Maven en Eclipse

### 2022

## http://hdwallpapers.lt/wallpaper/abstract_red_swirls_white_backdrop_3d_and_hd-wallpaper-205370.jpgProyecto Maven desde IDE Eclipse

#### Autor: RAFAEL RAMÍREZ MEDINA

**Índice**

Contenido

[Introducción 3](#_Toc120370421)

[Instalación de Maven en Eclipse 3](#_Toc120370422)

[Creación del proyecto 4](#_Toc120370423)

[Compilación y Ejecución del proyecto 8](#_Toc120370424)

[Gestión de Dependencia 13](#_Toc120370425)

[Creación de fichero java 14](#_Toc120370426)

[Preparación de los archivos para la ejecución desde consola 15](#_Toc120370427)

[Proyecto HelloWorldPDF 19](#_Toc120370428)

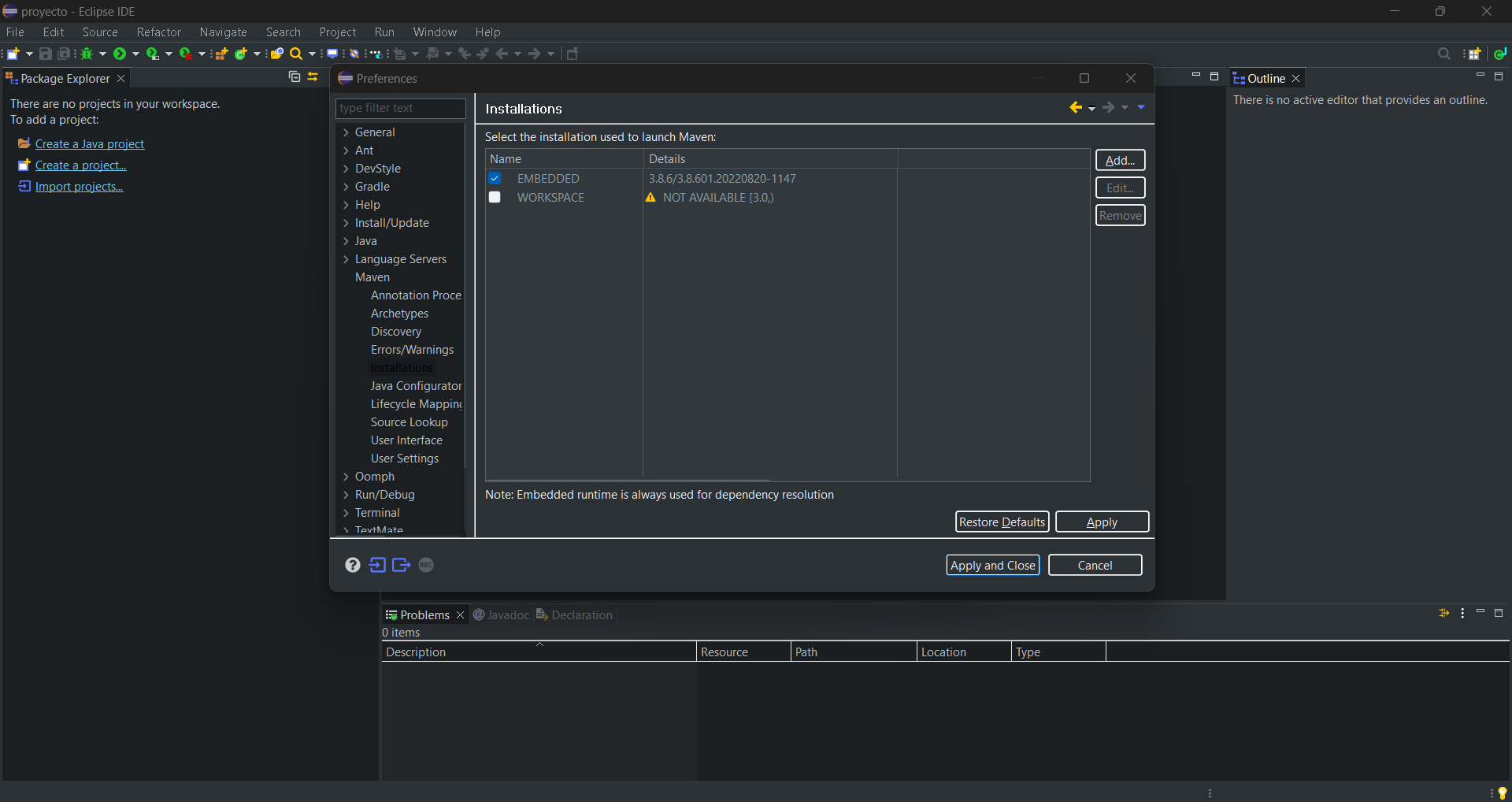
# Introducción

En esta aplicación vamos a realizar la instalación, creación, configuración y ejecución de un proyecto Maven directamente desde eclipse. El proyecto en cuestión va a tratar sobre la sucesión de Fibonacci.

# Instalación de Maven en Eclipse

Para la realización de la práctica vamos a usar Eclipse versión 2022-09(4.25.0). Esta versión de Eclipse ya trae integrado una versión de Maven instalada.

Vamos a revisar la versión de Maven instalada en Eclipse. Para ello vamos a la pestaña “Windows -> Preferences” y se abrirá la siguiente ventana, en la cual entraremos en Maven -> Installations:

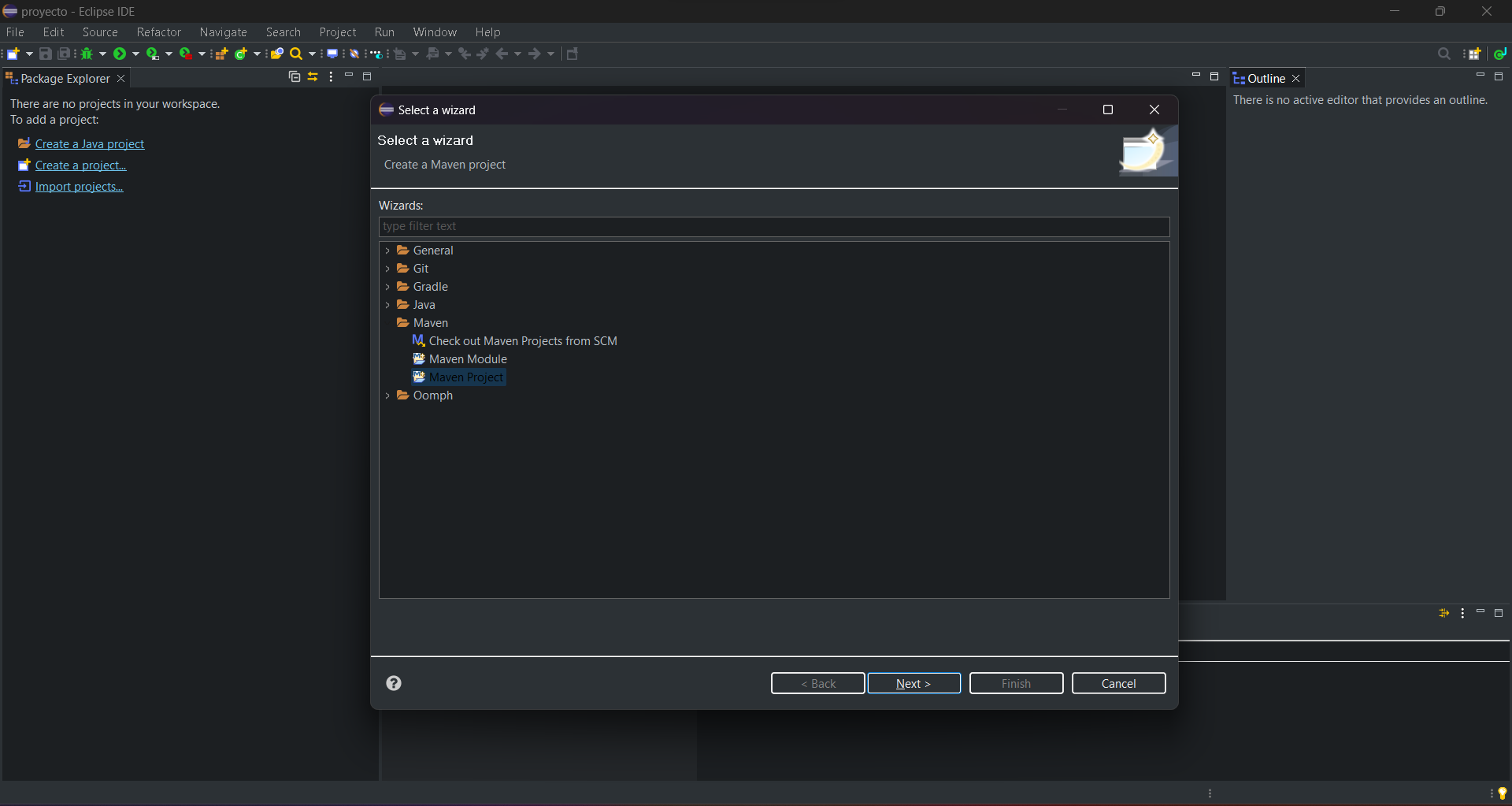


La versión de Maven instalada es la 3.8.6 la cual podemos verificar que es la misma que se encuentra disponible en la web de Maven para su descarga por tanto no deberemos realizar instalación alguna en este caso.

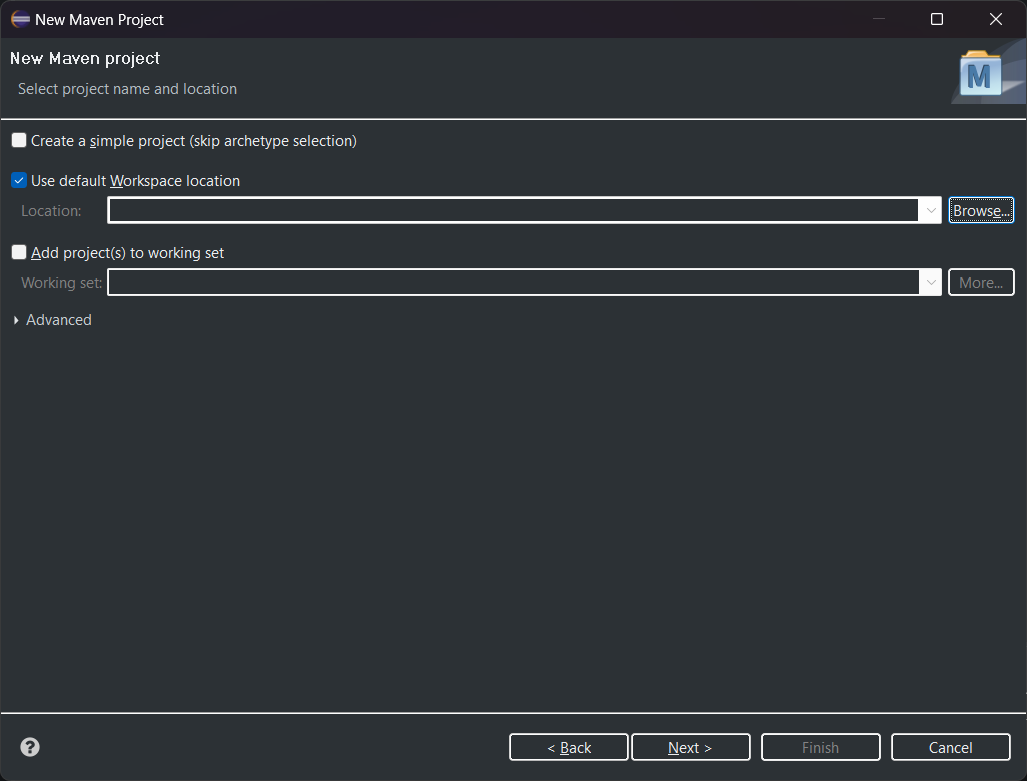
Si deseáramos usar una versión de Maven específica solo debemos descargarla y definir la ruta donde se encuentra instalado en el equipo pulsando en el botón “Add”.

# Creación del proyecto

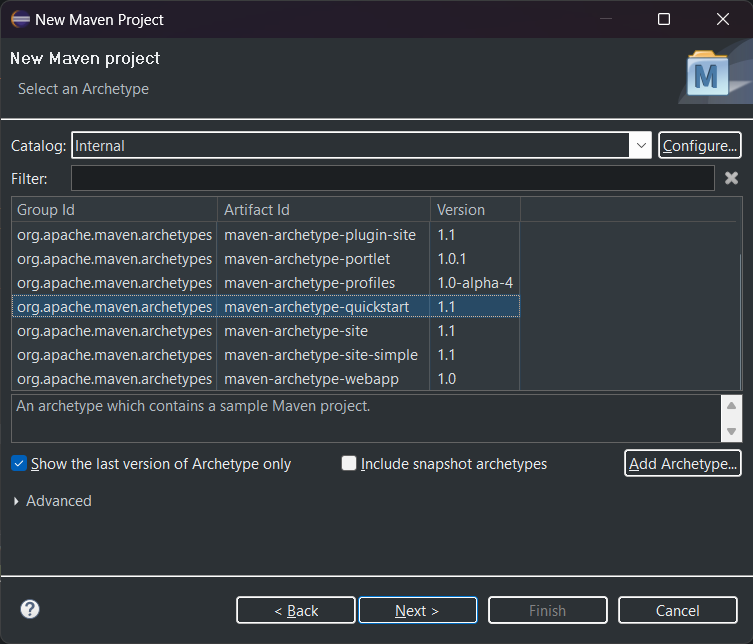
Para crear un proyecto Maven accederemos a la opción “Create a Project” y se abrirá un configurador en una ventana emergente con los distintos tipos de proyecto a crear. Habrá una opción de Maven y dentro “Maven Project”.



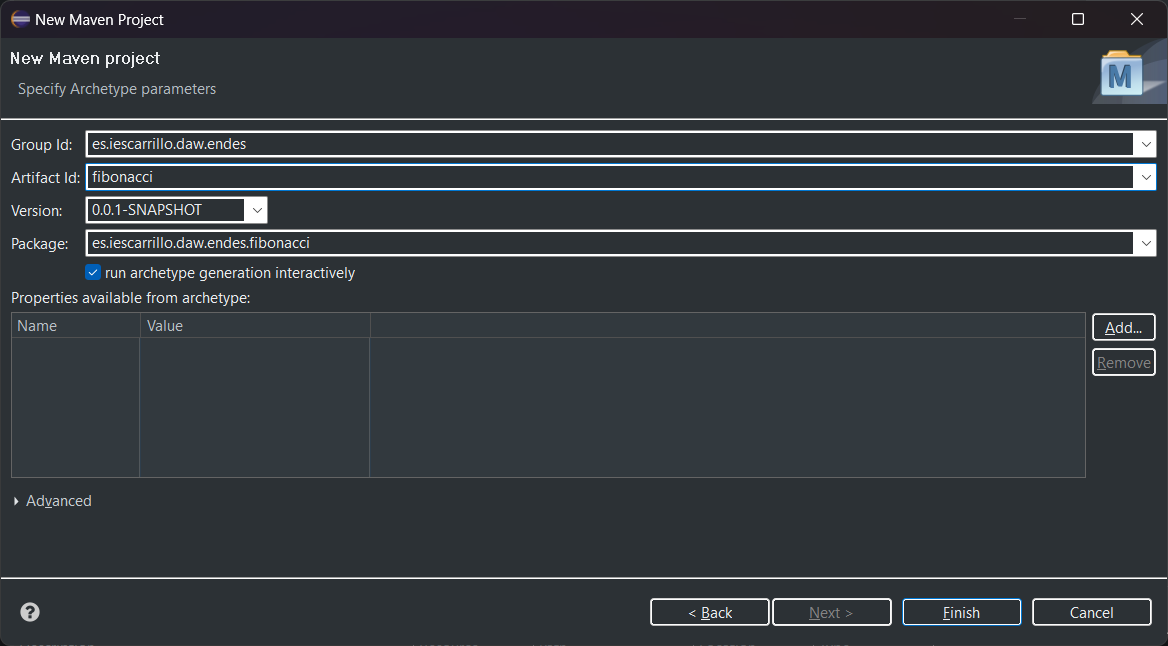
Una vez comience el proceso de creación se nos preguntará donde queremos que se establezca el Workspace del proyecto, simplemente habrá que establecer la ruta o si no se hace se establece el workspace por defecto.



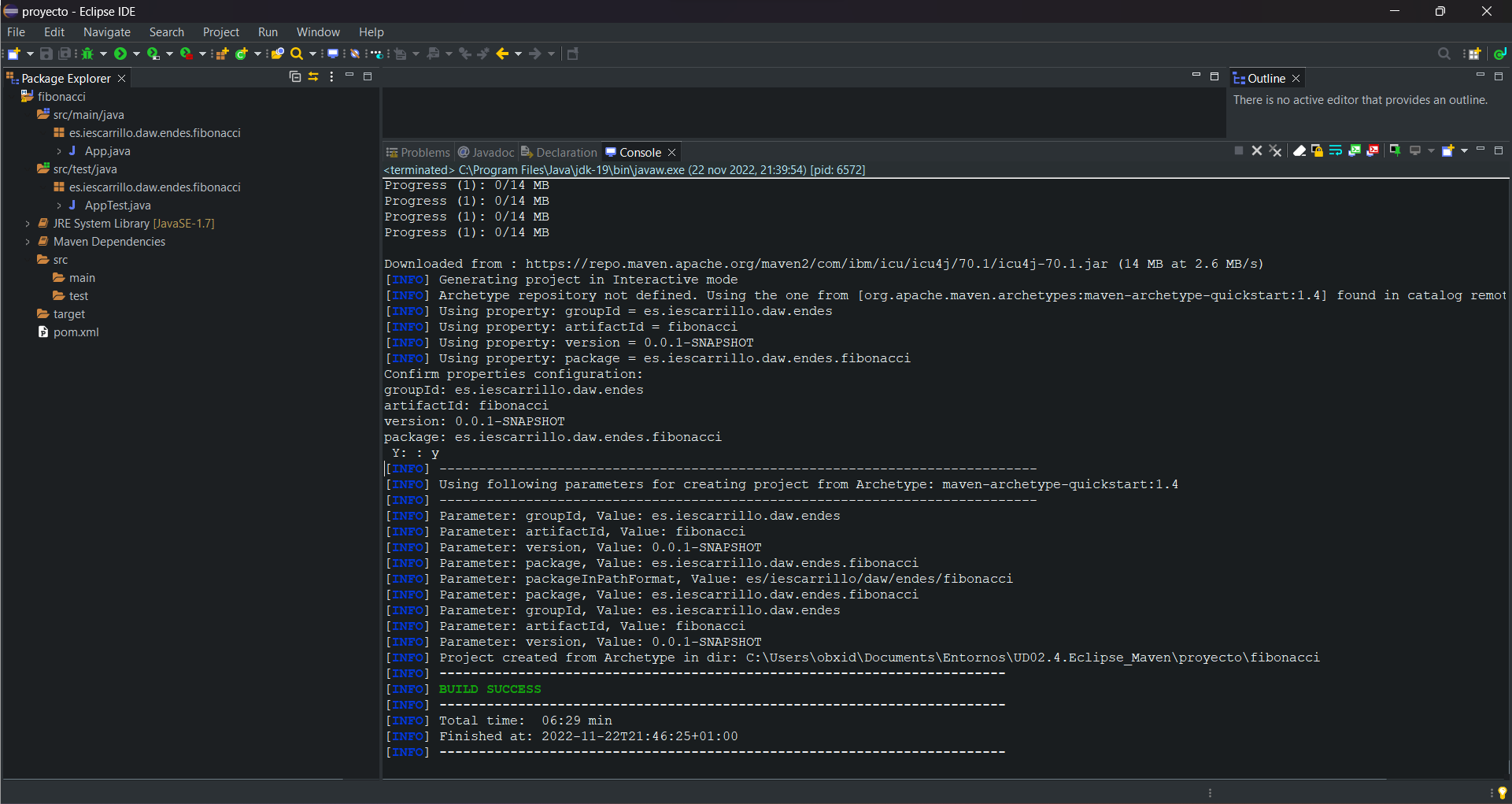
Ahora seleccionaremos el arquetipo en base al cual se va a generar el proyecto, en este caso vamos a usar el mismo arquetipo que usado en la práctica anterior de Maven el cual es el que tiene “Group Id” como “org.apache.maven.archetypes”, y de “Artifact Id” como “maven-archetype-quickstart” marcando en “Catalog” la opción a “Internal”.



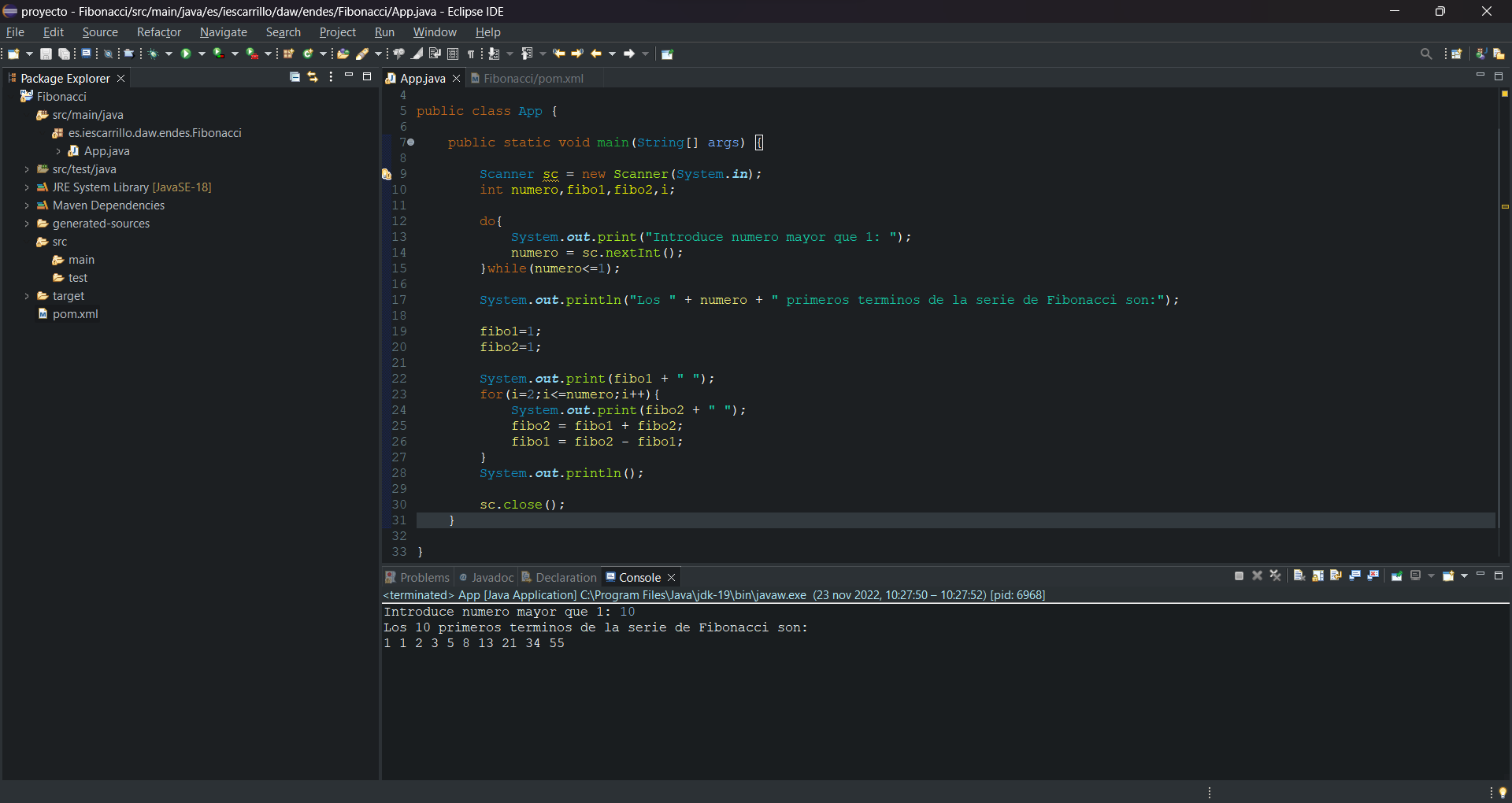
La siguiente ventana de la creación del proyecto nos pedirá que establezcamos el Group ID y el Artifact Id del proyecto:



Una vez finalizado el proceso de creación del proyecto comenzará a crearse y mostrar en la consola de eclipse los pasos que ya vimos anteriormente en la otra práctica de Maven cuando creamos el proyecto por comandos y verificaremos que se ha creado el directorio con el proyecto llamado “fibonacci" y e



Creamos dentro de la ruta “**es.iescarrillo.daw.endes.Fibonacci**” el archivo Fibonacci.java

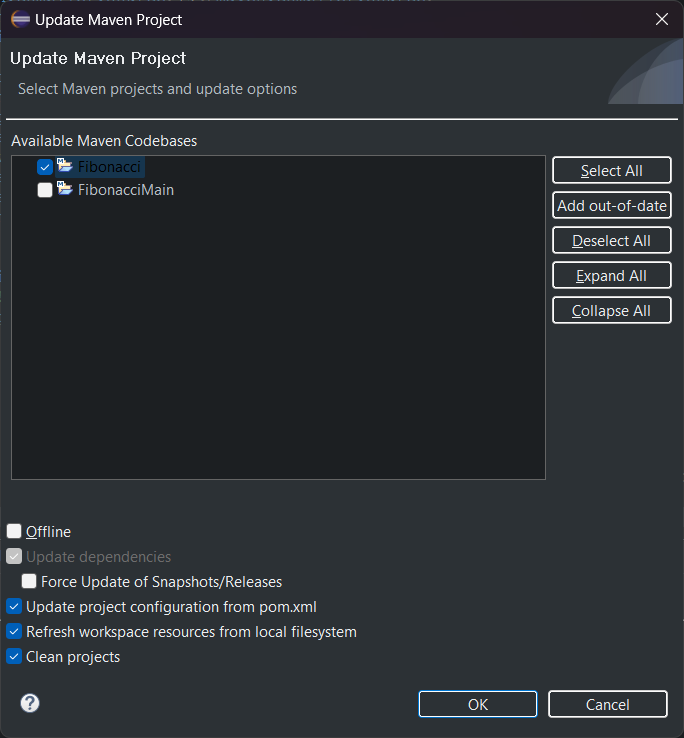


El siguiente paso será modificar el archivo “pom.xml” para establecer la configuración de las dependencias y plugins tal y como queremos tenerlo.

# Compilación y Ejecución del proyecto

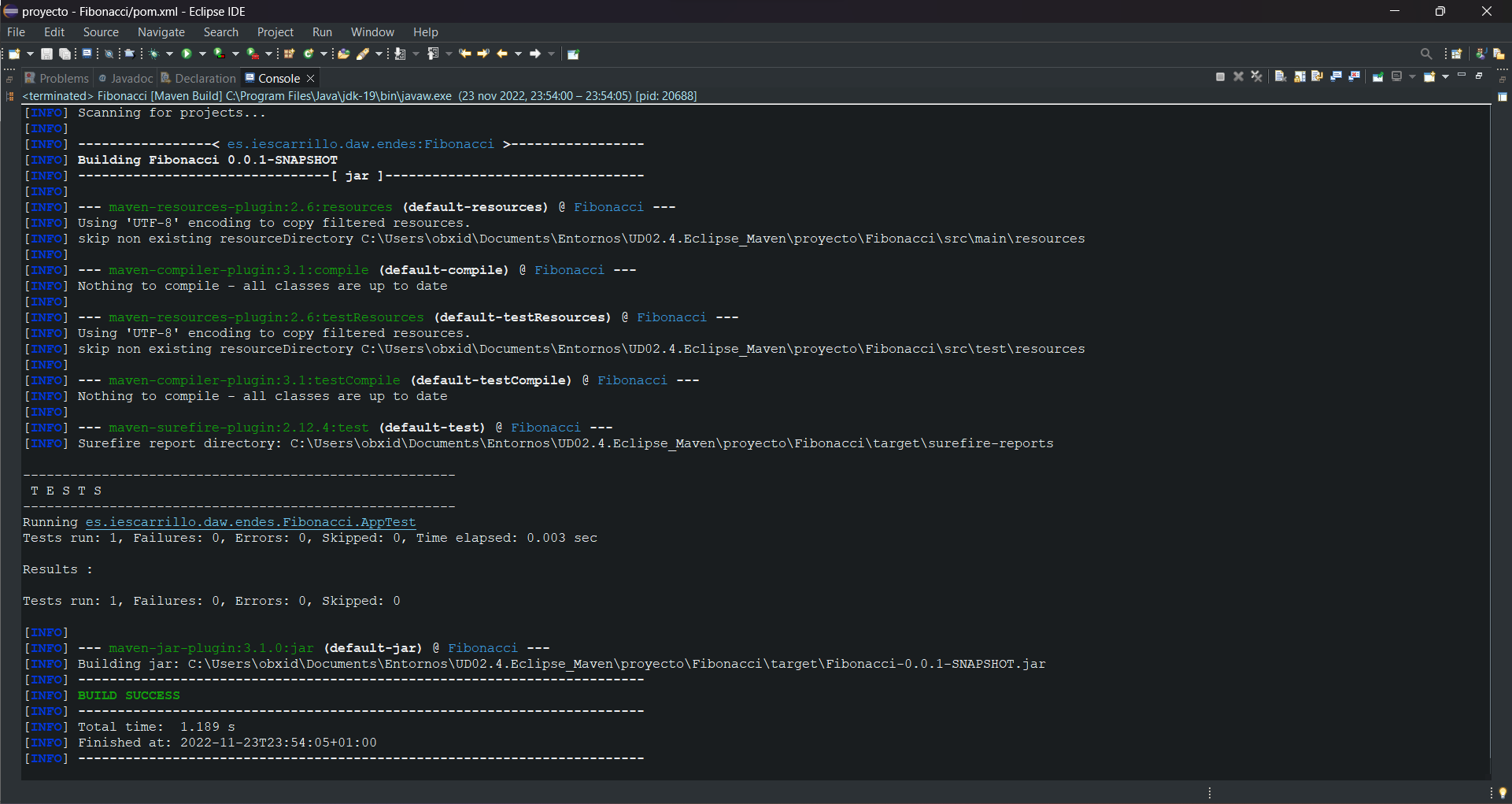
Una vez establecido el proyecto vamos a proceder a compilar, ejecutar y almacenar en el repositorio local nuestro proyecto Maven.

Para compilar el proyecto lo primero deberemos actualizar es el archivo pom.xml. Para ello haremos clic derecho en el proyecto vamos al apartado “Maven -> Update Project”

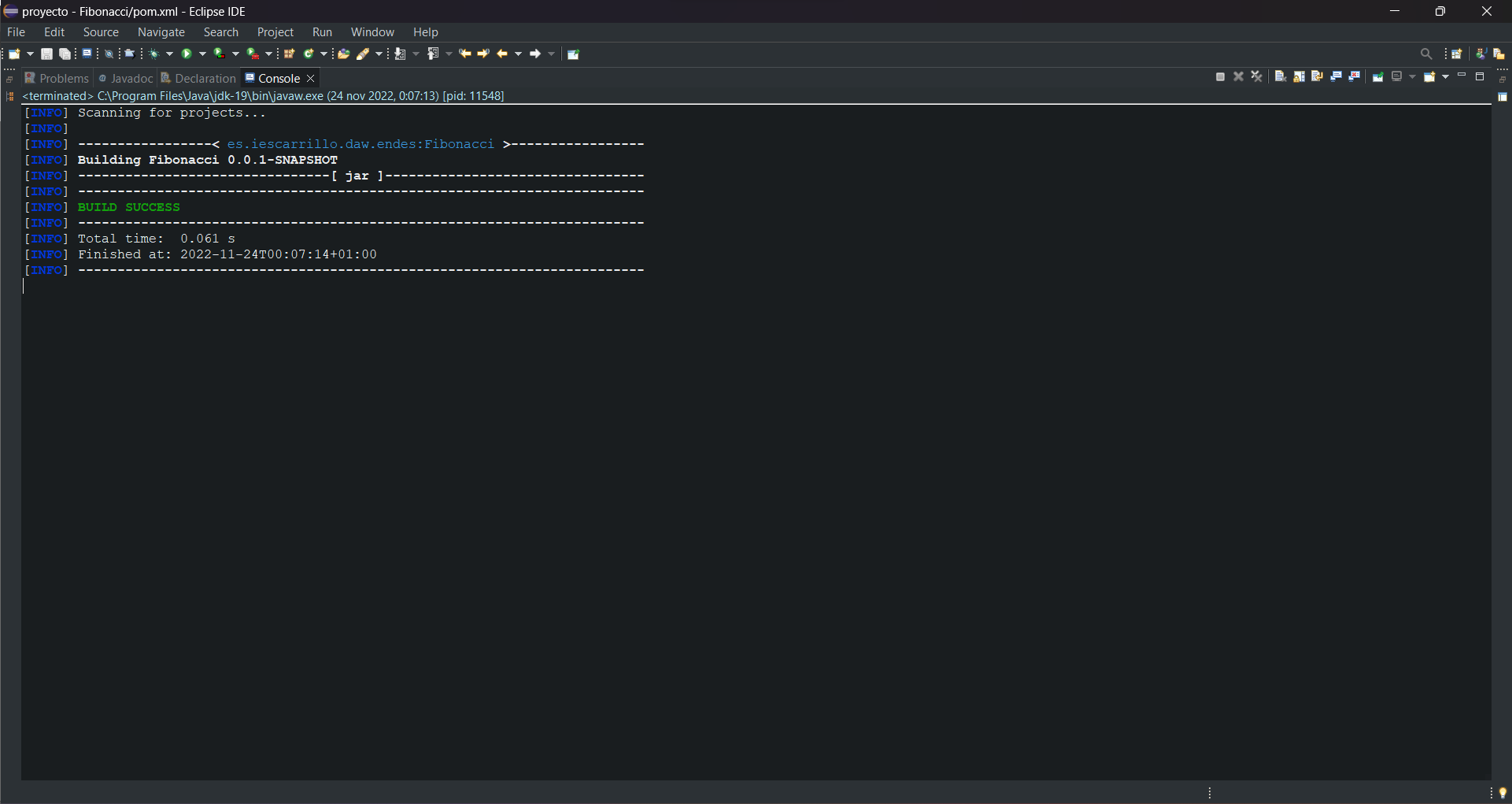


Una vez actualizado el fichero haremos clic derecho en el proyecto y no iremos a la opción “Run As -> Maven Build”. Si todo está correcto se compilará el programa y aparecerá la consola indicando la compilación se ha creado correctamente:

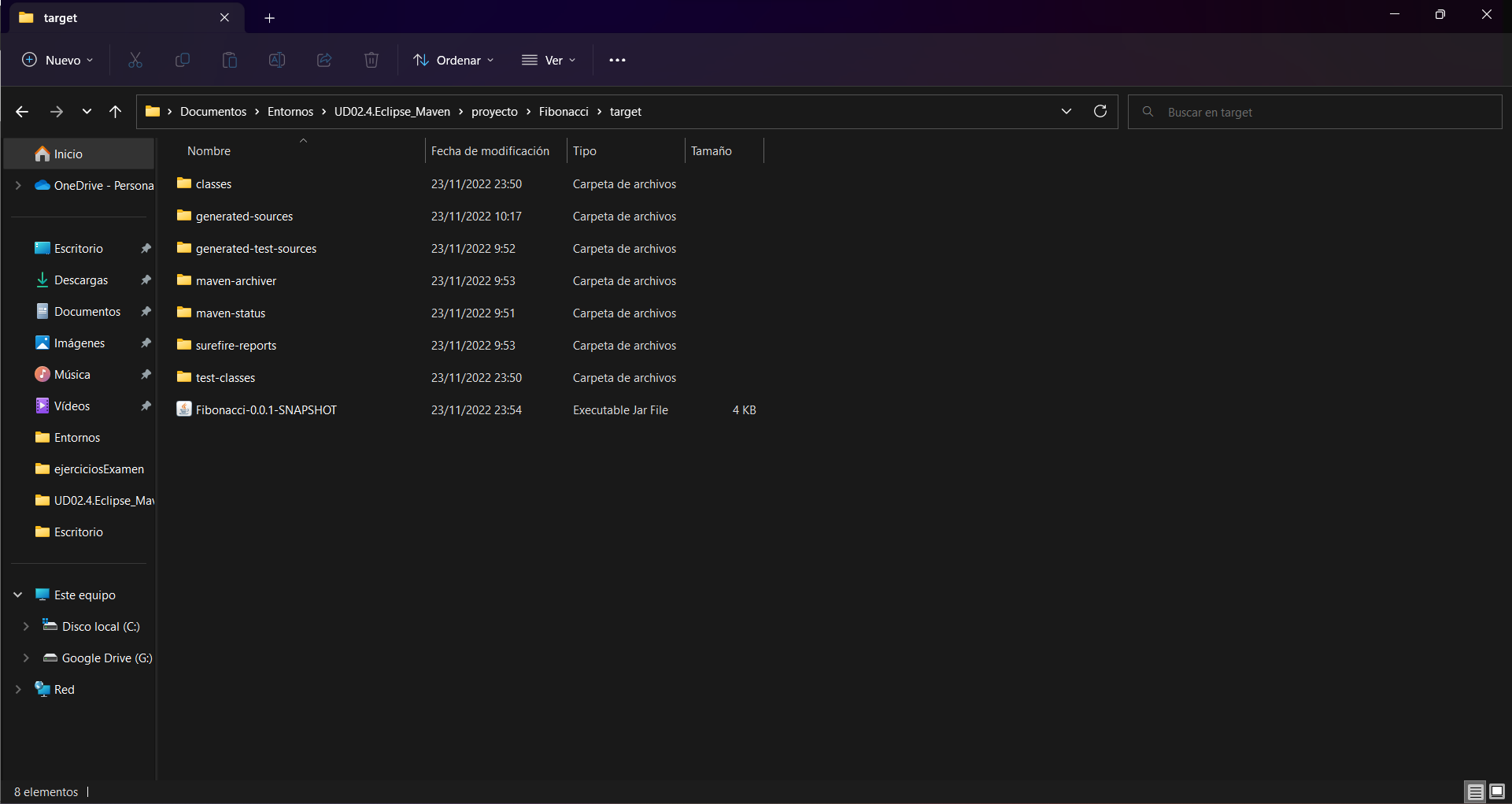
.



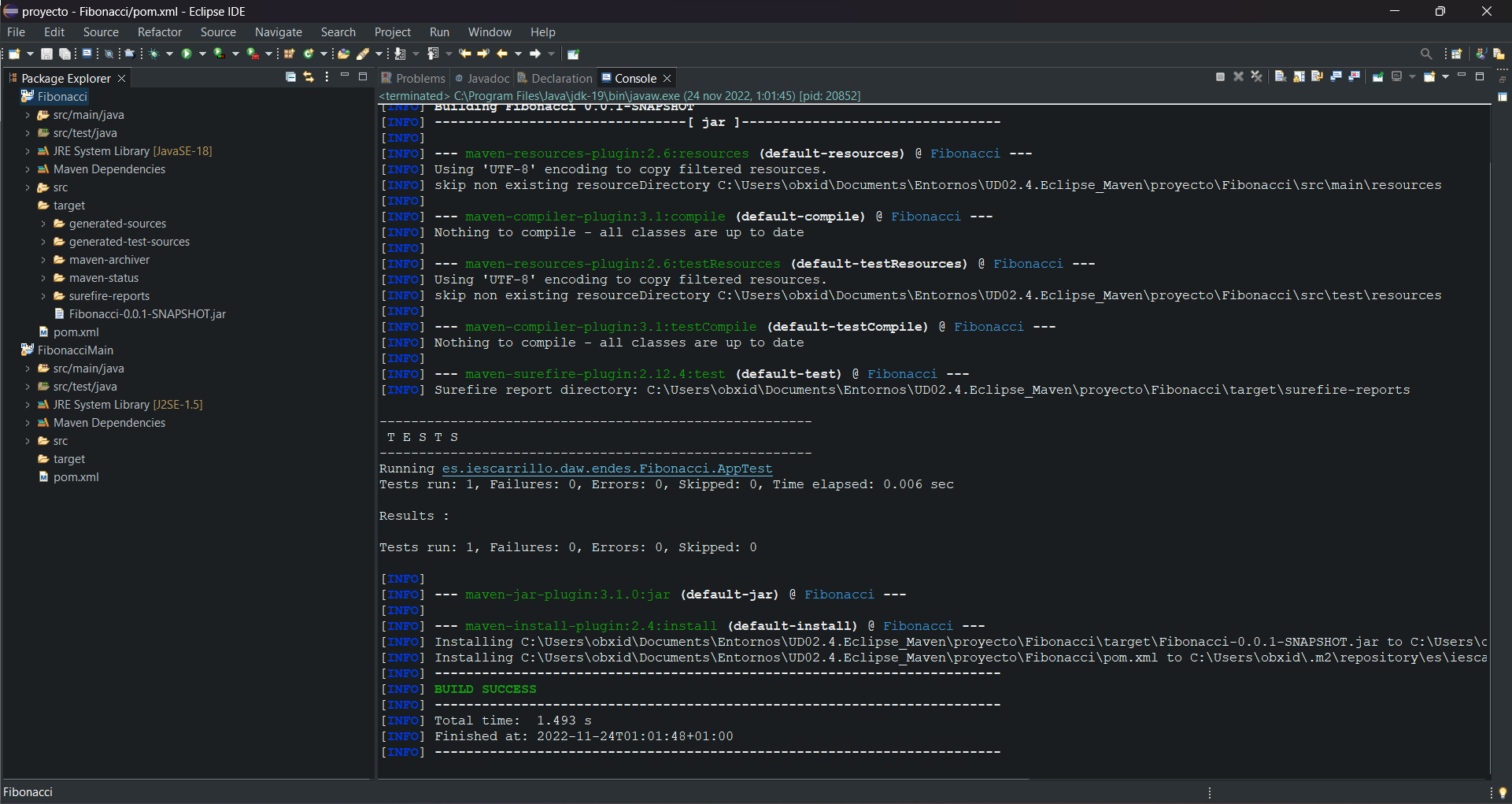
Una vez compilado el proyecto debemos de empaquetarlo para crear el .jar. Para ello damos clic derecho en el proyecto y no iremos a la opción “Run As -> Maven generate-resources”



Esto genera la carpeta target en el directorio principal del proyecto Maven y junto a ella encontramos varias subcarpetas generadas junto al archivo .jar, en este caso “Fibonacci-0.0.1-SNAPSHOT”

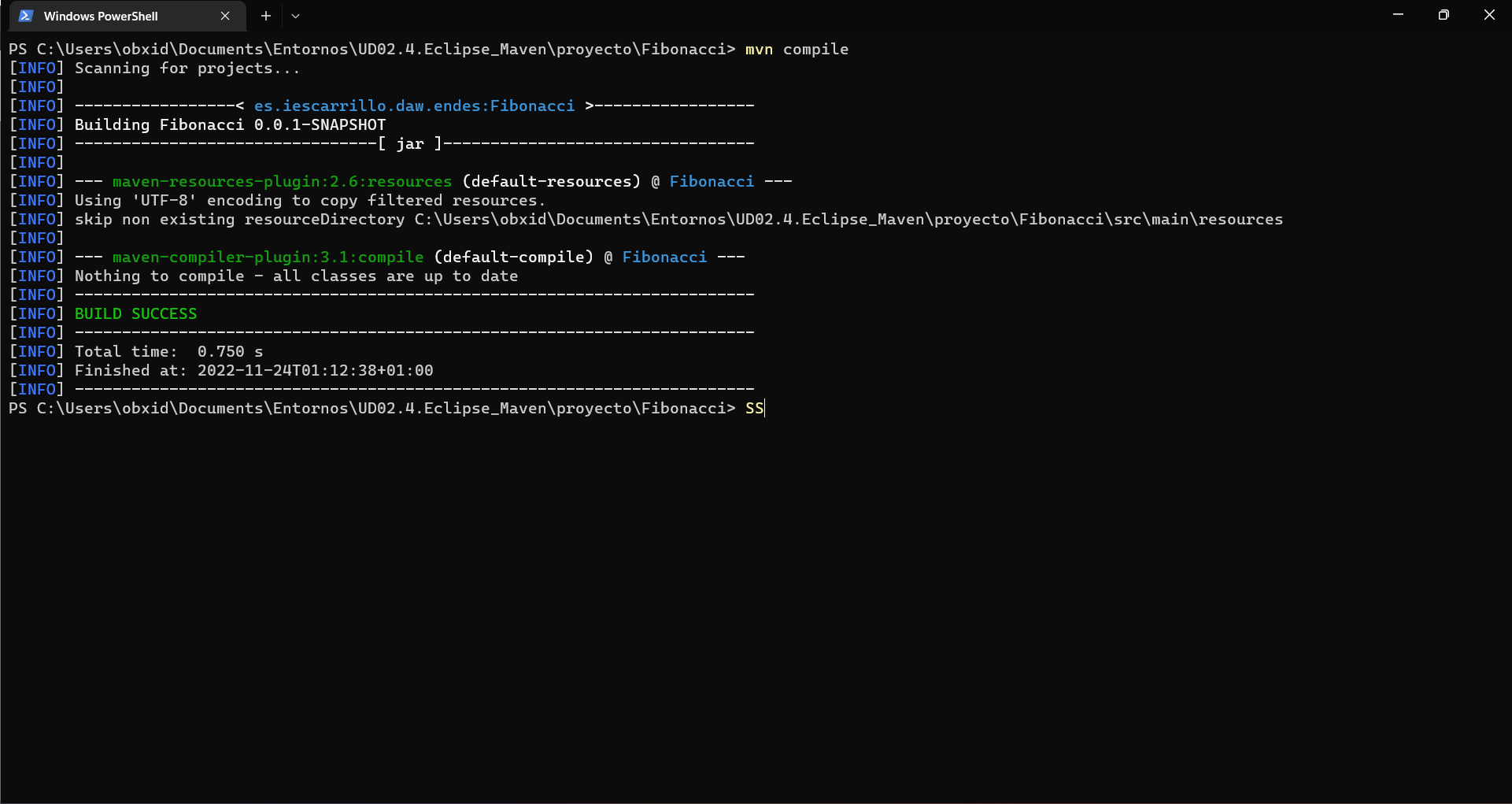


El último paso será instalar el proyecto a un repositorio local junto con el archivo .jar. Para ello volvemos a dar clic derecho sobre el proyecto y pulsamos en “Run As -> Maven Install”.

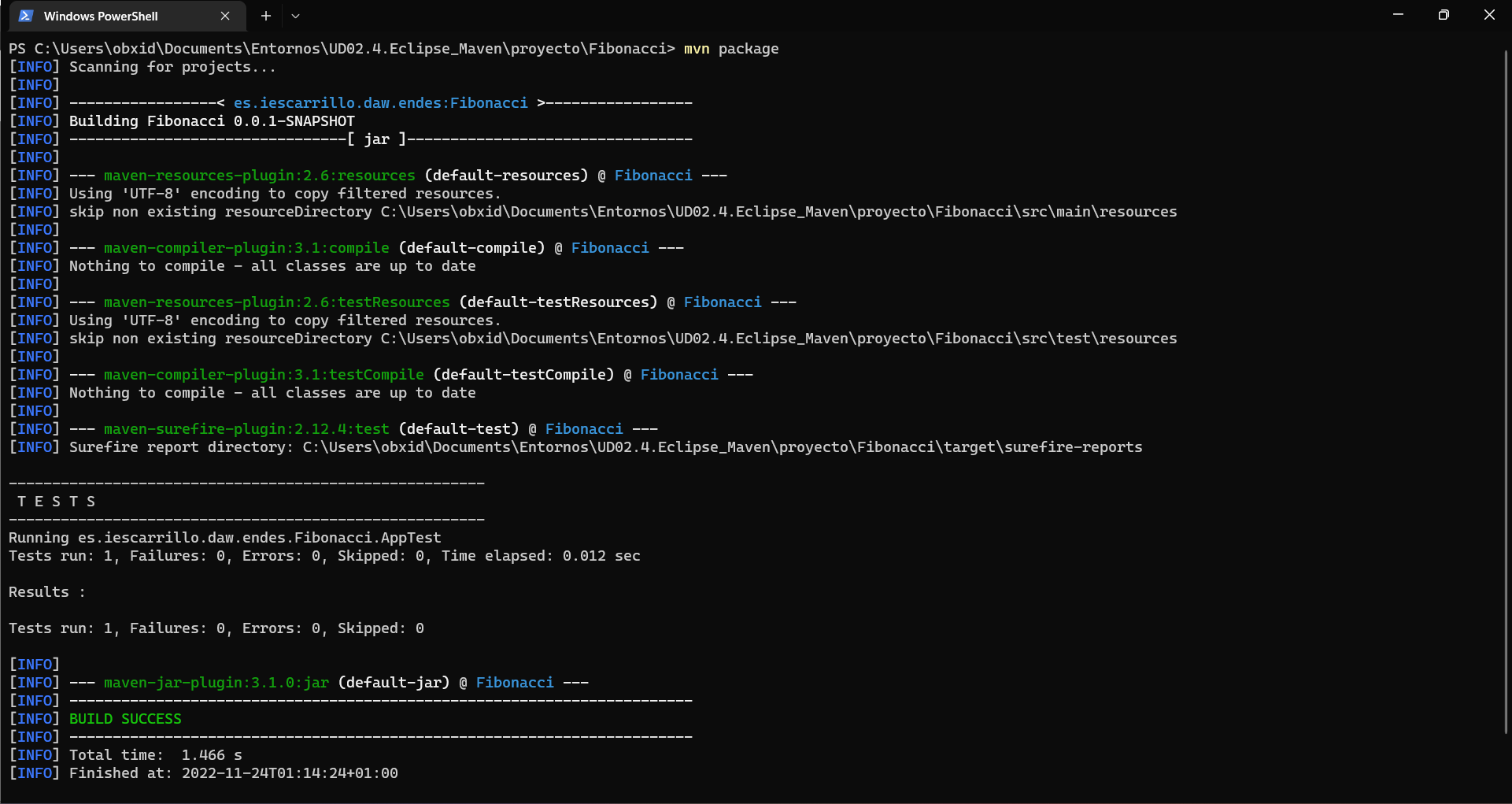


Por último, vamos a verificar que se pueda realizar estos mismos pasos desde una terminal con comandos Maven.

#### Compilación



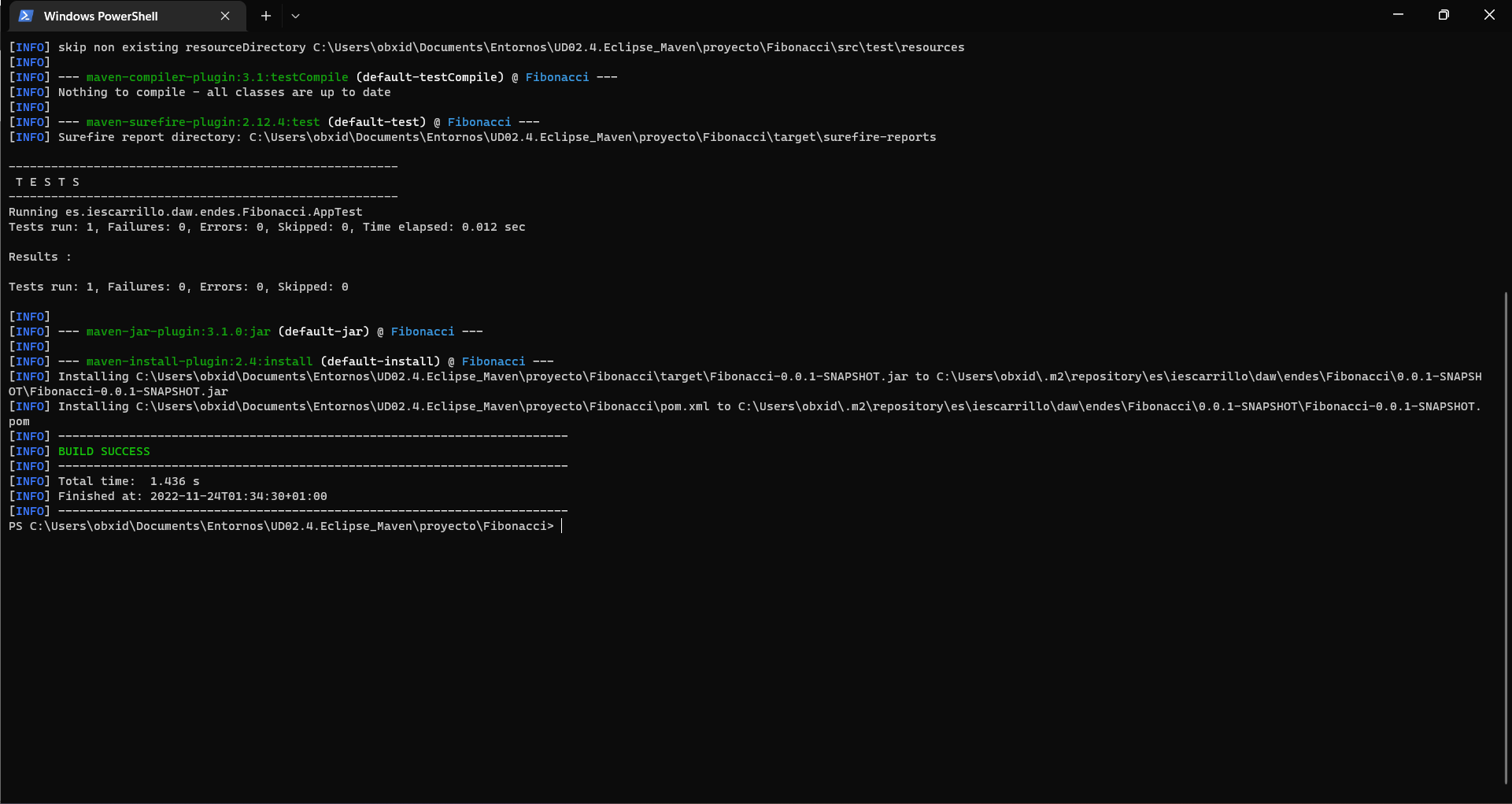
#### Empaquetado



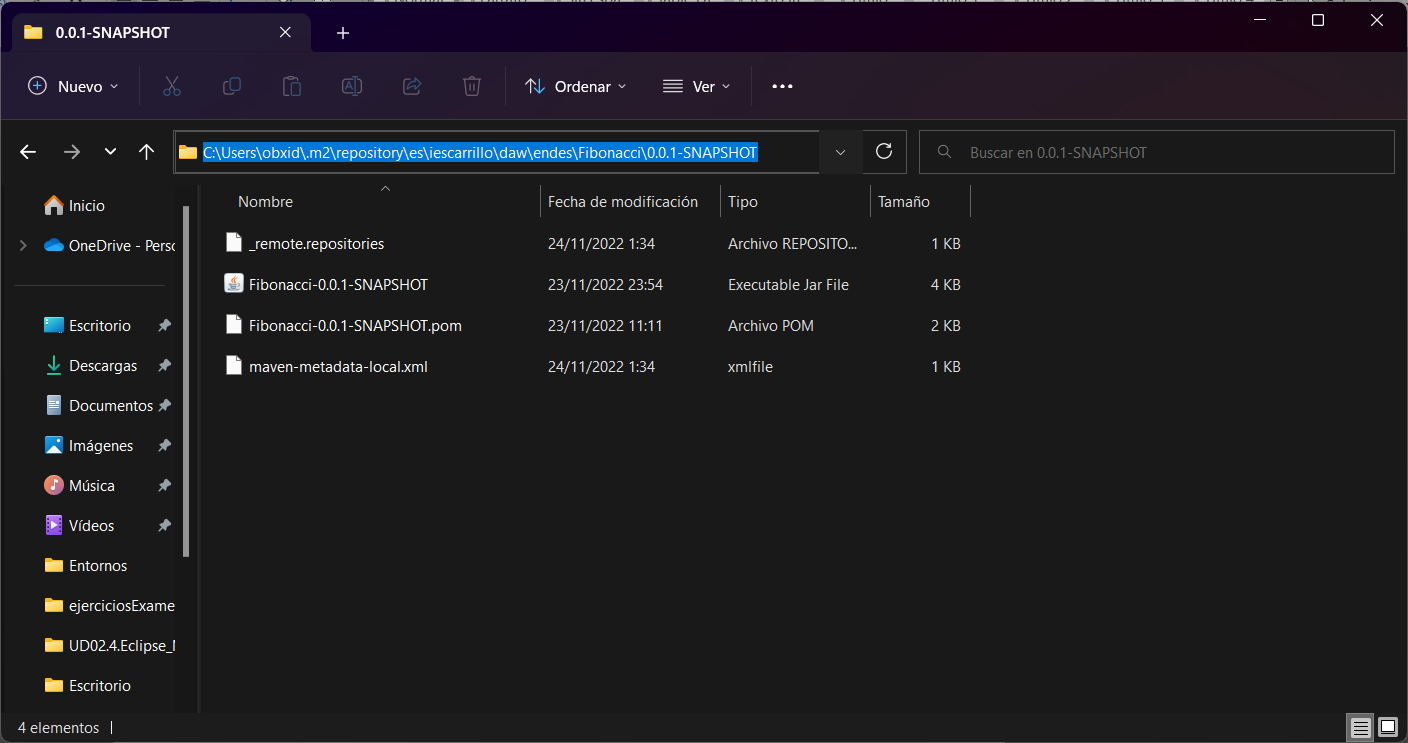
#### Ejecución

#### 

#### Instalación



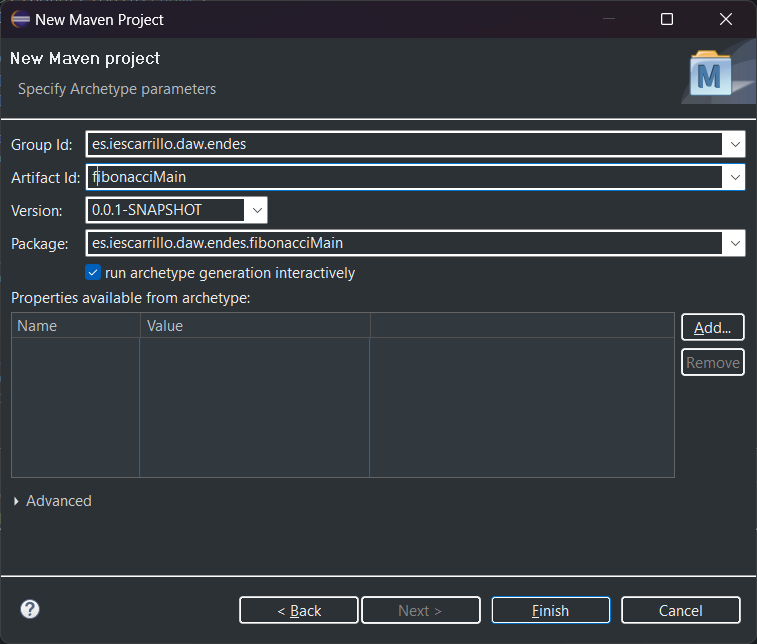
Tras revisar el repositorio local de Maven de nuestro equipo podemos ver que se ha instalado en el equipo correctamente.

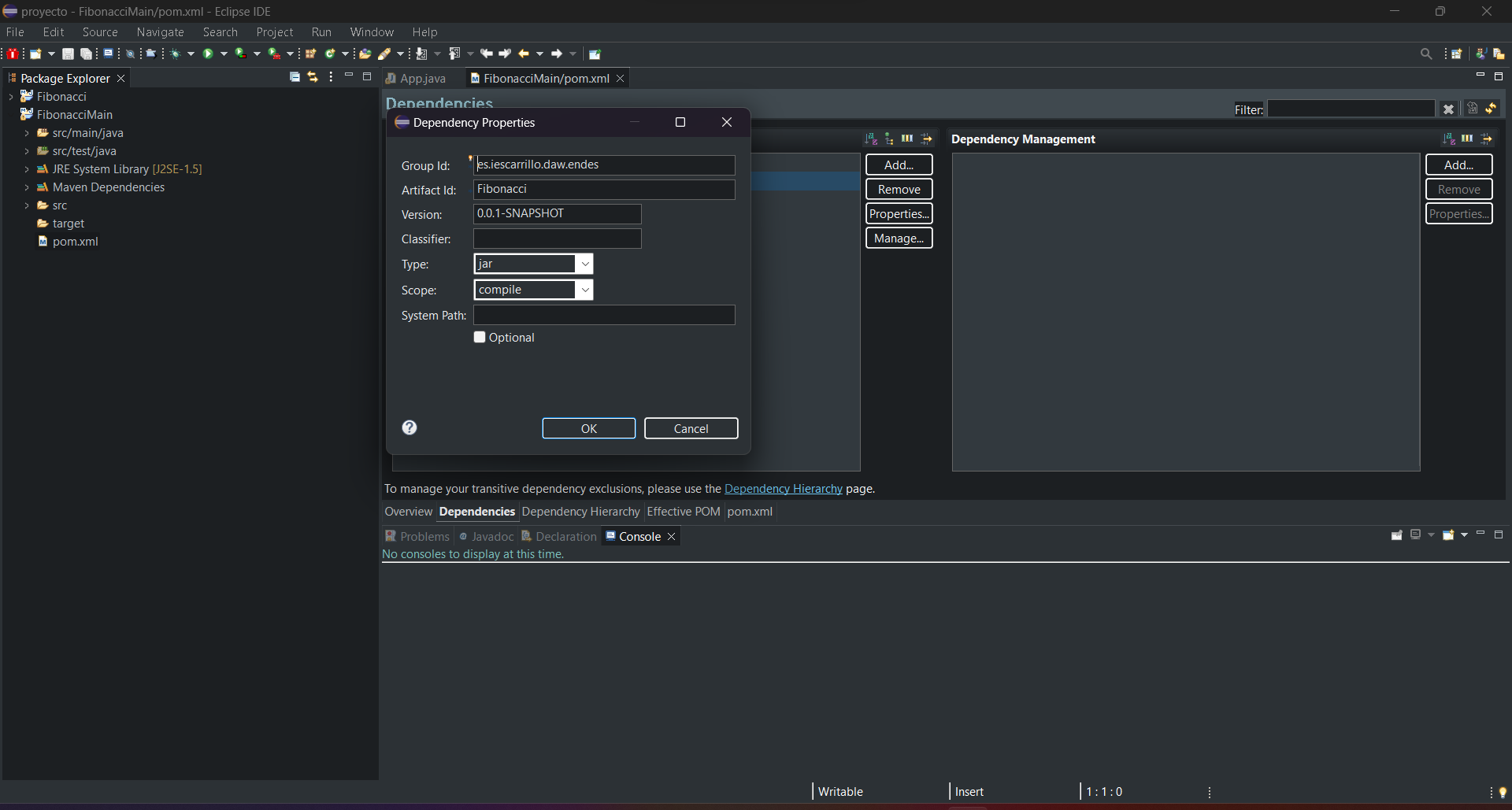


# Gestión de Dependencia

El primer paso para realizar la gestión de las dependencias será crear un nuevo proyecto Maven, en nuestro caso se nos solicita que se llame FibonacciMain que se encuentre en el mismo GroupId y que use el artefacto del proyecto Fibonacci.

Para ello creamos el proyecto igual que creamos el proyecto Fibonacci:



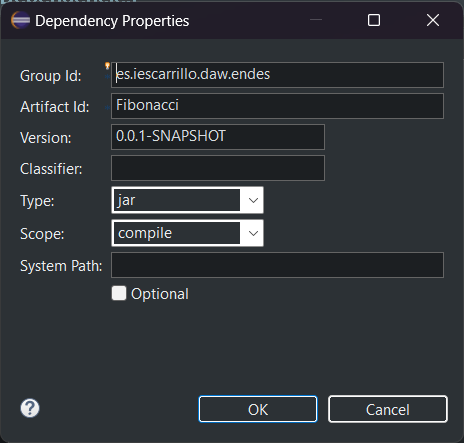
 Una vez creado el proyecto deberemos de ir al pom.xml para configurar las dependencias.

1

3

2

4



5

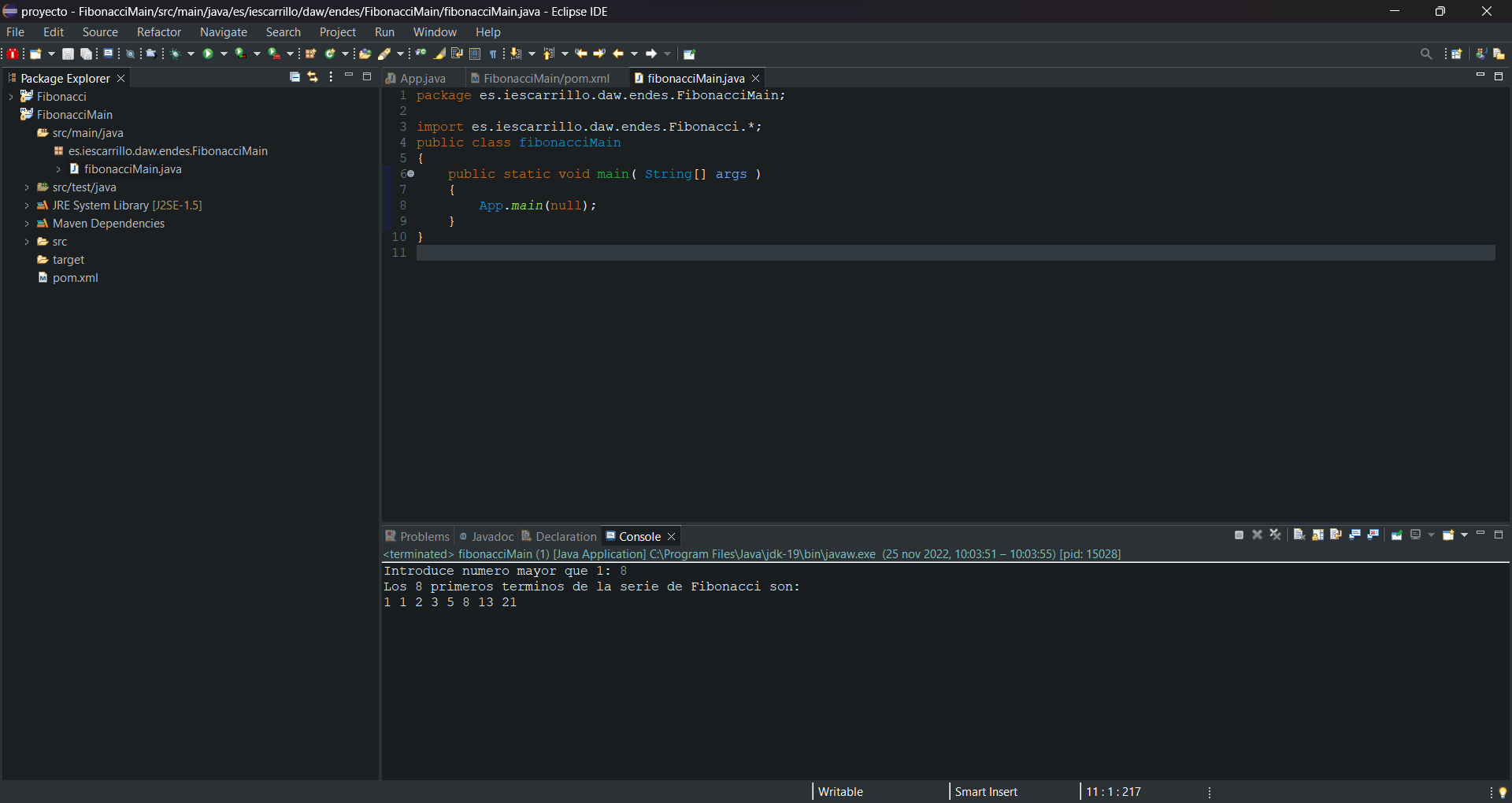
Deberemos de añadir el Group Id que comparten los dos proyectos, el Artefact Id del proyecto anterior, en este caso “Fibonacci”, y la versión utilizada, en este caso “0.0.1-SNAPTSHOT”.

## Creación de fichero java

El siguiente paso será crear el fichero .java donde realizaremos la importación del proyecto Fibonacci en el proyecto FibonacciMain. Para ello vamos a usar:

Una vez importado deberemos realizar la llamada a la clase “App.java” del proyecto Fibonacci. Para realizar la llamada dentro del método main de nuestra clase Fibonacci lo siguiente:

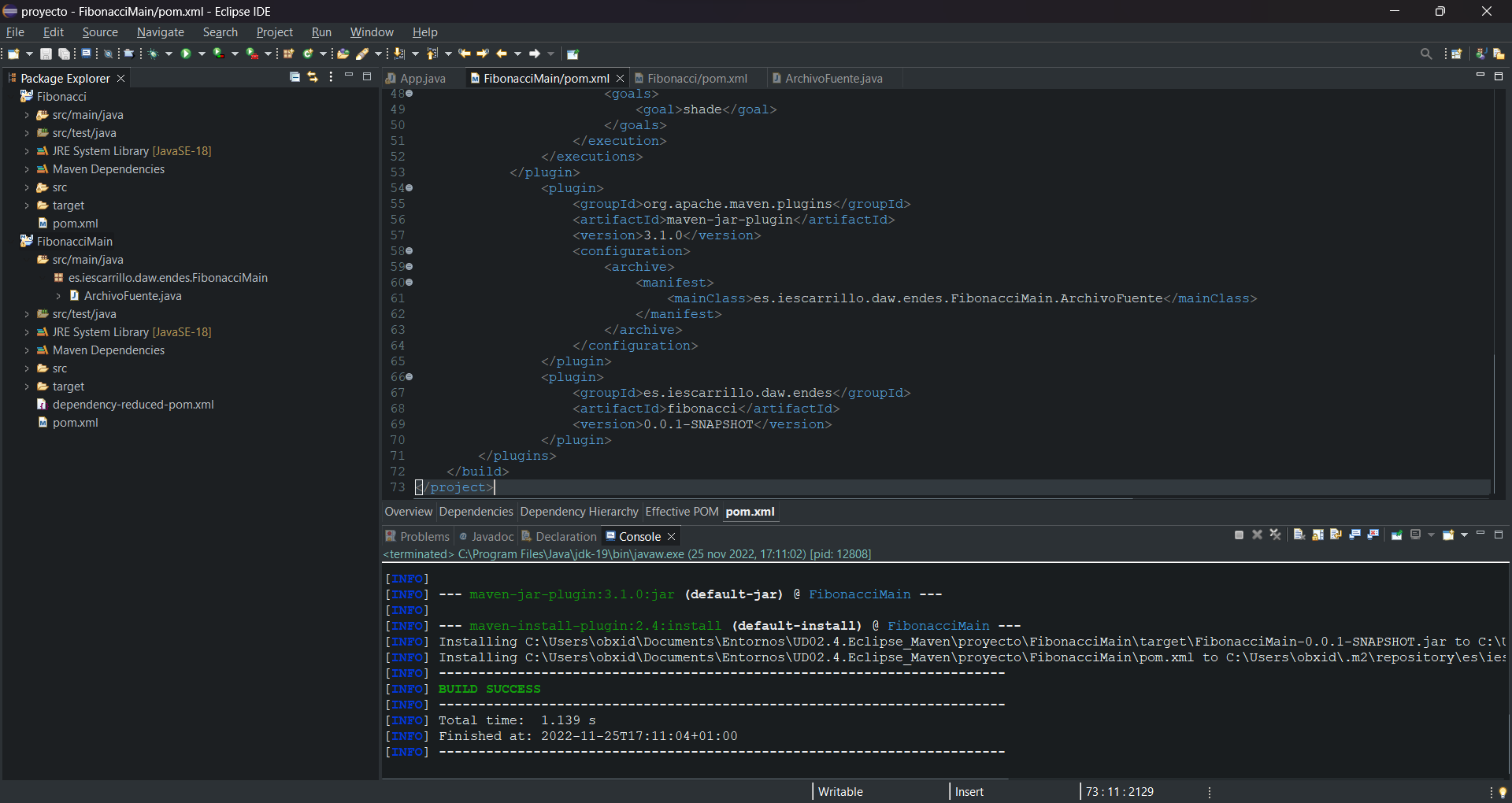
Al ejecutar podemos ver que se ejecuta correctamente:



Tras esto realizaremos de nuevo los comandos sobre el proyecto al igual que en los pasos anteriores:

# Preparación de los archivos para la ejecución desde consola

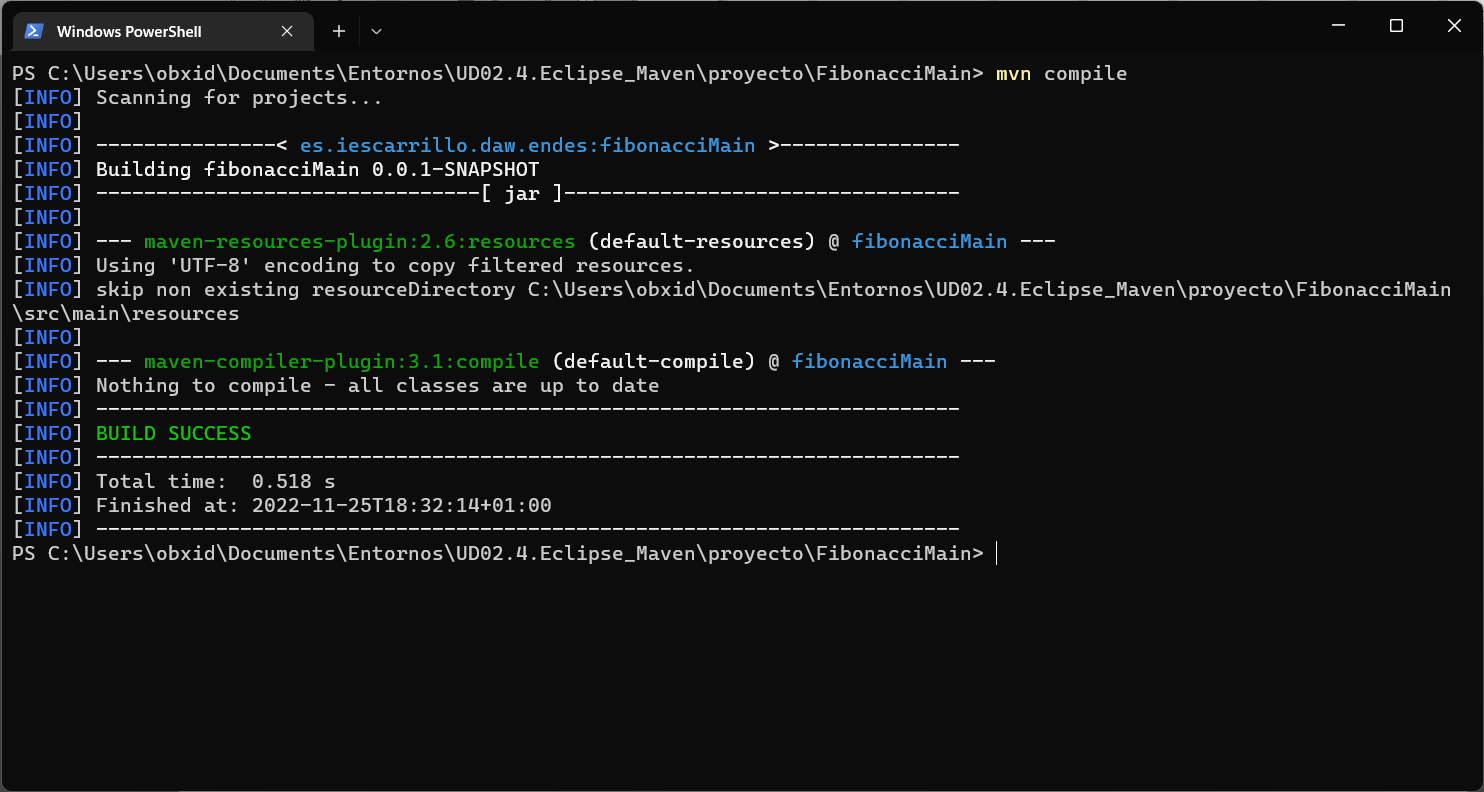
El siguiente paso será realizar modificación en el pom.xml para cuando se realice la ejecución mediante consola. Tememos que definir de donde debe de tomar el método Main para su ejecución.



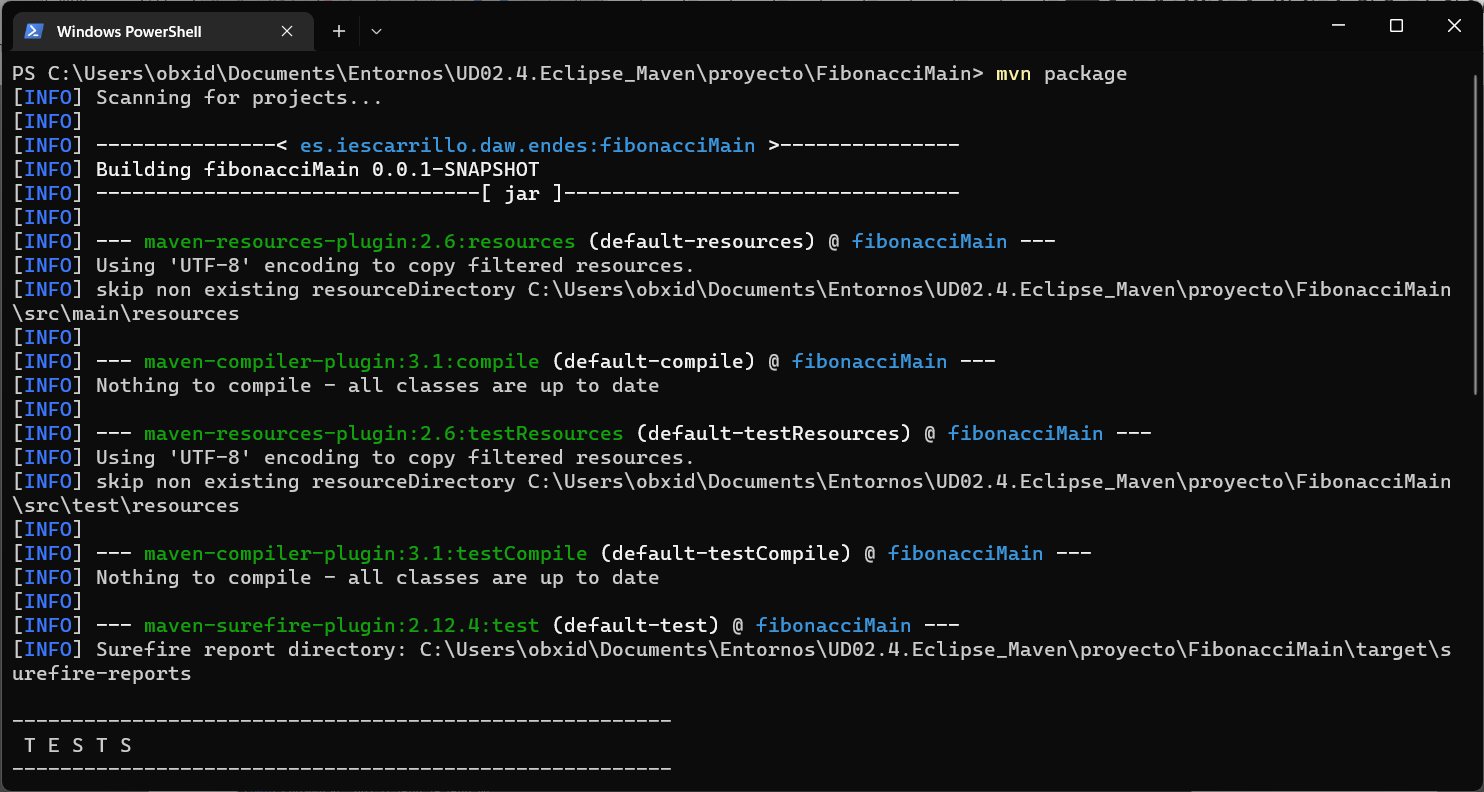
Para evitar errores en la ejecución fuera de eclipse deberemos de añadir también el plugin Maven shades para que el proyecto pueda gestionar todas las dependencias de los distintos proyectos ubicados en el Group Id.

En mi caso el pom.xml usado es el siguiente:

#### Compilación



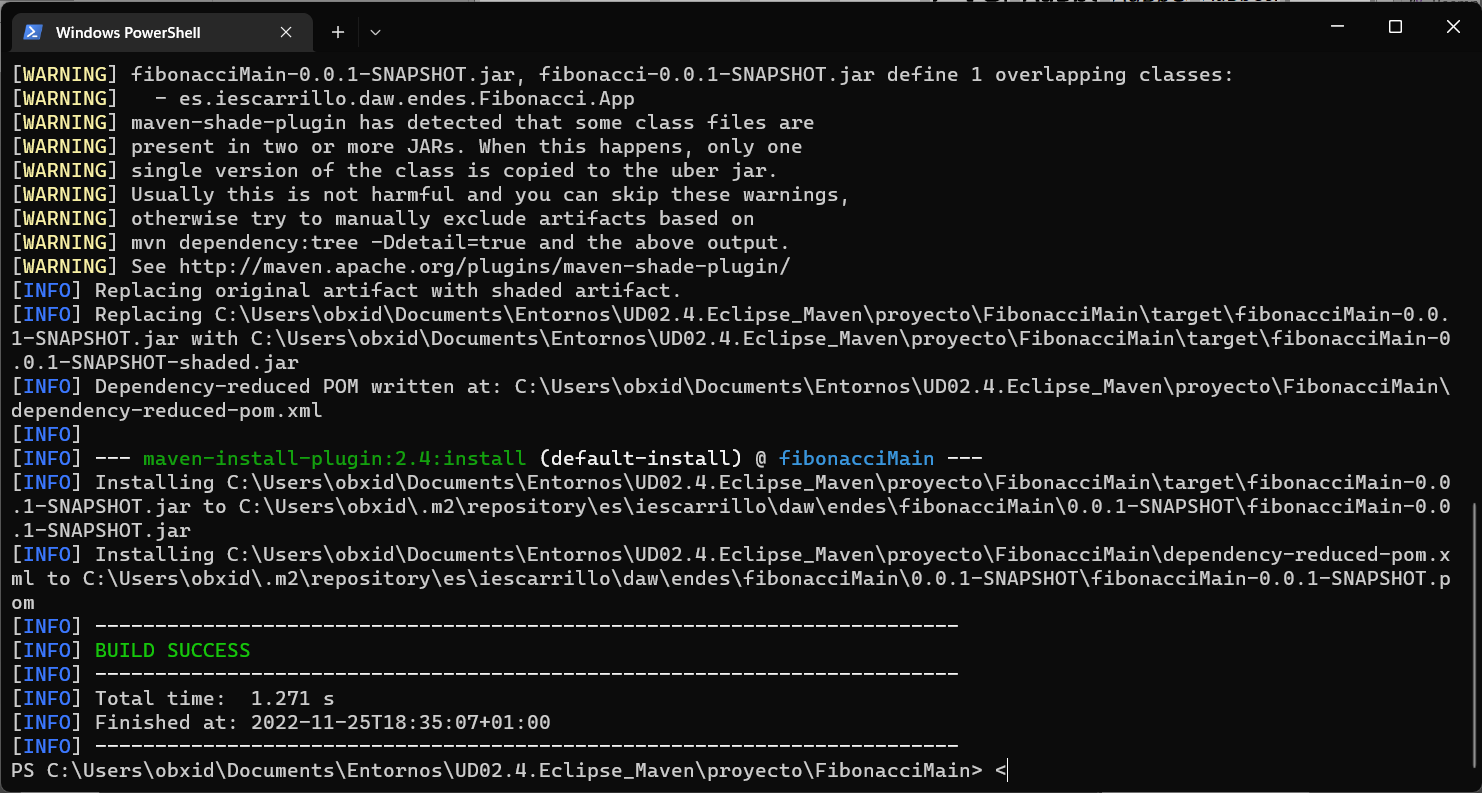
#### Empaquetado



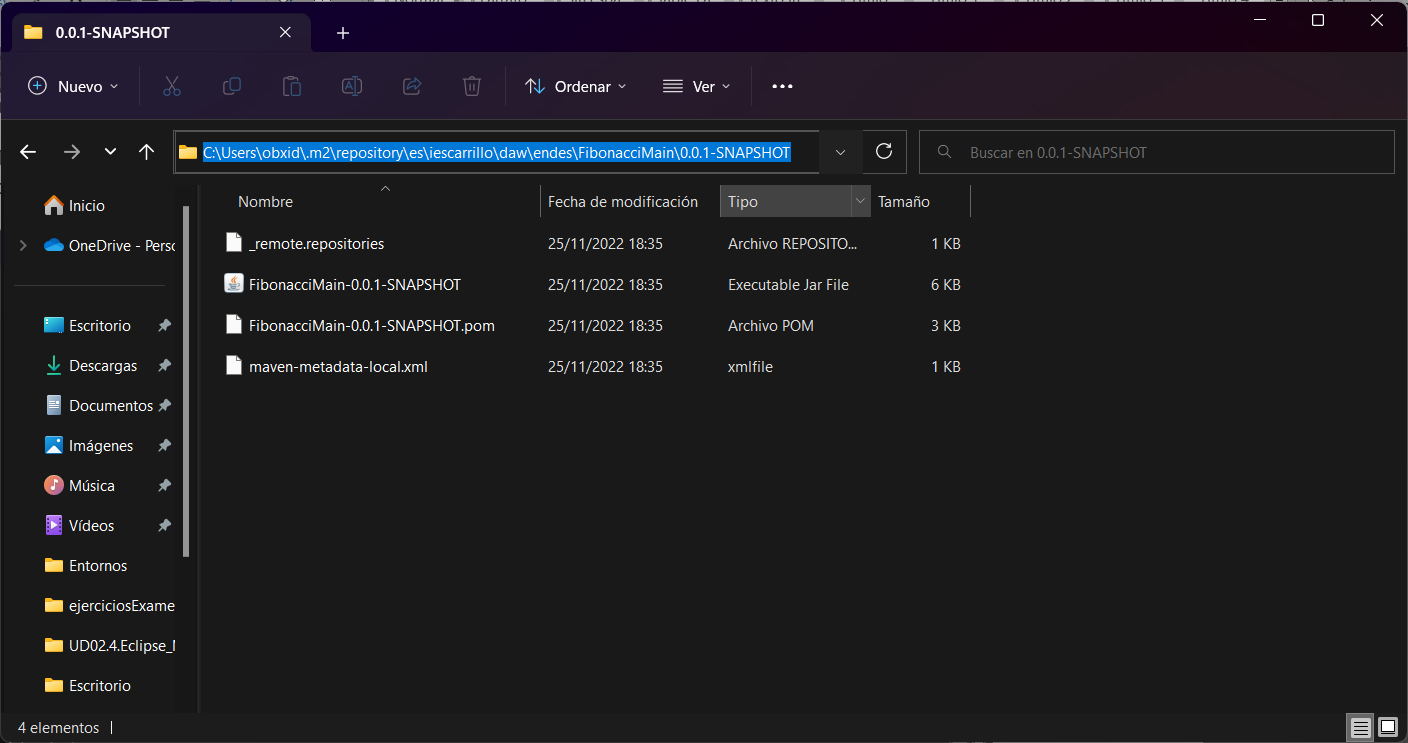
#### Ejecución

#### 

#### Instalación



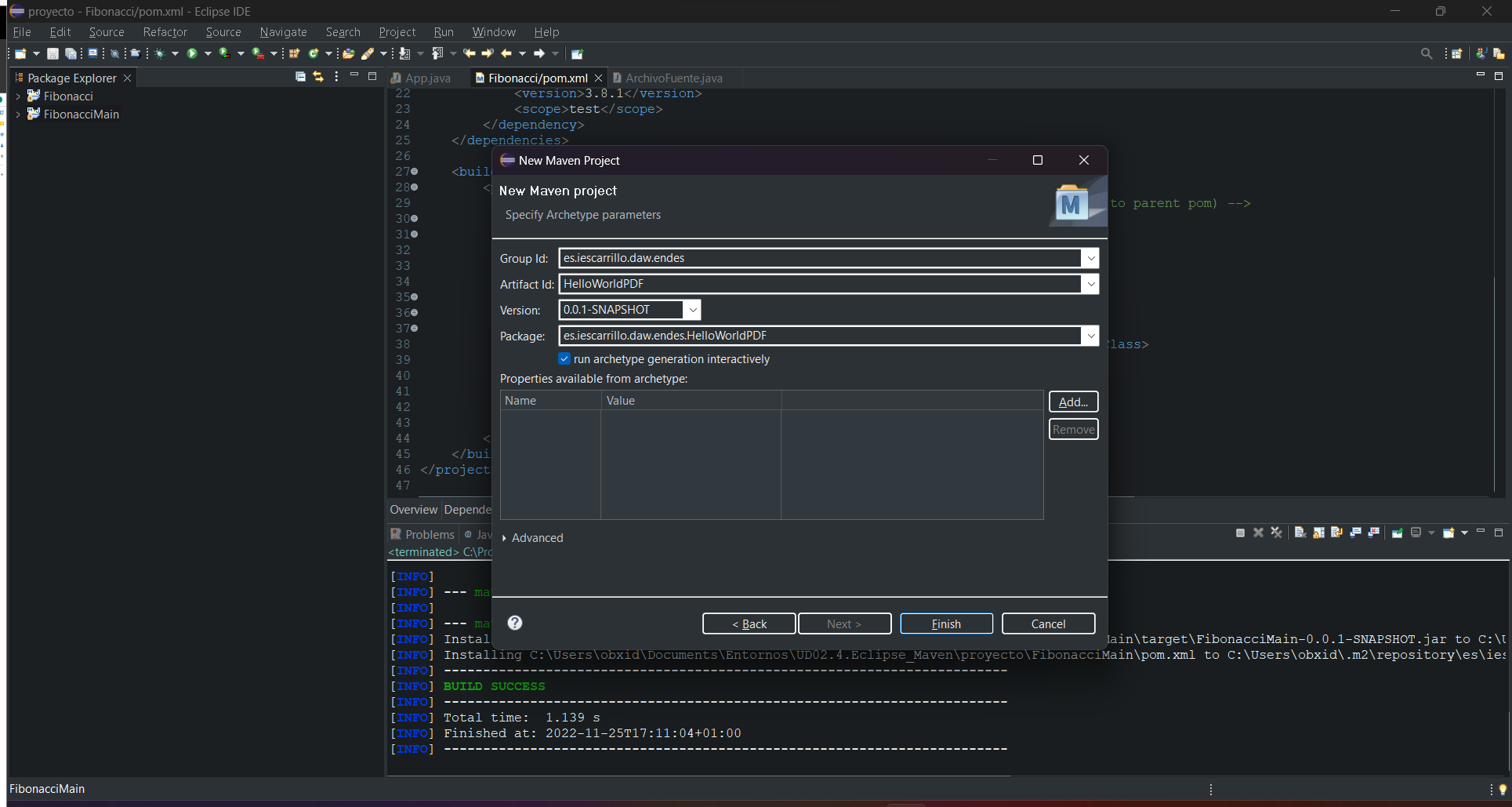
Tras revisar el repositorio local de Maven de nuestro equipo podemos ver que se ha instalado en el equipo correctamente.



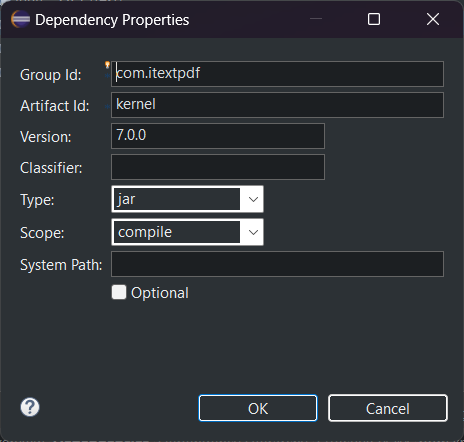
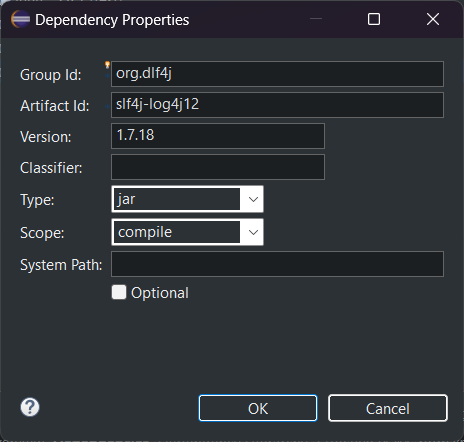
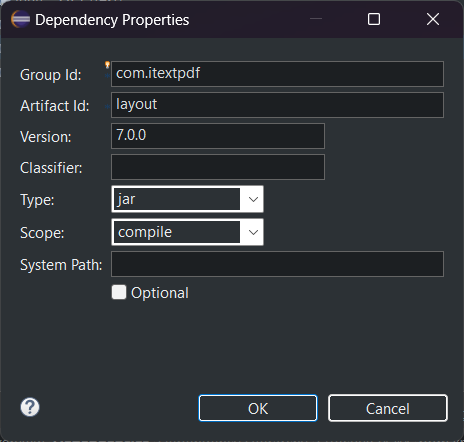
# Proyecto HelloWorldPDF

En esta parte vamos a crear un proyecto Maven dentro del mismo Group Id anteriormente usado que use la librería iText para que se genere un documento PDF con el texto.

Para ello creamos el proyecto Maven como en los pasos anteriores nombrándolo como “HelloWorldPdf”

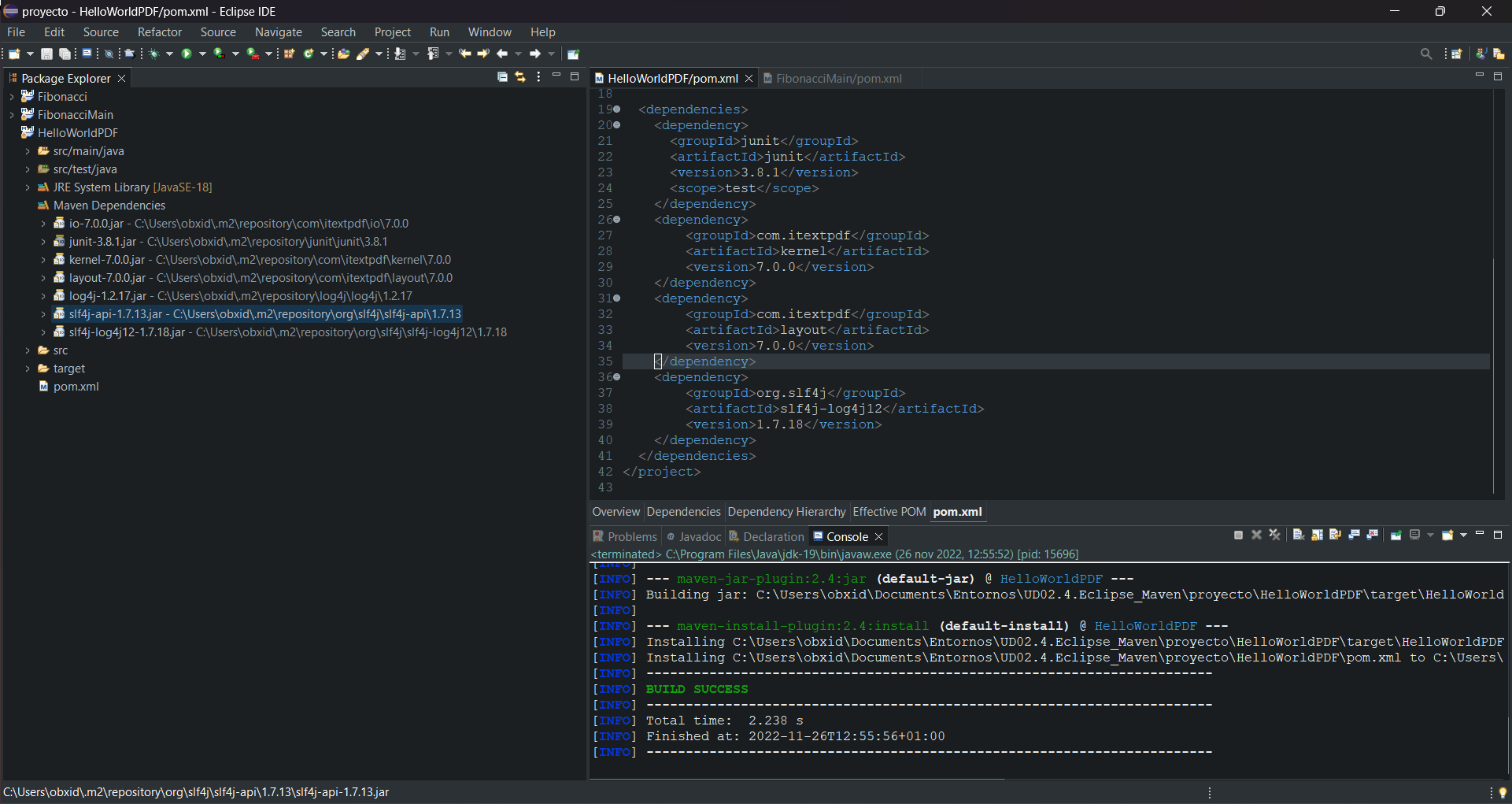


El siguiente paso será añadir las dependencias que se van a usar en el proyecto.



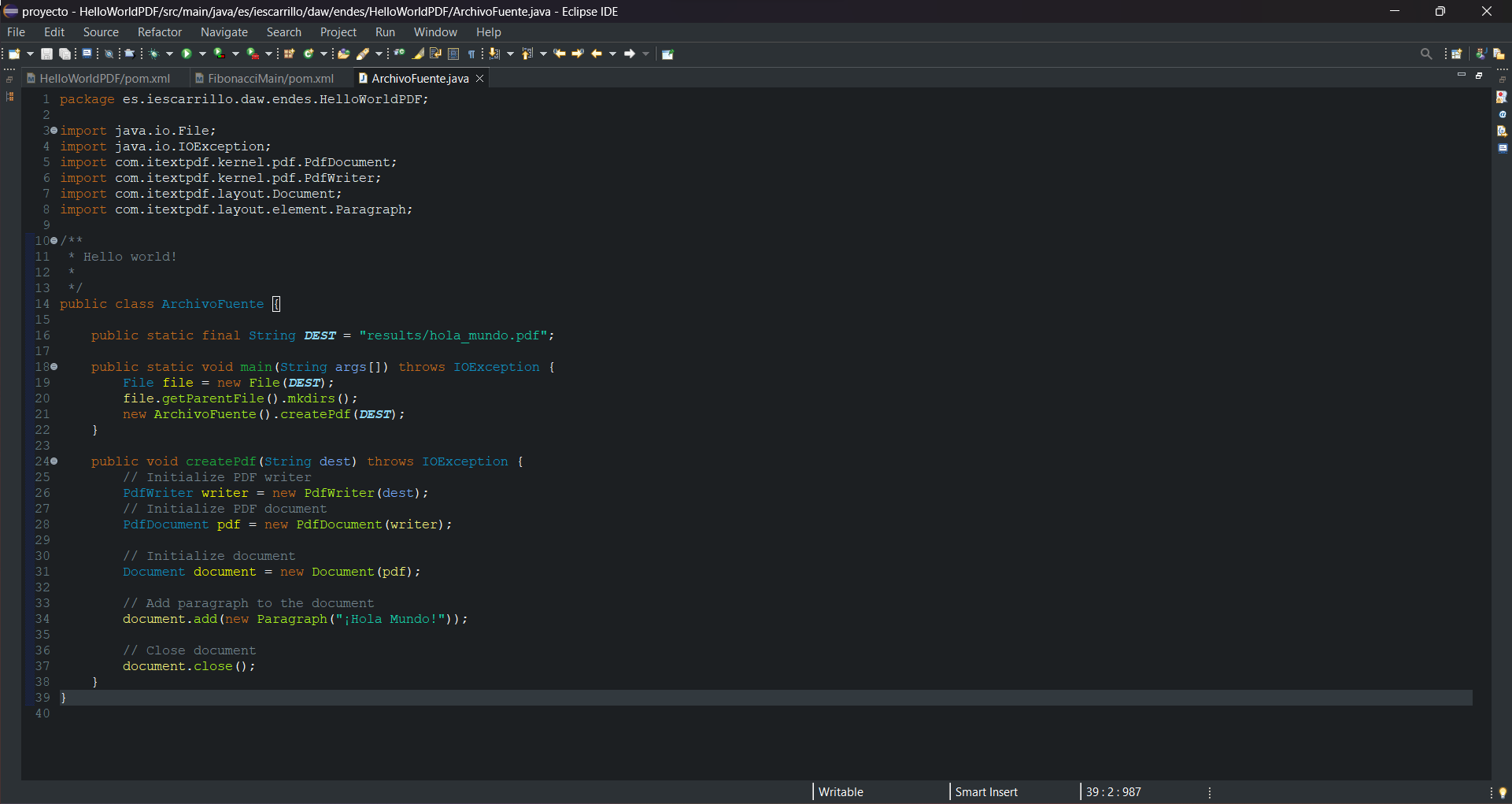
Una vez hecho esto nos iremos al proyecto y haciendo clic derecho “Run As -> Maven Install” para instalar estas dependencias en el repositorio local de Maven.

Tras esto podemos ver que dentro del proyecto hay un apartado con nombre “Maven Dependences” donde aparecen estas dependencias ya importadas.

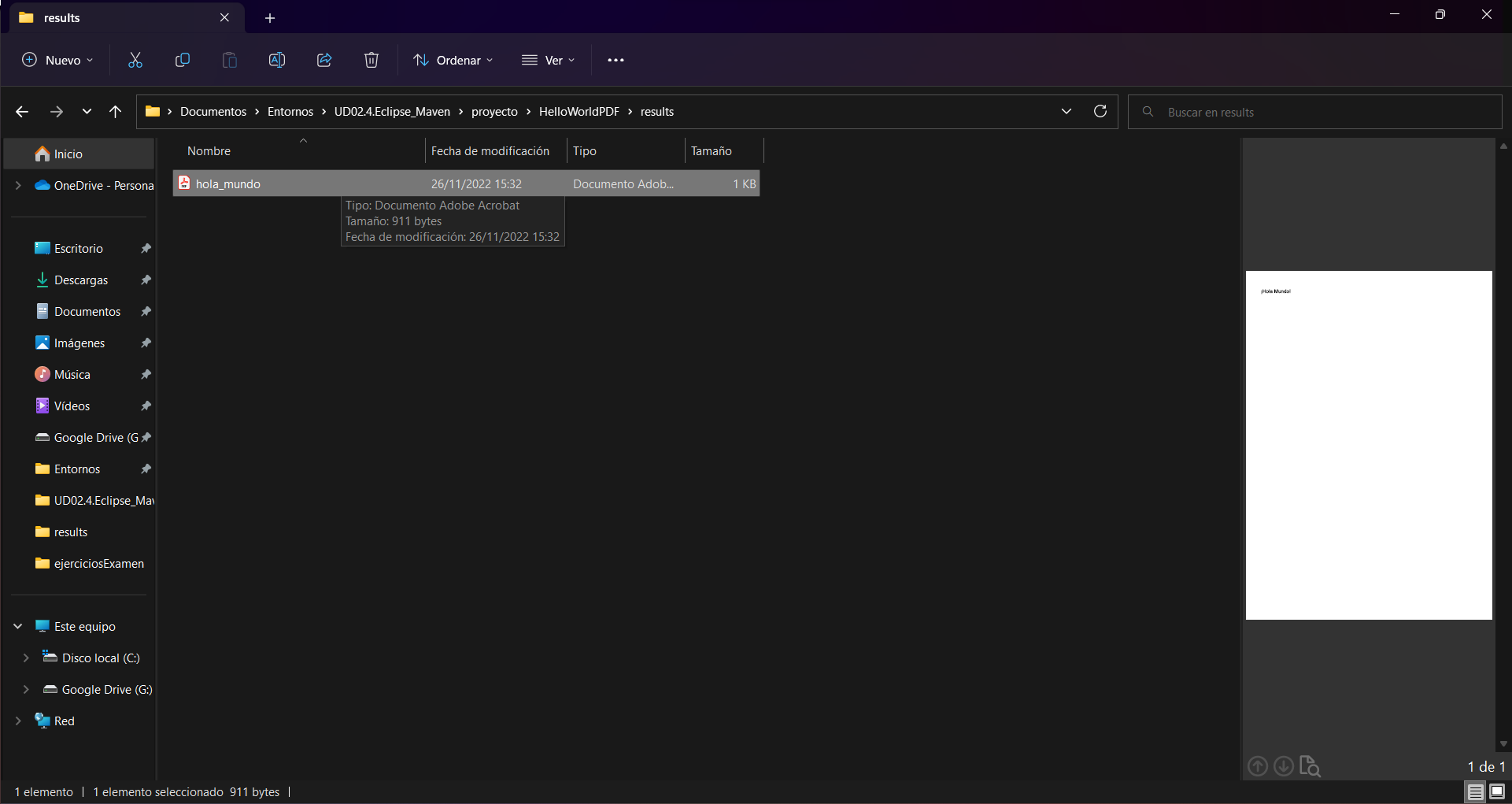


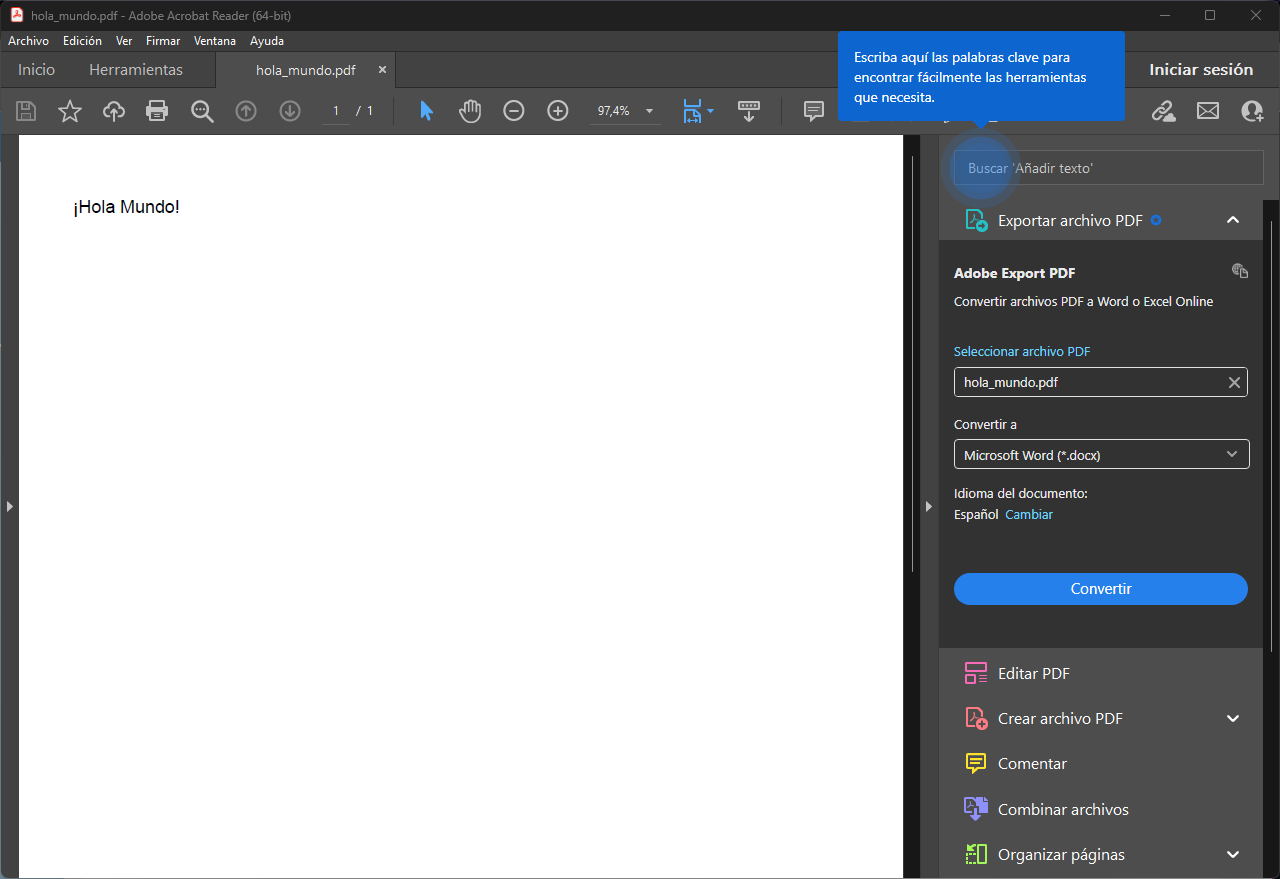
Ahora vamos a agregar los plugins para estas dependencias instaladas. Para eso haremos clic derecho sobre el proyecto y nos iremos a “Maven -> Add Plugin”. Una vez hecho el apartado plugins el archivo pom.xml quedará de la siguiente forma (añadiendo también los apartados para que el proyecto ejecute el archivo correcto en la main class y el plugin shade para evitar errores de ejecución a la hora de la ejecución desde consola):

Actualizaremos el archivo pom.xml y el siguiente paso será crear el código fuente para que se realice la creación del archivo pdf.



Una vez lo ejecutemos podemos observar que se crea el directorio “results” si no estuviera ya creado y se crea el archivo hola\_mundo.pdf con el texto “¡Hola Mundo!”.





Tras esto realizaremos de nuevo los comandos sobre el proyecto al igual que en los pasos anteriores:

#### Compilación

#### Empaquetado

#### Ejecución

#### 

#### 

Comprobamos que se ha creado el archivo:

#### 

#### Instalación

Tras revisar el repositorio local de Maven de nuestro equipo podemos ver que se ha instalado en el equipo correctamente.

