CALONGE BRIEGA, CLARA

LOPEZ ROCHER, ADOLFO

MURILLO ZARZA, RAFAEL

PROYECTO SISTEMAS WEB 1

PROYECTO AHORCADO

INDICE

[1 DOCUMENTACIÓN 2](#_Toc503359150)

[1.1 RESUMEN DEL PROPOSITO DEL PROYECTO 2](#_Toc503359151)

[1.2 LISTA DE FUNCIONALIDADES DEL PROYECTO 3](#_Toc503359152)

[1.3 DOCUEMTO DE DISEÑO, DECISIONES DEL PROYECTO. 5](#_Toc503359153)

[1.4 PUESTA EN MARCHA DEL PROYECTO 7](#_Toc503359154)

[1.4.1 Importar Base de Datos 7](#_Toc503359155)

[1.4.2 Instalar jdk 8 y NetBeans 8](#_Toc503359156)

[1.4.3 Manipulación del código del proyecto 9](#_Toc503359157)

[1.4.4 Conexión con la base de datos y el proyecto 10](#_Toc503359158)

[1.4.5 Ejecución del programa 13](#_Toc503359159)

[2 DIAGRAMAS 14](#_Toc503359160)

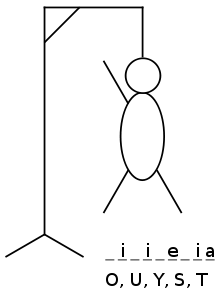
[2.1 UML 14](#_Toc503359161)

[2.2 Diagrama E-R 15](#_Toc503359162)

# DOCUMENTACIÓN

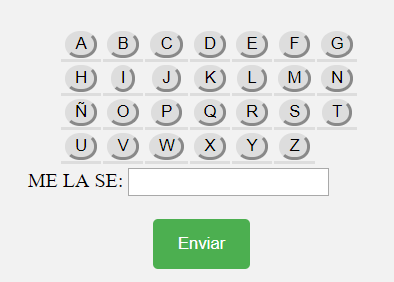
## RESUMEN DEL PROPOSITO DEL PROYECTO

El proyecto consiste en llevar a cabo el *“El juego del Ahorcado”* como aplicación web, el ahorcado clásico (también llamado colgado) es un juego de adivinanzas de lápiz y papel para dos o más jugadores. Un jugador piensa, en una palabra, frase u oración y el otro trata de adivinarla según lo que sugiere por letras.



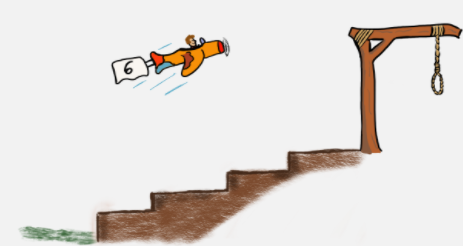
*“Juego clásico del Ahorcado”*

En nuestro caso hemos modificado el juego clásico, será una “competición” contra la máquina, el juego extrae una palabra de la base de datos, la palabra no se elige al azar sino recorre un orden secuencial (todos los usuarios pasarán por las mismas palabras con el mismo orden) y el usuario o jugador debe adivinarla, se mostrará el alfabeto (Español) en botones y el jugador debe ir pulsando según crea cual es la letra que estará en la palabra, también se da la opción de responder directamente a través de una caja de texto.



*“Botones y Caja de texto”*

El jugador tiene 6 vidas o intentos para adivinar la palabra, estas van restando si el jugador dice una letra o palabra incorrecta. La puntuación son las vidas que ha conseguido mantener intactas. También se presentará en el Inicio del juego una media de éxito de palabras adivinadas por el jugador.



*“Las vidas se mostrarán como imágenes de un individuo acercándose a la ahorca con cada fallo”*

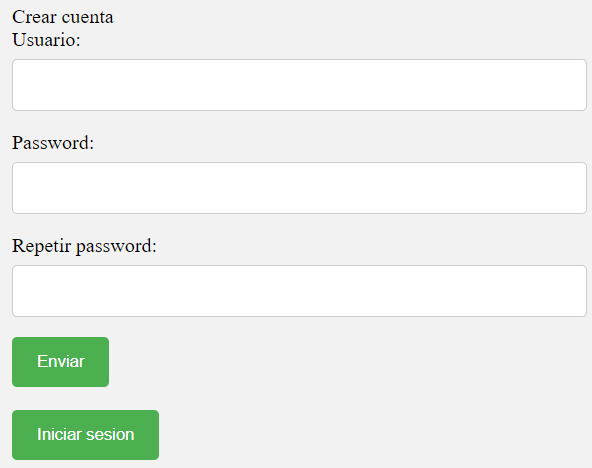
Las principales ventajas de usar una aplicación web frente a una aplicación simple o de escritorio son:

* Portabilidad, se pueden ejecutar desde cualquier ordenador con acceso a internet.
* Aplicaciones muy ligeras, por lo que el usuario no necesita un ordenador muy potente.
* Facilidad de actualizar y mantener.
* Su funcionalidad es independiente del SO.
* No hay incompatibilidad entre versiones, todos trabajan en la misma.
* Los usuarios pueden participar en la elaboración de los contenidos.

## LISTA DE FUNCIONALIDADES DEL PROYECTO

Debería ser capaz de:

* 1. Crear un usuario e insertarlo en la base de datos.

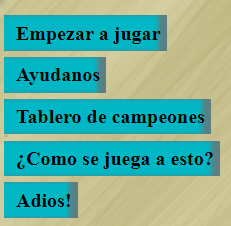


“*Formulario para crear usuario”*

* 1. Iniciar Sesión, comprobar si existe y dar una respuesta correcta si no es así.

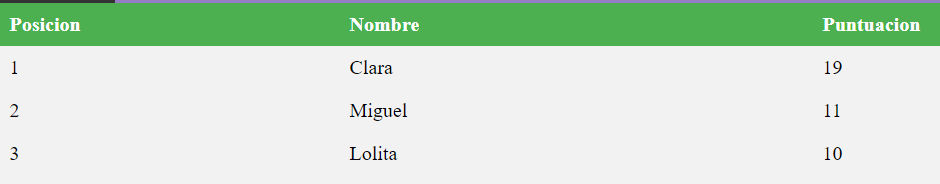
Recorte de pantalla

* 1. En el Inicio del juego dar diversas opciones (Empezar a jugar, ver tablero…)



*“Menú de Inicio”*

* 1. Dar la opción de ver *“Tablero de Campeones”* donde se mostrará la puntuación ordenada de los mejores jugadores.



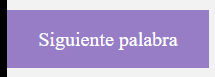
*Ejemplo Tablero*

* 1. Ver un tutorial simple de cómo se juega y las distintas opciones que hay dentro del juego.
  2. La opción de *“Empezar a jugar”* donde te llevará a otra página donde se mostrarán:
     1. La palabra oculta por este signo: ‘\_’ , aparecerán el mismo número de signos que de letras tenga la palabra.
     2. Tabla con 27 botones, los cuales al pulsar deberán cambiar de color y bloquearse. Si la letra se encuentra en la palabra este botón se pondrá en verde y bloqueado, si es incorrecto se pondrá de color rojo y bloqueado.
     3. *Input text* donde se puede introducir un texto, si el texto coincide con la palabra se ganará inmediatamente independientemente del número de letras que falte por adivinar (Si se ha ganado el botón de enviar otra palabra se bloquea), si la palabra no es correcta se resta una vida.
     4. Una imagen que indique el número de vidas o intentos que le quedan al jugador.



*Ejemplo de juego*

* 1. Independientemente se gane o pierda una partida aparecerá un botón que dará la opción de “siguiente palabra”, la palabra (internamente) ya ha cambiado independientemente del jugador lo pulse o no.



* 1. Al terminar la partida se almacenará y actualizará la nueva puntuación, al igual de un registro de si se ha perdido o ganado.
  2. En todo momento aparecerá un menú en la parte de arriba que facilitará en desplazamiento, exceptuando en las pagina de Tutorial, Iniciar Sesión y Crear Usuario.



## DOCUEMTO DE DISEÑO, DECISIONES DEL PROYECTO.

Estos fueron los pasos que fuimos siguiendo a lo largo del proyecto:

* Primer objetivo contar número de intentos con un único botón y saber si la letra está en la palabra o no.
* Insertar un *input text* con un botón de enviar y comprobar que la letra está en la palabra.
* Ir mostrando la palabra según la letra sea o no correcta.
* Condición de ganar o perder.
* Insertar una tabla de botones del alfabeto español, solo envían la letra, ni se bloquean ni cambian de color.
* Página de inicio de sesión empleando un fichero de texto como registro, se debe insertar a mano con 3 columnas Nombre, Contraseña y Puntuación.
* Mostrar una puntuación simple.

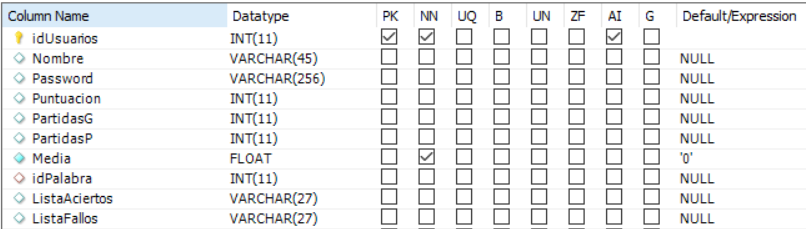
En este punto realizamos la primera presentación del proyecto.

Cambios propuestos o sugeridos, usar una base de datos, añadir página de crear usuario, tutorial, forma de contabilizar algún tipo de puntuación (en mente usar número de letras).

* Problema y solución del envió de la letra ‘Ñ’.
* Creación de la base de datos, con dos tablas:
  + Usuarios
  + Palabras

Evolución de las tablas:

1. Usuarios:
   1. idUsuarios, Nombre, Password, Puntuación, idPalabra. (En el proyecto se usará como primary key el Nombre)
   2. idUsuarios, Nombre, Password, Puntuación, PartidasG, PartidasP, Media, idPalabra.
   3. idUsuarios, Nombre, Password, Puntuación, PartidasG, PartidasP, Media, idPalabra, ListaAciertos, ListaFallos.



1. Palabras:
   1. idPalabra, palabra, tema.
   2. idPalabra, palabra, propietario.



* Añadir las siguientes opciones de Inicio, *“Añadir jugador”* (Opción en mente para dos jugadores), *“Tutorial”* breve información de cómo se juega, *“Tablero de Campeones”* muestra la lista de jugadores sin ordenar.
* Añadimos CSS.
* Añadimos imágenes en la aplicación web, tanto decorativas como vidas.
* Añadir mensajes de error si el usuario no es correcto o se ha creado mal.
* Cambio radical del código, la funcionalidad dividirla de la vista.

Se realiza la segunda presentación del proyecto, cambios que se piden o se sugieren, quitar código de los jsp, guardar partida, dos jugadores, centrarse en los mínimos.

* Quitar código de los jsp, no es posible quitarlo en Tutorial.jsp.
* Revisión de mínimos, añadir Cookie, Listener…
* Lograr guardar partida del usuario.
* Cambio en 2 jugadores, la decisión es que el usuario colabore añadiendo palabras.

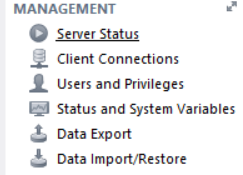
## PUESTA EN MARCHA DEL PROYECTO

### Importar Base de Datos

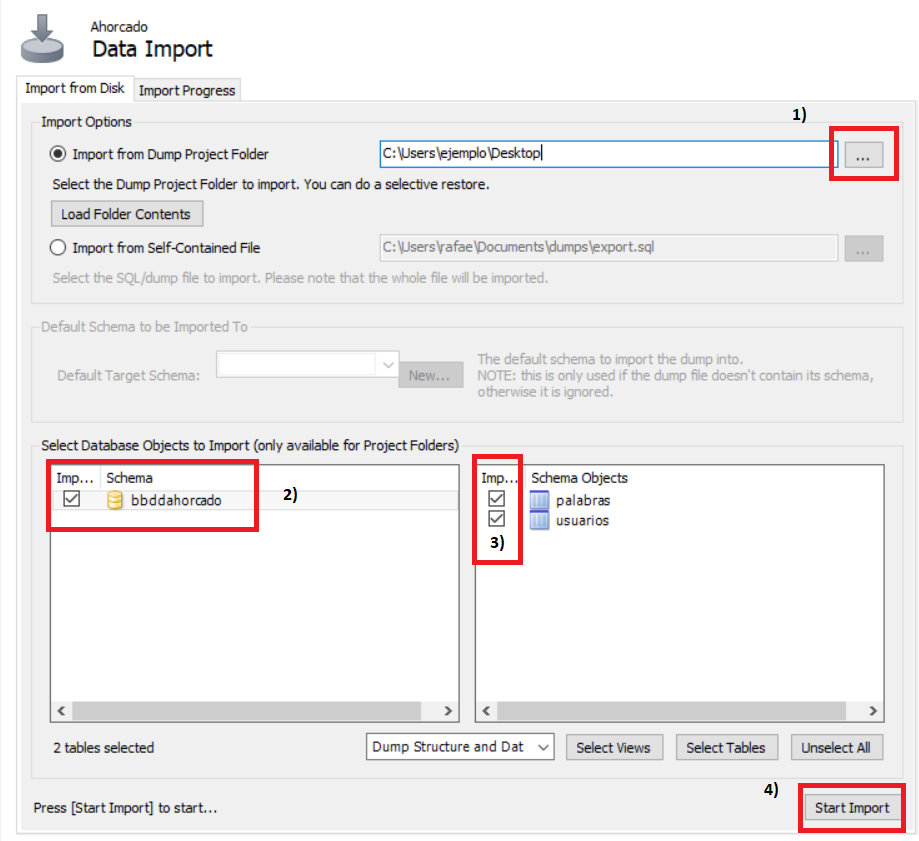
Deberá instalar o tener instalado MySQL Workbench, al igual que MySQL Server, aquí podrá descargar los programas <https://dev.mysql.com/downloads/installer/> .

Una vez descargado e instalado los dos programas debe tener dos ficheros *bbddahorcado\_palabras.sql* y *bbddahorcado\_usuarios.sql*, puede guardarlos por ejemplo en el escritorio (una vez utilizados se pueden eliminar)

Debe abrir MySQL Workbench y crear una nueva conexión (Ejemplo Ahorcado). Dentro debe seleccionar la opción “*Data Import/Restore”.*



*Ultima línea*



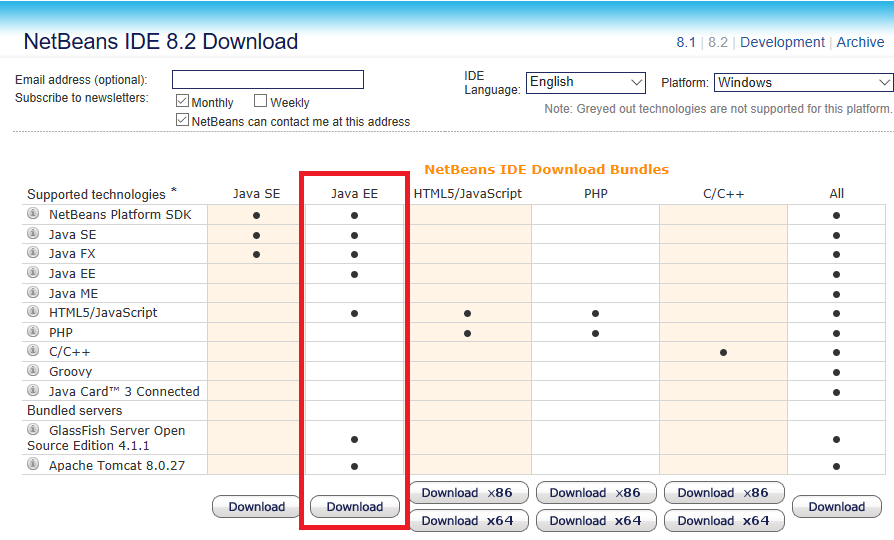
* + 1. Seleccione donde están los 2 ficheros guardados
    2. Una vez seleccionado la ubicación aparecerá inmediatamente la Base de Datos, hacemos clic encima.
    3. Marcamos las dos casillas.
    4. Pulsamos *Start Import*

Una vez realizado estos pasos deberíamos poder trabajar y visualizar con las dos tablas.

### Instalar jdk 8 y NetBeans

Deberá instalar Java jdk, [www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html](http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html) seleccione la última versión (nuestro caso jdk8) y su sistema operativo.

Necesitará NetBeans versión 8.2, con la opción de Java EE. <https://netbeans.org/downloads/>

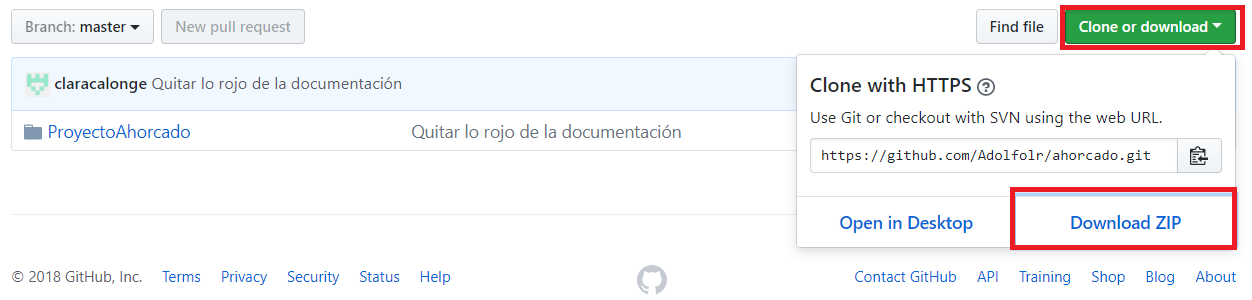


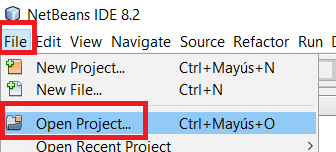


### Manipulación del código del proyecto

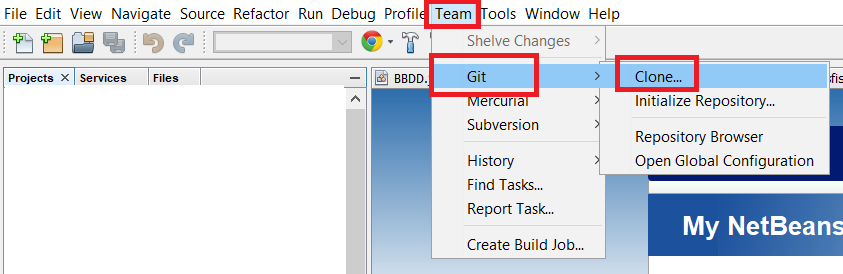
Una vez iniciado NetBeans puede abrir el proyecto o clonarlo a través de github.

Si tiene el proyecto descargado o quiere descargarlo: <https://github.com/Adolfolr/ahorcado>





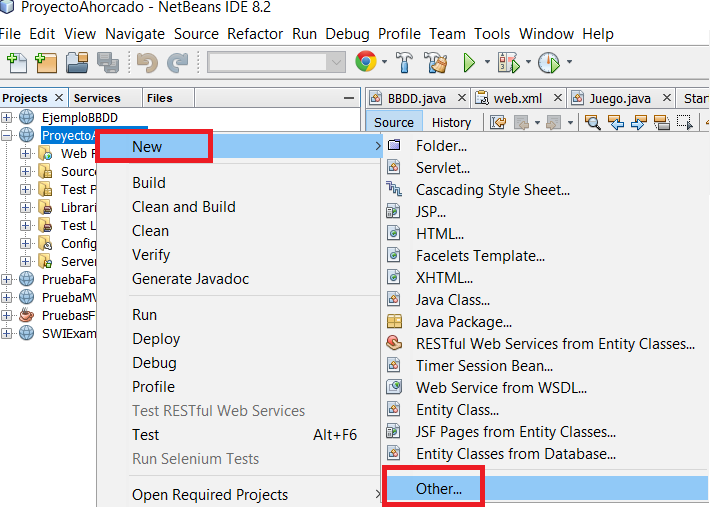
Si desea clonarlo desde Github y colaborar: <https://github.com/Adolfolr/ahorcado.git> (Recomendamos que tenga cuenta en git)



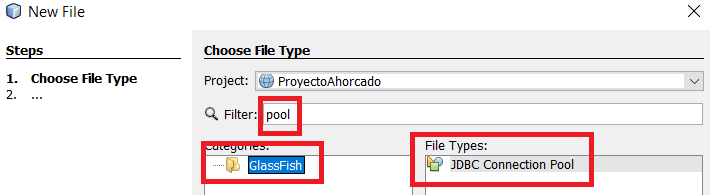
### Conexión con la base de datos y el proyecto

Una vez tenemos la base de datos y el proyecto, debemos generar la conexión.

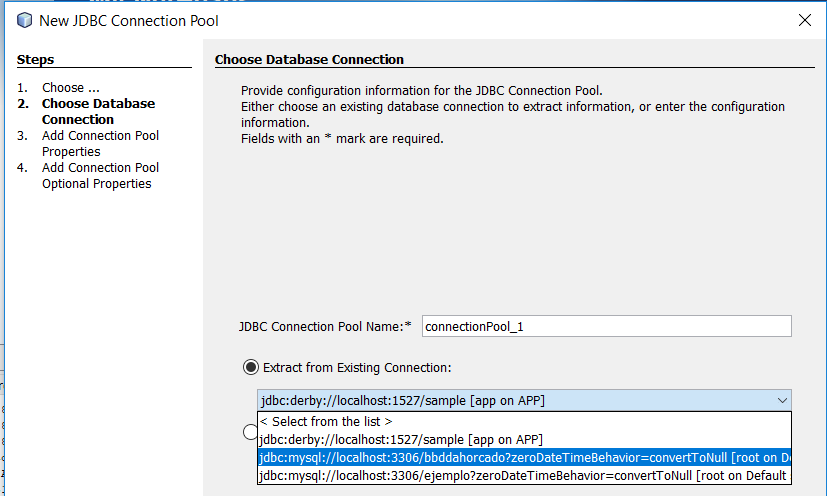
Botón derecho sobre el proyecto y pulsar en New ; Other…



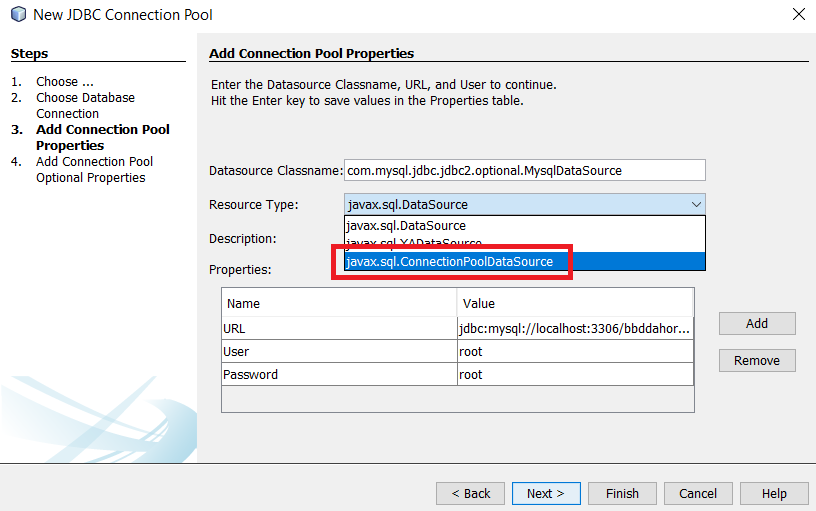
Busque *JDBC Connection Pool*



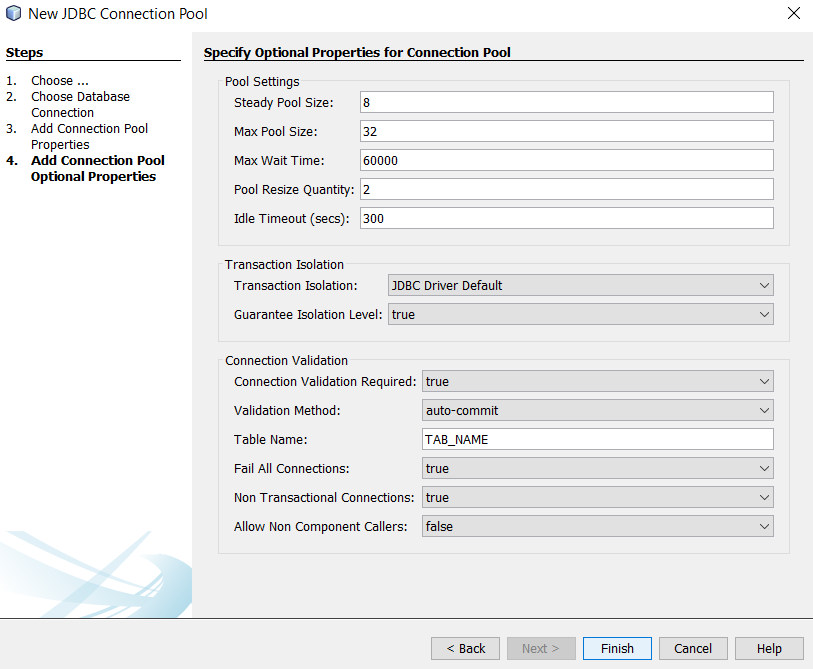
Elija un nombre para la Pool y seleccione la conexión correcta (en nuestro caso se ve marcado en azul)



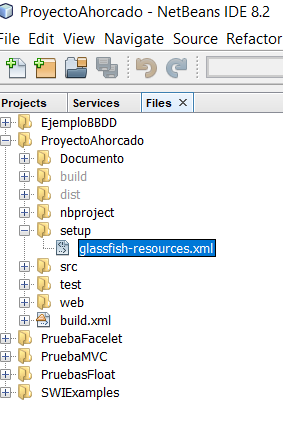
Elija el tipo de conexión:



Dejamos valores por defecto y *Finish.*

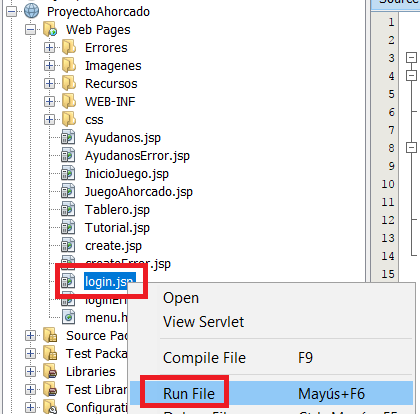


Vamos a la pestaña *Files* y debemos mover el nuevo fichero a la carpeta *setup*

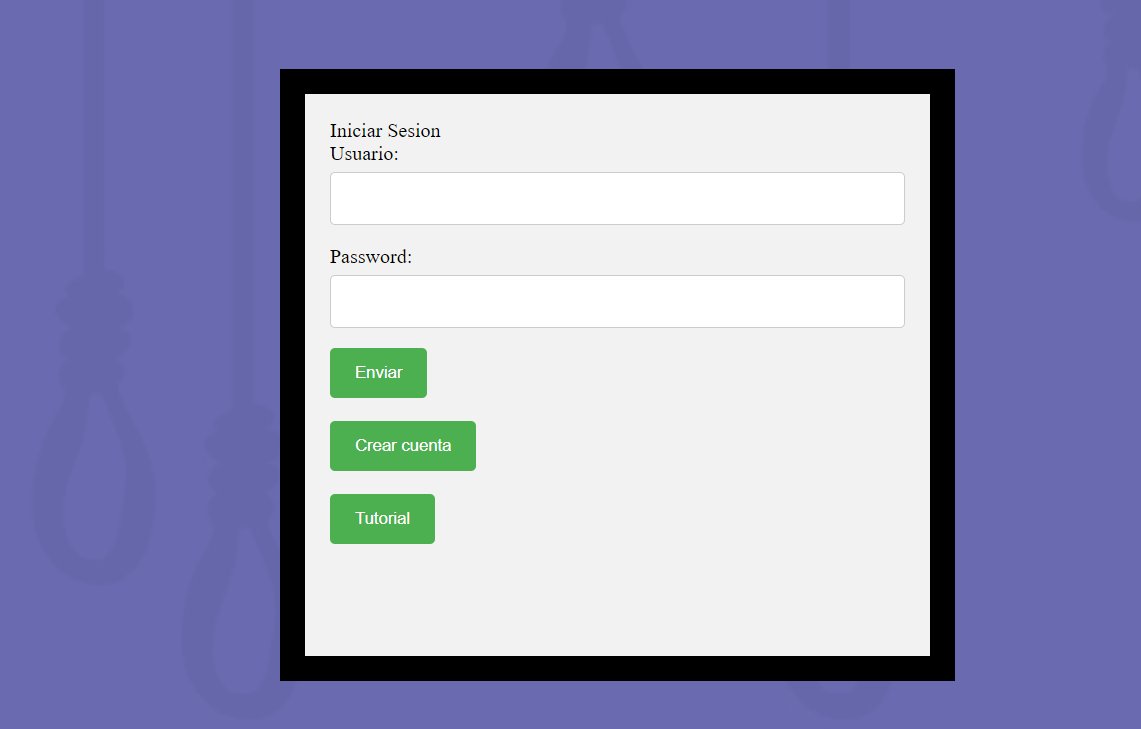


### Ejecución del programa

Vamos a *login.jsp* botón derecho *Run File*



Debería salir esta página (Recomendamos usar Chrome)



# DIAGRAMAS

## UML

## Diagrama E-R

