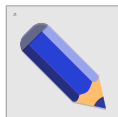


Proba de Desenvolvemento de Interfaces: UD7

7.1 - CA2.1:	7.2 - CA2.2:	7.3 - CA2.3:	7.4 - CA2.4:	7.5 - CA2.5:
10% (ME)	20%	10% (ME)	20% (ME)	10%



Apellidos, Nombre: Amado Cibreiro, Andrés

Fecha: 17/03/2025

Unidad 7: Distribución de aplicaciones

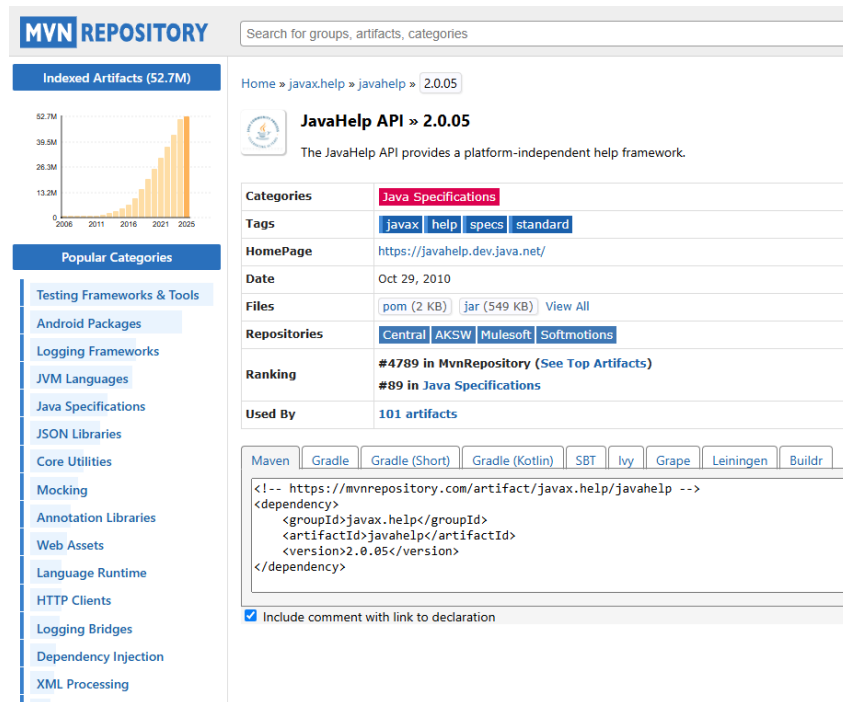
CA2.1 - Empaquetáronse os compoñentes que require a aplicación.

7.1 El nuevo desarrollador que se ha incorporado a tu equipo de trabajo no tiene conocimientos de NetBeans. En la empresa cliente le acaban de entregar el proyecto denominado TIC de NetBeans y le han pedido que genere in-situ el distribuible TIC.jar de la aplicación. En problema con el que se encuentra el compañero es que el proyecto TIC depende de la librería “javahelp-2.0.05.jar” que no está instalada en el sistema operativo ni descargada. Indica el procedimiento paso a paso que debe seguir el nuevo desarrollador para crear el distribuible TIC.jar empleando NetBeans. [Incluye la respuesta en este apartado]

Para descargar la librería indicada debemos ir al siguiente enlace:

<https://mvnrepository.com/artifact/javahelp/javahelp/2.0.05>.

Una vez en la web debemos ver algo similar a esto: En esta web debemos darle click en el apartado Files al fichero jar que en este caso pesa 549 KB y se nos descargará el archivo que queremos.



The screenshot shows the Maven Repository website. The main content area displays the 'JavaHelp API > 2.0.05' page. The page includes a search bar at the top, a sidebar with 'Indexed Artifacts (52.7M)' and 'Popular Categories', and a main content area with details about the artifact. The 'Files' section shows two files: 'pom (2 KB)' and 'jar (549 KB)'. The 'Ranking' section shows the artifact is #4789 in Maven Repository and #89 in Java Specifications. The 'Used By' section shows 101 artifacts. The 'Files' section also includes a 'View All' link. The 'Files' section also includes a 'View All' link.

Categories Java Specifications

Tags javax help specs standard

HomePage <https://javahelp.dev.java.net/>

Date Oct 29, 2010

Files [pom \(2 KB\)](#) [jar \(549 KB\)](#) [View All](#)

Repositories [Central](#) [AKSW](#) [Mulesoft](#) [Softmotions](#)

Ranking #4789 in MvnRepository (See Top Artifacts)
#89 in Java Specifications

Used By 101 artifacts

Files [Maven](#) [Gradle](#) [Gradle \(Short\)](#) [Gradle \(Kotlin\)](#) [SBT](#) [Ivy](#) [Grape](#) [Leiningen](#) [Buildr](#)

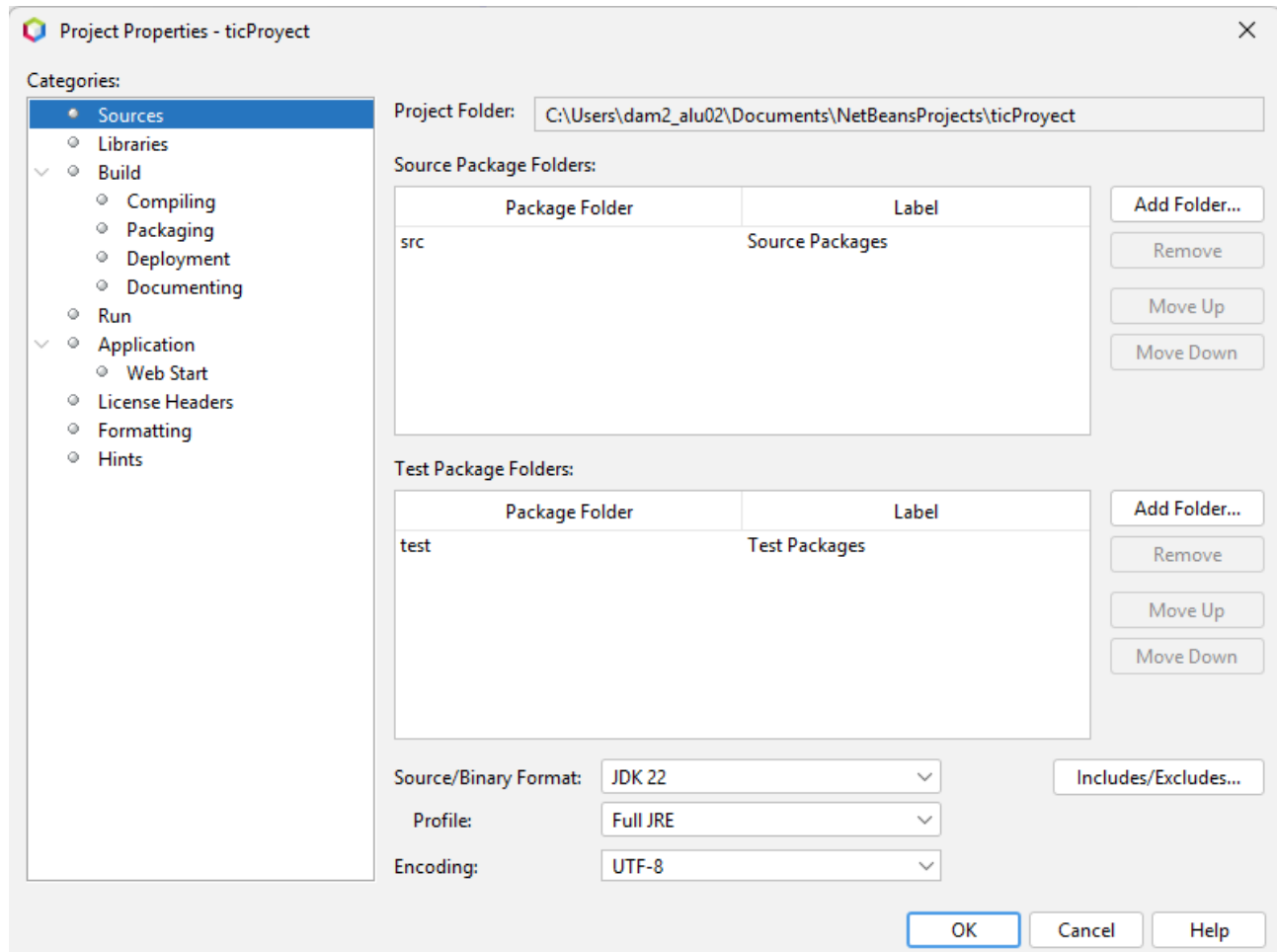
```
<!-- https://mvnrepository.com/artifact/javahelp/javahelp -->
<dependency>
  <groupId>javax.help</groupId>
  <artifactId>javahelp</artifactId>
  <version>2.0.05</version>
</dependency>
```

☒ Include comment with link to declaration

Para poder usarlo en el proyecto debemos ir a la carpeta del proyecto de NetBeans.

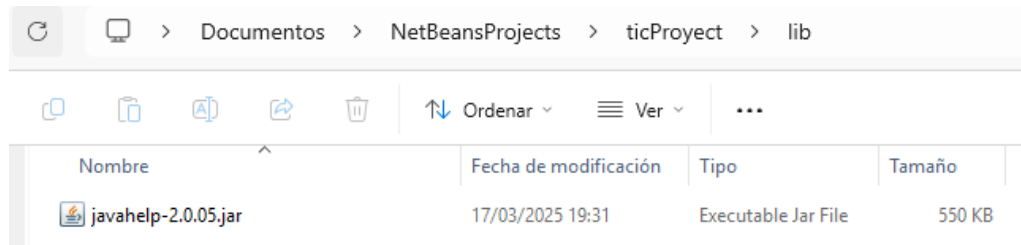
Si no sabemos en dónde está localizado el proyecto podemos ir en la pestaña Projects que suele estar en la izquierda de la pantalla y darle click derecho al proyecto y entrar en las propiedades (Properties).

Aquí veremos en la parte superior una línea de texto que dice Project Folder en donde veremos la ruta completa.



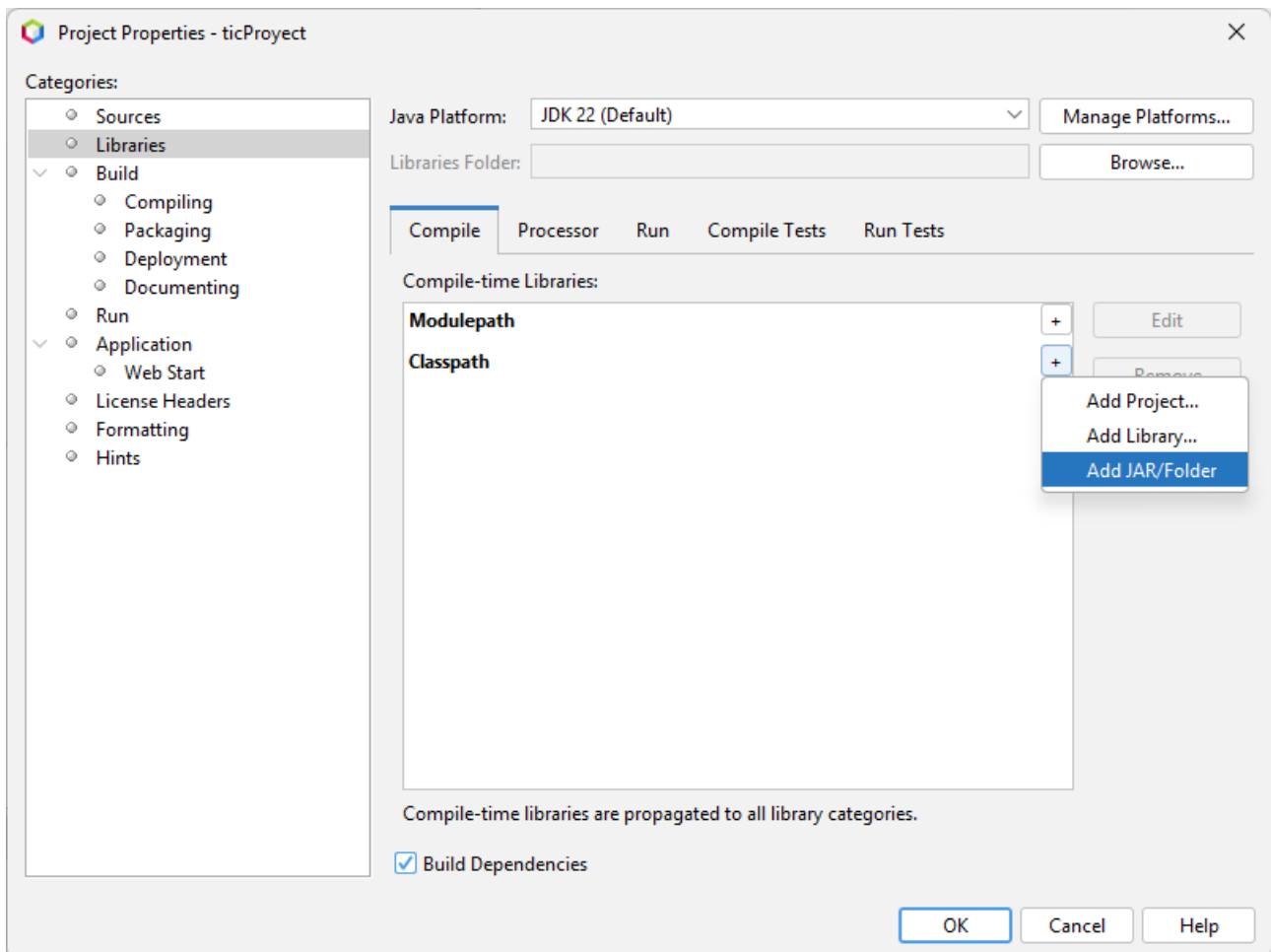
Ahora en el explorador de archivos de nuestro sistema operativo debemos ir a la carpeta origen de nuestro proyecto, ya una vez localizada. En esta crearemos una subcarpeta llamada lib dentro de esta carpeta debemos copiar el archivo descargado.

La ruta del fichero terminaría siendo la siguiente: /lib/javahelp-2.0.05.jar



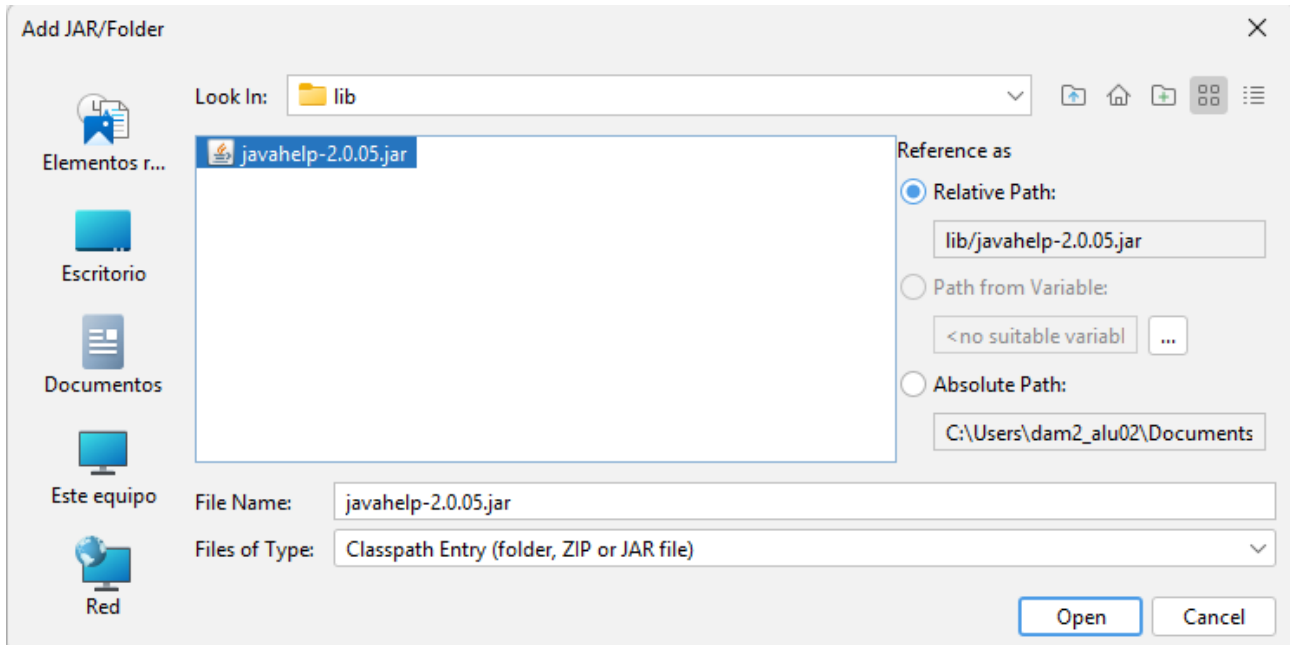
Aun que tengamos la librería dentro de la carpeta indicada todavía debemos añadirla como tal al proyecto desde NetBeans. Para esto iremos a Properties dando click derecho en el proyecto.

En el menú izquierdo nos desplazaremos a Libraries. A continuación debemos darle Donde indica Classpath al botón “+” para añadir una librería. Esto nos desplegará un menú y debemos darle a “Add JAR/Folder”

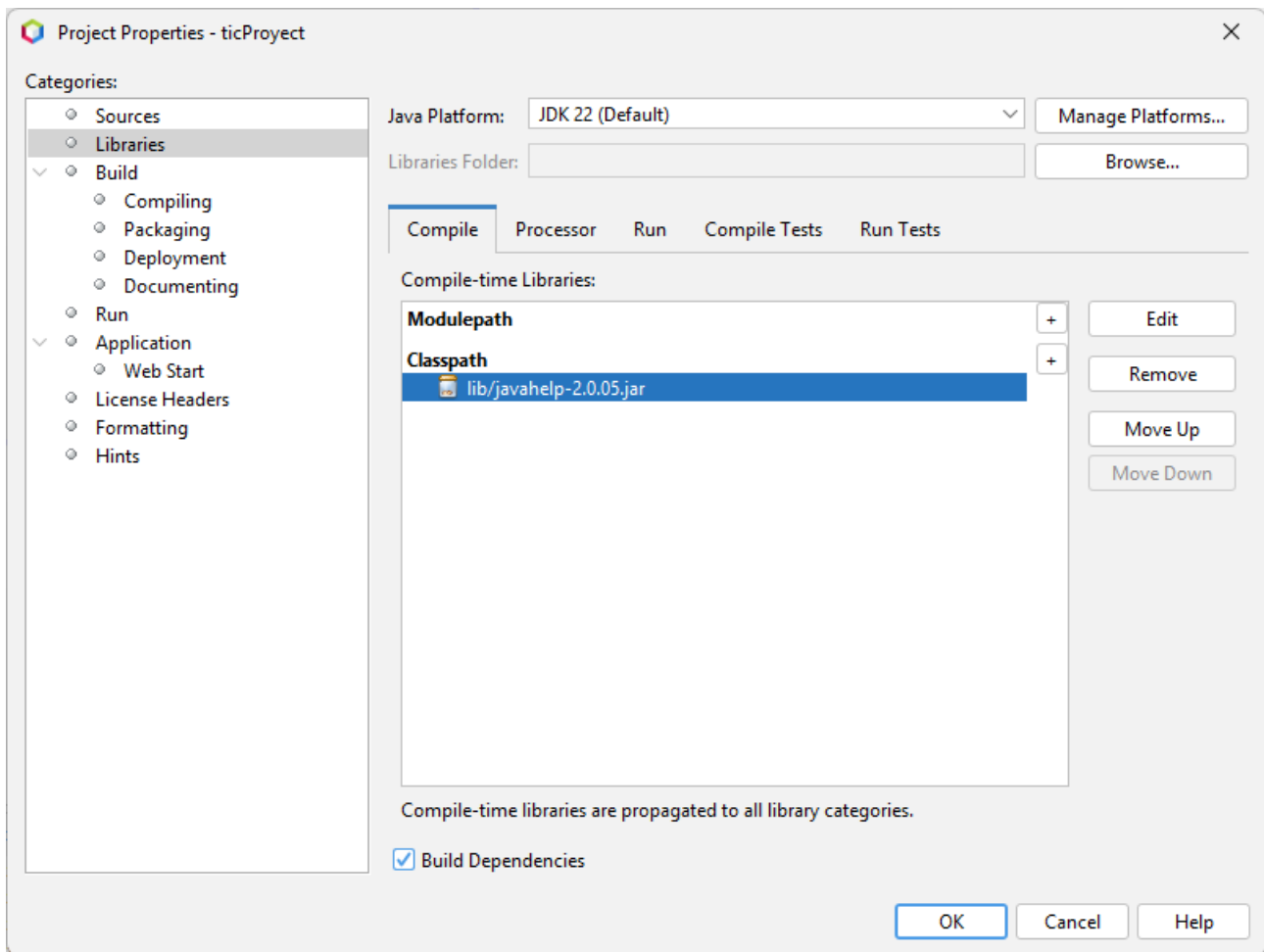


Ahora veremos un seleccionador de fichero y debemos desplazarnos hasta la carpeta lib que creamos anteriormente y seleccionar nuestra librería javahelp-2.0.05.jar.

Muy importante, en esta ventana debemos marcar la opción de “Relative path” como se indica en la siguiente imagen:



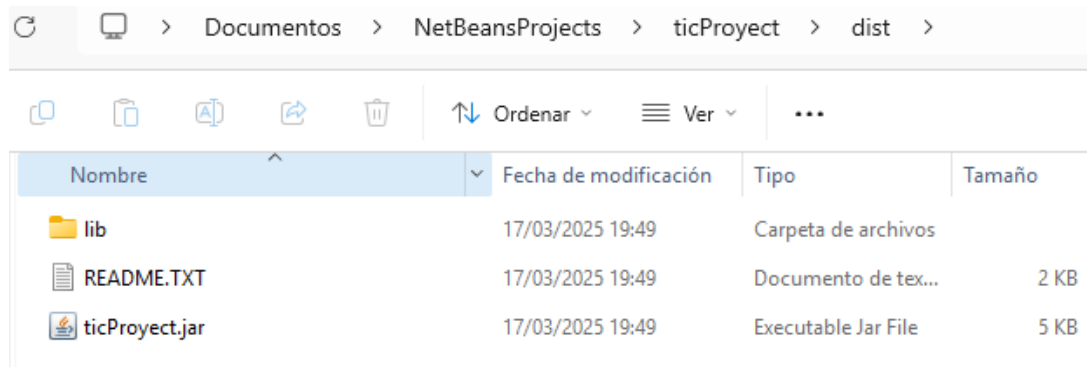
Una vez añadida la librería deberíamos ver el apartado Libraries de Properties así:



Ahora para realizar el jar de nuestra aplicación con la librería debemos hacer lo siguiente:

Dentro de NetBeans debemos darle click derecho en la ventana Projects a nuestro proyecto y le debemos dar a la opción Clean and Build.

Esto generará dentro de nuestro proyecto la carpeta /dist la que incluye el archivo .jar, un README.txt. Debemos hacer una copia de la carpeta lib carpeta lib



CA2.2 - Personalizouse o asistente de instalación.

7.2 Personaliza el siguiente script de NSIS para que la aplicación se instale necesariamente en el escritorio del usuario:

```
Section "Demo" SecConverter
  SetOutPath $INSTDIR
  File Demo.jar
  WriteUninstaller "$INSTDIR\Uninstall.exe"
SectionEnd
```

Script personalizado:

```
Section "Demo" SecConverter
  SetOutPath $DESKTOP
  File Demo.jar
  WriteUninstaller "$DESKTOP\Uninstall.exe"
SectionEnd
```

Simplemente debemos de cambiar \$INSTDIR por \$DESKTOP

CA2.3 Empaquetouse a aplicación para ser instalada de xeito típico, completo ou personalizado.

7.3 Dado el siguiente fragmento procedente de un archivo de instalación NSIS, indica qué es lo que haría cada una de las líneas del script:

```
Section "Converter" SecConverter
    SetOutPath $INSTDIR
    File Converter.jar
    SetOutPath $INSTDIR\lib
    File lib\javahelp-2.0.05.jar
    SetOutPath $INSTDIR\help
    File /nonfatal /a /r "help\"
    WriteUninstaller "$INSTDIR\Uninstall.exe"
SectionEnd
```

```
Section "ConverterReadme" SecConverterReadme
    SetOutPath $DESKTOP
    File "Converter Readme"
SectionEnd
```

1	Section "Converter" SecConverter	/* Crea una sección con nombre SecConverter */
2	SetOutPath \$INSTDIR	/* Indica el directorio de creación de los ficheros */
3	File Converter.jar	/* Crea el fichero Converter.jar en la ruta indicada previamente */
4	SetOutPath \$INSTDIR\lib	/* Establece el directorio /lib donde se crearán las librerías */
5	File lib\javahelp-2.0.05.jar	/* Crea la librería de Javahelp en lib/ */
6	SetOutPath \$INSTDIR\help	/* Establece el directorio /help donde se añadirá la ayuda */
7	File /nonfatal /a /r "help\"	/* /none fatal indica que el archivo podría no existir
8		/a indica que se copiarán los atributos del fichero a la copia
9		/r se buscará la carpeta "help" también en subdirectorios*/
10	WriteUninstaller "\$INSTDIR\Uninstall.exe"	/* Crea el desinstalador en la ruta origen llamado Uninstall.exe */
11	SectionEnd	/* Final de la sección SecConverter*/
12		
13		
14	Section "ConverterReadme" SecConverterReadme	/* Crea la sección SecConverterReadme */
15	SetOutPath \$DESKTOP	/* Indica la ruta del escritorio */
16	File "Converter Readme"	/* Añade el archivo que se llame "Converter Readme" sin importar la extensión */
17	SectionEnd	/* Final de la sección SecConverterReadme */

CA2.4 - Xeráronse paquetes de instalación utilizando o contorno de desenvolvemento.

7.4 Explica brevemente el procedimiento para generar un paquete instalable (instalador) de una aplicación desarrollada en Java Swing utilizando el entorno de desarrollo NetBeans. Indica además, al menos una herramienta externa que podría ser empleadas en este proceso si se desea distribuir la aplicación en forma de instalador para sistemas Windows.

Haz clic derecho sobre el proyecto selecciona "Clean and Build". Esto generará los archivos necesarios en la carpeta `dist` del proyecto junto al `.jar`

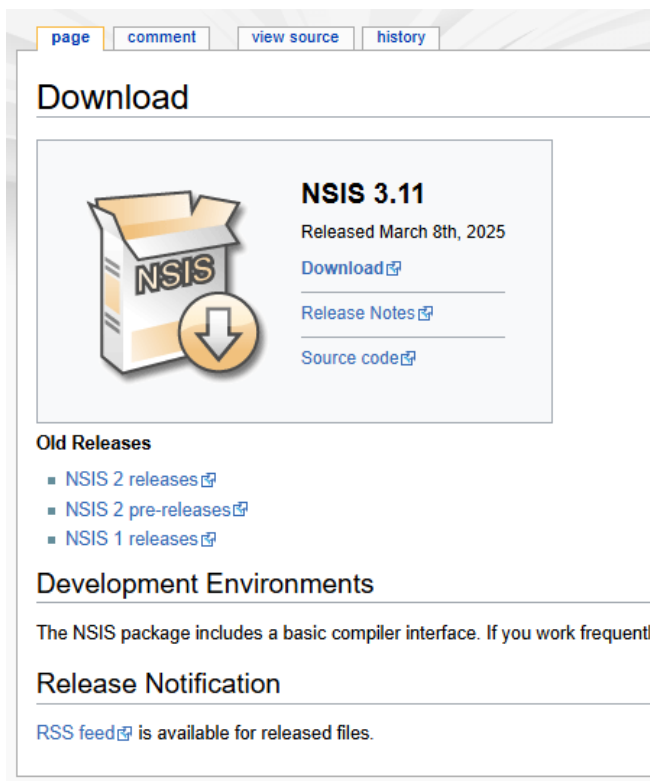
El instalador como tal no lo podemos generar, así que precisaremos de una herramienta externa como NSIS

CA2.5 Xeráronse paquetes de instalación utilizando ferramentas externas.

7.5 El nuevo desarrollador que se ha incorporado a tu equipo de trabajo no tiene conocimientos de NSIS. En la empresa cliente le acaban de entregar el proyecto denominado TIC que contiene, además del archivo TIC.jar y sus dependencias, el archivo tic.nsis que permite generar el instalador TIC.exe. Al nuevo desarrollador le han pedido que genere in-situ el archivo TIC.exe que permite instalar la aplicación. Indica el procedimiento paso a paso que debe seguir el nuevo desarrollador para crear el instalador. [Incluye la respuesta en este apartado]

Primero debemos disponer de NSIS en nuestra máquina de sistema operativo Windows. Para descargar el software NSIS debemos ir a esta web:

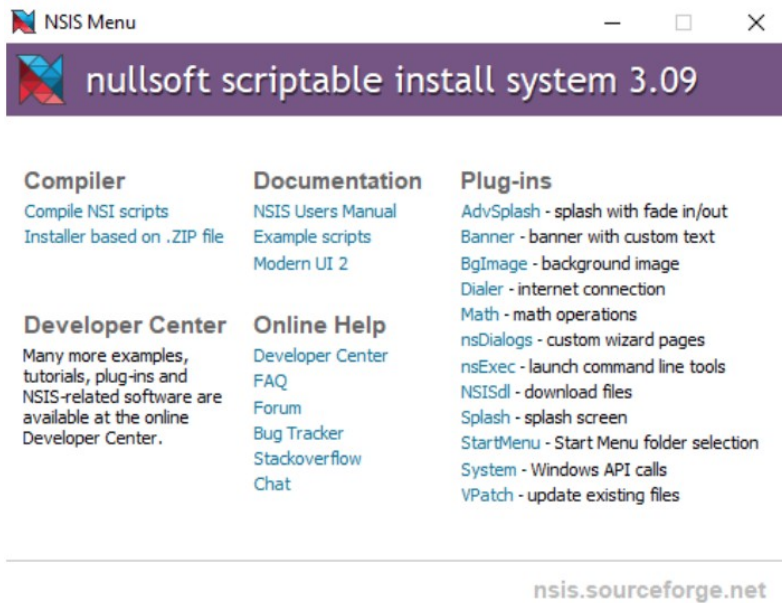
<https://nsis.sourceforge.io/Download>.



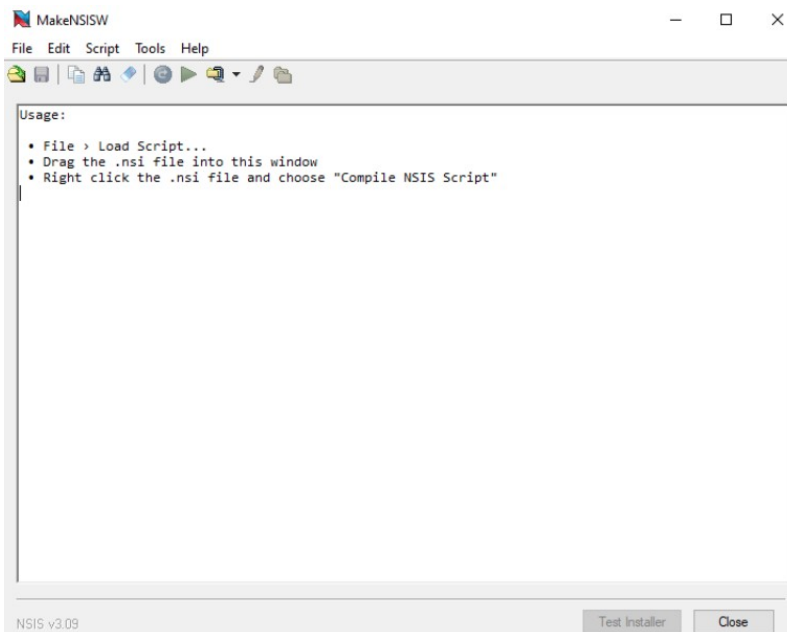
The screenshot shows the 'Download' section of the NSIS 3.11 page on SourceForge. At the top, there are tabs for 'page', 'comment', 'view source', and 'history'. The main heading is 'Download'. Below it, there is a box containing an icon of a box with 'NSIS' and a download arrow, the version 'NSIS 3.11', the release date 'Released March 8th, 2025', and three links: 'Download', 'Release Notes', and 'Source code'. Below this box, there is a section 'Old Releases' with three links: 'NSIS 2 releases', 'NSIS 2 pre-releases', and 'NSIS 1 releases'. Further down, there is a section 'Development Environments' with a paragraph: 'The NSIS package includes a basic compiler interface. If you work frequently'. Below that is a section 'Release Notification' with a link: 'RSS feed is available for released files.'

Debemos darle al botón Azul que dice Download. Con esto se nos instalará el instalador de NSIS. Ejecutamos el instalador y como en nuestro caso queremos todo simplemente aceptamos todo hasta terminar la instalación defecto.

Una vez ejecutamos el programa veremos la siguiente pantalla:



Ahora debemos darle en el apartado Compiler a Compile NSI scripts esto nos abrirá una pestaña como la siguiente:



Ahora debemos darle al menú superior File > Load script... Esto nos abrirá el explorador de archivos y debemos ir a donde tengamos el fichero tic.nsi, en el momento en el que le demos a abrir se ejecutará solo y se creará correctamente el instalador en la misma ruta en la que se encuentra el script. Para probar el instalador podemos darle al botón inferior derecho Test Installer.

Este es el fichero que generó:

