

## ¿Qué es Spring?

Como definición podemos decir que Spring es un framework de código abierto para la creación de aplicaciones empresariales Java, con soporte para Groovy y Kotlin. Tiene una estructura modular y una gran flexibilidad para implementar diferentes tipos de arquitectura según las necesidades de la aplicación.

## ¿Qué es un framework?

Spring se considera un framework, pero este es un concepto que en ocasiones se confunde, así que vamos a ver qué es un framework.

Una librería es un conjunto de clases, de funciones y de utilidades que nos permiten realizar algunos procesos. Un buen ejemplo de librería sería una librería matemática, a la cual le podemos dar muchos datos y nos puede calcular, por ejemplo, la desviación típica, o le podríamos plantear una integral y la podría resolver.

A diferencia de una librería, un framework es:

- Un conjunto de artefactos software, es decir, que puede incluir una librería, de conceptos y de metodologías.
- Nos provee de un mecanismo genérico para resolver uno o más problemas de un tipo determinado.
- Es extensible a través de código escrito por los usuarios.
- Ofrece facilidad para el desarrollo y despliegue.

Si tuviéramos que desarrollar una aplicación web, podríamos utilizar un framework que nos facilite la tarea, que nos aporte soluciones a ese desarrollo. Uno de ellos podría ser, por ejemplo Spring MVC, que nos permitiría crear fácilmente una aplicación web, ya que nos aislaría de determinados problemas, como el hecho de crear servlet o registrar las peticiones, así nos podríamos dedicar a lo que realmente importa.

## Qué es una aplicación empresarial

Una aplicación empresarial normalmente es:

- Una gran aplicación, es decir, suele ser bastante amplia y orientada a un ámbito comercial o industrial.
- Compleja, es decir, con bastantes funcionalidades, con muchos requisitos y que será utilizada por muchas personas, por lo que debería ser escalable, que pueda crecer con el tiempo, es decir, que si hoy tenemos mil usuarios y dentro de un año tenemos un millón, que no haya que rehacer la aplicación de nuevo.
- Distribuida, es decir, que incluso la podamos tener deslocalizada en distintos servidores.
- Crítica, es decir, que no permita o no tolere fallos e incluso, si tuviéramos algún tipo de fallo, que sea capaz de reponerse.
- Orientada a desplegarse dentro de redes corporativas o, el esquema más usual a día de hoy, en internet a través de la nube.

- Centrada en los datos, por lo que todas las funcionalidades se suelen desarrollar en torno a los mismos.
- Intuitiva, de un uso fácil para evitar el rechazo de los usuarios.
- Suele tener, además, unos grandes requisitos de seguridad y de mantenibilidad.

Todas estas serían las características que definirían a una aplicación empresarial.

## **Spring funciona sobre JVM**

Spring funciona con Java, aunque también tiene soporte para otras tecnologías.

Inicialmente se diseñó para trabajar con Java SE y algunas especificaciones o APIs de Java EE, pero a día de hoy trabaja con el JDK 8 y JDK 9. También podemos trabajar con Groovy y tiene soporte para Kotlin.

## **Spring tiene estructura modular**

Spring ya no es solo un framework para la inyección de dependencias, sino que tiene toda una familia de proyectos que abarcan muchos ámbitos: el ámbito de desarrollo de aplicaciones web, aplicaciones web reactivas, seguridad, servicios web, microservicios, Android, etcétera.

Además, dentro de alguno de esos proyectos, podemos encontrar que tiene una estructura modular, es decir, que está orientada a poder tener distintos módulos que agrupan diversas funcionalidades, desde el contenedor de inversión de control, la programación orientada a aspectos, el acceso a datos, etcétera.

## **Spring es flexible**

Spring nos permite desarrollar todo tipo de aplicaciones diferentes:

Aplicaciones de escritorio, aplicaciones de línea de comando, aplicaciones web clásicas, web reactivas, microservicios...

Aplicaciones que acceden a base de datos vía SQL directamente, a través de algún tipo de ORM, bases de datos NoSQL...

Aplicaciones con esquemas de seguridad clásica donde almacenamos en nuestra base de datos los elementos de seguridad, clave o credenciales, pero también con otros sistemas.

Aplicaciones pequeñas, medianas y grandes, aplicaciones que tienden a ser escalables, aplicaciones que vayamos a gestionar a través de contenedores, aplicaciones que necesitemos desplegar en la nube.

## **Crear Nuestro Primer proyecto en Spring**

Link video Crear proyecto con java Spring: <https://youtu.be/qdPEZjhziIU>