

Retail Financiero



INSTALACIÓN AMBIENTE POS JUMBO

Octubre 2018

Equipo Retail Financiero

Documento que permite configurar el ambiente de desarrollo para ejecutar el POS JUMBO

RETAIL FINANCIERO

Contenido

Información del Documento	2
Pre-requisitos	2
Descargar e instalar Gradle	2
Descargar e instalar Git	3
Crear directorio NCR	3
Configuración de variables de entorno	3
Configuración de Eclipse y pluggins	4
Configuración SSH en GitLab	6
Descargar el proyecto	10
Descargar las dependencias del proyecto	12
Configurar Eris desde CMD	17
Construir la caja	18
Ejecutar caja desde Eclipse	19

Información del Documento

Fecha	Autor	Rol	Versión Doc.	Validado por
16/10/2018	Luis Barrera	Analista	1.0	

Pre-requisitos

Para levantar el ambiente POS de Jumbo se debe tener las siguientes consideraciones.

1. **JDK 1.6** de 32 bits.
2. **JDK 1.8**.
3. **Gradle** versión 3.3 o superior. <https://gradle.org/releases/>
4. **Git**. <https://git-scm.com/downloads/>
5. **Eclipse**. <https://www.eclipse.org/downloads/>
6. Usuario **Active Directory Cencosud**.
7. Permisos sobre el proyecto **cenco-ars-pos** en gitlab.
8. **SourceTree** (opcional, este es un software para la interfaz gráfica de Git).

Descargar e instalar Gradle

1. Descargar Gradle desde: <https://gradle.org/gradle-download/>
2. Luego, se debe crear una carpeta llamada **Tools** en el disco **C:**
3. Finalmente se debe copiar la carpeta Gradle descargada previamente.

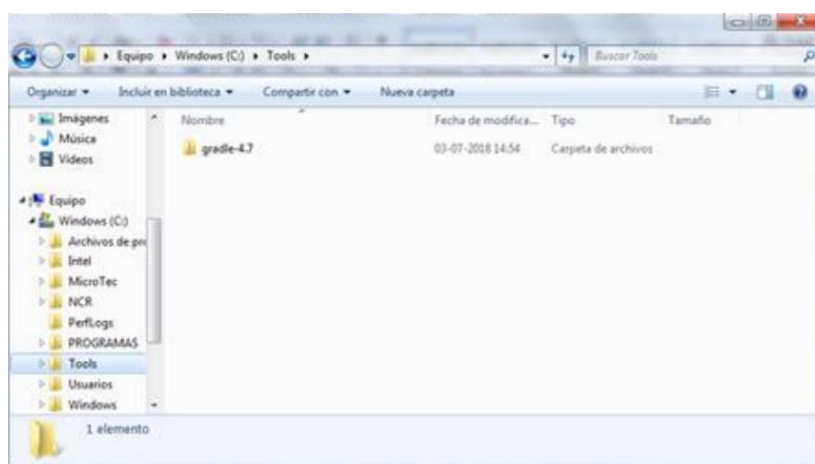


Fig. 1. Creación carpeta **Tools**.

Descargar e instalar Git

1. Descargar Git desde <https://git-scm.com/downloads/>
2. Luego ejecutar el instalador de Git.
3. Posteriormente se debe seguir la instalación por defecto, es decir, se debe seleccionar **next** en cada pantalla que aparezca.

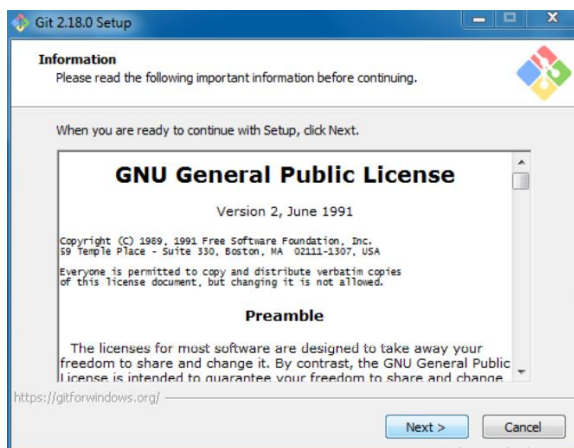


Fig. 2. Instalador de Git.

Crear directorio NCR

1. En la raíz del disco **C:** se debe crear una carpeta llamada **NCR**.
2. Esta carpeta posteriormente alojará el proyecto clonado desde Git.

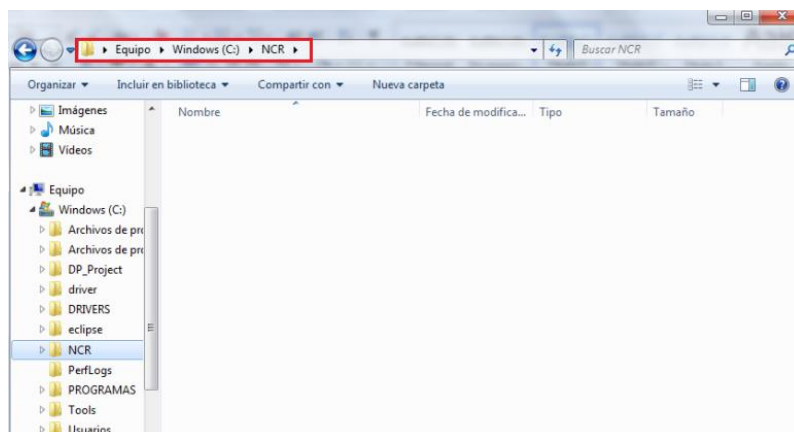


Fig. 3. Creación directorio **NCR**.

Configuración de variables de entorno

1. Se deben configurar tres variables de sistema. Estas son:
 - a. **ARS_POS_JAVA**: *DIRECCION DEL JDK 1.6*
 - b. **GRADLE_HOME**: *DIRECCION DE GRADLE*
 - c. **PATH**: *PATH*+ "%GRADLE_HOME%\bin\;". En el siguiente ejemplo se visualiza como se debe quedar configurada la variable **PATH**.
 C:\ProgramData\Oracle\Java\javapath;%SystemRoot%\system32;%SystemRoot%\system32\WindowsPowerShell\v1.0\;%GRADLE_HOME%\bin\

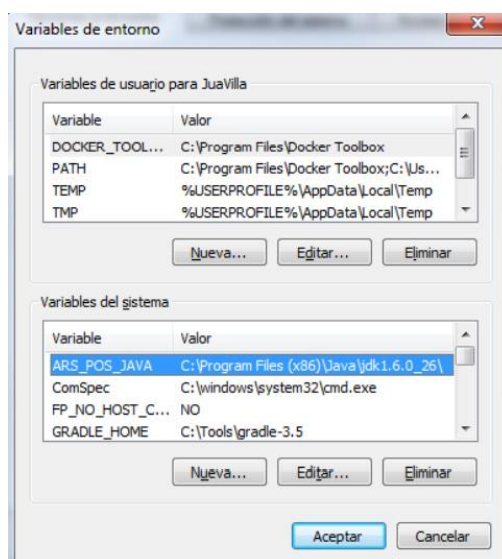


Fig. 4. Configuración variables de entorno.

Configuración de Eclipse y pluggins

Una vez instalado Eclipse se debe proceder a realizar los siguientes pasos:

1. Acceder al **MarketPlace** de Eclipse, mediante el menú **Help**, como se aprecia en la figura 5.

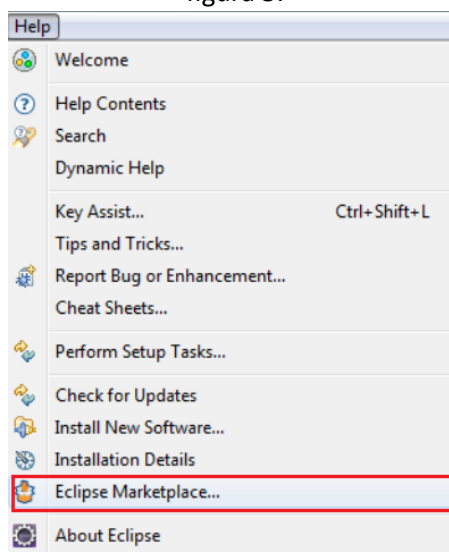


Fig. 5. Acceso a **MarketPlace**.

2. Posteriormente se deben instalar los siguientes softwares.
 - a. Eclipse Class Decompiler.

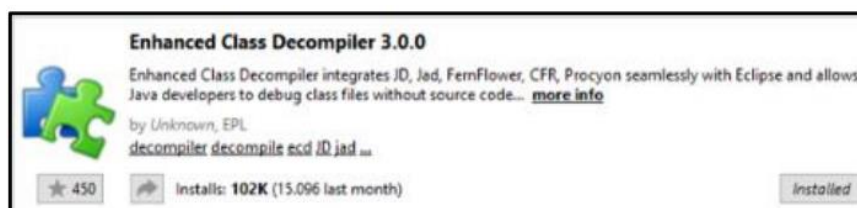


Fig. 5.1. Class decompiler.

b. EcEmma Java Code Coverage.



Fig. 5.2. Java code coverage.

c. JUnit.

d. eGradle (opcional).



Fig. 5.3. EGradle.

3. Posteriormente se debe configurar el compilador JDK 1.6 de Eclipse, para el levantamiento de la caja, a través de **Windows > Preferences**.

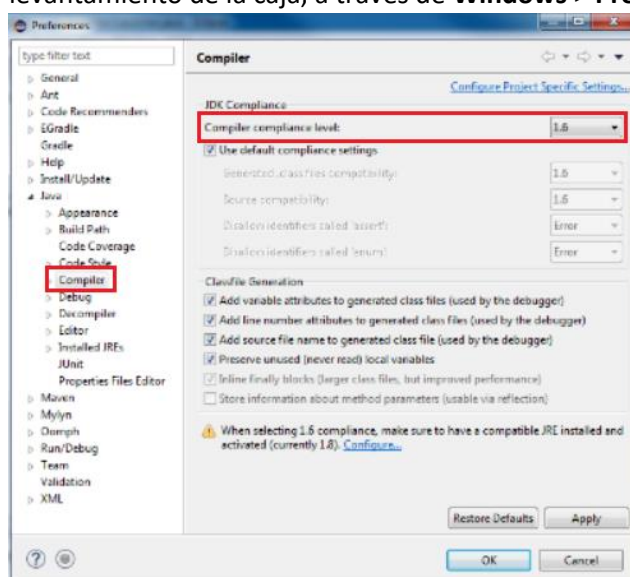


Fig. 6.1. Ventana de configura de JDK.

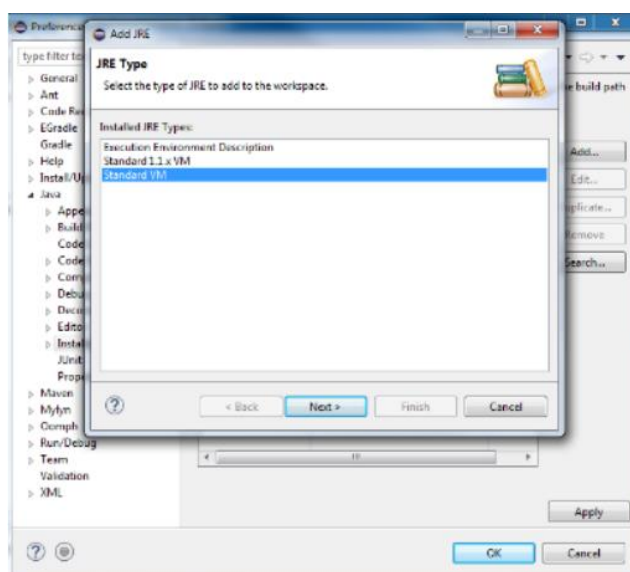


Fig. 6.2. Asignación del JRE.

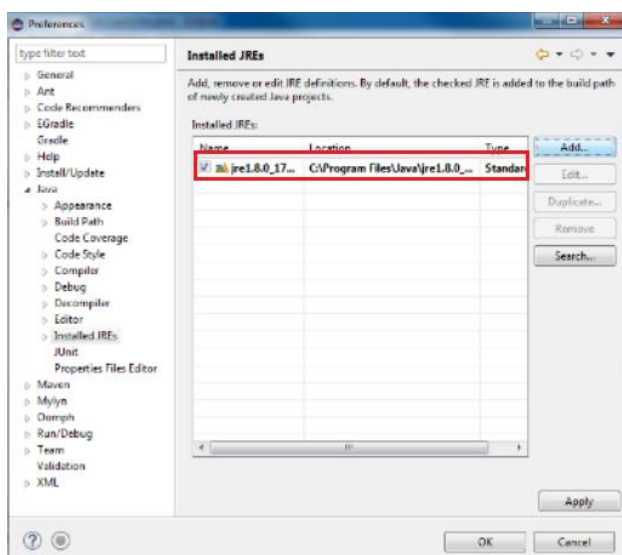
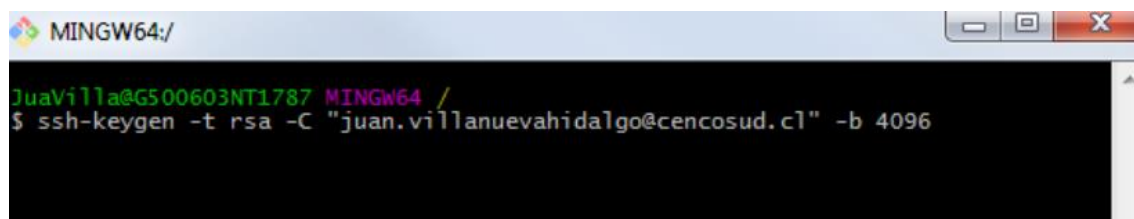


Fig. 6.3. Configuración del JRE.

4. El JRE se encuentra ubicado en la ruta C:\Program Files (x86)\Java\jdk1.6.0_45.
5. Una vez configurado se debe presionar en Finish.
6. Finalmente se debe seleccionar el JDK 1.6 y presionar en aplicar.

Configuración SSH en GitLab

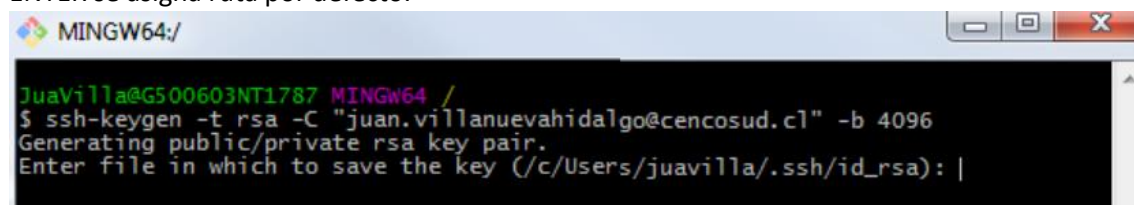
1. Se debe generar la llave SSH en el equipo utilizando el siguiente comando:
 - a. `ssh-keygen -t rsa -C "your.email@example.com" -b 4096`



```
MINGW64:/  
JuaVilla@G500603NT1787 MINGW64 /  
$ ssh-keygen -t rsa -C "juan.villanuevahidalgo@cencosud.cl" -b 4096
```

Fig. 7.1. Generación llaves.

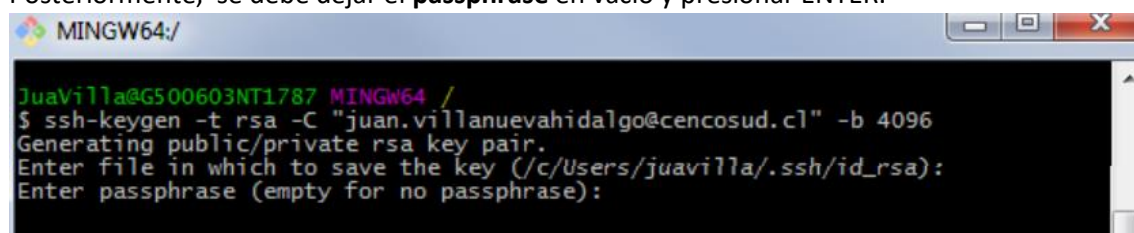
- b. Luego se debe ingresar la ruta donde se almacenará la llave SSH. Al presionar ENTER se asigna ruta por defecto.



```
MINGW64:/  
JuaVilla@G500603NT1787 MINGW64 /  
$ ssh-keygen -t rsa -C "juan.villanuevahidalgo@cencosud.cl" -b 4096  
Generating public/private rsa key pair.  
Enter file in which to save the key (/c/Users/juavilla/.ssh/id_rsa): |
```

Fig. 7.2. Almacenamiento de Keys.

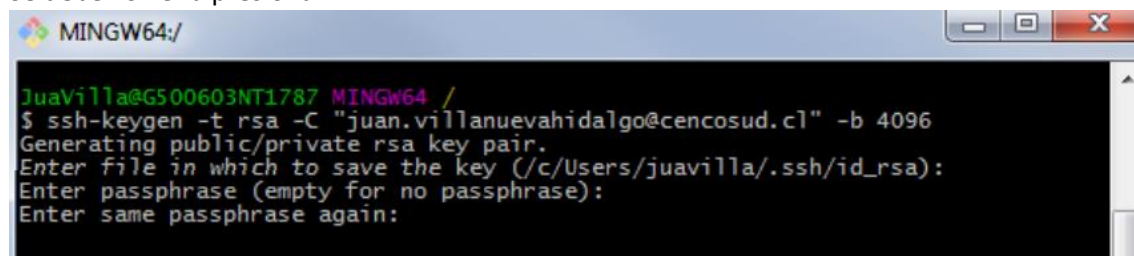
- c. Posteriormente, se debe dejar el **passphrase** en vacío y presionar ENTER.



```
MINGW64:/  
JuaVilla@G500603NT1787 MINGW64 /  
$ ssh-keygen -t rsa -C "juan.villanuevahidalgo@cencosud.cl" -b 4096  
Generating public/private rsa key pair.  
Enter file in which to save the key (/c/Users/juavilla/.ssh/id_rsa):  
Enter passphrase (empty for no passphrase):
```

Fig. 7.3. Configuración del **passphrase**.

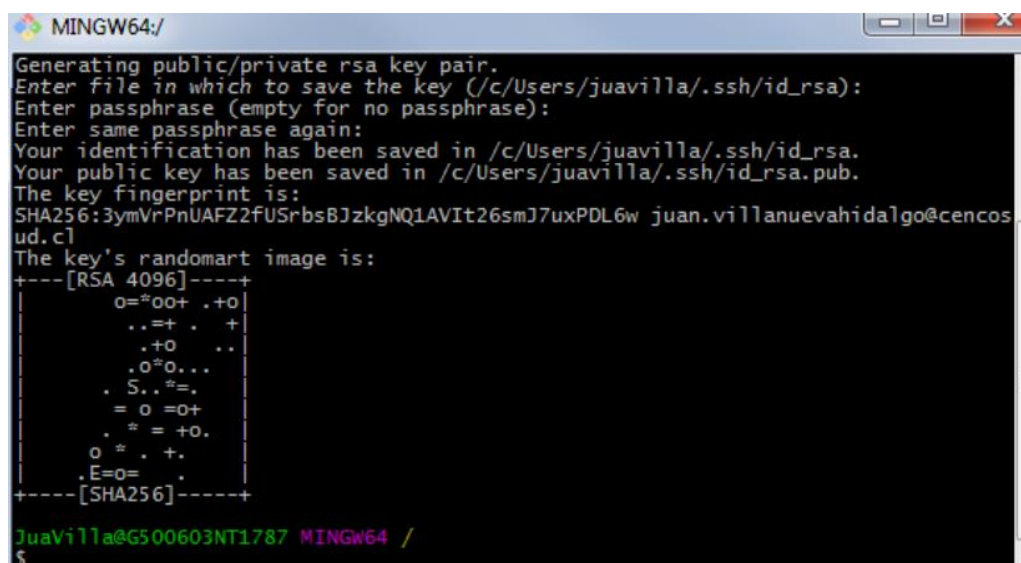
- d. Se debe volver a presionar ENTER.



```
MINGW64:/  
JuaVilla@G500603NT1787 MINGW64 /  
$ ssh-keygen -t rsa -C "juan.villanuevahidalgo@cencosud.cl" -b 4096  
Generating public/private rsa key pair.  
Enter file in which to save the key (/c/Users/juavilla/.ssh/id_rsa):  
Enter passphrase (empty for no passphrase):  
Enter same passphrase again:
```

Fig. 7.4. Confirmación del **passphrase**.

- e. Finalmente la llave se debe generar del siguiente modo.



```

MINGW64:/
Generating public/private rsa key pair.
Enter file in which to save the key (/c:/Users/juavilla/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /c:/Users/juavilla/.ssh/id_rsa.
Your public key has been saved in /c:/Users/juavilla/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:3ymVrPnUAFZ2fUSrbsBJzkgnQ1AVIt26smJ7uxPDL6w juan.villanuevahidalgo@cencosud.cl
The key's randomart image is:
+---[RSA 4096]-----+
|
|o=*oo+ .+o|
|..=+ . +|
|. +o ..|
|.o^o...|
|. S..#=|
|= o=o+|
|. = +o.|
|o * . +|
|.E=o= .|
+---[SHA256]-----+
JuaVilla@G500603NT1787 MINGW64 /
$

```

Fig. 7.5. Proceso de generación de Keys completado correctamente.

- f. Por otro lado, se debe ir a la carpeta donde se generó la llave SSH.

C:\Users*USUARIO*\ .ssh

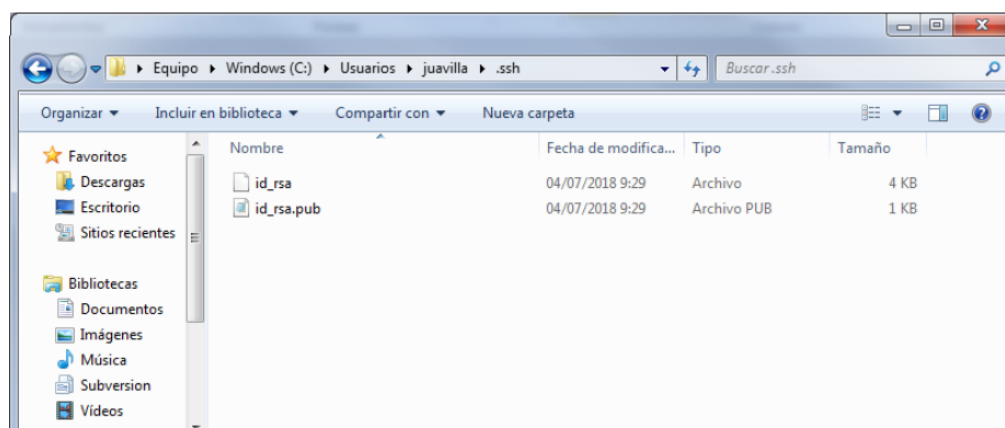


Fig. 7.6. Ruta de acceso a archivo de llaves SSH.

- g. Además se debe copiar el contenido del archivo id_rsa.pub.

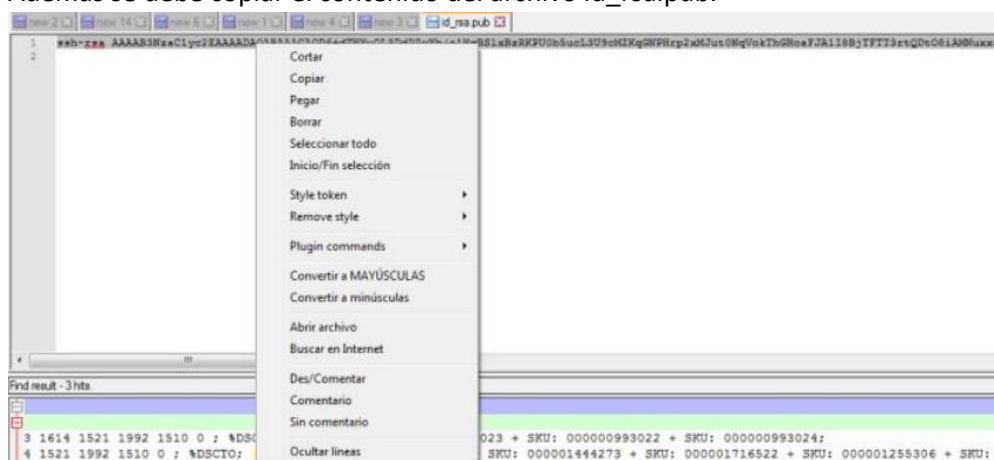


Fig. 7.7. Se debe copiar la llave.

- h. Acceder a la página de GitLab corporativo e ingresar con el usuario AD

<http://gitlab.cencosud.corp/>

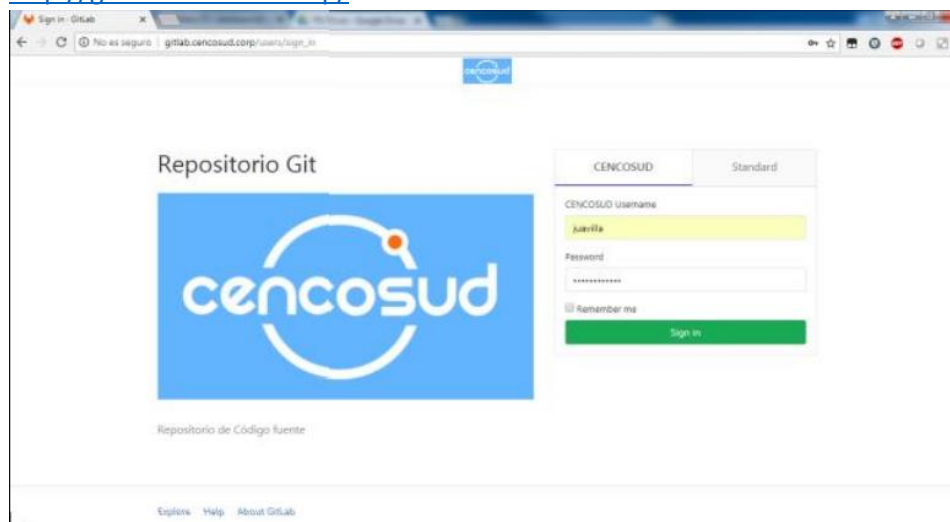


Fig. 7.8. Página de GitLab corporativo.

- i. Luego se debe acceder a las configuraciones de la cuenta.

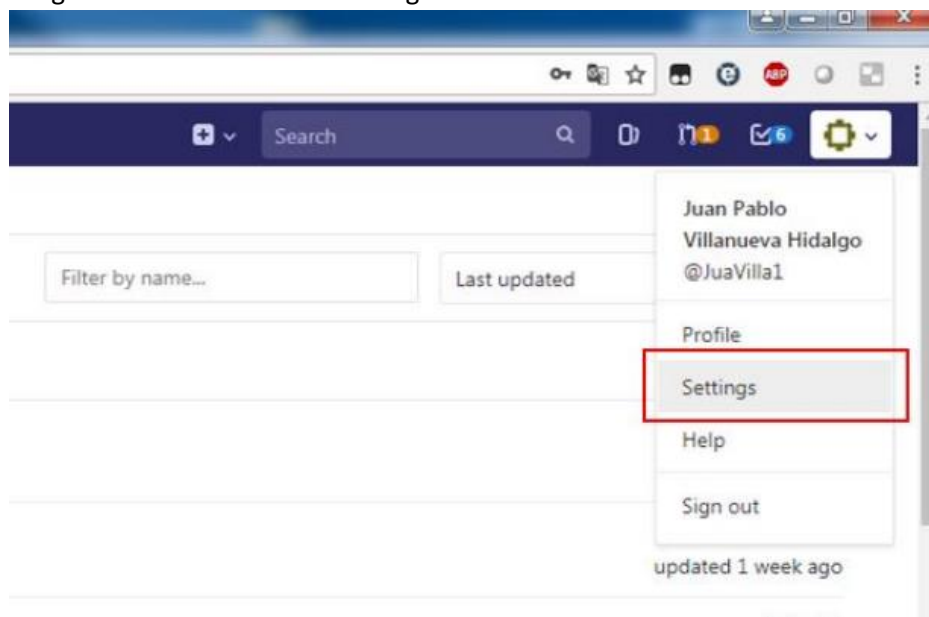


Fig. 7.9. Configuración de la cuenta.

- j. Acceder a la opción SSH **Keys** y pegar la llave copiada en la figura 7.7 dentro del recuadro de texto Key.

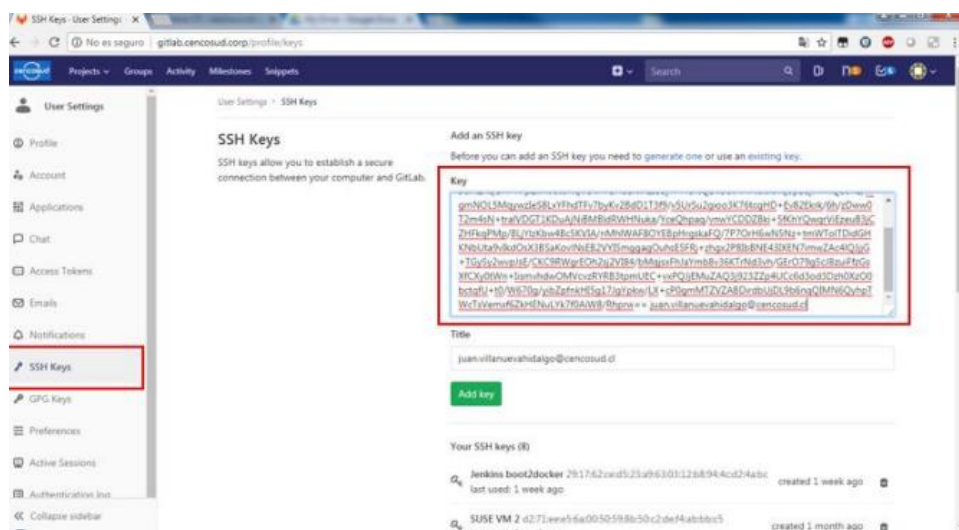


Fig. 7.10. Configuración claves SSH.

- k. Por último se debe presionar en Add Key.
- l. Con esta configuración el equipo queda vinculado a la cuenta Git, permitiendo bajar repositorios, al cual el usuario tenga acceso sin la necesidad de ingresar su clave.

Descargar el proyecto

Una vez configurada la llave SSH, además de poseer los permisos en GitLab, se puede clonar el proyecto con la ruta SSH. Para ello se deben realizar los siguientes pasos.

1. Acceder a Your Projects en la página de GitLab.

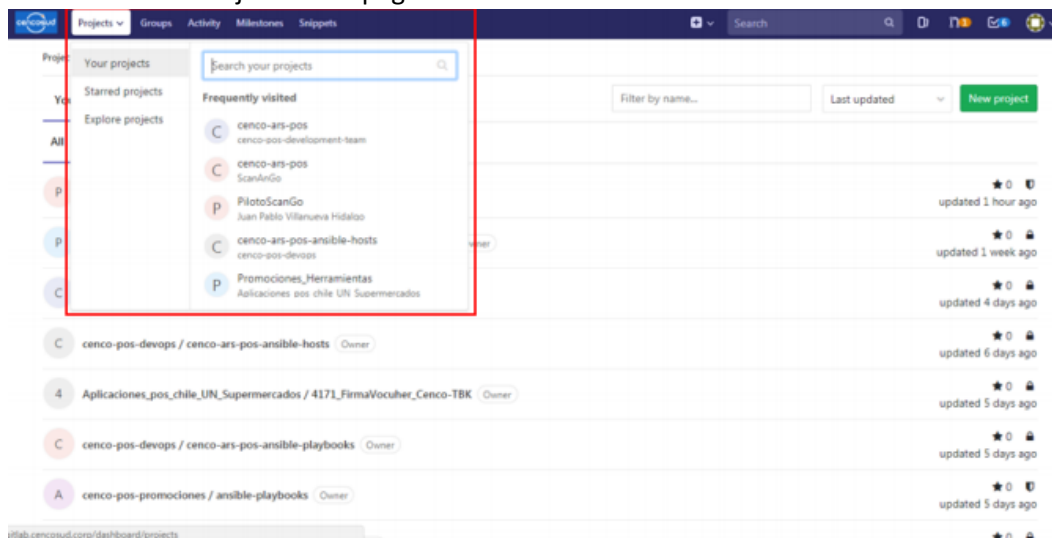


Fig. 8.1. Acceso a proyectos visibles.

2. Seleccionar el proyecto **cenco-pos-developmentteam/cenco-ars-pos**

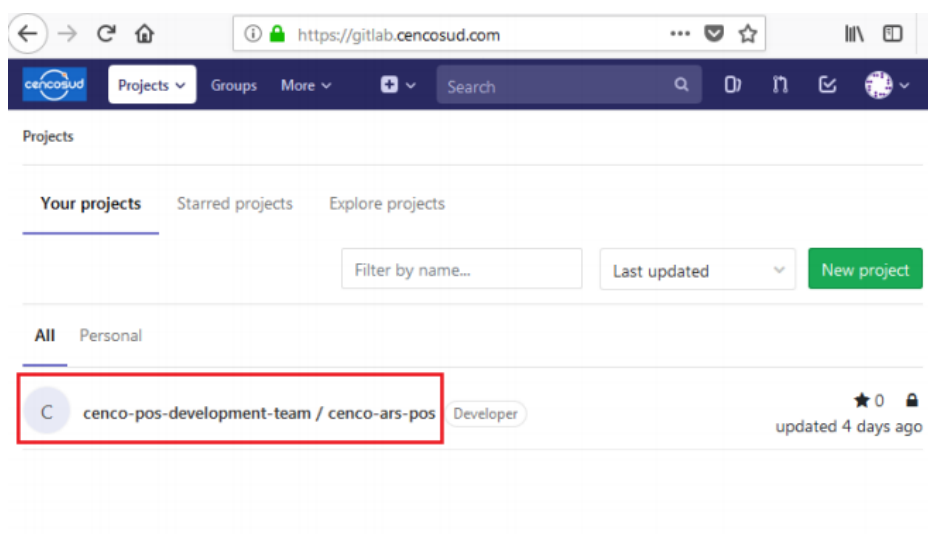


Fig. 8.2. Proyecto que se debe descargar.

3. Seleccionar la ruta **SSH** del repositorio y luego copiar la ruta.

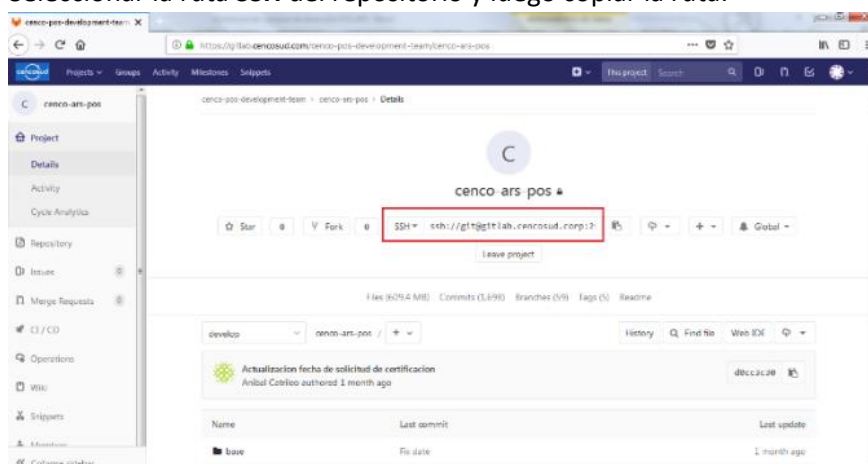


Fig. 8.3. URL del SSH para descargar el proyecto.

4. En el directorio de NCR (C:\NCR\), hacer botón derecho y seleccionar Git Bash Here

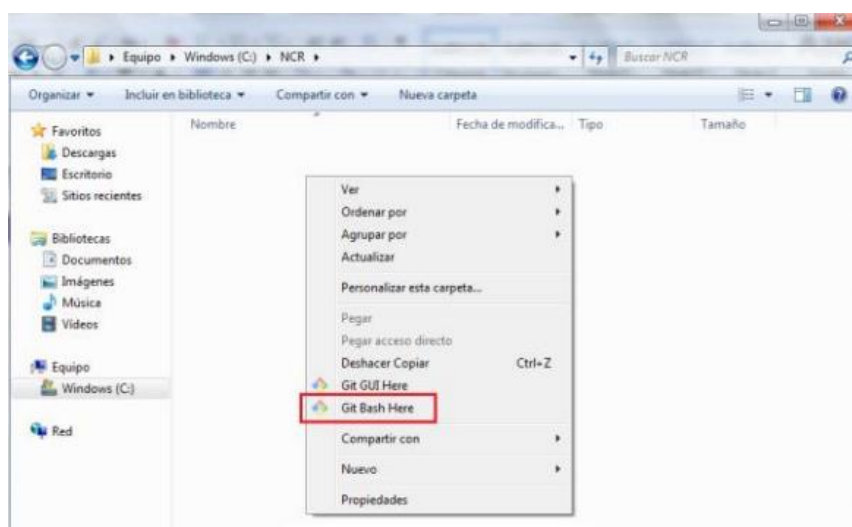


Fig. 8.4. Ejecutar Git en la carpeta de descarga del proyecto.

5. Clonar el proyecto con el comando git clone + la ruta copiada.

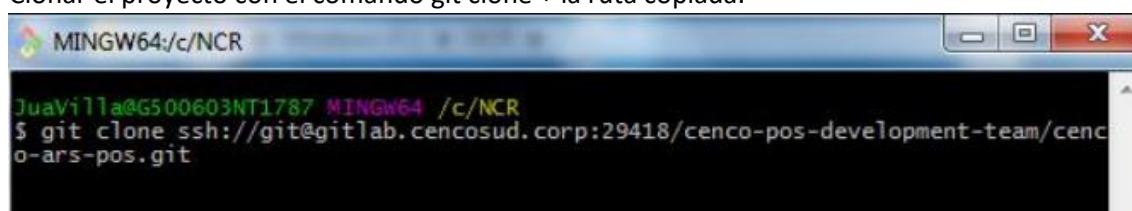


Fig. 8.5. Se clona el proyecto en el directorio seleccionado.

6. Luego el proyecto se descarga completamente.

Descargar las dependencias del proyecto

Una vez descargado el proyecto será necesario descargar las dependencias.

1. Acceder a la carpeta “cenco-ars-pos/base” del proyecto clonado desde la consola

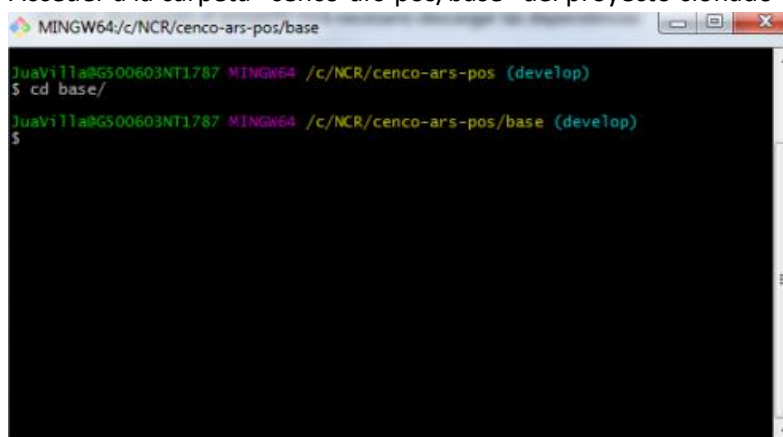
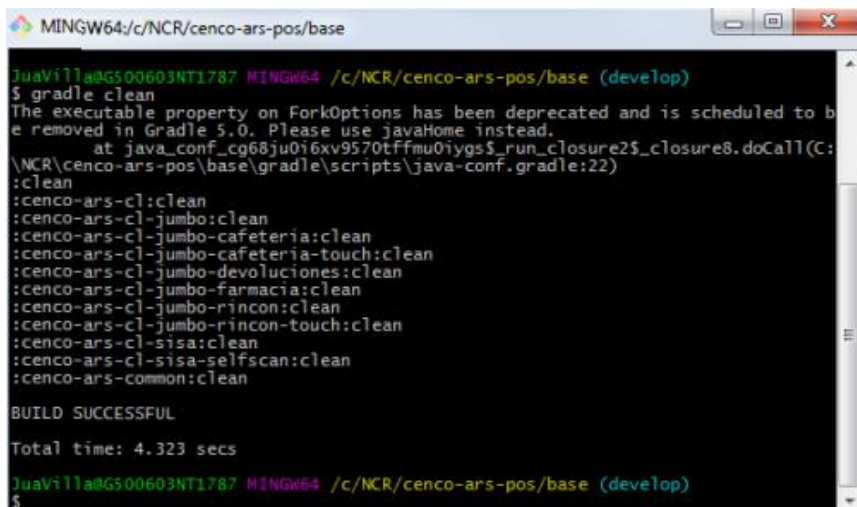


Fig. 9.1. Posicionarse en la ruta base del proyecto.

2. Ejecutar los siguientes comandos
 - a. gradle clean



```

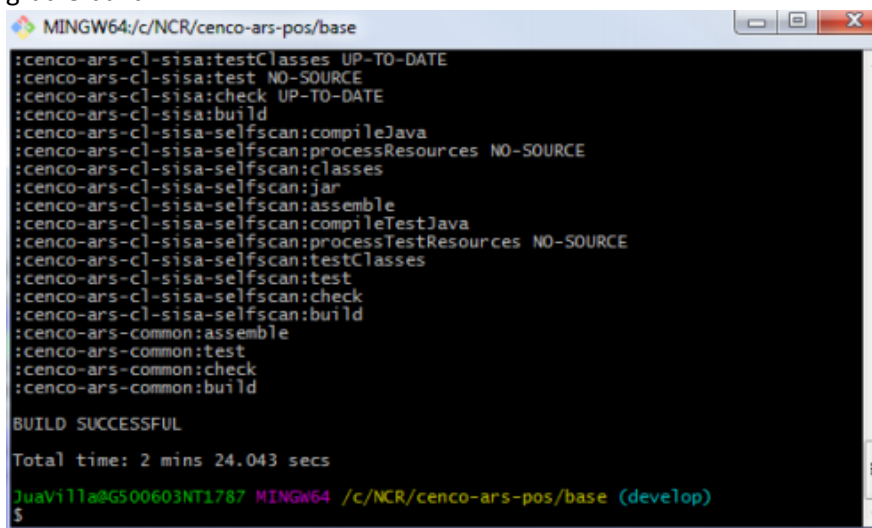
MINGW64/c/NCR/cenco-ars-pos/base
JuaVilla@GS00603NT1787 MINGW64 /c/NCR/cenco-ars-pos/base (develop)
$ gradle clean
The executable property on ForkOptions has been deprecated and is scheduled to be removed in Gradle 5.0. Please use javaHome instead.
    at java_conf_cg68ju0i6xy9570tffmu0iygs$run_closure2$_closure8.doCall(C:\NCR\cenco-ars-pos\base\gradle\scripts\java-conf.gradle:22)
:clean
:cenco-ars-cl:clean
:cenco-ars-cl-jumbo:clean
:cenco-ars-cl-jumbo-cafeteria:clean
:cenco-ars-cl-jumbo-cafeteria-touch:clean
:cenco-ars-cl-jumbo-devoluciones:clean
:cenco-ars-cl-jumbo-farmacia:clean
:cenco-ars-cl-jumbo-rincon:clean
:cenco-ars-cl-jumbo-rincon-touch:clean
:cenco-ars-cl-sisa:clean
:cenco-ars-cl-sisa-selfscan:clean
:cenco-ars-common:clean

BUILD SUCCESSFUL
Total time: 4.323 secs
JuaVilla@GS00603NT1787 MINGW64 /c/NCR/cenco-ars-pos/base (develop)
$

```

Fig. 9.2. Ejecución del comando gradle clean.

b. gradle build



```

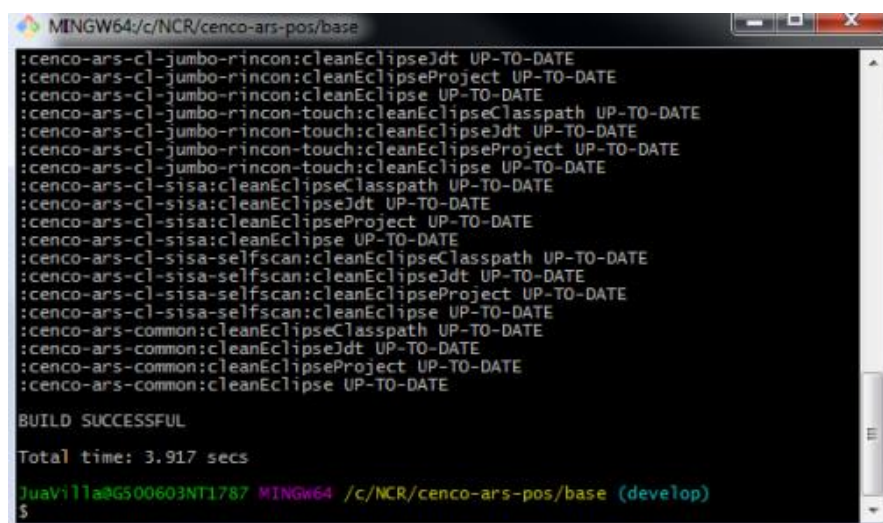
MINGW64/c/NCR/cenco-ars-pos/base
:cenco-ars-cl-sisa:testClasses UP-TO-DATE
:cenco-ars-cl-sisa:test NO-SOURCE
:cenco-ars-cl-sisa:check UP-TO-DATE
:cenco-ars-cl-sisa:build
:cenco-ars-cl-sisa-selfscan:compileJava
:cenco-ars-cl-sisa-selfscan:processResources NO-SOURCE
:cenco-ars-cl-sisa-selfscan:classes
:cenco-ars-cl-sisa-selfscan:jar
:cenco-ars-cl-sisa-selfscan:assemble
:cenco-ars-cl-sisa-selfscan:compileTestJava
:cenco-ars-cl-sisa-selfscan:processTestResources NO-SOURCE
:cenco-ars-cl-sisa-selfscan:testClasses
:cenco-ars-cl-sisa-selfscan:test
:cenco-ars-cl-sisa-selfscan:check
:cenco-ars-cl-sisa-selfscan:build
:cenco-ars-common:assemble
:cenco-ars-common:test
:cenco-ars-common:check
:cenco-ars-common:build

BUILD SUCCESSFUL
Total time: 2 mins 24.043 secs
JuaVilla@GS00603NT1787 MINGW64 /c/NCR/cenco-ars-pos/base (develop)
$

```

Fig. 9.3. Ejecución del comando gradle build.

- Esto bajará todas las dependencias del proyecto (se puede agregar la extensión `-x test` para omitir la ejecución de los JUnit Test
- c. Importar el proyecto en Eclipse, ejecutando los siguientes comandos.



```

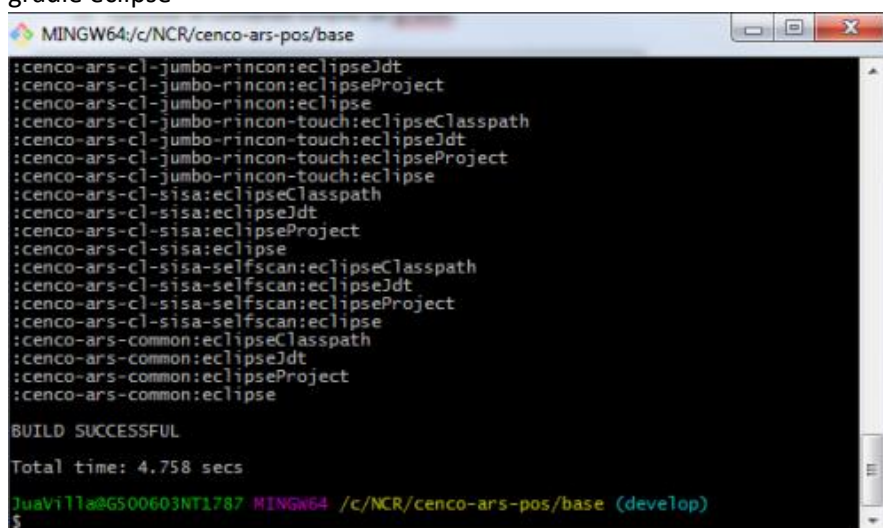
MINGW64/c/NCR/cenco-ars-pos/base
:cenco-ars-cl-jumbo-rincon:cleanEclipseJdt UP-TO-DATE
:cenco-ars-cl-jumbo-rincon:cleanEclipseProject UP-TO-DATE
:cenco-ars-cl-jumbo-rincon:cleanEclipse UP-TO-DATE
:cenco-ars-cl-jumbo-rincon-touch:cleanEclipseClasspath UP-TO-DATE
:cenco-ars-cl-jumbo-rincon-touch:cleanEclipseJdt UP-TO-DATE
:cenco-ars-cl-jumbo-rincon-touch:cleanEclipseProject UP-TO-DATE
:cenco-ars-cl-jumbo-rincon-touch:cleanEclipse UP-TO-DATE
:cenco-ars-cl-sisa:cleanEclipseClasspath UP-TO-DATE
:cenco-ars-cl-sisa:cleanEclipseJdt UP-TO-DATE
:cenco-ars-cl-sisa:cleanEclipseProject UP-TO-DATE
:cenco-ars-cl-sisa:cleanEclipse UP-TO-DATE
:cenco-ars-cl-sisa-selfscan:cleanEclipseClasspath UP-TO-DATE
:cenco-ars-cl-sisa-selfscan:cleanEclipseJdt UP-TO-DATE
:cenco-ars-cl-sisa-selfscan:cleanEclipseProject UP-TO-DATE
:cenco-ars-cl-sisa-selfscan:cleanEclipse UP-TO-DATE
:cenco-ars-common:cleanEclipseClasspath UP-TO-DATE
:cenco-ars-common:cleanEclipseJdt UP-TO-DATE
:cenco-ars-common:cleanEclipseProject UP-TO-DATE
:cenco-ars-common:cleanEclipse UP-TO-DATE

BUILD SUCCESSFUL
Total time: 3.917 secs
juaVill@GS00603NT1787 MINGW64 /c/NCR/cenco-ars-pos/base (develop)
$

```

Fig. 9.4. Ejecución del comando gradle cleanEclipse.

d. gradle eclipse



```

MINGW64/c/NCR/cenco-ars-pos/base
:cenco-ars-cl-jumbo-rincon:eclipseJdt
:cenco-ars-cl-jumbo-rincon:eclipseProject
:cenco-ars-cl-jumbo-rincon:eclipse
:cenco-ars-cl-jumbo-rincon-touch:eclipseClasspath
:cenco-ars-cl-jumbo-rincon-touch:eclipseJdt
:cenco-ars-cl-jumbo-rincon-touch:eclipseProject
:cenco-ars-cl-jumbo-rincon-touch:eclipse
:cenco-ars-cl-sisa:eclipseClasspath
:cenco-ars-cl-sisa:eclipseJdt
:cenco-ars-cl-sisa:eclipseProject
:cenco-ars-cl-sisa:eclipse
:cenco-ars-cl-sisa-selfscan:eclipseClasspath
:cenco-ars-cl-sisa-selfscan:eclipseJdt
:cenco-ars-cl-sisa-selfscan:eclipseProject
:cenco-ars-cl-sisa-selfscan:eclipse
:cenco-ars-common:eclipseClasspath
:cenco-ars-common:eclipseJdt
:cenco-ars-common:eclipseProject
:cenco-ars-common:eclipse

BUILD SUCCESSFUL
Total time: 4.758 secs
juaVill@GS00603NT1787 MINGW64 /c/NCR/cenco-ars-pos/base (develop)
$

```

Fig. 9.5. Ejecución del comando gradle eclipse.

e. Abrir Eclipse, seleccionar el Workspace (C:/NCR/) e importar el proyecto.

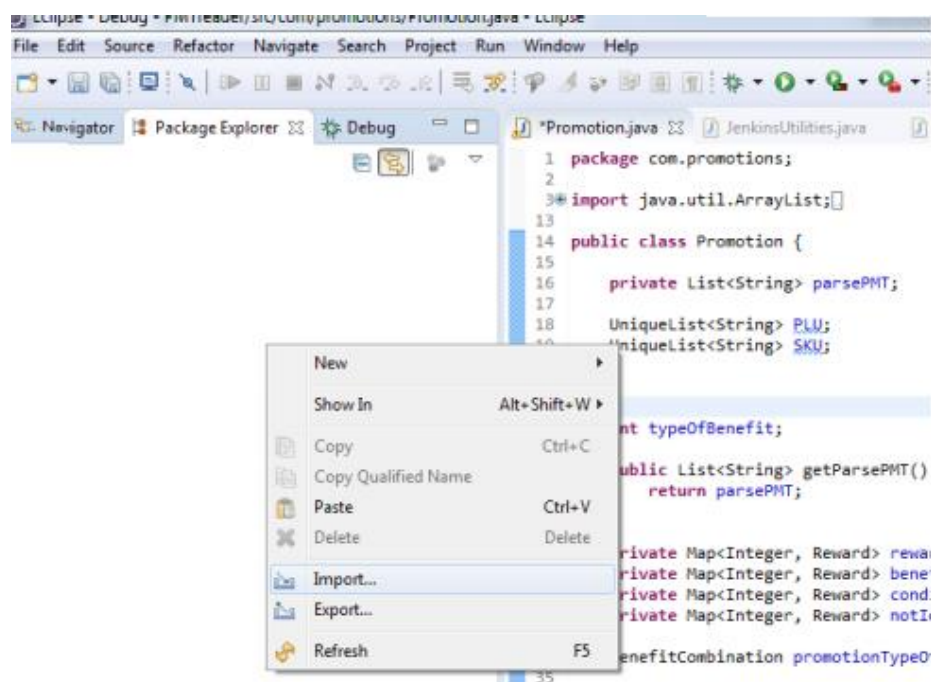


Fig. 9.6. Importar el proyecto.

- f. Seleccionar la opción “Existing Project into Workspace” y presionar Next.

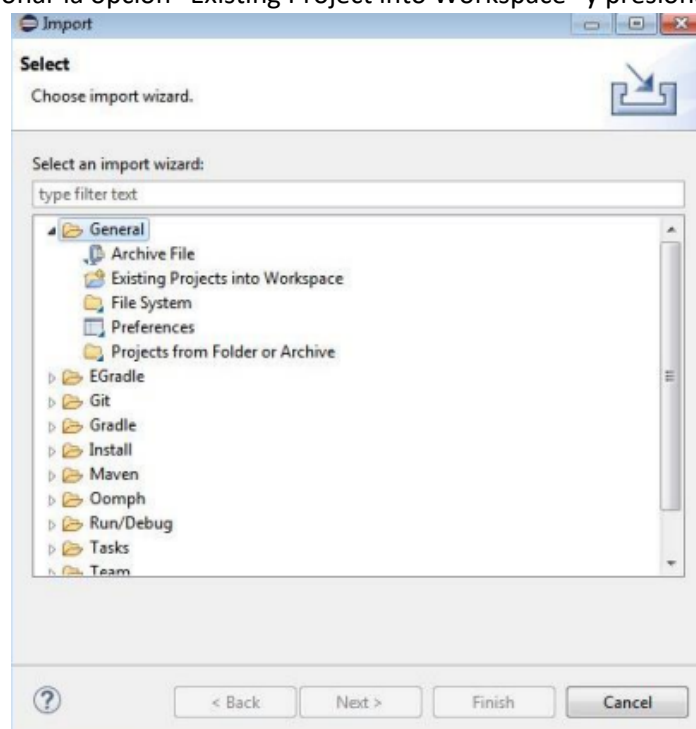


Fig. 9.7. Selección de proyecto existente.

- g. Ingresar la ruta “C:\NCR\censo-ars-pos” (donde se clonó el proyecto), seleccionar todos los proyectos y presionar Finish.

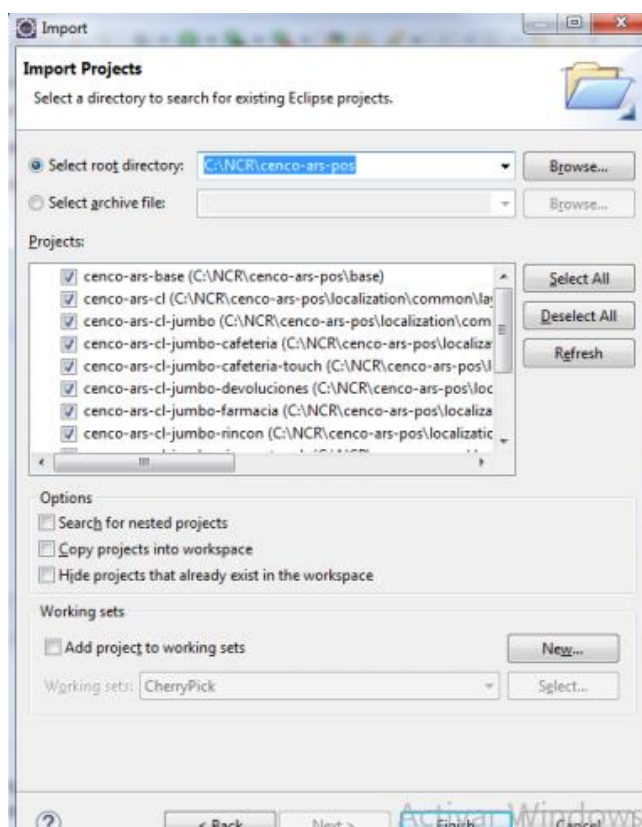


Fig. 9.8. Importar proyectos.

- h. Una vez terminado el paso anterior deberían mostrarse los proyectos como en la siguiente imagen

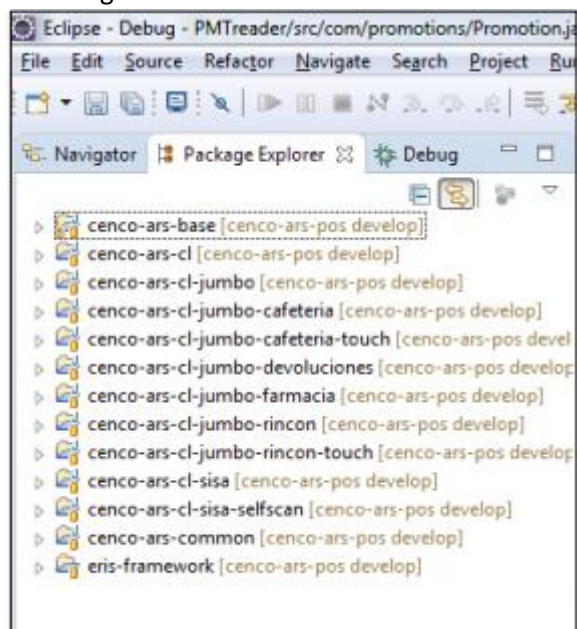


Fig. 9.9. Proyecto cargados correctamente.

Configurar Eris desde CMD

1. Iniciar la consola de Windows como administrador.
2. Ingresar a la carpeta de eris, que se encuentra disponible en la siguiente ruta de acceso. **cd C:\NCR\cenco-ars-pos\eris_framework\eris**

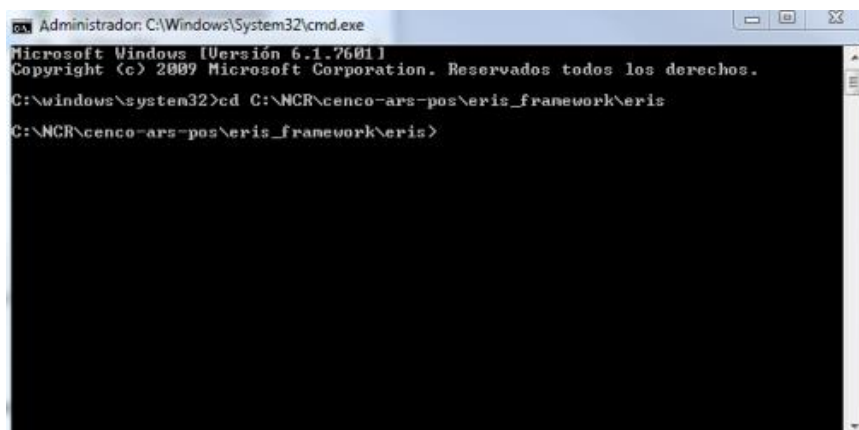


Fig. 10. Acceder a la carpeta de eris.

3. Ejecutar init.bat

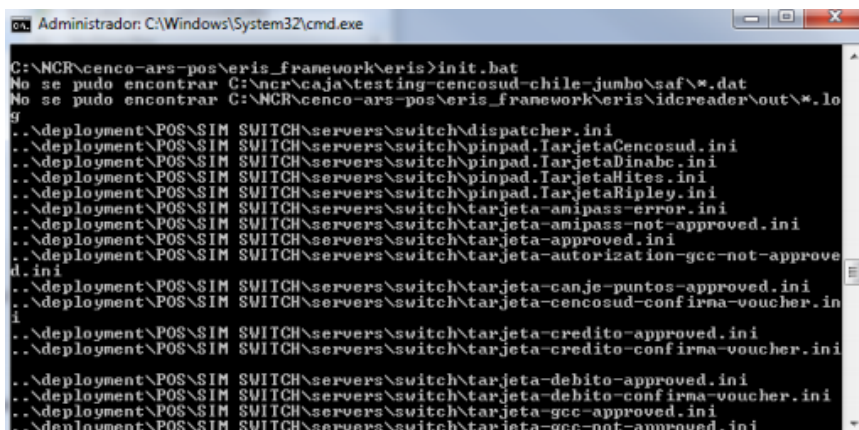


Fig. 11. Ejecutar comando init.bat.

4. Esto debe generar una carpeta "caja" en NCR.

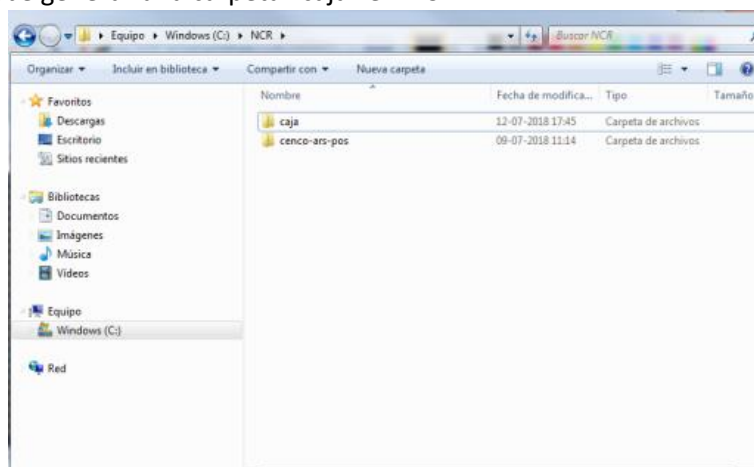
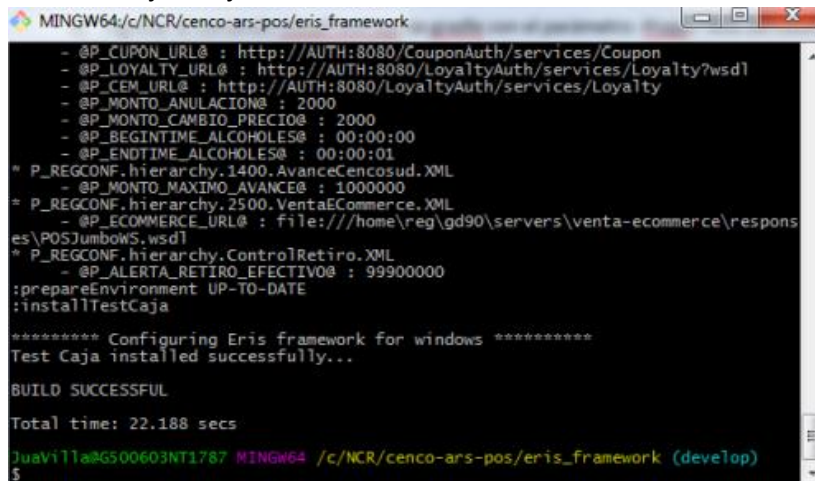


Fig. 12. Validar la creación de la carpeta caja.

Construir la caja

1. Acceder a la ruta “cenco-ars-pos/eris_framework/” desde consola
2. Ingresar los siguientes comandos de gradle.
 - a. gradle clean
 - b. gradle installTestCaja -Pcaja=CL-JUMBO



```

MINGW64/c/NCR/cenco-ars-pos/eris_framework
- @P_CUPON_URL@ : http://AUTH:8080/CouponAuth/services/Coupon
- @P_LOYALTY_URL@ : http://AUTH:8080/LoyaltyAuth/services/Loyalty?wsdl
- @P_CEM_URL@ : http://AUTH:8080/LoyaltyAuth/services/Loyalty
- @P_MONTO_ANULACION@ : 2000
- @P_MONTO_CAMBIO_PRECIO@ : 2000
- @P_BEGINTIME_ALCOHOLES@ : 00:00:00
- @P_ENDTIME_ALCOHOLES@ : 00:00:01
* P_REGCONF.hierarchy.1400.AvancaCencosud.XML
- @P_MONTO_MAXIMO_AVANCE@ : 1000000
* P_REGCONF.hierarchy.2500.VentaECommerce.XML
- @P_ECOMMERCE_URL@ : file:///home/reg/gd90/servers/venta-ecommerce/respons
es\POSJumbowS.wsdl
* P_REGCONF.hierarchy.ControlRetiro.XML
- @P_ALERTA_RETIRO_EFECTIVO@ : 99900000
:prepareEnvironment UP-TO-DATE
:installTestCaja

***** Configuring Eris framework for windows *****
Test Caja installed successfully...

BUILD SUCCESSFUL
Total time: 22.188 secs
JuaVilla@G500603NT1787 MINGW64 /c/NCR/cenco-ars-pos/eris_framework (develop)
$
  
```

Fig. 13. Instalación de la caja.

* Esto creará una carpeta en “C:\NCR\caja” con el contenido necesario para ejecutar la caja.

ROLES
CL-JUMBO
CL-JUMBO-RINCON
CL-JUMBO-RINCON-TOUCH
CL-JUMBO-FARMACIA
CL-JUMBO-CAFETERIA
CL-JUMBO-CAFETERIA-TOUCH
CL-JUMBO-DEVOLUCIONES
CL-SISA
CL-SISA-SELFSCAN

- c. Finalmente se debe ejecutar la caja. **./ARS.bat**



Fig. 14. Caja levantada en el ambiente.

Ejecutar caja desde Eclipse

1. Acceder a “Run Configurations...” de Eclipse.

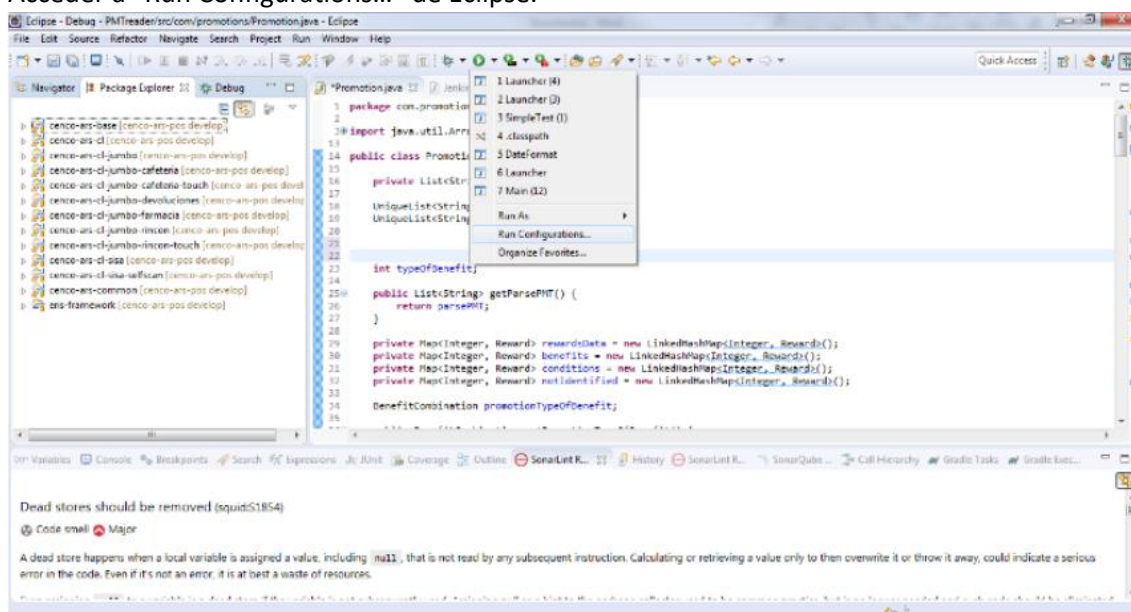


Fig. 15. Ingresar al Run configurations.

2. Crear una nueva aplicación java.

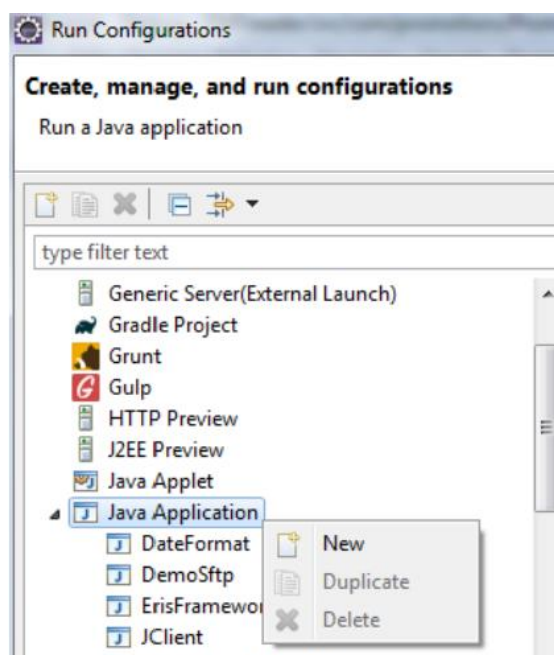


Fig. 16. Run configurations.

3. Escribir el nombre de la aplicación (En este caso se nombre como POS JUMBO).

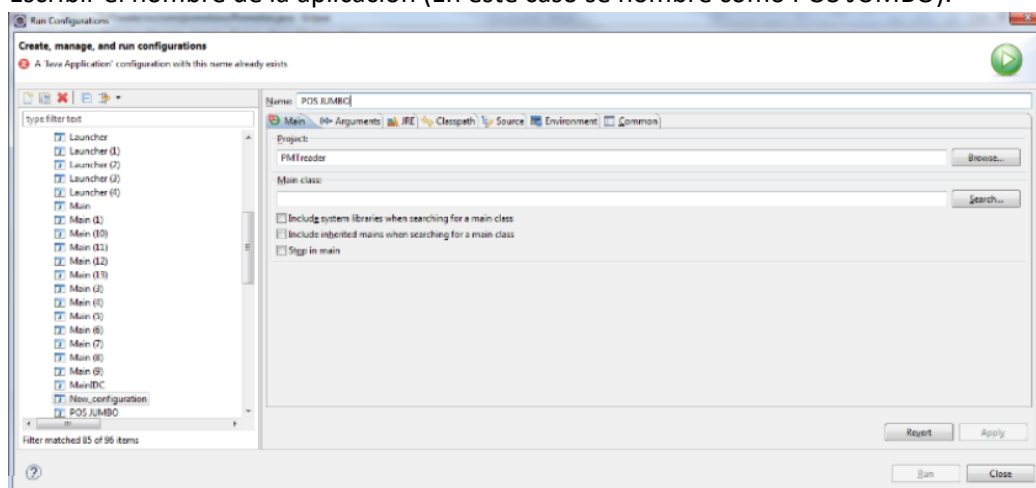
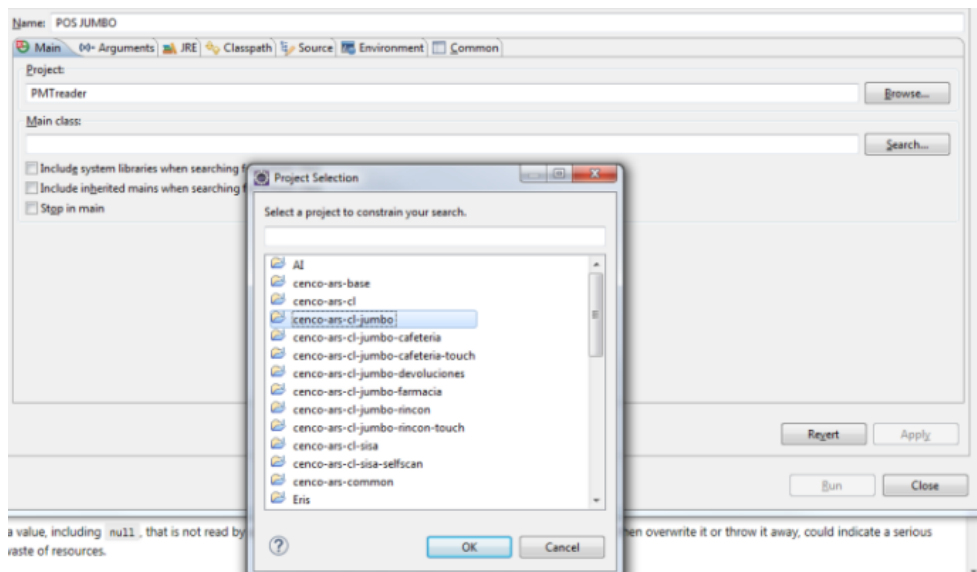
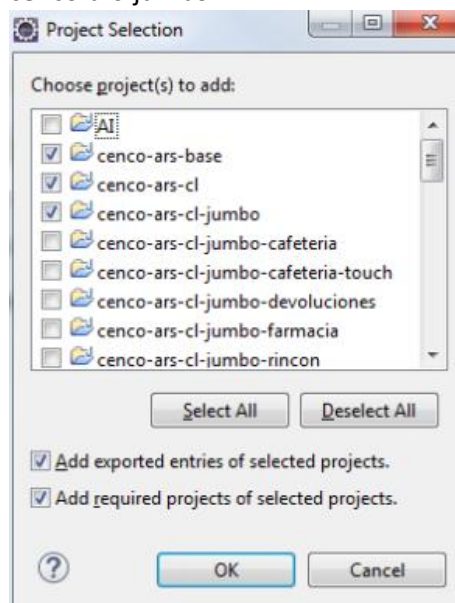


Fig. 17. Nombre de la aplicación.

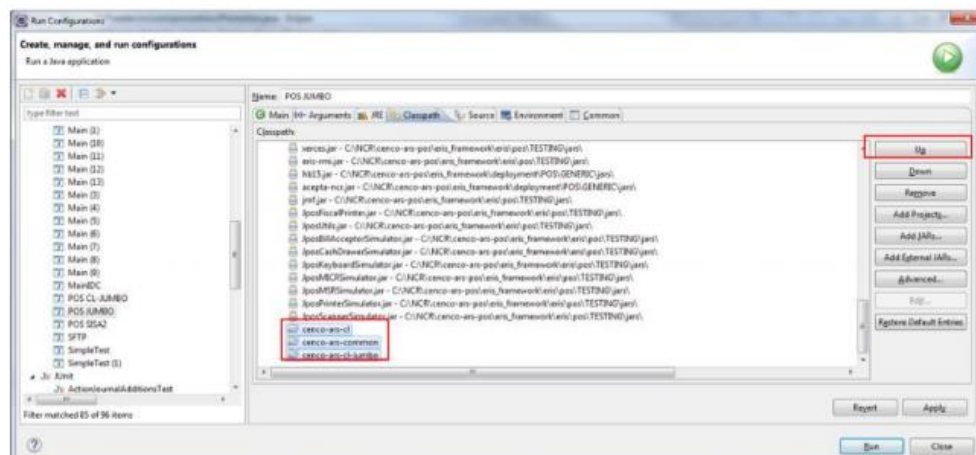
4. Se selecciona el proyecto del rol que se quiere ejecutar. Para este caso ejecutaremos el rol **CL-JUMBO**.



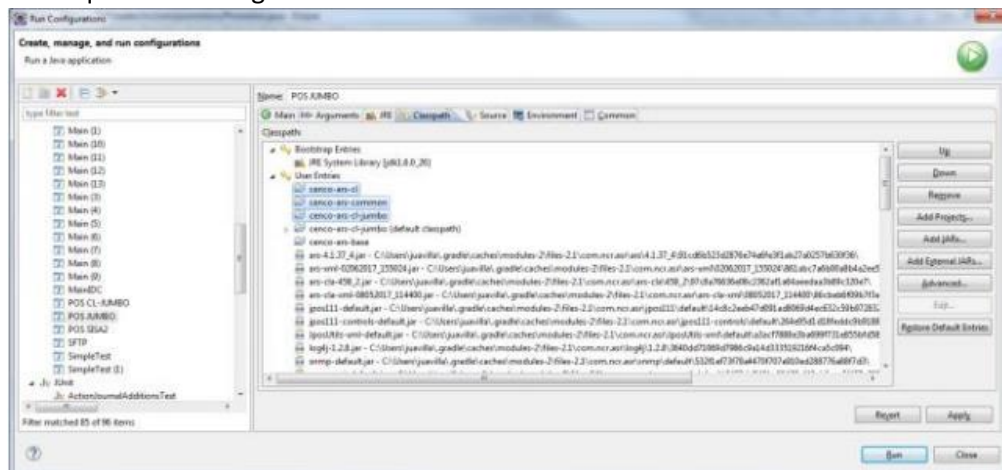
5. En Arguments ingresar los siguientes campos:
 - a. VM Arguments
 - i. -Djava.library.path=. -Duser.country=CL -Duser.language=es
 - b. En Working directory, other
 - i. C:\NCR\caja\testing-cencosud-chile-jumbo
6. Asegurarse de que el JRE sea el 1.6.
7. En Classpath se deben agregar las dependencias de otros proyectos, seleccionar "User Entries" y luego "Add Project".
 - a. Agregar los proyectos dependencias:
 - i. cenco-ars-base
 - ii. cenco-ars-cl
 - iii. cenco-ars-jumbo



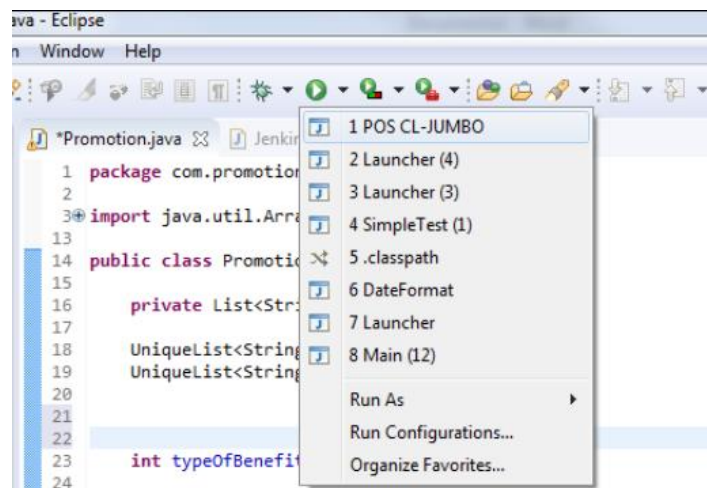
8. Subir las dependencias de proyectos al principio.



9. Debe quedar de la siguiente manera.



10. Guardar todo y ejecutar.



IMPORTANTE: Tener en consideración que todos los archivos de configuración (XML, DAT, etc) serán leídos desde “C:\NCR\caja\testing-cencosud-chile-jumbo”. Por ende si estos son modificados, se debe reinstalar la caja.