

**Universidad de El Salvador.**

**Facultad Multidisciplinaria de Occidente.**

**Departamento de Ingeniería y Arquitectura**

**Catedra: Curso progra 2.5**



**Objetivo:** que el estudiante entienda o al menos tenga una noción de la estructura básica de un proyecto Enterprise de Java así como la configuración necesaria del servidor de aplicaciones para poder desplegar el proyecto.

**Requisitos:**

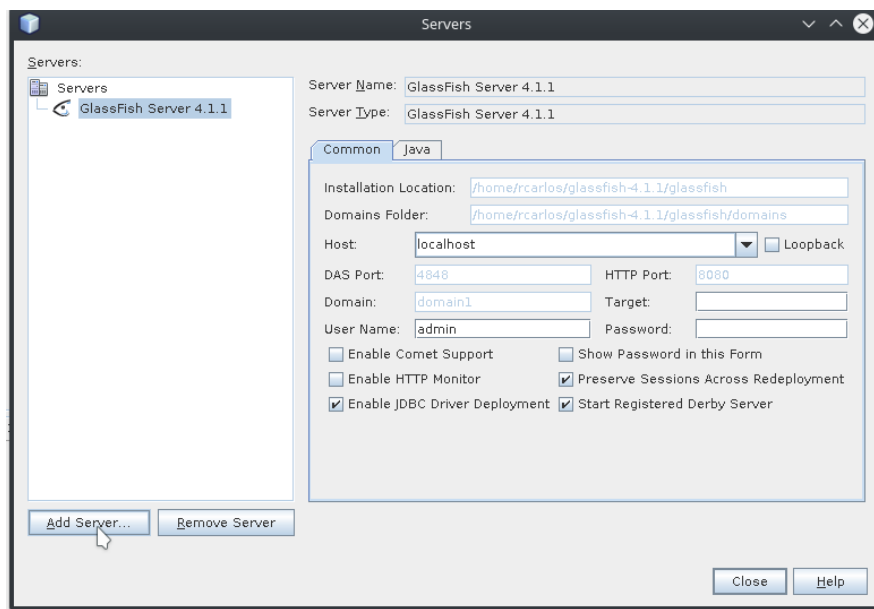
- Tener instalado postgresql (Localmente o en un contenedor, es indiferente)
- Usuario “mortal”, contraseña “mortal” configurado en postgresql
- Base de datos “resbar”

**NOTA:** pude acceder al script para generar las tablas en el siguiente repositorio:  
<https://github.com/rcarlos97/recursos> en la carpeta “/local”

Para poder ejecutar nuestros proyectos desde NetBeans, debemos registrar una instalación del servidor Glassfish como una instancia de un servidor de NetBeans. A continuación se detalla el proceso para registrar el servidor.

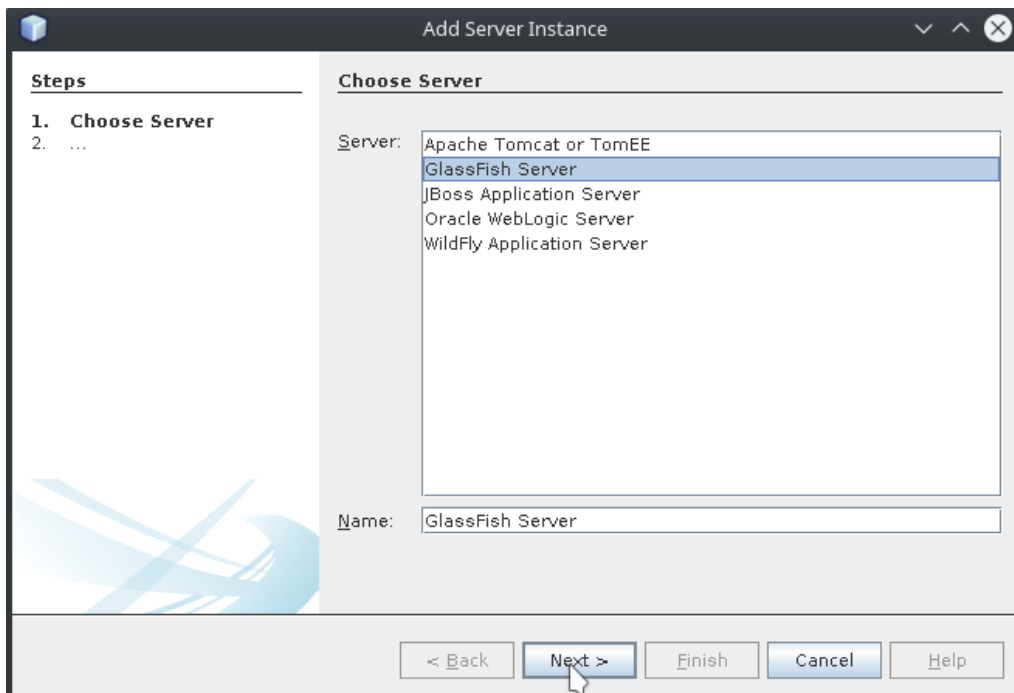
Primero nos dirigimos a la pestaña superior de NetBeans **Tools > Servers**

Lo siguiente es dar click al botón Add Server

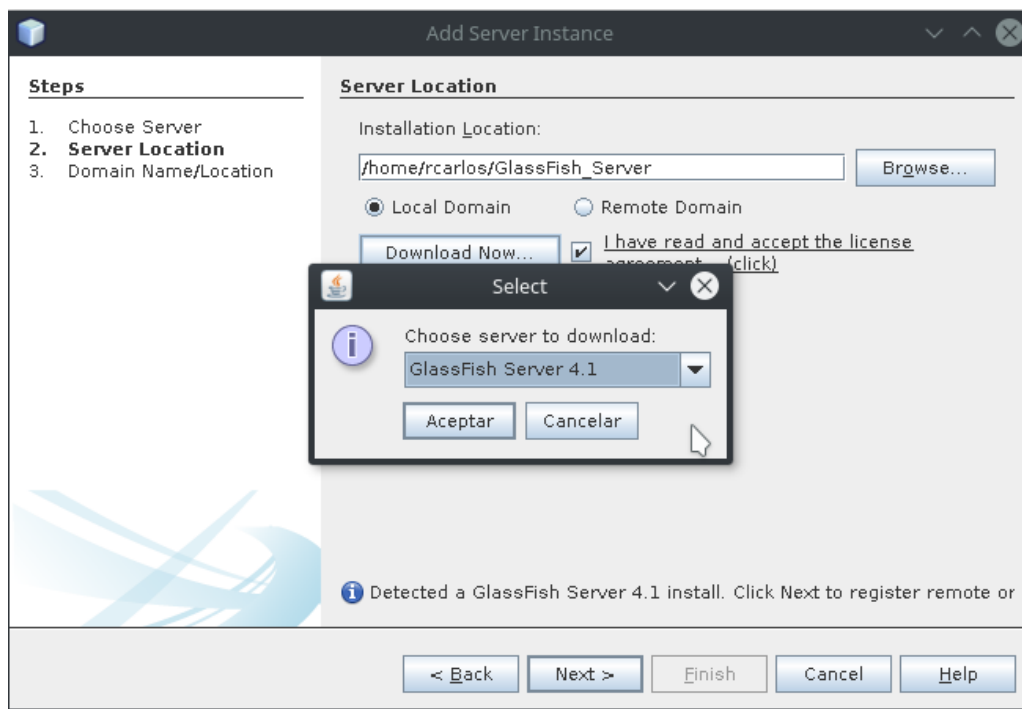


(Glassfish Server 4.1.1 es el servidor que a veces se instala junto con NetBeans pero posee un par de bugs por eso es necesario añadir otro servidor, en nuestro caso: Glassfish Server 4.1)

Tercero, seleccionamos el servidor que deseamos y definimos un nombre para nuestro servidor, damos click en siguiente.



Luego seleccionamos la ubicación donde se descargará e instalará nuestro servidor Glassfish, además seleccionamos que será un **dominio local**, aceptamos los acuerdos de licencia y damos click en **Download Now**, seleccionamos la versión del servidor (dicho anteriormente: Glassfish Server 4.1) y click en aceptar.

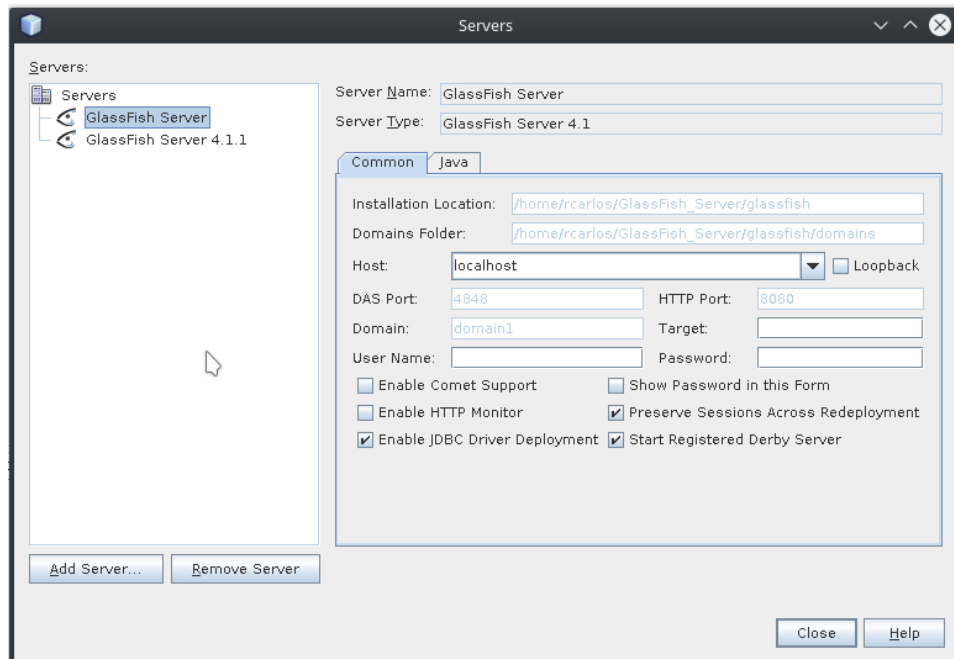


A continuación NetBeans empezará a descargar(100 Mb aproximadamente) los archivos necesarios para la instalación. Al terminar nos aparecerá un mensaje como el mostrado abajo. Damos click en siguiente y completamos los campos que nos pide el asistente, quedando llenado como se muestra en las siguiente imágenes. Finalmente click en finalizar.

The screenshot shows the 'Add Server Instance' dialog box with the 'Server Location' step selected. The 'Steps' list on the left includes '1. Choose Server', '2. Server Location', and '3. Domain Name/Location'. The 'Server Location' section has an 'Installation Location' text box containing '/home/rcarlos/GlassFish\_Server' and a 'Browse...' button. Below this are radio buttons for 'Local Domain' (selected) and 'Remote Domain'. A 'Download Now...' button is next to a checked checkbox labeled 'I have read and accept the license agreement... (click)'. A status message states 'Download & Install completed in 8 minutes. 31 seconds.' At the bottom, a message says 'Detected a GlassFish Server 4.1 install. Click Next to register remote or'. Navigation buttons at the bottom are '< Back', 'Next >', 'Finish', 'Cancel', and 'Help'. A mouse cursor is pointing at the 'Next >' button.

The screenshot shows the 'Add Server Instance' dialog box with the 'Domain Name/Location' step selected. The 'Steps' list on the left includes '1. Choose Server', '2. Server Location', and '3. Domain Name/Location'. The 'Domain Location' section contains fields for 'Domain:' (dropdown menu with 'domain1'), 'Host:' (dropdown menu with 'localhost' and a checked 'Loopback' checkbox), 'DAS Port:' (text box with '4848'), 'HTTP Port:' (text box with '8080' and a checked 'Default' checkbox), 'Target:', 'User Name:', and 'Password:'. A status message at the bottom says 'Register existing embedded domain: domain1'. Navigation buttons at the bottom are '< Back', 'Next >', 'Finish', 'Cancel', and 'Help'. A mouse cursor is pointing at the 'Next >' button.

Como vemos el servidor que hemos agregado (**Glassfish Server**) ahora se encuentra dentro de la lista de servidores de NetBeans. Cerramos la pestalla y ahora procedemos a la configuración del mismo.

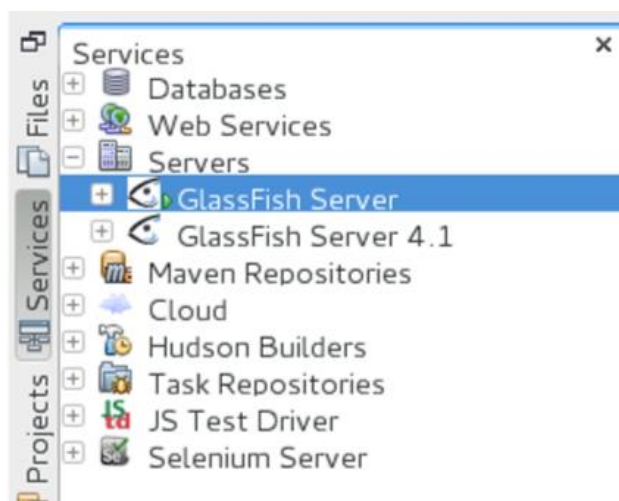


Ahora lo que debemos hacer es descargar el driver de postgresql para JDBC y copiarlo en la carpeta de instalación de Glassfish, dentro del directorio lib(/glassfish/lib) tambien copiarlo dentro de /glassfish/domains/domain1/lib

El driver se puede descargar de: <https://jdbc.postgresql.org/download.html>

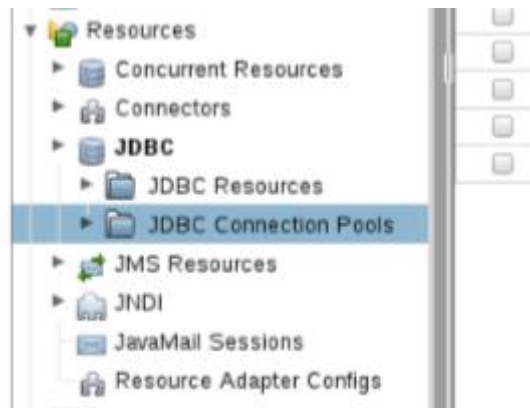
Este driver se debe de descargar de acuerdo a la versión de postgresql que esta siendo utilizada asi como el JDK instalado. Al haber descargado y copiado el driver a las carpetas mencionadas, procedemos a iniciar el servidor desde NetBeans.

Al lado izquierdo de nuestro NetBeans, aparece una pestaña "**Services**" seleccionamos la opcion de **Servers** y veremos nuestros servidores Glasfish, damos click derecho sobre el servidor que creamos y los iniciamos con **Start Server**.

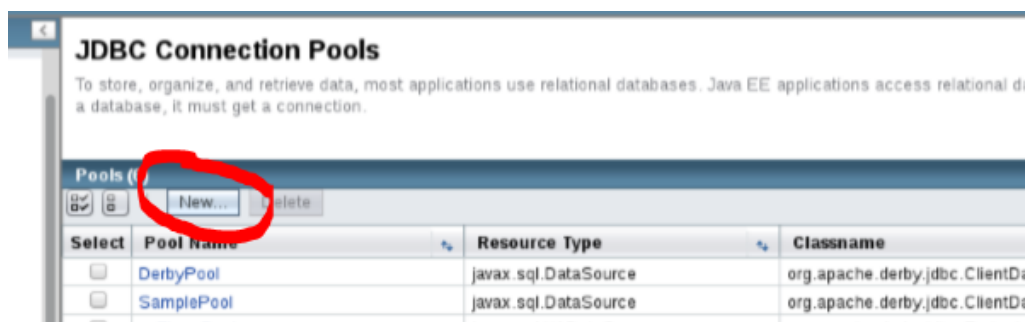


Lo siguiente es esperar a que se despliegue el servidor apareciendo un mensaje de éxito en la pestaña **output** de NetBeans. Una vez iniciado el servidor puedes acceder a el, desde el navegador con la url: **localhost:4848/**

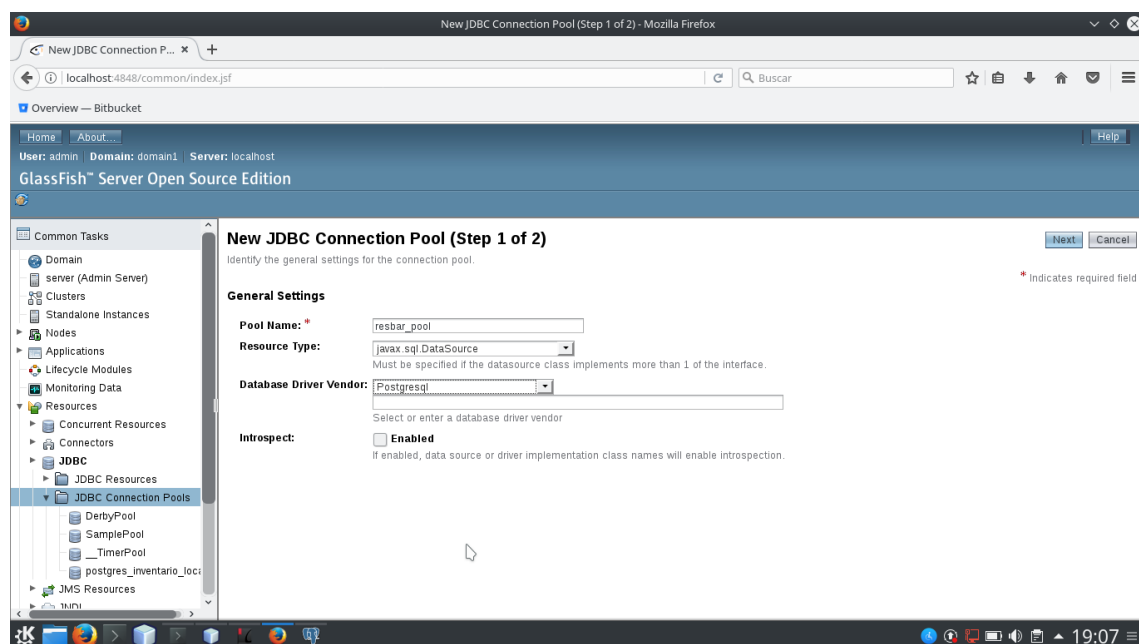
Abierto el servidor en nuestro navegador debemos dirigirnos al menu que se encuentra al lado izquierdo y seleccionamos: **Resources > JDBC > JDBC Connection Pools**



Lo que haremos es crear un pool de conexiones a nuestra base de datos. Damos clic en **New**

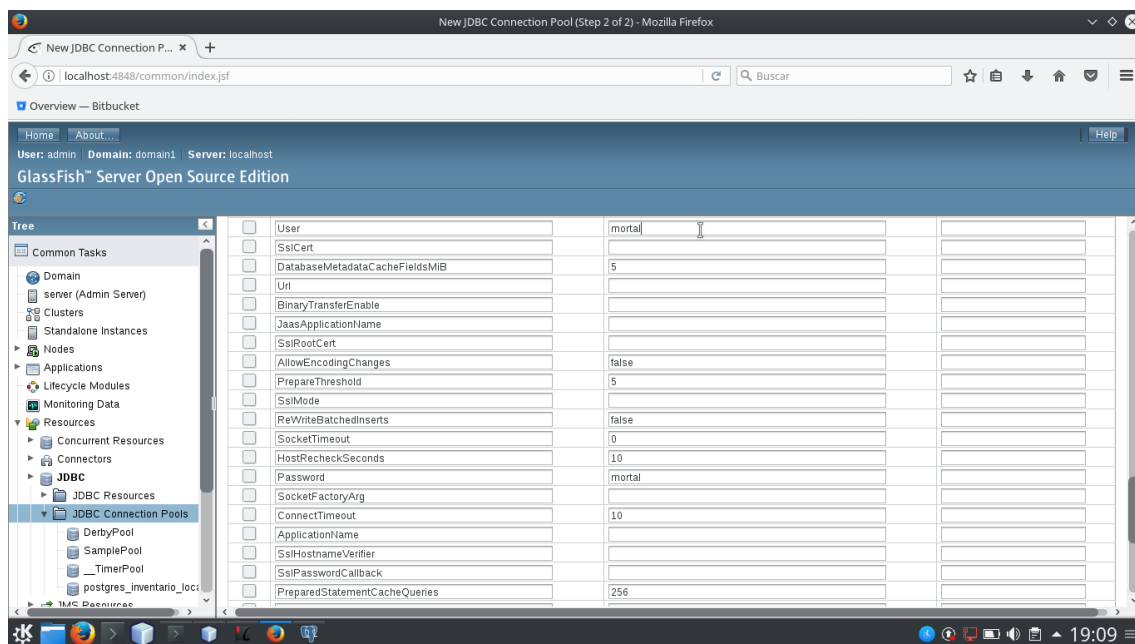


Luego completamos los campos que se nos pide de la siguiente forma:



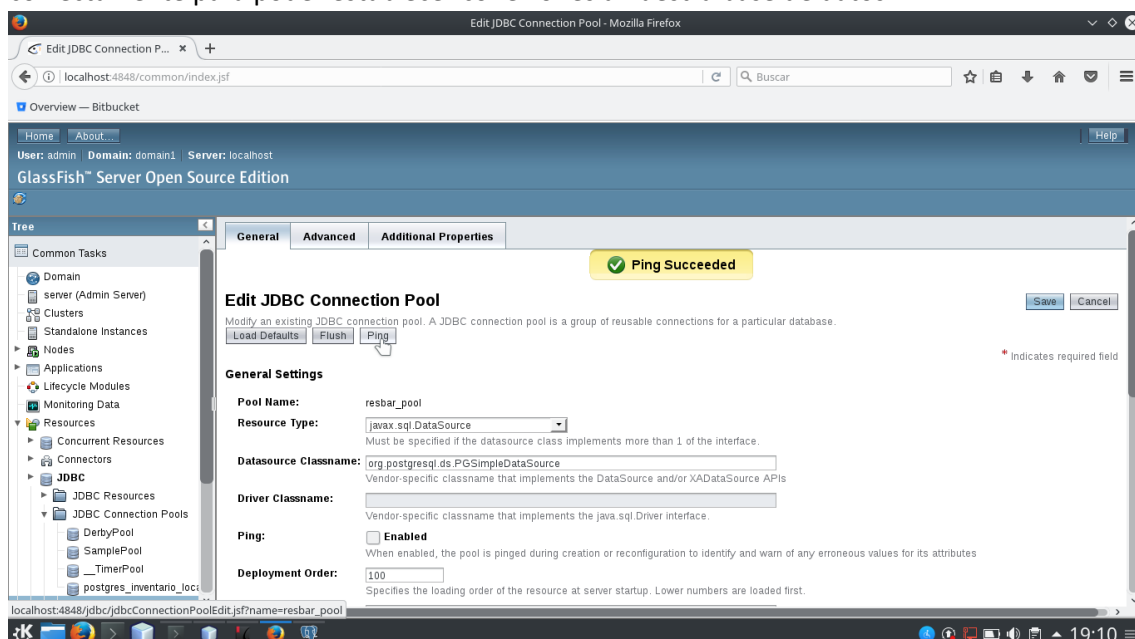
Damos clic en Next y nos apareceran mas opciones de configuracion para el pool de conexiones entre ellas **Pool Settings** donde podemos modificar la cantidad máxima de conexiones que se pueden realizar y tiempos relacionados con las conexiones que se realicen en este pool pero **lo que nos interesa** esta mas abajo que son las propiedades adicionales de la conexión, de las cuales debemos configurar:

**user:** mortal                      **serverName:** localhost                      **databaseName:** resbar  
**password:** mortal                      **portNumber:** 5432                      **url:** (Dejar vacia)

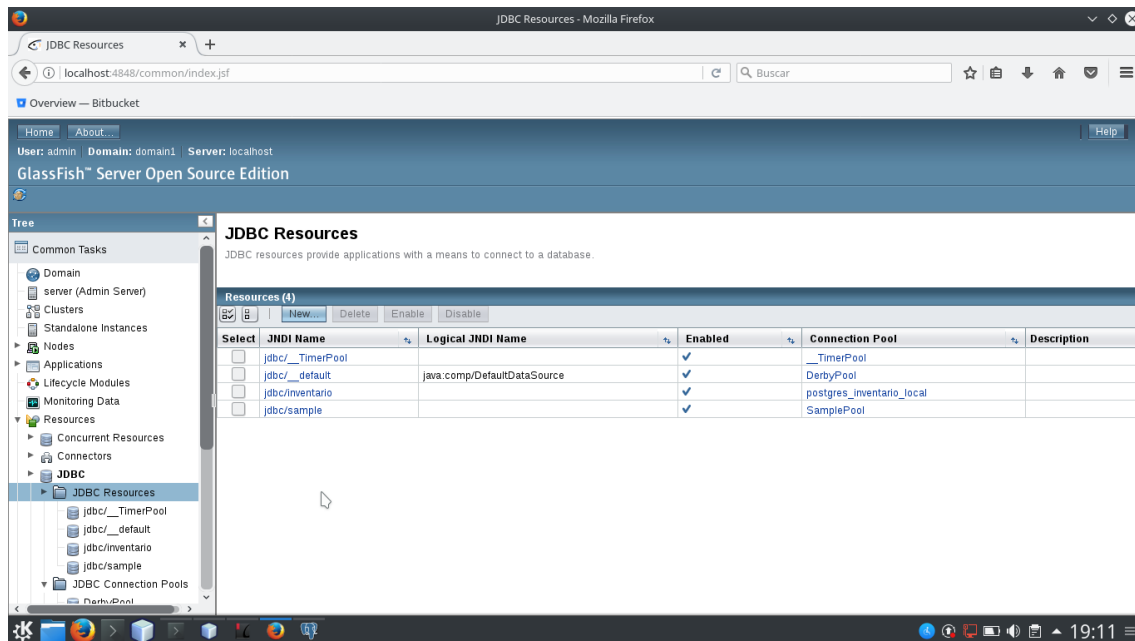


(Captura solo para fines ilustrativos, algunas propiedades a configurar no se alcanzan a ver)

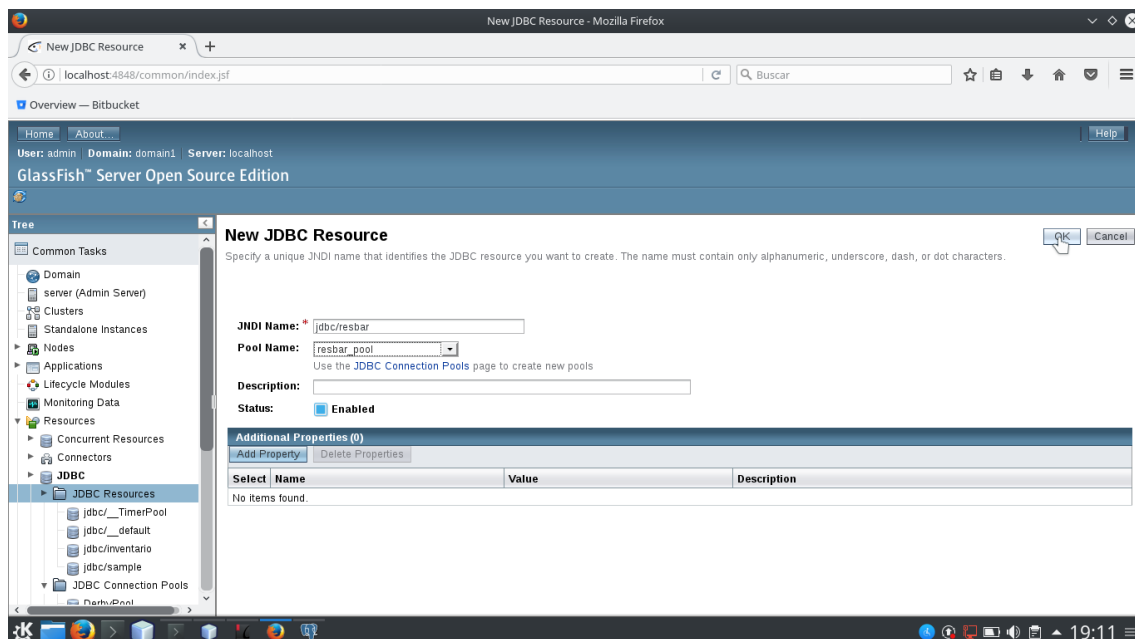
Al haber llenado los campos necesarios, damos clic en **Finish** y nuestro pool de conexión ya nos aparecerá registrado, seleccionamos el pool que acabamos de crear y damos click en **Ping**, nos deber aparecer un mensaje de “**Ping succeeded**” comprobando así que todo se realizó correctamente para poder establecer conexiones a nuestra base de datos.



Luego nos dirigimos al menu y seleccionamos **Resources > JDBC > JDBC Resources** y damos clic en **New**



Una vez hecho click en **New** se nos pedirá llenar unos campos, en los cuales asignaremos un nombre al recurso y seleccionaremos el pool de conexión creado anteriormente.

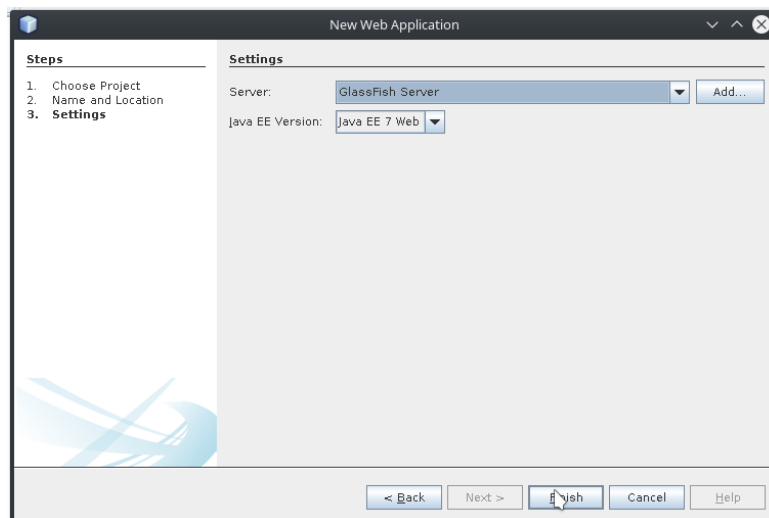
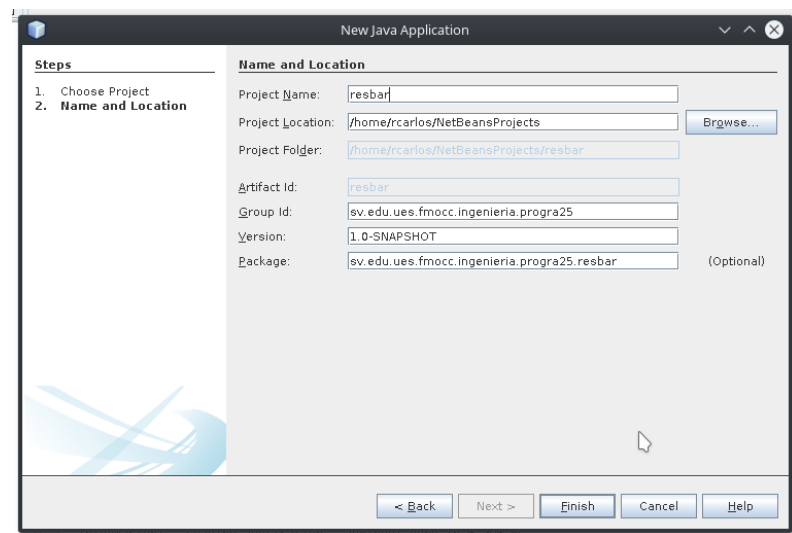
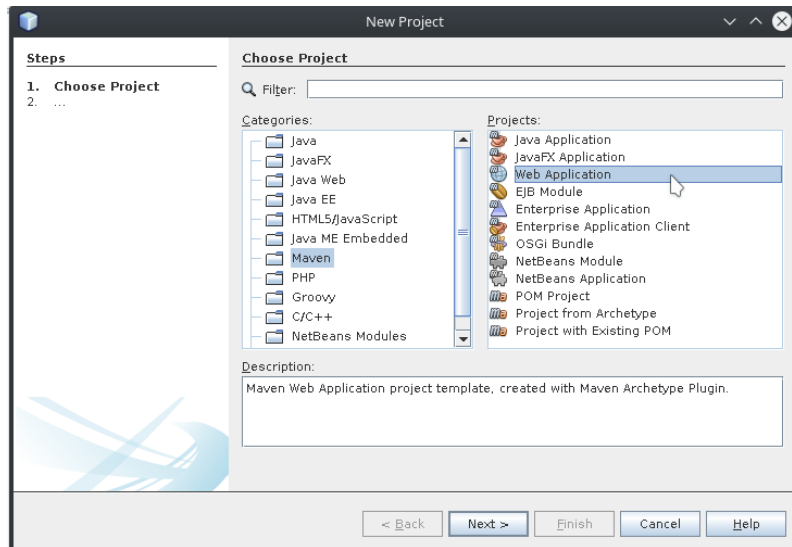


Para finalizar damos click en **OK** y nos aparecerá ya registrado nuestro nuevo recurso.

Hecho todos los pasos anteriores ya tenemos nuestro servidor configurado para poder ejecutar en el las aplicaciones que queramos y necesitemos.

## Creando un nuevo proyecto

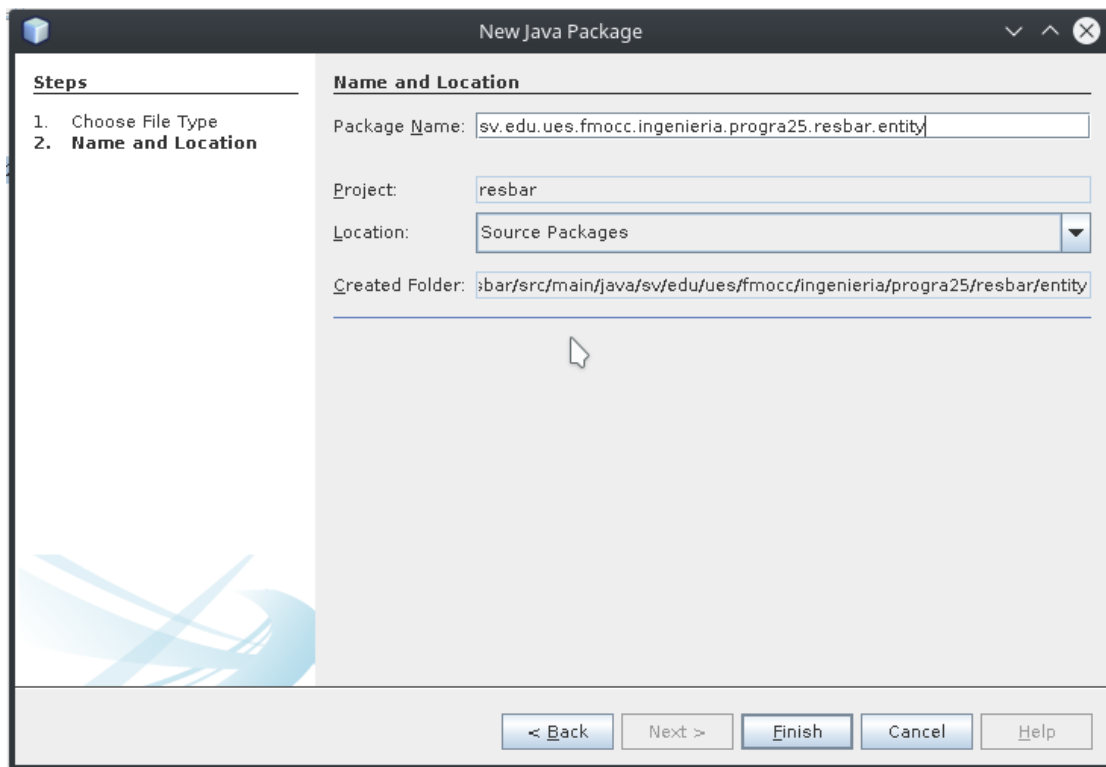
A modo de ejemplo se creará un nuevo proyecto con maven:



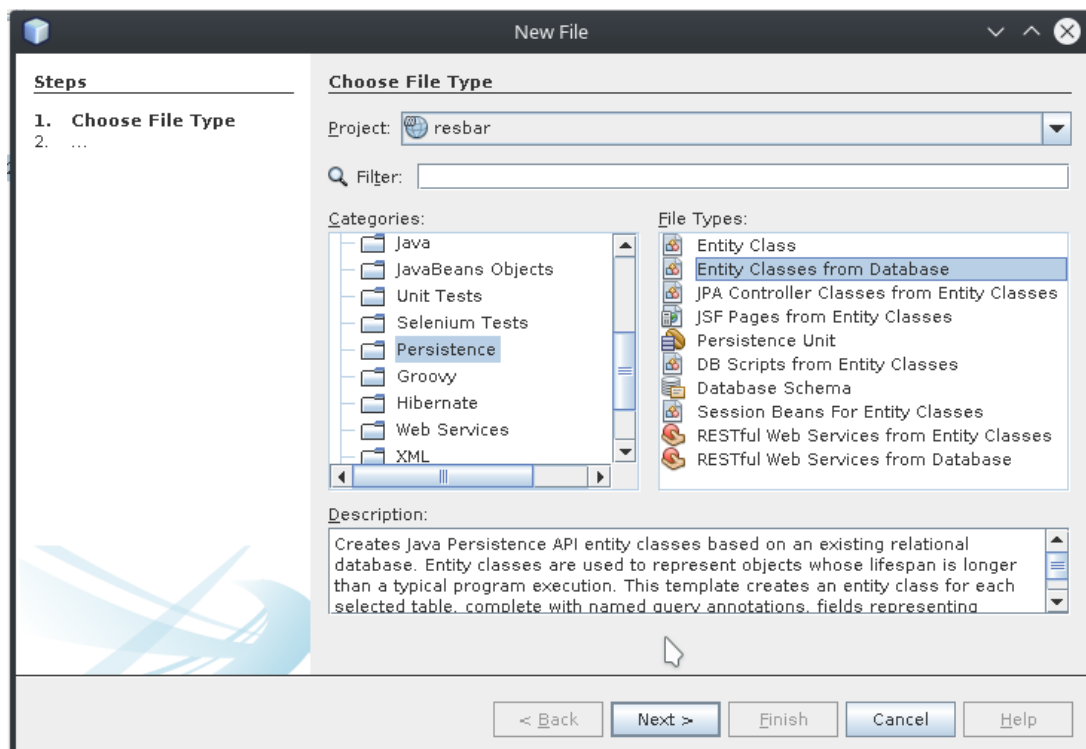
(Se debe seleccionar el server creado al inicio de la guía)



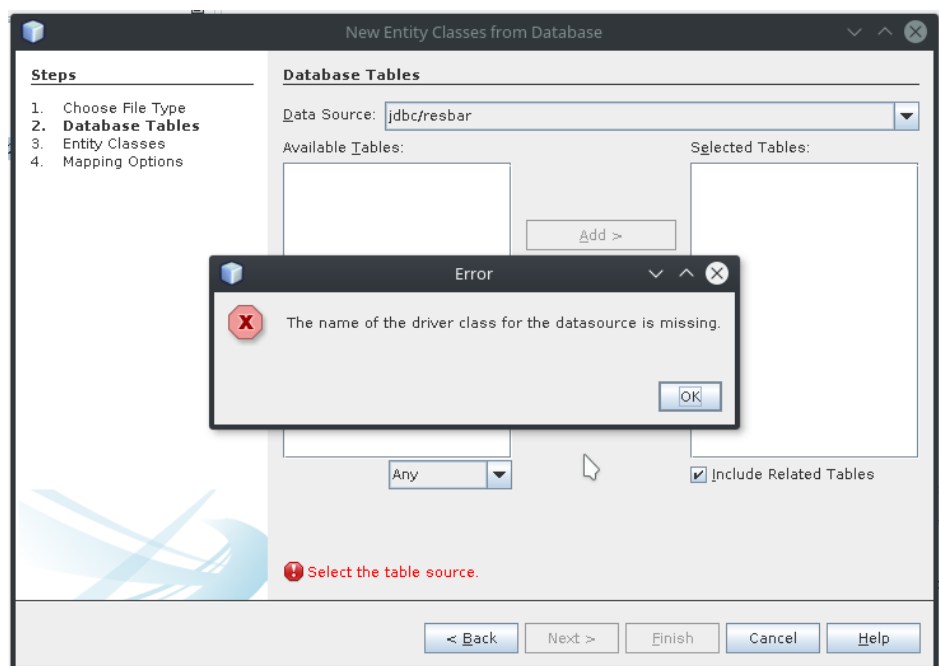
Creamos un nuevo paquete:



El paquete creado anteriormente servira para el mapeo de la base de datos con JPA y una vez creado sobre el proyecto damos **Click derecho > New > Other** y en el asistente abierto seleccionar:



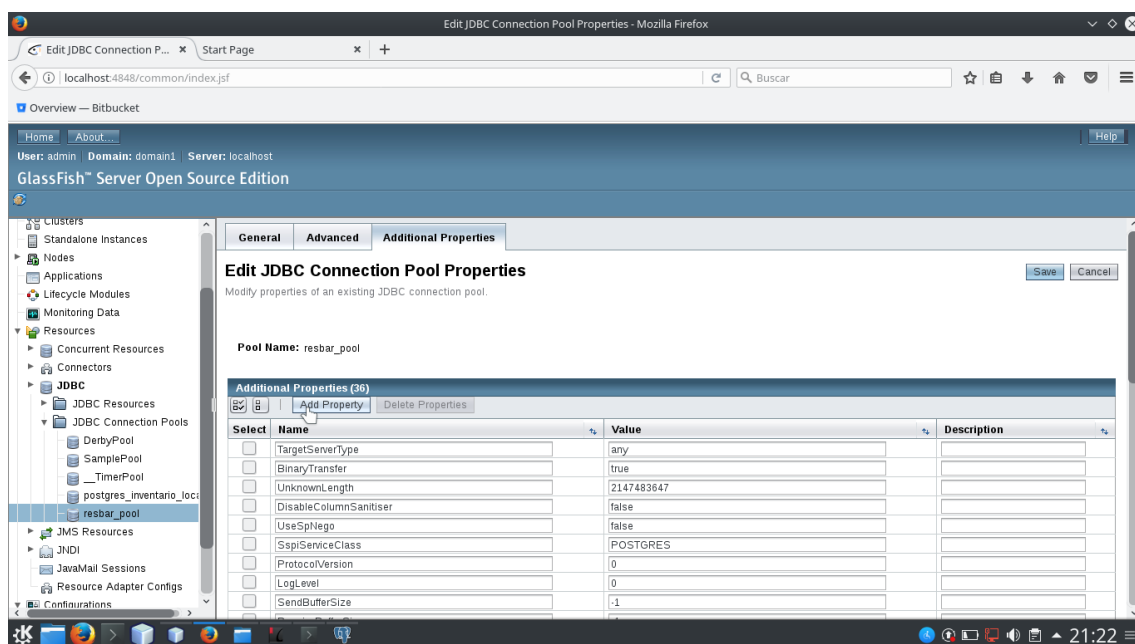
En la siguiente ventana se debera seleccionar el datasource **"jdbc/resbar"** que creamos cuando estabamos configurando el servidor:



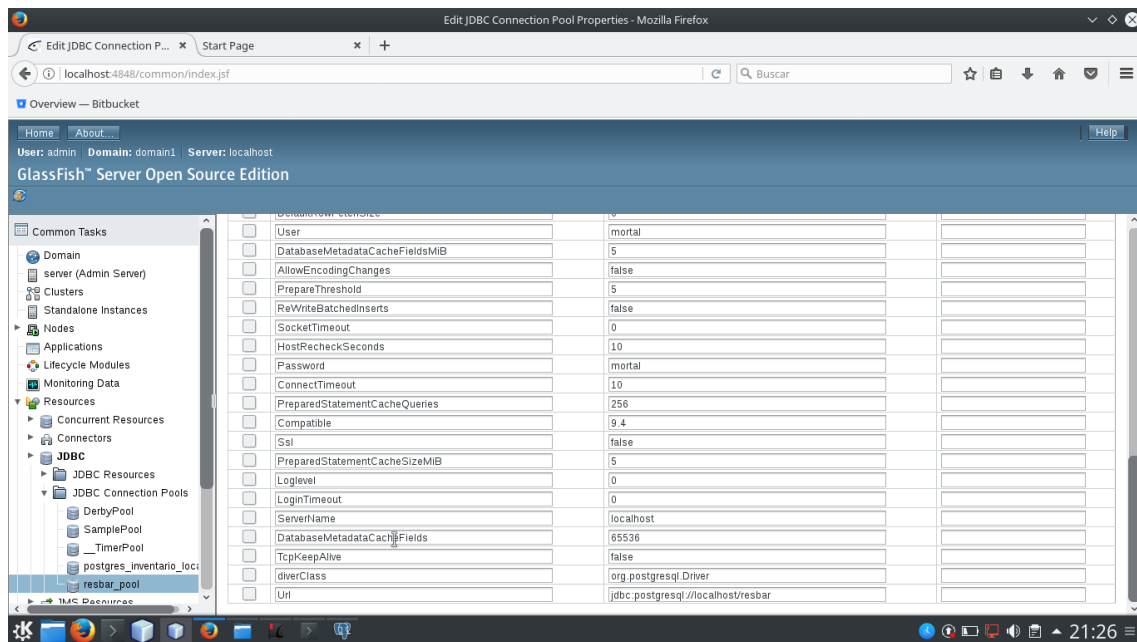
Como se puede ver en la captura hay un error y googleando sobre el se puede concluir que es mal configuración del pool de conexiones creado en el servidor.

NOTA: En proyectos anteriores me ha bastado la configuración ya realizada y claro ejemplo de ello es que cuando hicimos “Ping” sin problemas devolvio el mensaje “Ping succeeded” pero como no es el caso, procedamos a darle solución al problema.

SOLUCIÓN: Acceder a la url **localhost:4848/** desde el navegador, dirigirnos al menú de Glassfish **Resources > JDBC > JDBC Connection Pools** seleccionar “resbar\_pool” y dar click en la pestaña **Additional Properties** luego click en **Add Property**:

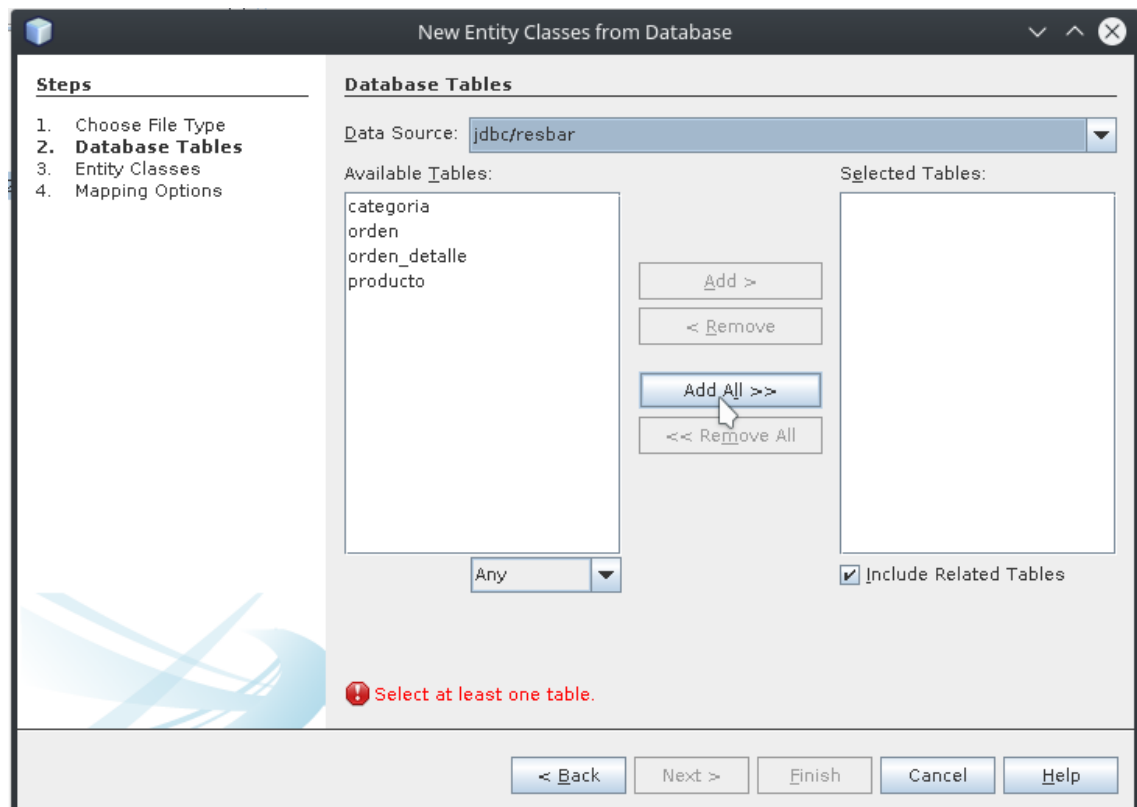


En este punto añadiremos 2 propiedades: **driverClass** y **Url** como se puede ver en la siguiente captura junto con sus valores a configurar(ultimos 2 de la lista):



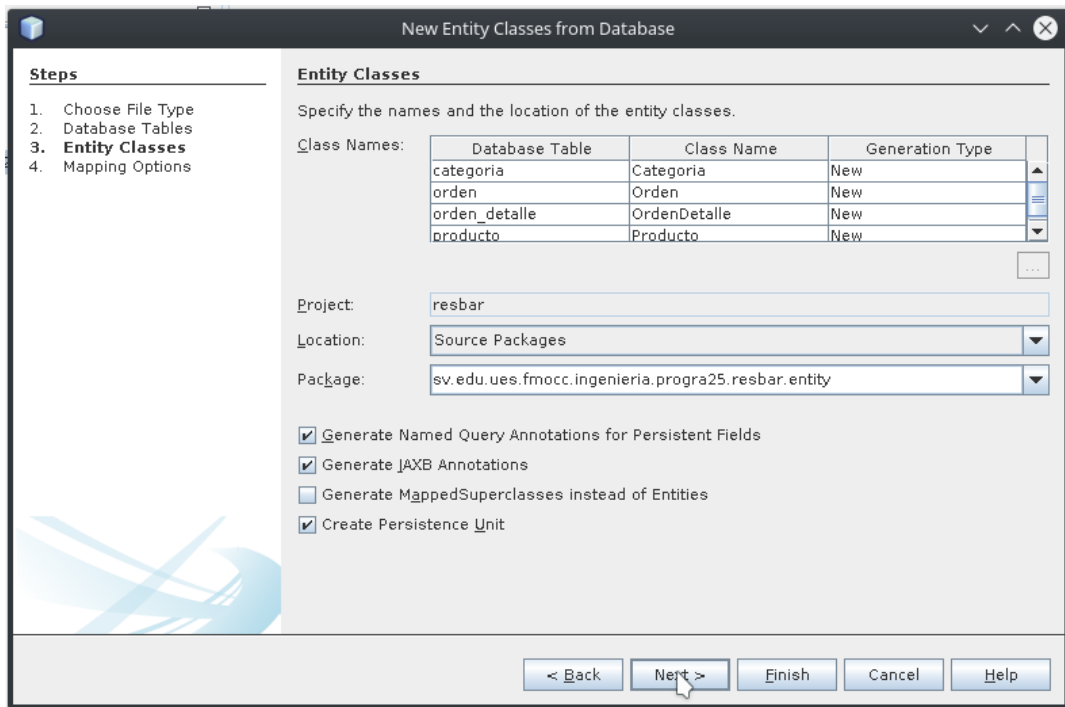
Esto debería ser suficiente para solventar el problema así que continuemos...

Efectivamente sin problemas podemos ver las tablas disponibles en la base de datos, damos click en **Add All** seguido de click en **Next**

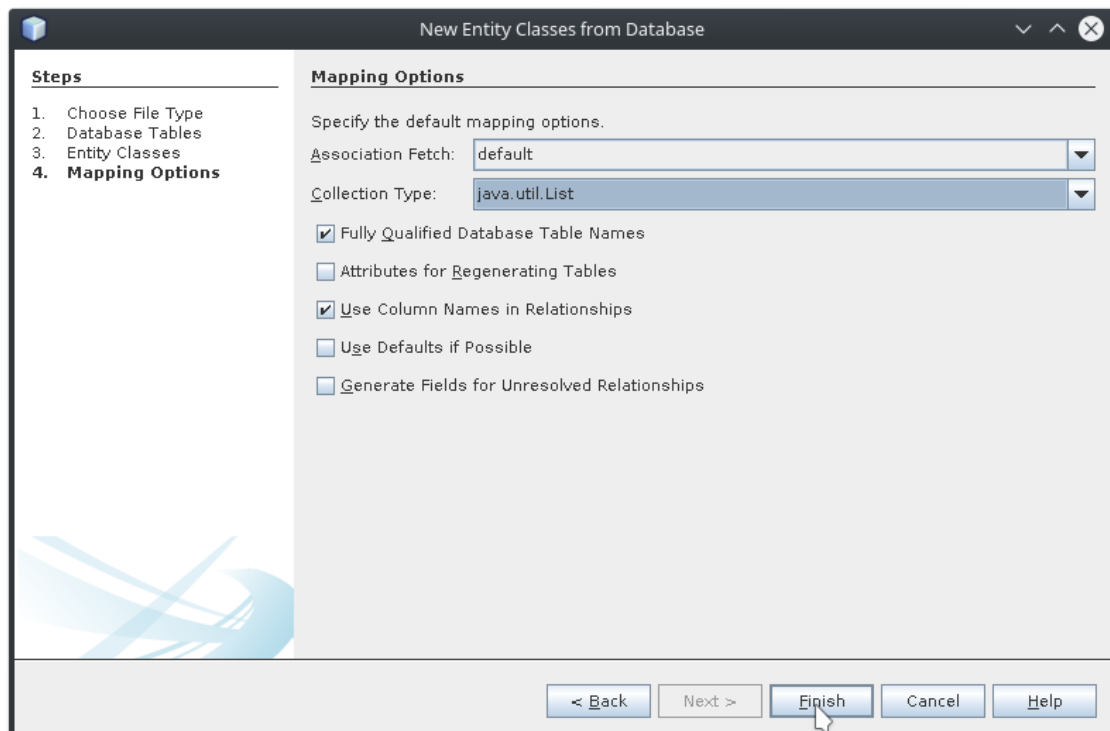


En la siguiente ventana se deja todo con los valores por defecto:

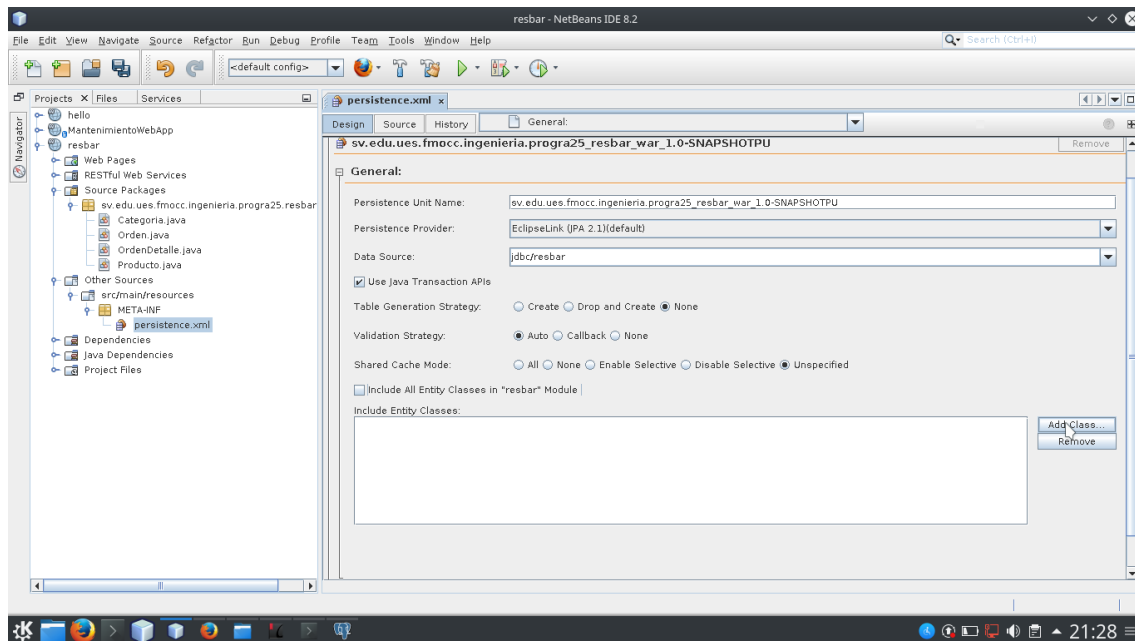
NOTA: Recalcar que el paquete donde a seleccionar debe ser el que creamos al inicio, esto para mantener un orden y una estructura bien definida en el proyecto.



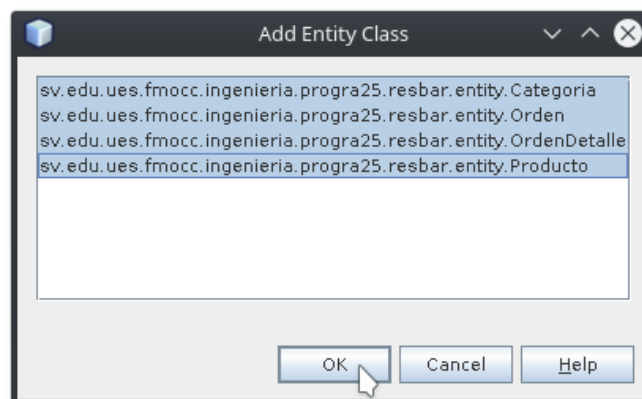
En la siguiente ventana en **Collection Type** se debe seleccionar java.util.List y marcar **Fully Qualified Database Table Names** asi:



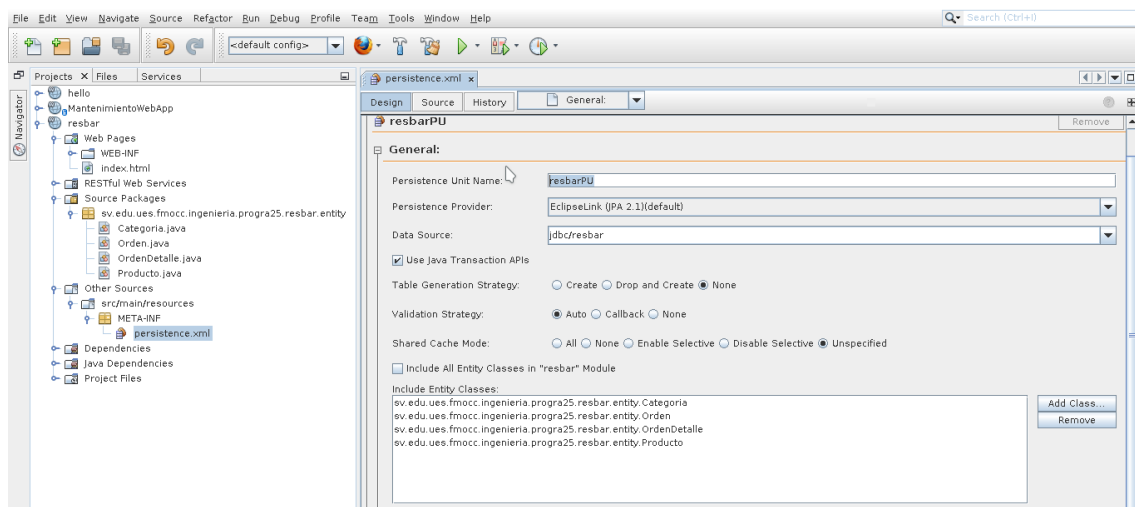
Si todo salio bien al darle **Finish** el resultado obtenido deberia ser un proyecto con la siguiente estructura(izquierda) con un archivo "persistence.xml" donde deberan desmarcar **"Include All Entity Classes in "resbar" Module"** y dar click en **Add Class**:



Les aparecera un cuadro de dialogo como el que se muestra a continuación en donde se debe seleccionar todas las clases y dar click en **OK**.

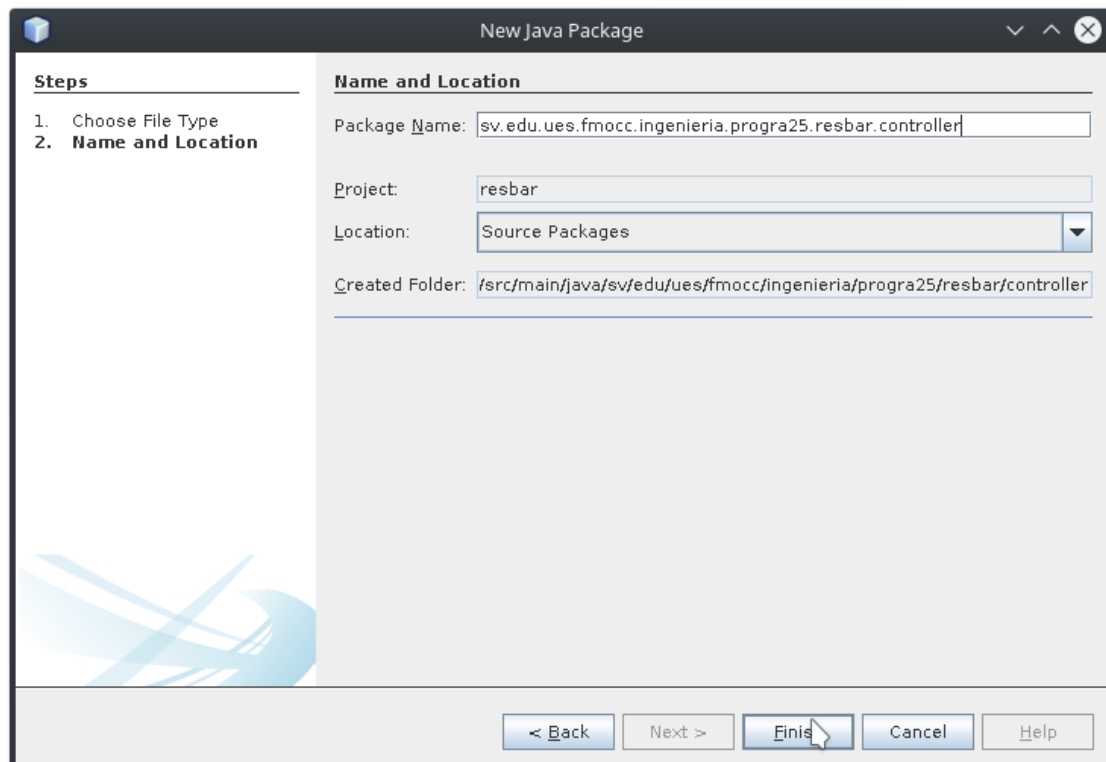


Luego solo queda cambiar el nombre de la unidad de persistencia(pesistence.xml) por un nombre mas facil de recordar:

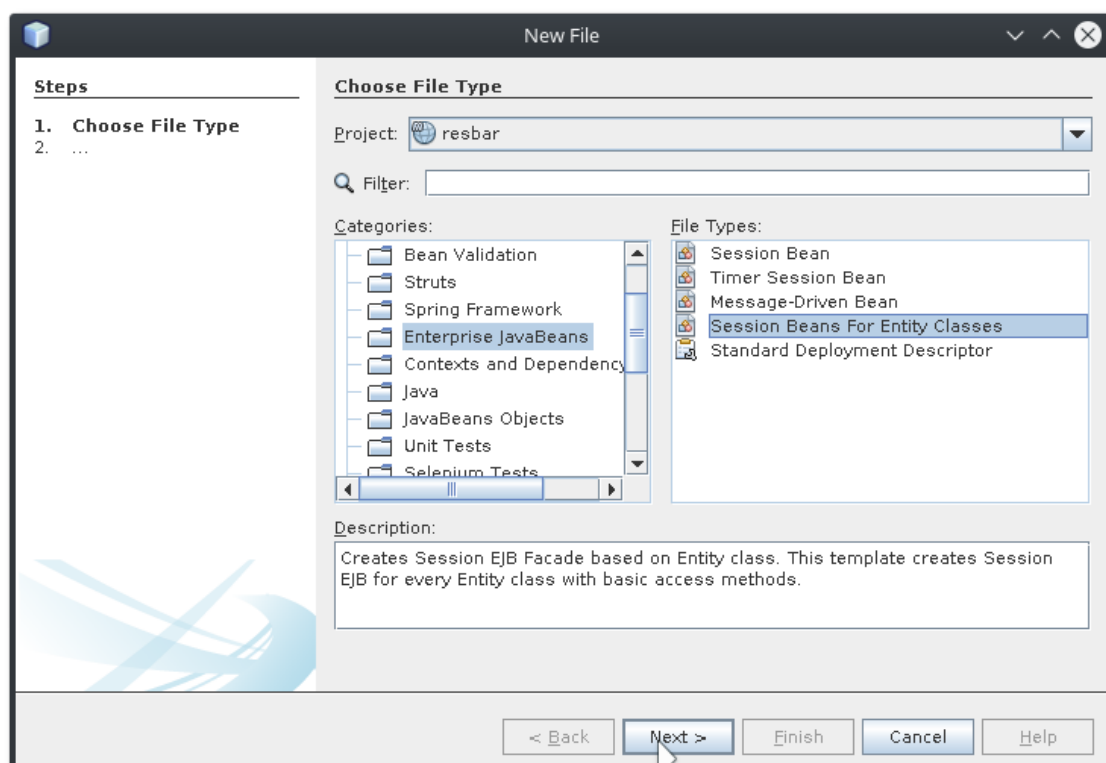


NOTA: Como buena practica, este seria el momento ideal para hacer commit de que se ha realizado(opcional).

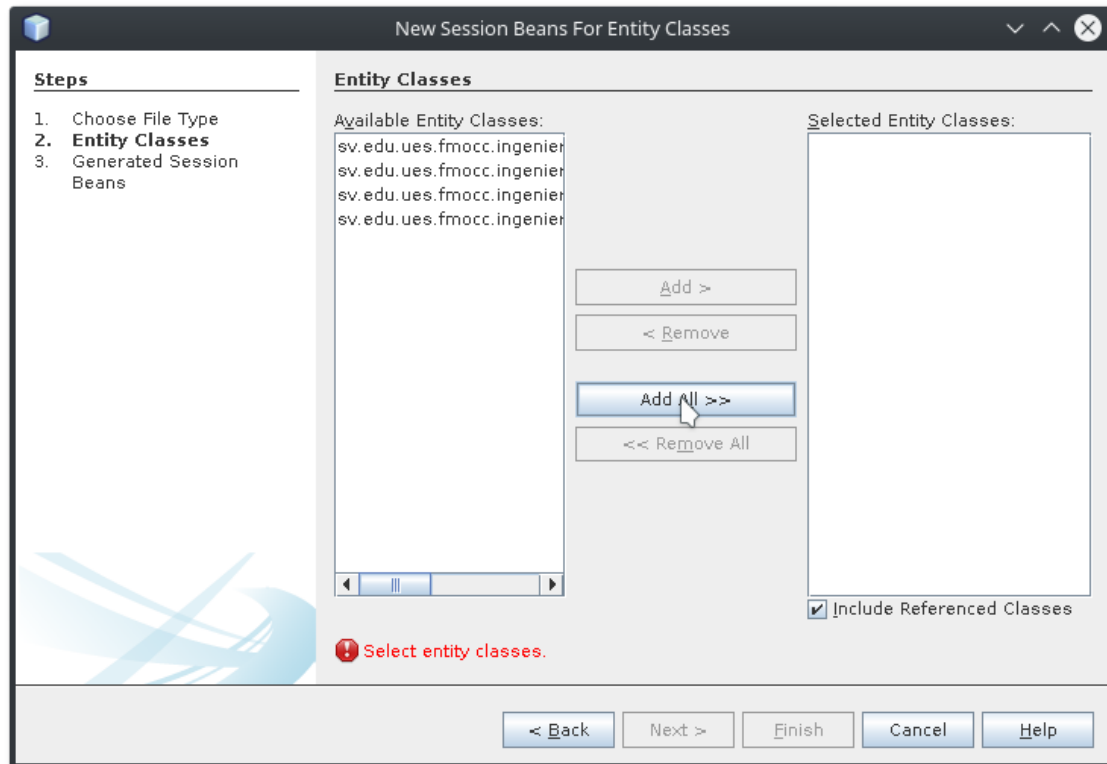
Para continuar, creamos un nuevo paquete:



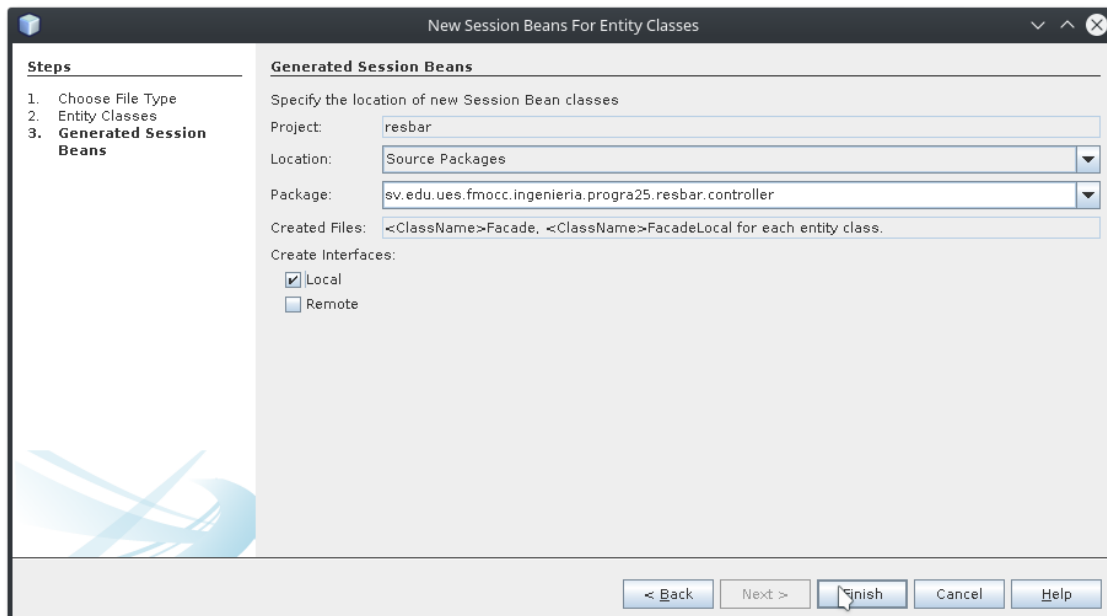
Siguiente: Sobre el proyecto damos **Click derecho > New > Other** y en el asistente abierto seleccionar:



En la siguiente pantalla deberían aparecer todas las entidades mapeadas con anterioridad, se debería seleccionar la opción **Add All** y click en **Next**



Continuando con el uso del asistente: En la opción **Create Interfaces** seleccionar **Local** y clic en **Finish**



Al finalizar el uso del asistente(vaya forma de programar xd) deberian tener la siguiente estructura en su proyecto:

