

Ingeniería de Software 2018-2

Guia de instalación

Jonathan Barragán Jiménez

21 de febrero de 2018
Facultad de Ciencias UNAM

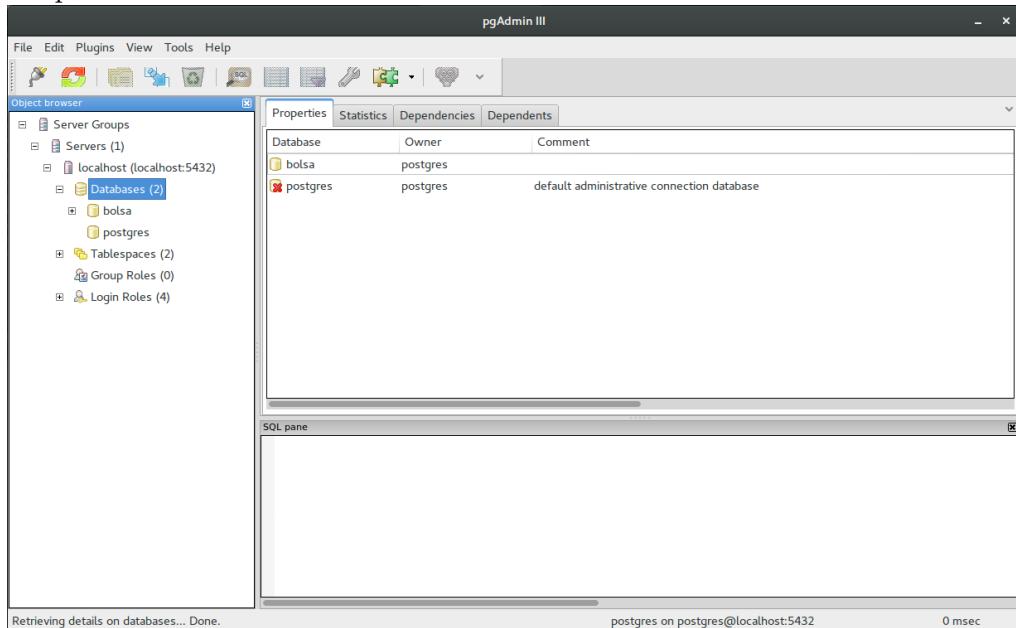
1. Objetivo

Ayudar a los estudiantes a configurar la tecnología que se ocupara para el curso de ingeniería de software 2018-2. Esta guía se hizo en Ubuntu 16.04, toda la tecnología esta disponible para cualquier sistema operativo.

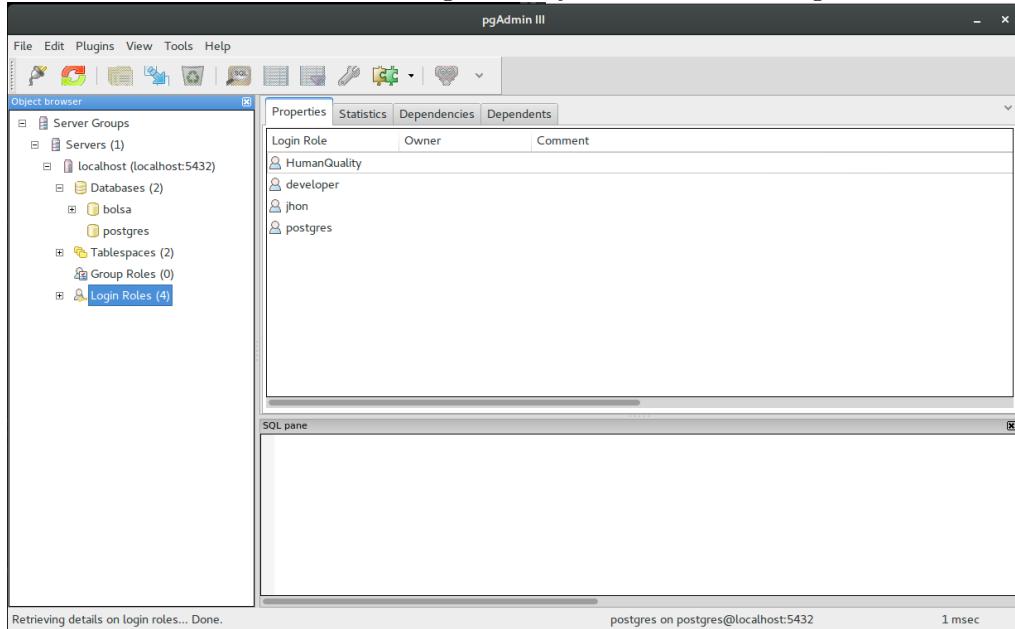
2. Configuración del manejador de base de datos PostgreSQL y pgadmin

Suponemos que postgresql y pgadmin ya están instalados en su versión 9.5 y 3 respectivamente, es decir que ya creamos una nueva conexión en pgadmin con el usuario *postgres* por omisión que configuramos en la instalación de postgresql. Suponemos que también creamos una base de datos de la forma que se vio en clase.

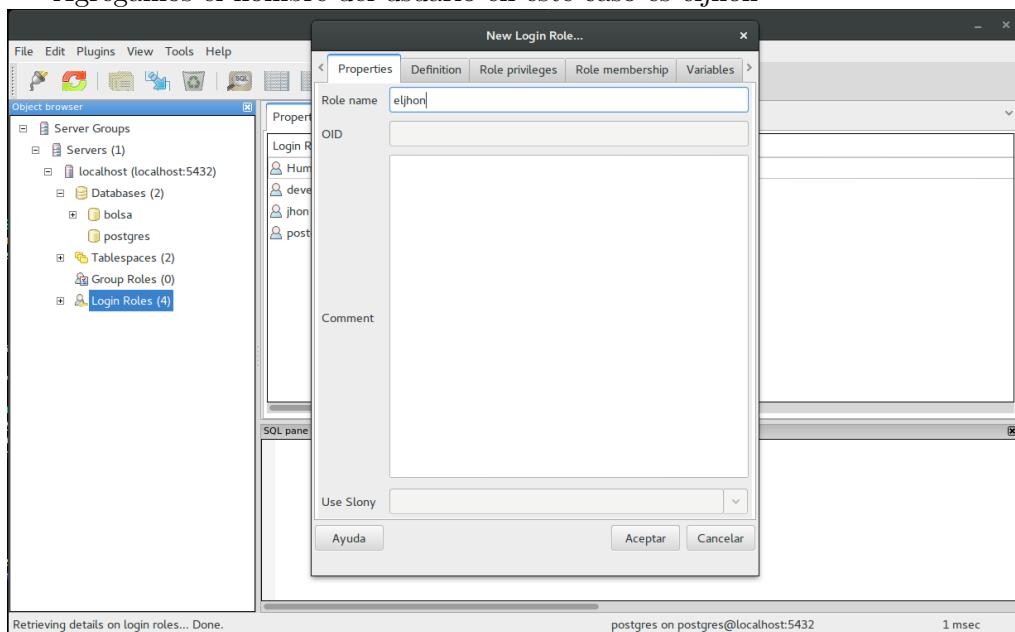
Aquí solo hay que crear un nuevo usuario con todos los privilegios, este usuario será con el que accederemos a la base de datos desde netbeans.



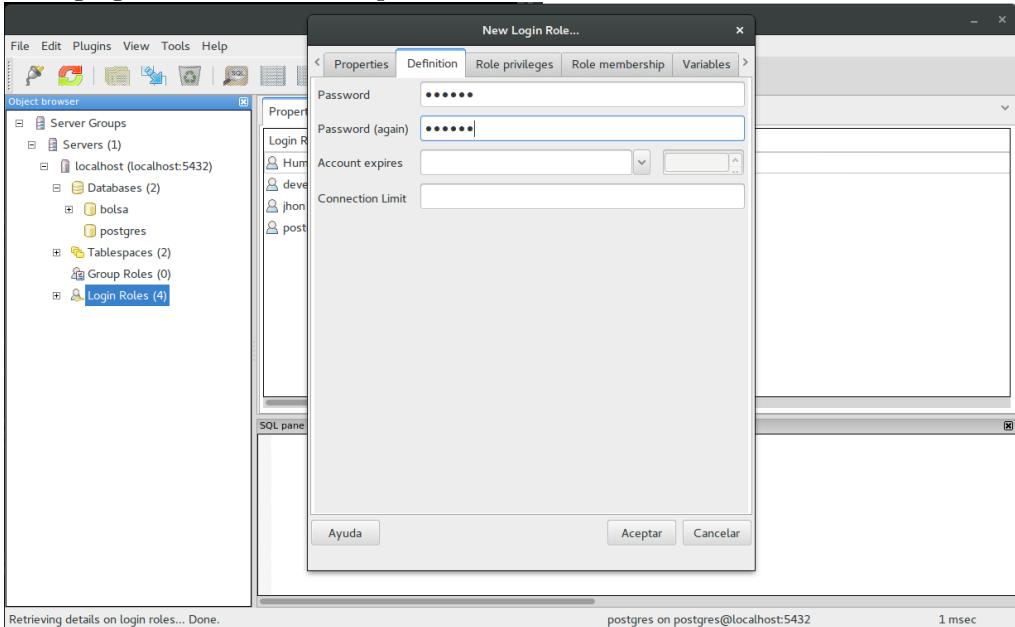
Le damos clic derecho sobre login roles y le damos a new login roles.



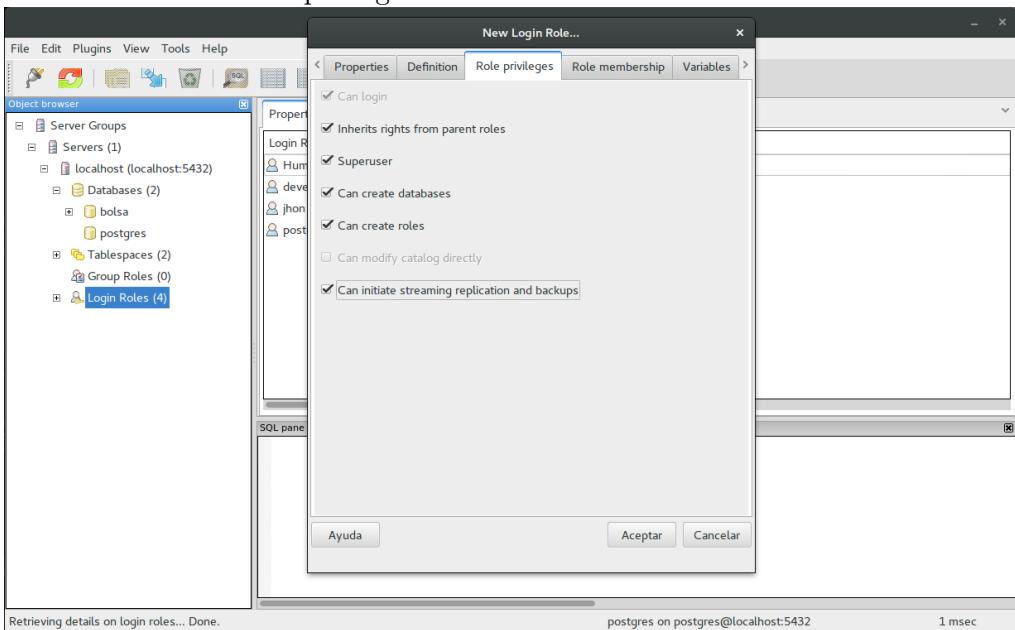
Agregamos el nombre del usuario en este caso es *eljhon*



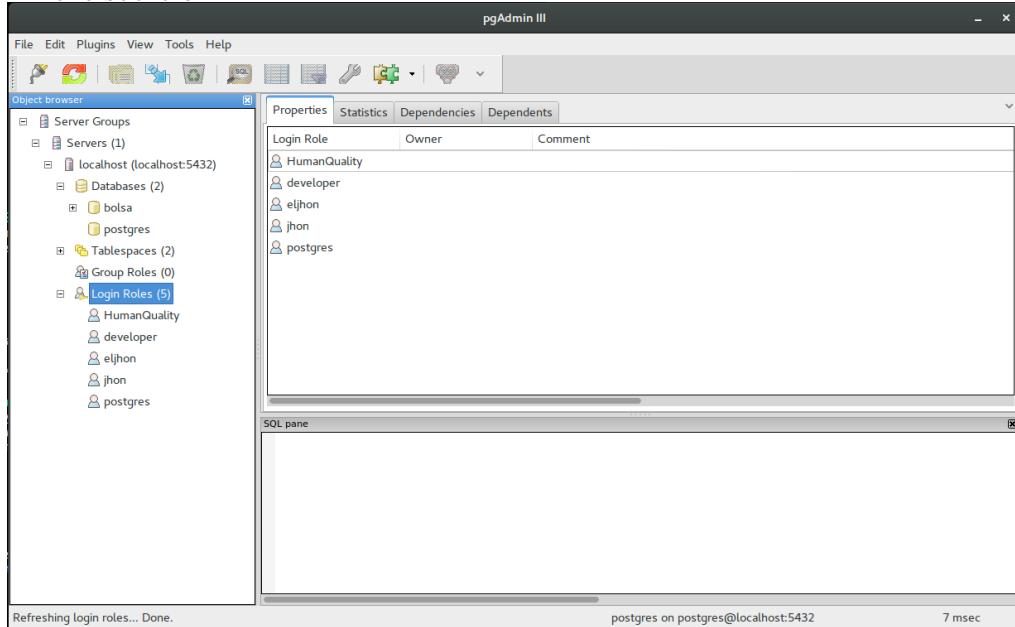
Agregamos la contraseña para el usuario



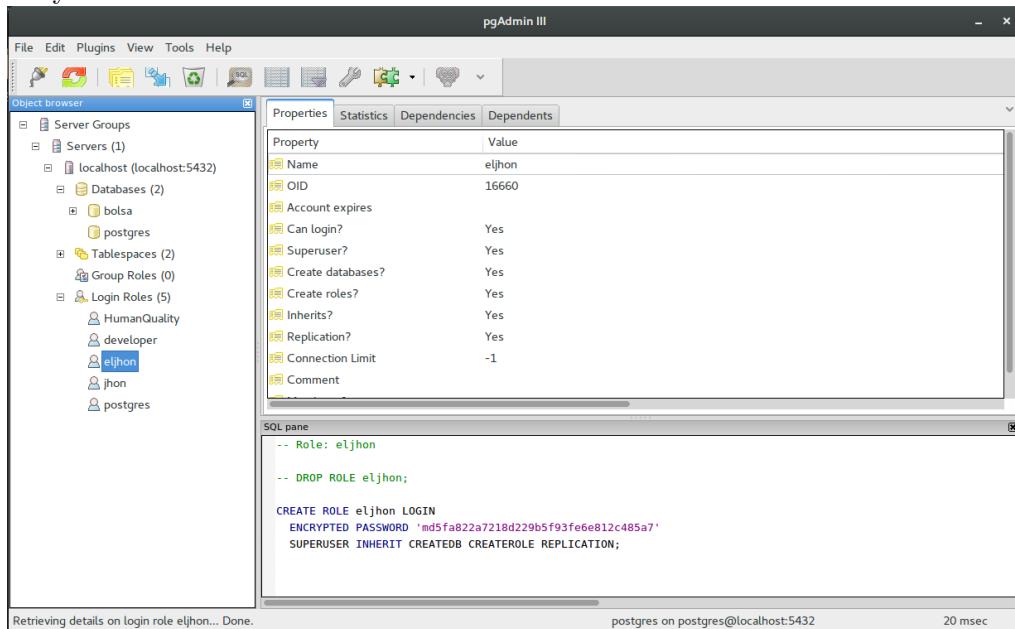
Le damos todos los privilegios al usuario



clic sobre ok



ya esta creado el nuevo usuario

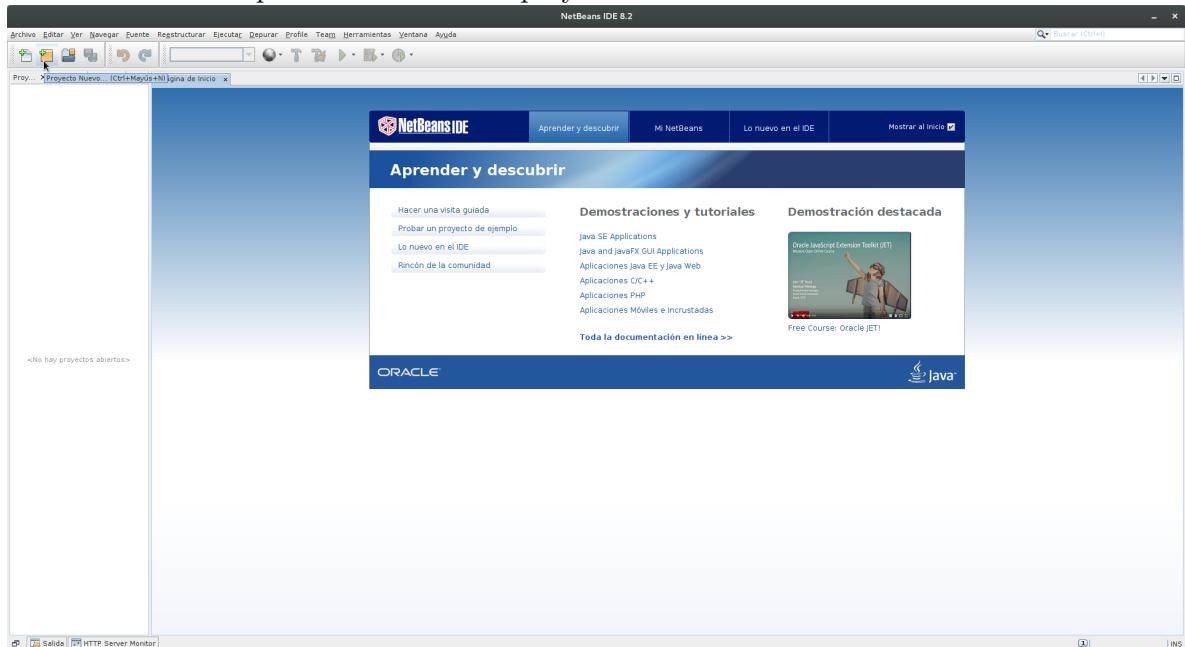


3. Creación de un nuevo proyecto en NetBeans con todos los frameworks

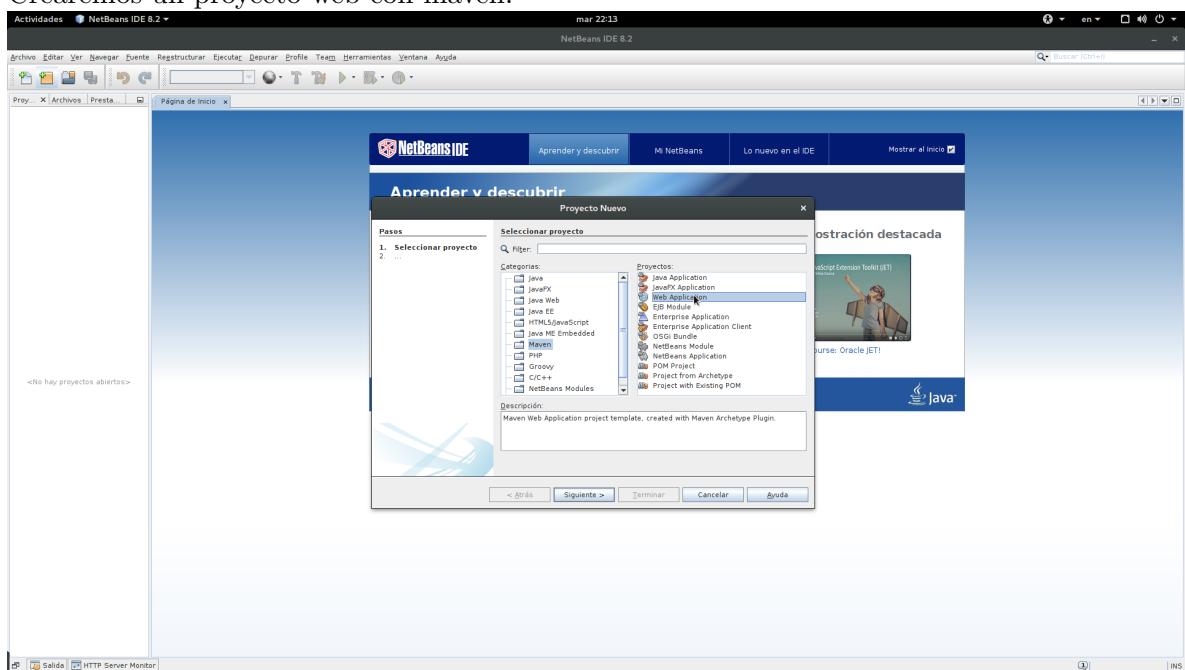
Suponemos que ya netbeans esta instalado de manera correcta en su version 8.2

3.1. Creación de un nuevo proyecto en NetBeans

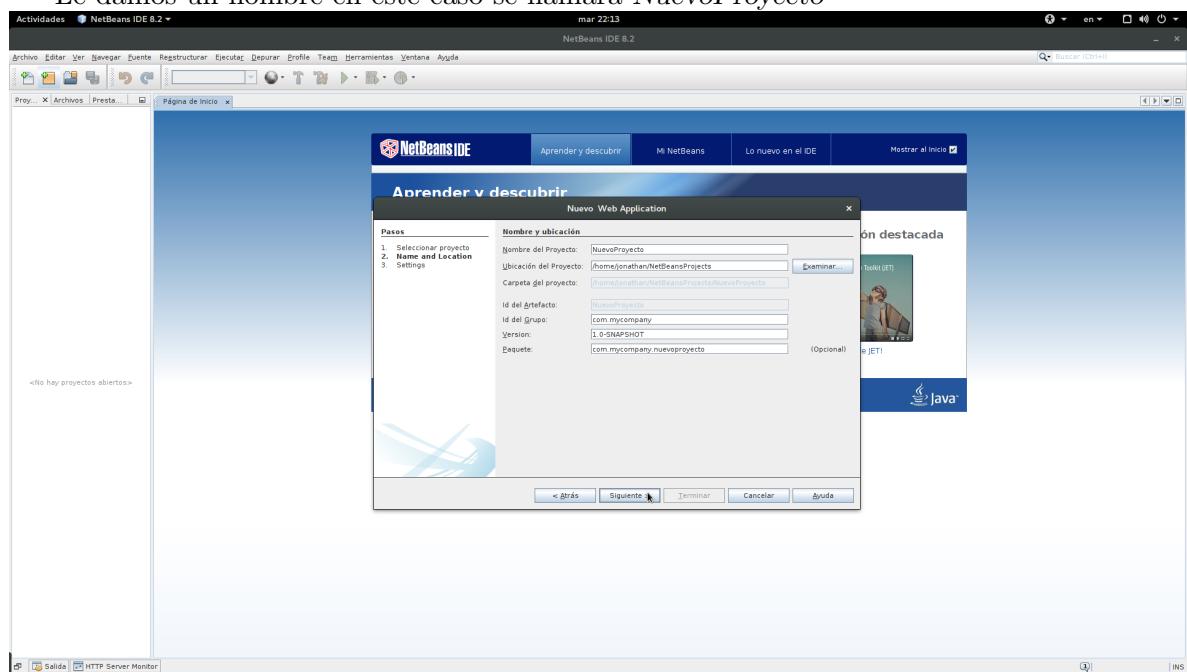
Clic sobre el folder para crear un nuevo proyecto



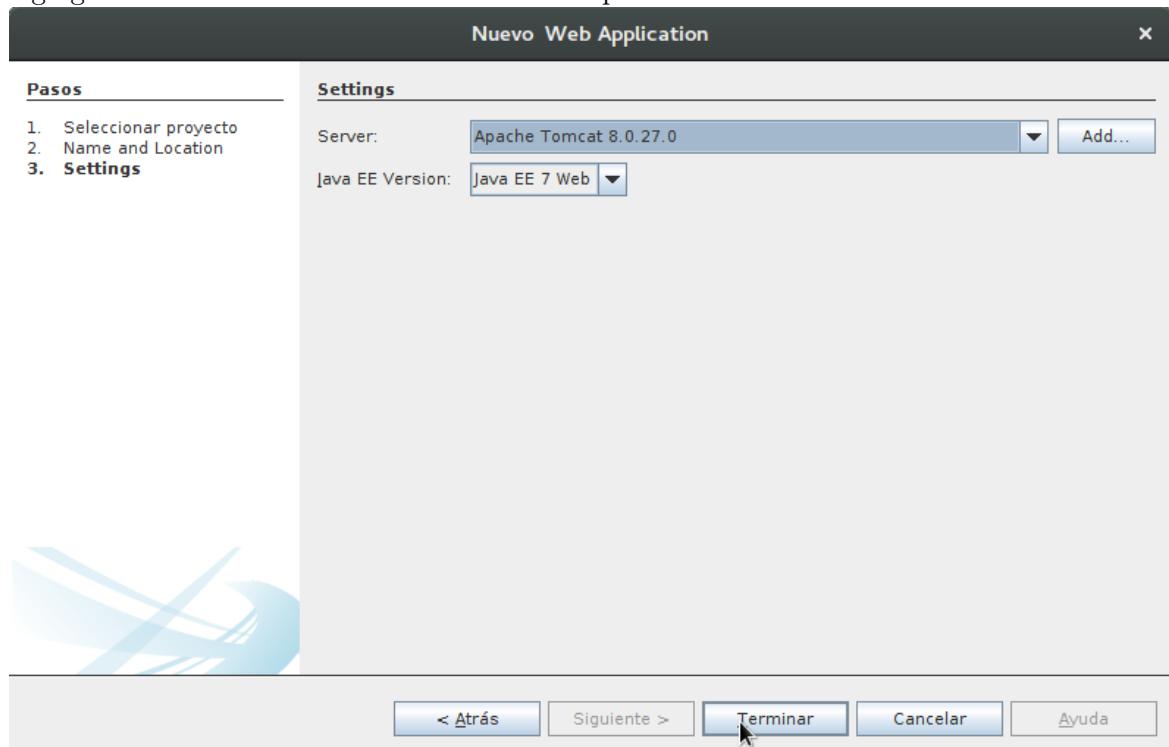
Crearemos un proyecto web con maven.



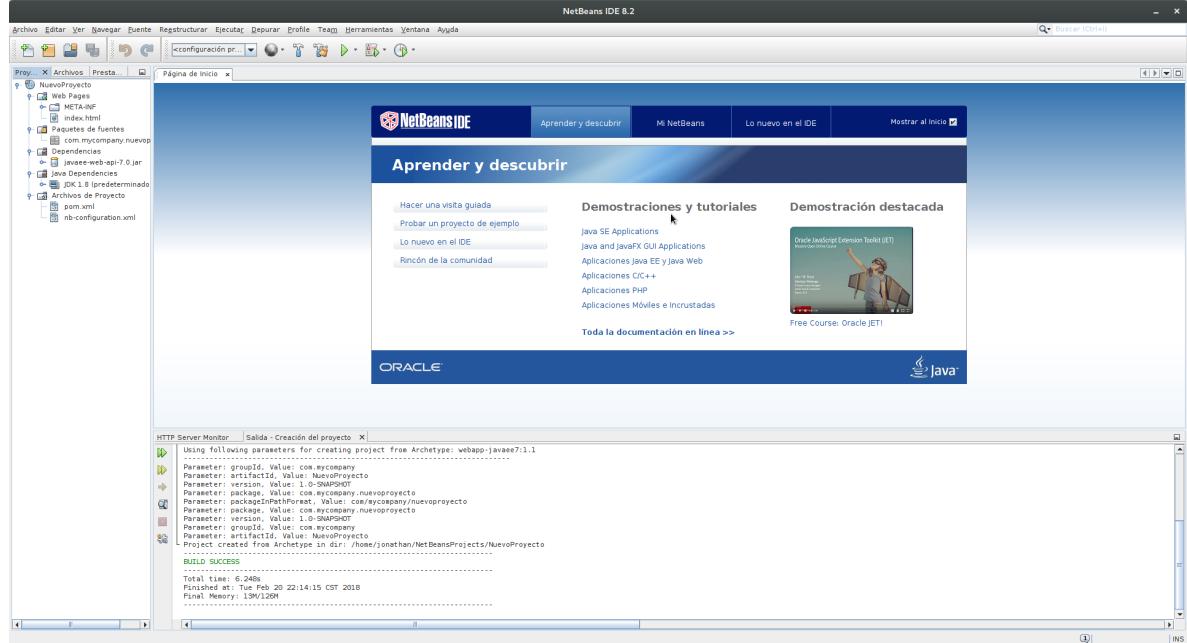
Le damos un nombre en este caso se llamará *NuevoProyecto*



Agregamos el servidor web en este caso sera Apache Tomcat 8.0.27.0

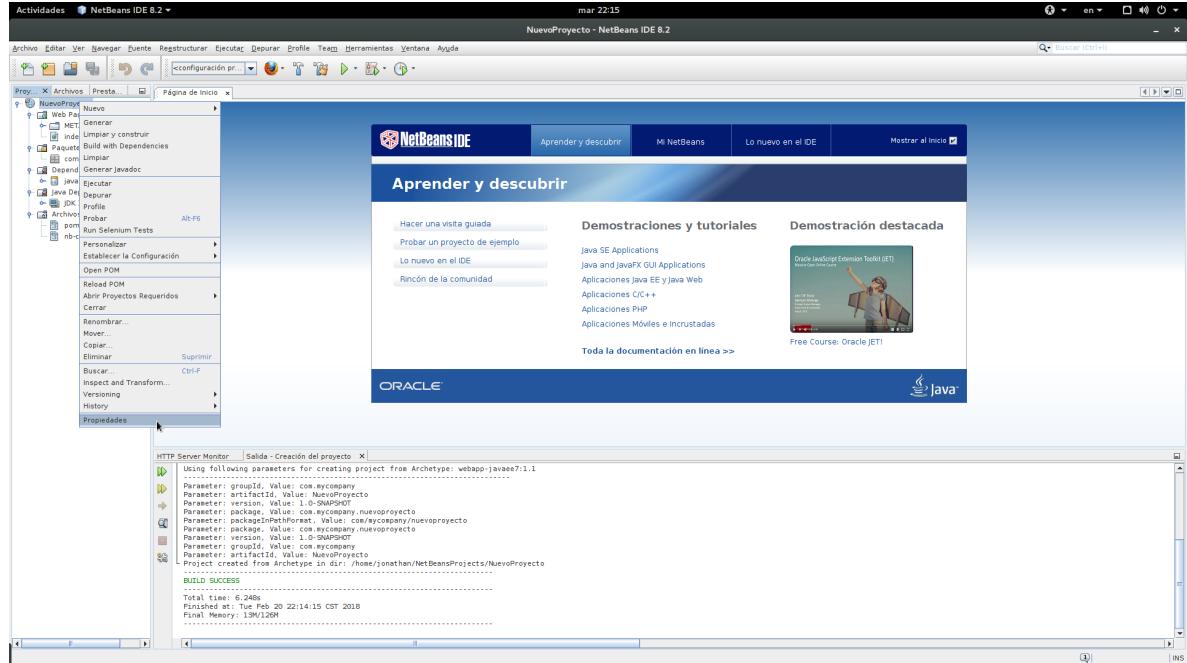


Clic sobre terminar y asi se crea el poryecto. **Siempre se debe estar conectado a internet.**

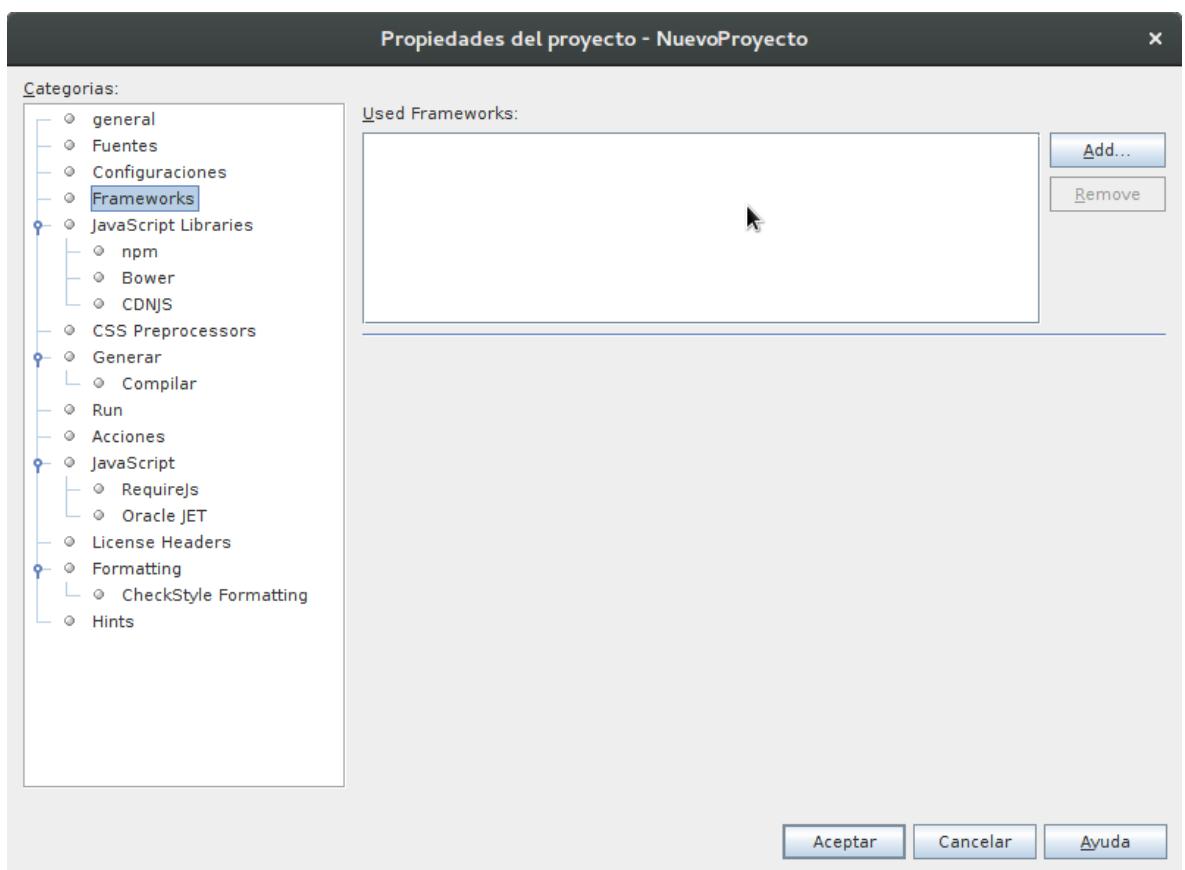


3.2. Agregamos los framework

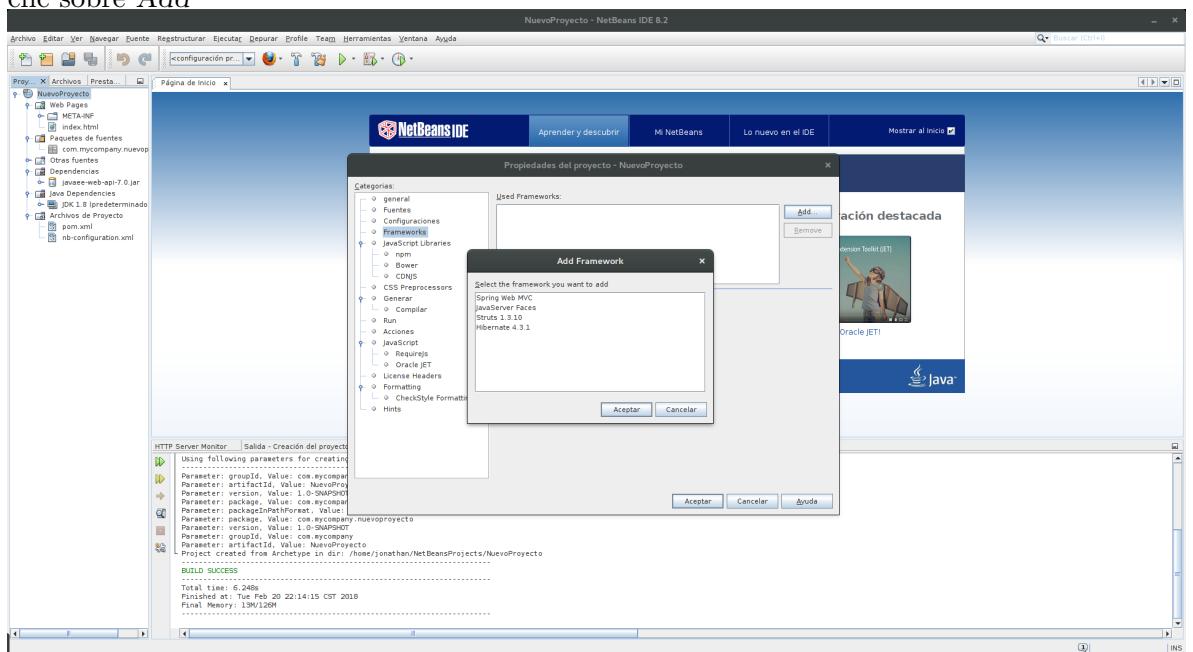
Clic derecho sobre el nombre del proyecto y clic sobre propiedades



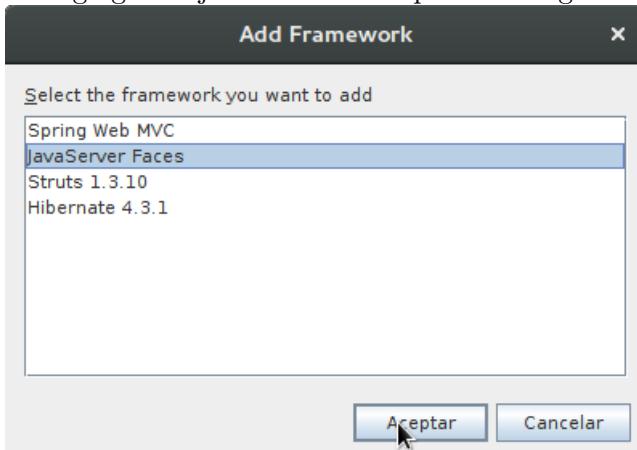
Clic sobre *frameworks*.



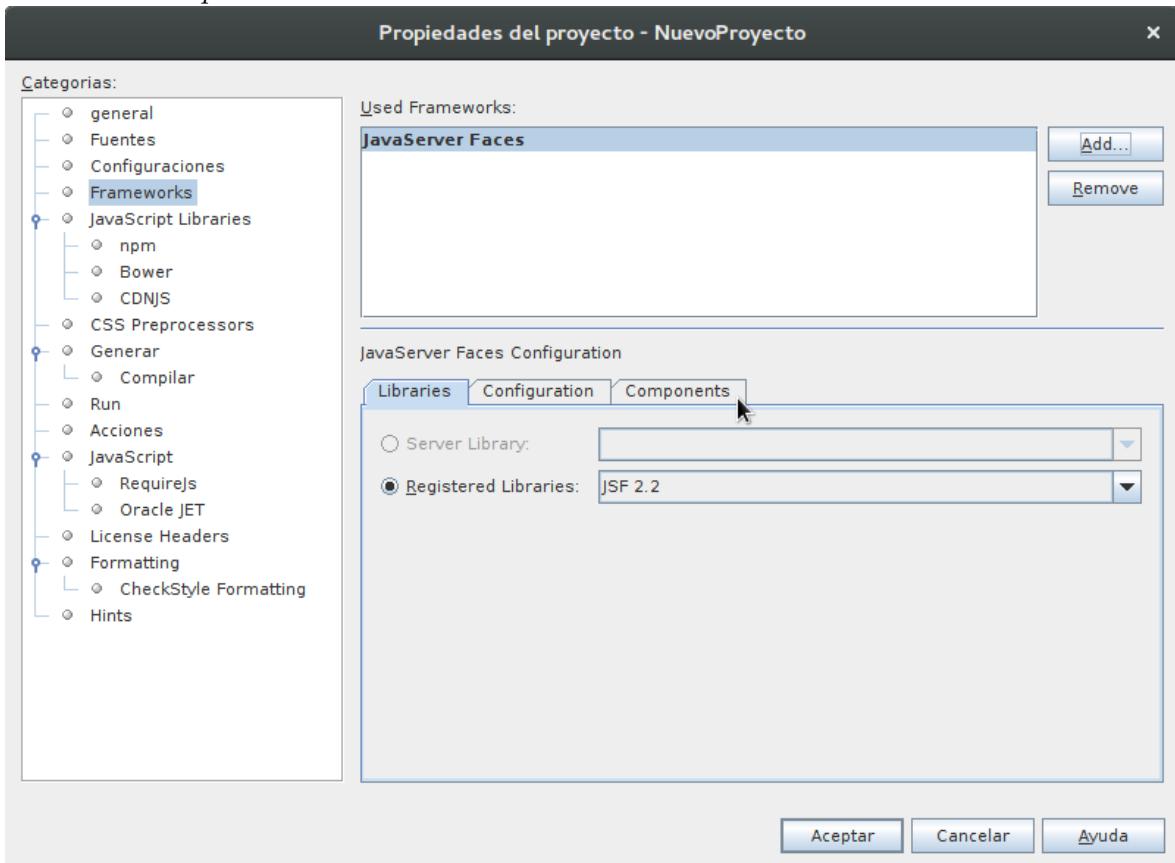
clic sobre Add



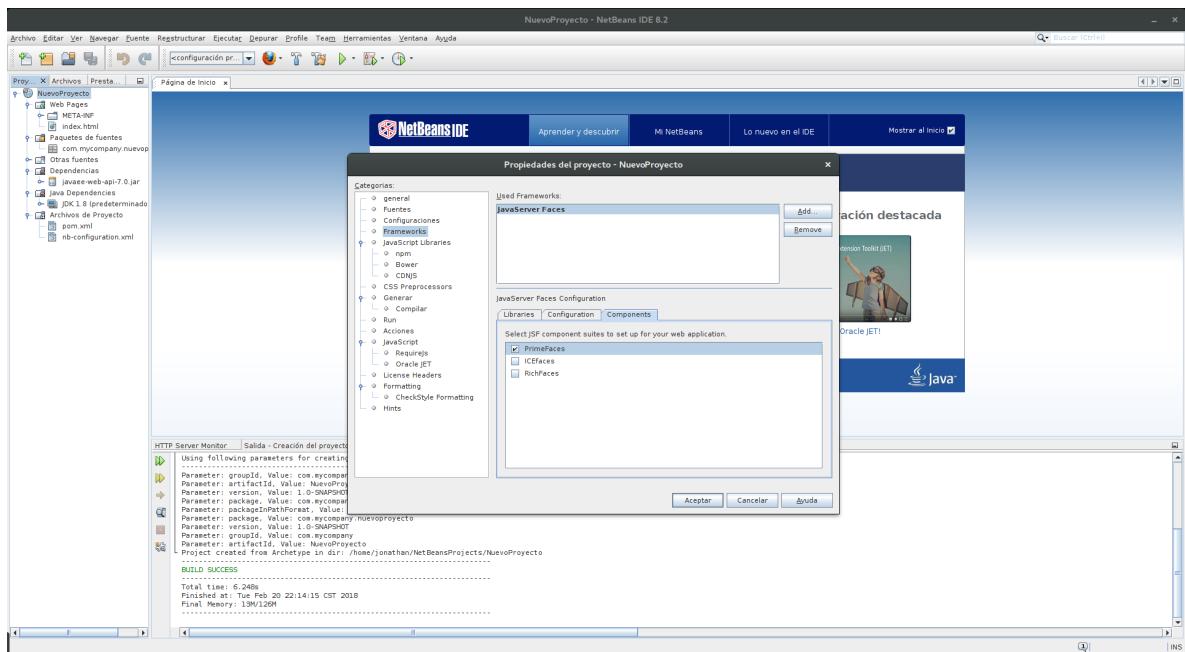
Agregamos javaServerFaces que se encargará de las peticiones a nuestra aplicación web



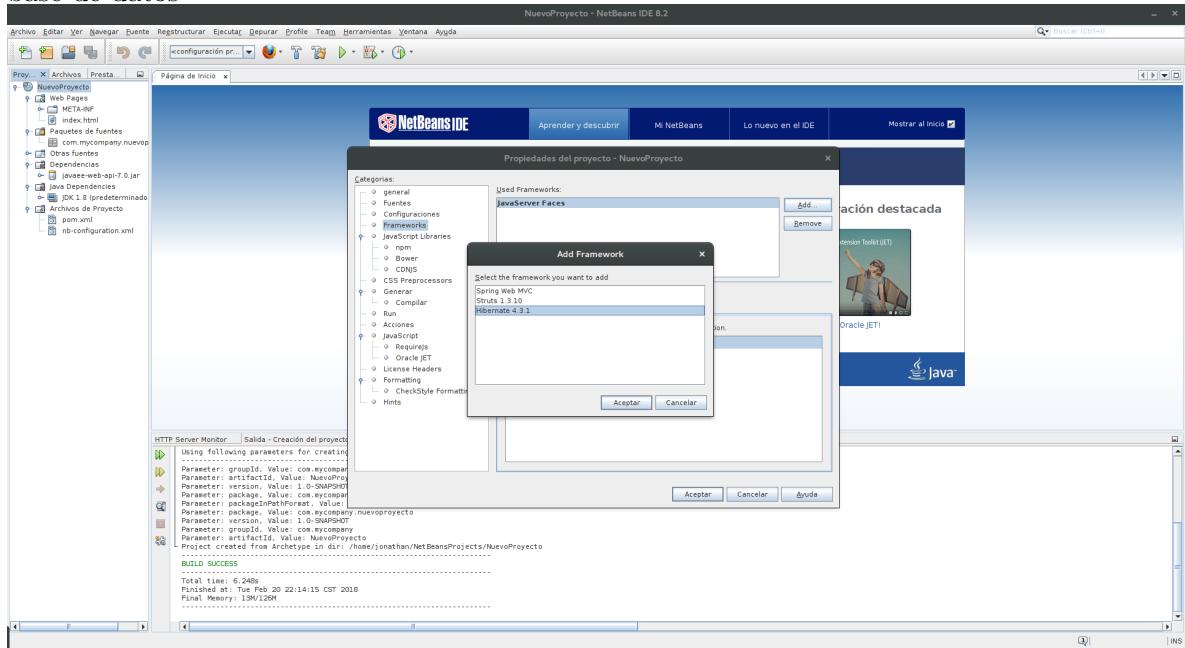
Clic sobre *Components*



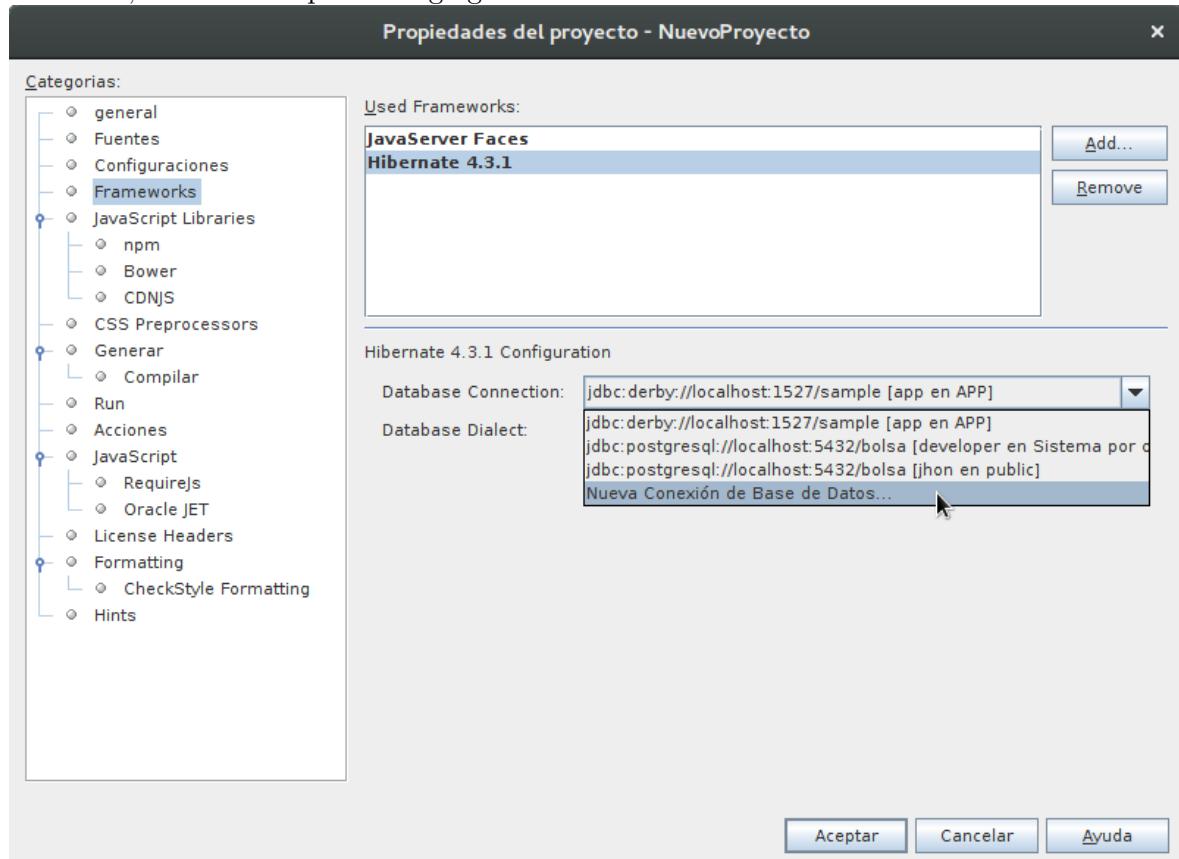
Click sobre primefaces



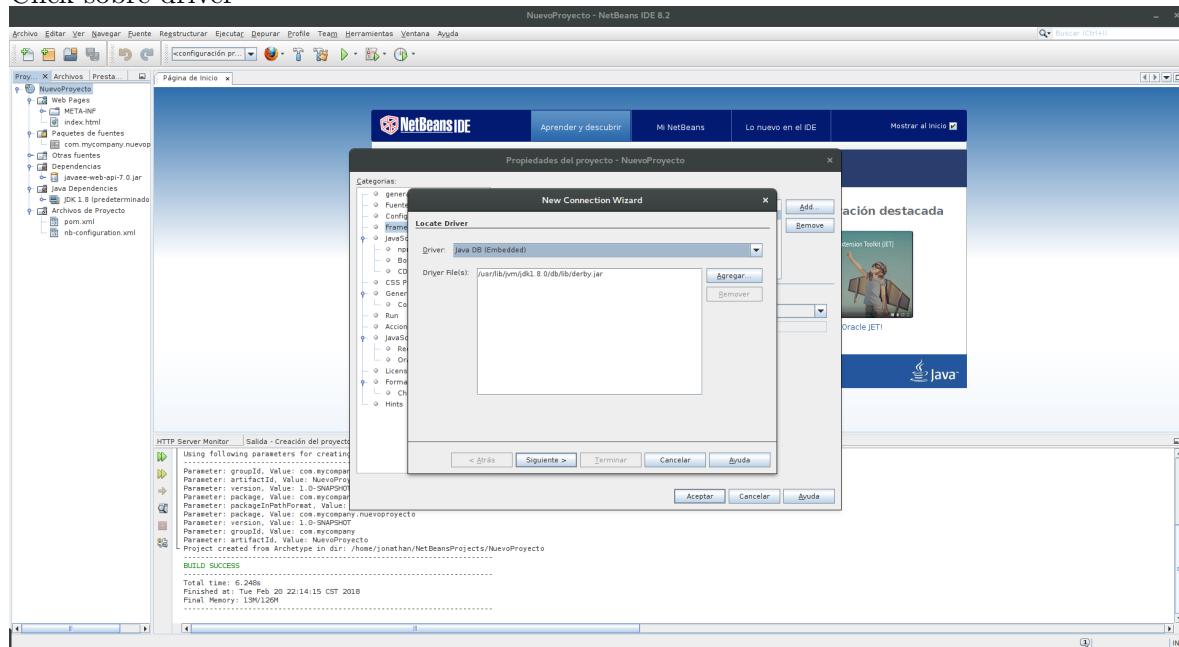
Clic otra vez sobre Add. Ahora agregamos hibernate, se encarga de la conecciones a la base de datos



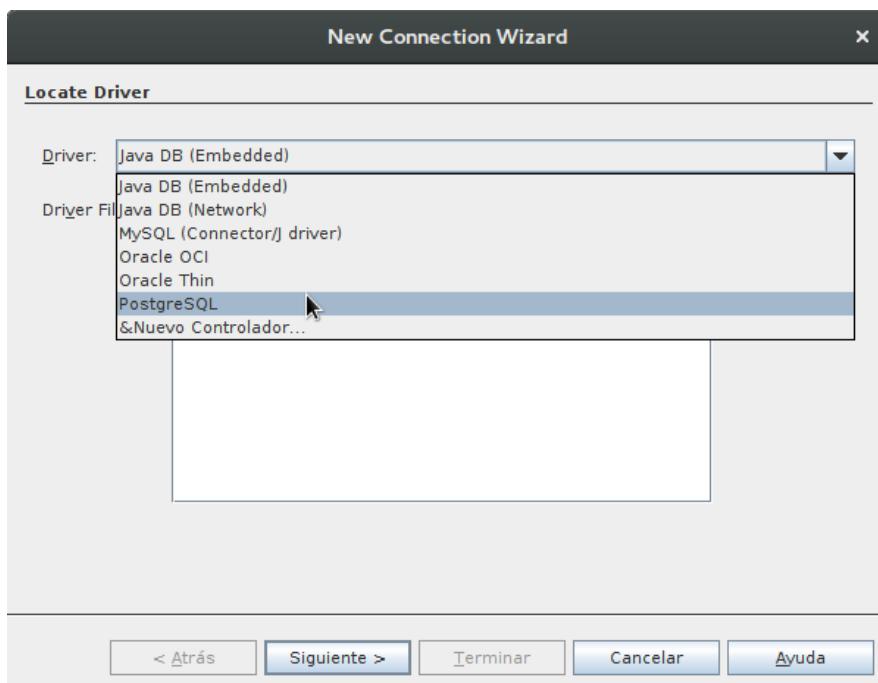
Clic sobre Database connection y clic sobre nueva conexión de base de datos. Si no sale nada, le damos aceptar así agregará los frameworks



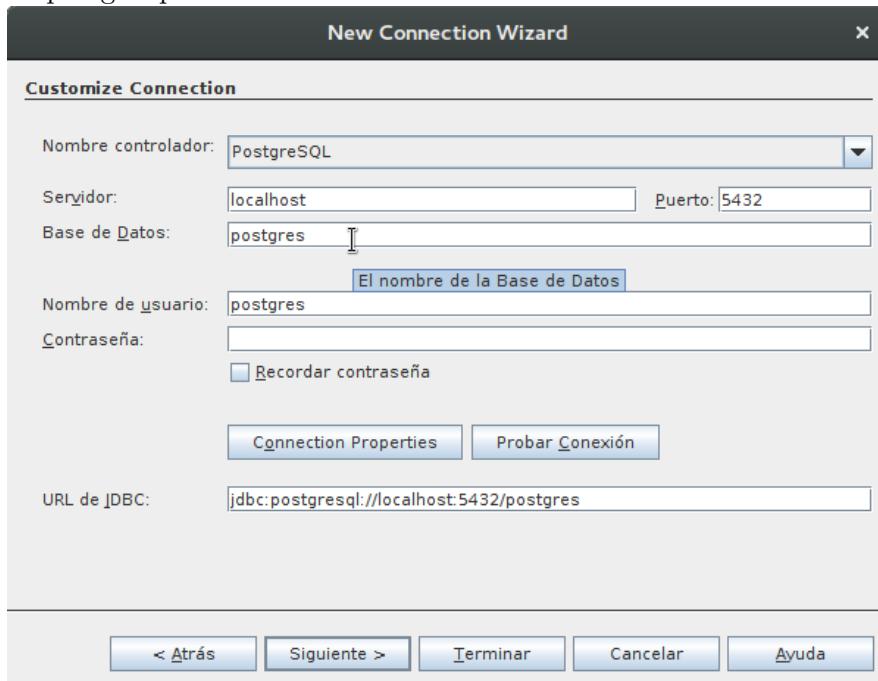
Click sobre driver



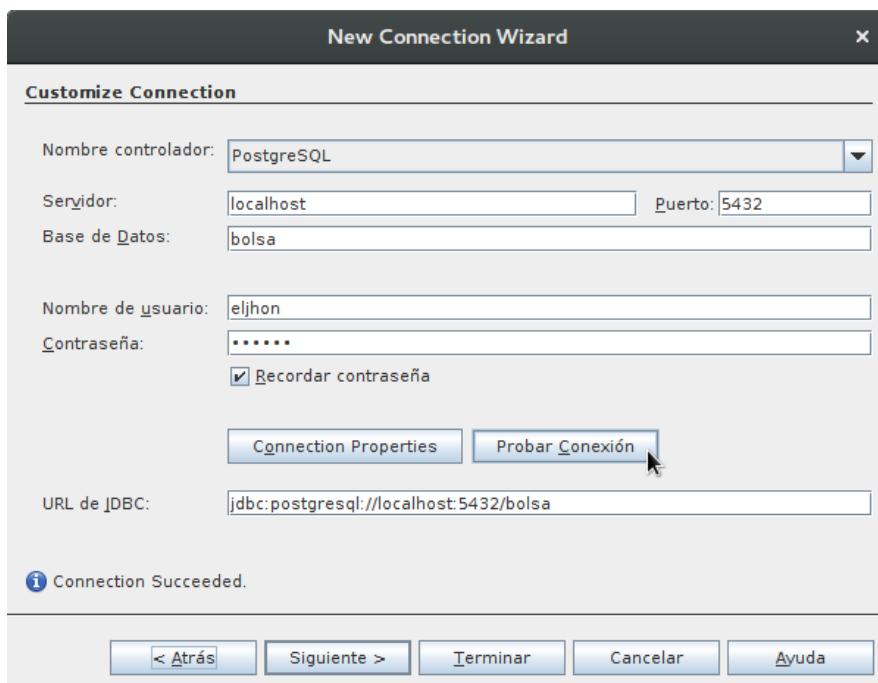
Clic sobre postgresql



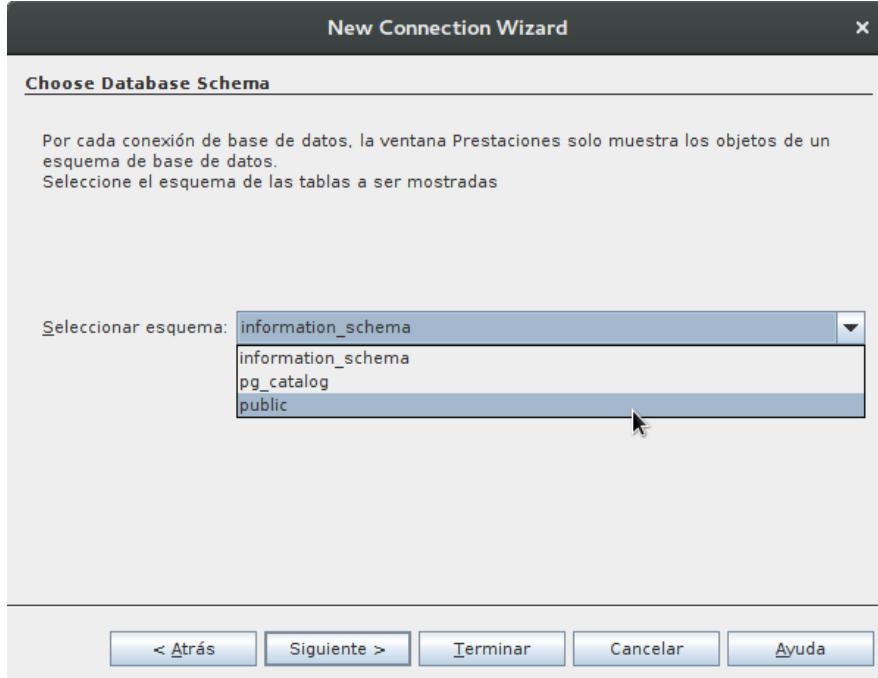
Aquí cambiamos el nombre de la base de datos y el usuario con su contraseña que creamos en postgresql



Así quedaría en este ejemplo. Clic sobre probar conexión para ver que todo esté bien



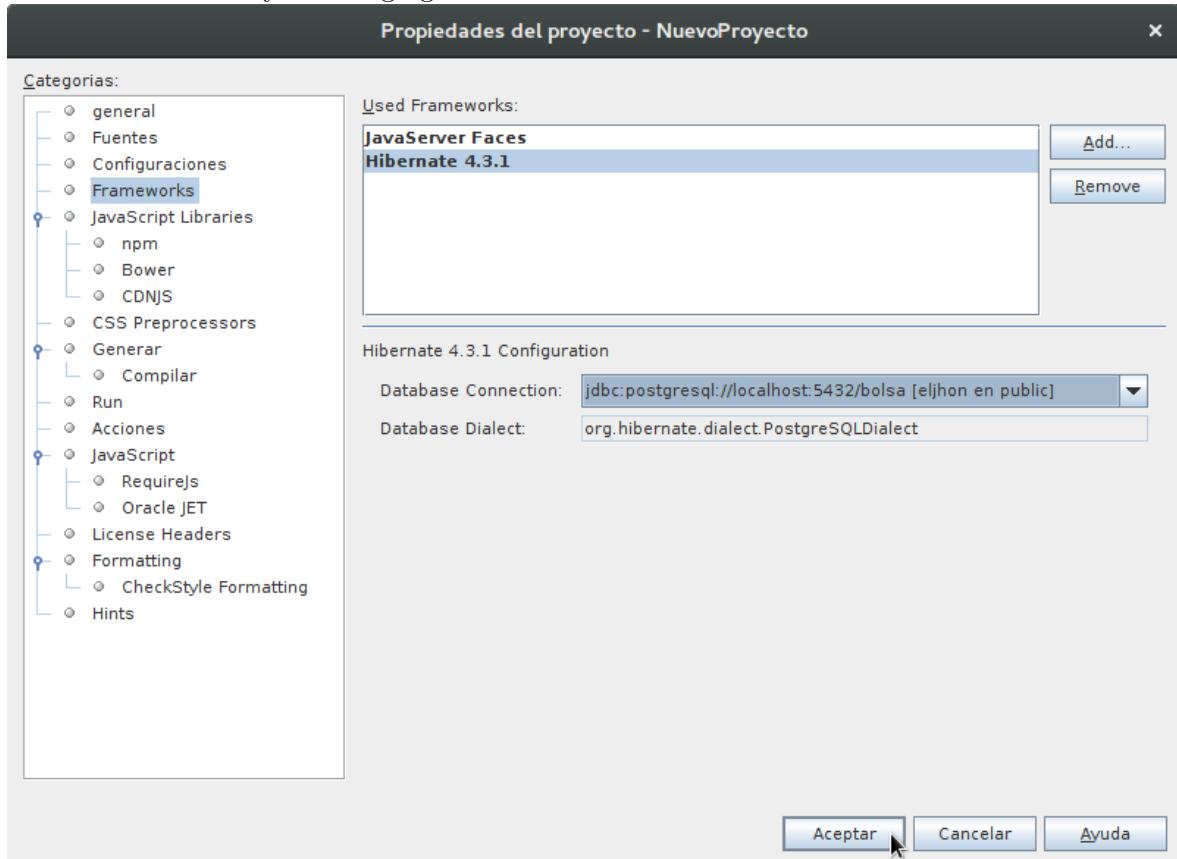
Clic en siguiente y clic sobre seleccionar esquema y escojemos public.



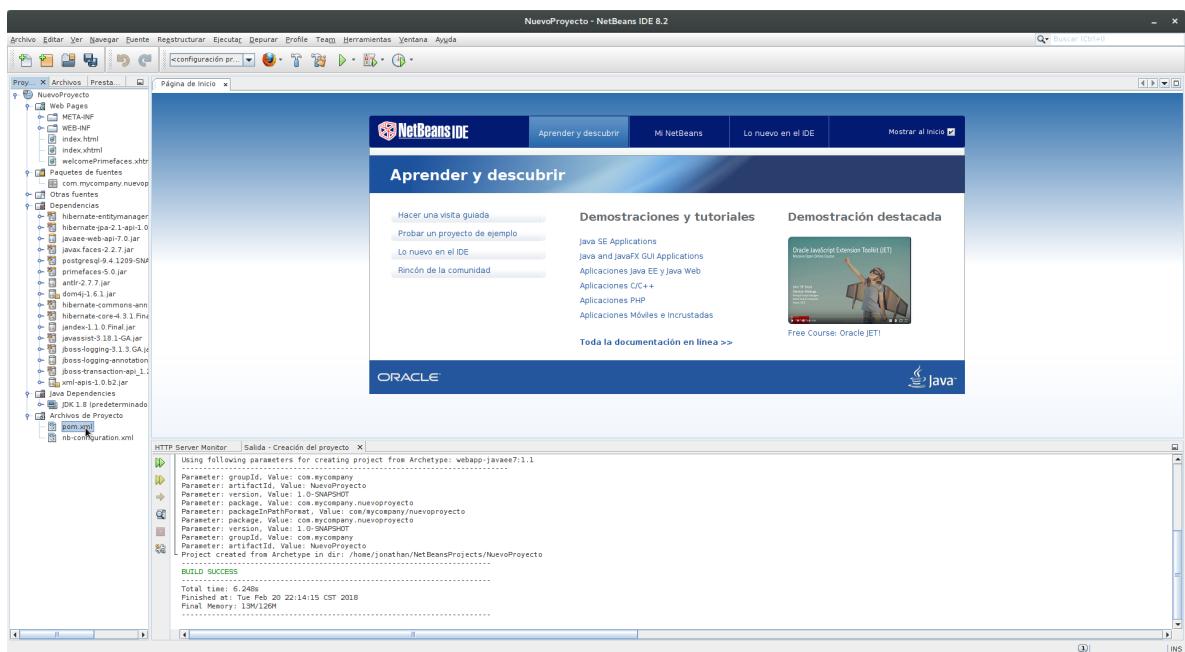
Clic sobre terminar



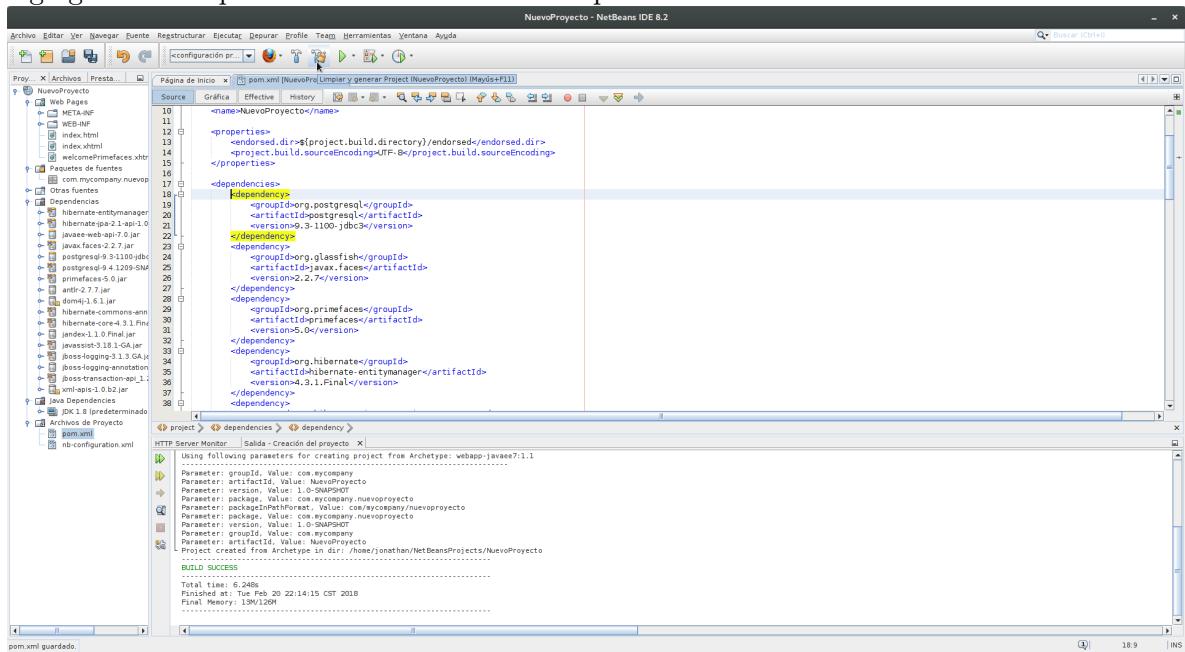
Clic sobre terminar y asi se agregan los frameworks.



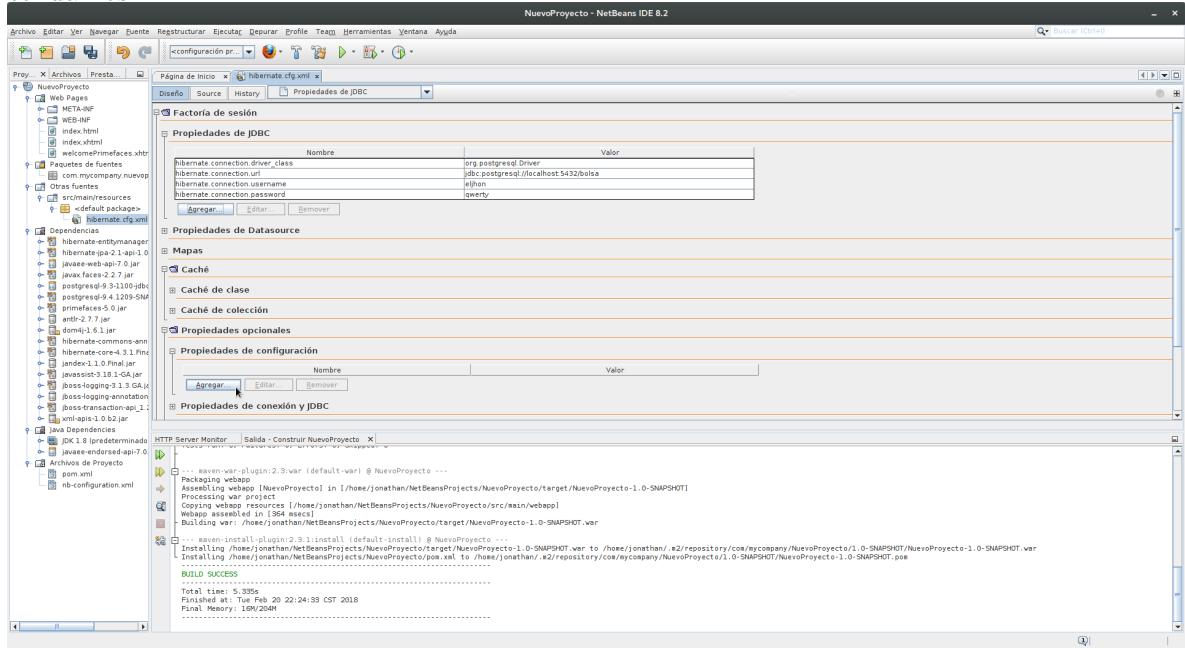
Asi se debe ver nuestro proyecto. click sobre pom.xml para agregar el driver de postgresql



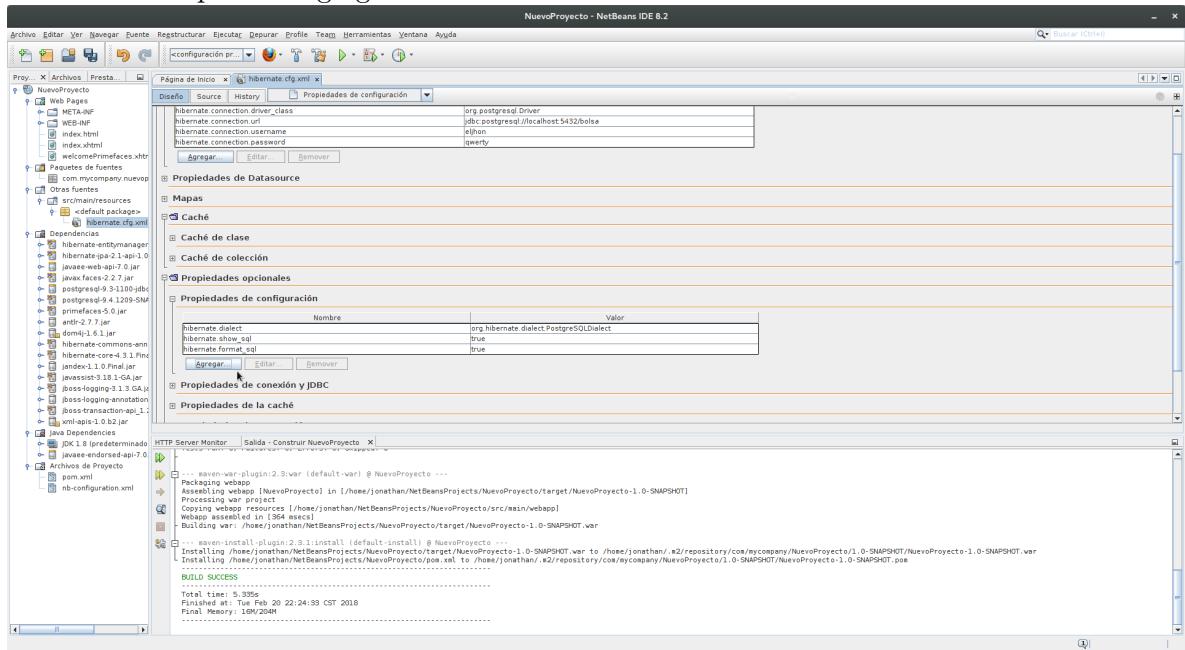
Agregamos la dependencia en la sección de dependencias



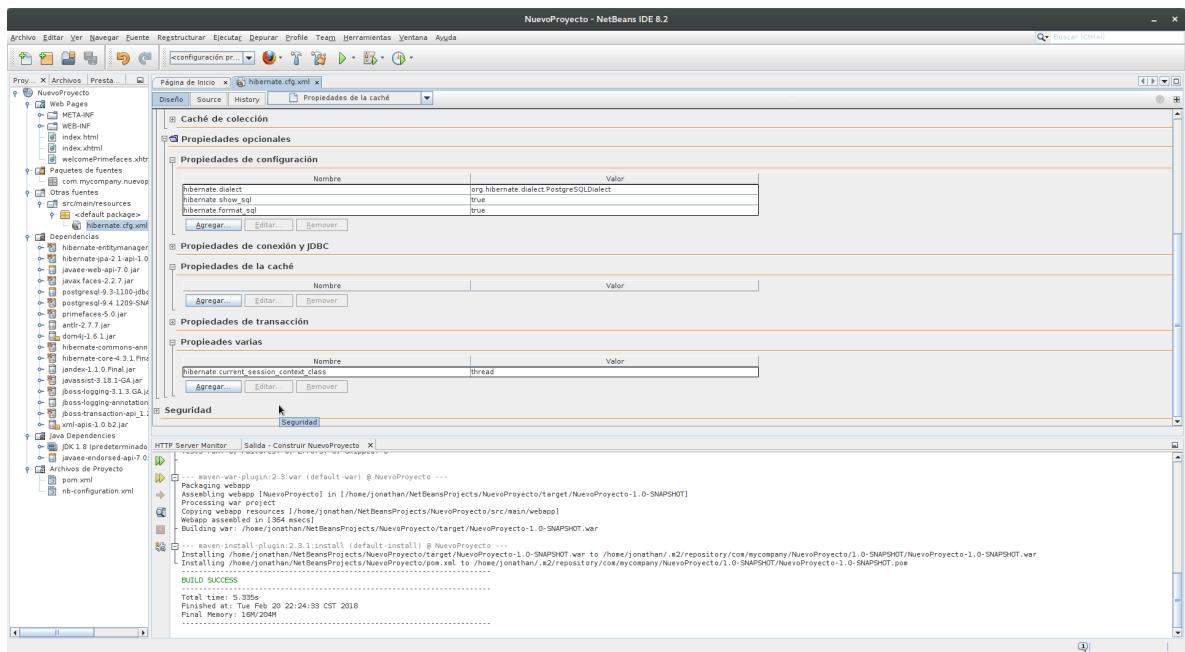
doble clic en hibernate.cfg.xml, agregamos hibernate.dialect, hibernate.show_sql y hibernate.format_sql para que podemos ver las peticiones a postgres en una forma que entendamos



asi se veria despues de agregarlos

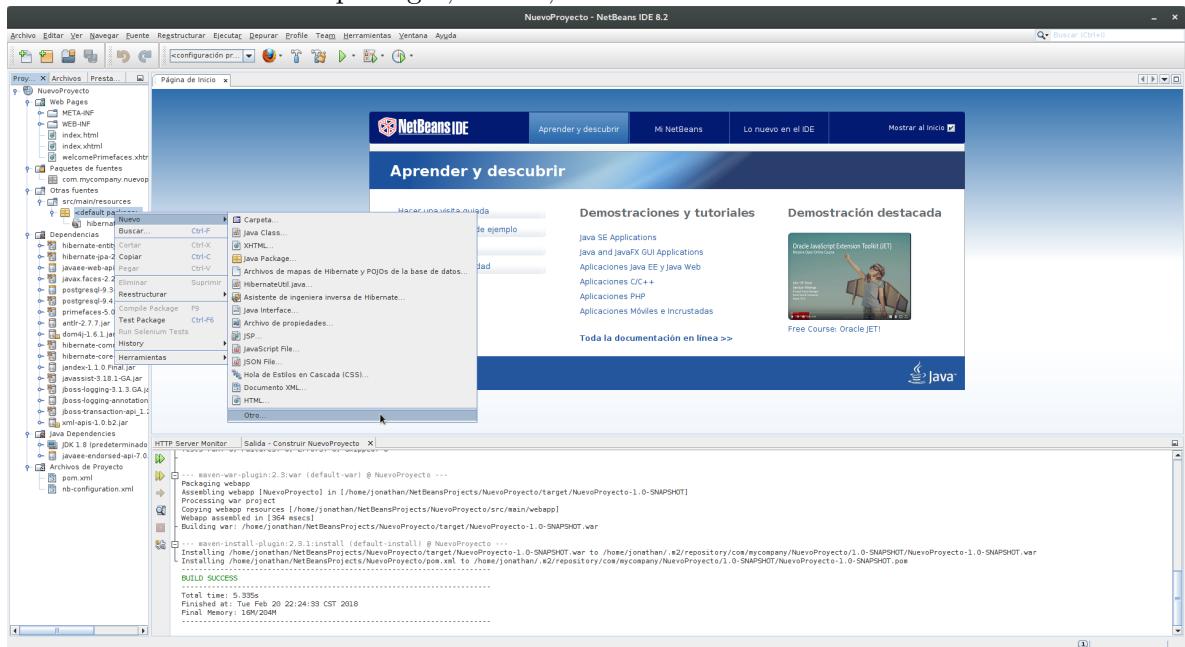


Finalmente agregamos hibernate.current_session_context_class thread para poder crear una inica coneccion a la base de datos y no una cada vez que hacemos peticiones a postgresql.

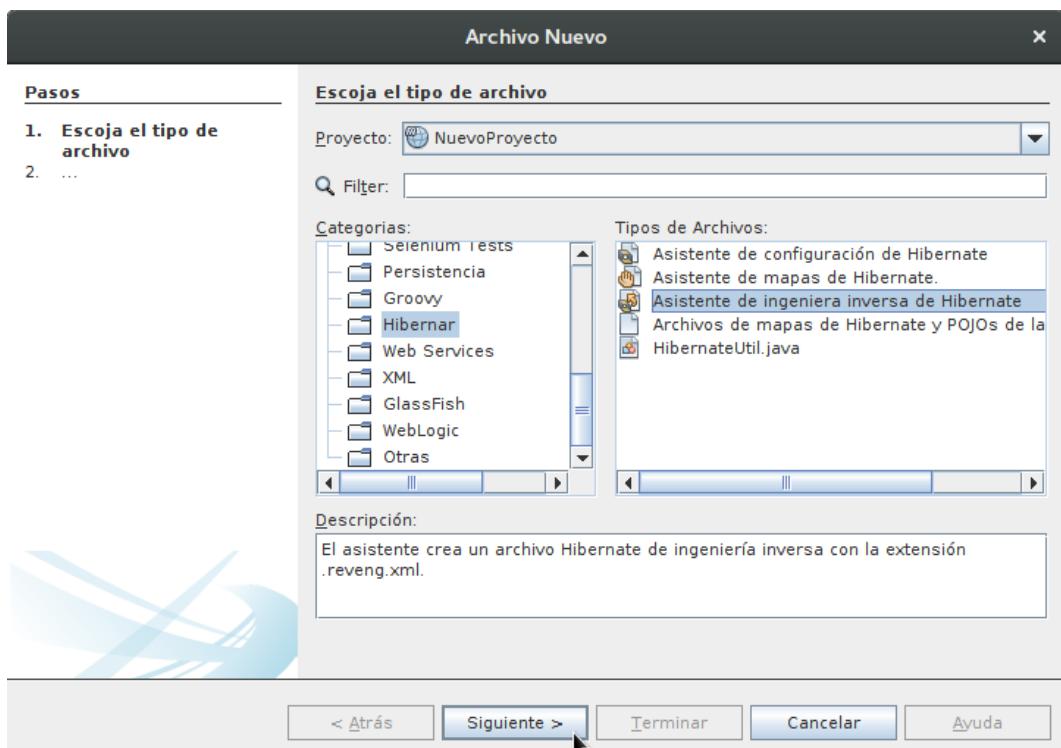


3.3. Hacemos el mapeo de tablas a clases

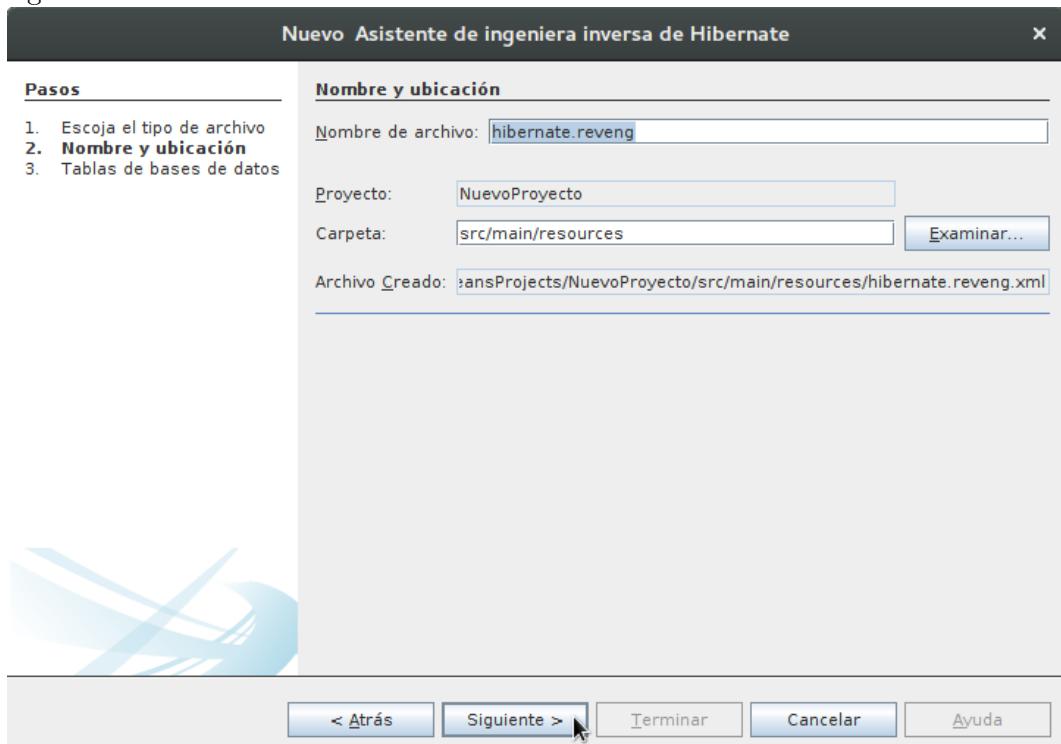
Clic derecho sobre default packages, nuevo , otro.



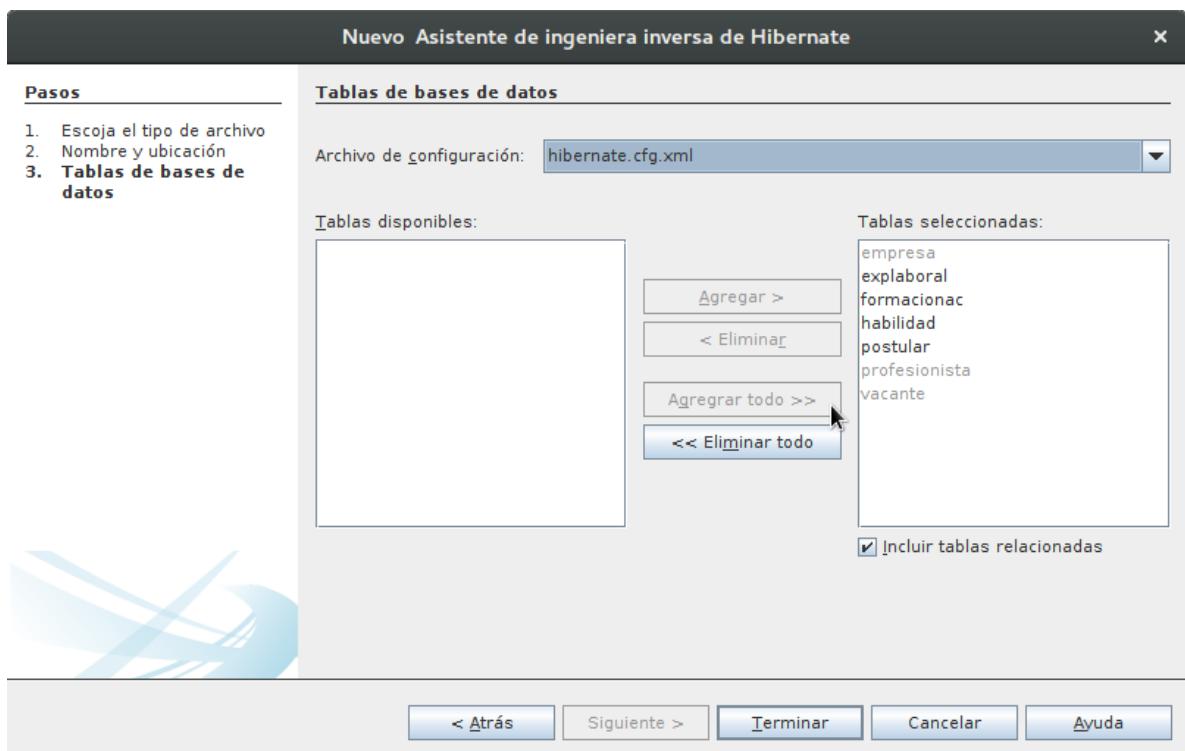
hibernate, Asistente de ingeniería inversa. clic siguiente



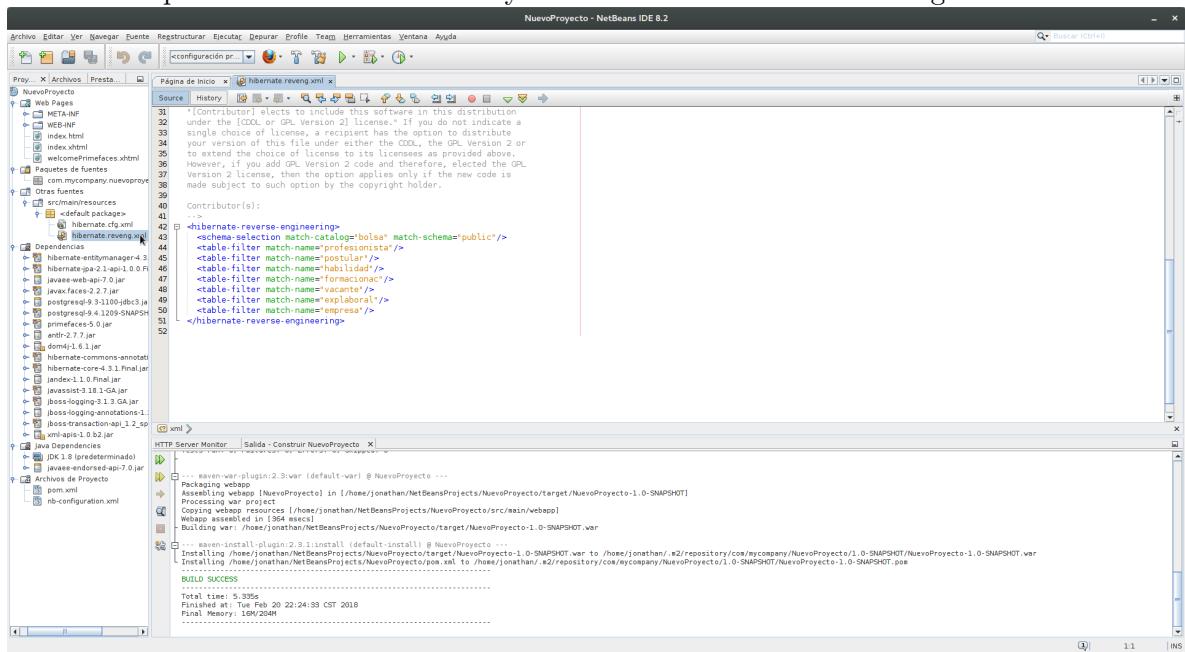
siguiente



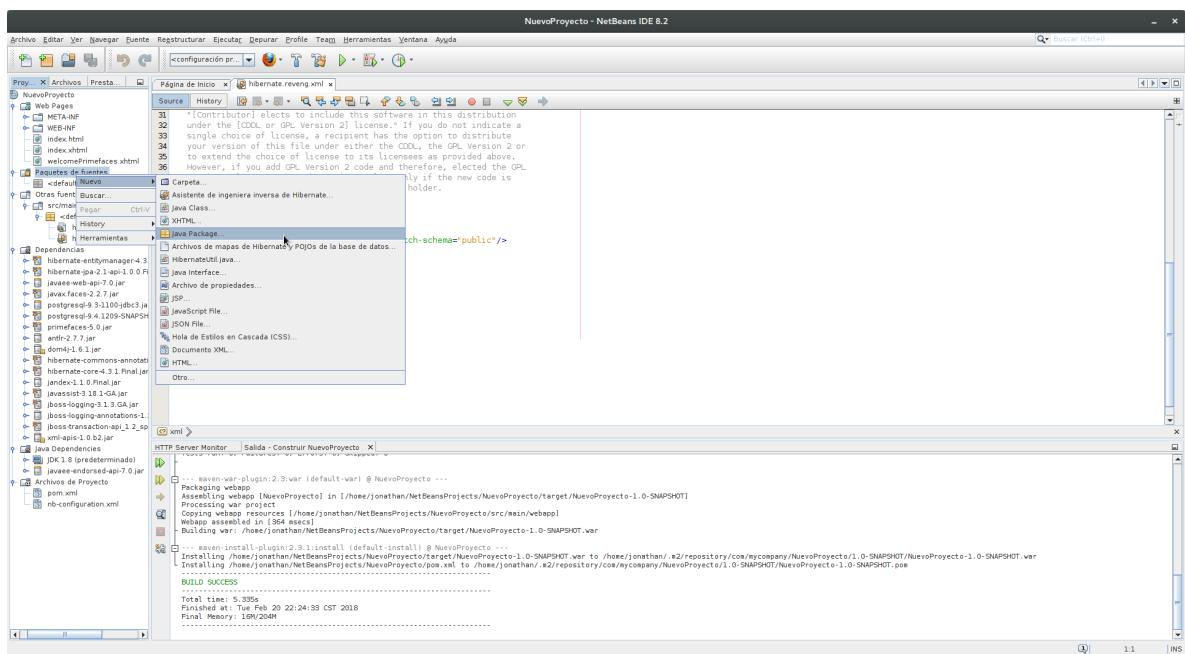
Agregamos las tablas de nuestra base de datos.



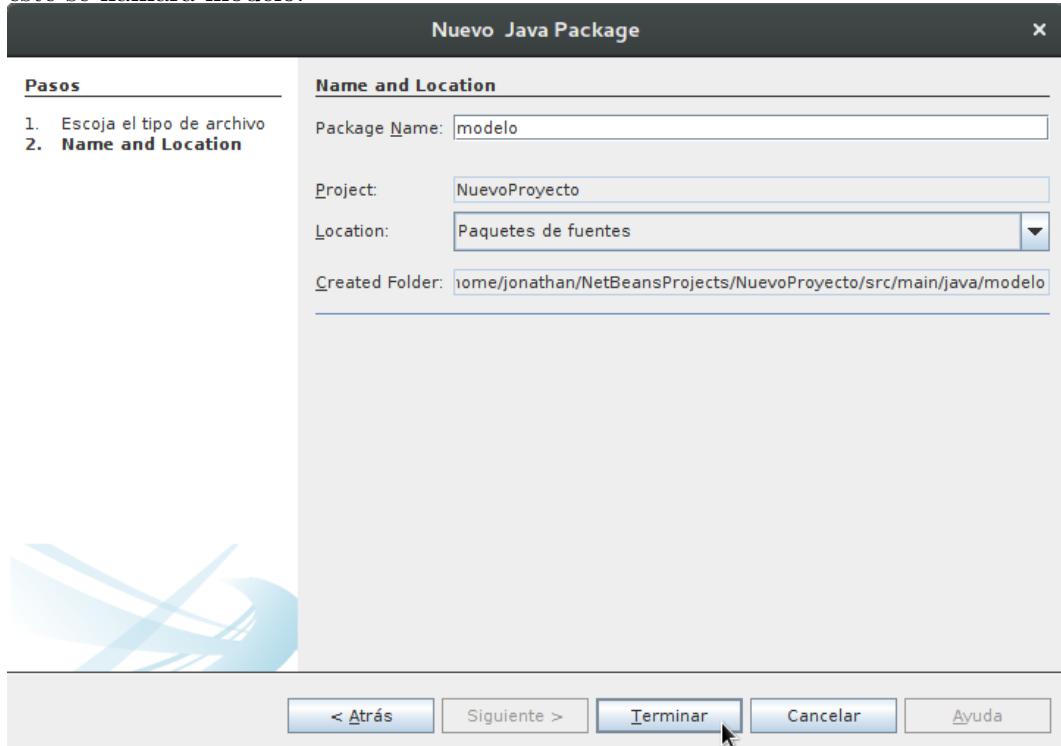
así se ve después de clic sobre termina y crear el archivo hibernate.reveng.xml



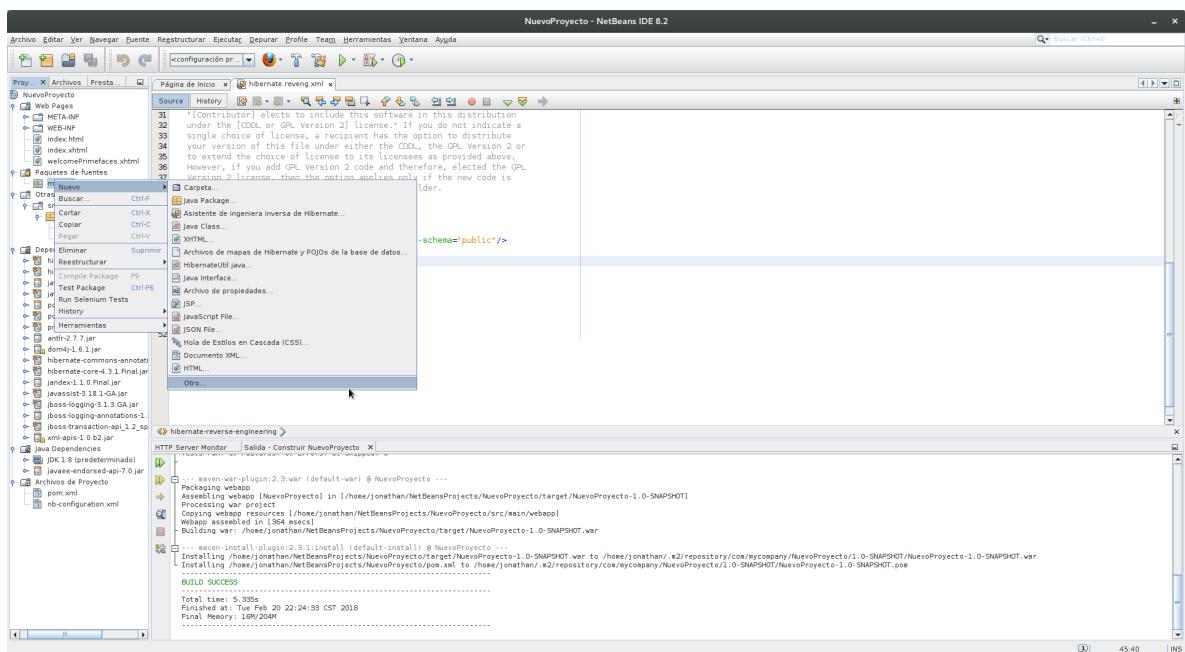
Creamos un nuevo paquete de java



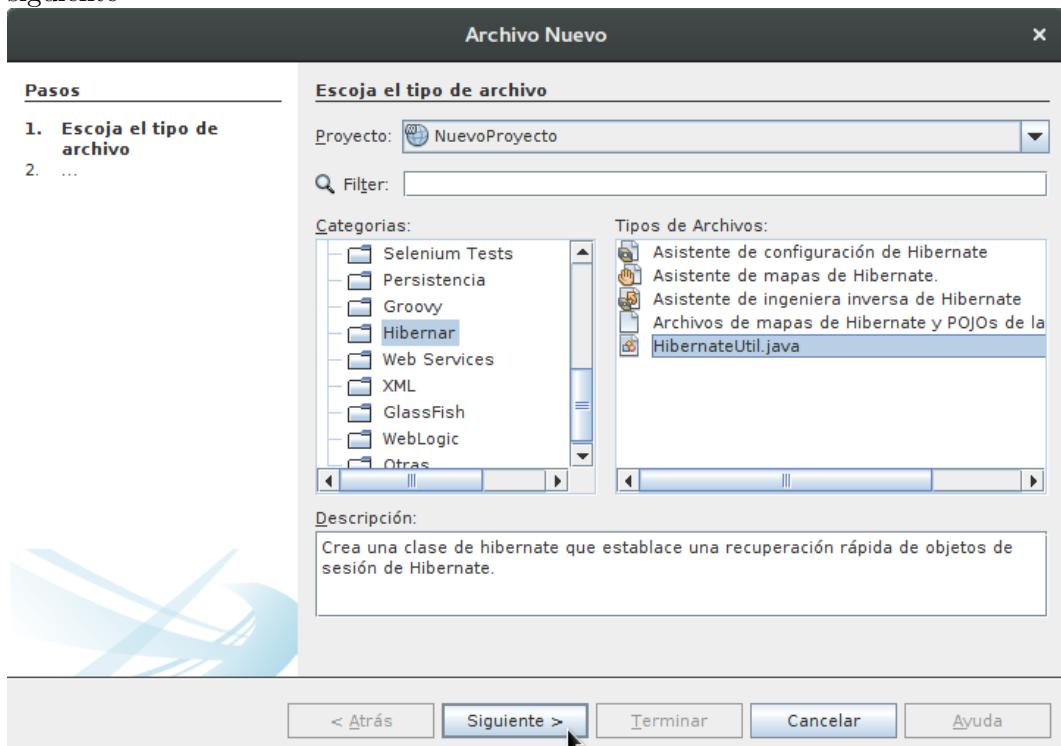
este se llamara modelo.

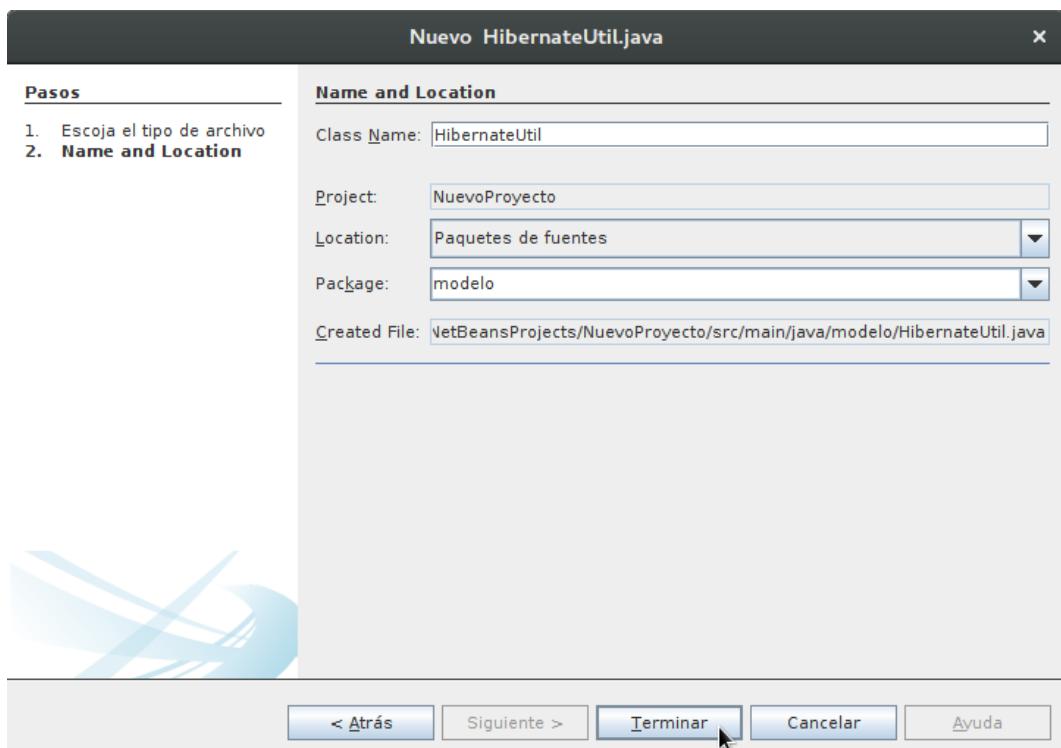


Creamos el archivo de hibernate llamado hibernateUtil.java

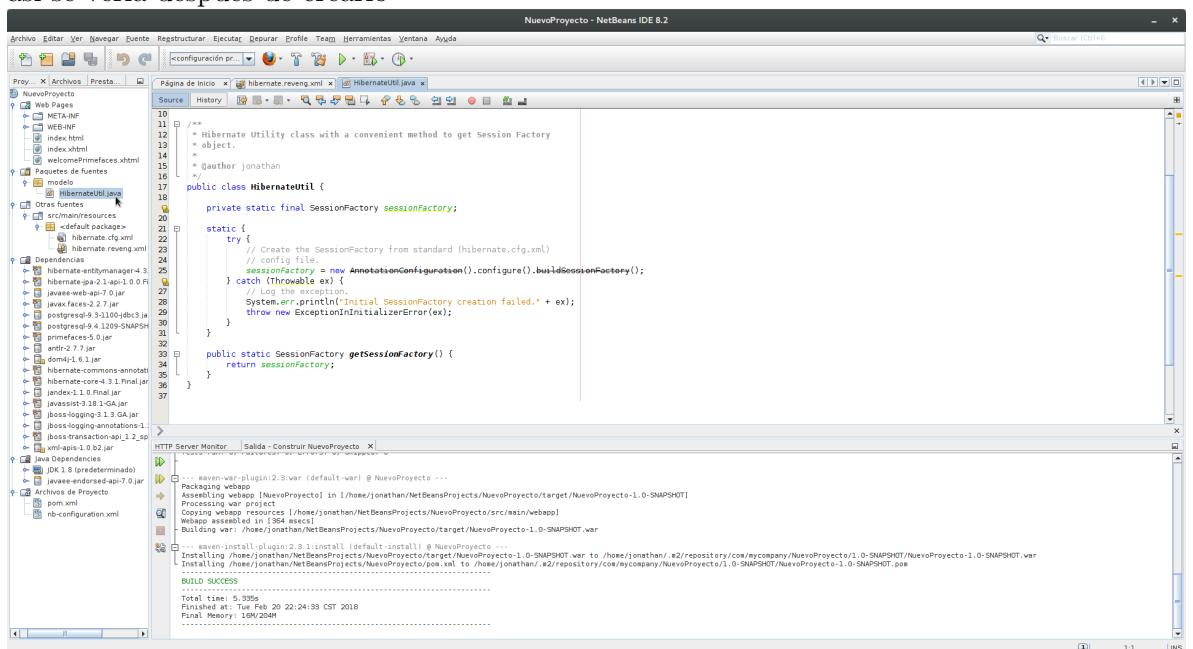


siguiente

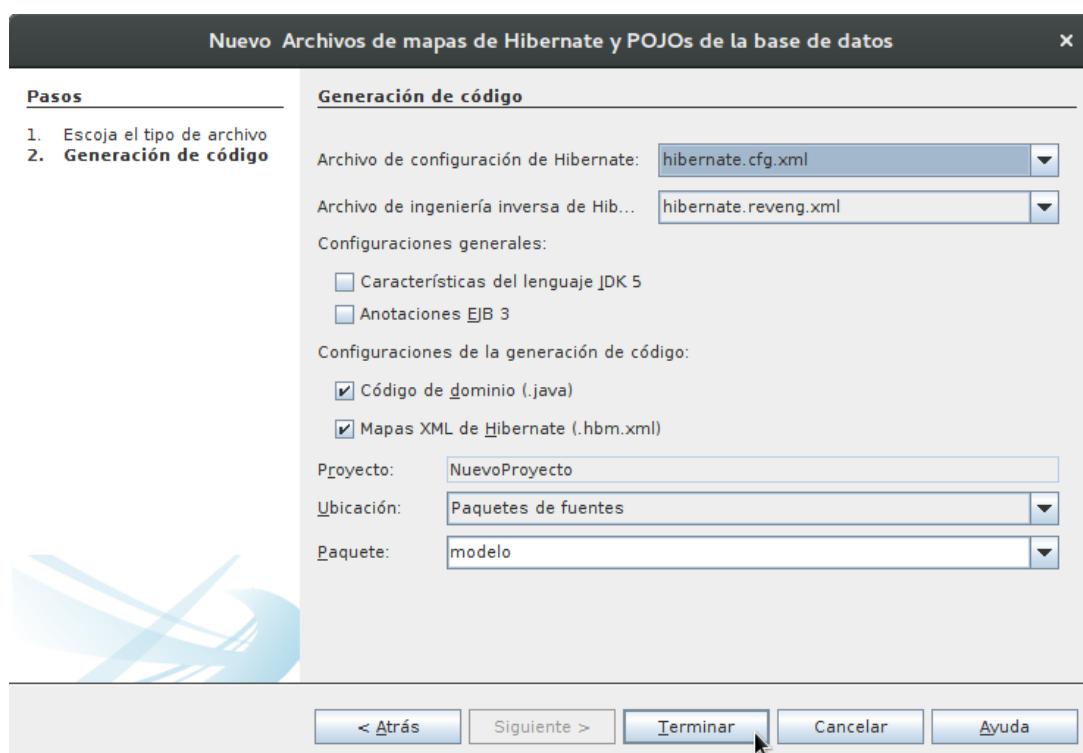
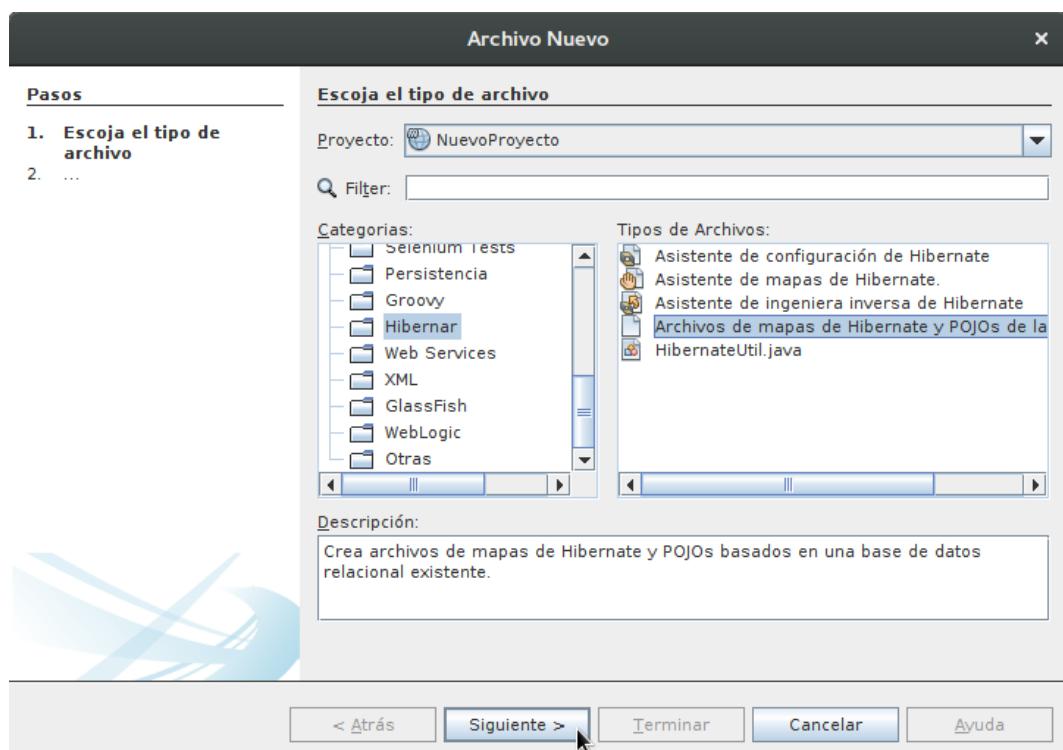




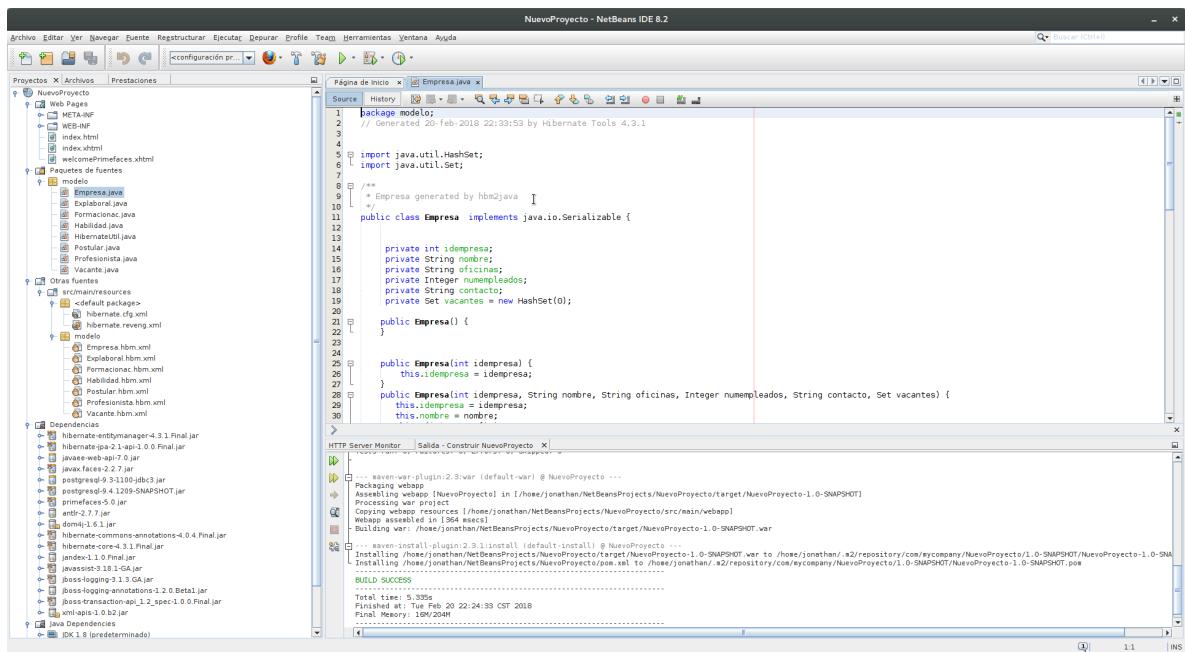
así se vería después de crearlo



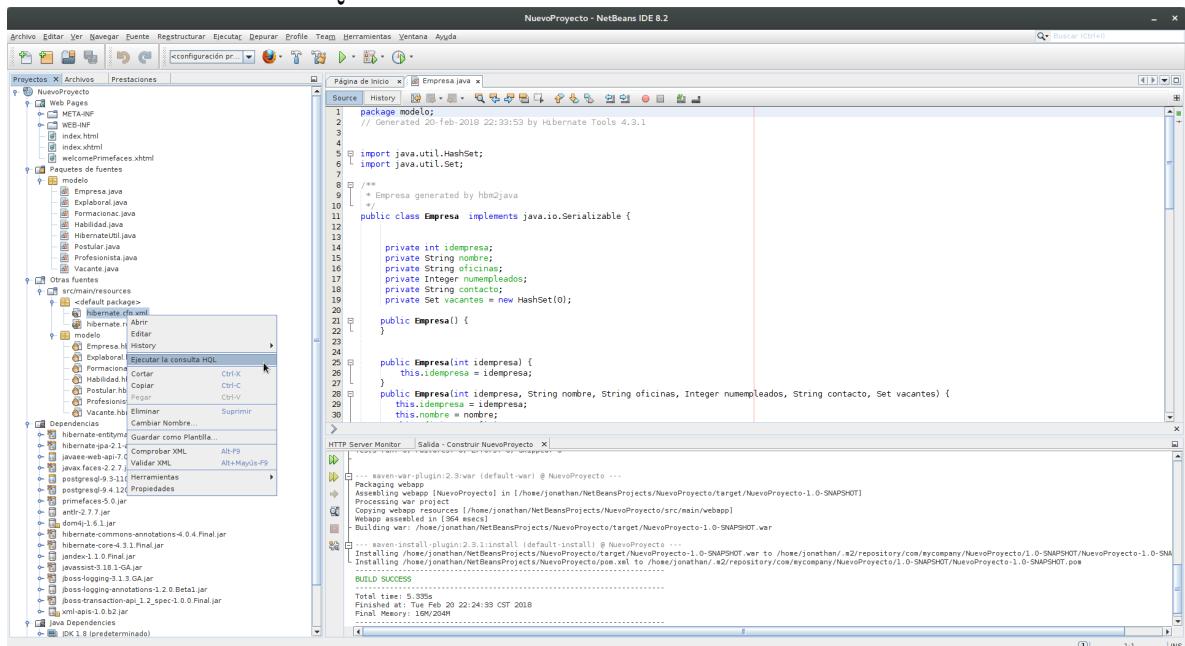
Finalmente hacemos el mapeo de las tablas en postgres a clases de java con ayuda de hibernate.

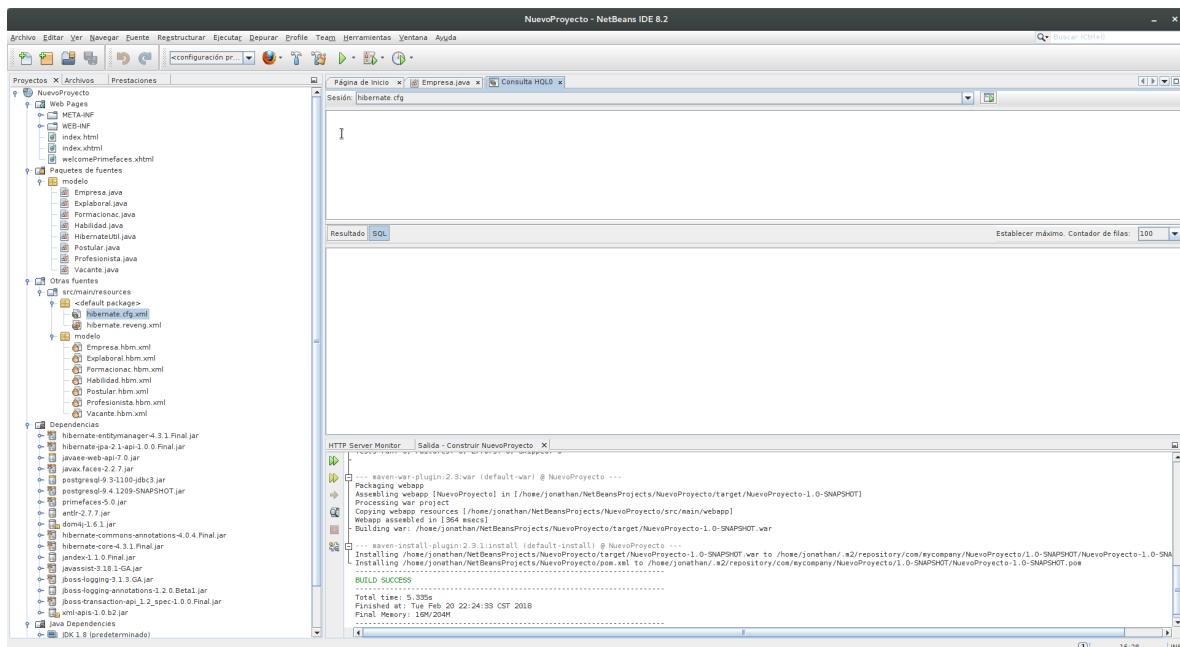


asi se veria despues de agregarlos



Finalmente abrimos una terminal que nos ofrece hibernate para que podamos probar nuestras consultas con HQL.





Un ejemplo seria seleccionar a todas las empresas

