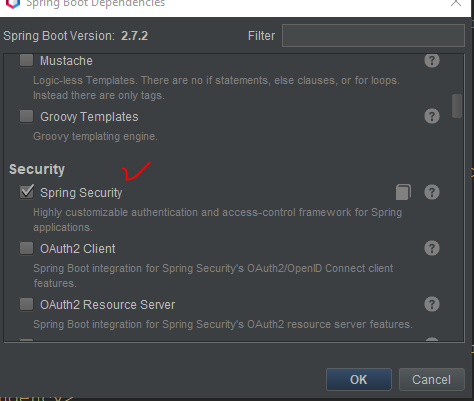
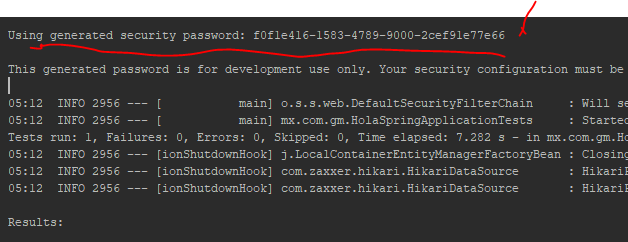
120. Seguridad con Spring

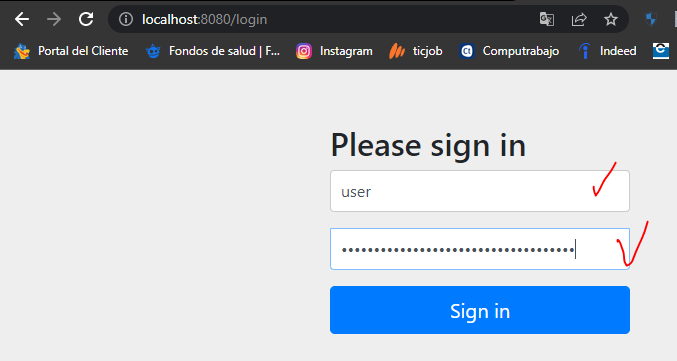
Tema importante en Spring el tema de seguridad a nuestra app web continuar proyecto clientes

1. Modificar POM
   1. Agregar dependencia de seguridad springboot
   2. Compilar para descargar librería me entragara un usuario y una contraseña de seguridad



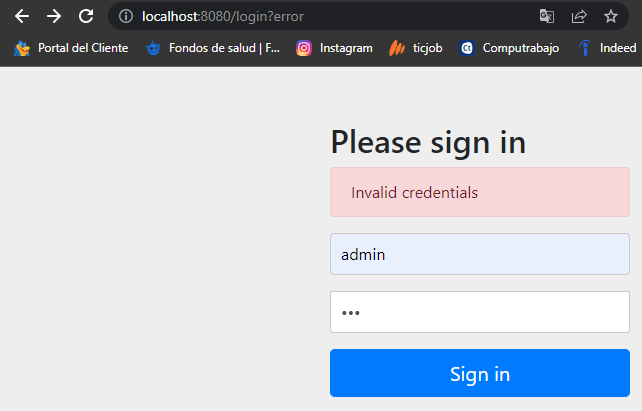
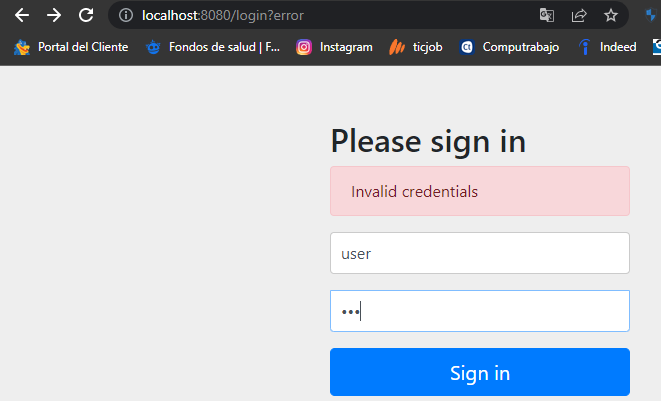
f0f1e416-1583-4789-9000-2cef91e77e66

1. Pruebas

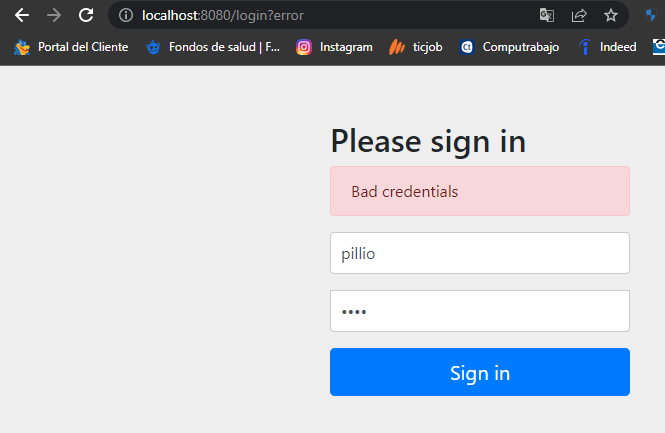
 

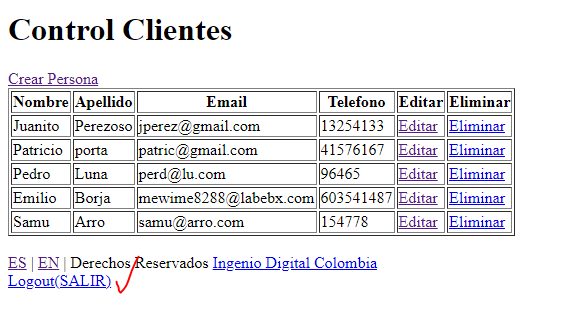
Establecer Dominios y Roles

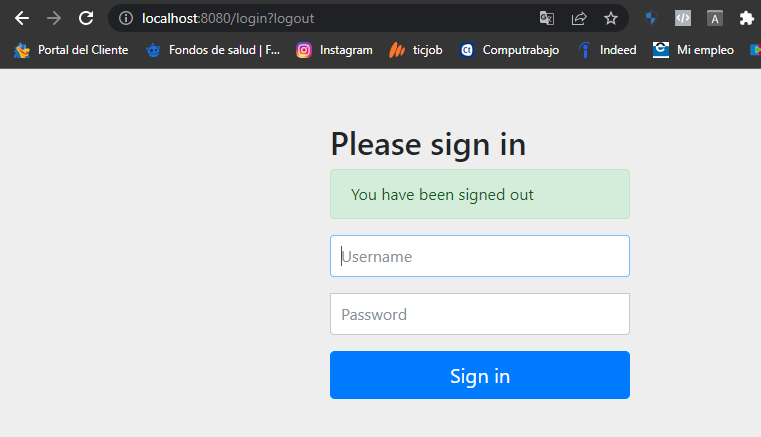
1. Crear clase SecurityConfig
   1. Agregar anotación @Config
   2. Agregar anotación @EnableWebSecurity para habilitar la configuración
   3. Configurar parámetro de entrada Http security para restringuir la url y usuarios que podrá usar según el roll asignado
   4. Mapeo de usuario
2. Agregar Logout(Salir) – Modificar plantilla
3. Pruebas



Credenciales invalidas





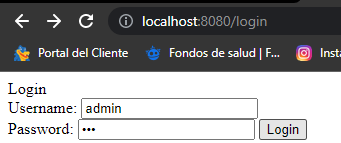


RESTRINGIR URL’s DE LA APLICACIÓN

Restringir las Url’s

1. Crear método SecurityConfig.AUTORIZACION en la clase security
2. Crear método en WebConfig.addViewControllers de la url Mapin por default
3. Agregar pagina de templates.login personalizada
4. Crear pagina de templates.errores.403 - mapear ruta – agregar en security config la configuración
5. Agrgar el método de inicio la variable para identificar el roll que ha hecho login @AutenticationPrincipal
6. Pruebas

Rol admin -🡪 Tiene derecho a todos los métodos en la tabla

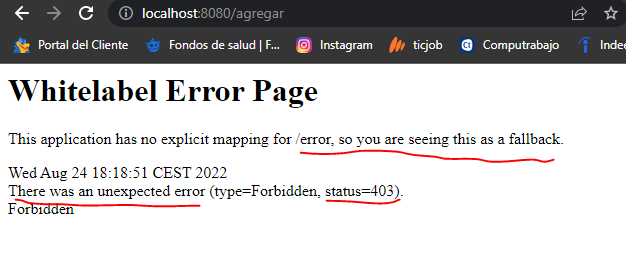




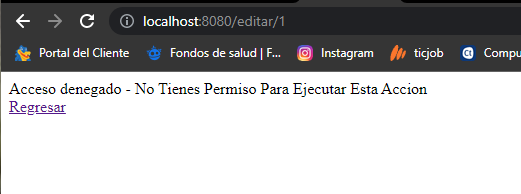
Roll user



No puede editar ni eliminar, de lo contrario saldrá el siguiente error sino se tiene una pagina mapeada para el error



Ya agregado el pto 4



Log.role



Seguridad con Spring y Thymeleaf

Tema de seguridad utilizando las etiquetas de tymyleaft

1. Agregar librería POM tymelleafth para seguridad
2. Agregar concepto de seguridad a la plantilla
   1. Agregar namespace de thymelead de seguridad
3. Modificar Index.
   1. Hacer cambios para que el usuario no tenga la opción de editar, eliminar ni crear
4. Pruebas

