

Accenture

Eduardo Aguilar López

Spring framework

Escrito inicialmente por Rod Johnson, fue lanzado por primera vez en el mes de Junio del año 2003 bajo la licencia Apache 2.0, siendo una plataforma Java de código abierto. Convirtiéndose desde entonces en el framework más popular para Java empresarial, para crear código de alto rendimiento, liviano y reutilizable. Ya que su finalidad es estandarizar, agilizar, manejar y resolver los problemas que puedan ir surgiendo en el trayecto de la programación.



Spring, ofrece como elemento clave el soporte de infraestructura a nivel de aplicación, brindando un completo modelo tanto para la configuración como para la programación de aplicaciones empresariales desarrolladas bajo Java, sin discriminación en cuanto al despliegue de la plataforma.

Todo esto trae consigo una gran ventaja, ya que permite que los equipos de desarrollo puedan enfocarse directamente en la lógica empresarial que requiere la aplicación, haciendo el proceso más corto, rápido y eficaz, ahorrando líneas de código evitando tareas repetitivas.

Spring se puede considerar como el padre de los frameworks Java, ya que da soporte a varios frameworks como: Hibernate, Struts, Tapestry, EJB, JSF, entre otros.

Características

Entre las características de Spring, tenemos las siguientes que ofrecen una cantidad considerable de servicios:

- Tecnologías: como la inyección de dependencias, eventos, recursos, i18n, validación, enlace de datos, conversión de tipo, SpEL.
- Acceso a datos: soporte DAO, JDBC, ORM, Marshalling XML.
- Gestión de transacciones.
- Integración: comunicación remota, JMS, JCA, JMX, correo electrónico, tareas, programación, caché.
- Pruebas (Testing): simulacro de objetos, el framework TestContext, Spring MVC prueba, WebTestClient.

- Programación orientada a aspectos (AOP): permite la implementación de rutinas transversales.
- MVC (Modelo Vista Controlador).
- Seguridad.
- Frameworks web: Spring WebFlux y Spring MVC.
- Procesamiento de datos por lotes.
- Administración Remota: a través de este módulo se puede configurar la visibilidad y gestión de los objetos Java para la configuración local o remota vía JMX.
- Es un framework liviano debido a su implementación POJO (Plain Old Java Object), Spring Framework no obliga al programador a heredar ninguna clase ni a implementar ninguna interfaz.

Aunque hay que destacar la siguiente:

- La Inyección de Dependencias (Dependency Injection): Al momento de escribir una aplicación Java compleja, las clases de la aplicación deben ser lo más independientes posible de otras clases Java, para aumentar la posibilidad de reutilizarlas y probarlas independientemente de otras clases, mientras se prueban las unidades. Básicamente la inyección de dependencias (DI) ayuda a unir estas clases y al mismo tiempo mantenerlas

Spring MVC

Spring MVC es un subproyecto Spring, un marco de Java para crear aplicaciones web y servicios Restful de forma óptima y fácil. Sigue el patrón de diseño Modelo-Vista-Controlador e implementa todas las características básicas de un marco de resorte central como Inversión de Control e Inyección de Dependencia.

El patrón MVC da como resultado la separación de los diferentes aspectos de la aplicación como lógica de entrada, lógica comercial y lógica de la interfaz de usuario, al mismo tiempo que proporciona un acoplamiento entre los elementos.

Aspectos más importantes de Spring MVC

- La Vista: Éste es responsable de hacer una representación de los modelos y, en general una salida HTML, una vista representa la información proporcionada en un formato particular. Normalmente, JSP+JSTL se utilizan para crear una página de vista. Aunque Spring también es compatible con otras tecnologías de visualización como Apache Velocity, Thymeleaf y FreeMarker.
- El Controlador: Procesa las solicitudes de los usuarios, construye un modelo que sea apropiado y luego, lo pasa a la vista de su representación, Es responsable de administrar el flujo de la aplicación Spring MVC.
- El Modelo: los datos de la aplicación son encapsulados, y en general consistirán en POJO.

Como funciona Spring MVC

Después de recibir una solicitud HTTP, el DispatcherServlet hace la consulta a HandlerMapping para llamar al controlador apropiado, éste toma la solicitud y llama a los métodos de servicio según el método GET o POST, y el DispatcherServlet tomará la solicitud con la ayuda de ViewResolver. Después de todo ese trayecto, DispatcherServlet pasa los datos del modelo a la vista y finalmente éste se representa en el navegador.

Ventajas de Spring MVC

- Mapeo flexible: proporciona las anotaciones específicas que redirigen fácilmente la página.
- Desarrollo rápido: Spring MVC facilita el desarrollo rápido y paralelo.
- Ligero: utiliza un contenedor de servlet liviano para desarrollar e implementar su aplicación.
- Código comercial reutilizable: en lugar de crear nuevos objetos, nos permite reutilizar los objetos existentes.
- Fácil de probar: en Spring, generalmente creamos clases de JavaBeans que le permiten inyectar datos de prueba utilizando los métodos setter.
- Funciones separadas: Spring MVC separa cada función, donde el objeto modelo, el controlador, el objeto de comando, la resolución de vista, DispatcherServlet, el validador, etc. pueden ser cumplidos por un objeto especializado.

Spring Boot

Spring Boot es un framework desarrollado para el trabajo con Java como lenguaje de programación. Se trata de un entorno de desarrollo de código abierto y gratuito. Spring Boot cuenta con una serie de características que han incrementado su popularidad y, en consecuencia, su uso por parte de los desarrolladores back-end. Estas son las características más destacadas de este framework:

- Permite crear todo tipo de aplicaciones en el lado del back-end de forma independiente.
- Facilita el trabajo con otras herramientas como Tomcat, Jetty o Undertow. Lo hace directamente, sin necesidad de implementar archivos específicos para ello.
- Simplifica las dependencias para mejorar la configuración final del proyecto que se desarrolla con Spring Boot.
- Se trata de un framework que se configura de manera simple y es compatible con bibliotecas de terceros.
- Facilita la creación de listas, controles de estado y mejora la configuración externa para el desarrollo de aplicaciones.
- No es necesario generar código para los aspectos que controla Spring Boot y no hay requisitos para la configuración XML.

Gracias a estas características, Spring Boot facilita la creación de todo tipo de aplicaciones basadas en él de manera independiente con el mínimo esfuerzo por parte de los desarrolladores. Y es que se trata de una tecnología que facilita que los desarrolladores se centren solo en la parte de programación, sin necesidad de preocuparse por aspectos como la arquitectura.



En esencia, por tanto, se trata de un enfoque que piensa el diseño de las aplicaciones Java de manera modular. Esto es, se desarrolla de tal forma que se crean pequeñas piezas que funcionan de manera independiente pero que se pueden integrar para funcionar de forma conjunta. Esto puede llegar a tener diversas ventajas como:

- Reducción del equipo de trabajo, haciendo que este sea mínimo
- Al tratarse de un enfoque de desarrollo modular resulta más sencillo escalar la aplicación según las necesidades del cliente
- Funcionalidades modulares, cada modulo funciona de manera independiente
- Libertad para los desarrolladores a la hora de desplegar servicios de manera independiente
- Despliegue muy rápido de la aplicación y puesta en marcha de la misma mediante el uso de los contenedores.

Esto es lo que se puede conseguir con Spring Boot. Este framework permite el desarrollo de proyectos más pequeños que convergen en algo más grande. La creación de estos microservicios está cada vez más a la orden del día, sobre todo, por su gran capacidad de escalabilidad en un entorno tan potente como es el de Java.

Bibliografía

- <https://openwebinars.net/blog/conoce-que-es-spring-framework-y-por-que-usarlo/>

- <https://www.cnac.es/noticias/que-es-spring-mvc/#:~:text=Spring%20MVC%20es%20un%20subproyecto,Control%20e%20Inyecci%C3%B3n%20de%20Dependencia>.
- <https://www.tokioschool.com/noticias/spring-boot/>