



# ***Universidad Autónoma de Santo Domingo***

**Facultad de Ciencia  
Escuela de Informática**

## **Asignatura:**

Laboratorio Lenguaje de Programación III

## **Tema:**

Spring Framework - Describir las funcionalidades de este Framework

## **Estudiante:**

Genesis Batista Mejía - 100572904

## **Maestro:**

Radhames Silverio González

## ¿Qué es Spring Framework (Spring)?

Spring Framework (Spring) es un marco de desarrollo de software de código abierto que proporciona soporte de infraestructura para crear aplicaciones basadas en Java en cualquier plataforma de implementación.

Lanzado en junio de 2003 por Rod Johnson bajo la licencia Apache 2.0, Spring Framework está alojado por SourceForge.

## Cómo funciona Spring Framework

Una aplicación web (arquitectura en capas) comúnmente incluye tres capas principales:

1. Una **capa de presentación**, o interfaz de usuario , que maneja la visualización del contenido y las interacciones del usuario.
2. Una **capa de lógica empresarial** que se ocupa de las especificaciones funcionales subyacentes de un programa.
3. Una **capa de acceso a datos** que supervisa los procedimientos de recuperación de datos.

Cada capa depende de las demás para que una aplicación funcione. Por ejemplo, la capa de presentación podría necesitar comunicarse con la capa de lógica de negocio, que a su vez interactúa con la capa de acceso a datos. Cada una de estas interacciones genera lo que comúnmente se conoce como *dependencias* y crea acoplamientos entre los distintos componentes del sistema.

Una de las características principales de Spring es su capacidad para realizar inyección de dependencias , un patrón de programación que permite a los desarrolladores crear arquitecturas más desacopladas . Spring comprende las diferentes anotaciones de Java que un desarrollador coloca sobre las clases y puede ayudar a garantizar que todas las instancias creadas tengan las dependencias correctamente rellenas.

Para que Spring Framework instancia objetos y llene las dependencias, un programador simplemente le indica a Spring qué objetos administrar y cuáles son las dependencias de cada clase mediante anotaciones. Por ejemplo, la anotación **@component** permite a Spring saber qué clases administrar, marca los objetos como componentes administrados e identifica las clases para la inyección de dependencias. Por otro lado, la anotación **@autowired** le indica a Spring cómo gestionar la instanciación de la clase y cómo buscar esa dependencia entre componentes y clases para encontrar una coincidencia.

## Ventajas de Spring Framework

La principal ventaja de Spring es que elimina muchas de las complejidades asociadas con la programación en Java y ayuda a agilizar el desarrollo y las pruebas de aplicaciones. Esto se debe a que es un framework ligero, admite aplicaciones débilmente acopladas y proporciona plantillas predefinidas para JDBC, Hibernate , etc. Otra ventaja es que Spring es un framework no invasivo, ya que no obliga a los desarrolladores a heredar clases ni a implementar interfaces durante el desarrollo.

Con Spring, los desarrolladores pueden escribir código fácilmente para usar API de persistencia para almacenar y acceder a datos persistentes en una base de datos . También pueden crear aplicaciones web basadas en la arquitectura Spring Web MVC, un framework web basado en la API de Servlets compatible con diversos flujos de trabajo .

Spring admite dos lenguajes: Kotlin y Groovy . Kotlin permite a los desarrolladores escribir código conciso y elegante, y es interoperable con las bibliotecas Java existentes. Groovy ofrece una sintaxis concisa y una integración fluida con las aplicaciones Java existentes. Además, Spring admite el uso de clases y objetos definidos con un lenguaje dinámico compatible, lo que permite a los desarrolladores usar contenedores IoC de Spring para instanciar, configurar e inyectar dependencias de forma transparente en los objetos resultantes.

Otros beneficios y ventajas de Spring incluyen los siguientes:

- Se integra con numerosas tecnologías, incluidos clientes de transferencia de estado representacional , extensiones de administración de Java, Java Message Service y Java Virtual Machine Checkpoint Restore.
- El uso correcto de IoC simplifica tanto las pruebas unitarias como las pruebas de integración .
- Proporciona soporte integral para la gestión de transacciones con un modelo de programación consistente, una API más simple para la gestión programática de transacciones e integración con las abstracciones de acceso a datos de Spring.
- Con el lenguaje de expresión Spring, los gráficos de objetos se pueden consultar y manipular en tiempo de ejecución.
- El componente AOP de Spring, con aspectos que definen la modularidad, permite la modularización de preocupaciones en múltiples tipos y objetos.

## **Tecnologías y características importantes de Spring**

Spring Framework incluye numerosas tecnologías que lo diferencian de otros frameworks Java (y, en muchos casos, lo convierten en una mejor opción que ellos).

### **Inversión de control**

La inversión de control ( IoC ) es un principio de diseño en ingeniería de software que otorga control de ejecución al framework, en este caso, Spring. Al tomar el control del flujo del programa y realizar llamadas a código personalizado , Spring facilita la creación de programas modulares, la prueba de programas y la alternancia entre diferentes implementaciones según sea necesario.

### **Inyección de dependencia**

La inyección de dependencia, una forma especializada de IoC en Spring , es un patrón de diseño en el que los objetos definen sus dependencias de una de tres maneras:

1. Argumentos del constructor.
2. Argumentos para un método de fábrica.
3. Propiedades establecidas en la instancia del objeto después de que se construye o se devuelve desde un método de fábrica.

Un ensamblador conecta o inyecta objetos con otros objetos. La inyección de dependencias promueve un acoplamiento flexible en el código, por lo que es menos probable que los cambios en un área de la aplicación afecten a otra.

Con la inyección de dependencias, el código es más limpio. Además, el IOC y la inyección de dependencias facilitan la prueba y el mantenimiento del código .

## Contenedor de IoC

Esta es la parte de Spring Framework donde se crean, conectan, configuran y gestionan los objetos a lo largo de su ciclo de vida. Su objetivo principal es instanciar, configurar, ensamblar y gestionar beans de Spring. El contenedor lee los metadatos de configuración (que contienen instrucciones del desarrollador y están representadas en Lenguaje de Marcado Extensible, anotaciones de Java o código Java) para comprender qué objetos (beans) instanciar, configurar y ensamblar. Normalmente no se requiere código de usuario explícito para instanciar una o más instancias de un contenedor IoC de Spring.

## Frijoles

En Spring Framework, los beans son los objetos que conforman la columna vertebral de cualquier aplicación. Los beans son gestionados por el contenedor IoC de Spring y representados en él como objetos `BeanDefinition`. Todos los beans y sus dependencias se reflejan en los metadatos de configuración utilizados por un contenedor y proporcionados por el desarrollador de la aplicación. Cada bean tiene un identificador único, aunque puede tener otros identificadores, conocidos como *alias*.

## Spring Boot

- Subproyecto de Spring que permite crear aplicaciones listas para producción con configuración mínima.
- Incluye un servidor embebido (como Tomcat) y autoconfiguración para acelerar el desarrollo.

## Acceso a servicios web (REST y SOAP)

- Soporte para la creación y consumo de APIs RESTful con facilidad.
- También permite trabajar con servicios web SOAP si es necesario.

## Soporte para tareas y eventos

- Facilita la ejecución de tareas programadas y manejo de eventos de aplicación de manera asíncrona.

## Integración con otros frameworks y tecnologías

- Compatible con tecnologías modernas como Kafka, RabbitMQ, WebSockets, GraphQL, y servicios en la nube (AWS, Azure).