
SISTEMAS WEB

CURSO 2023/2024

Tomcat

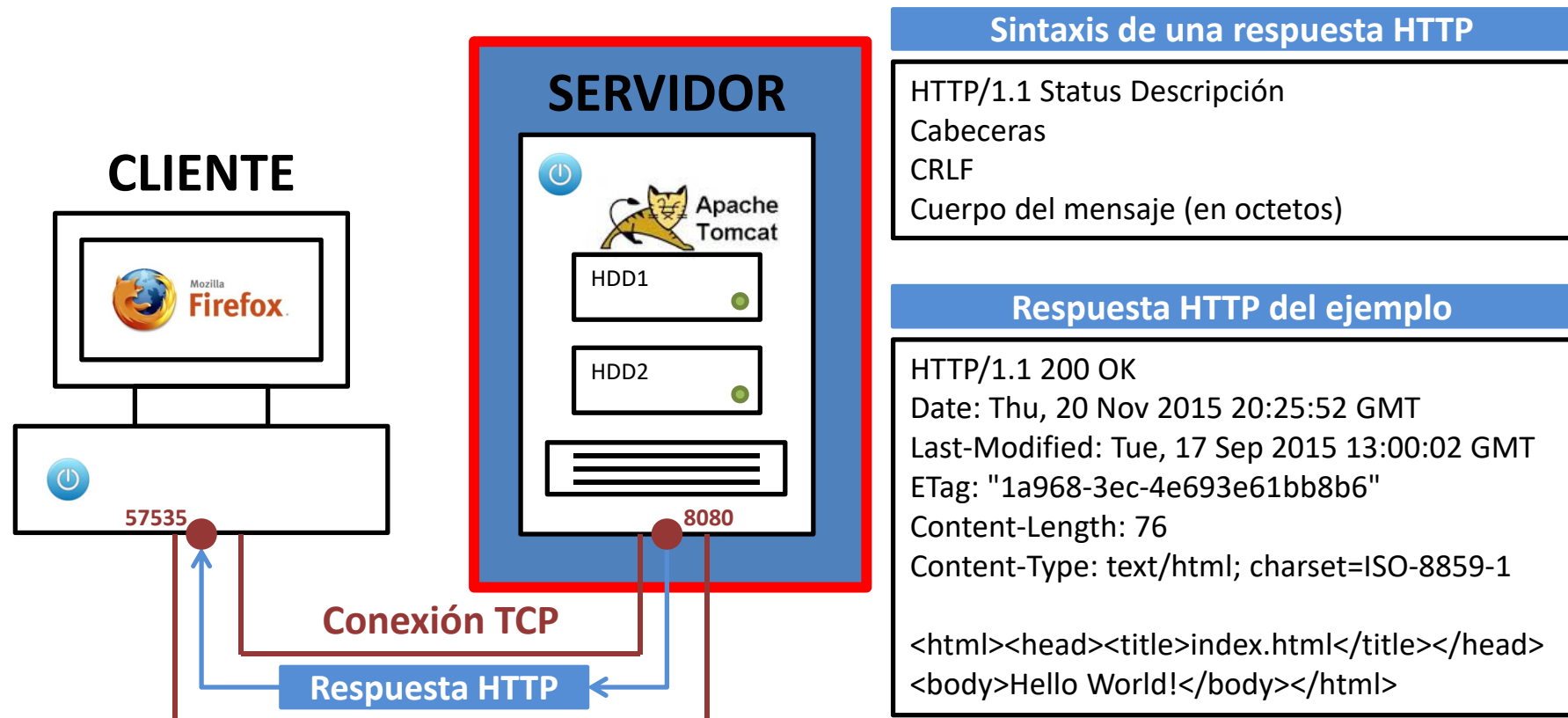
Introducción y Ejemplos



Web Sistemak by [Oskar Casquero](#) & [María Luz Álvarez](#) is licensed under a [Creative Commons Reconocimiento 4.0 Internacional License](#).

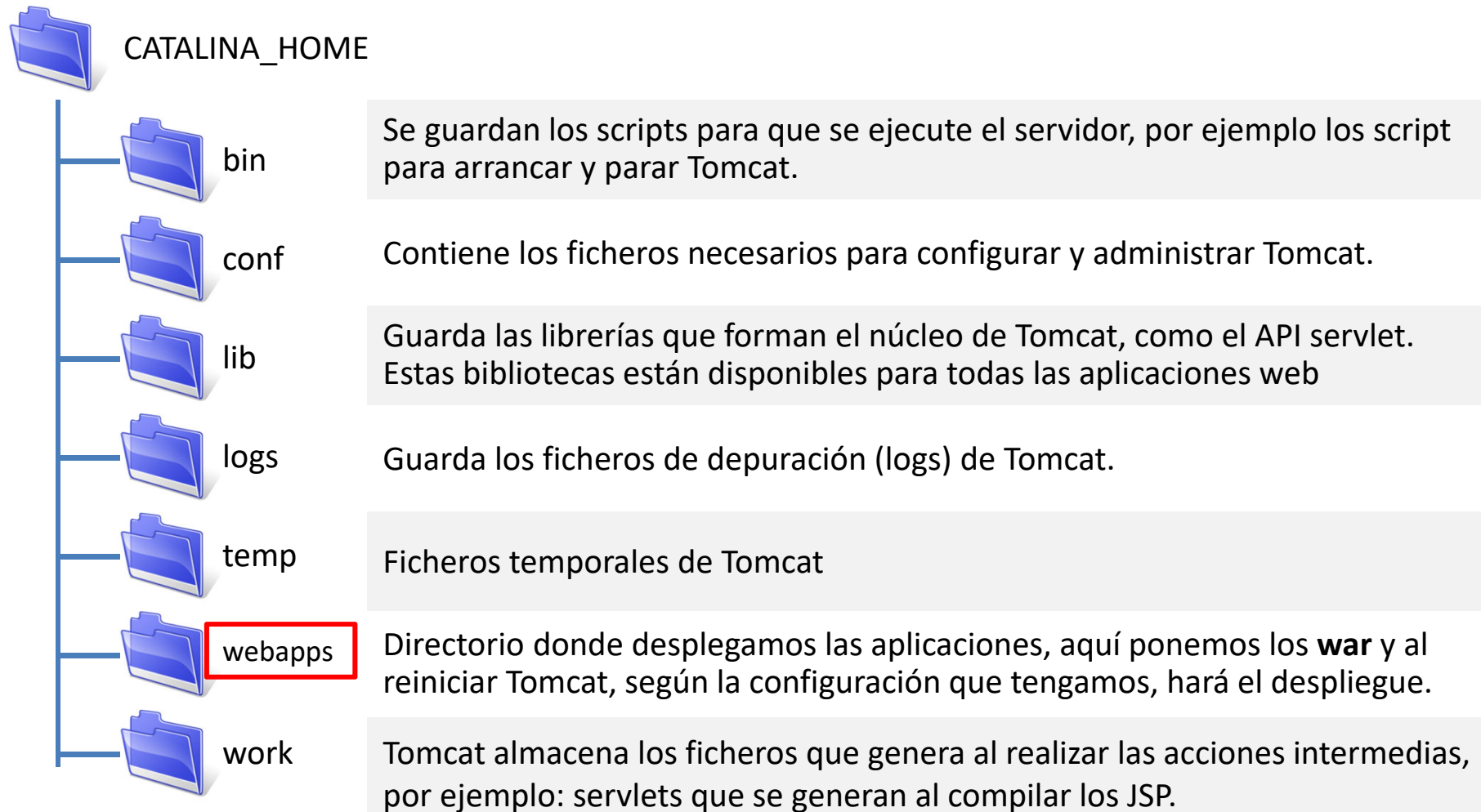
SERVIDOR WEB:

PROCESA PETICIONES HTTP Y GENERA RESPUESTAS HTTP



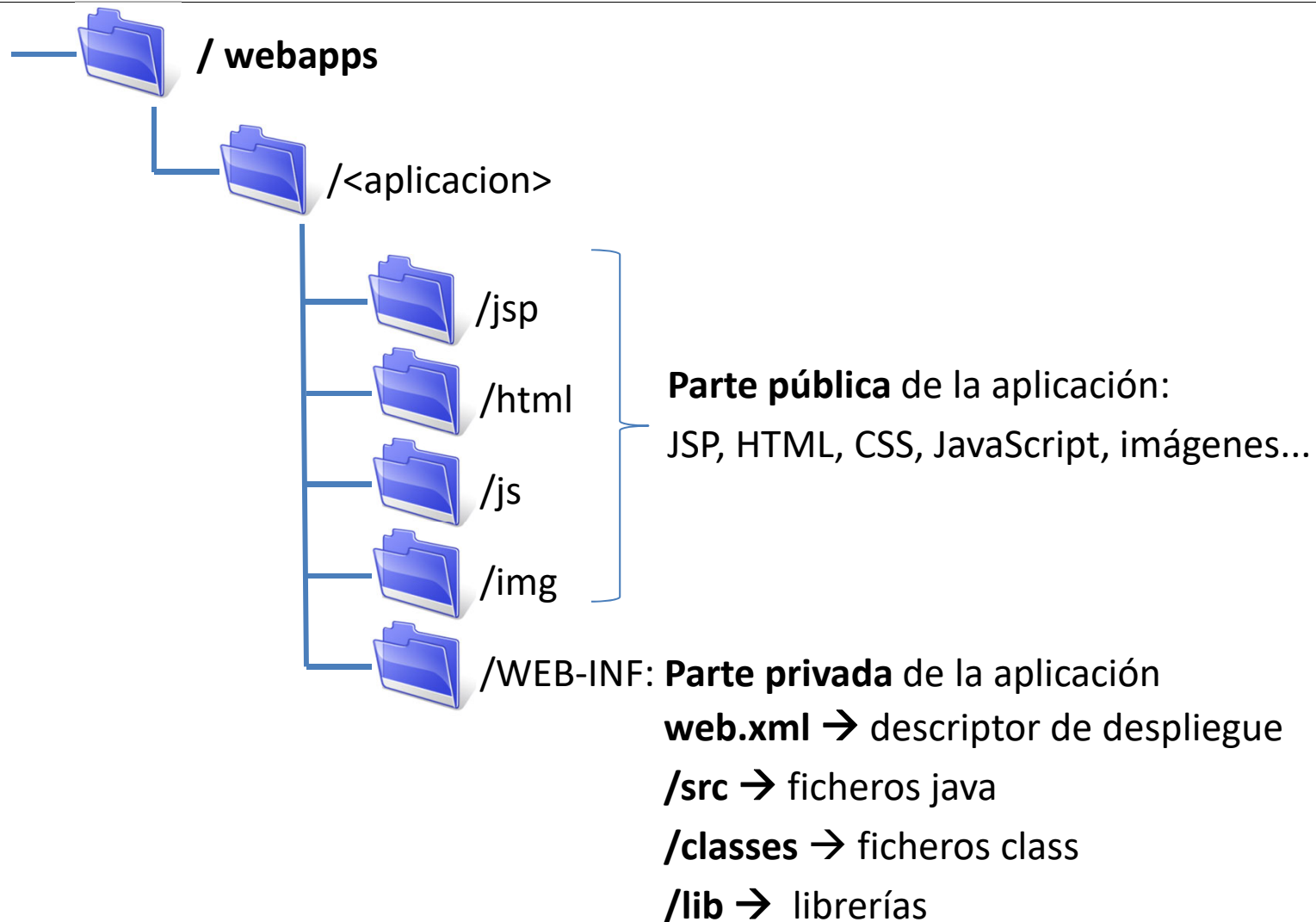
SERVIDOR WEB: TOMCAT

ESTRUCTURA DE DIRECTORIOS



SERVIDOR WEB: APLICACIÓN WEB

ESTRUCTURA DE DIRECTORIOS APLICACIÓN



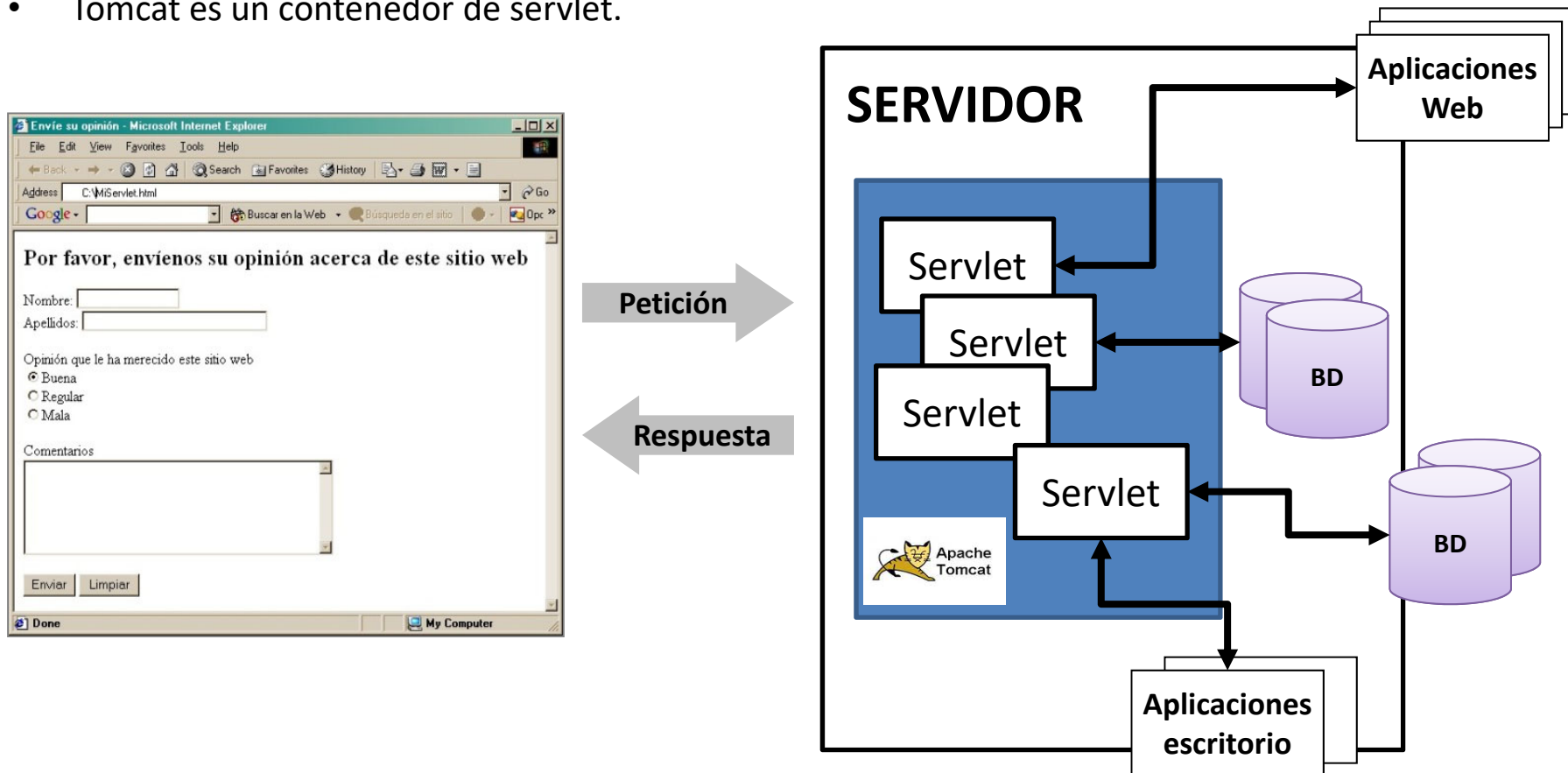
TOMCAT

DESPLIEGUE DE UNA APLICACIÓN WEB

- **Directamente** : Se puede crear la estructura de directorios y meter los ficheros en el directorio que corresponda.
- **Archivos WAR**
 - Un archivo **WAR** no es más que el **directorio de la aplicación** comprimido en un archivo en formato **zip**.
 - Eclipse generan los **war** automáticamente en :
 - Project → Export → War file

SERVLETS

- Un **servlet** es un programa que procesa y responde peticiones de un cliente web.
- Una aplicación web puede estar compuesta por varios servlets.
- Tomcat es un contenedor de servlet.



SERVLETS

- Los **Servlets** pueden responder a cualquier tipo de petición, aunque comúnmente se usan en servidores Web que reciben, procesan y responden a peticiones **HTTP**.
- Para esta función la tecnología **Java Servlet** utiliza una API definida en las interfaces **javax.servlet** y **javax.servlet.http**
Tomcat es una implementación de esta API

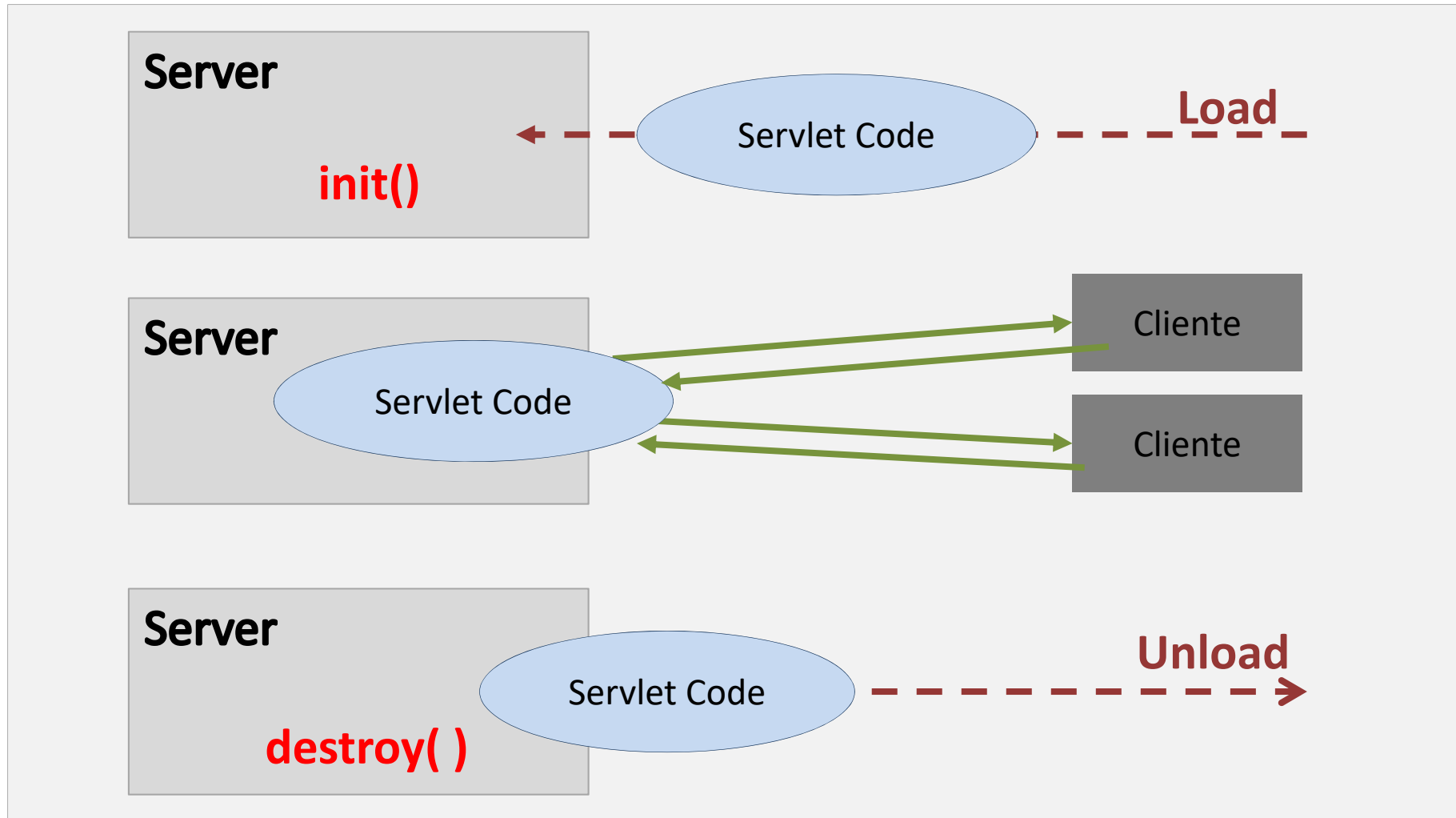
- *javax.servlet*

- <http://docs.oracle.com/javaee/5/api/javax/servlet/package-summary.html>

- *javax.servlet.http*

- <http://docs.oracle.com/javaee/5/api/javax/servlet/http/package-summary.html>

SERVLETS - CICLO DE VIDA

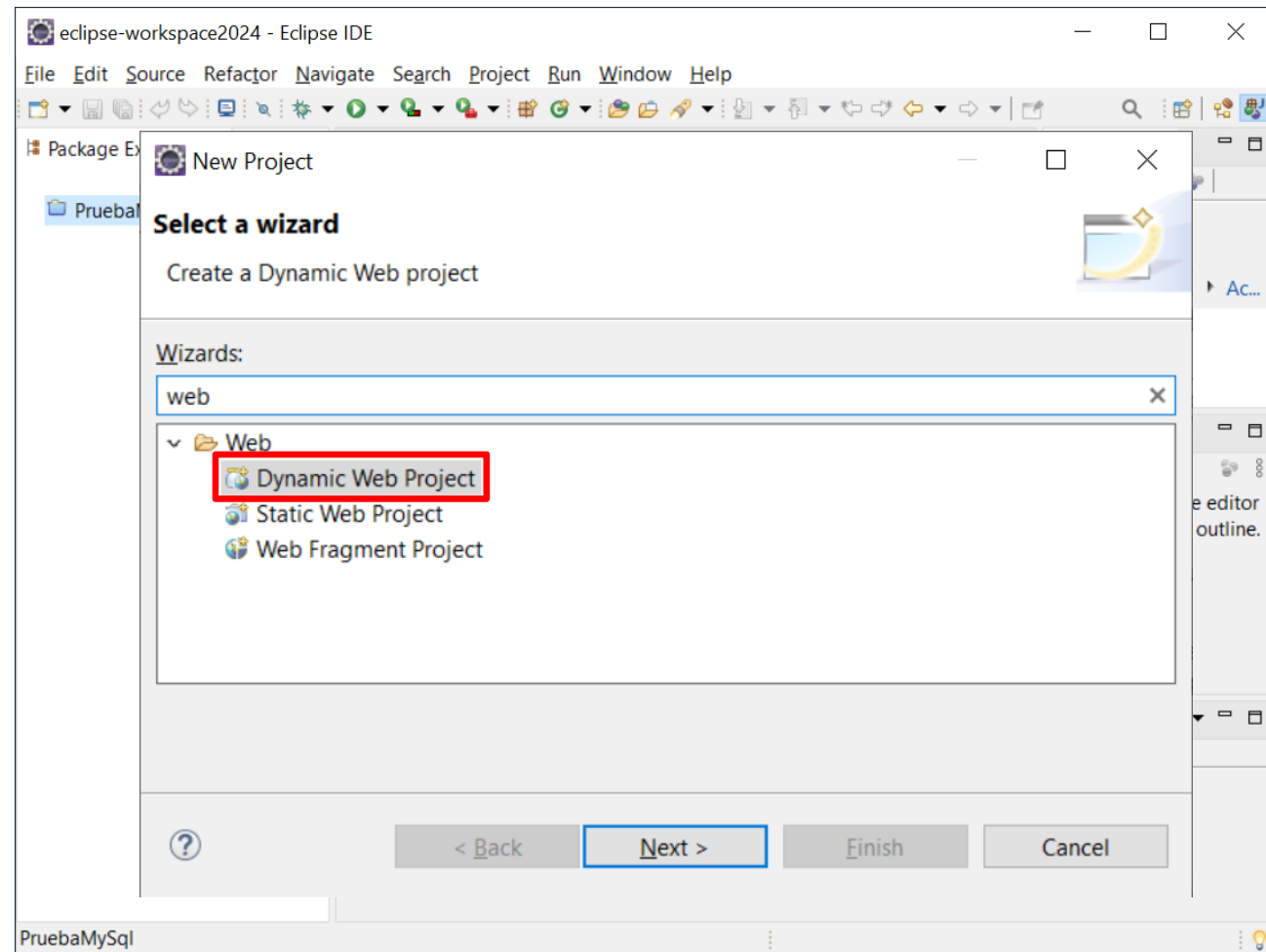


ECLIPSE - DESARROLLO DE UNA APLICACIÓN WEB

CREAR PROYECTO WEB

IntelliJ → Transparencia 29 al final

- File
 - New
 - Project



ECLIPSE - DESARROLLO DE UNA APLICACIÓN WEB

ENLAZAR CON TOMCAT

New Dynamic Web Project

Dynamic Web Project

Create a standalone Dynamic Web project or add it to a new or existing Enterprise Application.

Project name: **prueba**

Project location

☒ Use default location

Location: C:\Users\cvzcaio\workspace\proba

Target runtime

<None> New Runtime...

Dynamic web module version

3.0

Configuration

Default Configuration Modify...

The default configuration provides a good starting point. Additional facets can later be installed to add new functionality to the project.

EAR membership

☐ Add project to an EAR

EAR project name: EAR New Project...

Working sets

☐ Add project to working sets New...

Working sets: Select...

< Back Next > Finish Cancel

New Server Runtime Environment

Define a new server runtime environment

Select the type of runtime environment:

type filter text

- Apache
 - Apache Tomcat v7.0
 - Apache Tomcat v8.0
 - Apache Tomcat v8.5**
 - Apache Tomcat v9.0
 - Apache Tomcat v10.0

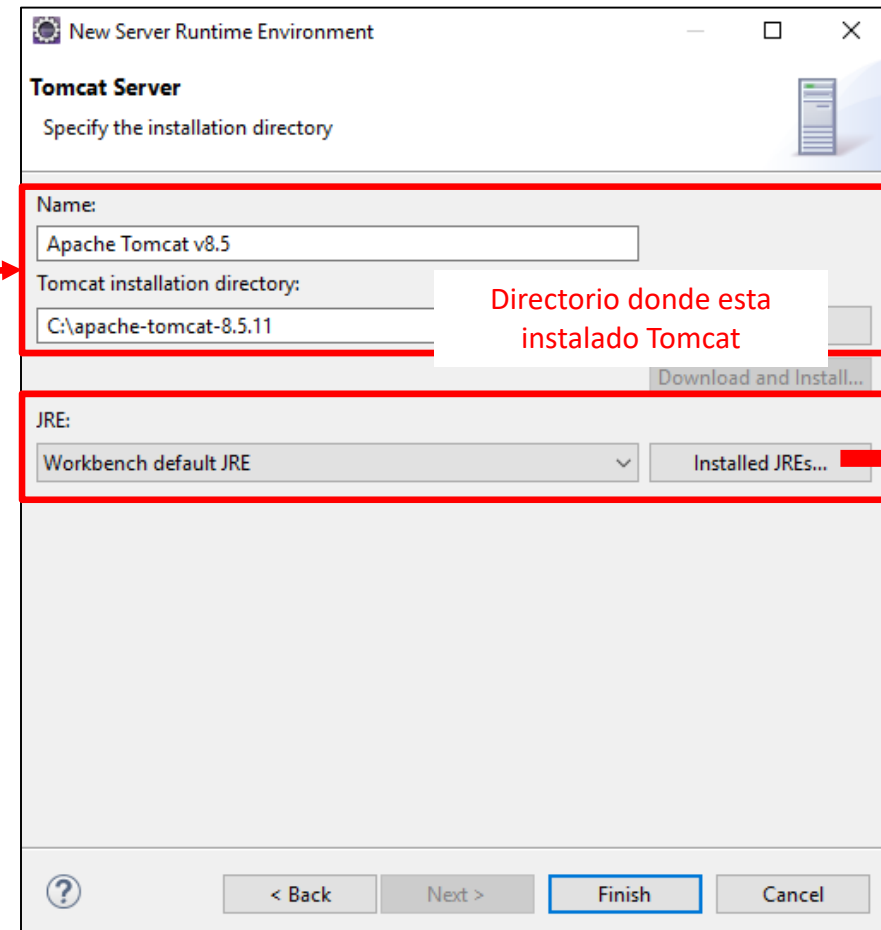
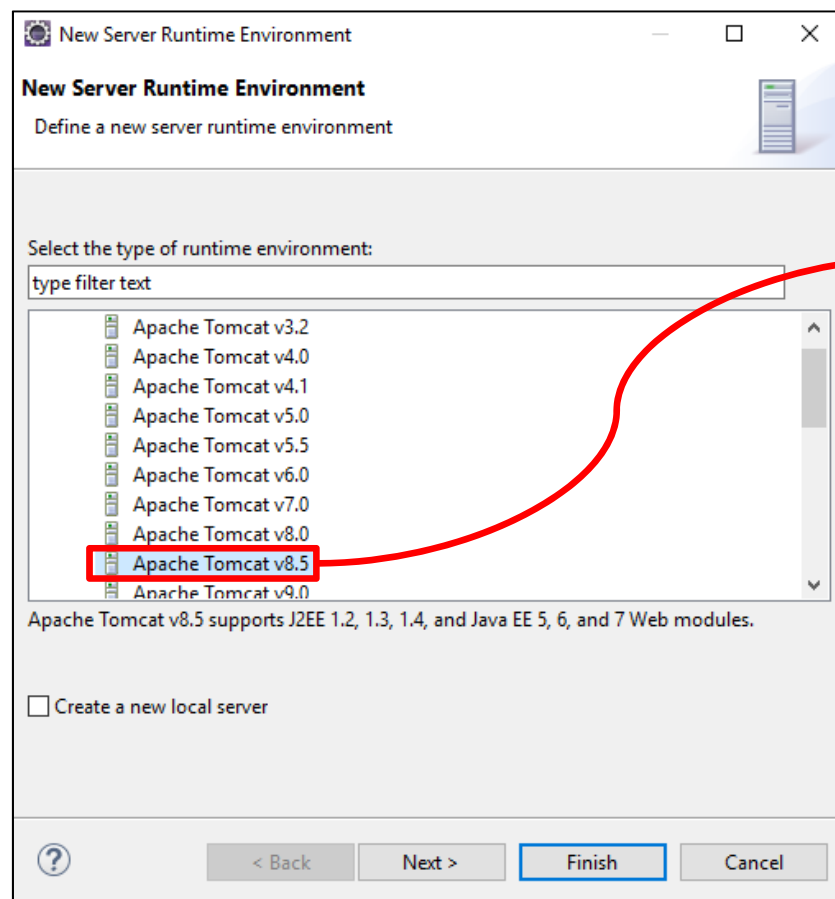
Apache Tomcat v8.5 supports J2EE 1.2, 1.3, 1.4, and Java EE 5, 6, and 7 Web modules.

☐ Create a new local server

? < Back **Next >** Finish Cancel

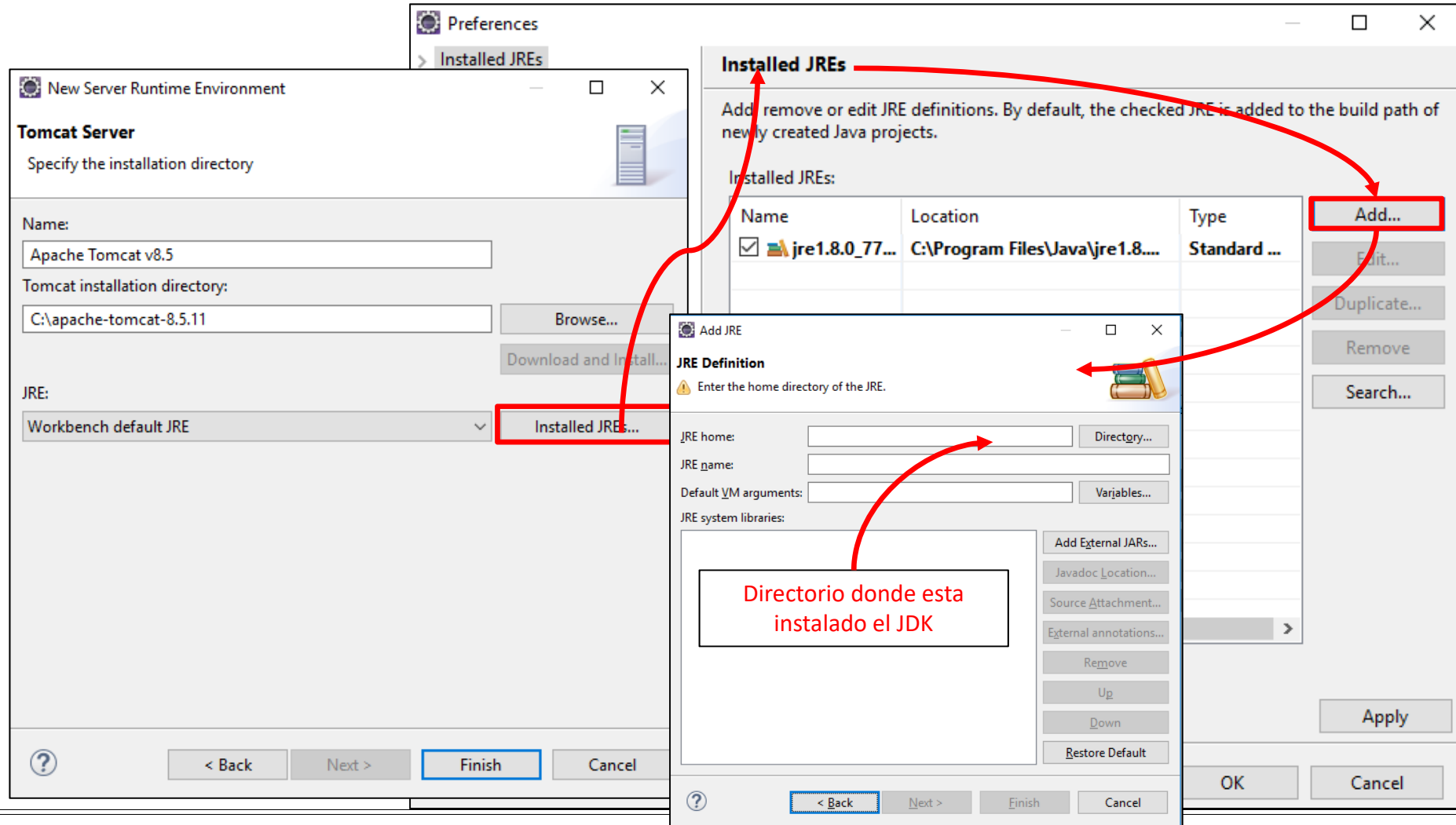
ECLIPSE - DESARROLLO DE UNA APLICACIÓN WEB

ENLAZAR CON TOMCAT



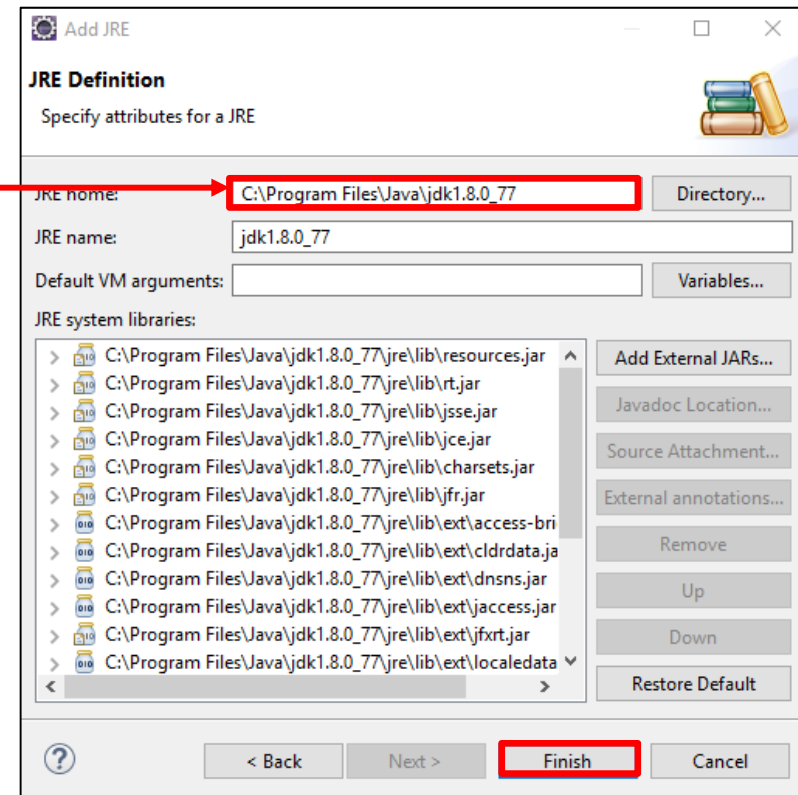
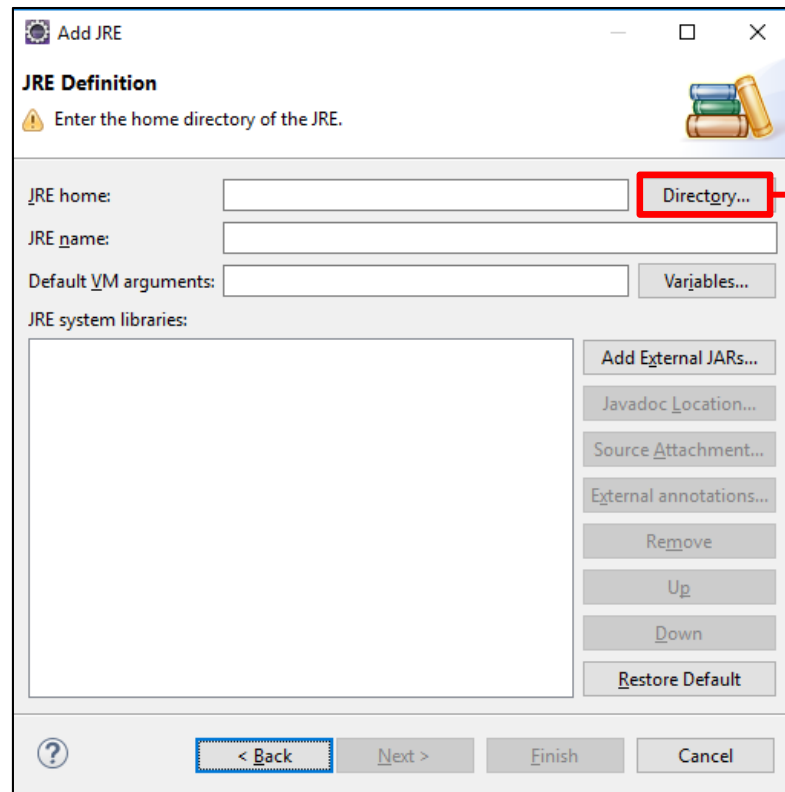
ECLIPSE - DESARROLLO DE UNA APLICACIÓN WEB

ENLAZAR CON TOMCAT



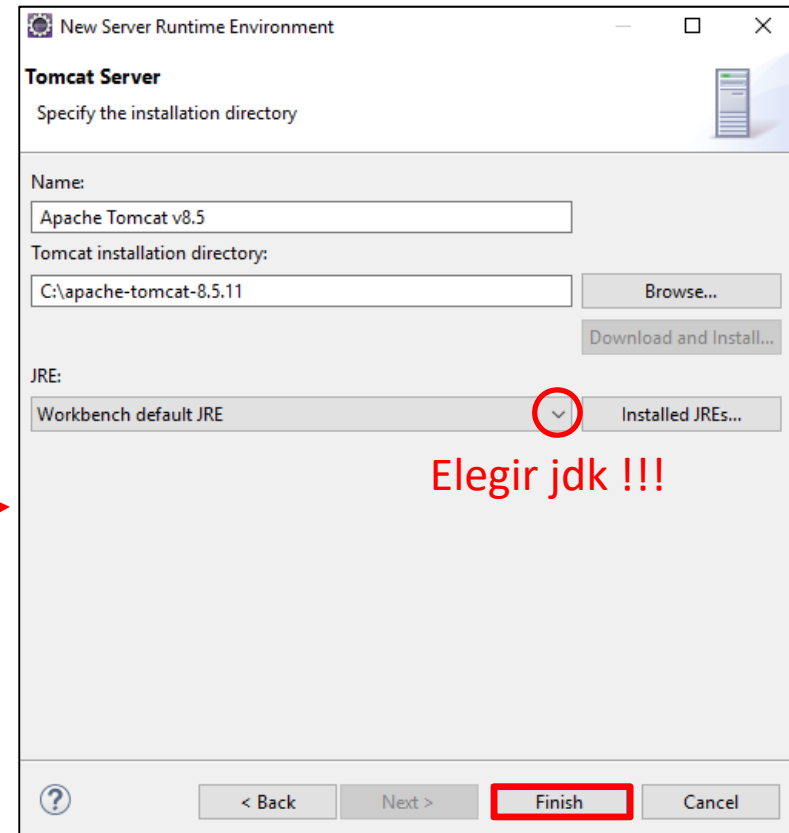
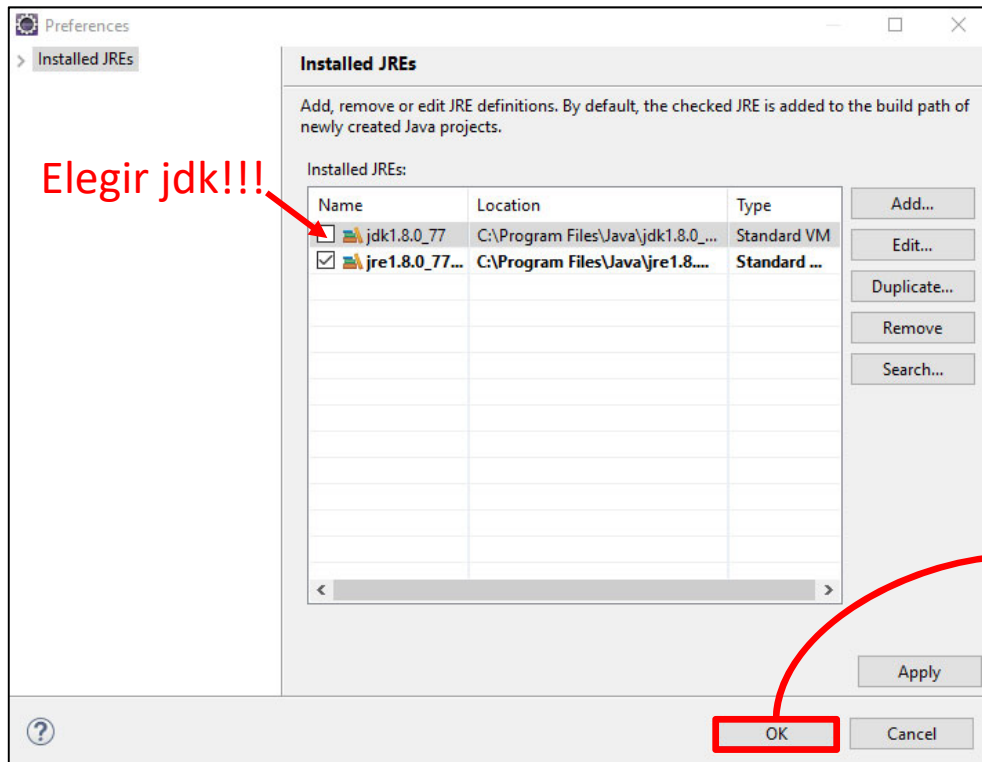
ECLIPSE - DESARROLLO DE UNA APLICACIÓN WEB

ENLAZAR CON TOMCAT



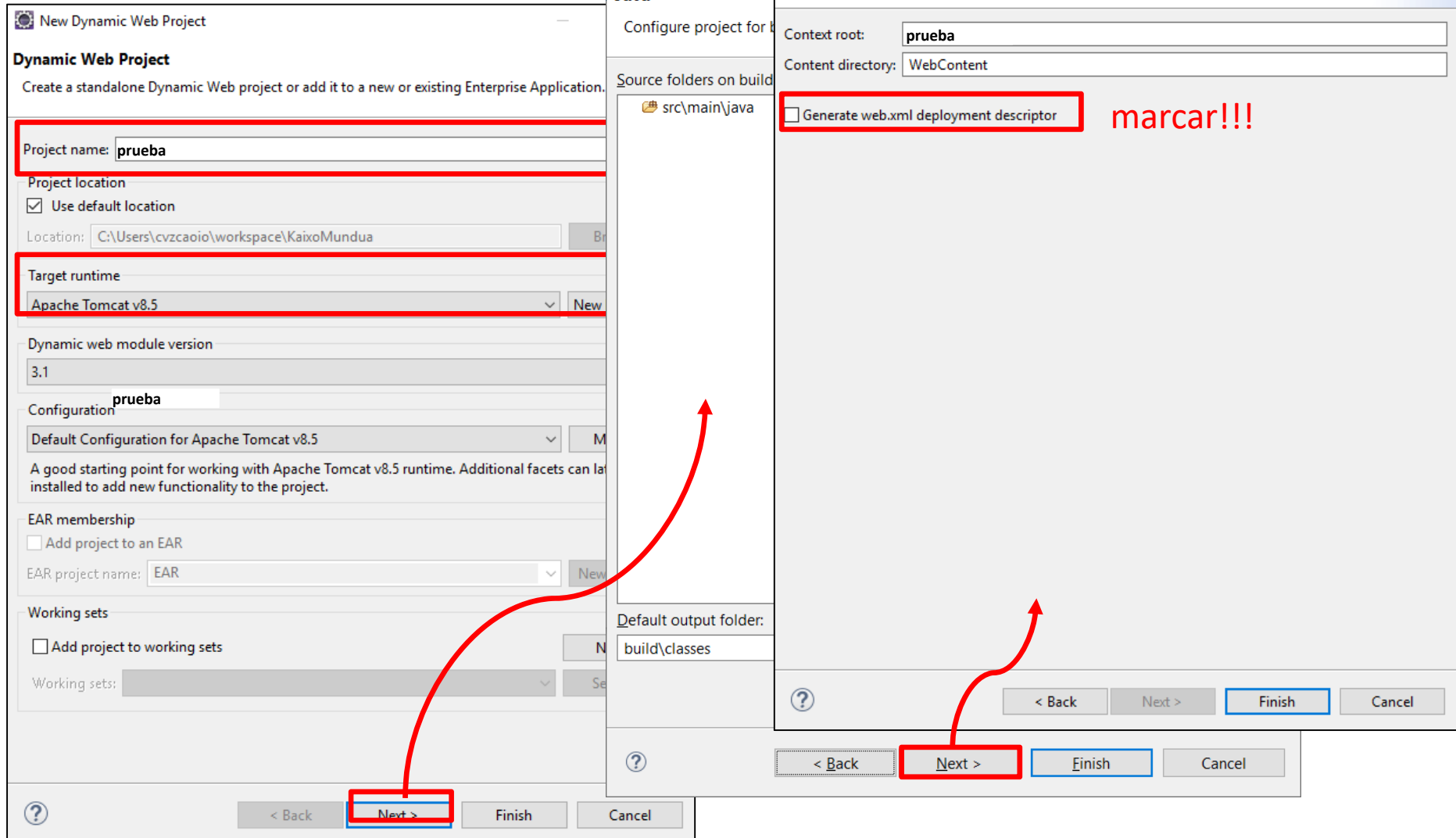
ECLIPSE - DESARROLLO DE UNA APLICACIÓN WEB

ENLAZAR CON TOMCAT



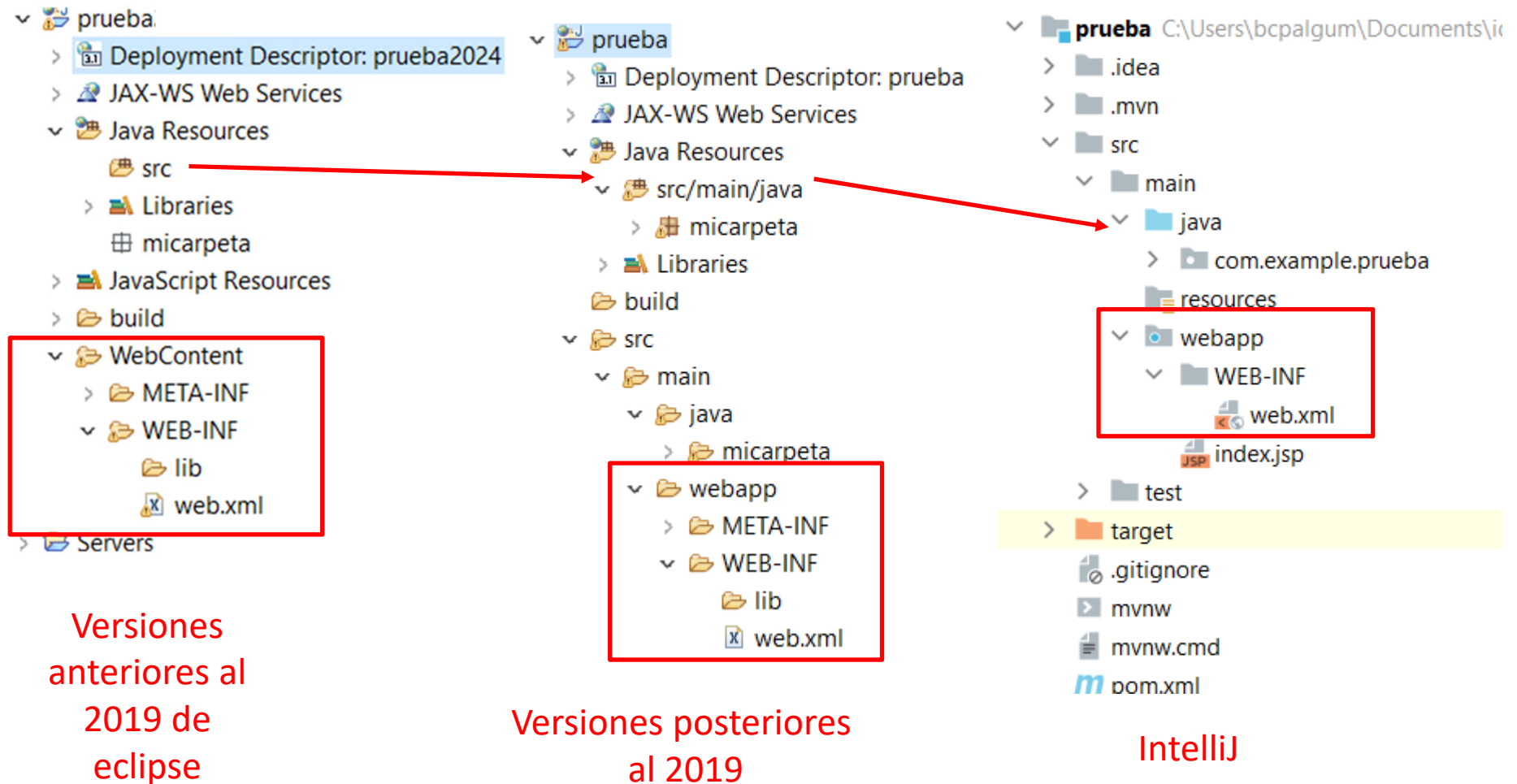
ECLIPSE - DESARROLLO DE UNA APLICACIÓN WEB

ENLAZAR CON TOMCAT



ECLIPSE - DESARROLLO DE UNA APLICACIÓN WEB

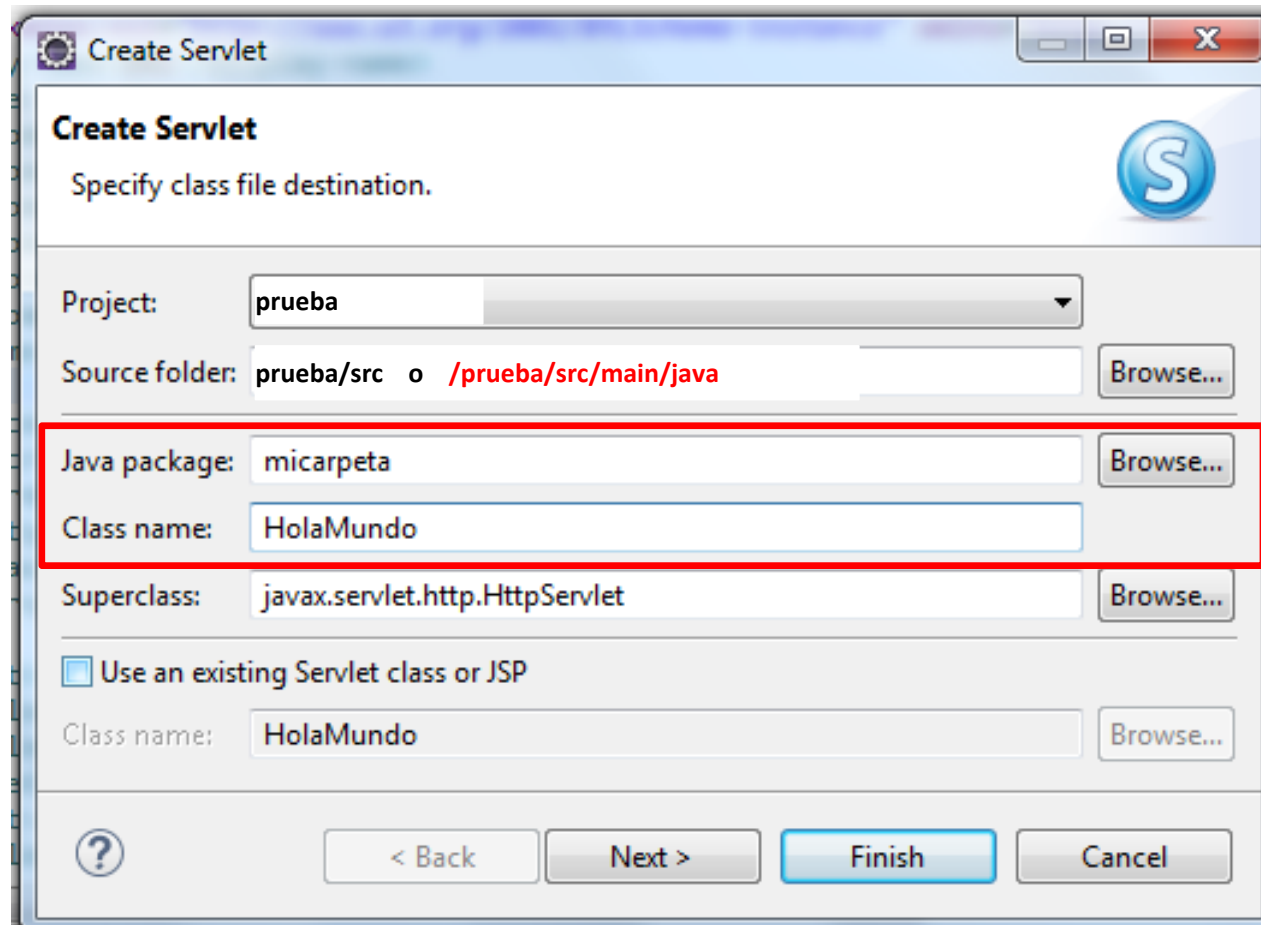
PROYECTO CREADO



SERVLETS

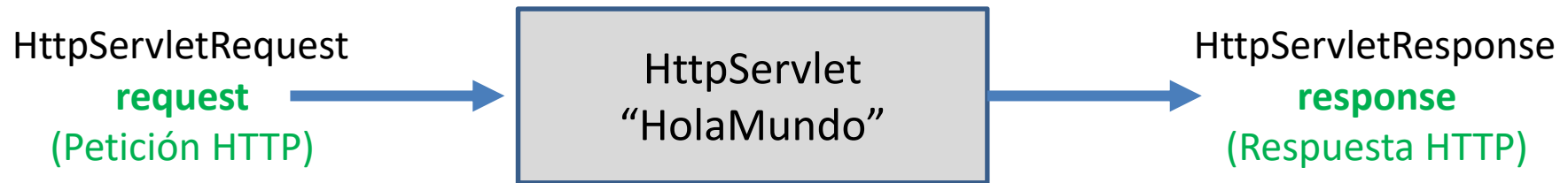
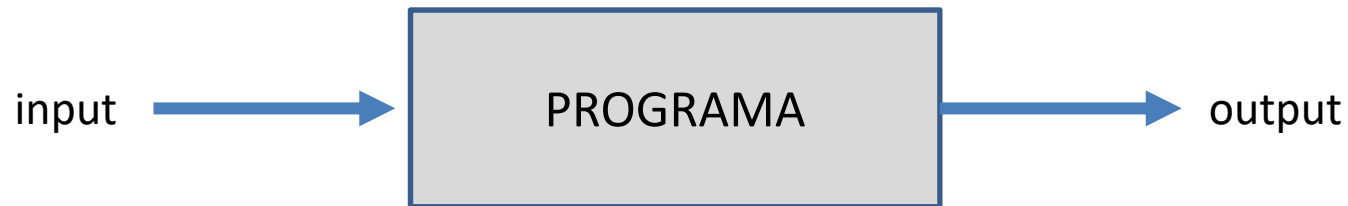
MI PRIMER SERVLET

- File → New → Servlet



SERVLETS

MI PRIMER SERVLET



```
package micarpeta:
```

```
import java.io.*;
```

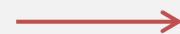
```
import javax.servlet.*;
```

```
import javax.servlet.http.*;
```



APIs Servlet

```
public class HolaMundo extends HttpServlet{
```



Servlet tipo HTTP

```
    public void doGet(HttpServletRequest request, HttpServletResponse response)
```

```
        throws ServletException, IOException {
```

```
        PrintWriter http_out = response.getWriter();
```

```
        http_out.println("Hola Mundo");
```

```
    }
```

```
}
```

SERVLETS - MI PRIMER SERVLET

web.xml

```
<?xml version="1.0" encoding="UTF-8"?>

<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance
  xmlns=http://xmlns.jcp.org/xml/ns/javaee
  xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee
    http://xmlns.jcp.org/xml/ns/javaee/web-app_3_1.xsd" id="WebApp_ID" version="3.1">

  <servlet>
    <servlet-name>HolaMundo</servlet-name>
    <servlet-class>micarpeta.HolaMundo</servlet-class>
  </servlet>
  <servlet-mapping>
    <servlet-name>HolaMundo</servlet-name>
    <url-pattern>/servlet/holamundo</url-pattern>
  </servlet-mapping>

</web-app>
```

Barra del Navegador



<http://localhost:8080/prueba/servlet/holamundo>

SERVLETS - MI PRIMER SERVLET

DEPURACIÓN

```
package micarpeta;

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class HolaMundo extends HttpServlet {

    public void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

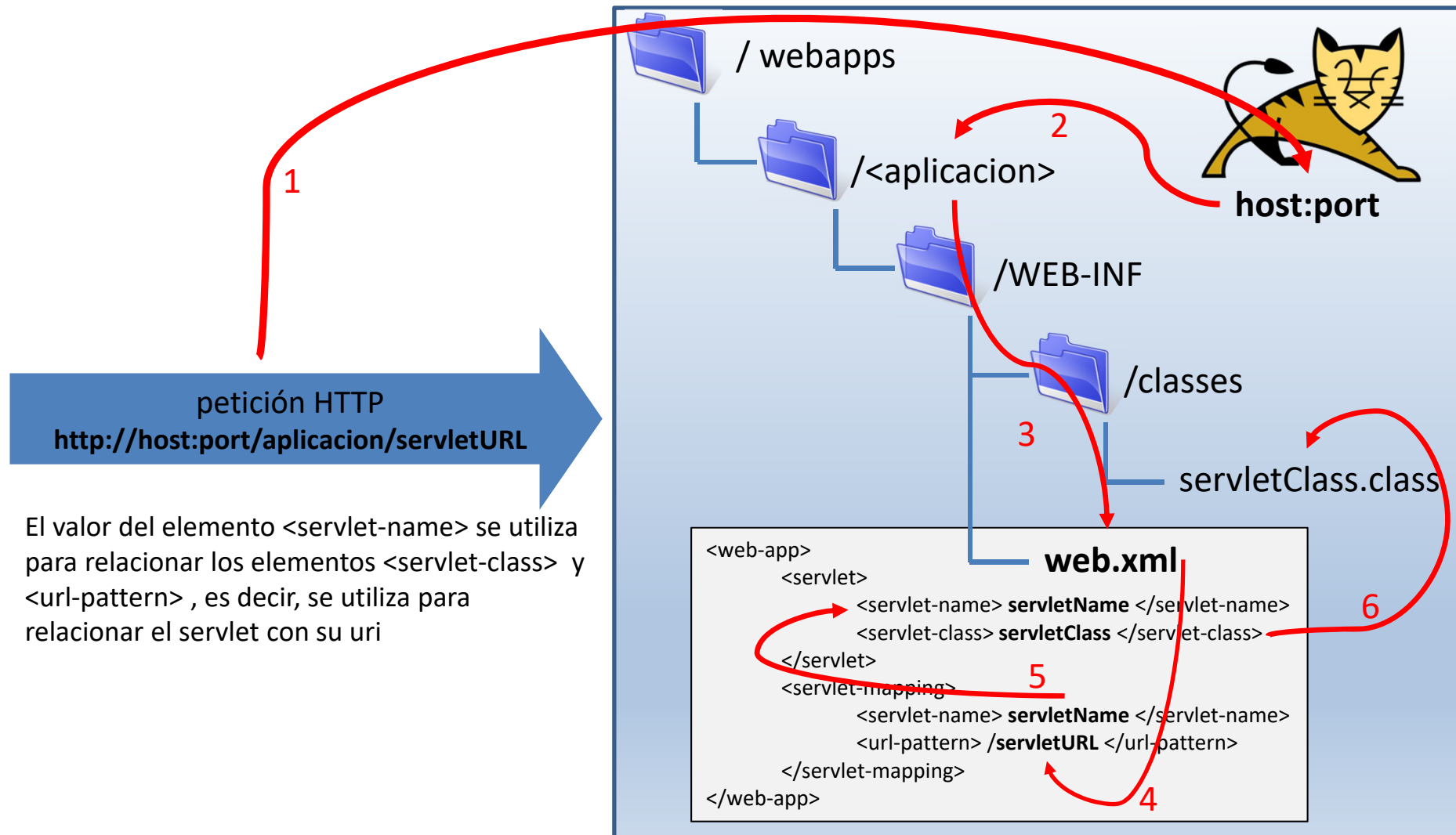
        System.out.println("---> Entering HolaMundo servlet"); // log

        PrintWriter out = response.getWriter();
        out.println("Hola Mundo");

        System.out.println("<--- Exiting HolaMundo servlet"); // log
    }
}
```

SERVLETS - MI PRIMER SERVLET

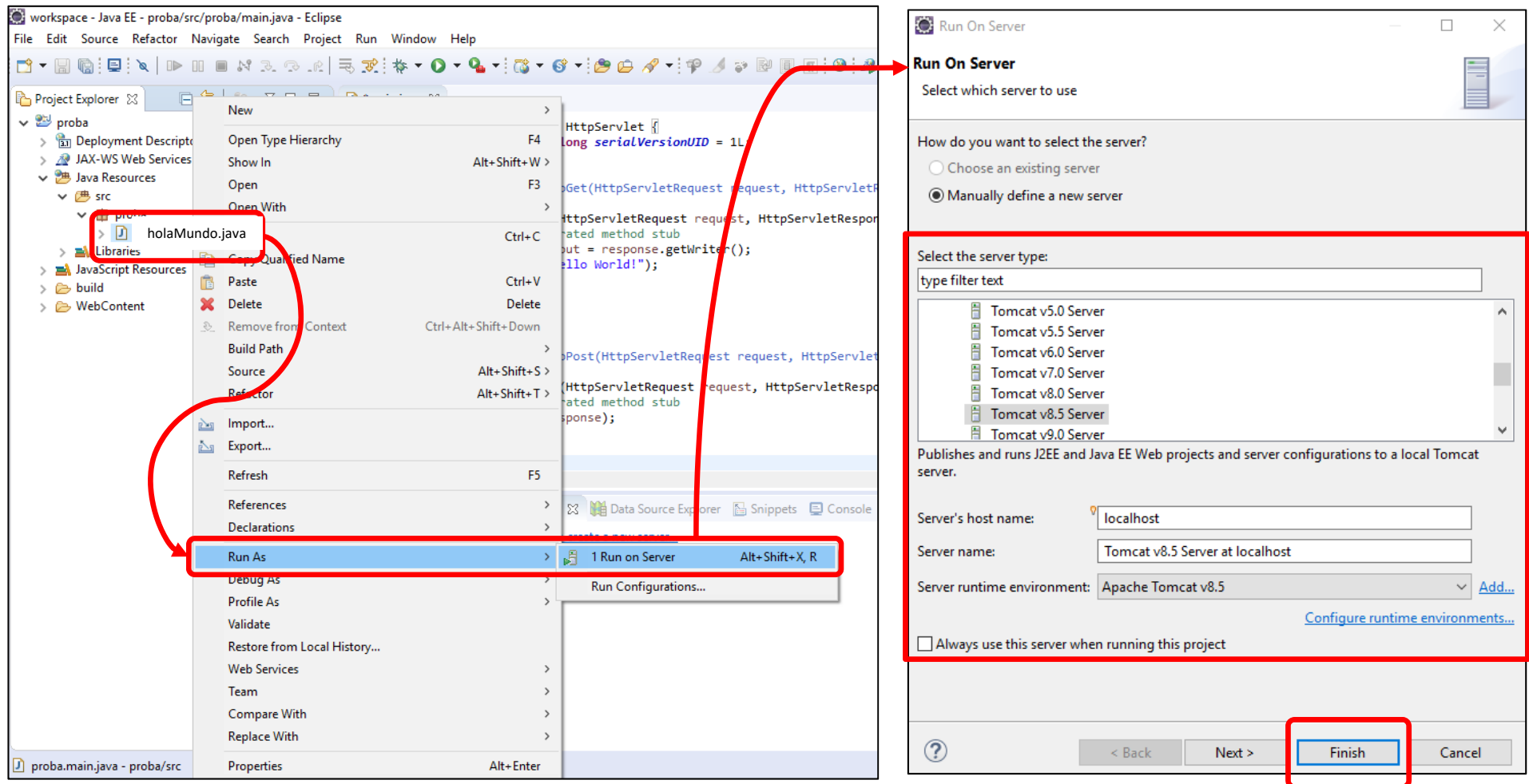
web.xml



SERVLETS - MI PRIMER SERVLET

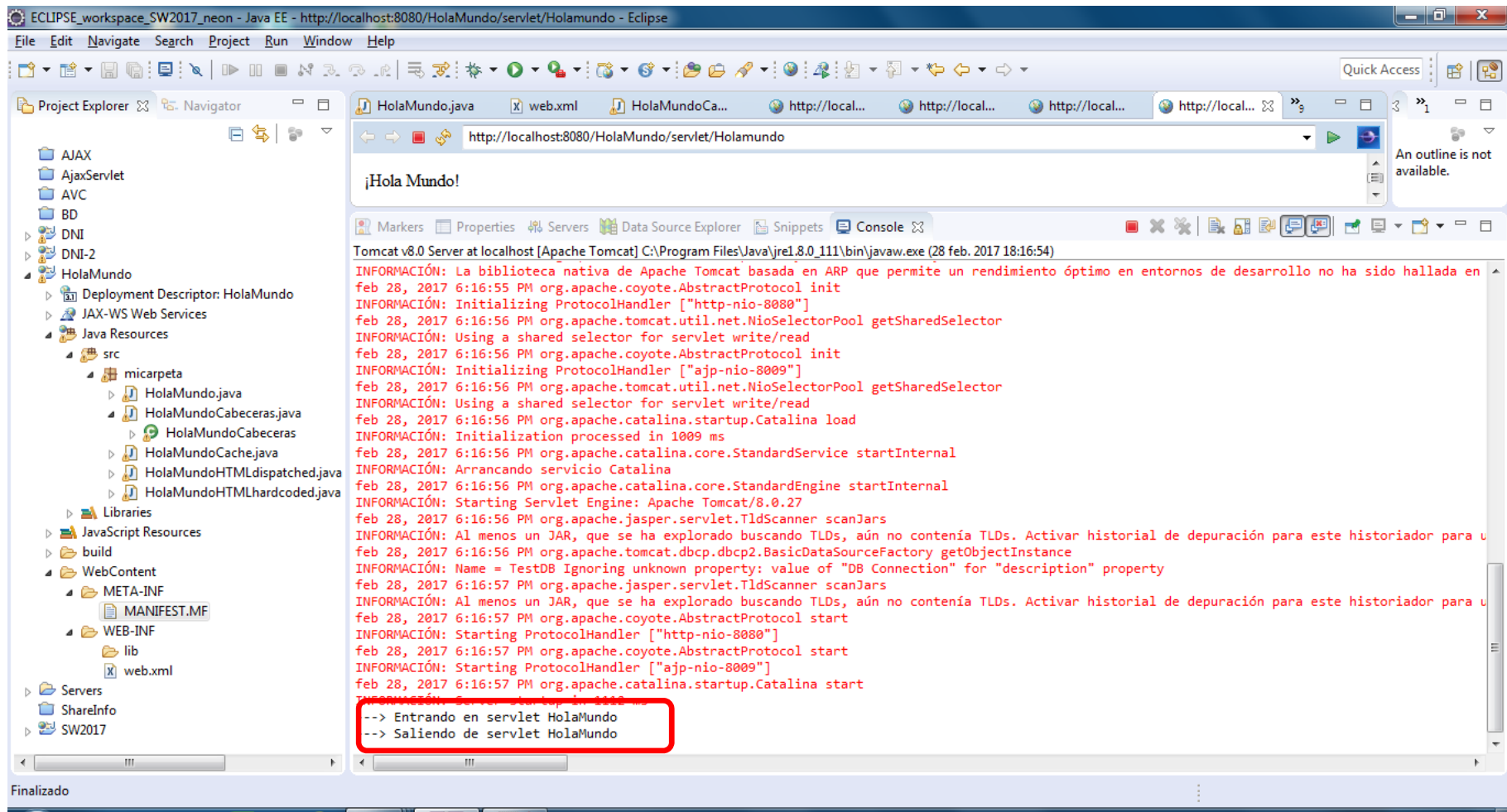
PROBAR (PRIMERA VEZ)

- La aplicación web se puede lanzar desde Eclipse a Tomcat.



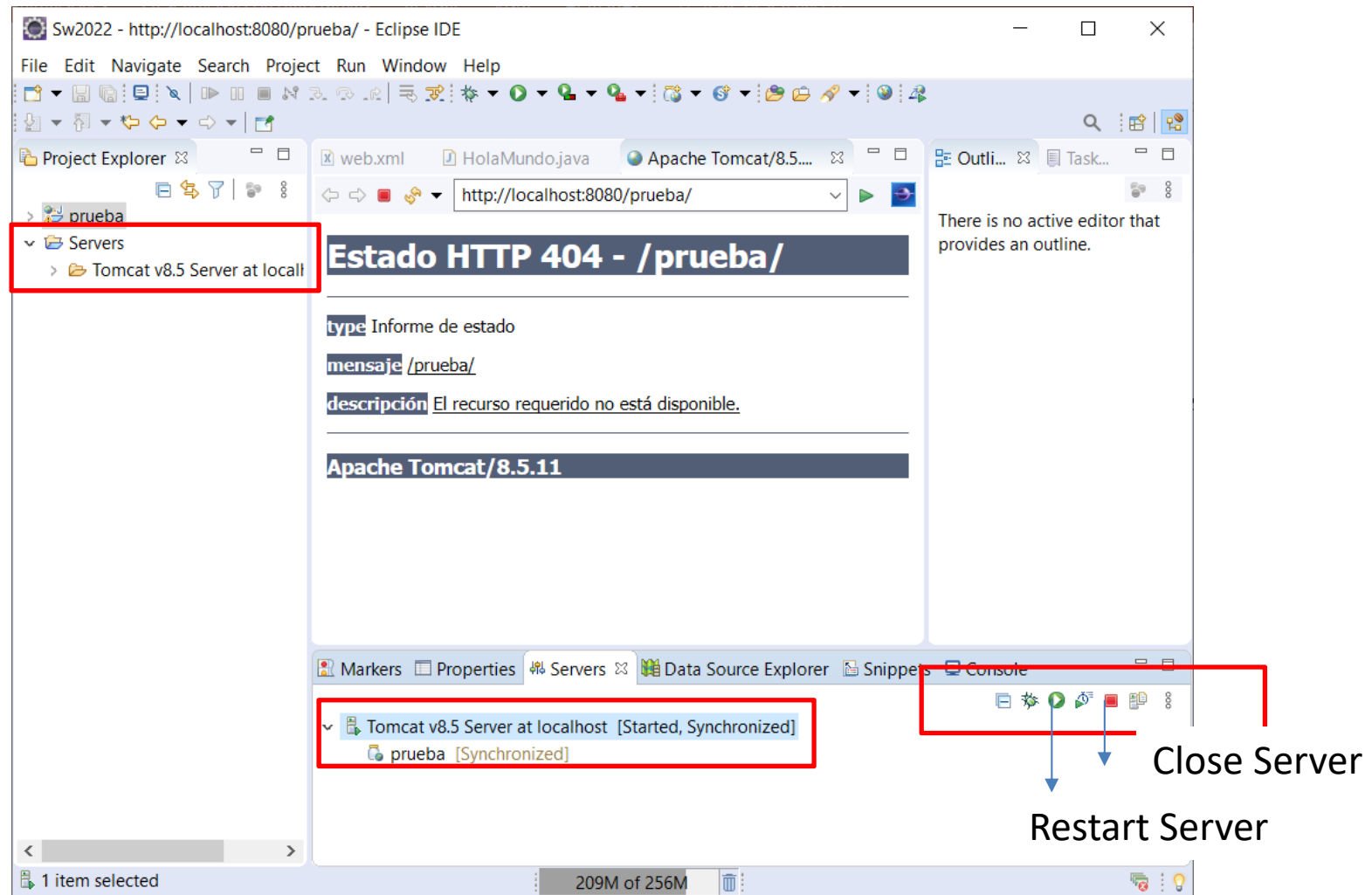
SERVLETS - MI PRIMER SERVLET

DEPURACIÓN



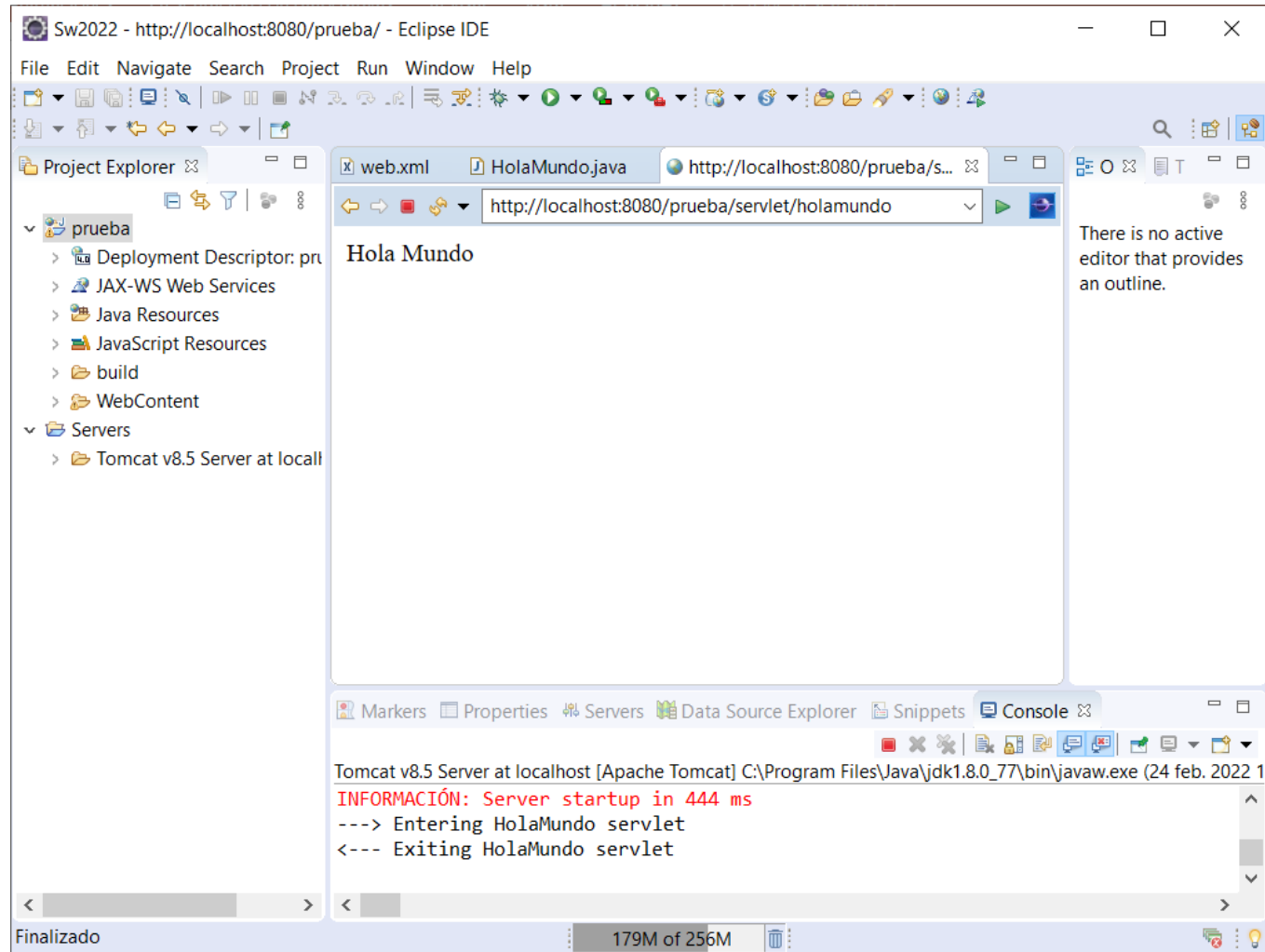
SERVLETS - MI PRIMER SERVLET

PROBAR



SERVLETS - MI PRIMER SERVLET

PROBAR



SERVLETS

MÉTODOS DE **HTTP**SERVLET**REQUEST**

<http://docs.oracle.com/javaee/6/api/javax/servlet/http/HttpServletRequest.html>

- String **getMethod()**
- String **getHeader**(String name)
- java.util Enumeration<String> **getHeaderNames()**
- String **getParameter**(String name)
- java.util Enumeration<String> **getParameterNames()**
-

SERVLETS

MÉTODOS DE **HTTP**SERVLET**RESPONSE**

<http://docs.oracle.com/javaee/6/api/javax/servlet/http/HttpServletResponse.html>

- void **setStatus**(int sc)
- void **setContentType**(String type)
- void **setContentLength**(String type)
- void **addHeader**(String name, String value)
- void **addCookie**(Cookie cookie)
- java.io.PrintWriter **getWriter**()
- void **sendRedirect**(String location)
- ...

SERVLETS

LEER CABECERAS

Actividad: Modificar el programa para que lea todas las cabeceras de la petición

The screenshot shows the Eclipse IDE interface. The top toolbar includes File, Edit, Navigate, Search, Project, Run, Window, and Help. The Project Explorer on the left shows a project named 'HolaMundo' with a 'src' folder containing 'micarpeta' and 'HolaMundo.java'. The Navigator on the right shows the 'HolaMundoCabeceras.java' file. The main editor displays the content of 'HolaMundoCabeceras.java', which is a servlet that reads HTTP headers. The browser window shows the URL 'http://localhost:8080/HolaMundo/servlet/HolaMundoCabeceras' and the HTTP headers: 'Cabeceras petición HTTP: accept: application/x-ms-application, image/jpeg, application/xhtml+xml, image/gif, image/pjpeg, application/x-m... accept-language: es-ES cache-control: no-cache ua-cpu: AMD64 accept-encoding: gzip, deflate user-agent: Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; Win64; x64; Trident/5.0) host: localhost:8080 connection: Keep-Alive'. The console window shows the Tomcat v8.0 Server logs, including the startup sequence and the servlet 'HolaMundoCabeceras' being executed. The status bar at the bottom indicates 'Finalizado' and the date '18:23 28/02/2017'.

```
Cabeceras petición HTTP:
accept: application/x-ms-application, image/jpeg, application/xhtml+xml, image/gif, image/pjpeg, application/x-m
accept-language: es-ES
cache-control: no-cache
ua-cpu: AMD64
accept-encoding: gzip, deflate
user-agent: Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; Win64; x64; Trident/5.0)
host: localhost:8080
connection: Keep-Alive
```

```
Tomcat v8.0 Server at localhost [Apache Tomcat] C:\Program Files\Java\jre1.8.0_111\bin\javaw.exe (28 feb. 2017 18:23:09)
INFORMACIÓN: Name = testub ignoring unknown property: value of 'no' connection for 'description' property
feb 28, 2017 6:23:12 PM org.apache.jasper.servlet.TldScanner scanJars
INFORMACIÓN: Al menos un JAR, que se ha explorado buscando TLDs, aún no contenía TLDs. Activar historial de depuración para este historial para u
feb 28, 2017 6:23:12 PM org.apache.coyote.AbstractProtocol start
INFORMACIÓN: Starting ProtocolHandler ["http-nio-8080"]
feb 28, 2017 6:23:12 PM org.apache.coyote.AbstractProtocol start
INFORMACIÓN: Starting ProtocolHandler ["ajp-nio-8009"]
feb 28, 2017 6:23:12 PM org.apache.catalina.startup.Catalina start
INFORMACIÓN: Server startup in 1213 ms
---> Entrando en servlet HolaMundoCabeceras
Cabeceras petición HTTP:
accept: application/x-ms-application, image/jpeg, application/xhtml+xml, image/gif, image/pjpeg, application/x-ms-xbap, */*
accept-language: es-ES
cache-control: no-cache
ua-cpu: AMD64
accept-encoding: gzip, deflate
user-agent: Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; Win64; x64; Trident/5.0)
host: localhost:8080
connection: Keep-Alive
---> Saliendo de servlet HolaMundoCabeceras
```

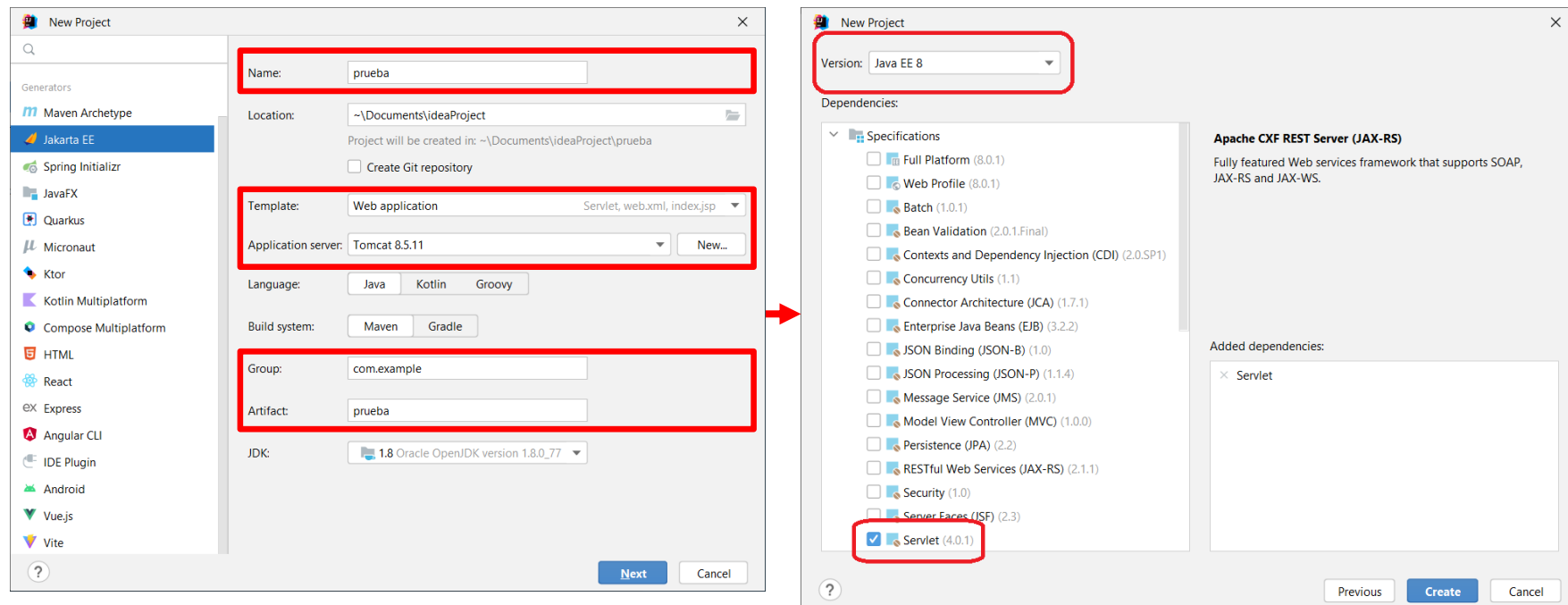
TOMCAT: INTEGRAR CON INTELIJ

Para uso directo desde Tomcat desde IntelliJ:

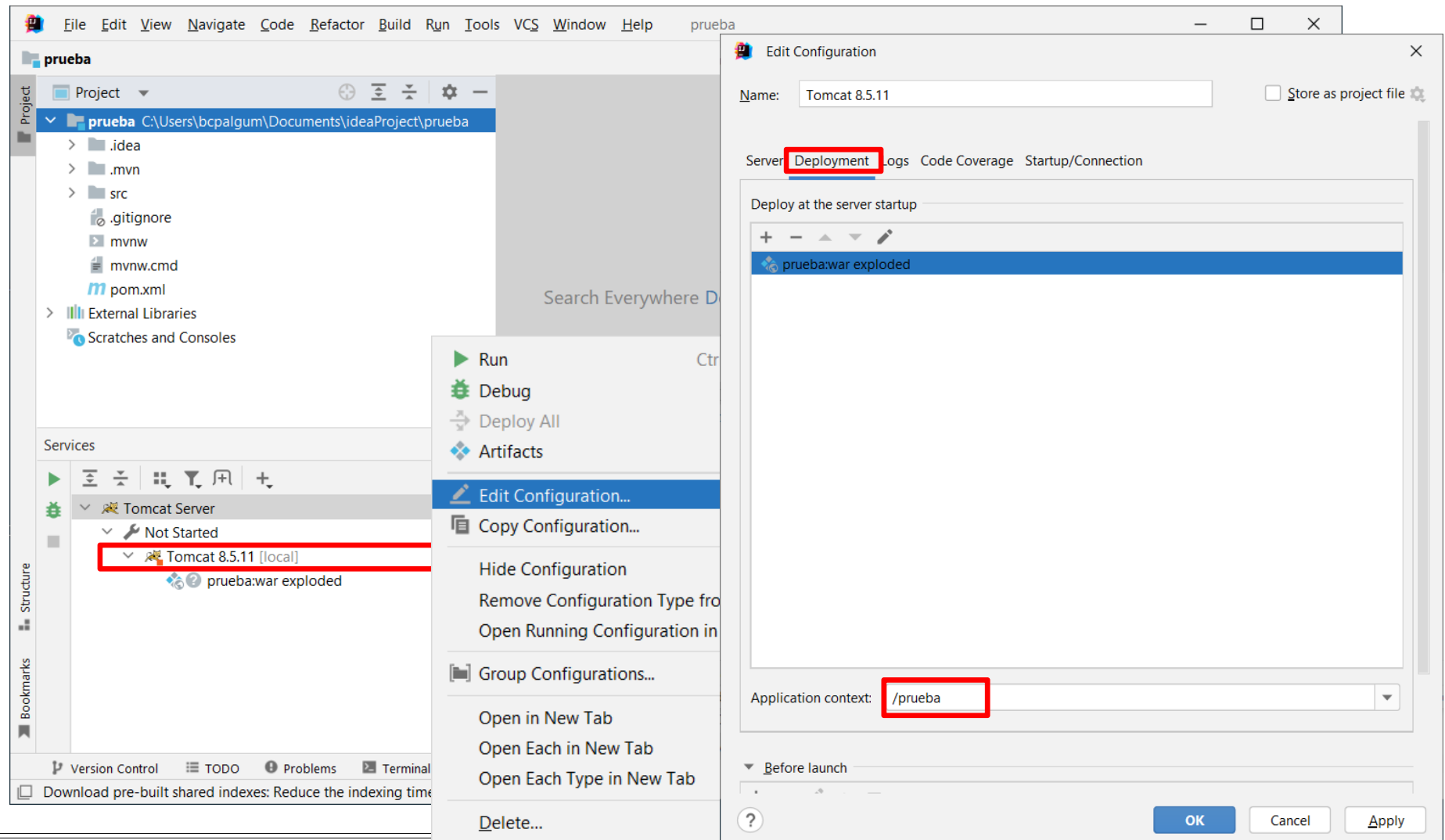
Integration with application servers:

<https://www.jetbrains.com/help/idea/configuring-and-managing-application-server-integration.html>

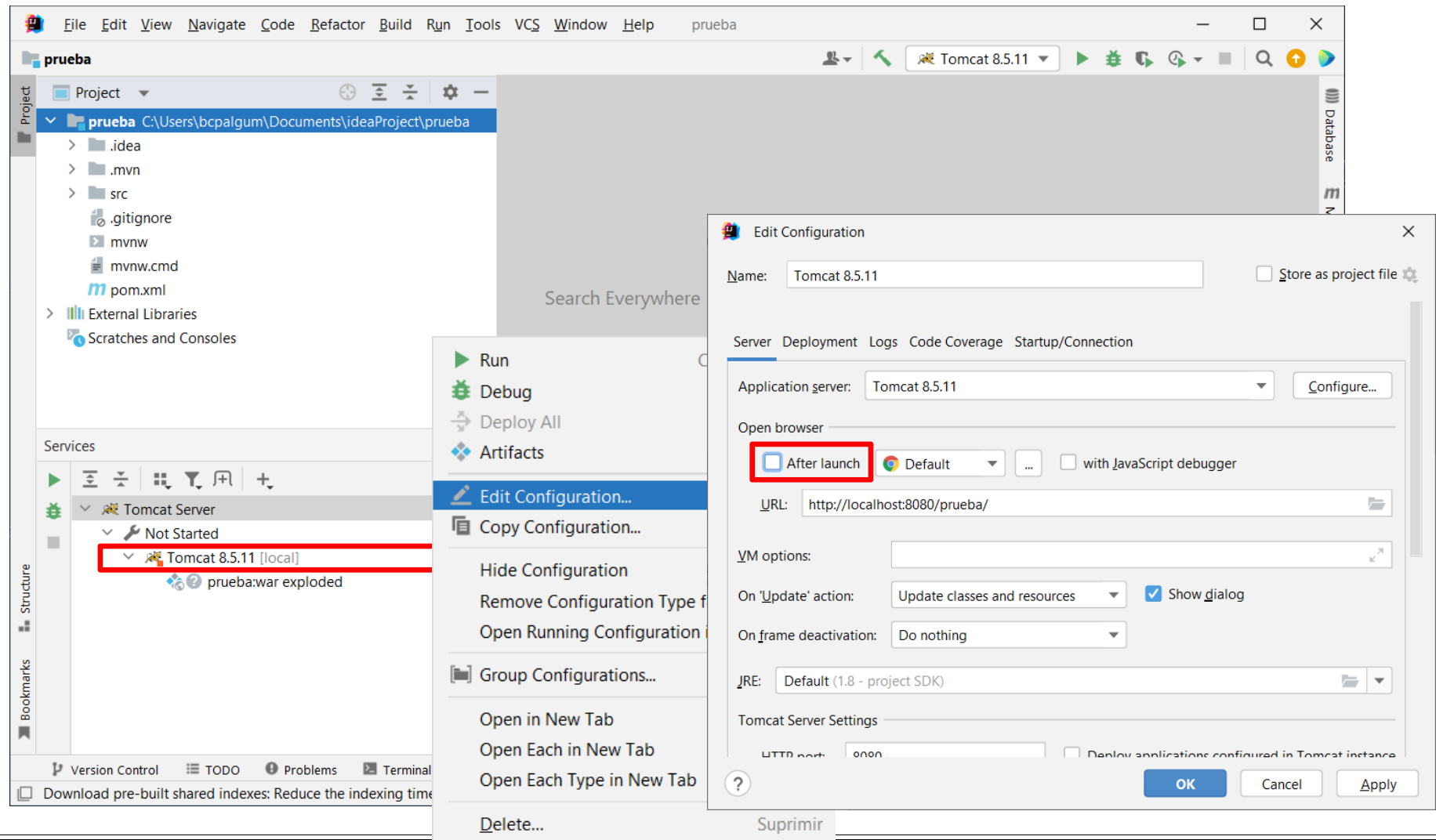
Crear aplicaciones web desde IntelliJ:



TOMCAT: INTEGRAR CON INTELIJ



TOMCAT: PRUEBA



TOMCAT: PRUEBA

