



Aplicación Java para pruebas de formularios

1.- Introducción

En el aula virtual tenéis disponible un jar (Java archive, un paquete de software Java) con una aplicación web que permite probar formularios.

En este documento se explica:

- Descarga de la aplicación
- Cómo lanzar la aplicación y cómo personalizar el puerto TCP en el que recibe conexiones
- Qué direcciones http están disponibles para pruebas una vez lanzada

2.- Descarga de la aplicación

Descargar el fichero .jar del aula virtual y colocarlo en algún punto del sistema local. No importa la ubicación, en principio se puede ejecutar desde cualquier dispositivo (como un pen drive) / directorio.

3.- Lanzar la aplicación

La aplicación está empaquetada en formato jar. Este formato contiene todo lo necesario para la ejecución de la aplicación, pero es necesario que dispongamos de un runtime (JRE) de Java, versión 17 o posterior.

Para comprobar que Java está instalado se puede ejecutar en una ventana de comandos (Windows) o un terminal Linux), desde la carpeta en la que hemos dejado el fichero jar:

```
java -version
```

Si muestra la versión, es que tenemos Java instalado.

```
jlllopezalvarez@barrayar:~$ java -version
openjdk version "17.0.8.1" 2023-08-24
OpenJDK Runtime Environment (build 17.0.8.1+1-Ubuntu-0ubuntu122.04)
OpenJDK 64-Bit Server VM (build 17.0.8.1+1-Ubuntu-0ubuntu122.04, mixed mode, sharing)
jlllopezalvarez@barrayar:~$
```

Si no, puede que esté instalado, pero no en el PATH del sistema. Si no está en el path, podremos añadirlo. Si se necesita más información sobre este punto, consultar con el profesor / profesora.

Para lanzar la aplicación debemos ejecutar el siguiente comando

```
java -jar <fichero.jar>
```

Una vez lanzada se debería ver algo similar a a esto:

```
.\ /-----(\
(( )) \----|'---|-|\---|V--|--\\ \\ \
\V) ----| |)| | | | | | (| )))) 
'| -----| :--||-|| | |\ , ||// // 
=====|_|=====|___/_/_/_/_/ 

:: Spring Boot ::                (v3.1.5)
```

```
2023-11-15T12:58:01.090+01:00 INFO 132916 --- [           main] e.j.t.formviewer.FormViewerApplication : jlllopezalvarez/repos/23-24/DME-23-24-AD/ejemplos/FormViewer/target/formviewer-0.0.1.jar started by jlllopez et)
2023-11-15T12:58:01.093+01:00 INFO 132916 --- [           main] e.j.t.formviewer.FormViewerApplication : 
2023-11-15T12:58:01.987+01:00 INFO 132916 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer
```

La aplicación arranca un servidor de aplicaciones Tomcat, y dentro de este una aplicación Spring Boot, que se queda preparada para atender peticiones en <http://localhost:8080>.

Si se produce algún error, aparecerá en los mensajes de log del servidor.

Uno de los errores que se puede producir es que el puerto 8080 esté ocupado, si hay algún otro software instalado que lo esté usando. En este caso el error será algo parecido a esto:

```
*****
APPLICATION FAILED TO START
*****

Description:

Web server failed to start. Port 8080 was already in use.

Action:

Identify and stop the process that's listening on port 8080 or configure this application to listen on another port.
```

En este caso podemos optar por arrancar el servicio en otro puerto, por ejemplo, el 8083. Para esto habría que ejecutar:

```
java -jar <fichero.jar> --server.port=8083
```

4.- Direcciones disponibles

La aplicación tiene abiertas tres direcciones, para poder realizar tres tipos de peticiones:

- Peticiones GET: <http://localhost:8080/view-form/get>
- Peticiones POST: <http://localhost:8080/view-form/post>
- Peticiones POST con ficheros adjuntos: <http://localhost:8080/view-form/files>

Además, para poder probar que la aplicación funciona sin necesidad de crear un formulario HTML, hay tres formularios de prueba. Uno por cada tipo de petición:

- Peticiones GET: <http://localhost:8080/view-form/test-form-get>
- Peticiones POST: <http://localhost:8080/view-form/test-form-post>
- Peticiones POST con ficheros adjuntos: <http://localhost:8080/view-form/test-form-files>

5.- Restricciones y limitaciones

En la versión actual el campo para prueba de ficheros debe llamarse necesariamente “file” (name=“file”).