

Migración de Backend - Configuración de Entorno

1 Instalar JDK 17

Descargar e instalar JDK 17 desde Oracle:

👉 <https://www.oracle.com/java/technologies/javase/jdk17-0-13-later-archive-downloads.html>

💡 NOTA:

Se recomienda usar el instalador MSI.

2 Instalar JDK 21

Descargar e instalar JDK 21 desde Oracle:

👉 <https://www.oracle.com/latam/java/technologies/downloads/#jdk21-windows>

💡 NOTA:

Se recomienda usar el instalador MSI.

3 Descargar Apache Maven 3.9.12

Descargar desde el sitio oficial:

👉 <https://maven.apache.org/download.cgi>

😬 NOTA:

No tiene instalador, es un archivo ZIP que debe descomprimirse en la ubicación que prefieras.

4 Eliminar referencias a Java del PATH

Eliminar de las variables de entorno del sistema cualquier referencia a:

```
C:\Program Files\Common Files\Oracle\Java\javapath
```

⚠️ NOTA:

A partir de este momento, si abres una terminal, ya no debería responder el comando `java -version`

5 Agregar JAVA_HOME

Agregar la variable `JAVA_HOME` (si no existe) en las variables de entorno del sistema apuntando al JDK 8 por defecto:

```
JAVA_HOME=C:\Program Files\Java\jdk1.8.0_xxx
```

6 Agregar JAVA_HOME al PATH

Agregar a la variable de sistema `Path` la siguiente entrada:

```
%JAVA_HOME%\bin
```

NOTA:

En una nueva consola ya debería responder el comando `java -version` con la versión 1.8.0_xxx

7 Crear carpeta bin en el perfil de usuario

Crear una carpeta llamada `bin` en tu perfil de usuario:

```
C:\Users\<usuario>\bin
```

8 Crear archivos de comandos Maven

Crear 2 archivos en la nueva carpeta `bin`:

a. Crear archivo mvn17.cmd

Con el siguiente contenido:

```
@echo off
setlocal
set JAVA_HOME=C:\Program Files\Java\jdk-17
set PATH=%JAVA_HOME%\bin;%PATH%
call "C:\Program Files\apache-maven-3.9.12\bin\mvn.cmd" %*
endlocal
```

b. Crear archivo mvn21.cmd

Con el siguiente contenido:

```
@echo off
setlocal
set JAVA_HOME=C:\Program Files\Java\jdk-21
set PATH=%JAVA_HOME%\bin;%PATH%
call "C:\Program Files\apache-maven-3.9.12\bin\mvn.cmd" %*
endlocal
```

9 Agregar carpeta bin al PATH del sistema

Agregar a las variables de entorno del sistema:

```
%USERPROFILE%\bin
```

10 Verificar la configuración

Verificar que todo quedó como esperamos (probar en el CMD de Windows):

✓ El comando `java -version` debería devolver:

```
java version "1.8.0_xxx"
Java(TM) SE Runtime Environment (build 1.8.0_281-b09)
Java HotSpot(TM) 64-Bit Server VM (build 25.281-b09, mixed mode)
```

✓ El comando `mvn -v` debería devolver:

```
Apache Maven 3.8.1 (05c21c65bdfed0f71a2f2ada8b84da59348c4c5d)
Maven home: C:\Program Files\apache-maven-3.8.1\bin\..
Java version: 1.8.0_281, vendor: Oracle Corporation, runtime: C:\Program
Files\Java\jdk1.8.0_xxx\jre
Default locale: es_ES, platform encoding: Cp1252
OS name: "windows 10", version: "10.0", arch: "amd64", family: "windows"
```

✓ El comando `mvn17 -v` debería devolver:

```
Apache Maven 3.9.12 (848fbb4bf2d427b72bdb2471c22fced7ebd9a7a1)
Maven home: C:\Program Files\apache-maven-3.9.12
Java version: 17.0.16, vendor: Oracle Corporation, runtime: C:\Program
Files\Java\jdk-17
Default locale: es_ES, platform encoding: Cp1252
OS name: "windows 10", version: "10.0", arch: "amd64", family: "windows"
```

✓ El comando `mvn21 -v` debería devolver:

```
Apache Maven 3.9.12 (848fbb4bf2d427b72bdb2471c22fced7ebd9a7a1)
Maven home: C:\Program Files\apache-maven-3.9.12
Java version: 21.0.8, vendor: Oracle Corporation, runtime: C:\Program
Files\Java\jdk-21
Default locale: es_ES, platform encoding: UTF-8
OS name: "windows 10", version: "10.0", arch: "amd64", family: "windows"
```

11 Clonar el repositorio del arquetipo

Con todo funcionando, clonar las fuentes del arquetipo desde GitLab:

👉 <https://gitlabgc.gcgestion.com.ar/beyond-edenor/beyond-mvn-archetype>

12 Compilar el arquetipo

Con el proyecto en el entorno local, compilar el arquetipo para que esté disponible en el repositorio local:

```
cd beyond-mvn-archetype  
mvn17 clean install
```

13 ¡Listo para probar!

Ya podríamos juntarnos 🙌