

Manual per a la creació d'un instal·lador

Material:

- Projecte MAVEN
- Launch4j 3.12
- Inno Setup Compiler 6.2.1

Preparació de l'aplicació

La màquina virtual de java no pot executar jars executables que a la seva classe principal continguin elements de javaFX (coses del java).

Per tal d'evitar aquest error farem una petita trampa:

Crearem una classe llançadora de la nostra aplicació que ha de quedar semblant a aquesta:

```
public class MainLauncher {  
    public static void main (String [] args) {  
        App.main(args);  
    }  
}
```

On App.main serà el main de la nostra aplicació. A partir d'aquí executarem sempre la classe llançador en comptes de la nostra aplicació. També la referenciem en el nostre pom per tal que la compil·li i l'empaqueti correctament.

Creació de JAR executable

Obrir el POM del nostre projecte i fer les següents modificacions:

Afegir el següent plugin:

<!-- Plugin per afegir dependències al jar, s'ha de crear una carpeta al projecte anomenada libs -->

```
<plugin>  
    <groupId>org.apache.maven.plugins</groupId>  
    <artifactId>maven-dependency-plugin</artifactId>  
    <executions>  
        <execution>  
            <id>copy-dependencies</id>  
            <phase>prepare-package</phase>  
            <goals>  
                <goal>copy-dependencies</goal>  
            </goals>  
            <configuration>  
                <outputDirectory>  
                    ${project.build.directory}/libs  
                </outputDirectory>  
            </configuration>  
        </execution>  
    </executions>
```

```
</plugin>
```

Modifiquem el següent plugin:

```
<plugin>
  <artifactId>maven-assembly-plugin</artifactId>
  <executions>
    <execution>
      <phase>package</phase>
      <goals>
        <goal>single</goal>
      </goals>
      <configuration>
        <archive>
          <manifest>
            <mainClass>Aquí posarem la nostra classe llançador</mainClass>
          </manifest>
        </archive>
        <descriptorRefs>
          <descriptorRef>jar-with-dependencies</descriptorRef>
        </descriptorRefs>
      </configuration>
    </execution>
  </executions>
</plugin>
```

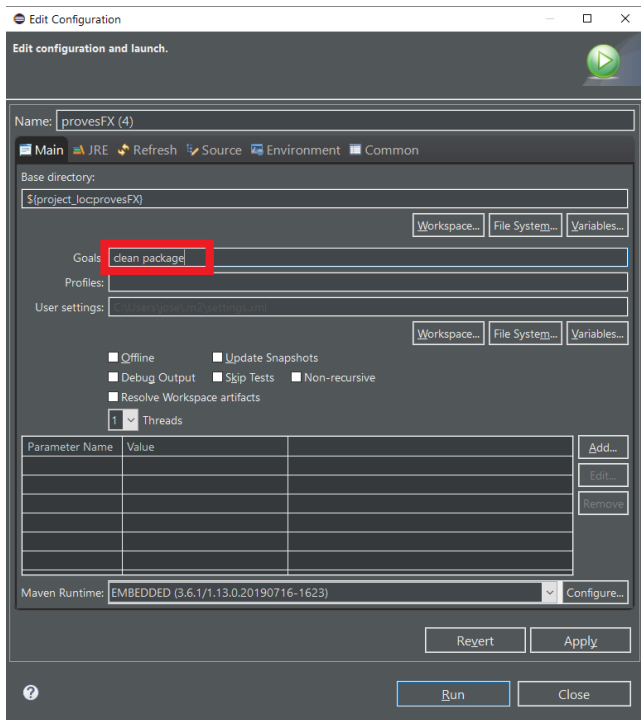
Afegim la següent línia al principi del POM:

```
<groupId>org.openjfx</groupId>
<artifactId>provesFX</artifactId>
<version>0.0.1-SNAPSHOT</version>
<packaging>jar</packaging> <!-- Afegir aquesta línia per crear el jar -->
```

Fem un update del projecte.

Anem a Run as → Maven Build...

A l'apartat goals posem **clean package** i executem amb el botó run.



Si tot ha sortit bé ens ha de sortir a la consola el següent missatge:

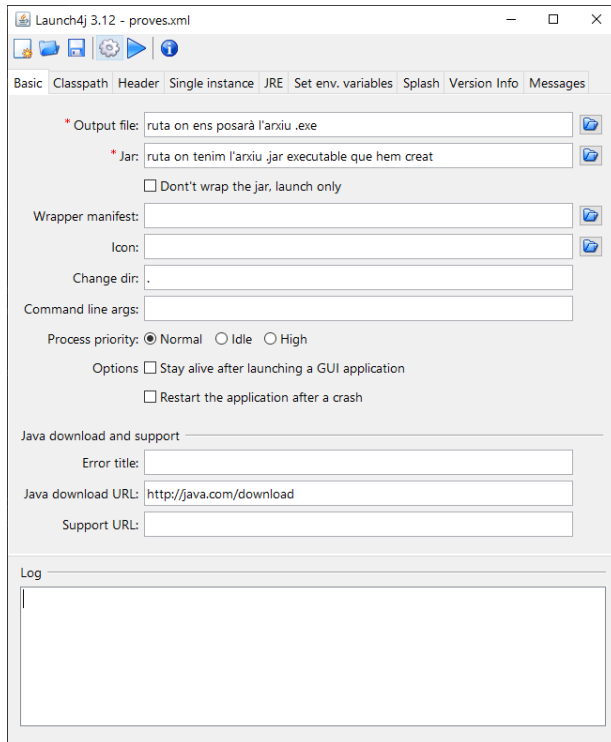
```
[INFO] -----  
[INFO] BUILD SUCCESS  
[INFO] -----  
[INFO] Total time: 6.289 s  
[INFO] Finished at: 2022-04-23T08:16:27+02:00  
[INFO] -----
```

Se'ns ha creat el nostre jar executable i ho ha deixat a la carpeta target del nostre projecte. Comprovem que funciona correctament, perquè funcioni hem de tenir instal·lat la màquina virtual de java al nostre equip.

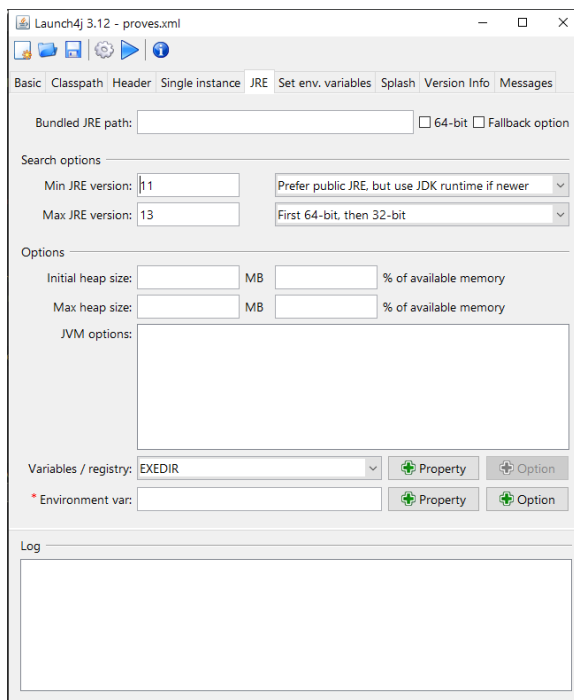
Creació de l'executable exe

Executem launch4j:

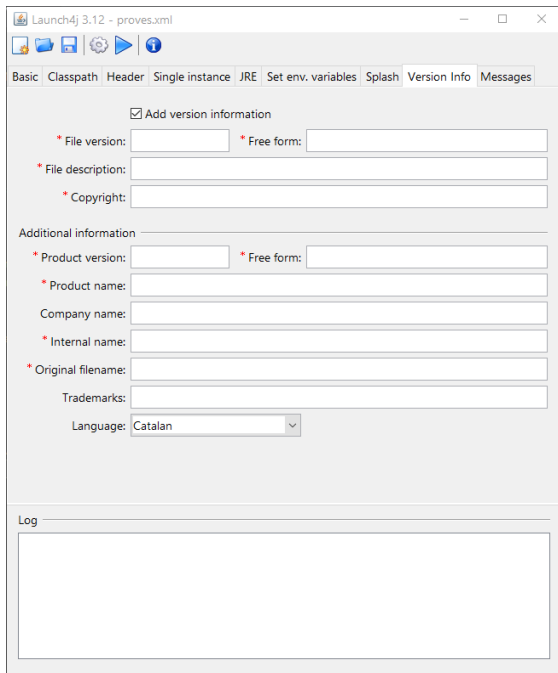
A la pestanya Basic omplim la ruta del nostre jar i la ruta on volem que guardi l'executable



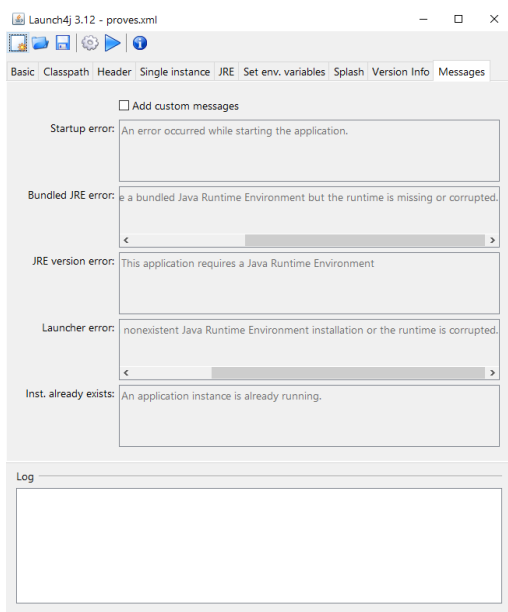
A la pestanya JRE omplim la versió de java mínima i màxima que accepta el nostre programa.



A la pestanya Version info podem posar les dades de la nostra empresa, del nostre programa, etc.



A la pestanya Messages podem customitzar els missatges d'error del nostre programa



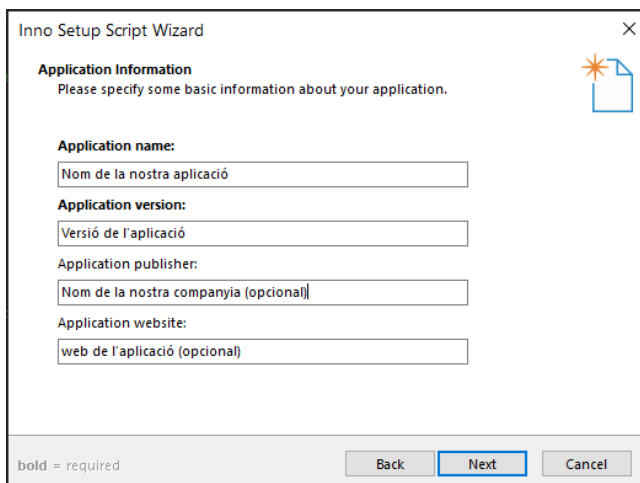
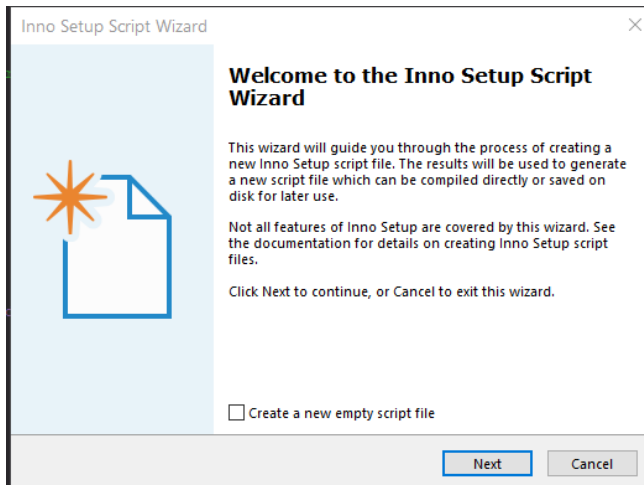
Un cop entrades totes les dades podem clicar el botó Build wrapper i ens crearà l'executable.



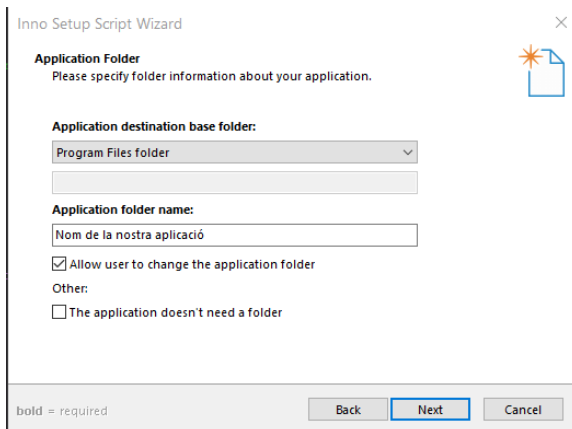
Un cop tenim l'executable, creem una carpeta i fem a dintre el nostre executable, i tots els arxius necessaris per al funcionament d'aquesta.

- Instal·lador de Mysql (si cal)
- script de creació de la base de dades.
- script d'introducció de dades bàsiques per a la nostra aplicació.

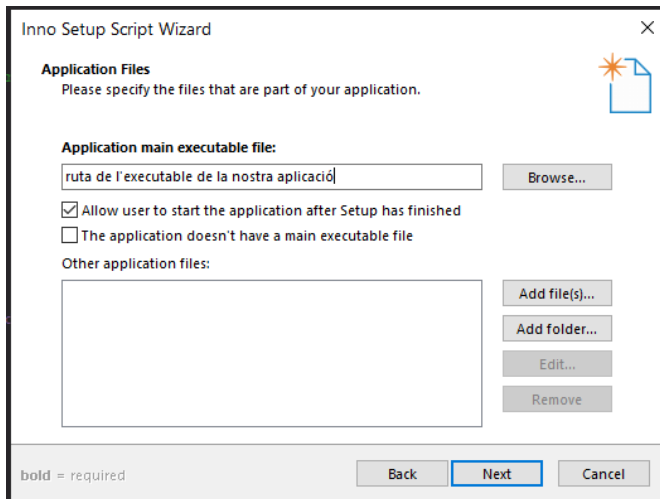
Obrim Inno setup, creem un nou arxiu i ens anirà guiant en la creació del nostre script:



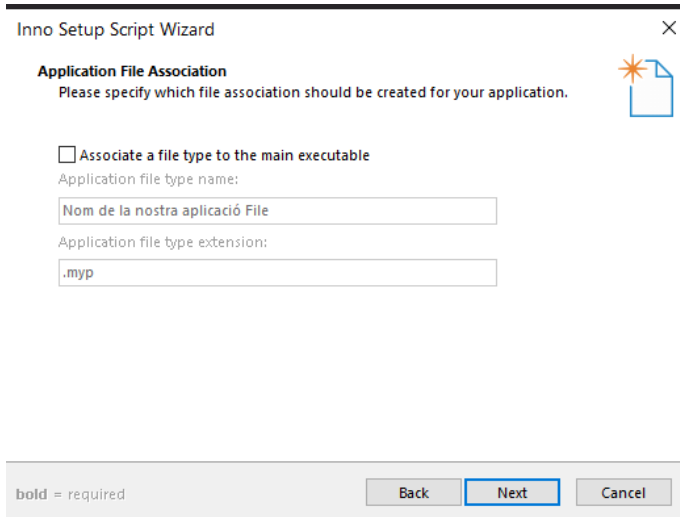
Indiquem on s'instal·larà el nostre programa:



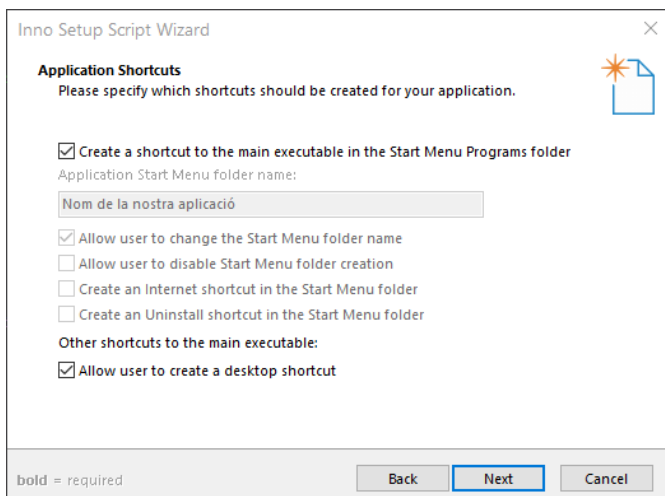
Indiquem on està l'executable:



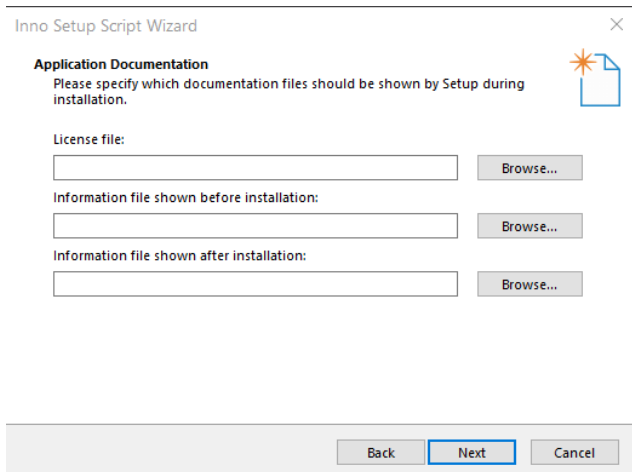
Associem extensions al nostre programa si cal:



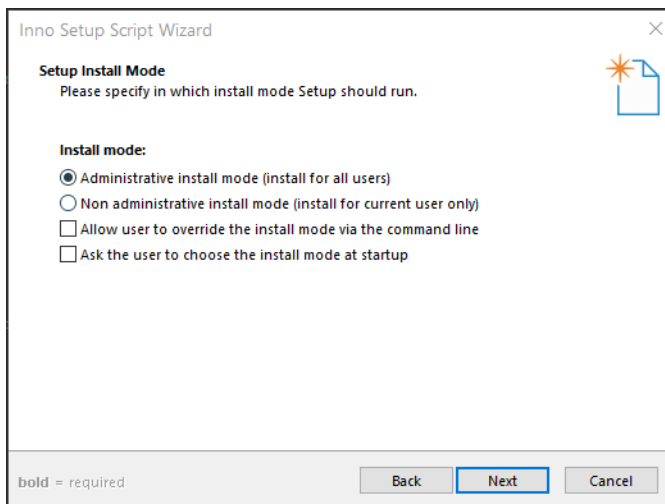
Indiquem opcions varies per la creació d'entrades al menú d'inici, etc...



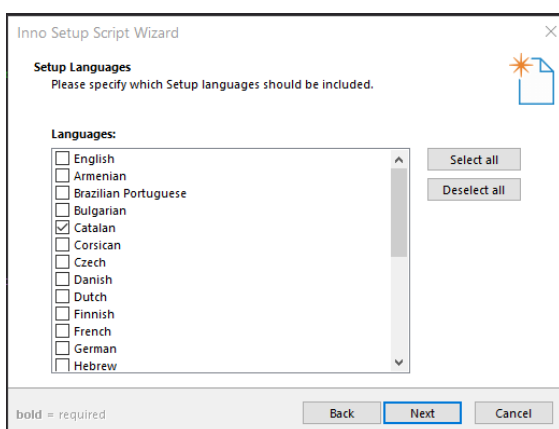
Posem la informació de la llicència (si cal)



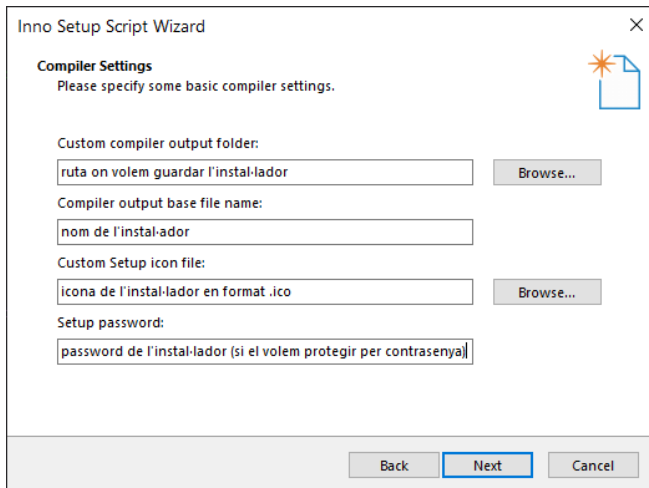
Indiquem si ha de tenir permisos d'administrador:



Indiquem l'idioma de l'instal·lador, serà l'idioma que li sortirà a l'usuari quan faci la instal·lació.



Per últim, indiquem dades de l'arxiu instal·lador



Ens crearem un script que podrem modificar:

```

#define MyAppName "Nom de la nostra aplicació"
#define MyAppVersion "Versió de l'aplicació"
#define MyAppPublisher "Nom de la nostra companyia (opcional)"
#define MyAppURL "web de l'aplicació (opcional)"
#define MyAppExeName "ruta de l'executable de la nostra aplicació"

[Setup]
; NOTE: The value of AppId uniquely identifies this application. Do not use the same AppId value in installers for other applications.
; (To generate a new GUID, click Tools | Generate GUID inside the IDE.)
AppId={A0807E2A-8DBF-41D9-AED5-E3AD60166F33}
AppName={MyAppName}
AppVersion={MyAppVersion}
AppVerNames={MyAppName} {MyAppVersion}
AppPublisher={MyAppPublisher}
AppPublisherURL={MyAppURL}
AppSupportURL={MyAppURL}
AppUpdatesURL={MyAppURL}
DefaultDirName={autopf}\{MyAppName}
DisableProgramGroupPage=yes
; Uncomment the following line to run in non administrative install mode (install for current user only.)
; PrivilegesRequired=lowest
OutputDir=ruta on volem guardar l'instal·lador
OutputBaseFilename=nom de l'instal·lador
SetupIconFile=icona de l'instal·lador en format .ico
Password=password de l'instal·lador (si el volem protegir per contrasenya)
Compression=lzma
SolidCompression=yes
WizardStyle=modern

[Languages]
Name: "catalan"; MessagesFile: "compiler:Languages\Catalan.isl"

[Tasks]
Name: "desktopicon"; Description: "{cm:CreateDesktopIcon}"; GroupDescription: "{cm:AdditionalIcons}"; Flags: unchecked

[Files]
Source: "{MyAppExeName}"; DestDir: "{app}"; Flags: ignoreversion
; NOTE: Don't use "Flags: ignoreversion" on any shared system files

[Icons]
Name: "{autoprogams}\{MyAppName}"; Filename: "{app}\{MyAppExeName}"
Name: "{autodesktop}\{MyAppName}"; Filename: "{app}\{MyAppExeName}"; Tasks: desktopicon

[Run]
Filename: "{app}\{MyAppExeName}"; Description: "{cm:LaunchProgram, {StringChange MyAppName, '{', '{}}"; Flags: shellexec postinstall skipifsilent

```

Haurem de modificar els apartats:

- Files, on afegirem la resta d'arxius que necessitem per la instal·lació.
- Run, on executarem les comandes que necessitem per completar la instal·lació