

**Alumno:** Kevin Zamora Amela  
**Tarea:** PSP06  
**Asignatura:** Programación de Servicios y Procesos

**Actividad 6.1.** Crea una aplicación que realice los siguientes pasos:

- Solicita el nombre del usuario que va a utilizar la aplicación. El “login” tiene una longitud de 8 caracteres y está compuesto únicamente por letras minúsculas.
- Solicita al usuario el nombre de un fichero que quiere mostrar. El nombre del fichero es como máximo de 8 caracteres y tiene una extensión de 3 caracteres.
- Visualiza en pantalla el contenido del fichero.

Es importante tener en cuenta que se tiene que realizar una validación de los datos de entrada y llevar un registro de la actividad del programa.

**Respuesta:**

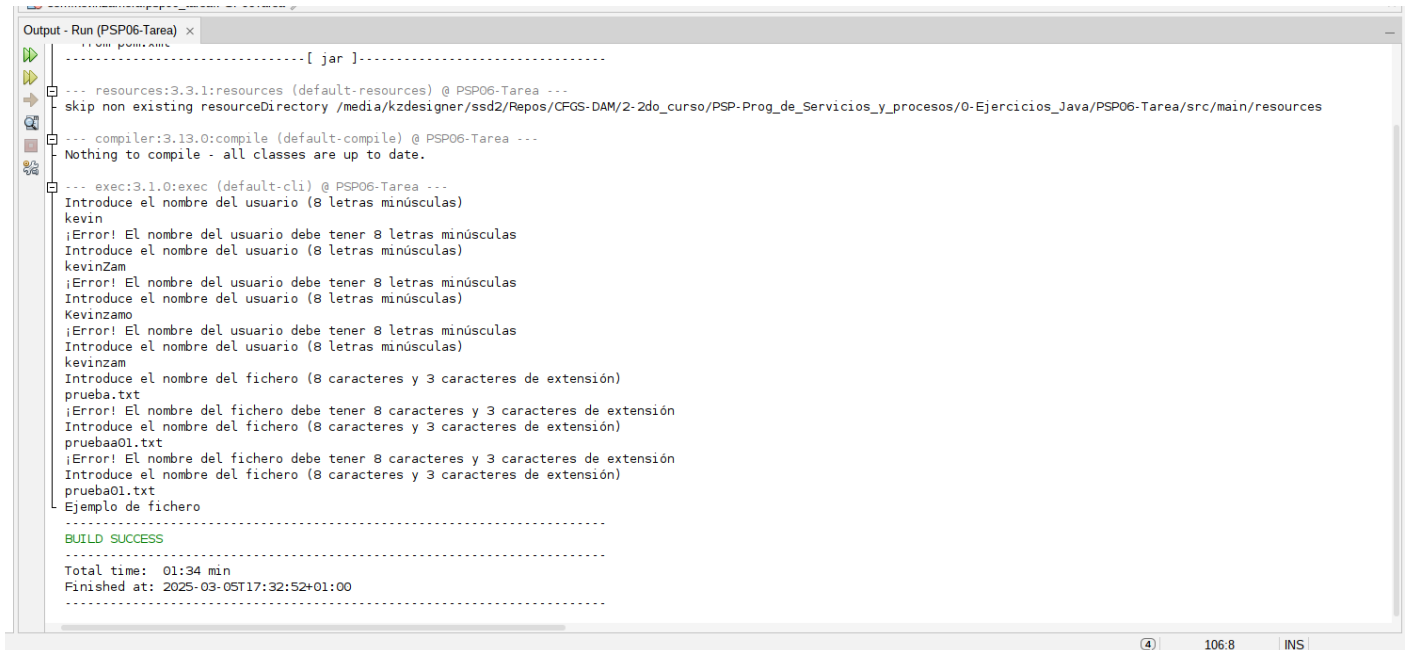
The screenshot displays the Apache NetBeans IDE interface. The main editor window shows the source code for `PSP06Tarea.java`. The code includes a package declaration, imports for `Scanner` and `System`, and a `main` method. The `main` method prompts the user for a username and a filename, validates the input (checking for 8 lowercase letters for the username and a valid 8-character filename with a 3-character extension), and then prints the content of the specified file. The `Output - Run (PSP06-Tarea)` window at the bottom shows the execution process, including building the project, compiling the source code, and the runtime output where the user enters 'kevinzam' and 'prueba01.txt', resulting in the file's content being displayed.

```
26
27
28 // mediante el terminal de comandos.
29
30 public static void main(String[] args) {
31
32     try {
33         Scanner scanner = new Scanner(System.in);
34         scanner.useDelimiter("\n");
35
36         String nombreUsuario;
37         do {
38             System.out.println("Introduce el nombre del usuario (8 letras minúsculas*");
39             nombreUsuario = scanner.next().trim();
40             if (!validarNombreUsuario(nombreUsuario)) {
41                 System.out.println("Error! El nombre del usuario debe tener 8 letras minúsculas*");
42                 escribirLog("Error! El nombre del usuario debe tener un formato válido: " + nombreUsuario);
43             }
44         } while (!validarNombreUsuario(nombreUsuario));
45
46         // Validar nombre de fichero
47         String nombreFichero;
48         do {
49             System.out.println("Introduce el nombre del fichero (8 caracteres y 3 caracteres de extensión)");
50             nombreFichero = scanner.next().trim();
51             if (!validarNombreFichero(nombreFichero)) {
52                 System.out.println("Error! El nombre del fichero debe tener un formato válido: " + nombreFichero);
53                 escribirLog("Error! El nombre del fichero debe tener un formato válido: " + nombreFichero);
54             }
55         } while (!validarNombreFichero(nombreFichero));
56
57         // Leer el fichero
58         try {
59             BufferedReader reader = new BufferedReader(new FileReader(nombreFichero));
60             String line;
61             while ((line = reader.readLine()) != null) {
62                 System.out.println(line);
63             }
64             reader.close();
65         } catch (IOException e) {
66             System.out.println("Error al leer el fichero: " + e.getMessage());
67             escribirLog("Error al leer el fichero: " + e.getMessage());
68         }
69     } catch (Exception e) {
70         System.out.println("Error: " + e.getMessage());
71         escribirLog("Error: " + e.getMessage());
72     }
73 }
```

Output - Run (PSP06-Tarea)

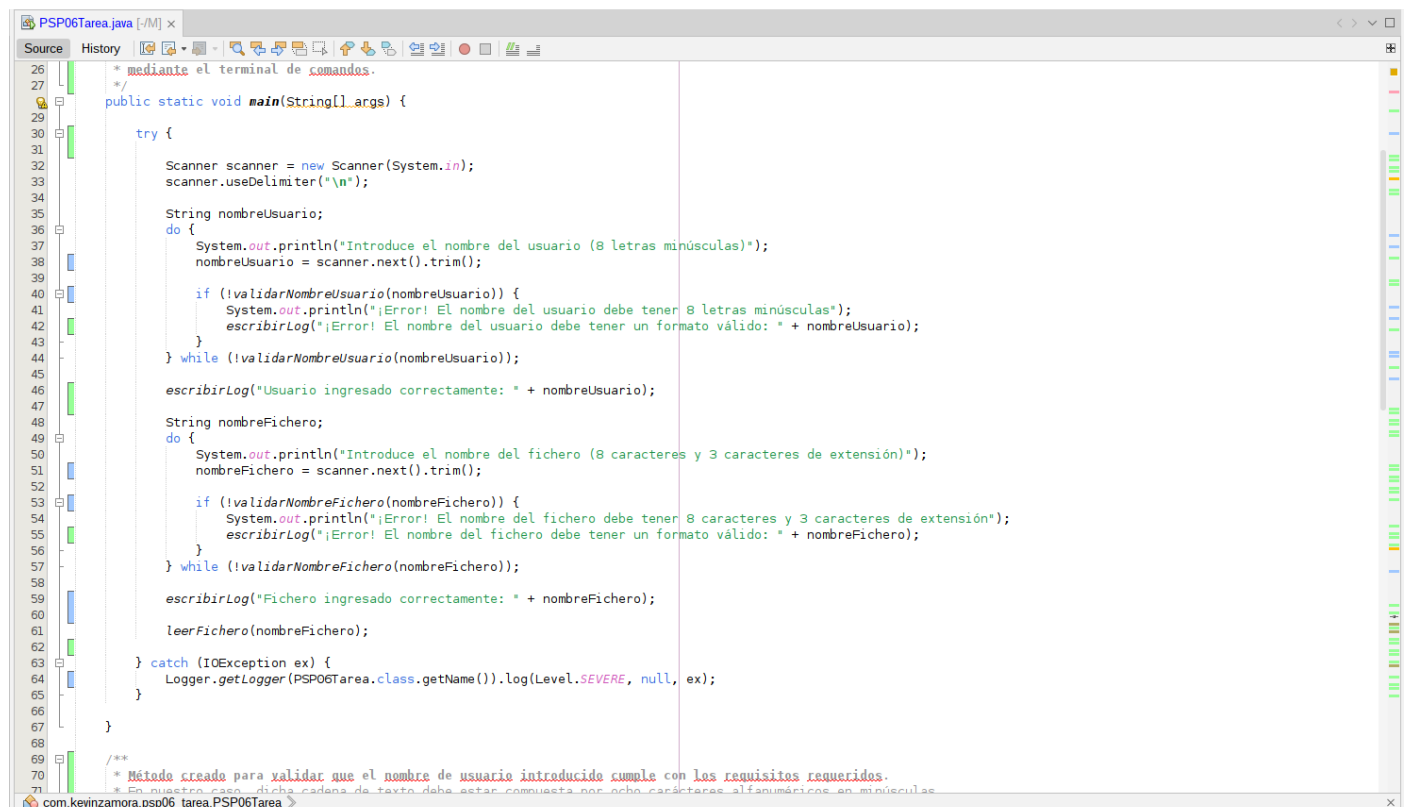
```
-----< com.kevinzamora:PSP06-Tarea >-----
Building PSP06-Tarea 1.0-SNAPSHOT
from pom.xml
[ jar ]
resources:3.3.1:resources (default-resources) @ PSP06-Tarea ---
skip non existing resourceDirectory /media/kzdesigner/ssd2/Repos/CFOS-DAM/2-2do_curso/PSP-Prog_de_Servicios_y_procesos/0-Ejercicios_Java/PSP06-Tarea/src/main/resources
compiler:3.13.0:compile (default-compile) @ PSP06-Tarea ---
Recompiling the module because of changed source code.
Compiling 1 source file with javac [debug release 21] to target/classes
exec:3.1.0:exec (default-cli) @ PSP06-Tarea ---
Introduce el nombre del usuario (8 letras minúsculas)
kevinzam
Introduce el nombre del fichero (8 caracteres y 3 caracteres de extensión)
prueba01.txt
Ejemplo de fichero
-----
BUILD SUCCESS
-----
Total time: 41.127 s
Finished at: 2025-03-05T17:29:54+01:00
-----
```

## Validación de los datos aplicando las restricciones requeridas:



```
Output - Run (PSP06-Tarea) x
[ jar ]-----
--- resources:3.3.1:resources (default-resources) @ PSP06-Tarea ---
skip non existing resourceDirectory /media/kzdesigner/ssd2/Repos/CFGs-DAM/2-2do_curso/PSP-Prog_de_Servicios_y_procesos/0-Ejercicios_Java/PSP06-Tarea/src/main/resources
--- compiler:3.13.0:compile (default-compile) @ PSP06-Tarea ---
Nothing to compile - all classes are up to date.
--- exec:3.1.0:exec (default-cli) @ PSP06-Tarea ---
Introduce el nombre del usuario (8 letras minúsculas)
kevin
¡Error! El nombre del usuario debe tener 8 letras minúsculas
Introduce el nombre del usuario (8 letras minúsculas)
kevinZam
¡Error! El nombre del usuario debe tener 8 letras minúsculas
Introduce el nombre del usuario (8 letras minúsculas)
Kevinzamo
¡Error! El nombre del usuario debe tener 8 letras minúsculas
Introduce el nombre del usuario (8 letras minúsculas)
kevinzam
Introduce el nombre del fichero (8 caracteres y 3 caracteres de extensión)
prueba.txt
¡Error! El nombre del fichero debe tener 8 caracteres y 3 caracteres de extensión
Introduce el nombre del fichero (8 caracteres y 3 caracteres de extensión)
pruebaa01.txt
¡Error! El nombre del fichero debe tener 8 caracteres y 3 caracteres de extensión
Introduce el nombre del fichero (8 caracteres y 3 caracteres de extensión)
prueba01.txt
Ejemplo de fichero
BUILD SUCCESS
Total time: 01:34 min
Finished at: 2025-03-05T17:32:52+01:00
```

## Programa/Aplicación implementado/a:



```
PSP06Tarea.java [-M] x
Source History
26  * mediante el terminal de comandos.
27  */
28  public static void main(String[] args) {
29
30      try {
31
32          Scanner scanner = new Scanner(System.in);
33          scanner.useDelimiter("\n");
34
35          String nombreUsuario;
36          do {
37              System.out.println("Introduce el nombre del usuario (8 letras minúsculas)*");
38              nombreUsuario = scanner.next().trim();
39
40              if (!validarNombreUsuario(nombreUsuario)) {
41                  System.out.println("¡Error! El nombre del usuario debe tener 8 letras minúsculas*");
42                  escribirLog("¡Error! El nombre del usuario debe tener un formato válido: " + nombreUsuario);
43              }
44          } while (!validarNombreUsuario(nombreUsuario));
45
46          escribirLog("Usuario ingresado correctamente: " + nombreUsuario);
47
48          String nombreFichero;
49          do {
50              System.out.println("Introduce el nombre del fichero (8 caracteres y 3 caracteres de extensión)*");
51              nombreFichero = scanner.next().trim();
52
53              if (!validarNombreFichero(nombreFichero)) {
54                  System.out.println("¡Error! El nombre del fichero debe tener 8 caracteres y 3 caracteres de extensión*");
55                  escribirLog("¡Error! El nombre del fichero debe tener un formato válido: " + nombreFichero);
56              }
57          } while (!validarNombreFichero(nombreFichero));
58
59          escribirLog("Fichero ingresado correctamente: " + nombreFichero);
60          leerFichero(nombreFichero);
61
62      } catch (IOException ex) {
63          Logger.getLogger(PSP06Tarea.class.getName()).log(Level.SEVERE, null, ex);
64      }
65
66  }
67
68  /**
69   * Método creado para validar que el nombre de usuario introducido cumple con los requisitos requeridos.
70   * En nuestro caso, dicha cadena de texto debe estar compuesta por ocho caracteres alfanuméricos en minúsculas
71   */
```

## Métodos implementados:

```
PSP06Tarea.java [-/M] x
Source History
68
69
70 /**
71  * Método creado para validar que el nombre de usuario introducido cumple con los requisitos requeridos.
72  * En nuestro caso, dicha cadena de texto debe estar compuesta por ocho caracteres alfanuméricos en minúsculas.
73  * @param nombreUsuario
74  * @return nombreUsuario.matches("[a-z]{8}");
75  */
76 public static boolean validarNombreUsuario(String nombreUsuario) {
77     return nombreUsuario.matches("[a-z]{8}");
78 }
79
80 /**
81  * Método creado para validar que el nombre de fichero introducido cumple con los requisitos requeridos.
82  * En nuestro caso, dicha cadena de texto debe estar compuesta por ocho caracteres alfanuméricos en minúsculas,
83  * un punto y tres caracteres alfanuméricos más, representando a la extensión del archivo.
84  * @param nombreFichero
85  * @return nombreFichero.matches("[a-zA-Z0-9]{8}\\.[a-zA-Z0-9]{3}");
86  */
87 public static boolean validarNombreFichero(String nombreFichero) {
88     return nombreFichero.matches("[a-zA-Z0-9]{8}\\.[a-zA-Z0-9]{3}");
89 }
90
91 /**
92  * Método creado para leer el fichero con el nombre introducido mediante consola de comandos, si este existe.
93  * @param nombreFichero
94  */
95 public static void LeerFichero(String nombreFichero) throws FileNotFoundException, IOException {
96     BufferedReader br = new BufferedReader(new FileReader(nombreFichero));
97
98     String linea;
99     while ((linea = br.readLine()) != null) {
100         System.out.println(linea);
101     }
102 }
103
104 /**
105  * Método creado para escribir cada uno de los eventos ocurridos durante la ejecución del programa
106  * en un log/registro.
107  * @param mensaje
108  */
109 private static void escribirLog(String mensaje) throws IOException {
110
111     try (PrintWriter pw = new PrintWriter(new FileWriter("registro.log", true))) {
112         DateTimeFormatter formato = DateTimeFormatter.ofPattern("yyyy-MM-dd HH:mm:ss");
113         String formatoFecha = LocalDateTime.now().format(formato);
114
115         pw.println("(" + formatoFecha + ") " + mensaje);
116     }
117 }
118
119
120 }
```

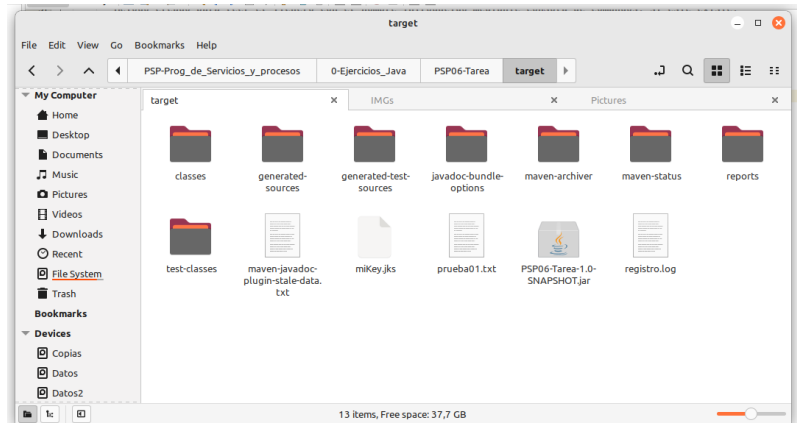
**Anotación:** Como se puede apreciar en la interfaz de nuestro IDE Netbeans, hemos aplicado la metodología de control de versiones mediante la herramienta Github.

**Actividad 6.2.** Utilizando la aplicación desarrollada en la actividad anterior, configura las políticas de acceso para:

- Firmar digitalmente la aplicación.
- Que sólo pueda leer los datos del directorio c:/datos.

1. Primeramente, generamos el archivo ejecutable con extensión .jar, seguimos los pasos siguientes:

- Seleccionamos el directorio raíz de nuestro proyecto y hacemos clic derecho.
- Seleccionamos la opción “Clean and Build” y dejamos que se ejecute y el presente proceso siga su curso.



Seguidamente, generamos la firma con la que proceder a firmar nuestra aplicación, utilizamos el siguiente comando mediante la terminal:

```
keytool -genkeypair -alias miCertificado -keyalg RSA -keysize 2048 -keystore miKey.jks -validity 365
```

```
kzdesigner@kzdesigner-PC: /media/kzdesigner/ssd2/Repos/CFGS-DAM/2-2do_curso/PSP-Prog_de_Servicios_y_procesos/0-Ejercicios_Java/...  
File Edit View Search Terminal Help  
kzdesigner@kzdesigner-PC:/media/kzdesigner/ssd2/Repos/CFGS-DAM/2-2do_curso/PSP-Prog_de_Servicios_y_procesos/0-Ejercicios_Java/PSP06-Tarea/target$ keytool -genkeypair -alias miCertificado -keyalg RSA -keysize 2048 -keystore miKey.jks -validity 365  
Enter keystore password:  
Re-enter new password:  
Enter the distinguished name. Provide a single dot (.) to leave a sub-component empty or press ENTER to use the default value in braces.  
What is your first and last name?  
[Unknown]: Kevin Zamora Amela  
What is the name of your organizational unit?  
[Unknown]: ONCE  
What is the name of your organization?  
[Unknown]: ONCE  
What is the name of your City or Locality?  
[Unknown]: Vinaròs  
What is the name of your State or Province?  
[Unknown]: Castellón  
What is the two-letter country code for this unit?  
[Unknown]: ES  
Is CN=Kevin Zamora Amela, OU=ONCE, O=ONCE, L=Vinaròs, ST=Castellón, C=ES correct?  
[no]: yes  
  
Generating 2,048 bit RSA key pair and self-signed certificate (SHA384withRSA) with a validity of 365 days  
for: CN=Kevin Zamora Amela, OU=ONCE, O=ONCE, L=Vinaròs, ST=Castellón, C=ES  
kzdesigner@kzdesigner-PC:/media/kzdesigner/ssd2/Repos/CFGS-DAM/2-2do_curso/PSP-Prog_de_Servicios_y_procesos/0-Ejercicios_Java/PSP06-Tarea/target$
```

Y mediante la ejecución del siguiente comando, procedemos a firmar nuestra aplicación:

```
jarsigner -keystore miKey.jks PSP06-Tarea-1.0-SNAPSHOT.jar miCertificado
```

```
kzdesigner@kzdesigner-PC:/media/kzdesigner/ssd2/Repos/CFGS-DAM/2-2do_curso/PSP-Prog_de_Servicios_y_procesos/0-Ejercicios_Java/PSP06-Tarea/target$ jarsigner -keystore miKey.jks PSP06-Tarea-1.0-SNAPSHOT.jar miCertificado  
Enter Passphrase for keystore:  
jar signed.  
  
Warning:  
The signer's certificate is self-signed.  
POSIX file permission and/or symlink attributes detected. These attributes are ignored when signing and are not protected by the signature.
```

A continuación, verificamos que nuestro ejecutable se ha firmado correctamente:

```
jarsigner -keystore miKey.jks PSP06-Tarea-1.0-SNAPSHOT.jar
```

```
kzdesigner@kzdesigner-PC: /media/kzdesigner/ssd2/Repos/CFG5-DAR/2-2do_curso/PSP-Prog_de_Servicios_y_procesos/0-Ejercicios_Java/PSP06-Tarea/target$ jarsigner -verify -verbose -certs PSP06-Tarea-1.0-SNAPSHOT.jar
s      569 Wed Mar 05 17:51:06 CET 2025 META-INF/MANIFEST.MF

>>> Signer
X.509, CN=Kevin Zamora Amela, OU=ONCE, O=ONCE, L=Vinaros, ST=Castellón, C=ES
Signature algorithm: SHA384withRSA, 2048-bit key
[certificate is valid from 3/5/25, 5:47 PM to 3/5/26, 5:47 PM]
[Invalid certificate chain: PKIX path building failed: sun.security.provider.certpath.SunCertPathBuilderException: unable to find valid certification path to requested target]

605 Wed Mar 05 17:51:06 CET 2025 META-INF/MICERTIF.SF
1563 Wed Mar 05 17:51:06 CET 2025 META-INF/MICERTIF.RSA
0 Wed Mar 05 17:05:28 CET 2025 META-INF/
0 Wed Mar 05 17:05:26 CET 2025 com/
0 Wed Mar 05 17:05:26 CET 2025 com/kevinzamora/
0 Wed Mar 05 17:05:26 CET 2025 com/kevinzamora/psp06_tarea/
0 Wed Mar 05 17:05:28 CET 2025 META-INF/maven/
0 Wed Mar 05 17:05:28 CET 2025 META-INF/maven/com.kevinzamora/
sm      4251 Wed Mar 05 17:05:26 CET 2025 META-INF/maven/com.kevinzamora/PSP06-Tarea/
sm      4251 Wed Mar 05 17:05:26 CET 2025 com/kevinzamora/psp06_tarea/PSP06Tarea.class

>>> Signer
X.509, CN=Kevin Zamora Amela, OU=ONCE, O=ONCE, L=Vinaros, ST=Castellón, C=ES
Signature algorithm: SHA384withRSA, 2048-bit key
[certificate is valid from 3/5/25, 5:47 PM to 3/5/26, 5:47 PM]
[Invalid certificate chain: PKIX path building failed: sun.security.provider.certpath.SunCertPathBuilderException: unable to find valid certification path to requested target]

sm      690 Wed Mar 05 12:59:04 CET 2025 META-INF/maven/com.kevinzamora/PSP06-Tarea/pom.xml

>>> Signer
X.509, CN=Kevin Zamora Amela, OU=ONCE, O=ONCE, L=Vinaros, ST=Castellón, C=ES
Signature algorithm: SHA384withRSA, 2048-bit key
[certificate is valid from 3/5/25, 5:47 PM to 3/5/26, 5:47 PM]
[Invalid certificate chain: PKIX path building failed: sun.security.provider.certpath.SunCertPathBuilderException: unable to find valid certification path to requested target]

sm      68 Wed Mar 05 17:05:28 CET 2025 META-INF/maven/com.kevinzamora/PSP06-Tarea/pom.properties

>>> Signer
X.509, CN=Kevin Zamora Amela, OU=ONCE, O=ONCE, L=Vinaros, ST=Castellón, C=ES
Signature algorithm: SHA384withRSA, 2048-bit key
[certificate is valid from 3/5/25, 5:47 PM to 3/5/26, 5:47 PM]
[Invalid certificate chain: PKIX path building failed: sun.security.provider.certpath.SunCertPathBuilderException: unable to find valid certification path to requested target]

s = signature was verified
m = entry is listed in manifest
k = at least one certificate was found in keystore

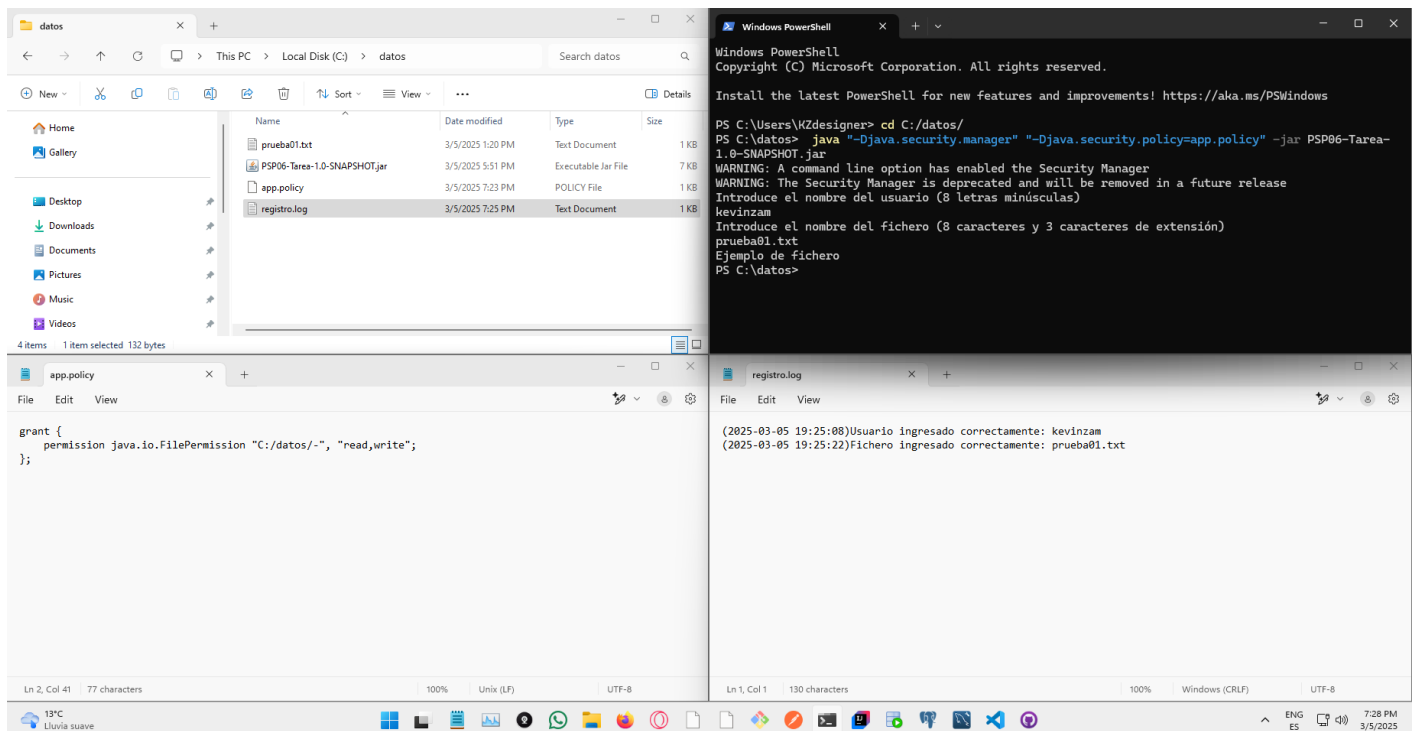
- Signed by "CN=Kevin Zamora Amela, OU=ONCE, O=ONCE, L=Vinaros, ST=Castellón, C=ES"
  Digest algorithm: SHA-384
  Signature algorithm: SHA384withRSA, 2048-bit key

jar verified.

Warning:
This jar contains entries whose certificate chain is invalid. Reason: PKIX path building failed: sun.security.provider.certpath.SunCertPathBuilderException: unable to find valid certification path to requested target
This jar contains entries whose signer certificate is self-signed.
This jar contains signatures that do not include a timestamp. Without a timestamp, users may not be able to validate this jar after any of the signer certificates expire (as early as 2026-03-05).
POSIX file permission and/or symlink attributes detected. These attributes are ignored when signing and are not protected by the signature.
```

2. Por otro lado, limitamos los permisos de acceso de nuestra aplicación, para que solamente pueda leer y escribir si se ejecuta desde un directorio llamado “datos”, el cual se encuentra alojado en el directorio raíz del disco duro principal (C:/) de nuestro sistema operativo Windows.

## Resultado:





## Errores previos solucionados:

```
kzdesigner@kzdesigner-PC: /media/kzdesigner/ssd2/Repos/CFG5-DAM/2-2do_curso/PSP-Prog_de_Servicios_y_procesos/0-Ejercicios_Java/PSP0...  
File Edit View Search Terminal Help  
kzdesigner@kzdesigner-PC: /media/kzdesigner/ssd2/Repos/CFG5-DAM/2-2do_curso/PSP-Prorog_rog_de_Servicrog_rorog_drog_derorroror  
rog_de_Servicios_y_procesos/0-Ejercicios_Java/PSP06-Tarea$ java -Djava.security.manager -Djava.security.policy=app.policy -j  
ar PSP06-Tarea-1.0-SNAPSHOT.jar  
WARNING: A command line option has enabled the Security Manager  
WARNING: The Security Manager is deprecated and will be removed in a future release  
java.security.policy: error parsing file:/media/kzdesigner/ssd2/Repos/CFG5-DAM/2-2do_curso/PSP-Prog_de_Servicios_y_procesos/  
0-Ejercicios_Java/PSP06-Tarea/app.policy:  
    expected [;], read [end of file]  
Introduce el nombre del usuario (8 letras minúsculas)  
kevinzam  
Exception in thread "main" java.security.AccessControlException: access denied ("java.io.FilePermission" "registro.log" "wri  
te")  
    at java.base/java.security.AccessControlContext.checkPermission(AccessControlContext.java:488)  
    at java.base/java.security.AccessController.checkPermission(AccessController.java:1071)  
    at java.base/java.lang.SecurityManager.checkPermission(SecurityManager.java:411)  
    at java.base/java.lang.SecurityManager.checkWrite(SecurityManager.java:833)  
    at java.base/java.io.FileOutputStream.<init>(FileOutputStream.java:218)  
    at java.base/java.io.FileOutputStream.<init>(FileOutputStream.java:150)  
    at java.base/java.io.FileWriter.<init>(FileWriter.java:84)  
    at com.kevinzamora.psp06_tarea.PSP06Tarea.escribirLog(PSP06Tarea.java:85)  
    at com.kevinzamora.psp06_tarea.PSP06Tarea.main(PSP06Tarea.java:42)
```

```
Windows PowerShell  
Copyright (C) Microsoft Corporation. All rights reserved.  
  
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows  
  
PS C:\Users\KZdesigner> cd C:\datos  
PS C:\datos> java "-Djava.security.manager" "-Djava.security.policy=app.policy" -jar PSP06-Tarea-1.0-SNAPSHOT.jar  
WARNING: A command line option has enabled the Security Manager  
WARNING: The Security Manager is deprecated and will be removed in a future release  
java.security.policy: error parsing file:/C:/datos/app.policy:  
    expected [;], read [end of file]  
Introduce el nombre del usuario (8 letras minúsculas)  
kevinzam  
Exception in thread "main" java.security.AccessControlException: access denied ("java.io.FilePermission" "registro.log"  
"write")  
    at java.base/java.security.AccessControlContext.checkPermission(AccessControlContext.java:488)  
    at java.base/java.security.AccessController.checkPermission(AccessController.java:1071)  
    at java.base/java.lang.SecurityManager.checkPermission(SecurityManager.java:411)  
    at java.base/java.lang.SecurityManager.checkWrite(SecurityManager.java:833)  
    at java.base/java.io.FileOutputStream.<init>(FileOutputStream.java:218)  
    at java.base/java.io.FileOutputStream.<init>(FileOutputStream.java:150)  
    at java.base/java.io.FileWriter.<init>(FileWriter.java:84)  
    at com.kevinzamora.psp06_tarea.PSP06Tarea.escribirLog(PSP06Tarea.java:85)  
    at com.kevinzamora.psp06_tarea.PSP06Tarea.main(PSP06Tarea.java:42)
```

```
Windows PowerShell  
Introduce el nombre del usuario (8 letras minúsculas)  
kevinzam  
Exception in thread "main" java.security.AccessControlException: access denied ("java.io.FilePermission" "registro.log"  
"write")  
    at java.base/java.security.AccessControlContext.checkPermission(AccessControlContext.java:488)  
    at java.base/java.security.AccessController.checkPermission(AccessController.java:1071)  
    at java.base/java.lang.SecurityManager.checkPermission(SecurityManager.java:411)  
    at java.base/java.lang.SecurityManager.checkWrite(SecurityManager.java:833)  
    at java.base/java.io.FileOutputStream.<init>(FileOutputStream.java:218)  
    at java.base/java.io.FileOutputStream.<init>(FileOutputStream.java:150)  
    at java.base/java.io.FileWriter.<init>(FileWriter.java:84)  
    at com.kevinzamora.psp06_tarea.PSP06Tarea.escribirLog(PSP06Tarea.java:85)  
    at com.kevinzamora.psp06_tarea.PSP06Tarea.main(PSP06Tarea.java:42)  
PS C:\datos> java "-Djava.security.manager" "-Djava.security.policy=app.policy" -jar PSP06-Tarea-1.0-SNAPSHOT.jar  
WARNING: A command line option has enabled the Security Manager  
WARNING: The Security Manager is deprecated and will be removed in a future release  
Introduce el nombre del usuario (8 letras minúsculas)  
kevinzam  
Exception in thread "main" java.security.AccessControlException: access denied ("java.io.FilePermission" "registro.log"  
"write")  
    at java.base/java.security.AccessControlContext.checkPermission(AccessControlContext.java:488)  
    at java.base/java.security.AccessController.checkPermission(AccessController.java:1071)  
    at java.base/java.lang.SecurityManager.checkPermission(SecurityManager.java:411)  
    at java.base/java.lang.SecurityManager.checkWrite(SecurityManager.java:833)  
    at java.base/java.io.FileOutputStream.<init>(FileOutputStream.java:218)  
    at java.base/java.io.FileOutputStream.<init>(FileOutputStream.java:150)  
    at java.base/java.io.FileWriter.<init>(FileWriter.java:84)  
    at com.kevinzamora.psp06_tarea.PSP06Tarea.escribirLog(PSP06Tarea.java:85)  
    at com.kevinzamora.psp06_tarea.PSP06Tarea.main(PSP06Tarea.java:42)  
PS C:\datos>
```

**- Comentarios relacionados con el proceso de generación de la documentación con JavaDoc de ambos proyectos/ejercicios:**

- Primeramente y como ya deberíamos saber, para ejecutar la herramienta JavaDoc y proceder a generar la documentación de nuestro proyecto, debemos hacer clic derecho sobre el directorio raíz de nuestro proyecto con java y seleccionar la opción “Generar JavaDoc/Generate JavaDoc”, según tengamos en Español o en Inglés, respectivamente, la interfaz de nuestro IDE Netbeans.
- Al ejecutar la citada opción “Generar JavaDoc”, nos aparecerá/apareció un mensaje de error comentándonos que nuestra versión es demasiado antigua (siendo la 5 y en consecuencia, siendo anterior a la 8) y por ende, la herramienta de JavaDoc no se encuentra disponible ni resulta compatible.
- Para solucionarlo, simplemente hemos creado un nuevo proyecto de Java 21 (como versión instalada en nuestro sistema) con Maven y hemos “migrado”(o copiado y pegado) las clases de nuestros programas, a dos proyectos de Java nuevos e independientes. Al ejecutar la opción de “Generar JavaDoc”, ahora ya se nos generará/generó correctamente la documentación de nuestros proyectos/ejercicios.