

UNIVERSIDAD DE GUAYAQUIL
FACULTAD DE CIENCIAS MATEMÁTICAS Y FÍSICAS

Trabajo Tutorización:
Gestión Integrada de Proyectos MAVEN Package

GRUPO 1

Docente:
Ing. Lilia Beatriz Santos Díaz, Msig.

Materia:
Gestión De La Configuración Del Software

Curso:
Sof-S-Ma-9-1

Guayaquil – Ecuador
08 de enero de 2024

TABLA DE CONTENIDOS

Introducción	3
<i>Instalación y Configuración de Maven en el Proyecto</i>	<i>3</i>
<i>Ciclo de Vida del Software en Maven aplicado al Proyecto.....</i>	<i>4</i>
<i>Aplicación de Maven en el Proyecto.....</i>	<i>7</i>
<i>Conclusiones y Recomendaciones.....</i>	<i>8</i>

Introducción

Maven es una herramienta de gestión y comprensión de proyectos de software. En el contexto del proyecto del Segundo Parcial, Maven proporciona un marco de trabajo estandarizado para construir el proyecto, gestionar dependencias y garantizar la calidad del código.

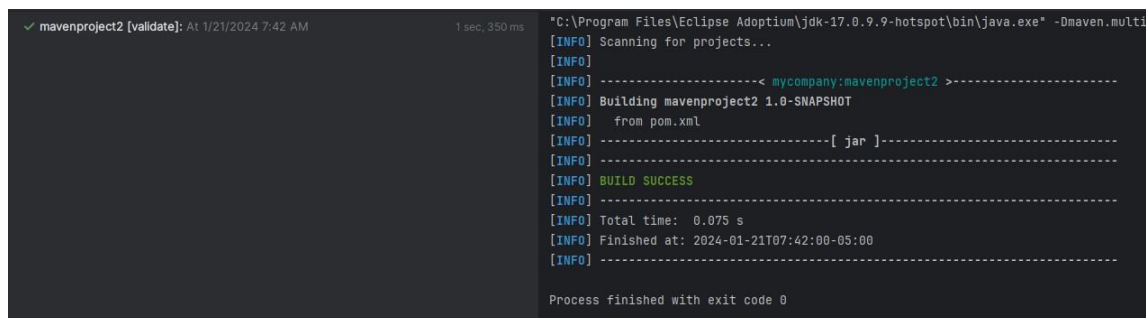
Instalación y Configuración de Maven en el Proyecto

```
14 <dependencies>
15 <dependency>
16 <groupId>com.toedter</groupId>
17 <artifactId>jcalendar</artifactId>
18 <version>1.4</version> <!-- Ajusta la versión según tus necesidades -->
19 </dependency>
20
21 <dependency>
22 <groupId>net.sf.jasperreports</groupId>
23 <artifactId>jasperreports</artifactId>
24 <version>6.17.0</version> <!-- Ajusta la versión según tus necesidades -->
25 </dependency>
26
27 <dependency>
28 <groupId>org.swinglabs</groupId>
29 <artifactId>swing-layout</artifactId>
30 <version>1.0.3</version>
31 </dependency>
32
33 <dependency>
34 <groupId>org.netbeans.external</groupId>
35 <artifactId>AbsoluteLayout</artifactId>
36 <version>RELEASE200</version>
37 </dependency>
```

En el caso del proyecto del Segundo Parcial, Maven ya está integrado con el entorno de desarrollo, lo que simplifica la configuración inicial. El archivo pom.xml incluido en el proyecto es el núcleo de la configuración de Maven, donde se definen las dependencias y los plugins necesarios para la construcción y ejecución del proyecto. Este archivo muestra que se han incluido dependencias específicas como JasperReports y controles de Swing, lo que indica una aplicación con posibles funcionalidades de generación de informes y una interfaz de usuario gráfica.

Ciclo de Vida del Software en Maven aplicado al Proyecto

El proyecto del Segundo Parcial aprovecha el ciclo de vida de Maven para estandarizar su construcción y despliegue. Los pasos del ciclo de vida de Maven que se aplicaron son:

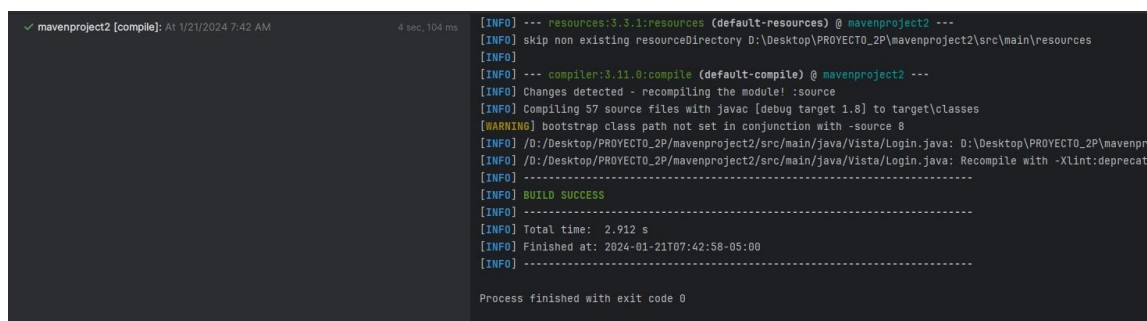


```

✓ mavenproject2 [validate]: At 1/21/2024 7:42 AM 1 sec, 350 ms
[C:\Program Files\Eclipse Adoptium\jdk-17.0.9-hotspot\bin\java.exe" -Dmaven.multi
[INFO] Scanning for projects...
[INFO]
[INFO] -----< mycompany:mavenproject2 >-----
[INFO] Building mavenproject2 1.0-SNAPSHOT
[INFO]    from pom.xml
[INFO] -----[ jar ]-----
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time:  0.075 s
[INFO] Finished at: 2024-01-21T07:42:00-05:00
[INFO]
Process finished with exit code 0

```

Validate: Se muestra en la imagen que el proyecto pasa esta fase, lo que significa que el pom.xml tiene toda la información necesaria y que todas las dependencias necesarias están disponibles o correctamente definidas.

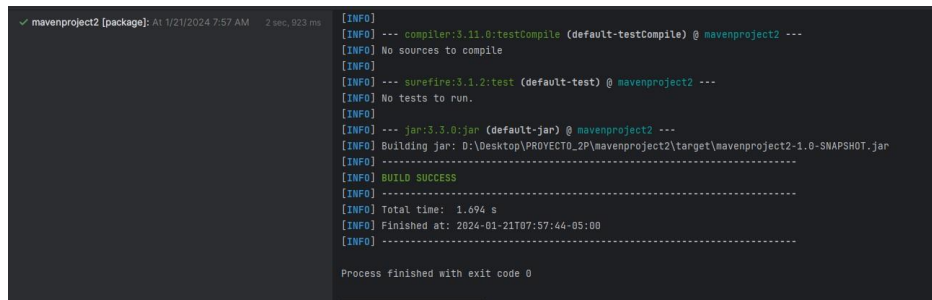


```

✓ mavenproject2 [compile]: At 1/21/2024 7:42 AM 4 sec, 104 ms
[INFO] --- resources:3.3.1:resources (default-resources) @ mavenproject2 ---
[INFO] skip non existing resourceDirectory D:\Desktop\PROYECTO_2P\mavenproject2\src\main\resources
[INFO]
[INFO] --- compiler:3.11.0:compile (default-compile) @ mavenproject2 ---
[INFO] Changes detected - recompiling the module! :source
[INFO] Compiling 57 source files with javac [debug target 1.8] to target\classes
[WARNING] bootstrap class path not set in conjunction with -source 8
[INFO] /D:/Desktop/PROYECTO_2P/mavenproject2/src/main/java/Vista/Login.java: D:\Desktop\PROYECTO_2P\mavenpr
[INFO] /D:/Desktop/PROYECTO_2P/mavenproject2/src/main/java/Vista/Login.java: Recompile with -Xlint:deprecat
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time:  2.912 s
[INFO] Finished at: 2024-01-21T07:42:58-05:00
[INFO]
Process finished with exit code 0

```

Compile: La imagen muestra que el código fuente del proyecto se compila sin errores, lo cual es una indicación de que el código está listo para las fases siguientes como pruebas y empaquetamiento.



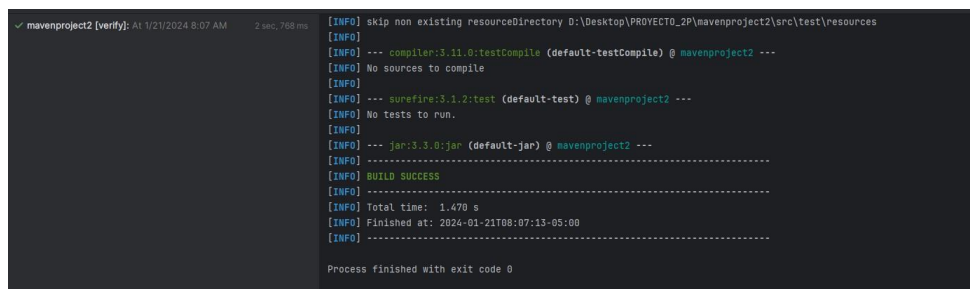
```

✓ mavenproject2 [package]: At 1/21/2024 7:57 AM 2 sec, 923 ms
[INFO] --- compiler:3.11.0:testCompile (default-testCompile) @ mavenproject2 ---
[INFO] No sources to compile
[INFO] --- surefire:3.1.2:test (default-test) @ mavenproject2 ---
[INFO] No tests to run.
[INFO] --- jar:3.3.0:jar (default-jar) @ mavenproject2 ---
[INFO] Building jar: D:\Desktop\PROYECTO_2P\mavenproject2\target\mavenproject2-1.0-SNAPSHOT.jar
[INFO] BUILD SUCCESS
[INFO] Total time: 1.694 s
[INFO] Finished at: 2024-01-21T07:57:44-05:00
[INFO] Process finished with exit code 0

```

Package: La salida muestra Building jar lo que indica que Maven ha empaquetado con éxito el proyecto en un archivo JAR, . El JAR es el resultado final de este proceso y es el artefacto que puede ser ejecutado o desplegado. El mensaje BUILD SUCCESS confirma que el proceso de empaquetado se ha completado sin errores.

En el proyecto del Segundo Parcial, la fase de empaquetado demuestra que el código fuente se ha transformado satisfactoriamente en un artefacto que puede ser distribuido y utilizado. Además, la creación de un JAR ejecutable significa que el proyecto se puede ejecutar en cualquier entorno con una JVM compatible sin necesidad de un IDE específico.



```

✓ mavenproject2 [verify]: At 1/21/2024 8:07 AM 2 sec, 768 ms
[INFO] skip non existing resourceDirectory D:\Desktop\PROYECTO_2P\mavenproject2\src\test\resources
[INFO] --- compiler:3.11.0:testCompile (default-testCompile) @ mavenproject2 ---
[INFO] No sources to compile
[INFO] --- surefire:3.1.2:test (default-test) @ mavenproject2 ---
[INFO] No tests to run.
[INFO] --- jar:3.3.0:jar (default-jar) @ mavenproject2 ---
[INFO] BUILD SUCCESS
[INFO] Total time: 1.470 s
[INFO] Finished at: 2024-01-21T08:07:13-05:00
[INFO] Process finished with exit code 0

```

Verify: La fase de verificación en Maven se realiza después del empaquetado y tiene como objetivo asegurar que cualquier criterio de calidad o prueba de integración se haya pasado antes de que el paquete se instale o despliegue. En el contexto del proyecto del Segundo Parcial, la captura de pantalla muestra que la fase de verificación ha sido completada con éxito. Aunque no se indican pruebas específicas ejecutadas, el BUILD SUCCESS sugiere que todos los pasos previos a esta fase se han cumplido adecuadamente.

```

✓ mavenproject2 [install]: At 1/21/2024 8:07 AM 2 sec, 789 ms
[INFO]
[INFO] --- surefire:3.1.2:test (default-test) @ mavenproject2 ---
[INFO] No tests to run.
[INFO]
[INFO] --- jar:3.3.0:jar (default-jar) @ mavenproject2 ---
[INFO]
[INFO] --- install:3.1.1:install (default-install) @ mavenproject2 ---
[INFO] Installing D:\Desktop\PROYECTO_2P\mavenproject2\pom.xml to C:\Users\Misha\.m2\repository
[INFO] Installing D:\Desktop\PROYECTO_2P\mavenproject2\target\mavenproject2-1.0-SNAPSHOT.jar to
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 1.480 s
[INFO] Finished at: 2024-01-21T08:07:35-05:00
[INFO] -----
Process finished with exit code 0

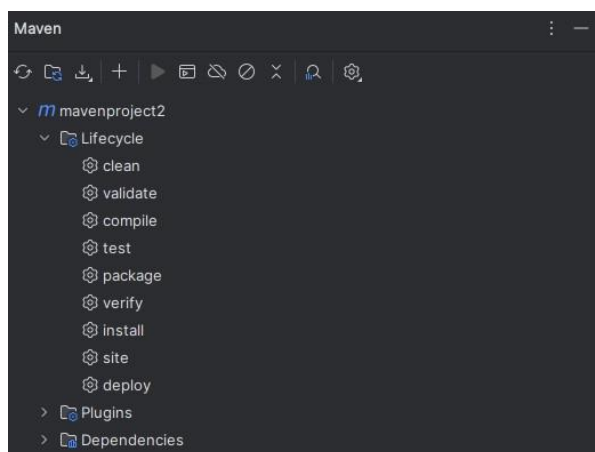
```

Install: La fase de instalación es crítica, ya que coloca el paquete en el repositorio local de Maven. Esto permite que el artefacto sea utilizado como dependencia en otros proyectos locales. imagen muestra que el proyecto del Segundo Parcial ha sido instalado correctamente en el repositorio local de Maven (~/.m2/repository). Este paso es esencial para la reutilización del código y para asegurar la consistencia en el desarrollo colaborativo o cuando se trabaja en múltiples proyectos que comparten dependencias.

Deploy: La fase de deploy es responsable de subir el artefacto final a un repositorio remoto configurado, haciéndolo accesible para otros desarrolladores y proyectos. Esto puede ser un repositorio interno de una empresa o un repositorio público como Maven Central. En el proyecto del Segundo Parcial, esta fase se configuraría para subir el JAR junto con archivos POM y cualquier artefacto adicional al repositorio remoto designado, utilizando las credenciales y configuraciones de seguridad apropiadas.

Aplicación de Maven en el Proyecto

El uso de Maven en el proyecto del Segundo Parcial se demuestra a través de varias capturas de pantalla:



Estructura del Proyecto: La estructura del proyecto reflejada en la imagen muestra una organización típica de Maven, con un directorio src/main/java que contiene el código fuente y un pom.xml en la raíz. Esto indica que el proyecto sigue las convenciones estándar de Maven, lo cual es importante para la integración con herramientas de CI/CD y para que nuevos desarrolladores puedan entender rápidamente la estructura del proyecto.

```
14 <dependencies>
15 <dependency>
16 <groupId>com.toedter</groupId>
17 <artifactId>jcalendar</artifactId>
18 <version>1.4</version> <!-- Ajusta la versión según tus necesidades -->
19 </dependency>
20
21 <dependency>
22 <groupId>net.sf.jasperreports</groupId>
23 <artifactId>jasperreports</artifactId>
24 <version>6.17.0</version> <!-- Ajusta la versión según tus necesidades -->
25 </dependency>
26
27 <dependency>
28 <groupId>org.swinglabs</groupId>
29 <artifactId>swing-layout</artifactId>
30 <version>1.0.3</version>
31 </dependency>
32
33 <dependency>
34 <groupId>org.netbeans.external</groupId>
35 <artifactId>AbsoluteLayout</artifactId>
36 <version>RELEASE200</version>
37 </dependency>
```

Gestión de Dependencias: El pom.xml muestra que el proyecto utiliza varias dependencias de terceros, lo que resalta una de las principales fortalezas de Maven: la gestión eficiente de librerías externas. Esto asegura que se utilicen versiones consistentes y compatibles de cada librería en todo el equipo de desarrollo.

Conclusiones y Recomendaciones

El uso de Maven en el proyecto del Segundo Parcial ha estandarizado y automatizado significativamente el proceso de desarrollo y construcción. Asegura que las dependencias están bien gestionadas y que el proceso de construcción es reproducible, lo cual es esencial para el mantenimiento a largo plazo y la colaboración en el proyecto.

El empaquetado exitoso del proyecto del Segundo Parcial con Maven es un indicador de que el proyecto está bien configurado y listo para avanzar hacia las etapas de verificación y despliegue.

El proyecto del Segundo Parcial hace uso de las fases de install y verify para asegurar que los artefactos del proyecto son construidos y verificados de acuerdo con las expectativas antes de ser utilizados internamente o compartidos externamente. Estas fases son cruciales para la automatización de las entregas y para la integración con procesos de integración y despliegue continuos (CI/CD).