MEDAC
Instituto Oficial de Formación Profesional

Tarea Tema 11

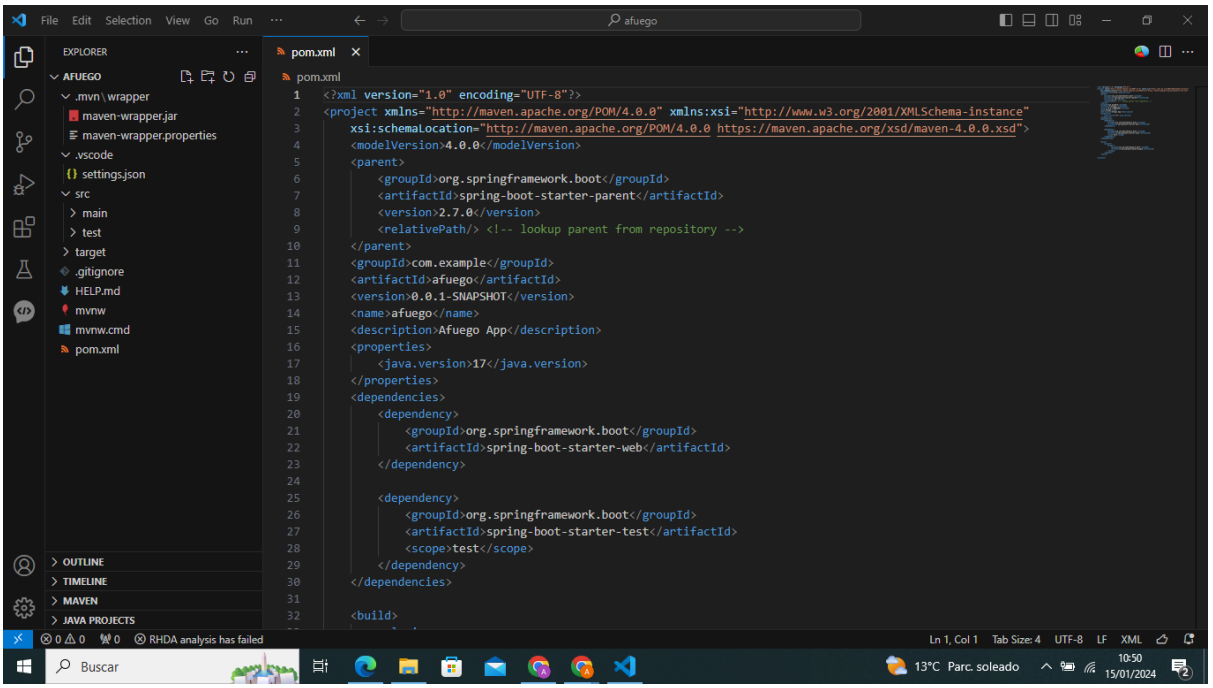
Desarrollo de Interfaces

Andrés Cuevas Rodríguez

1. Generación de Paquete de Instalación desde el IDE:

En este caso, vamos a usar un proyecto Maven de una Web App generado por Spring.

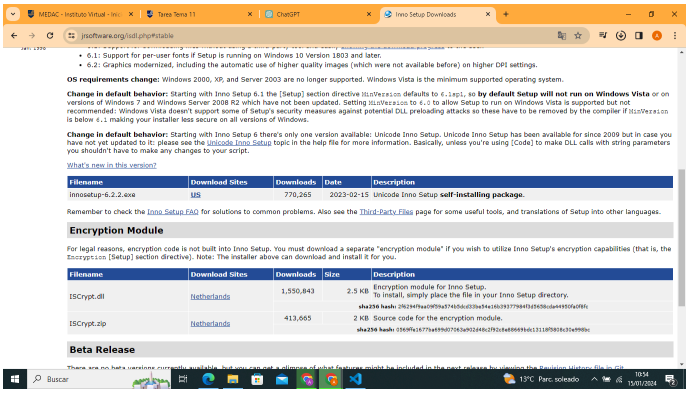
El proyecto fue creado con las dependencias base de Maven desde un archivo “pom.xml”.



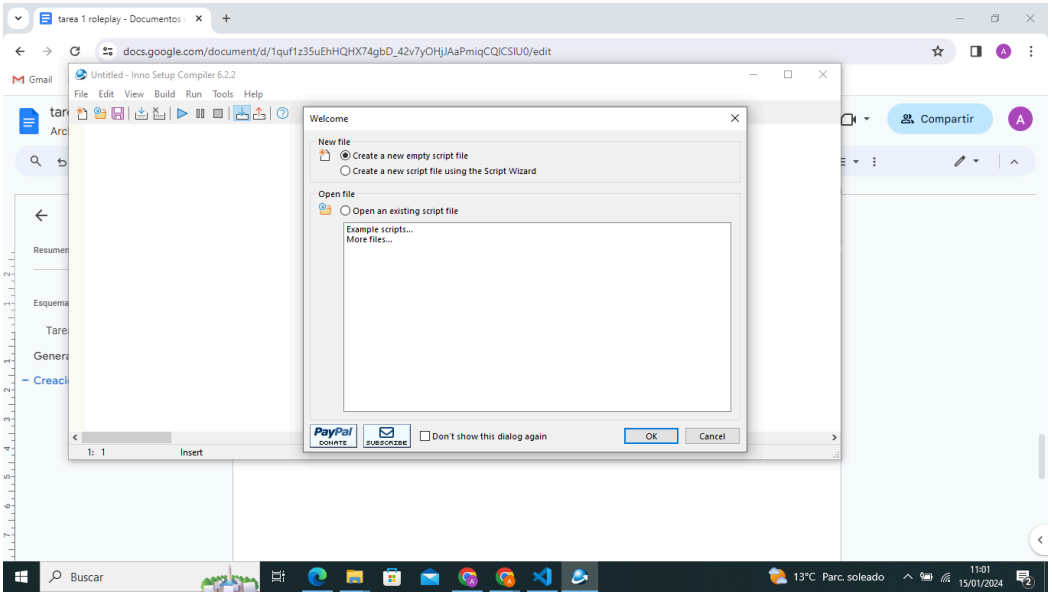
El archivo “pom.xml” será el archivo de paquetes de instalación que tocaremos para realizar la actividad.

2. Creación de Paquete de Instalación con Herramienta Externa:

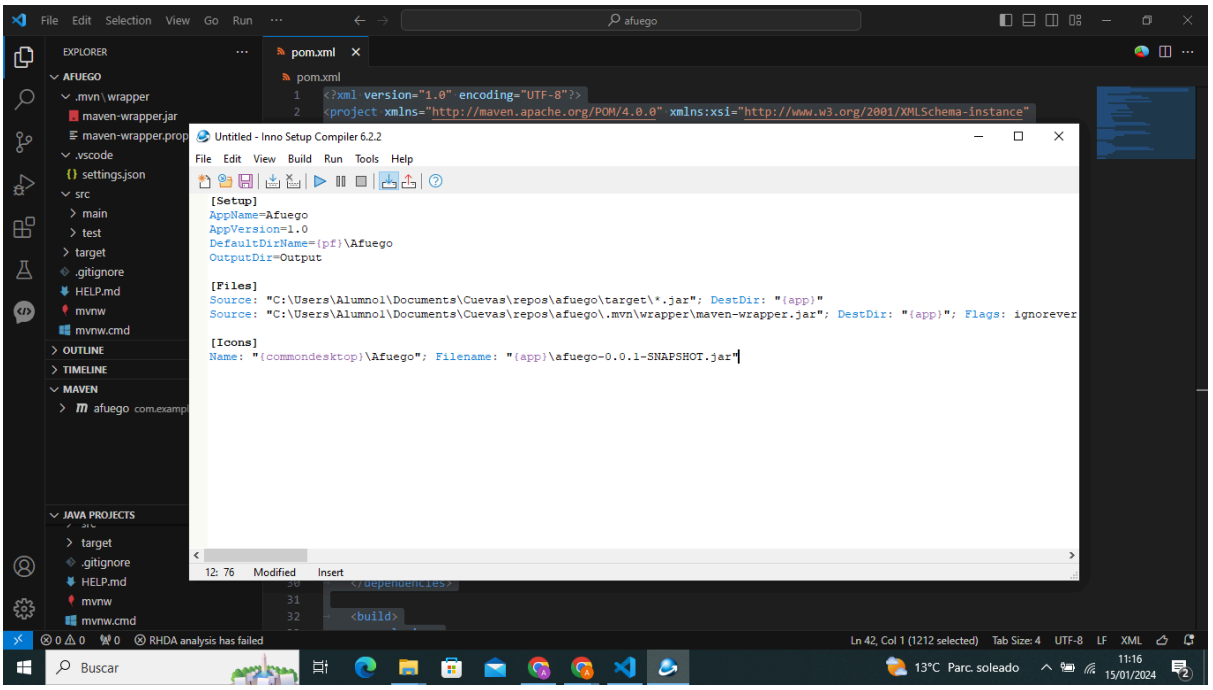
Instalamos la herramienta externa “Inno Setup” desde su página web principal:



Instalamos Inno Setup y creamos un nuevo script:



Añadimos este Script para crear el paquete para la misma aplicación:



Comparación entre Visual Studio Code (IDE) e Inno Setup:

Generación de Paquetes en Visual Studio Code:

Proceso:

Configuración en el POM.xml: La configuración para la generación de paquetes se realiza en el archivo `pom.xml`, donde se especifican detalles del proyecto y dependencias.
Utiliza Maven: Visual Studio Code utiliza Maven para gestionar dependencias y construir el proyecto. El paquete de instalación se genera mediante el comando `mvn clean install`.
Dependencias y Construcción: Maven maneja las dependencias y realiza la construcción del proyecto, generando un archivo JAR que contiene la aplicación.
Integración con el Proyecto: La generación de paquetes está integrada en el proceso de desarrollo del proyecto en Visual Studio Code.

Ventajas de Visual Studio Code (IDE):

- Integración de Desarrollo: La generación de paquetes está integrada en el flujo de desarrollo, facilitando la gestión de dependencias y construcción desde el IDE.
- Automatización con Maven: Maven automatiza tareas como la gestión de dependencias y la construcción del proyecto, simplificando el proceso para el desarrollador.

Desventajas de Visual Studio Code (IDE):

- Limitado a Proyectos Maven: El enfoque en Maven puede limitar la flexibilidad si el proyecto no sigue las convenciones de Maven.

Generación de Paquetes con Inno Setup:

Proceso:

Script de Instalación: Se utiliza un script de Inno Setup para especificar detalles de la instalación, como la ubicación de los archivos y la creación de accesos directos.
Configuración Manual: Es necesario configurar manualmente el script de Inno Setup para indicar los archivos y carpetas a incluir en el paquete de instalación.
Independiente del Proyecto: Inno Setup es una herramienta externa independiente del IDE y del proyecto, especializada en la creación de instaladores para sistemas Windows.

Ventajas de Inno Setup:

- Personalización Total: Permite una personalización total del proceso de instalación, incluyendo ubicaciones de instalación, accesos directos y otros detalles.
- Independencia del IDE: No está vinculado a un IDE específico ni a la estructura de un proyecto particular, lo que brinda flexibilidad.

Desventajas de Inno Setup:

- Configuración Manual: Requiere configuración manual mediante un script, lo que puede ser más propenso a errores y lleva más tiempo.
- No Automatiza Desarrollo: A diferencia de un IDE, no está integrado en el proceso de desarrollo del proyecto y no gestiona dependencias ni construcción de código.

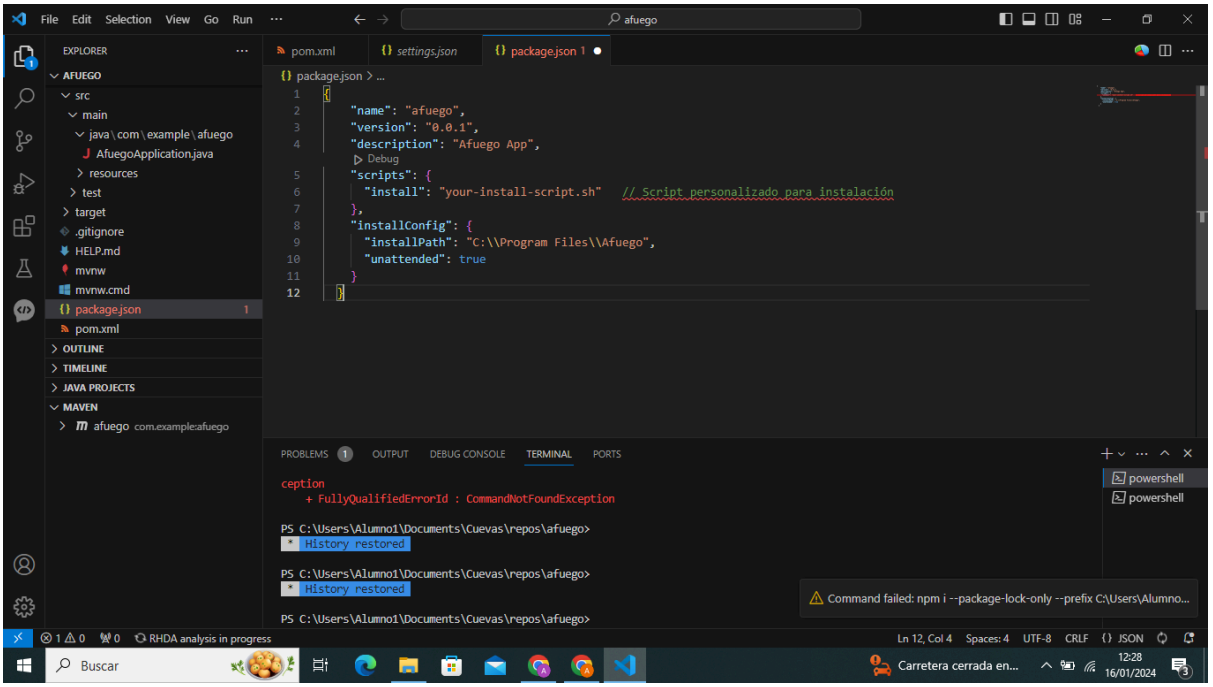
Conclusión:

La elección entre Visual Studio Code (IDE) e Inno Setup depende de las necesidades y preferencias del desarrollador y del proyecto. Visual Studio Code es más adecuado para proyectos que siguen las convenciones de Maven y buscan una integración más estrecha en el flujo de desarrollo. Inno Setup, por otro lado, brinda una personalización total del proceso de instalación y es independiente del entorno de

desarrollo, siendo útil para proyectos que requieren instaladores altamente personalizados.

3. Modo desatendido:

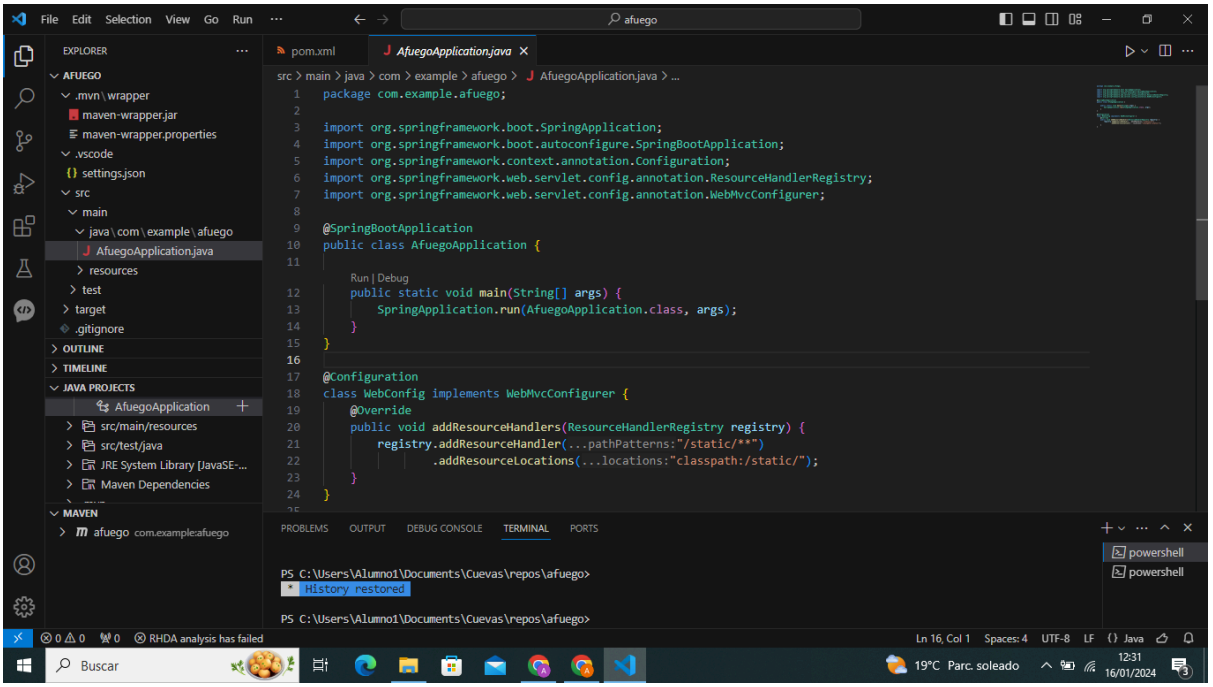
Añadimos el script de instalación en “package.json” generado anteriormente para ejecutar la instalación desatendida.



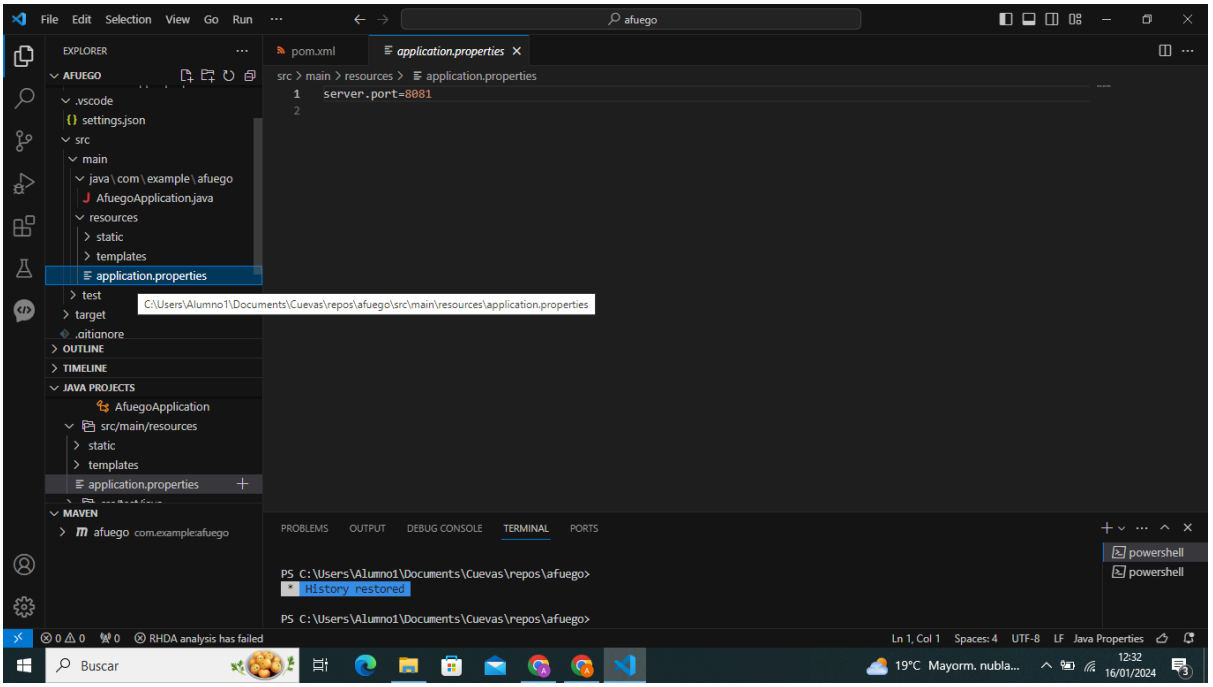
Debuggeamos la aplicación “AfuegoApplication.java” y comenzamos la instalación.

4. Despliegue de la App:

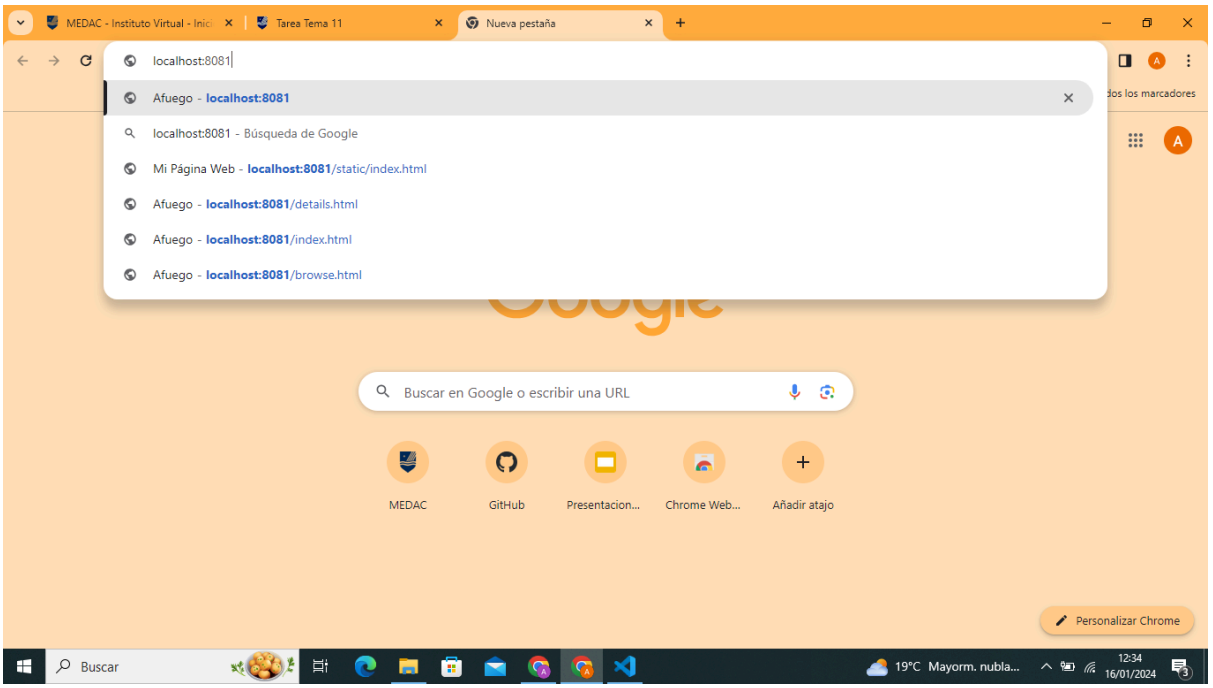
Una vez realizado todos los pasos desplegaremos la app configurando el archivo main.



También configuraremos el archivo “application.properties” para que escuche por el puerto 8081 (decisión personal).



Ponemos “http:localhost:8081” en chrome.



La aplicación está desplegada correctamente.