

# **DESPLIEGUE DE APLICACIONES WEB.**

IES LA ENCANTÁ.



## **TEMA 1.**

PRÁCTICA 2. Servidores de Aplicaciones.

# RA y CE.

RA1:Implanta arquitecturas Web analizando y aplicando criterios de funcionalidad.

- CE1a. Se han analizado aspectos generales de arquitecturas Web, sus características, ventajas e inconvenientes.
- CE1d. Se han clasificado y descrito los principales servidores de aplicaciones.
- CE1e.Se ha realizado la instalación y configuración básica de servidores de aplicaciones.
- CE1f. Se han realizado pruebas de funcionamiento de los servidores web y de aplicaciones.
- CE1h.Se han descrito los requerimientos del proceso de implantación de una aplicación Web.
- CE1i. Se han documentado los procesos de instalación y configuración realizados sobre los servidores Web y sobre las aplicaciones.

## 1 Introducción.

Los servidores de aplicaciones van algo más allá que lo servidores web. Se centran en grandes aplicaciones empresariales como banco, aseguradoras u organismos oficiales, por ejemplo WebFamilia se ha implementado sobre un antiguo servidor de aplicaciones llamada Jboss de Red Hat.

En esta práctica se realiza una instalación sencilla de un servidor de aplicaciones, se crea aplicaciones básicas, desplegándolas en el servidor y eliminándolas.

## 2 Enunciado.

La empresa Limontesinos que se dedica al cultivo, procesamiento y exportación de cítricos en la Vega Baja y ha decidido informatizar lo máximo posible su empresa. La idea es informatizar la mayor parte de los procesos que ahora se realizan de forma manual, como la gestión de los riegos, la recolección, proveedores o clientes entre otros muchos factores.

Dada la complejidad del sistema y ante la posibilidad de crecimiento se opta por utilizar el estándar Jakarta sobre servidores de aplicaciones web. Por supuesto estos servidores se han de ser virtualizados.

## 2.1 Creando la imagen.

Se quiere tener una imagen a partir de openjdk:11 ([https://hub.docker.com/\\_/openjdk](https://hub.docker.com/_/openjdk)) con.

- Nombre TUSINICIALES\_Payara.
- Instalar el servidor SSH.
- Tener el usuario TUSINICIALES creado y con acceso por ssh.
- Instalar el servidor de aplicaciones Payara 5 instalado. Instrucciones en <https://blog.payara.fish/back-to-basics-installing-payara-server-5-on-ubuntu>.

- Existen 2 alternativas para instalar el servidor:

- Descargar los ficheros de Payara en la creación de la imagen en el fichero Dockerfile, descomprimir:

- Instalar el comando wget y unzip necesarios para instalar el servidor.
- Para descargar el fichero usar:

```
RUN wget -O payara.zip "url"
```

- "url" el la dirección en la que instalar el servidor

- Descomprimir usando:

```
RUN unzip payara.zip
```

- O descargar en el anfitrión, descomprimir y pasar a la imagen con el comando COPY.

- Configurar el servidor, con un dominio por defecto, activar la seguridad, establecer la contraseña y reiniciar el servidor, definir variables de entorno para el usuario y la contraseña ENV ADMIN="admin" ENV PASSWORD="pedro":

```
#configurar el servidor
```

```
RUN echo 'AS_ADMIN_PASSWORD=\n\
```

```
AS_ADMIN_NEWPASSWORD='${PASSWORD}'\n\
```

```
EOF\n'\
```

```
>> ${INSTALL_PATH}/tmpfile
```

```
RUN echo 'AS_ADMIN_PASSWORD='${PASSWORD} '\n\
```

```
EOF\n'\
```

```
>> ${INSTALL_PATH}/pwdfile
```

```
RUN ${INSTALL_PATH}/payara5/bin/asadmin start-domain \  
  && ${INSTALL_PATH}/payara5/bin/asadmin --user $ADMIN --  
  passwordfile=${INSTALL_PATH}/tmpfile change-admin-  
  password \  
  && ${INSTALL_PATH}/payara5/bin/asadmin --user $ADMIN --  
  passwordfile=${INSTALL_PATH}/pwdfile enable-secure-  
  admin \  
  && ${INSTALL_PATH}/payara5/bin/asadmin restart-domain  
#se borran los temporales  
RUN rm ${INSTALL_PATH}/pwdfile ${INSTALL_PATH}/tmpfile
```

- Exponer los puertos 22,4848 y 8080.
- Al inicial el contenedor basada en la imagen se ha de arrancar el servidor ssh y el servidor de aplicaciones.
- Crear una imagen llamada TUSINICIALESPayara.

Entregar el fichero Dockerfile con las características anteriores comentado.

## 2.2 Definiendo el contenedor.

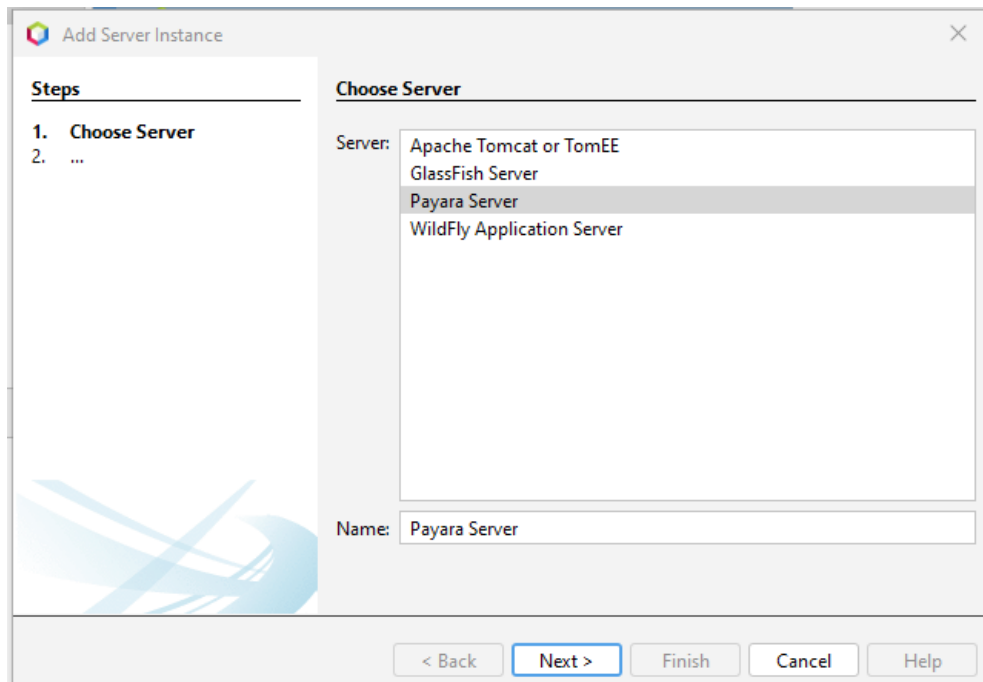
Crear un contenedor a partir de la imagen anterior. Los requisitos son:

1. Mapear los puertos del contenedor con el contenedor.
2. Probar en la línea de comandos del contenedor que los path's son correctos.
3. Probar el ssh.
4. Probar el puerto 4848 ¿qué sucede?
5. Probar el puerto 8080.
6. Ver las aplicaciones instaladas desde el panel de administrador

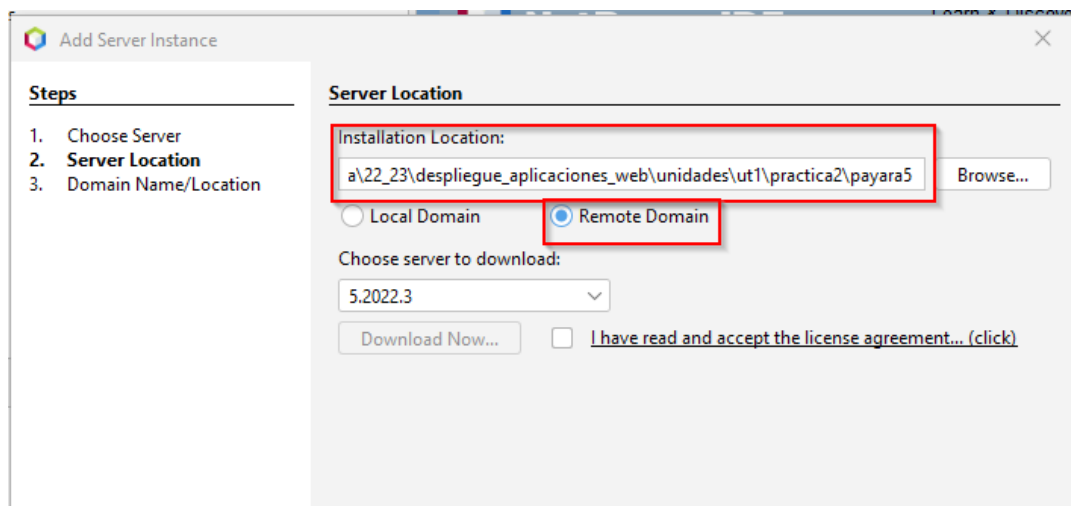
## 2.3 Creando la primera aplicación con Jakarata.

Usando NetBeans crear una aplicación sencilla de Jakarta para el servidor Payara.

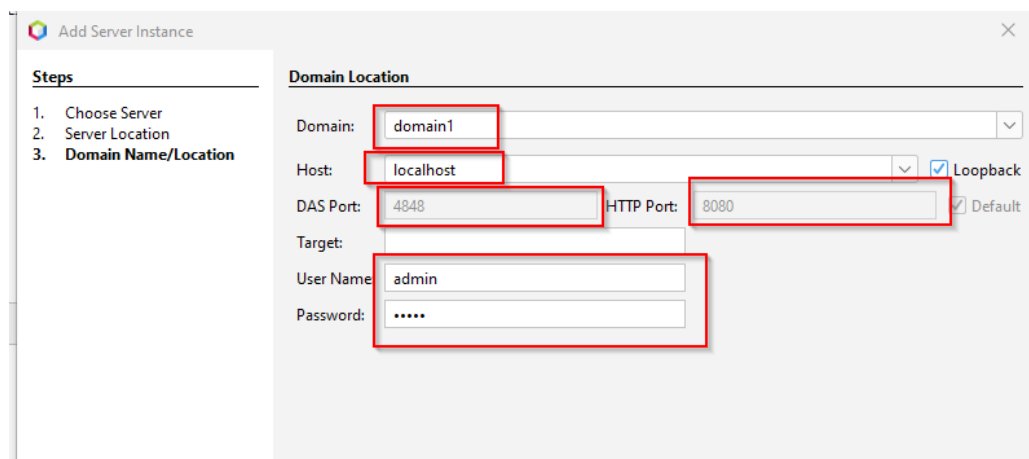
1. En primer lugar registrar el servidor en el IDE, de forma que se pueda realizar el proceso del despliegue de forma rápida. Ir a la pestaña Server y pulsar añadir servidor, seleccionar Payara Server.



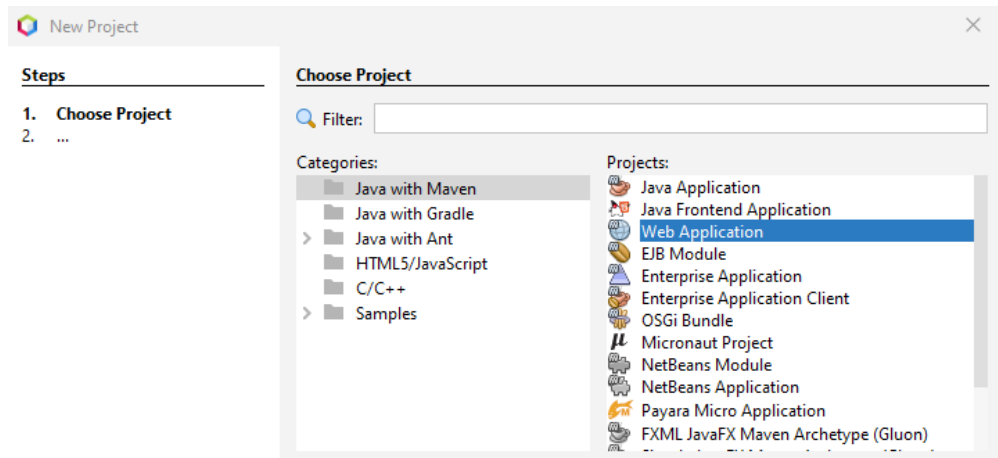
Indicar el directorio en que se encuentran **los ficheros** del servidor en el equipo local (es necesario), e indicar que se trata de una conexión remota:



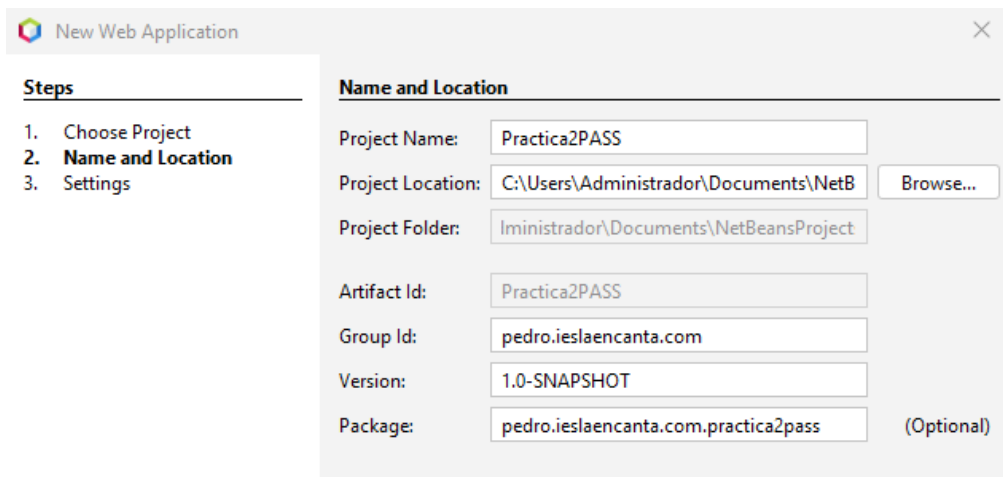
Por último se indican los datos del servidor remoto, en este caso el contenedor:



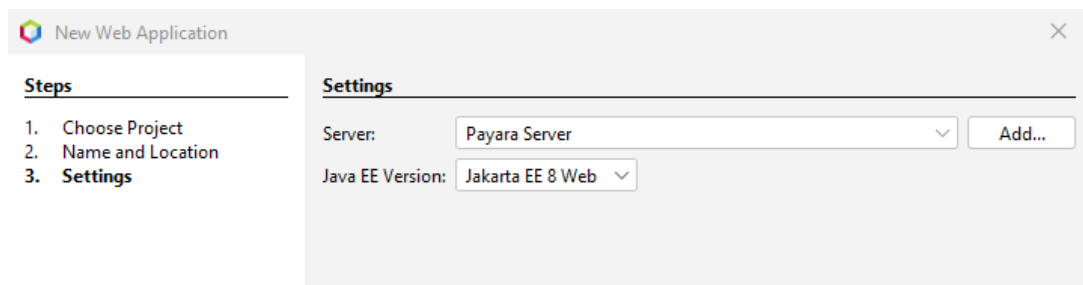
## 2. Ahora crear proyecto con Maven de tipo Web Application:



El nombre del proyecto Practica2TUSINICIALES:



Por último se indica que es el servidor Payara para Jakarta 8:



3. Compilar el proyecto. En caso de obtener un error, comprobar la versión del compilador usada (a de ser la 11). ¿Qué tipo de fichero ha generado?
4. Ir a la página web de administración del servidor a la opción **Applications**, pulsar **deploy** y subir el fichero war generado. ¿Qué sucede?. Investigar como entrar en la aplicación web.
5. Eliminar del servidor la aplicación desplegada.

## 2.4 Tecnologías básicas JSP.

1. Crear otra aplicación del mismo tipo llamada TUSINICIALESJSPYSERVLETS.
2. Añade una página JSP de nombre tus iniciales.
3. En el jsp anterior, crear un h1 con tu nombre y hacer que se genere la tabla de multiplica de 10 en el servidor y se muestre en el cliente (investigar cómo hacerlo).
4. Ejecutar en el contenedor la aplicación.
5. Mostrar la página y comparar el HTML con el contenido del JSP ¿Es diferente? ¿Qué ha sucedido?.

### 3 Cuestiones.

1. ¿Qué es el “Web Profile” de Jakarta? ¿Y el “MicroProfile”? Indicar al menos 3 elementos que se encuentren en el perfil web y no en el perfil micro.
2. Buscar los servidores de aplicaciones que soportan y se han certificado Jakarta EE 9 “Full”.
3. Buscar ahora los servidores de aplicaciones certificados para Jakarta EE 9 perfil web.
4. ¿A qué piensas que se debe la existencia de los diferentes perfiles “Full”, “Web” “Core” y “Micro”? (Este último no es un perfil estándar).
5. En la siguiente lista se detallan los elementos de Jakarta WebProfile 10:
  - Jakarta Annotations 2.1\*
  - Jakarta Authentication 3.0\*
  - Jakarta Bean Validation 3.0
  - Jakarta Concurrency 3.0\*
  - Jakarta Contexts and Dependency Injection 4.0\*
  - Jakarta Debugging Support for Other Languages 2.0
  - Jakarta Dependency Injection 2.0
  - Jakarta Enterprise Beans 4.0 Lite
  - Jakarta Expression Language 5.0\*
  - Jakarta Interceptors 2.1\*
  - Jakarta JSON Binding 3.0\*
  - Jakarta JSON Processing 2.1\*
  - Jakarta Persistence 3.1\*
  - Jakarta RESTful Web Services 3.1\*
  - Jakarta Security 3.0\*
  - Jakarta Server Faces 4.0\*
  - Jakarta Server Pages 3.1\*
  - Jakarta Servlet 6.0\*
  - Jakarta Standard Tag Library 3.0\*
  - Jakarta Transactions 2.0
  - Jakarta WebSocket 2.1\*

Buscar información y explicar la función de los siguientes componentes:

- Jakarta Server Faces.
- Jakarta WebSocket.
- Jakarta Persistence.
- Jakarta Servlet.

### 4 Entrega.

Entregar en Aules antes de la finalización del plazo el fichero Dockerfile y un documento pdf con los apartados:

- Realización de la práctica.



- Respuestas a las preguntas.