**Manual de instalación proyecto gestor**

**Requisitos**

|  |  |
| --- | --- |
| -WAMP Server:  -PHP -> 5.5.38  -MySQL -> 5.7.36  -Apache -> 2.4.51  -Eclipse | -MongoDB  -Java JDK 8  -MySQL Connector Driver 5.1  -Maven  -Github |

**Instrucciones**

# Instalación MongoDB

Descargar MongoDB desde la siguiente URL: <https://www.mongodb.com/try/download/community>

De igual forma necesitaras la consola de comandos de MongoDB Shell la cual puedes descargar desde la siguiente URL: <https://www.mongodb.com/try/download/shell>

Una vez instalado MongoDB y descargado el Shell de Mongo deberás copiar el contenido de la carpeta /bin del zip de mongoshell a la ubicación donde tengas instalado MongoDB; por ejemplo:

Interfaz de usuario gráfica

Descripción generada automáticamenteIntalación MongoDB: C:\Program Files\MongoDB\Server\7.0\bin

# Instalación Java JDK 8

Descargar e instalar el instalador de la siguiente URL: <https://www.oracle.com/java/technologies/javase/javase8-archive-downloads.html>

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

# Descargar MySQL Connector Driver 5.1

Descargar e instalar el instalador de la siguiente URL: <https://downloads.mysql.com/archives/c-j/>

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamenteY descargar el archivo zip, para posteriormente descomprimirlo y guardar la ubicación de la carpeta por que se utilizará más adelante.

# Instalación Eclipse

Descargar e instalar el instalador de la siguiente URL: <https://www.eclipse.org/downloads/>

Interfaz de usuario gráfica, Texto, Aplicación, Teams

Descripción generada automáticamenteUna vez descargado abrir el instalador y seleccionar Eclipse IDE for Enterprise Java and Web Developers

# Configuración de las bases de datos y descarga del repositorio

Lo primero es bajar el repositorio <https://github.com/vhackero/gestor.git> con el comando:

git clone https://github.com/vhackero/gestor.git

Nota: El repositorio es privado y se requiere permiso para clonarlo.

Después importamos en MySQL ambas bases de datos, una se encuentra en gestor\codigoFuente\bd y se llama des\_sisi\_gestor\_final.sql esta pertenece al gestor y la bd de Moodle la pueden descargar de la siguiente url: <https://aula-modelo41.unadmexico.mx/sisi_plataforma.sql>

Texto

Descripción generada automáticamente

Ahora hay que crear un usuario y darle permisos en ambas bases:

CREATE USER elearning@'localhost' IDENTIFIED BY 'elearning'

GRANT ALL ON des\_sisi\_gestor.\* TO elearning@'localhost';

GRANT ALL ON sisi\_plataforma.\* TO elearning@'localhost';

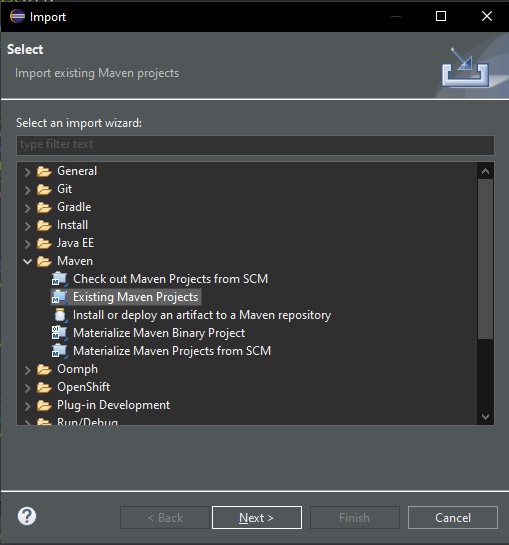
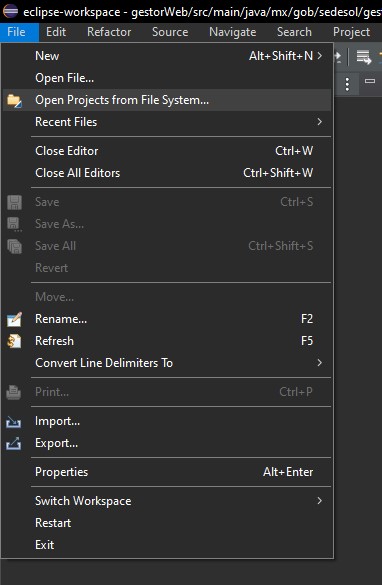
En MongoDB ejecutamos los siguientes comandos desde el mongosh:

use eLearning

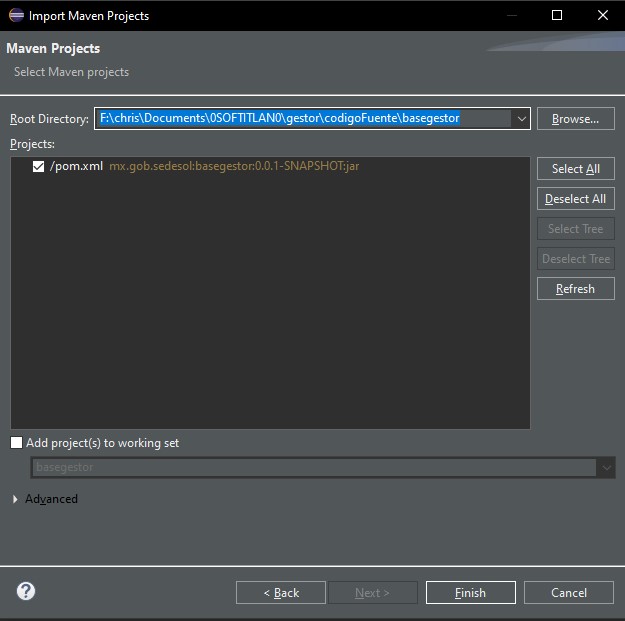
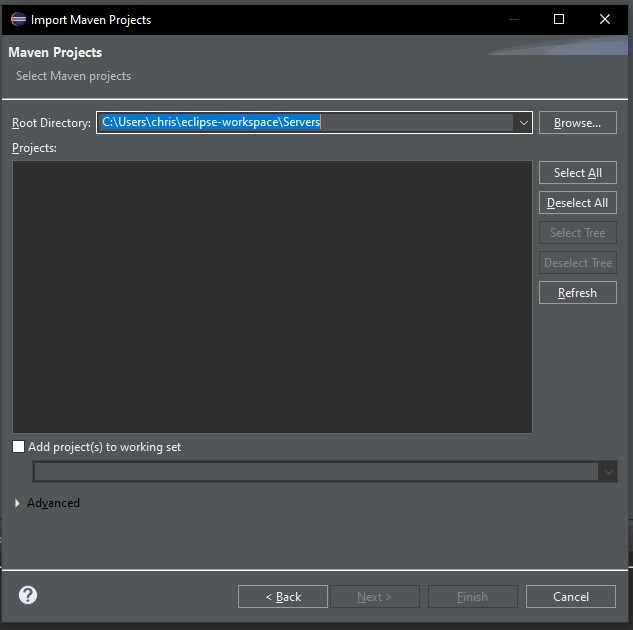
db.createUser( { user: "eLearning", pwd: "sisi.123@", roles: [{role:"dbOwner", db:"eLearning"} ] })

# Importar proyectos y compilarlos

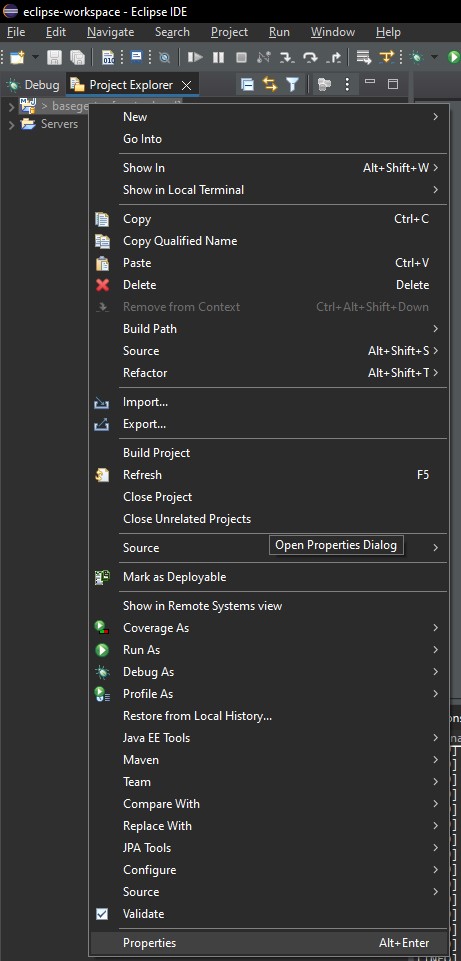
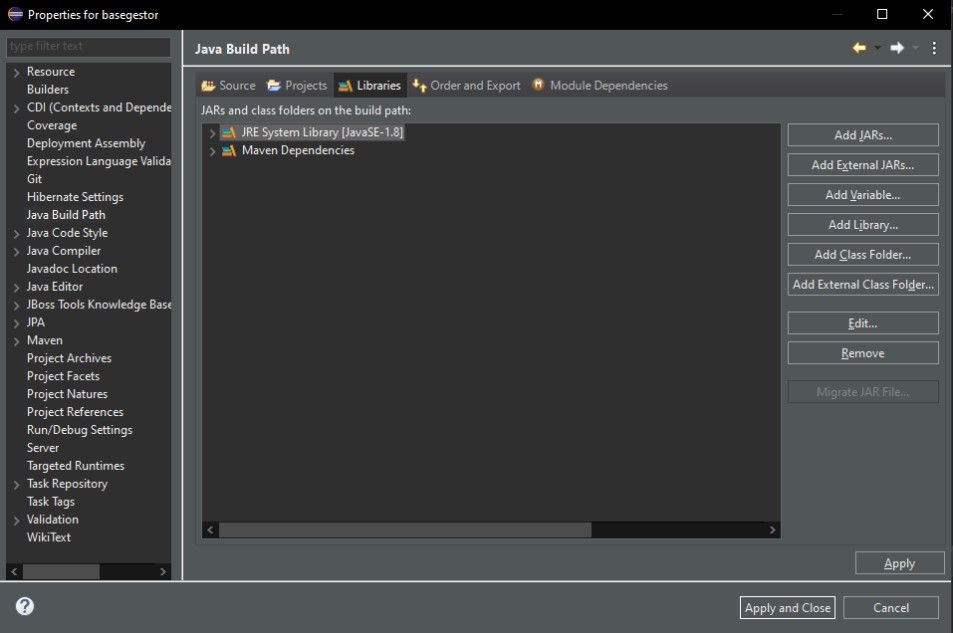
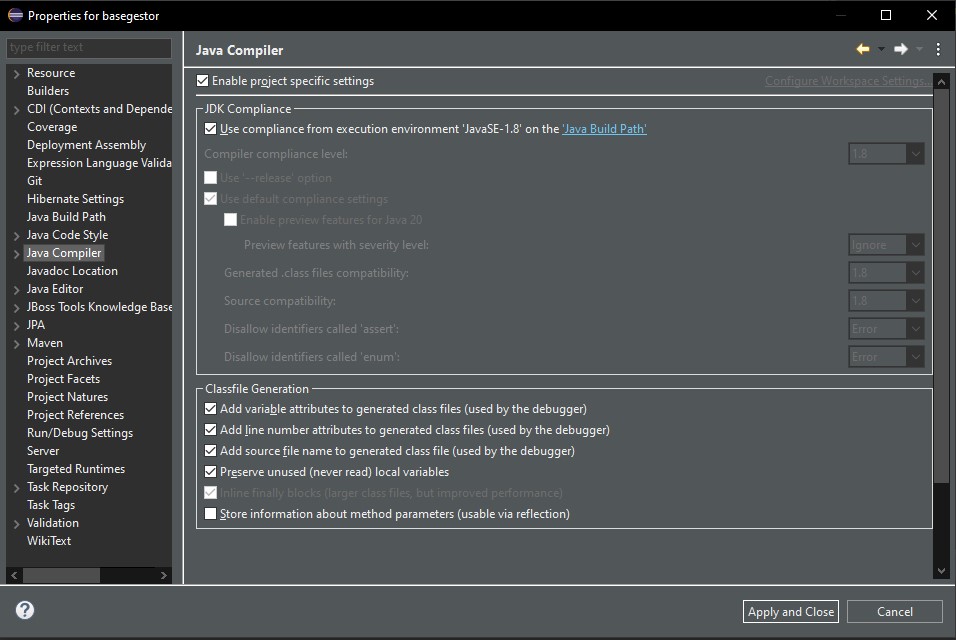
Primero para importarlos damos en **File** -> **Import**, en la siguiente ventana buscamos **Maven**, seleccionamos **Existing Maven Projects** y damos en **Next**



Ahora damos clic en **Browse** y buscamos el proyecto **basegestor** y **gestorweb** que están en la carpeta gestor/codigoFuente, los importamos individualmente. Verificamos que se encuentre seleccionado el archivo pom.xml y damos en **Finish**



En ambos proyectos hay que validar que la versión de Java sea la correcta, para ello hay que dar **clic derecho** en el proyecto y entrar a **Properties**. Después revisamos que este como JavaSE-1.8 tanto en **Java Compiler** como en **Java Build Path**.



Captura de pantalla de un celular

Descripción generada automáticamentePara asegurarse que se está seleccionado JavaSE-1.8 da clic en **Java Build Path**, selecciona el **JRE System Librery [JavaSE-1.8]** y da clic en **Edit**

Captura de pantalla de un celular

Descripción generada automáticamenteEn la siguiente pantalla da clic en **Installed JERs,** revisa que el jre sea la versión seleccionada sea la 1.8 , de lo contrario da clic en **Add**->**Estándar VM ->Next** y en JRE home dar clic en **Directory**

Interfaz de usuario gráfica, Texto

Descripción generada automáticamenteSeleccionar la ruta JRE 1.8 de Java, por ejemplo “C:\Program Files\Java\jdk1.8.0\_202\jre” por último da clic en **Finish**

Captura de pantalla de un celular

Descripción generada automáticamenteAsegúrate de seleccionar la ruta del paso anterior y da clic en **Apply and Close**

Igualmente, hay que revisar que se encuentre seleccionado el MYSQL Driver, para ello **click derecho** sobre el proyecto **Properties->JPA->Conection->Add conection**, en la siguiente pantalla dar clic en **MYSQL->Next,**  en la siguiente pantalla dar clic en el icono **New driver definition** y seleccionar MYSQL 5.1 y en la pestaña **JAR List**  clic en **Clear All** y después en **Add JAR/Zip** y seleccionar el driver de mysql que se descargó anteriormente y clic en **OK** agregar la base de datos “des\_sisi\_gestor” y el usuario y la contraseña “elearning” y darle clic en **Test Connecction** si todo sale correcto dar clic en **Finish**

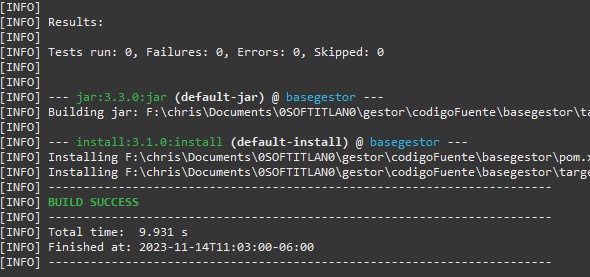
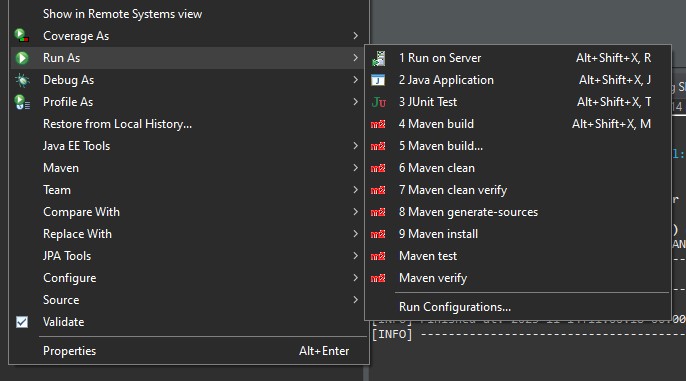
Interfaz de usuario gráfica, Texto

Descripción generada automáticamente Interfaz de usuario gráfica, Aplicación

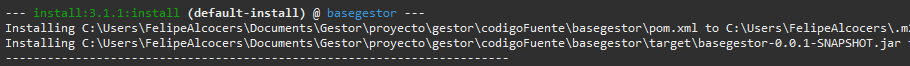
Descripción generada automáticamente Captura de pantalla de un celular

Descripción generada automáticamente

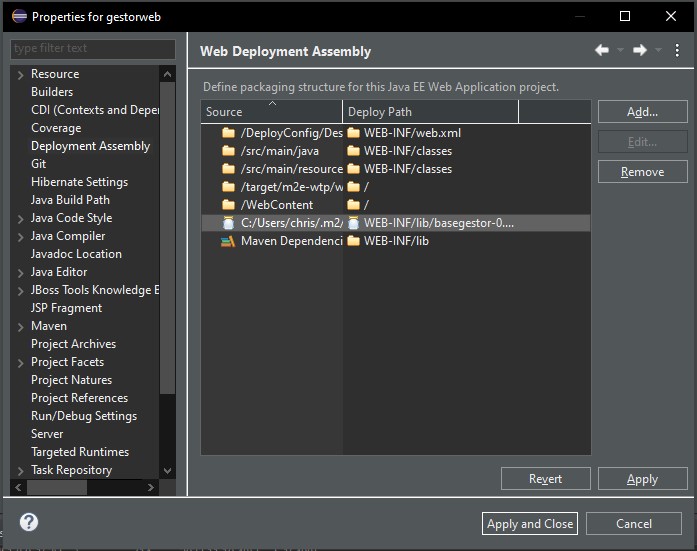
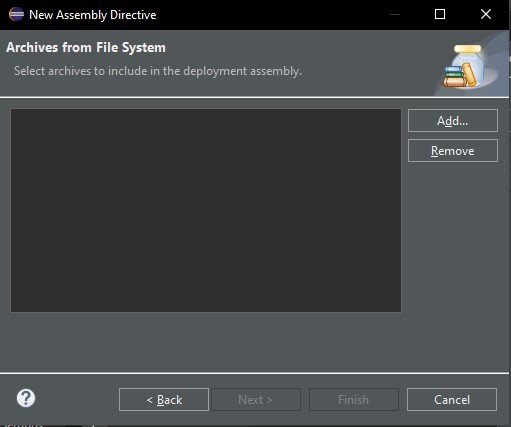
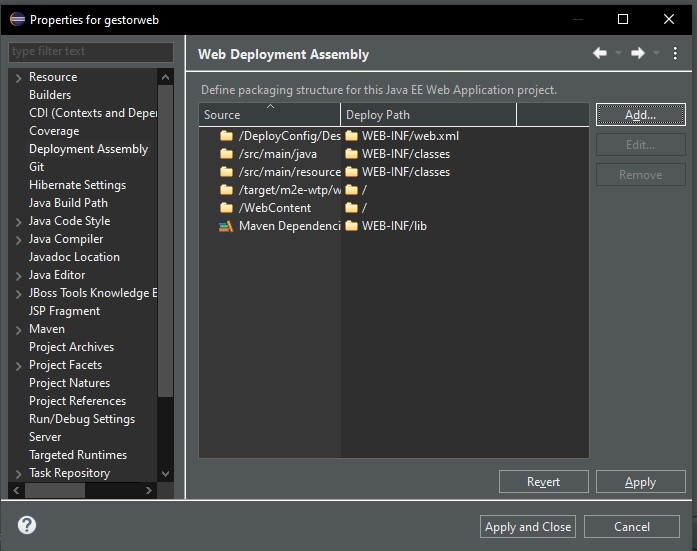
Ahora para compilarlo hay que dar clic derecho en el proyecto basegestor, luego en **Run as** y en **Maven install** y comprobamos en la consola que haya salido bien el proceso



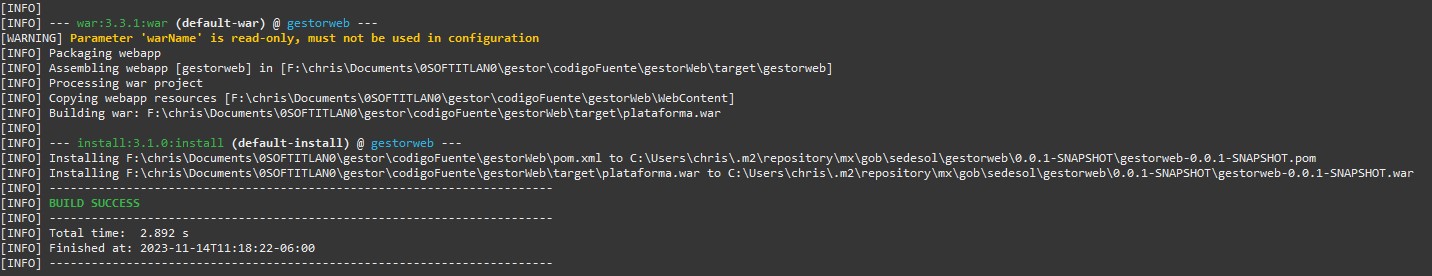
Es importante guardar la ruta del archivo basegestor-0.0.1-SNATSHOT.jar. Para identificar la ruta del archivo en la consola de IDE muestra la ubicación del archivo



Ahora vamos con el proyecto gestorweb. Primero hay que añadir el archivo de basegestor desde **Properties** y **Deployment Assembly**. Damos clic en **Add** y en la siguiente ventana **en Archives from File System**, luego en **Add** y buscamos la ruta del paso anterior para importar el archivo basegestor-0.0.1-SNATSHOT.jar. Después damos clic en **Finish** y en **Apply and Close**



Ahora repetimos los pasos de **Maven clean** y **Maven install**, esto nos generará el archivo plataforma.war



Texto

Descripción generada automáticamenteNOTA: En caso de que **Maven install** falle se requiera validar en la pestaña de problemas cual es el motivo por el cual no se instaló el proyecto, por ejemplo en este caso nos marca que tenemos 4 errores, el cual nos marca que no existe la dependencia **olap4j,** para validar que la dependencia existe no dirigimos a la ruta en la que intenta encontrar la dependencia.

Captura de pantalla de un celular

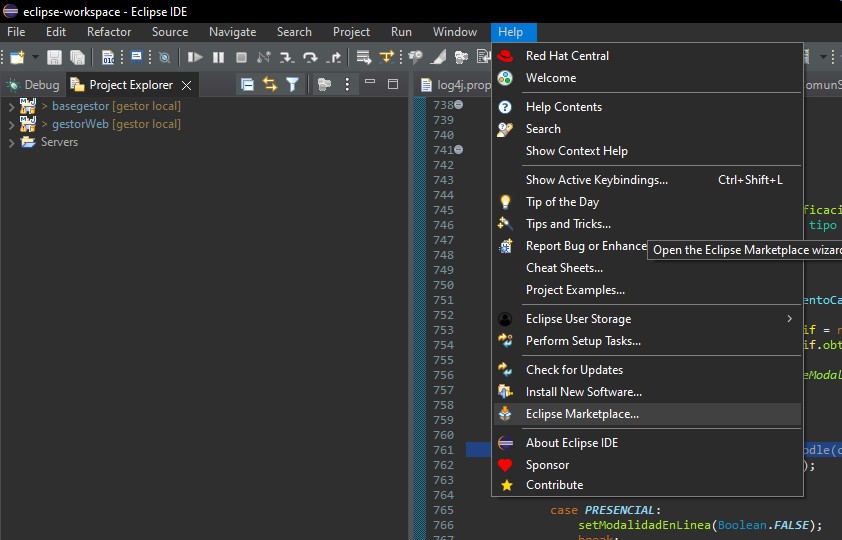
Descripción generada automáticamenteTexto

Descripción generada automáticamenteCaptura de pantalla de computadora

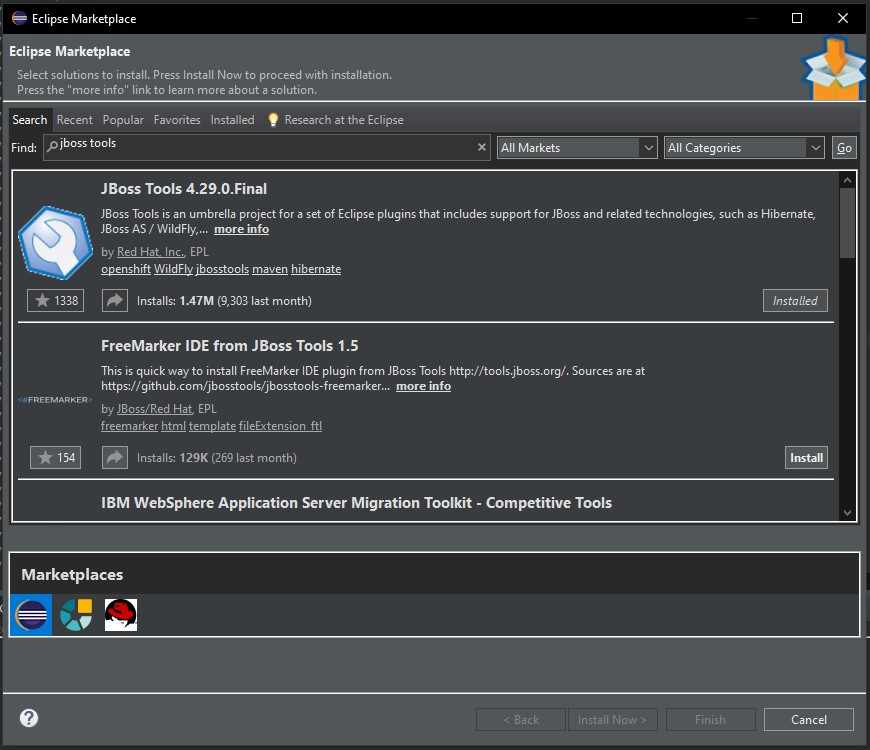
Descripción generada automáticamenteBuscamos el archivo en la web y copiamos en la ruta y actualizamos el proyecto dando **clic derecho -> Maven->Update Project**

# Configuración de eclipse y wildfly

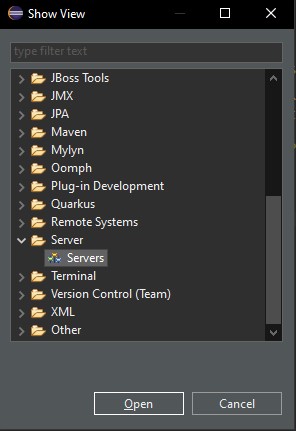
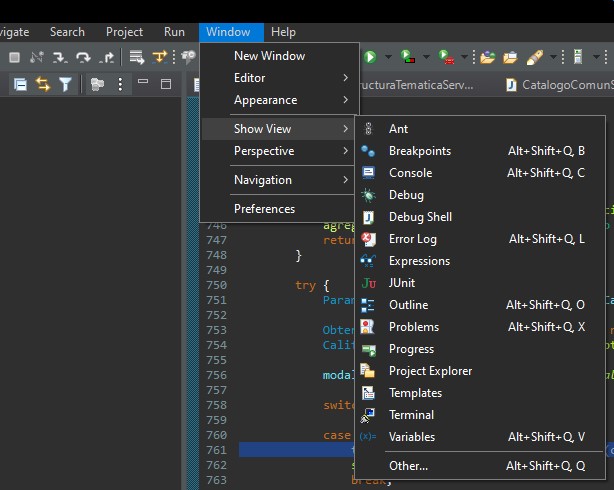
Dentro de eclipse hay que instalar **wildfly**, primero entramos a la **marketplace** de **eclipse**:



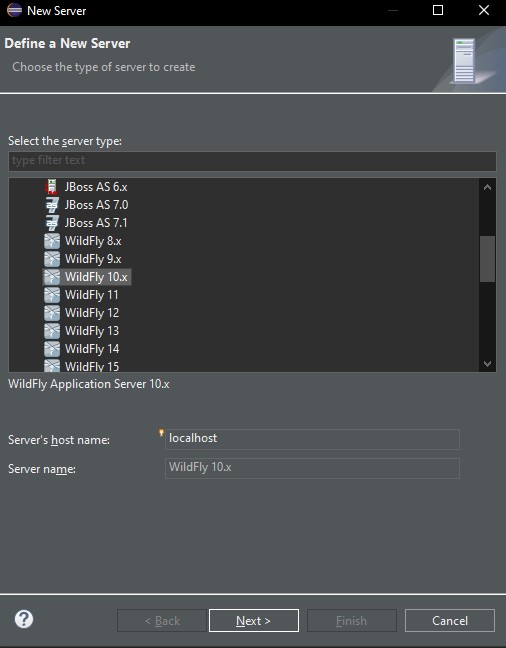
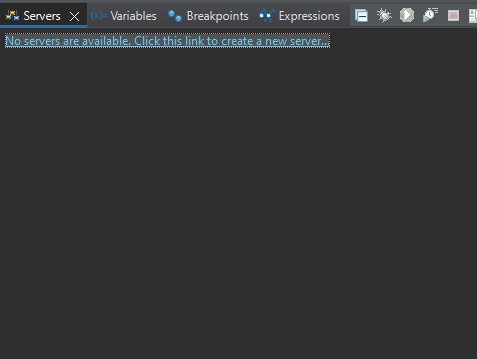
Buscamos “**Jboss tools**” y damos a instalar, después seguimos el proceso mediante el asistente.



Para comprobar que está bien instalado agregamos la vista de servidores, dando clic en **Window** -> **Show view** -> **Other.** Después buscamos **Server**, seleccionamos **Servers** y damos clic en **Open**.



Ahora nos aparece esta parte donde hay que dar clic para crear un nuevo server, en la siguiente ventana hay que desplegar **JBoss Community** y **Wildfly 10.X** y presionamos **Next**.

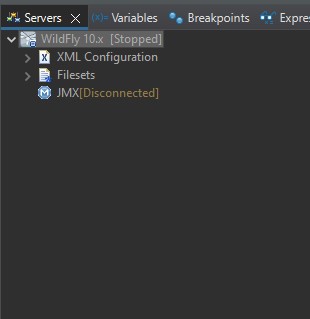
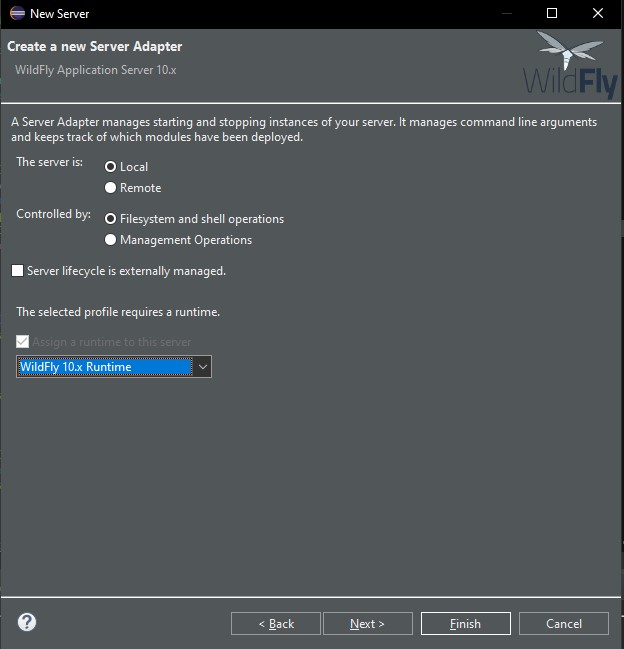


Si pide instalar algo extra lo instalamos, además si pide una ruta de instalación o tiene una por defecto la guardamos para saber dónde quedará instalado Wildfly.

Interfaz de usuario gráfica

Descripción generada automáticamente

En esta pantalla hay que dar **Finish** y ya aparecerá en Servers



Ahora hay que configurar las conexiones a la bd y el driver de MySQL, para esto necesitamos entrar al directorio donde está instalado Wildfly y buscar la ruta wildfly-10.0.0.Final\standalone\configuration y entrar al archivo standalone.xml. Dentro de este archivo buscamos la etiqueta datasources y agregamos el siguiente datasource:

<datasource jta="false" jndi-name="java:jboss/jdbc\_elearning" pool-name="elearningDS" enabled="true" use-ccm="false">

<connection-url>jdbc:mysql://127.0.0.1:3306/des\_sisi\_gestor</connection-url>

<driver-class>com.mysql.jdbc.Driver</driver-class>

<driver>mysql</driver>

<pool>

<min-pool-size>3</min-pool-size>

<max-pool-size>40</max-pool-size>

<prefill>true</prefill>

</pool>

<security>

<user-name>elearning</user-name>

<password>elearning</password>

</security>

<validation>

<validate-on-match>false</validate-on-match>

<background-validation>false</background-validation>

</validation>

<statement>

<share-prepared-statements>false</share-prepared-statements>

</statement>

</datasource>

<drivers>

<driver name="h2" module="com.h2database.h2">

<xa-datasource-class>org.h2.jdbcx.JdbcDataSource</xa-datasource-class>

</driver>

<driver name="mysql" module="com.mysql">

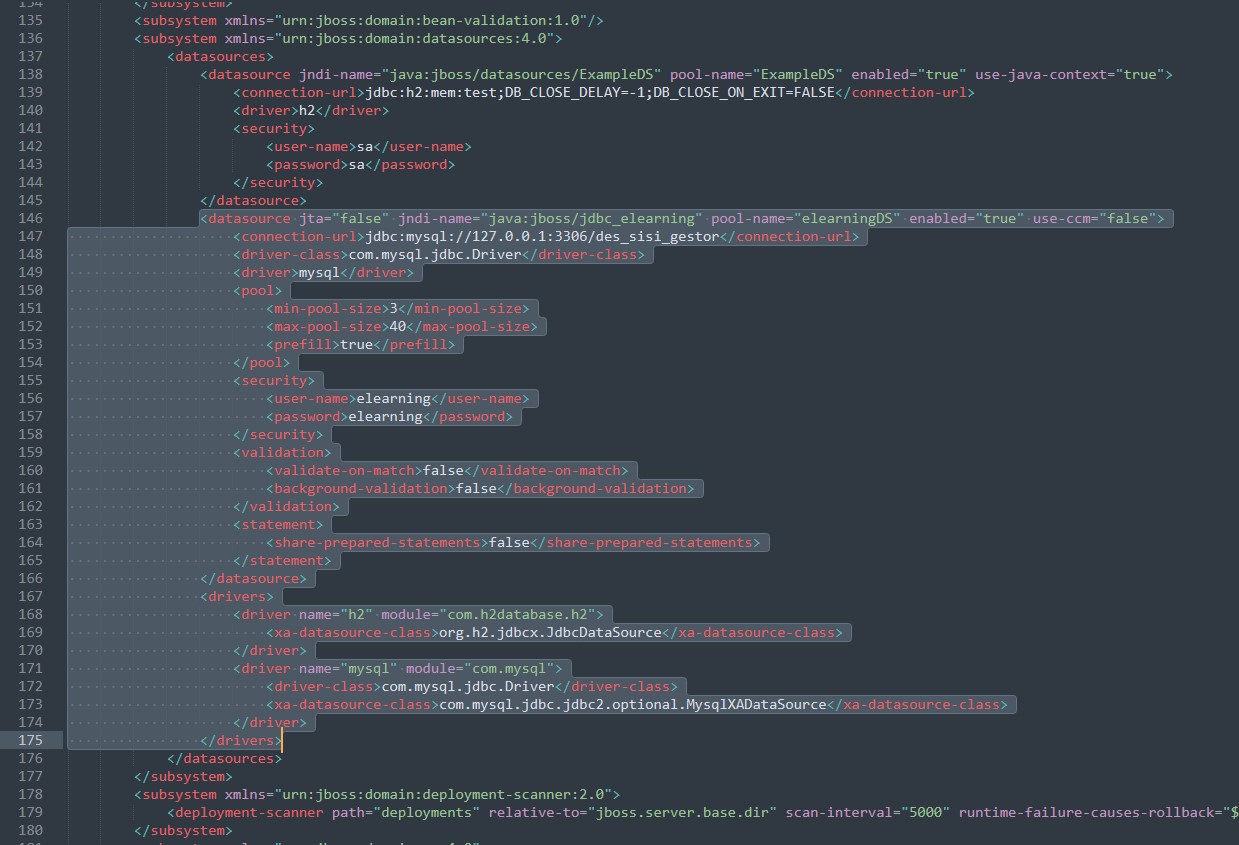
<driver-class>com.mysql.jdbc.Driver</driver-class>

<xa-datasource-class>com.mysql.jdbc.jdbc2.optional.MysqlXADataSource</xa-datasource-class>

</driver>

</drivers>

Es importante no eliminar el datasource ya existente, es decir el código anterior debe ir después de que se encuentra. Nos debe quedar asi:



Ahora hay que agregar el driver a Wildfly, para ello tenemos que buscar la ruta **wildfly-10.0.0.Final\modules\system\layers\base\com\mysql\main**

en caso de faltar carpetas hay que crearlas de forma manual hasta tener dicha ruta. Aquí vamos a pegar el archivo del driver MySQL y crear un archivo llamado module.xml que tendrá lo siguiente: <module xmlns="urn:jboss:module:1.1" name="com.mysql">

<resources>

<resource-root path="mysql-connector-java-5.1.23-bin.jar"/>

<!-- Insert resources here -->

</resources>

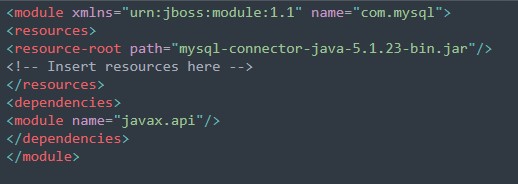
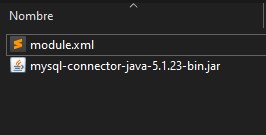
<dependencies>

<module name="javax.api"/>

</dependencies>

</module>

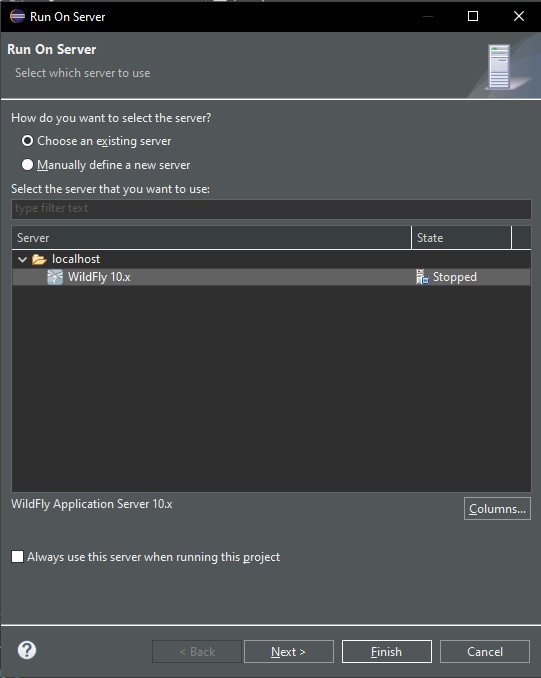
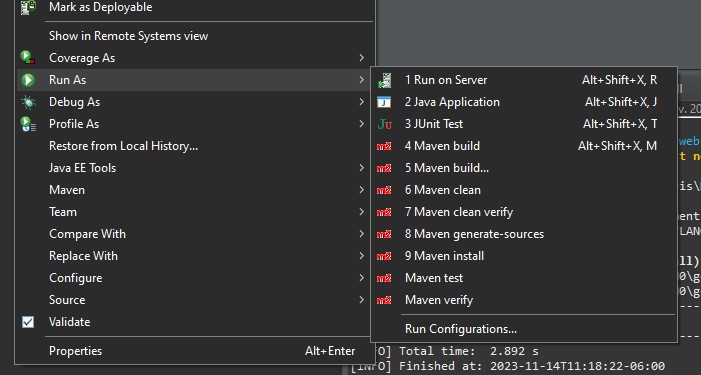
Nota: en la etiqueta resource en el atributo path hay que verificar que tienen el mismo nombre. Nos debe quedar de la siguiente forma.



En caso de fallar o necesitar agregar el driver de manera manual, consultar Configuración WildFly en el manual del archivo gestor\manuales de instalación\ Tutorial de configuración GESTOR.pdf

# Ejecutar proyecto

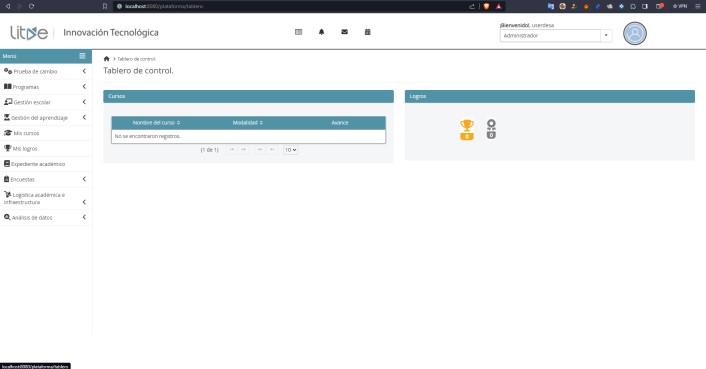
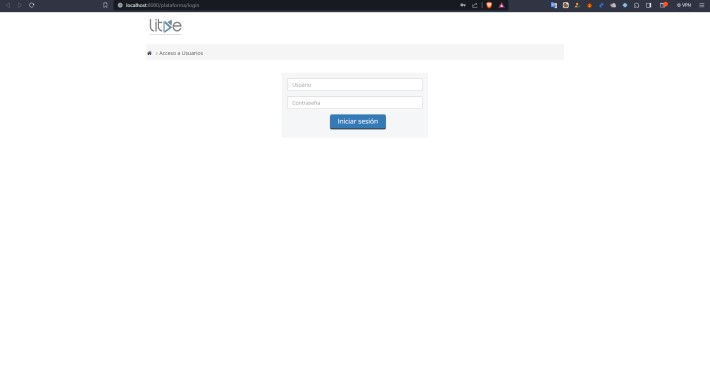
Damos clic derecho al proyecto y seleccionamos **Run As** y damos en **Run on server**. En la siguiente ventana seleccionamos **Choose an existing server** y escojemos **localhost** -> **Wildfly 10.x** y damos en **Finish**



Cuando el proyecto levanta, el acceso es <http://localhost:8080/plataforma/>

Usuario: userdesa

Contraseña: 123456



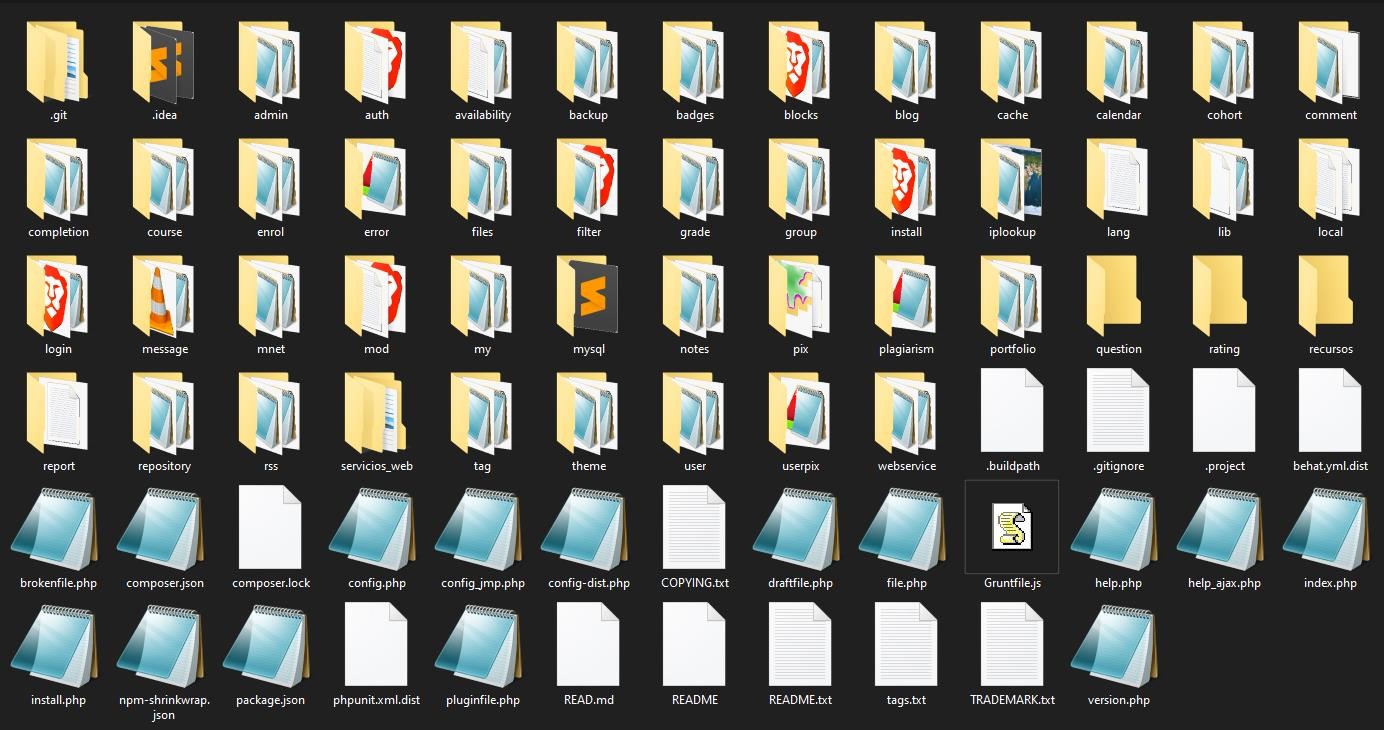
# Instalación de Moodle

Para obtener la versión de Moodle a utilizar hay que descargarla desde:

<https://drive.google.com/drive/u/2/folders/1vVqCpluxC1iYdAtT-tJBv5SLche_14SR>

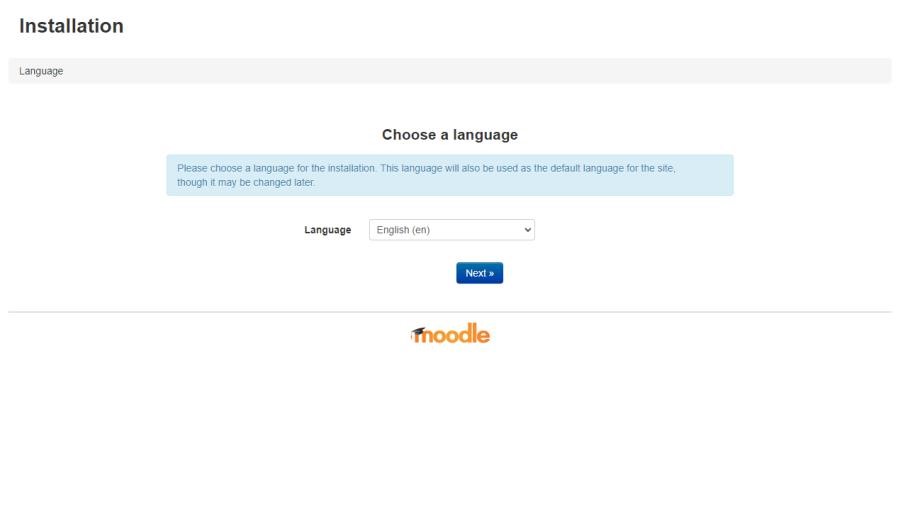
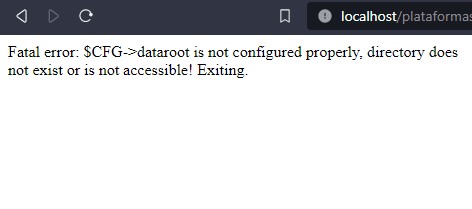
Nota: Es necesario solicitar permiso para poder descargarlo.

Ya con el archivo plataforma.zip lo descomprimimos y tenemos los siguientes archivos:

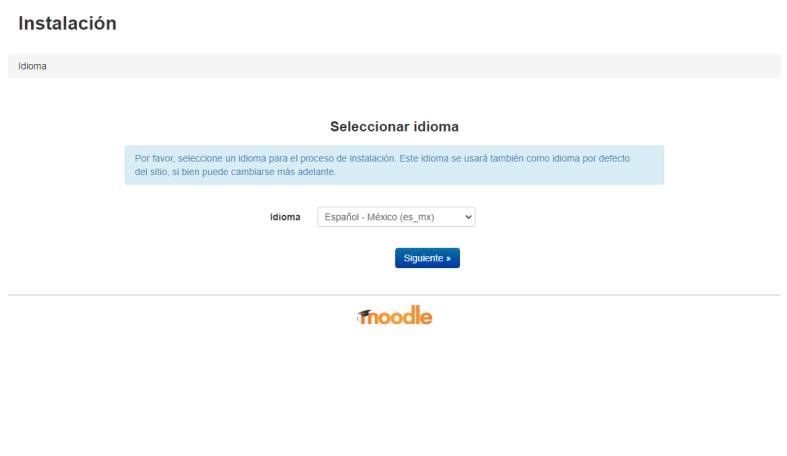


Ahora, esta carpeta la debemos pegar con los demás proyectos de wamp, para ello debe ir en la carpeta wamp64/www. De esta forma podremos acceder en el navegador en la url localhost/”nombre de la carpeta”.

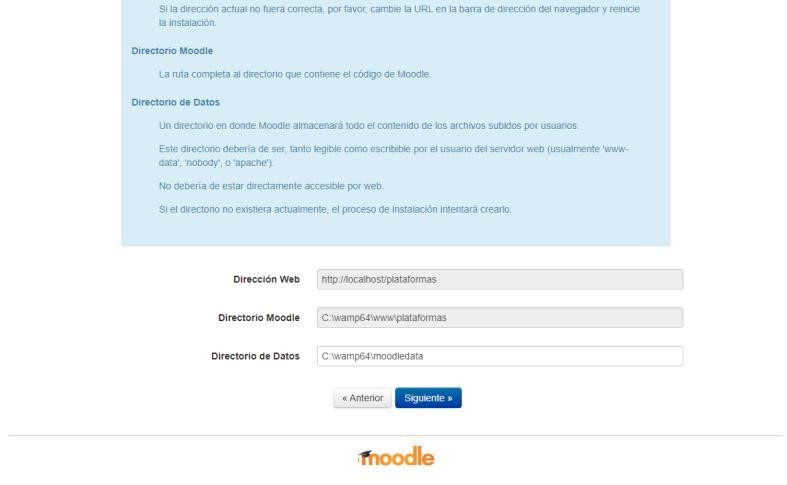
En caso de que apareza el siguiente error solo hay que borrar el archivo config.php en la carpeta de Moodle y aparecerá correctamente el asistente de instalación



Ahora toca configurar Moodle. Seleccionamos idioma **español de Méx**ico y **Siguiente.**



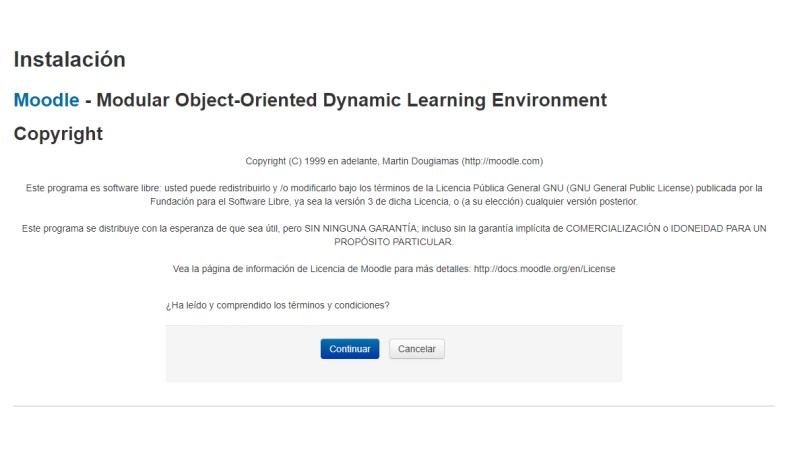
Aquí hay que poner la ruta donde vamos a almacenar los datos que genere Moodle, puede ser cualquier carpeta con los permisos para poder usarse (como estamos en Windows no debería haber problema en una carpeta creada por el usuario). Lo llenamos y damos **Siguiente**.



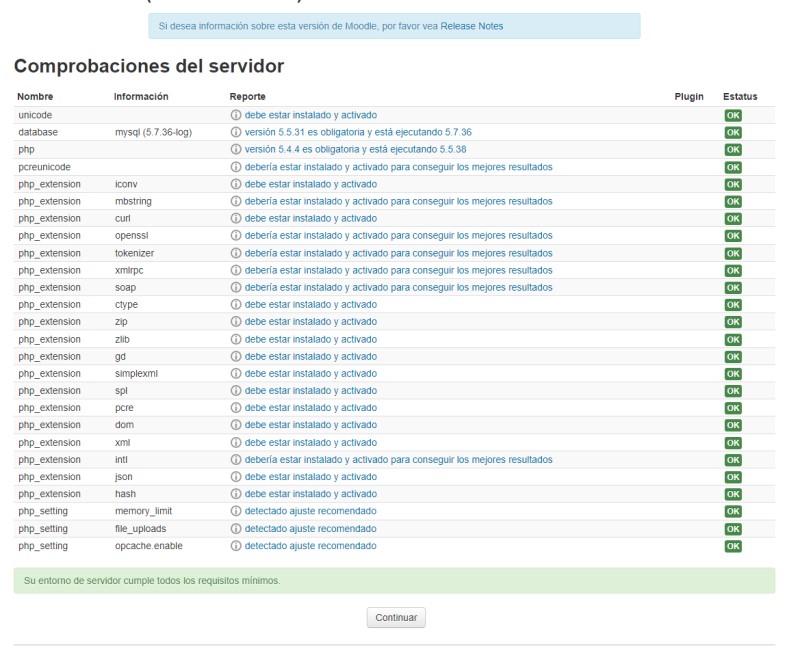
En esta parte hay que seleccionar el tipo de base de datos en **MySQL** y dar **Siguiente**.



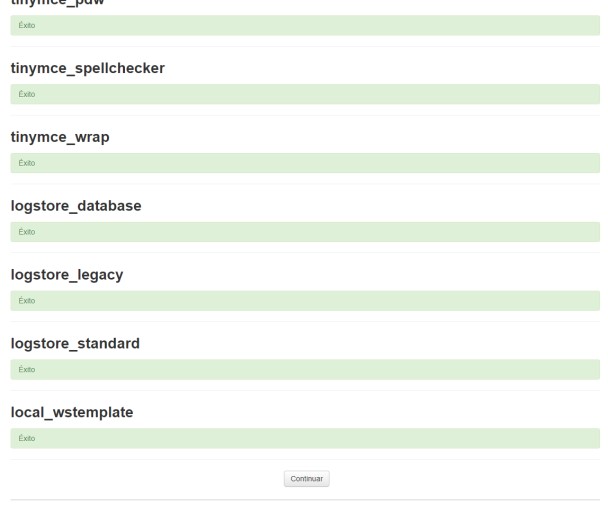
Ahora hay que aceptar esta parte y **Continuar**.



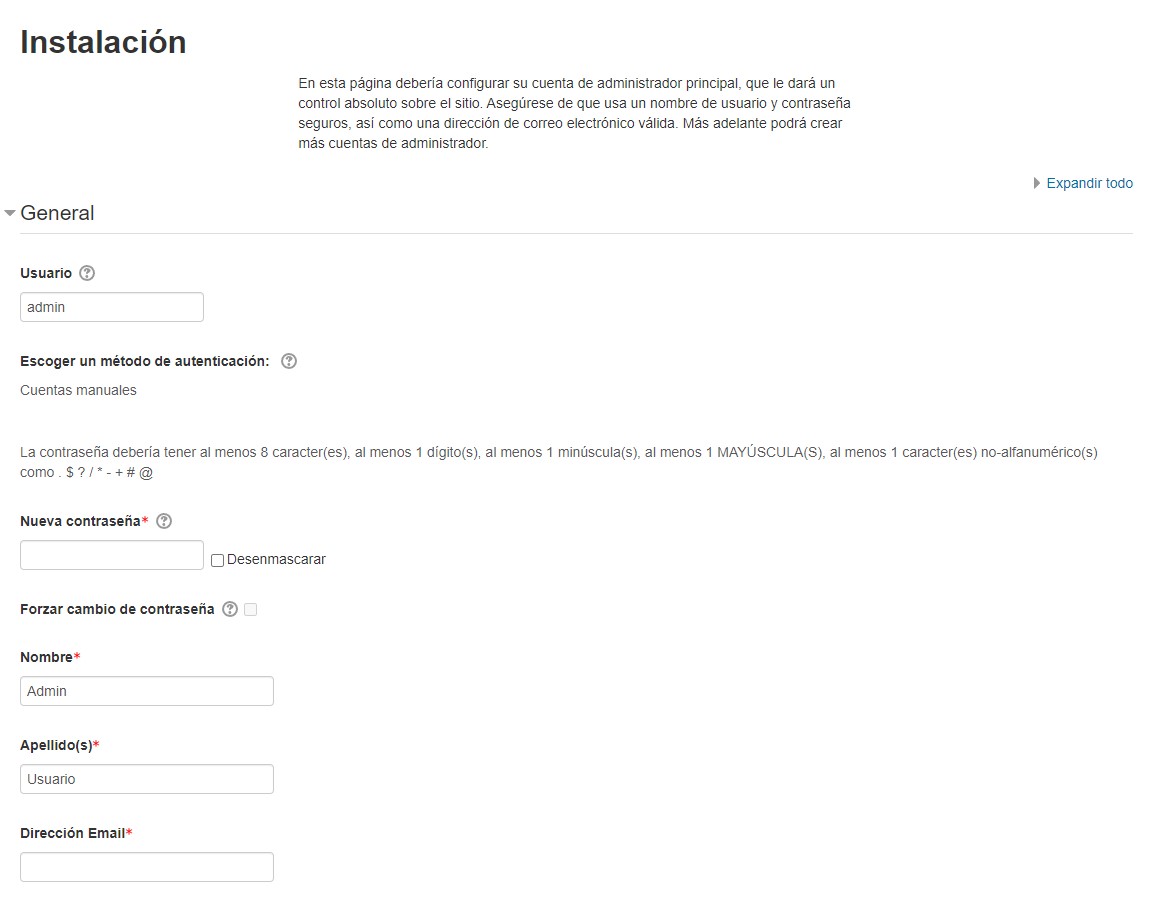
Después tenemos que validar que todos tengan un OK o revisar para poder continuar. Si algo falla y esta en rojo no podremos avanzar por lo que es necesario corregirlo.



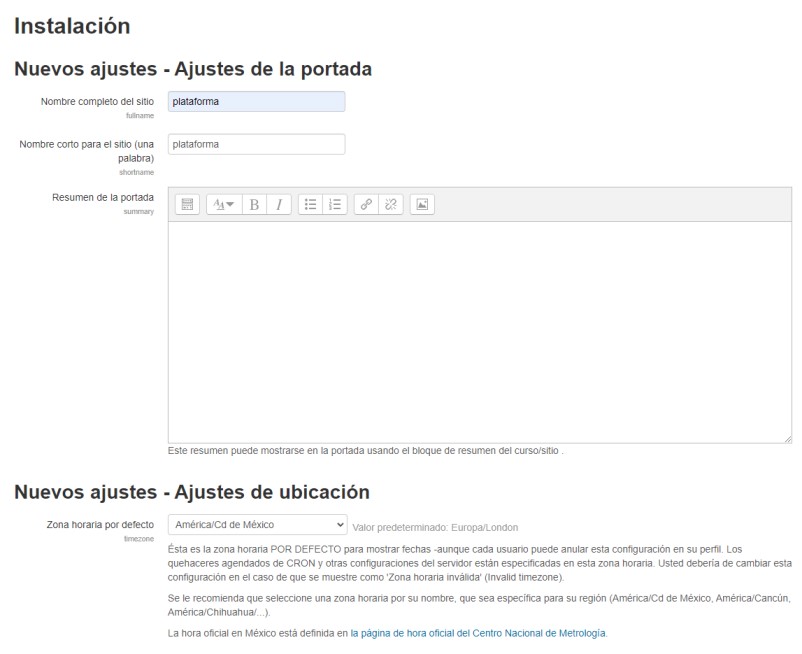
Cuando presionemos **Continuar** empezara la instalación y solo esperaremos hasta que salga el botón de **Continuar** nuevamente.



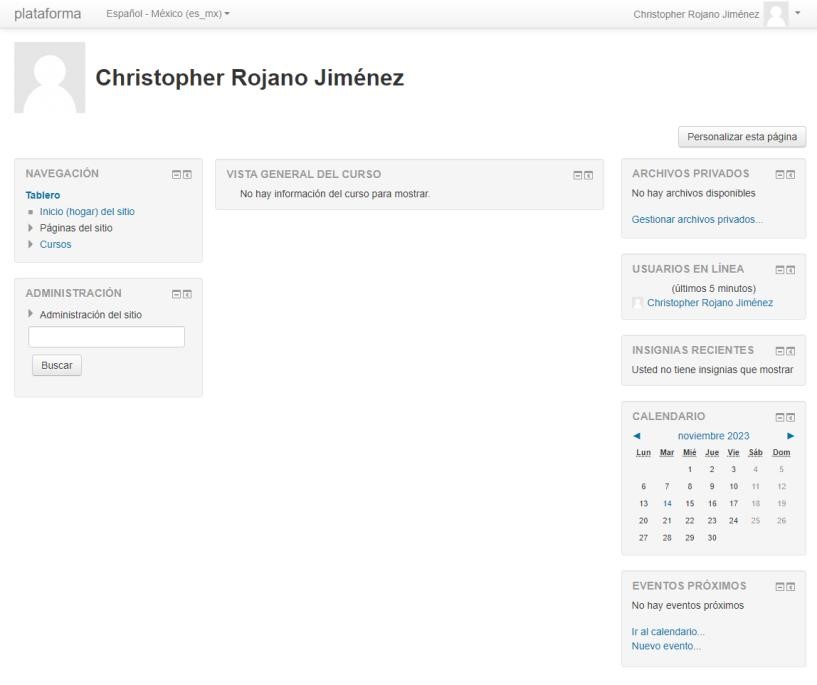
Lo siguiente es configurar la cuenta de administrador. Podemos ingresar los datos que queramos y guardarlos para después. Hacemos clic en Actualizar información personal.



Después debemos poner ciertos ajustes del sistema, aquí hay que asignarle un nombre y un nombre corto al sitio. Luego cambiamos lo zona horaria a America/Cd de México y presionamos Guardar cambios.



Una vez terminado el proceso podremos acceder al panel de administración de Moodle

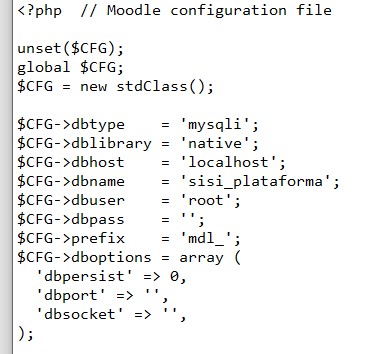


# Configurar Moodle

Lo primero es guardar la contraseña de administrador que pusimos en el sistema. Para ello ejecutamos en MySQL es siguiente query:

SELECT `password` FROM `mdl\_user` WHERE username="nombre\_de\_usuario\_del\_admin";

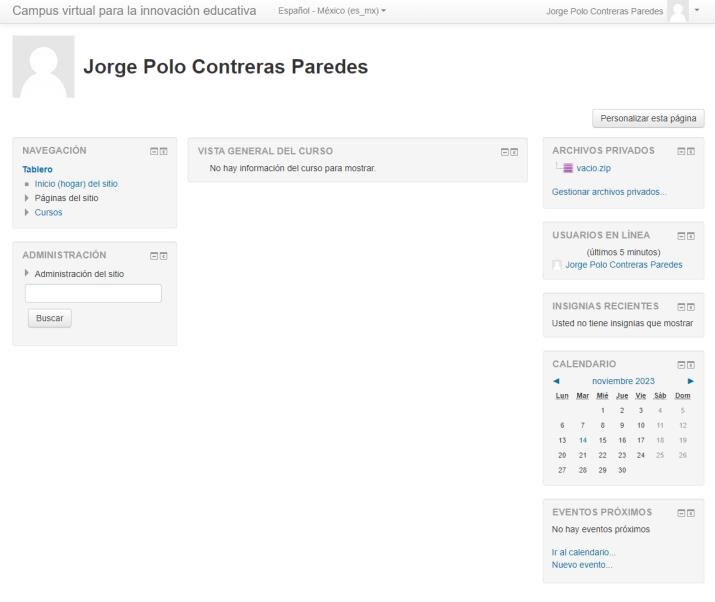
Ahora hay que cambiar la base de datos por sisi\_plataforma, para esto vamos a los archivos de Moodle y buscamos config.php. En dicho archivo cambiamos la bd que está en dbname por sisi\_plataforma. Nos debe quedar así



Ahora en la db sisi\_plataforma ejecutamos el siguiente query:

UPDATE `mdl\_user` SET `password` = 'contraseña\_del\_paso\_anterior' WHERE username="admin";

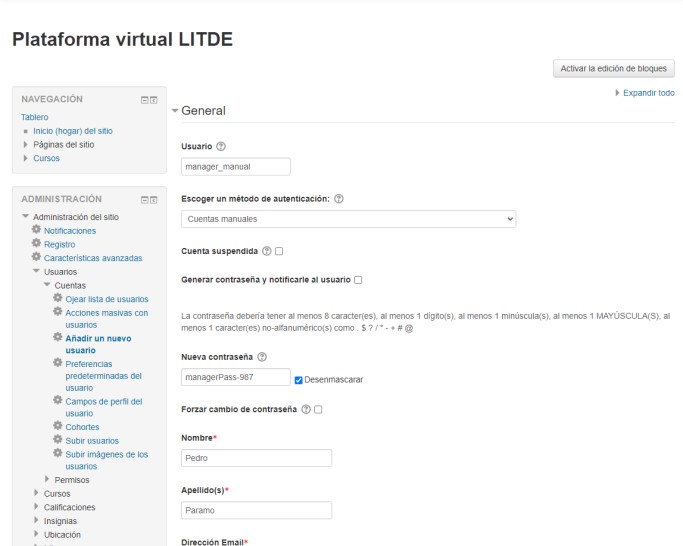
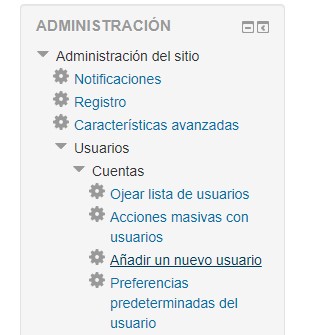
Con esto ahora accedemos al sistema con el usuario admin y la contraseña que usamos al instalar Moodle.



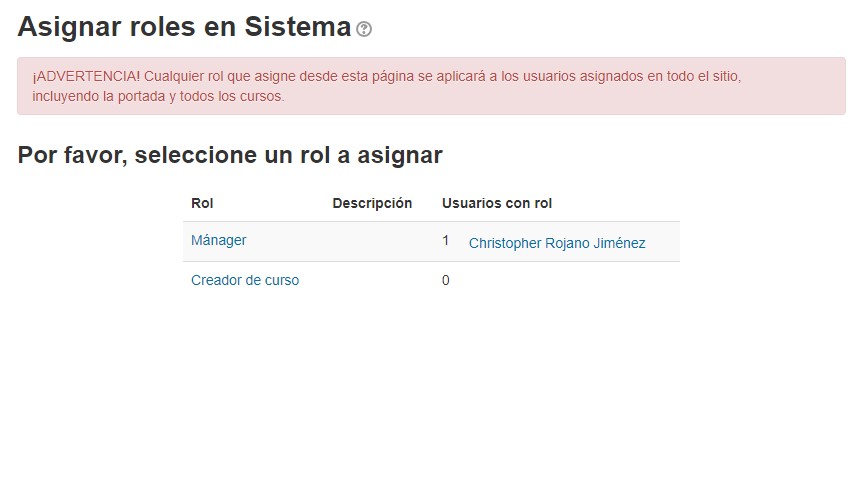
# Conectar Moodle con gestorweb

Primero debemos crear un usuario en Moodle y darle los permisos correctos. Nota: no podemos usar el mismo usuario administrador ya que se debe tener un manager para el tema de crear tokens mediante una petición.

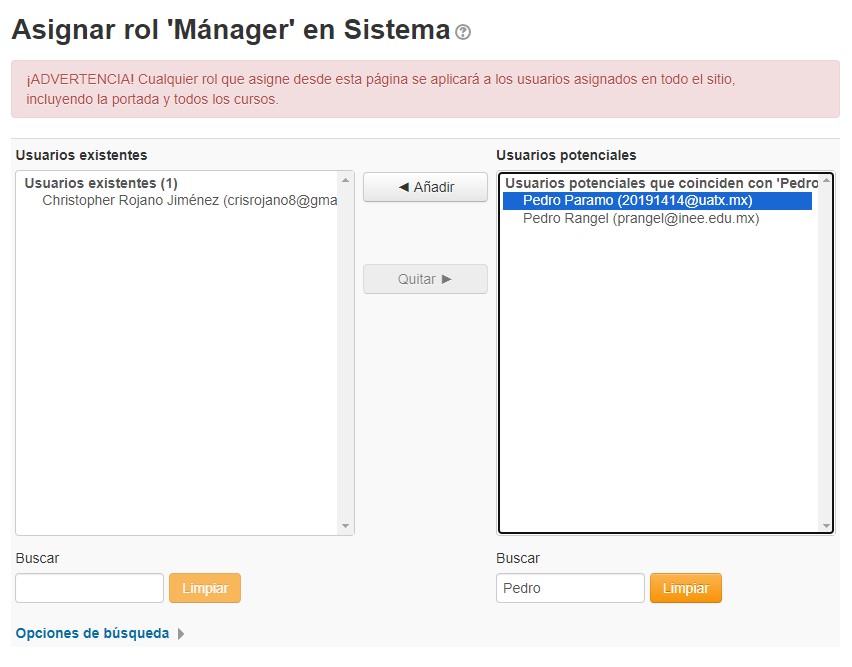
Vamos a **Administración**, a **Usuarios**, luego a **Cuentas** y damos clic en **Añadir nuevo usuario**. En la siguiente ventana llenamos los datos requeridos y presionamos **Crear usuario**.



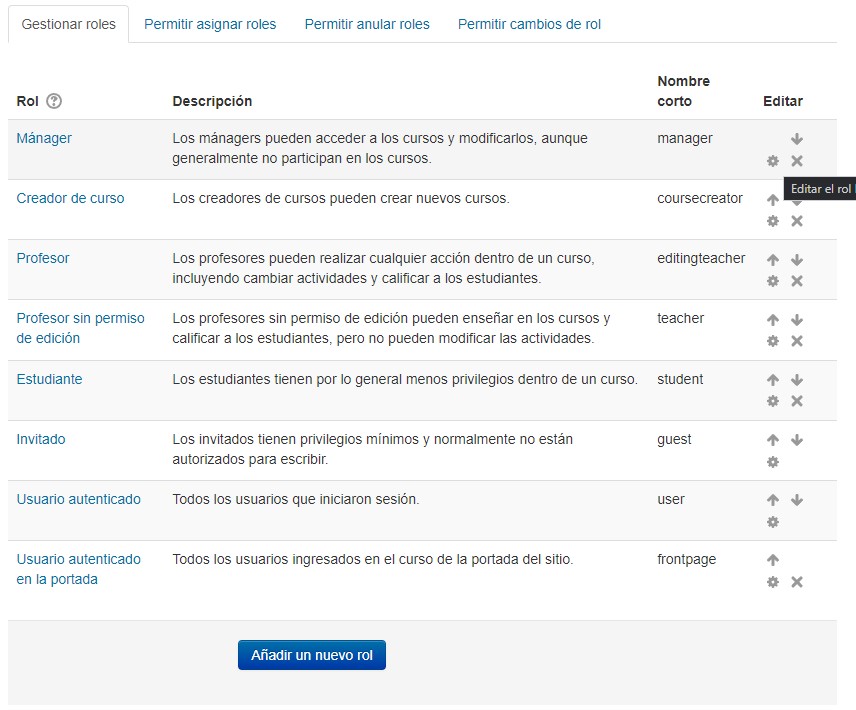
Ahora hay que asignarle el rol correcto. Vamos a **Administración**, a **Usuarios**, luego a **Permisos** y damos clic en **Asignar roles globales**. En la siguiente ventana damos clic en **Mánager**.



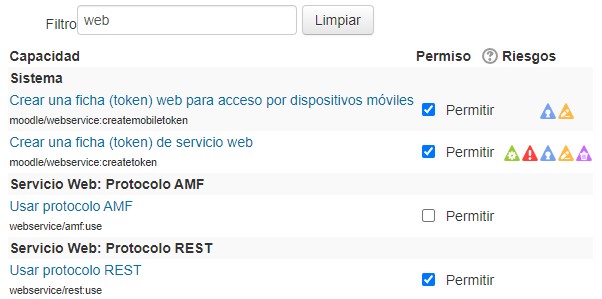
Después hay que poner el nombre dentro del buscador, seleccionar al usuario y dar clic en Añadir, esto le dará automáticamente el rol de Manager.



Ahora hay que comprobar que el rol tenga los permisos necesarios. Vamos a **Administración**, a **Usuarios**, luego a **Permisos** y damos clic en Definir roles. En la siguiente ventana damos clic en el icono de engrane para editar el rol Mánager.



En Editando el rol 'Mánager' bajamos hasta la lista de permisos y comprobamos que tenga los permisos como se muestra en las imágenes



Ahora toca activar los servicios web. Para ello hay que ir a **Administración**, luego a **Características avanzadas**. Después hay que buscar la parte de **Habilitar servicios web** y marcar la casilla si no lo está y darle en **Guardar cambios.**



Lo siguiente es cambiar los accesos en el gestor. Para ello ejecutamos los siguiente queries en la base de datos **des\_sisi\_gestor**:

UPDATE cat\_parametros\_wsmoodle SET `host`="url\_de\_moodle",

`password`="contraseña\_del\_usuario\_manager",

`username`="nombre\_de\_usuario\_del\_manager" WHERE `id\_parametro\_wsmoodle`=1;

Y con el proyecto debería estar funcionando y conectándose con Moodle…