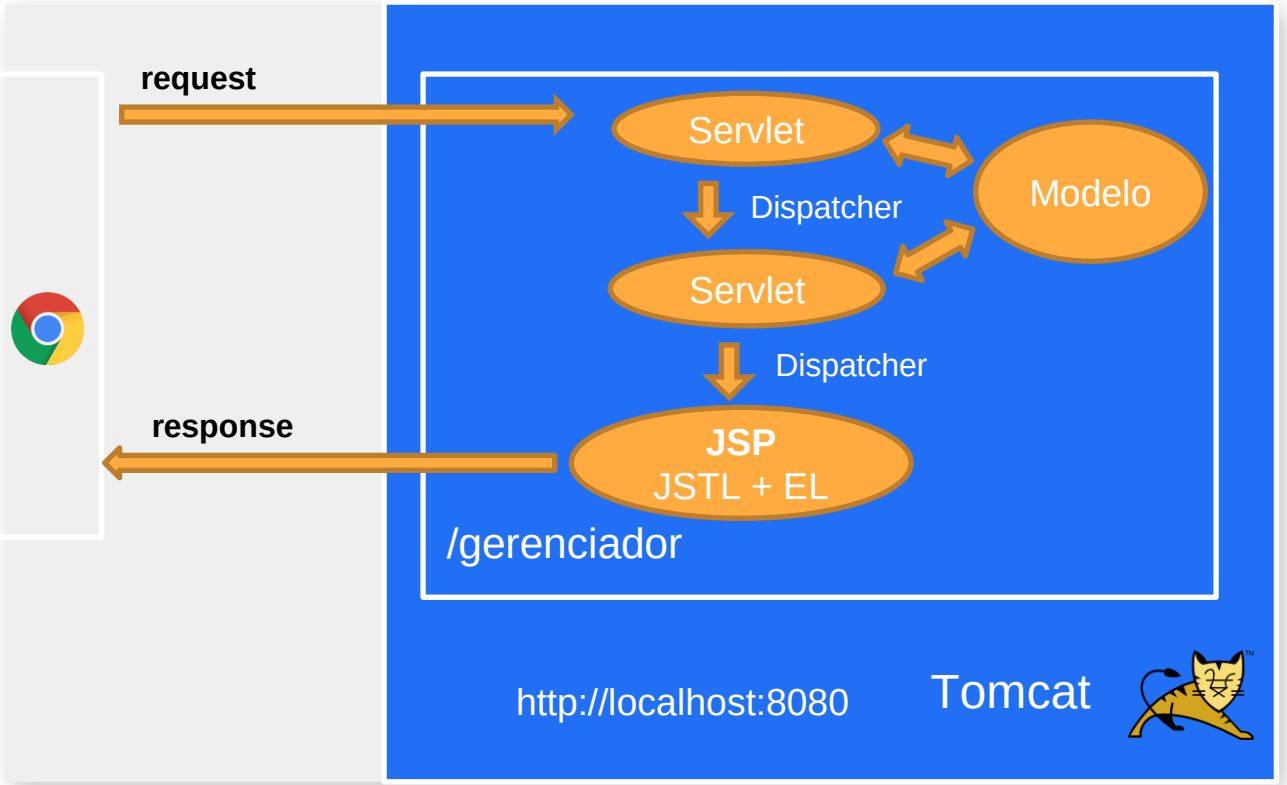
# JSTL (Java Standard Tag Library).

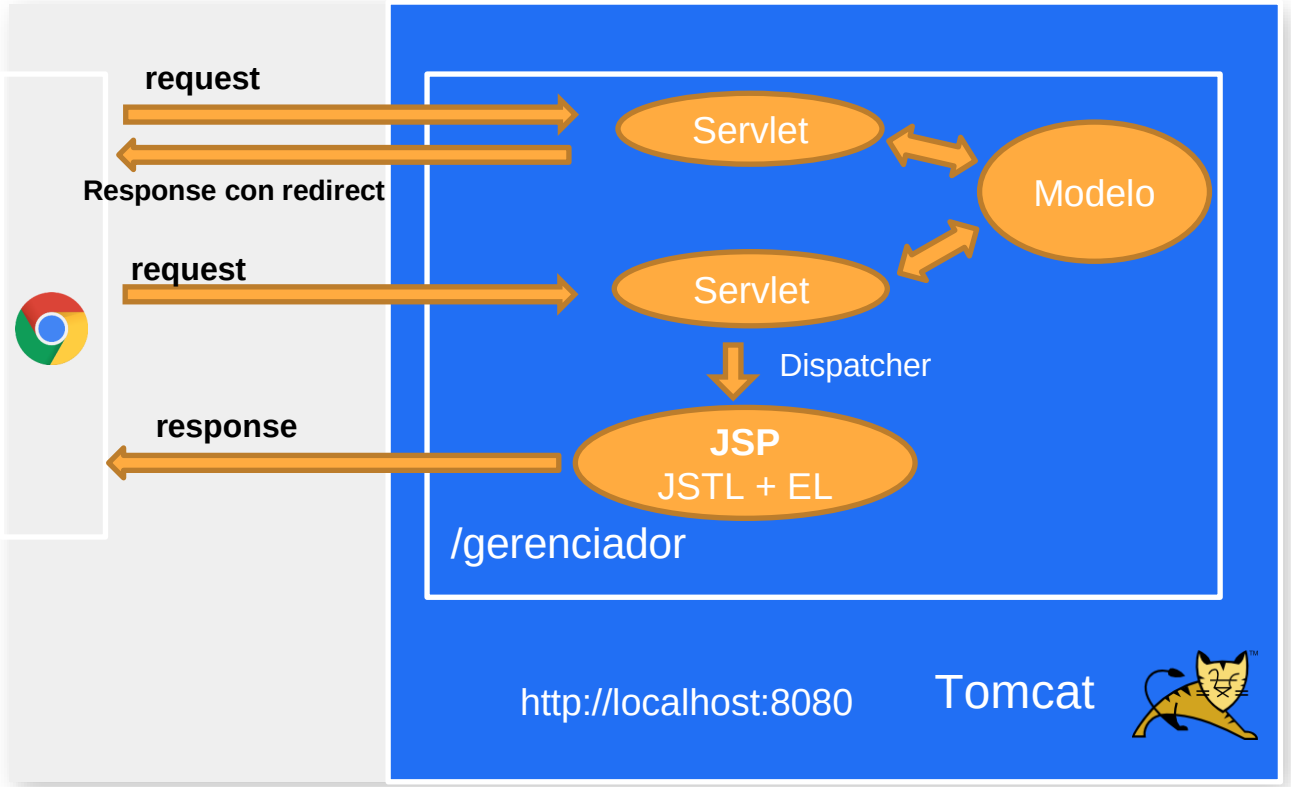
- core – control de flujo

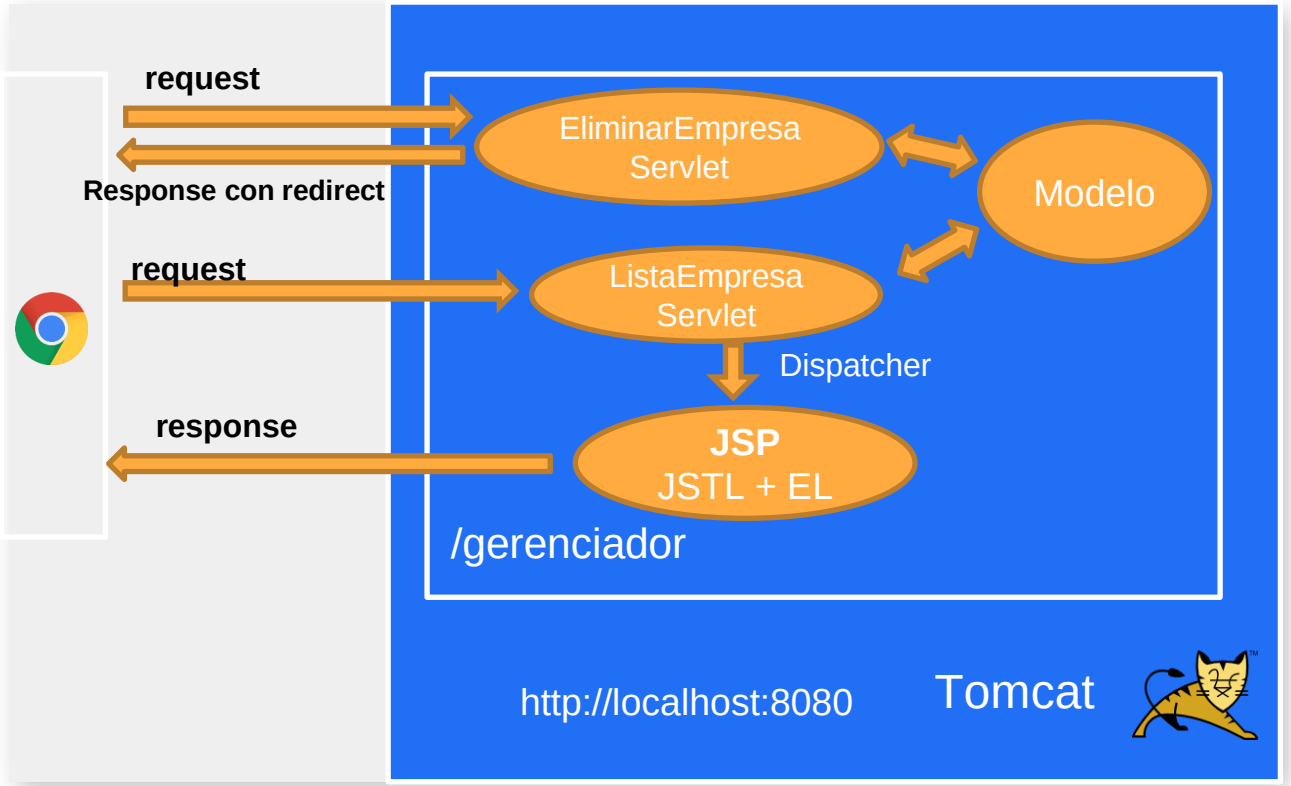
```
<%@ taglib uri = "http://java.sun.com/jsp/jstl/core" prefix = "c"%>
```

- fmt – formato /i18n (internacionalización)

```
<%@ taglib uri = "http://java.sun.com/jsp/jstl/fmt" prefix = "fmt"%>
```

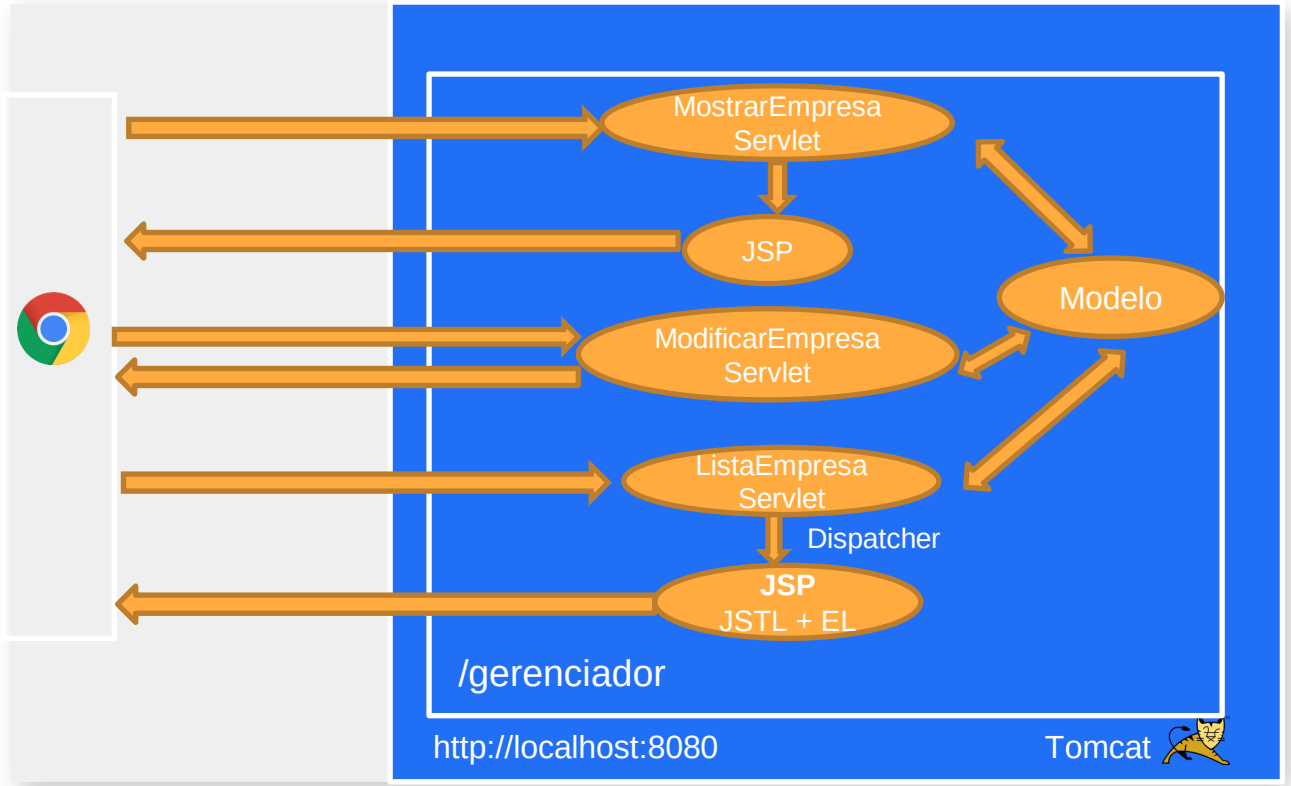

request
response
Servlet
Dispatcher
Servlet
Dispatcher
Modelo
JSP
JSTL + EL
/gerenciador
http://localhost:8080
Tomcat

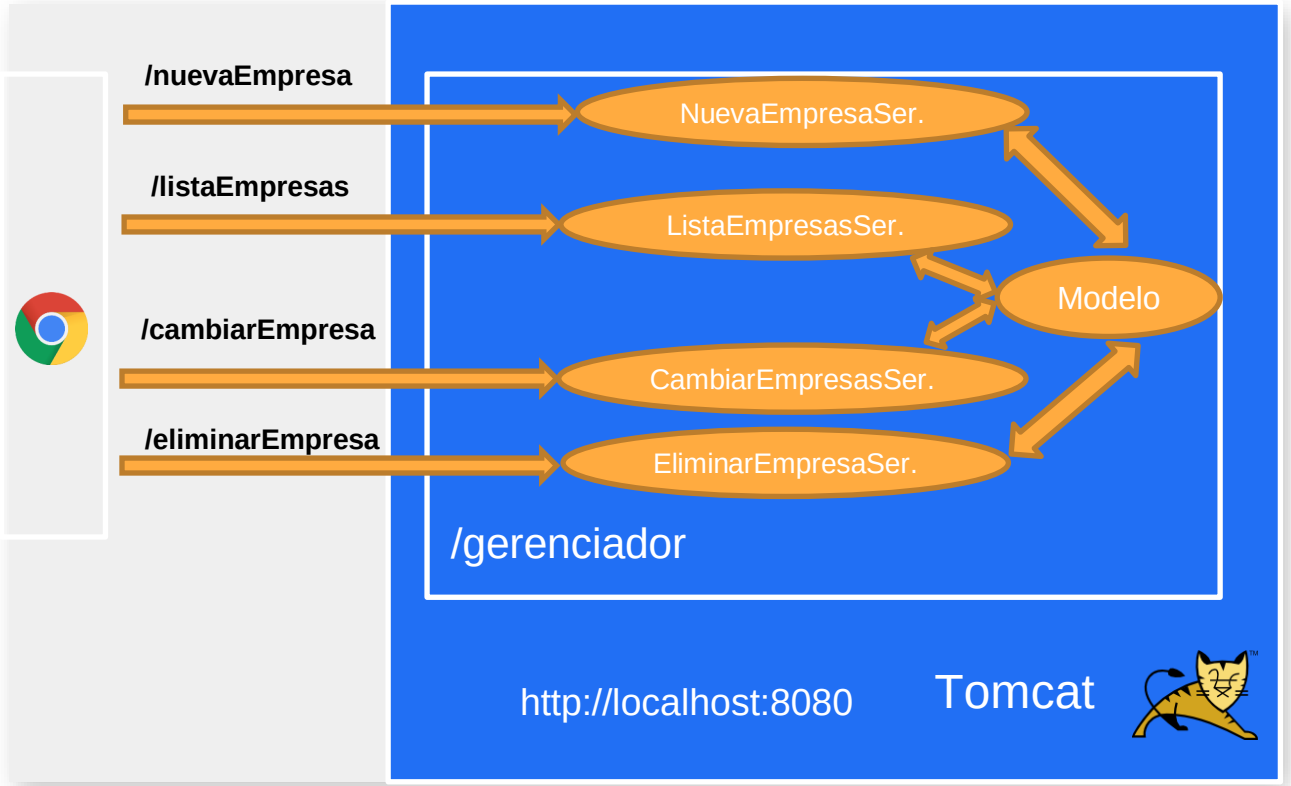

request
Response con redirect
request
response
Servlet
Modelo
Servlet
Dispatcher
JSP
JSTL + EL
/gerenciador
http://localhost:8080
Tomcat

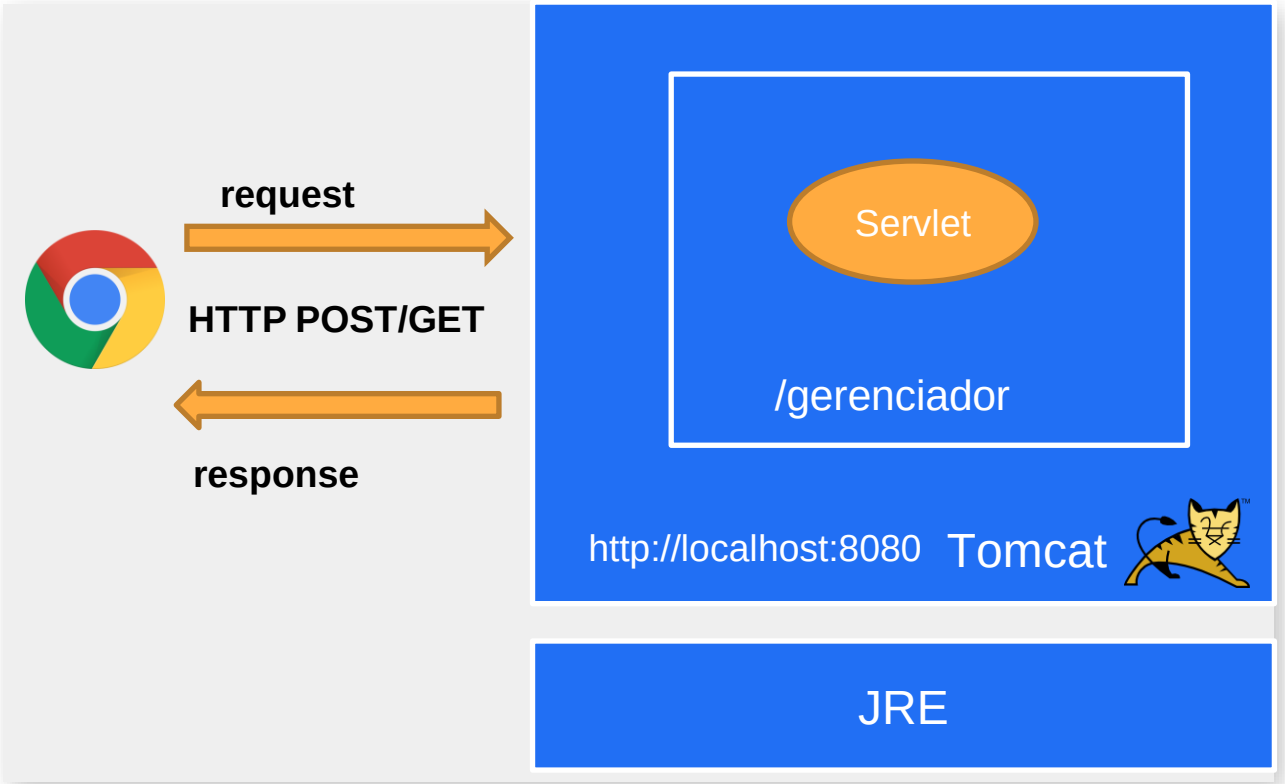

request
Response con redirect
request
response
EliminarEmpresa Servlet
Modelo
ListaEmpresa Servlet
Dispatcher
JSP
JSTL + EL
/gerenciador
http://localhost:8080
Tomcat

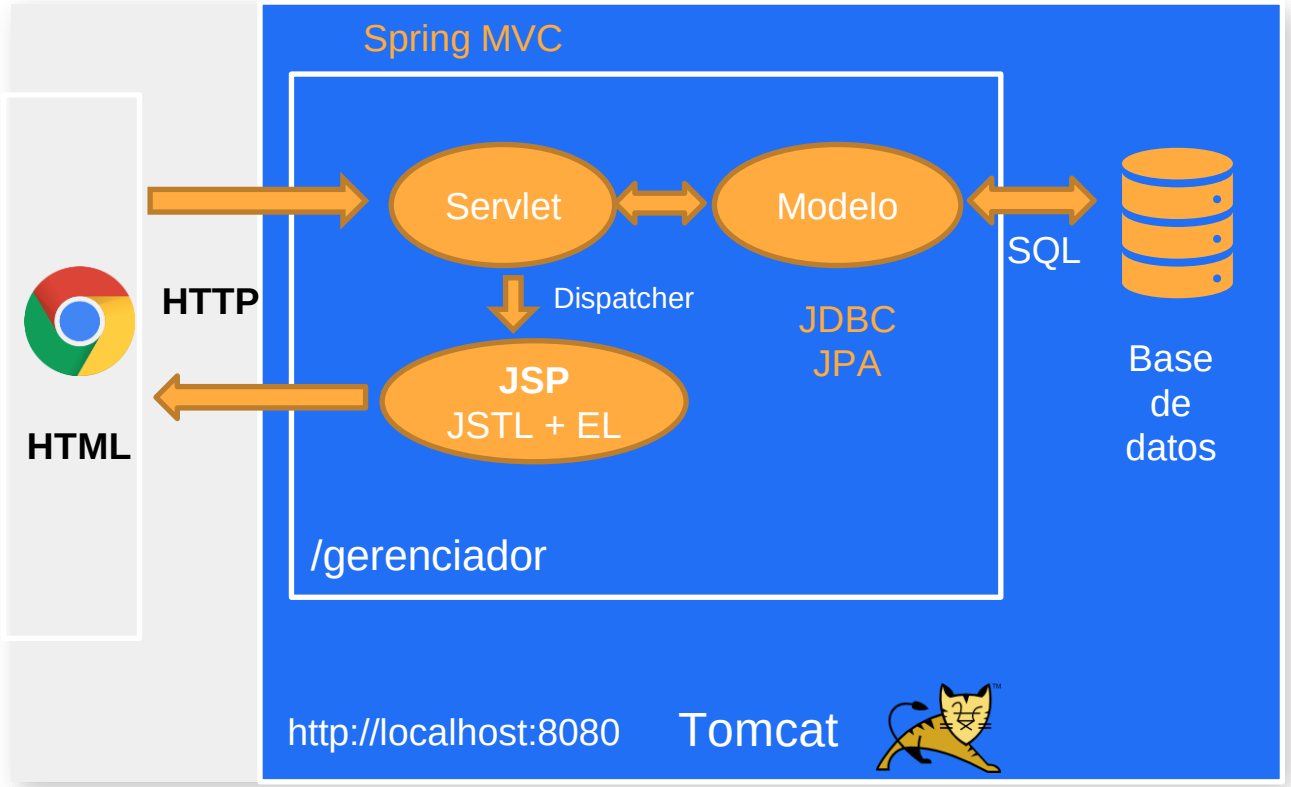
# CRUD.

- Create – creación de registro/objeto
- Read – lectura de registro(s), objeto(s)
- Update – actualizar registro/objeto
- Delete – eliminar registro/objeto

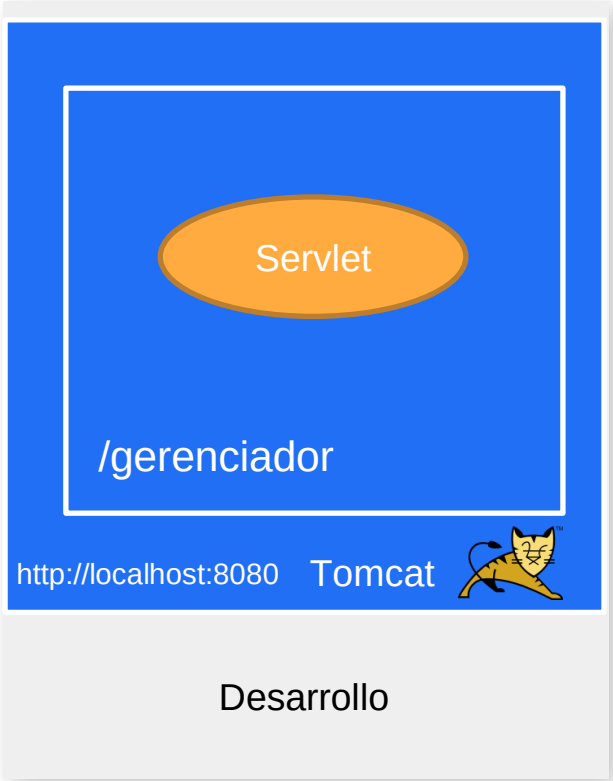

MostrarEmpresa
Servlet
JSP
Modelo
ModificarEmpresa
Servlet
ListaEmpresa
Servlet
Dispatcher
JSP
JSTL + EL
/gerenciador
http://localhost:8080
Tomcat


/nuevaEmpresa
/listaEmpresas
/cambiarEmpresa
/eliminarEmpresa
NuevaEmpresaSer.
ListaEmpresasSer.
Modelo
CambiarEmpresasSer.
EliminarEmpresaSer.
/gerenciador
http://localhost:8080
Tomcat

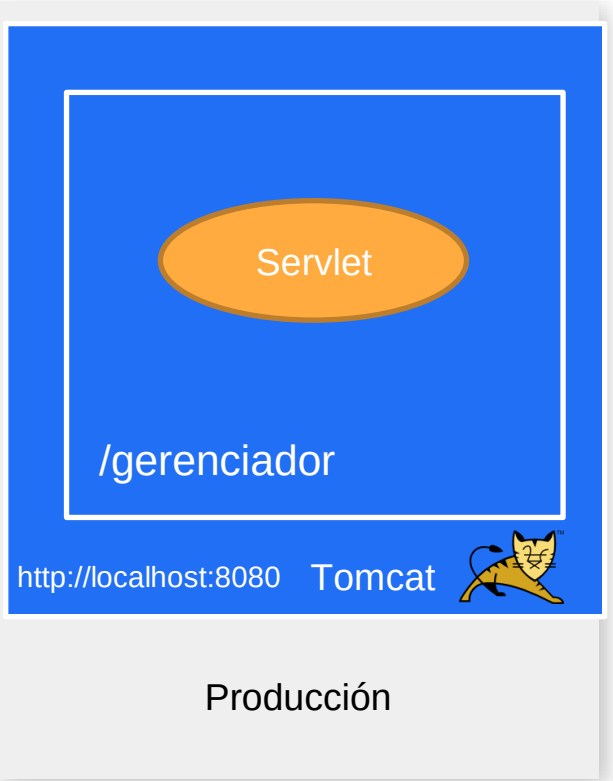

request
HTTP POST/GET
response
Servlet
/gerenciador
http://localhost:8080
Tomcat
JRE


Spring MVC
Servlet
Modelo
SQL
Dispatcher
JDBC
JPA
HTTP
JSP
JSTL + EL
HTML
Base
de
datos
/gerenciador
http://localhost:8080
Tomcat

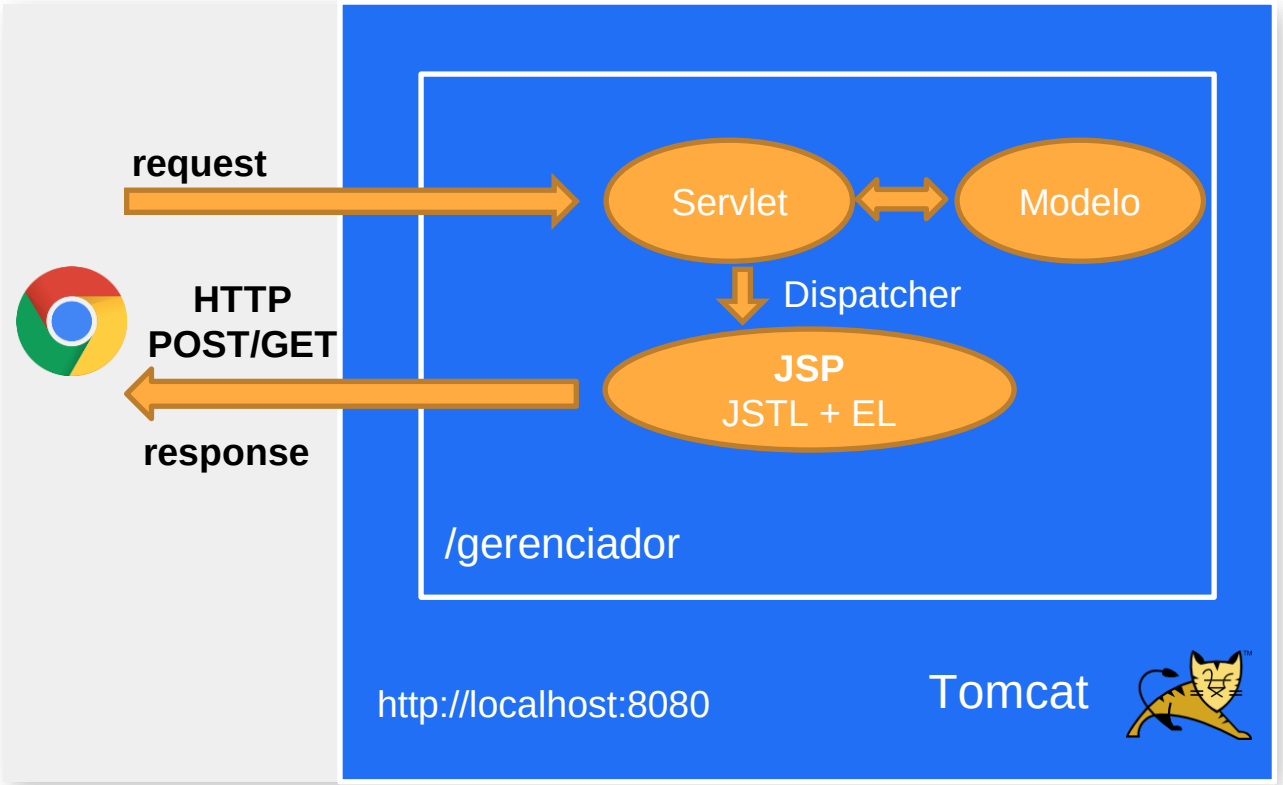
# Deploy WAR – Web ARchive

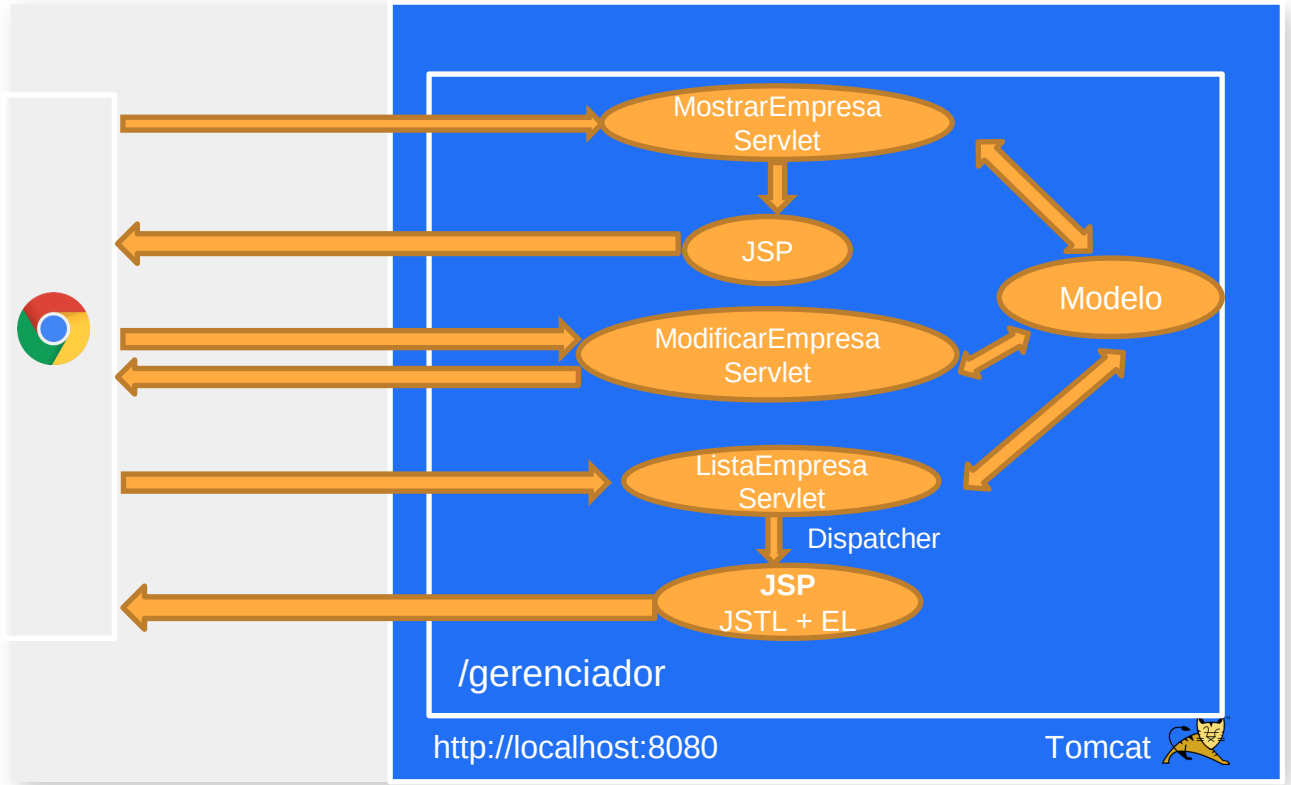


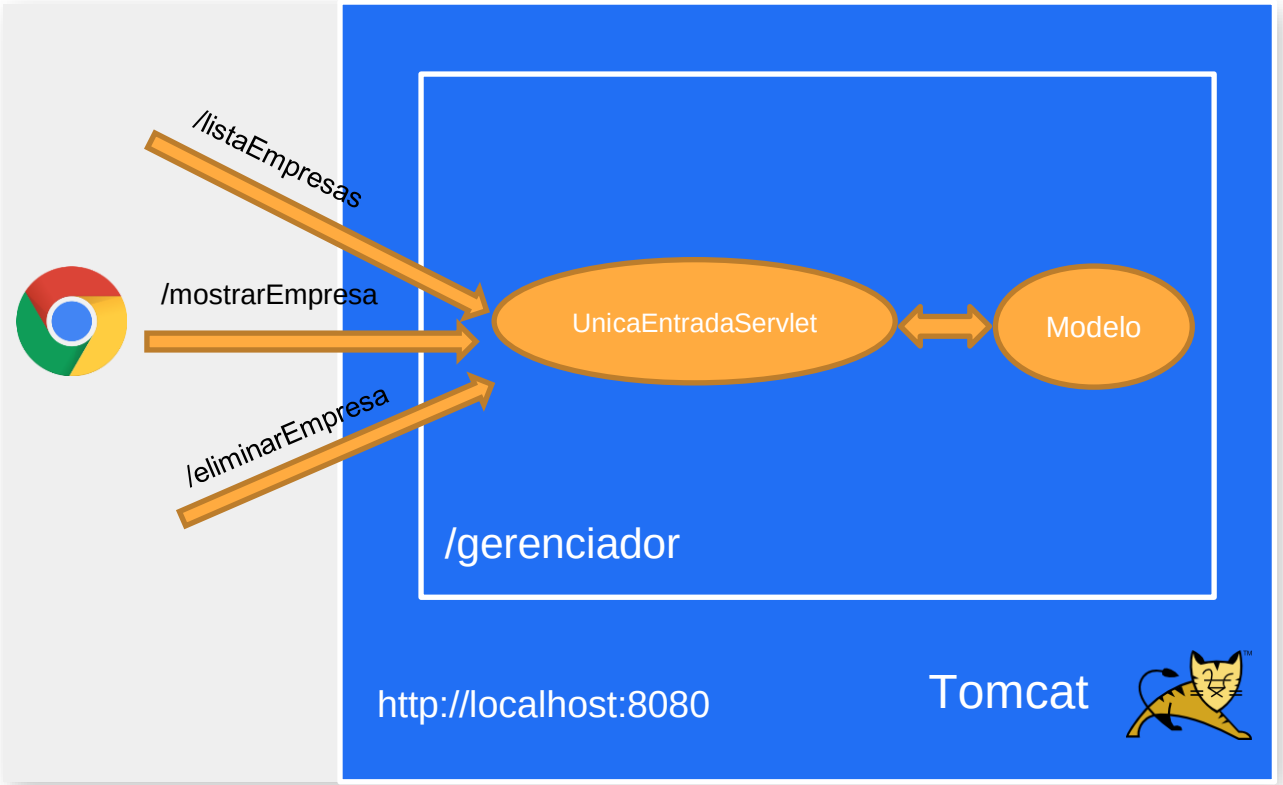

Desarrollo

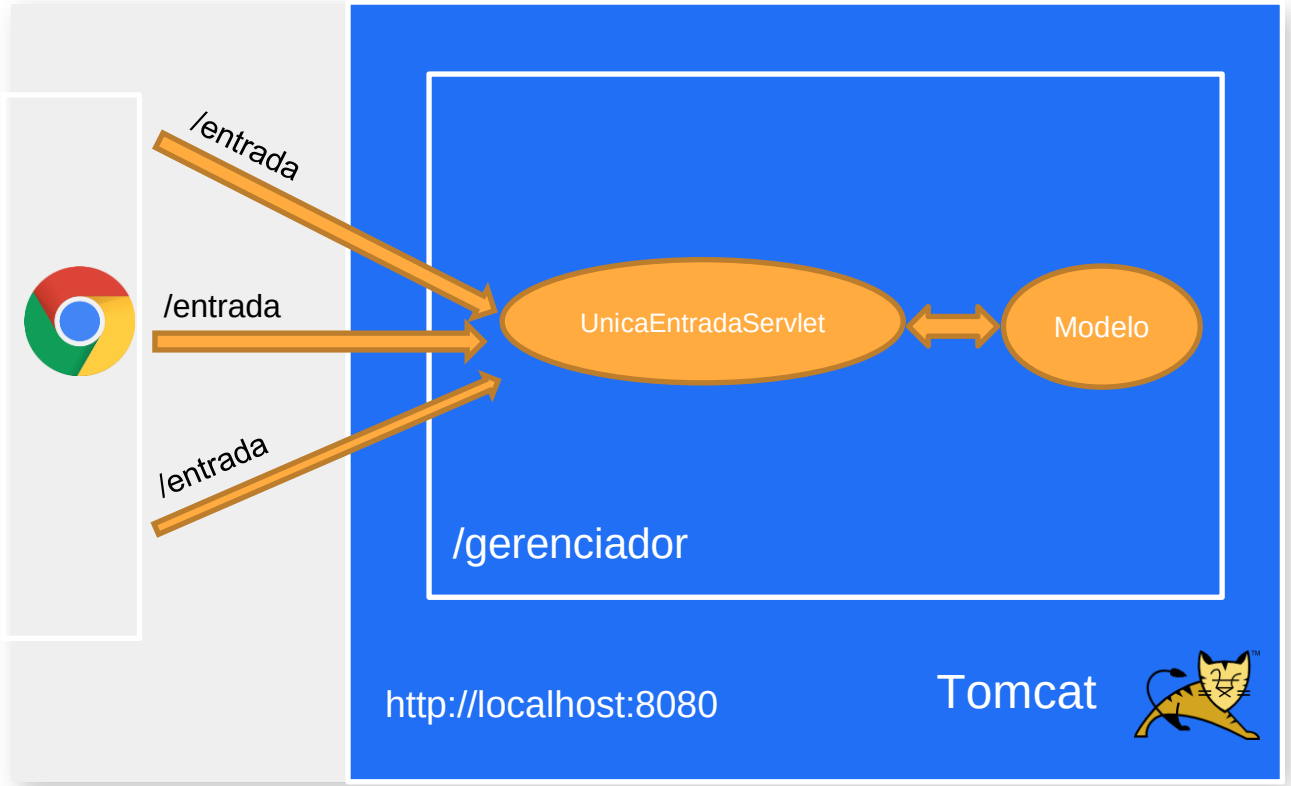


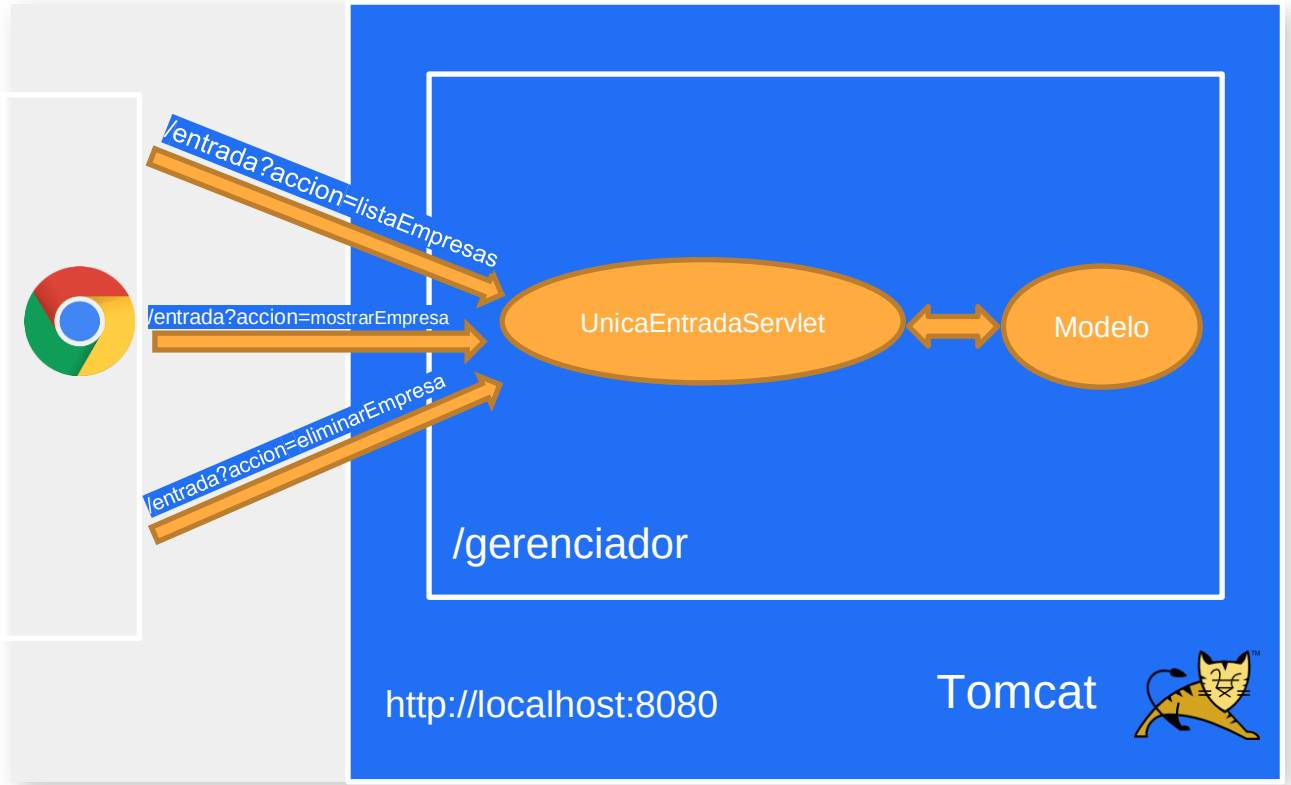

Producción

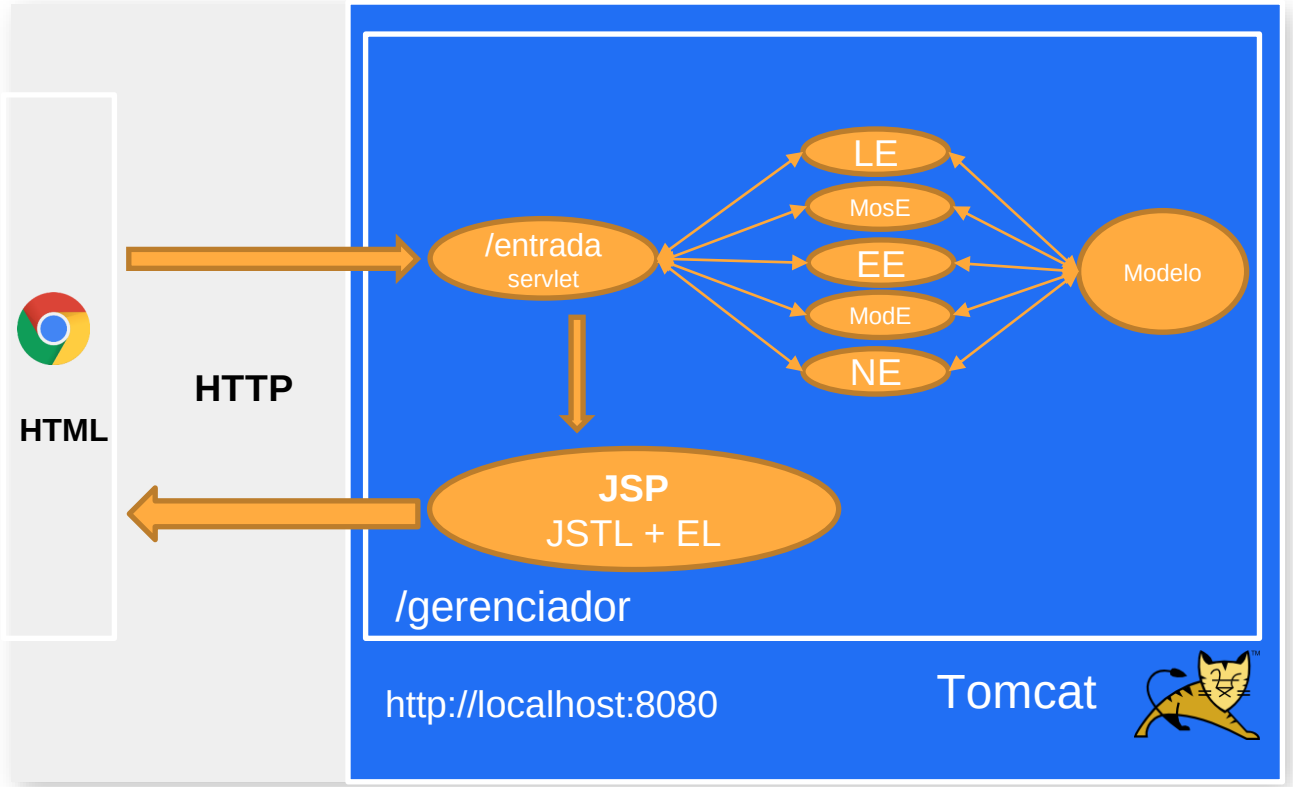

request
HTTP
POST/GET
response
Servlet
Modelo
Dispatcher
JSP
JSTL + EL
/gerenciador
http://localhost:8080
Tomcat

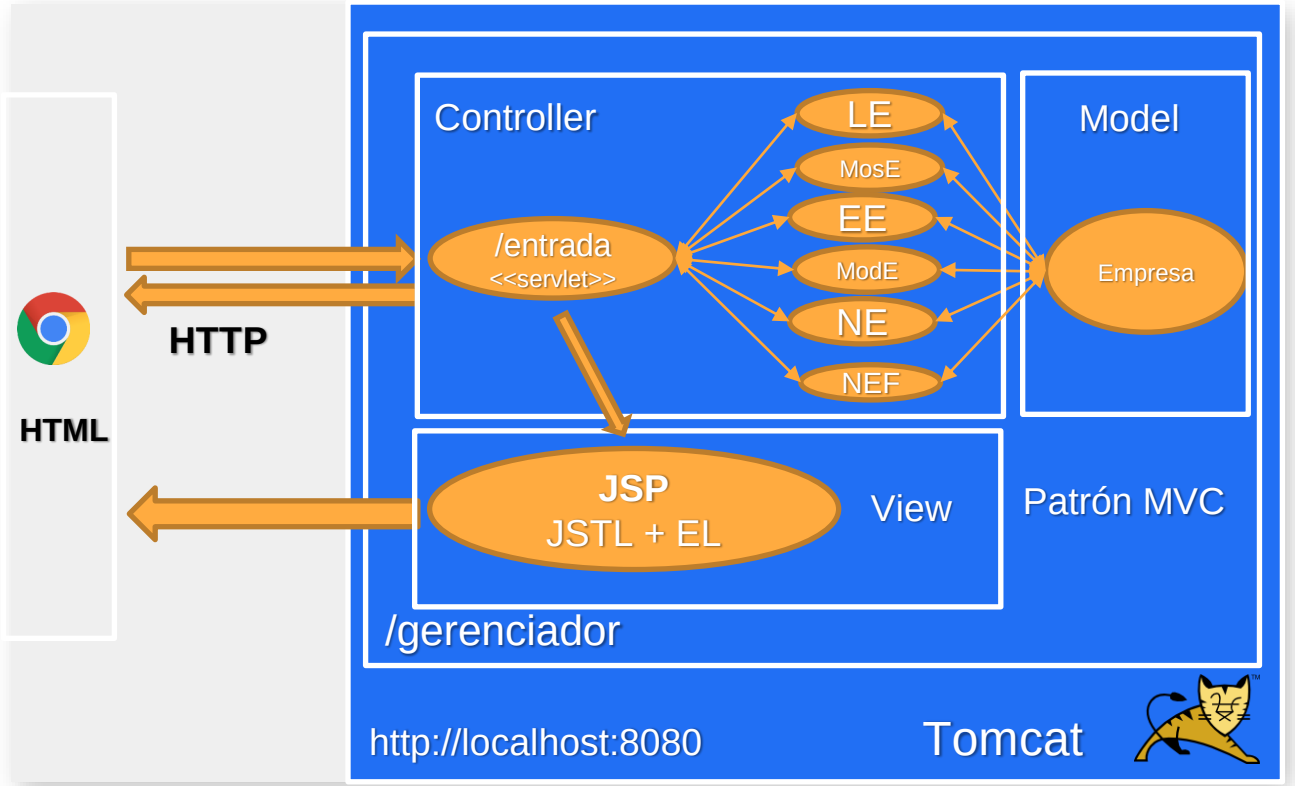

MostrarEmpresa
Servlet
JSP
Modelo
ModificarEmpresa
Servlet
ListaEmpresa
Servlet
Dispatcher
JSP
JSTL + EL
/gerenciador
http://localhost:8080
Tomcat

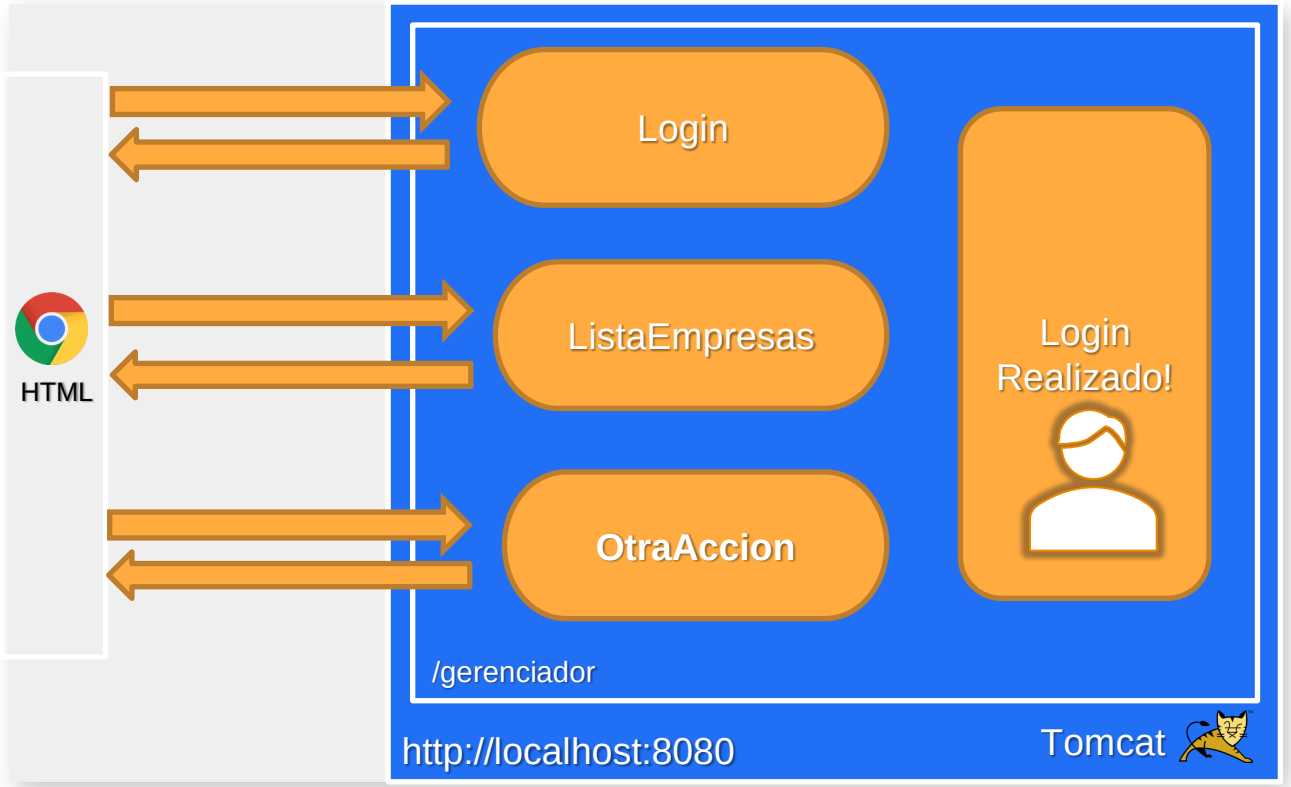

/listaEmpresas
/mostrarEmpresa
/eliminarEmpresa
UnicaEntradaServlet
Modelo
/gerenciador
http://localhost:8080
Tomcat

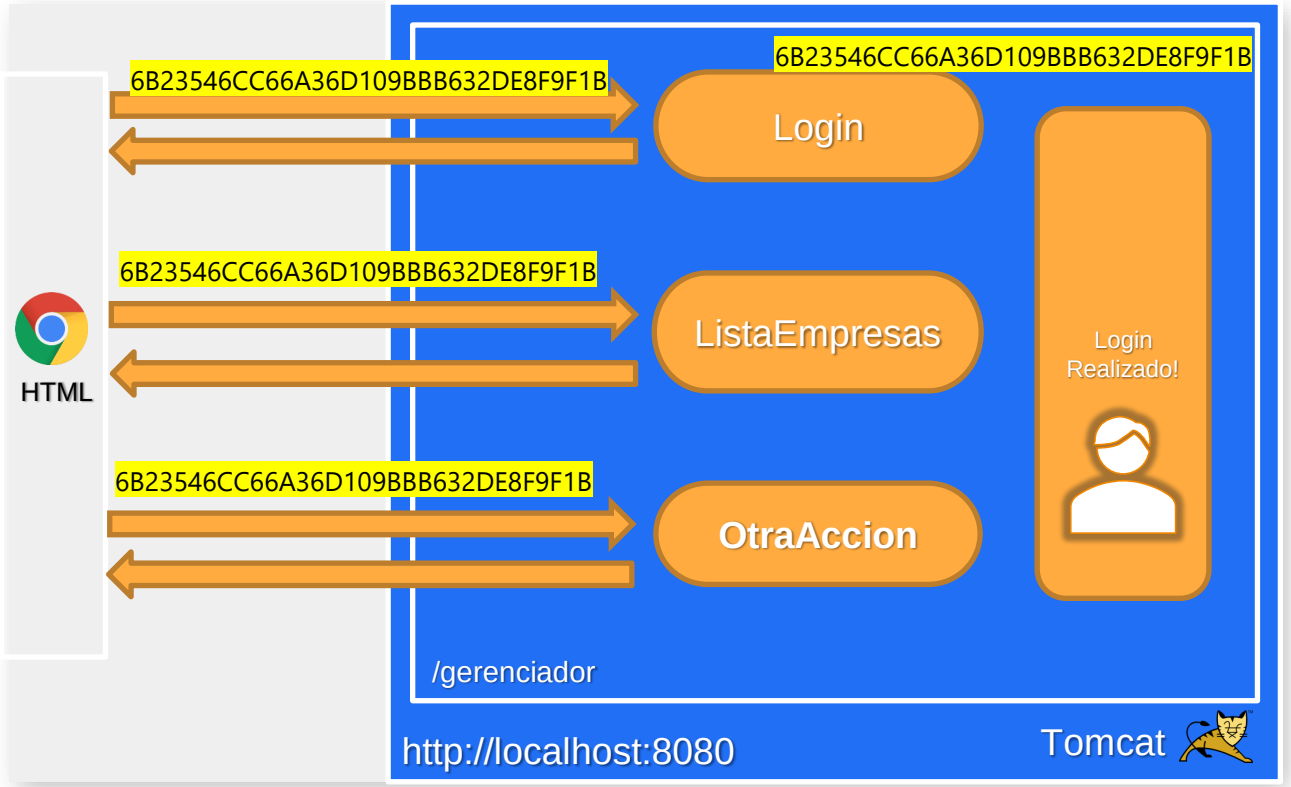

/entrada
/entrada
/entrada
UnicaEntradaServlet
Modelo
/gerenciador
http://localhost:8080
Tomcat

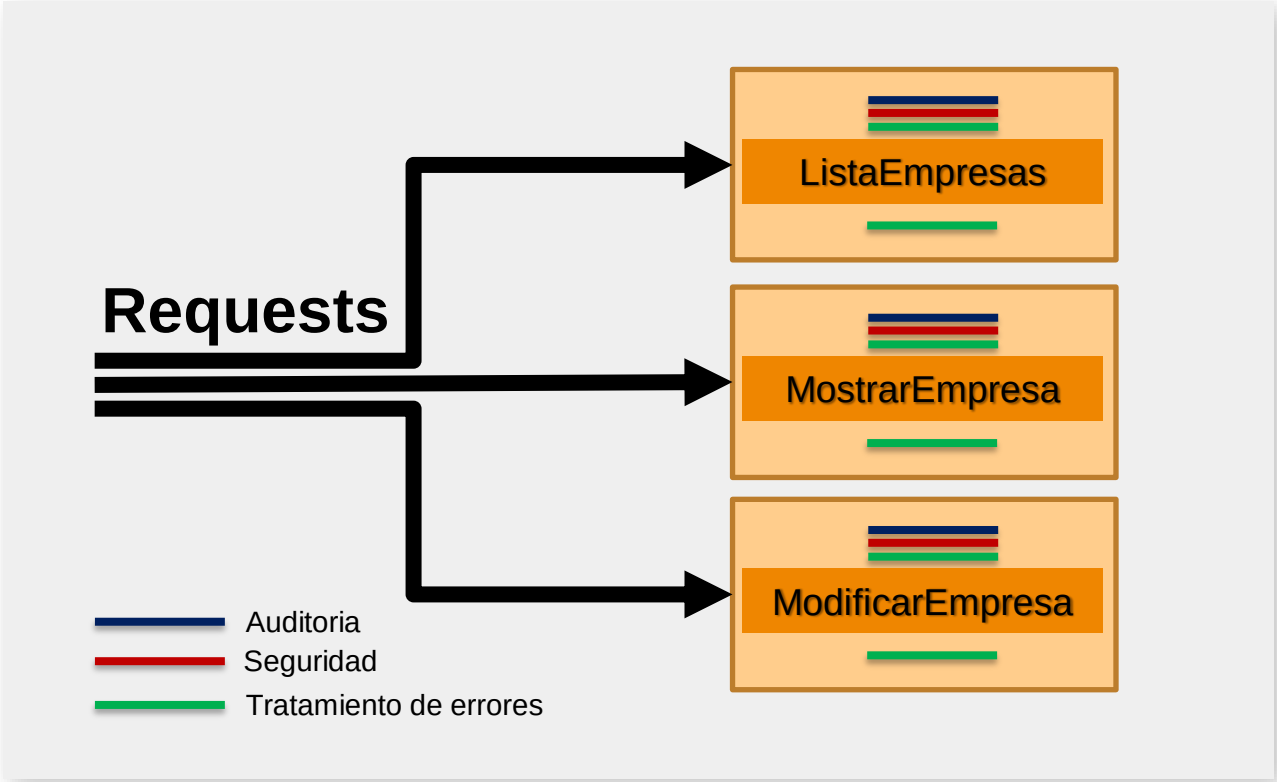

/entrada?accion=listaEmpresas
/entrada?accion=mostrarEmpresa
/entrada?accion=eliminarEmpresa
UnicaEntradaServlet
Modelo
/gerenciador
http://localhost:8080
Tomcat

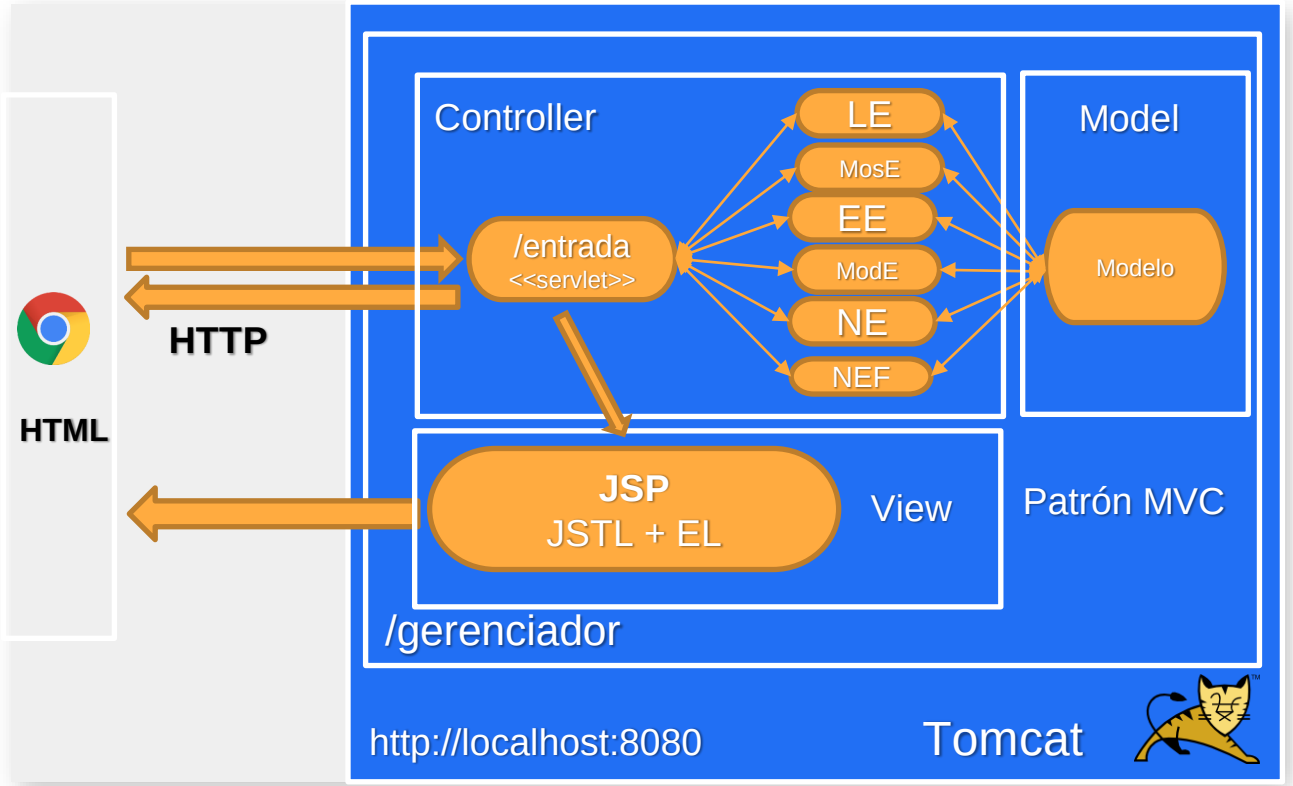

HTML
HTTP
/entrada
servlet
LE
MosE
EE
ModE
NE
Modelo
JSP
JSTL + EL
/gerenciador
http://localhost:8080
Tomcat

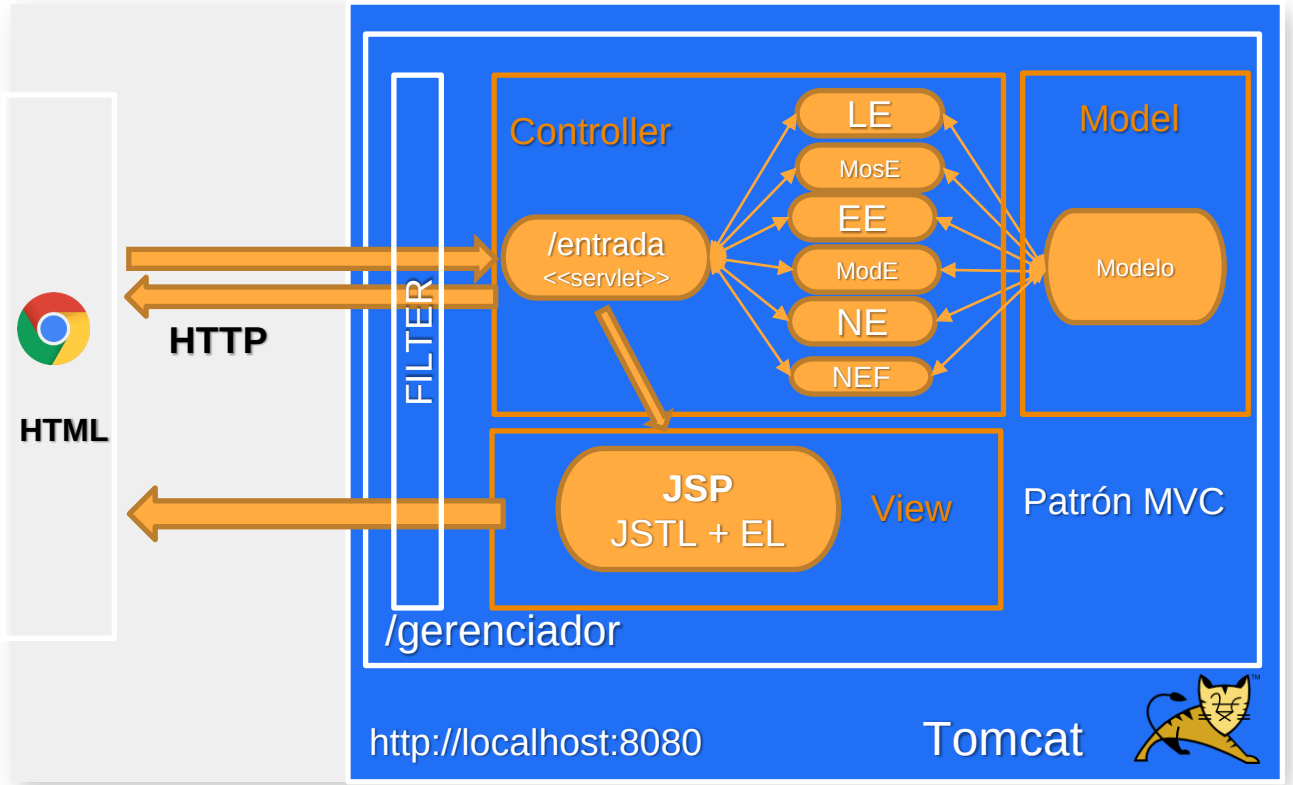

Controller
LE
MosE
EE
ModE
NE
NEF
Model
/entrada
<<servlet>>
Empresa
HTTP
HTML
JSP
JSTL + EL
View
Patrón MVC
/gerenciador
http://localhost:8080
Tomcat

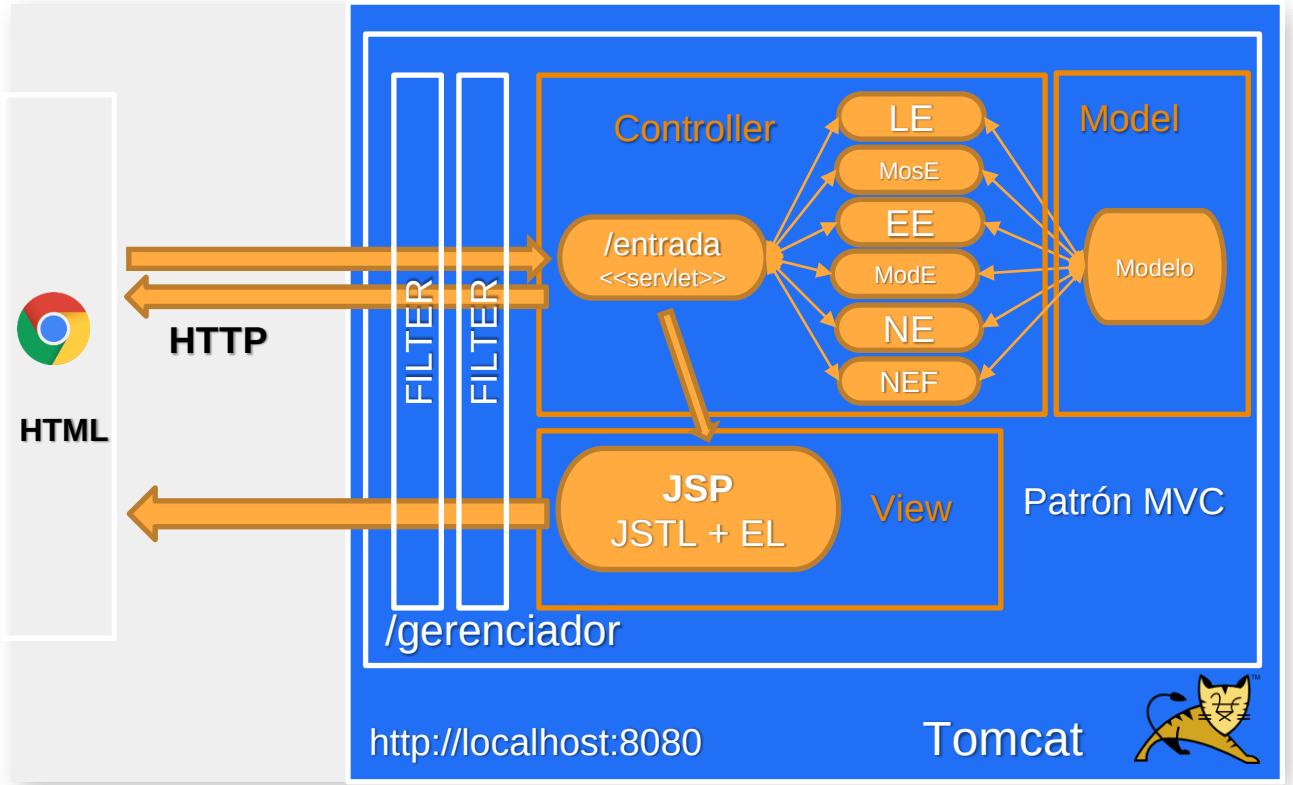

Login
ListaEmpresas
OtraAccion
Login
Realizado!
HTML
/gerenciador
http://localhost:8080
Tomcat

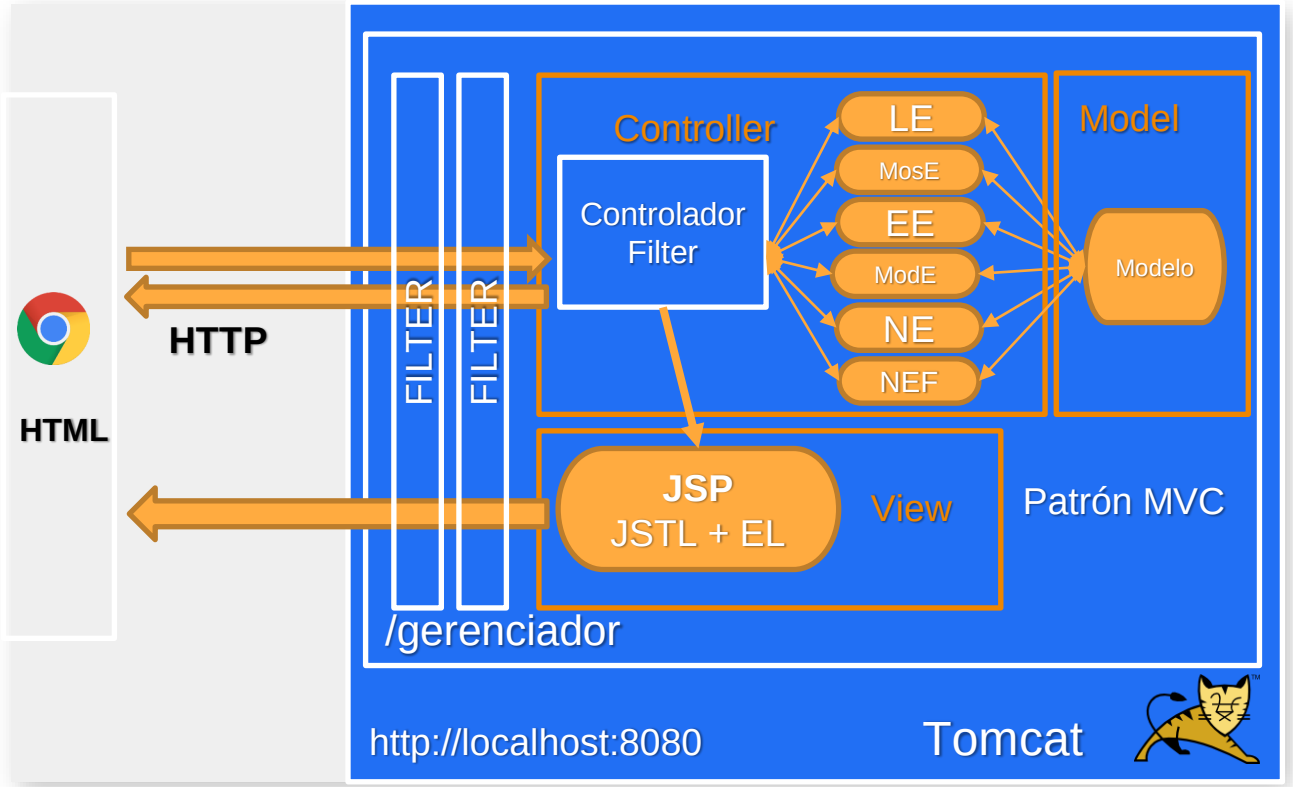

6B23546CC66A36D109BBB632DE8F9F1B
6B23546CC66A36D109BBB632DE8F9F1B
Login
6B23546CC66A36D109BBB632DE8F9F1B
ListaEmpresas
HTML
Login
Realizado!
6B23546CC66A36D109BBB632DE8F9F1B
OtraAccion
/gerenciador
http://localhost:8080
Tomcat


Requests
ListaEmpresas
MostrarEmpresa
ModificarEmpresa
Auditoria
Seguridad
Tratamiento de errores


Controller
LE
MosE
EE
ModE
NE
NEF
Model
Modelo
/entrada
<<servlet>>
HTTP
HTML
JSP
JSTL + EL
View
Patrón MVC
/gerenciador
http://localhost:8080
Tomcat


HTML
HTTP
FILTER
Controller
/entrada
<<servlet>>
LE
MosE
EE
ModE
NE
NEF
Model
Modelo
JSP
JSTL + EL
View
Patrón MVC
/gerenciador
http://localhost:8080
Tomcat

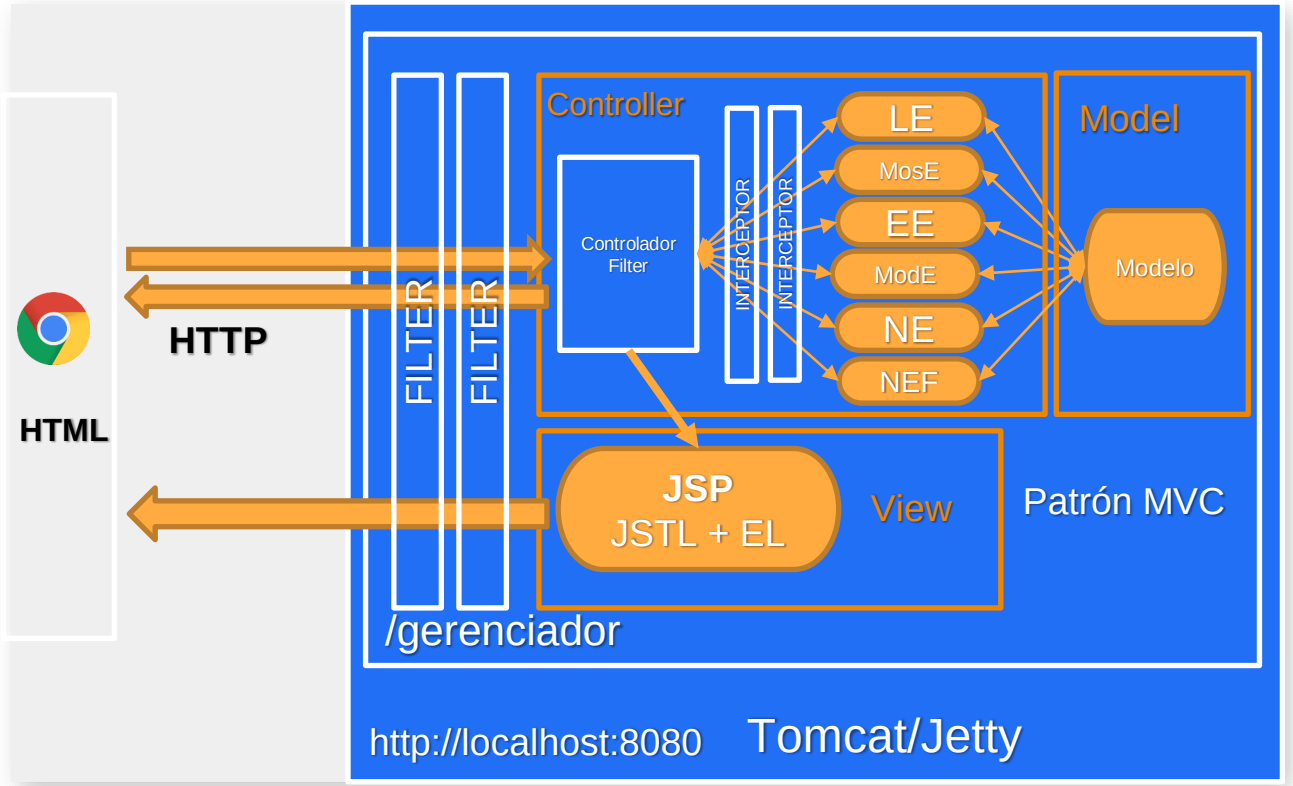

HTML
HTTP
FILTER
FILTER
Controller
/entrada
<<servlet>>
LE
MosE
EE
ModE
NE
NEF
Model
Modelo
JSP
JSTL + EL
View
Patrón MVC
/gerenciador
http://localhost:8080
Tomcat

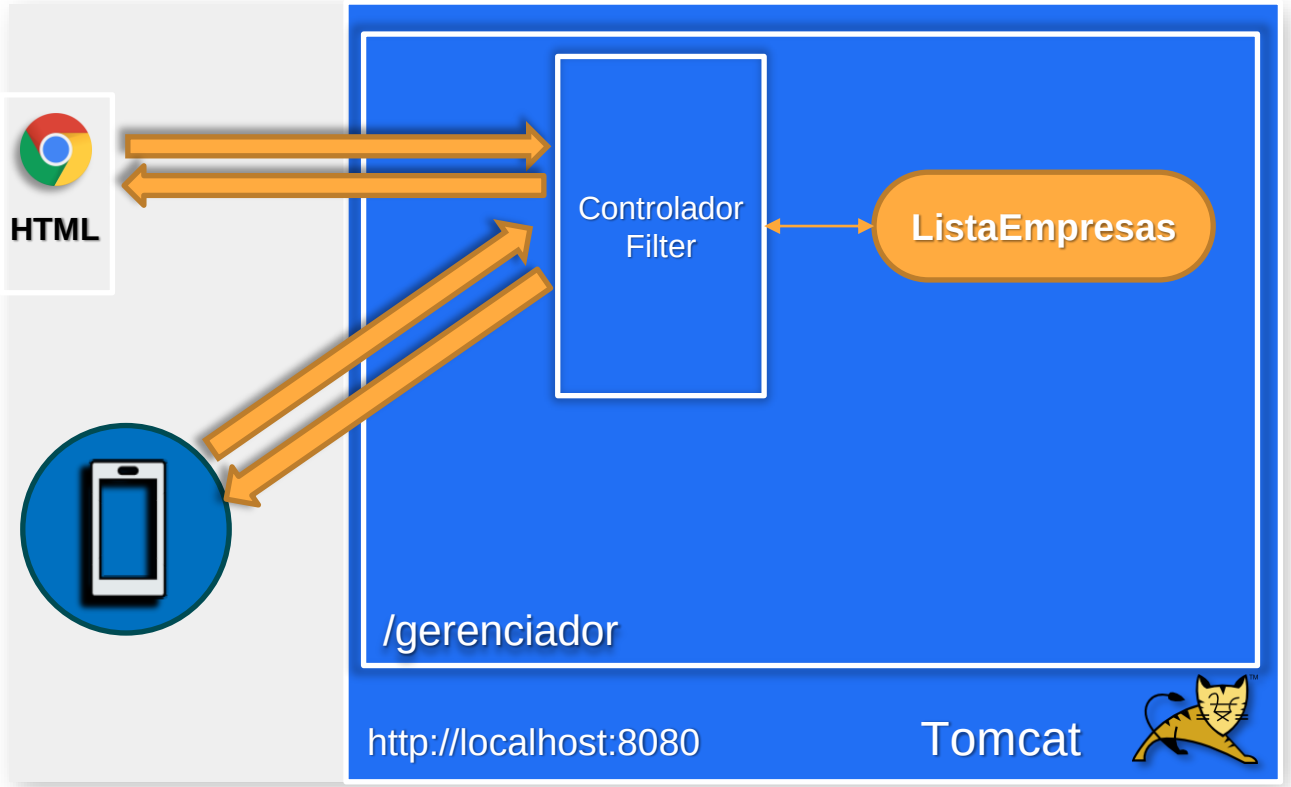



En otros Frameworks:

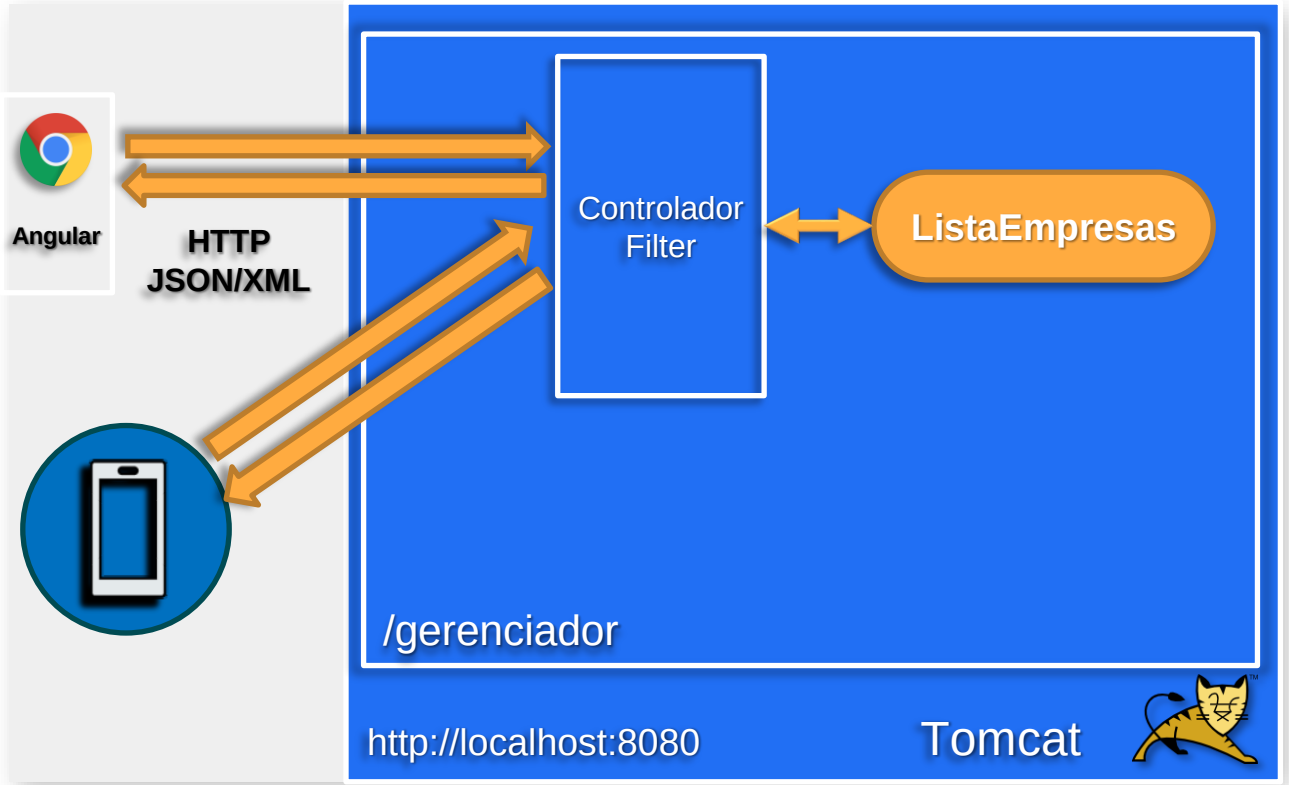
- **Controlador** más sofisticado
- **Modelo** con bibliotecas de más alto nivel
- **JSP:** Jquery, Javascript, otras tags más específicas.
- **Actions:** SpringMVC deja agrupar en una clase varias acciones, sin Interface.
- **Filtros:** Spring ya trae algunos listos.

Lo importante es que comprendan los conceptos y flujos de datos. La estructura se usa en otros lenguajes también.

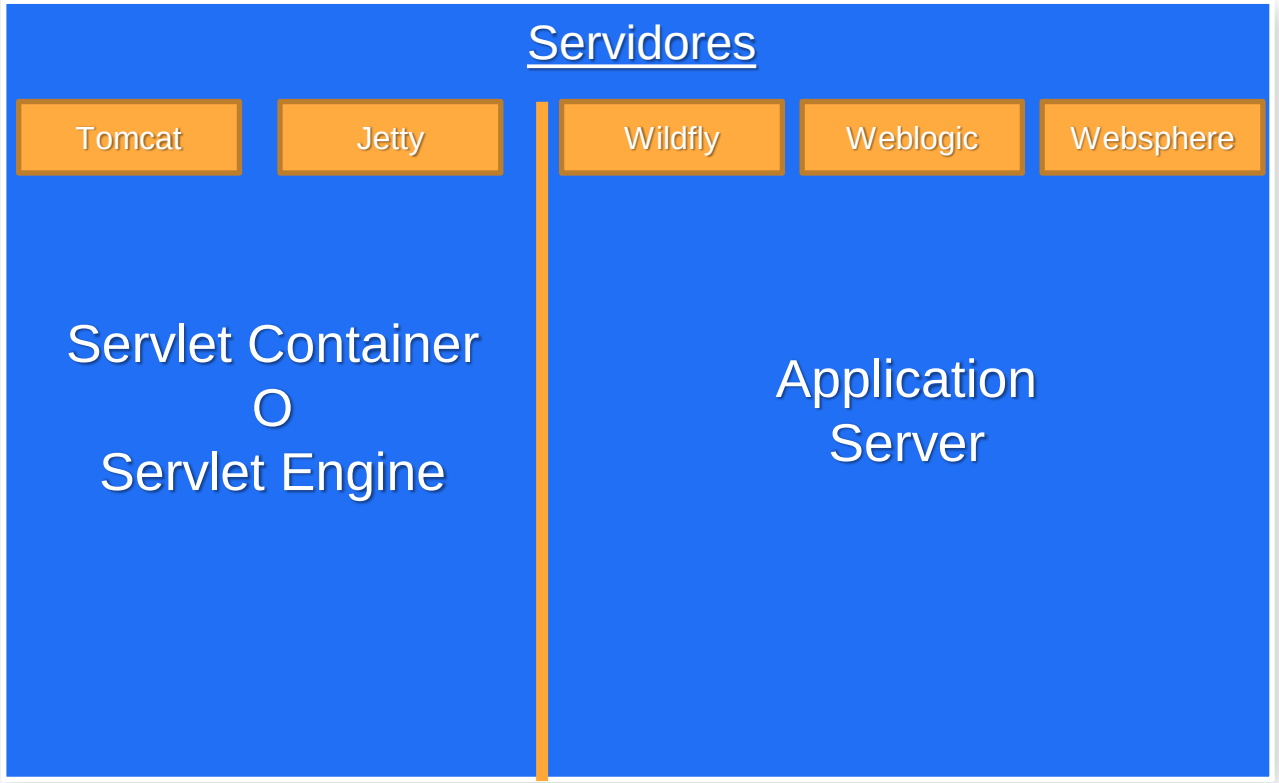

HTML
HTTP
FILTER
FILTER
Controller
Controlador
Filter
INTERCEPTOR
INTERCEPTOR
LE
MosE
EE
ModE
NE
NEF
Model
Modelo
JSP
JSTL + EL
View
Patrón MVC
/gerenciador
http://localhost:8080
Tomcat/Jetty

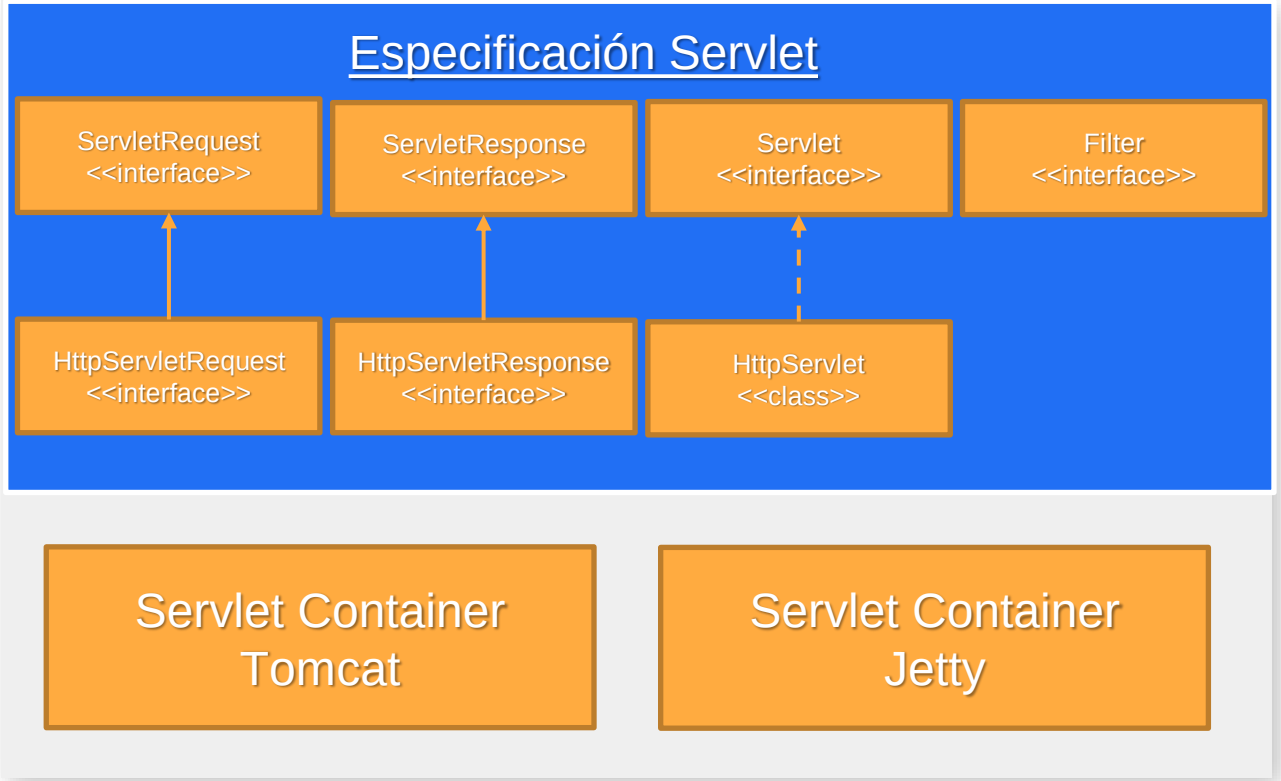



- ¿Y si nuestro cliente **no es un navegador**?
- Una **app** por ejemplo no necesariamente usa un navegador para comunicarse.
- Vamos a mostrar la **lista de empresas sin HTML**. A veces solo queremos devolver datos.




- 2 formatos: **JSON y XML**
- Son Strings muy grandes, con datos.
- **Angular, React, Vue.js** hacen un **MVC dentro del navegador**.
- Crearemos nuestro primer **Web Service**.
- Los Web Service usan el **protocolo HTTP**, llaman una funcionalidad remotamente, y devuelven los datos en formato **JSON/XML**.
- Recomiendo ver el curso de **HTTP** de Alura.


Servidores
Tomcat
Jetty
Wildfly
Weblogic
Websphere
Servlet Container
O
Servlet Engine
Application
Server


Especificación Servlet
ServletRequest
<<interface>>
ServletResponse
<<interface>>
Servlet
<<interface>>
Filter
<<interface>>
HttpServletRequest
<<interface>>
HttpServletResponse
<<interface>>
HttpServlet
<<class>>
Servlet Container
Tomcat
Servlet Container
Jetty