

# Marcos de Desarrollo Web

**Sesión 13**



**Universidad  
Tecnológica  
del Perú**

# ¿Tienen alguna consulta o duda sobre la clase previa?



# Logro de la sesión

Al finalizar la sesión, el estudiante integra la biblioteca Spring Security para validar el acceso en páginas y recursos privados o públicos en una aplicación web.

**¿Qué son las páginas privadas y públicas en un sitio web? ¿Cómo se permite el acceso a las privadas?**





# Spring Security

*¿Cuál es la importancia de implementar bibliotecas de seguridad en proyectos web?*



Universidad  
Tecnológica  
del Perú

# Contenido

## Introducción a Spring Security

- Definiciones
- Tipos de validación
- Configuración de Spring Security mediante en un proyecto web.





# Introducción a Spring Security

- Spring Security es un framework de seguridad integral para aplicaciones Java y Spring que proporciona autenticación y autorización.
- Diseñado para proteger aplicaciones basadas en Spring, soportando varias capas de seguridad para gestionar usuarios, roles y permisos.

# Características principales

- Proporciona **autenticación** y **autorización**.
- Protege contra ataques como Cross-Site Scripting (**XSS**) y Cross-Site Request Forgery (**CSRF**).
- Soporte integrado para **OAuth2**, **JWT**, y autenticación basada en **formularios** o **tokens**.
- Ofrece una estructura fácil de integrar y personalizar dentro de cualquier proyecto Spring Boot.



# Tipos de validación en Spring Security

## Validación básica (**Basic authentication**):

- Se usa el encabezado HTTP Authorization para enviar credenciales codificadas en base64.
- Es simple, pero no muy segura en aplicaciones públicas, porque las credenciales están en texto plano (aunque codificadas).



# Tipos de validación en Spring Security

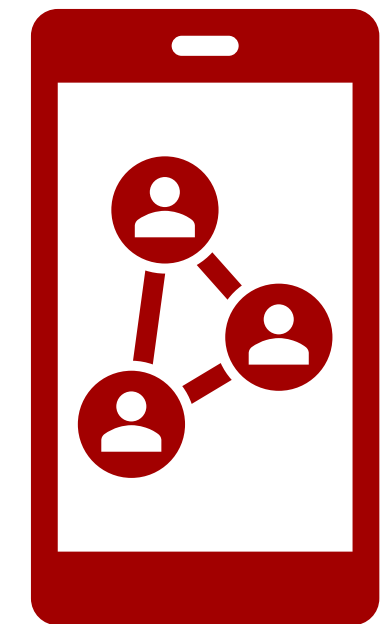
## Validación en formularios (**Form-based Authentication**)

- Implica un formulario de inicio de sesión para capturar las credenciales de usuario.
- Mayor personalización del flujo de autenticación y mejor experiencia de usuario.

# Tipos de validación en Spring Security

## OAuth y OpenID Connect

- Protocolo moderno que permite a los usuarios autenticarse usando proveedores de identidad externos como Google, Facebook, etc.
- Ideal para aplicaciones que permiten iniciar sesión mediante redes sociales.



# Arquitectura básica de Spring Security

- **Filtros de seguridad (Security Filters):** interceptan todas las peticiones HTTP antes de que lleguen al controlador.
- **Autenticación:** se verifica la identidad del usuario con un mecanismo que varía según el tipo de validación.
- **Autorización:** después de la autenticación, se verifica si el usuario tiene permisos para acceder a los recursos.
- **Manejo de sesiones:** Spring Security también ofrece mecanismos para controlar sesiones y proteger contra ataques de sesión como el Session Fixation.

# Configuración de Spring Security

Se puede configurar con:

- **Anotaciones:** Anotaciones como `@EnableWebSecurity`, `@Secured` y `@PreAuthorize` permiten habilitar seguridad y roles específicos.
- **Java Config:** En el archivo de configuración (`SecurityConfig.java`), se puede sobrescribir el método `configure()` para personalizar la seguridad.
- **application.properties:** Configuración básica de usuarios y contraseñas desde `application.properties` (ideal para entornos de desarrollo o pruebas).



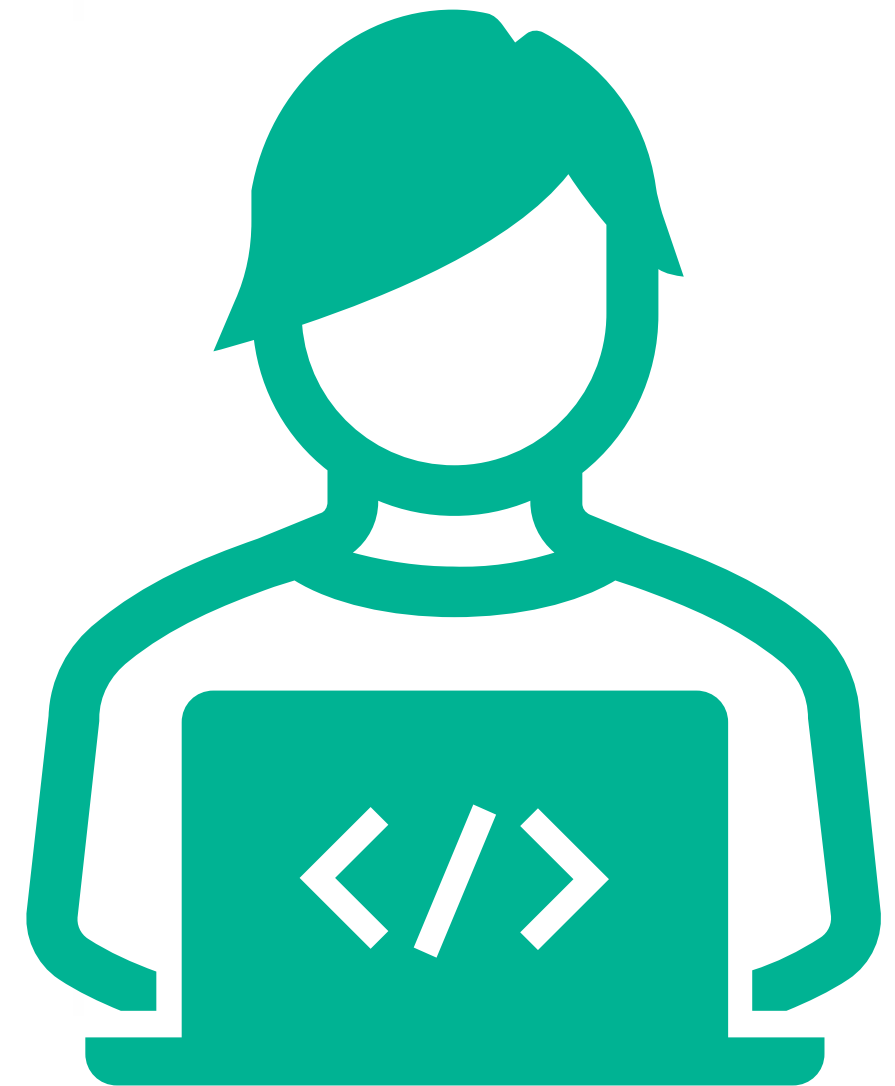
# Buenas prácticas de seguridad

- **No almacenar contraseñas en texto plano:** Usa un algoritmo de hashing seguro como BCrypt.
- **Uso de HTTPS:** Asegurar la conexión utilizando HTTPS.
- **Configurar límites de sesión:** Definir tiempo de expiración y gestionar la fijación de sesión.
- **Autenticación Multi-Factor (MFA):** Implementar mecanismos de autenticación de dos factores para una capa adicional de seguridad.



# Programando

- Construir un sitio web con páginas públicas y privadas empleando **validación básica y en formularios**.



# ¿Tienen alguna consulta o duda?



# Actividad



Resolver la actividad planteada en la plataforma.

# Cierre

## ¿Qué hemos aprendido hoy?



Elaboramos nuestras conclusiones sobre el tema tratado



**Universidad  
Tecnológica  
del Perú**