

Crear en *MySQL* una base de datos llamada **examen**. Si no se va a utilizar el usuario de instalación por defecto **root** con contraseña en blanco, crear el usuario **dwes** y contraseña **dwes** si es que no está creado aún. Importar el archivo **Ex-Eval1.sql** proporcionado.

Construir una pequeña aplicación que muestre una página inicial de **Control de Accesos**. Aquí se comprobará que se introduzcan ambos datos: *Usuario* y *Contraseña*. Estos datos se validarán contra los existentes en la tabla de *usuarios*.

Si falta por introducir alguno(s) de los datos, se mostrará un mensaje indicando cuál(es) falta(n) y no se dejará continuar.

Si el par *usuario/contraseña* no son válidos, se mostrará un mensaje apropiado y no se dejará continuar.



Login

Usuario:

Contraseña:

Debe introducir el usuario
Debe introducir la contraseña



Login

Usuario: ana

Contraseña:

Usuario/Contraseña no válidos

Estas validaciones pueden hacerse tanto desde *JavaScript* como desde el propio archivo PHP. Elegir la que más os guste.

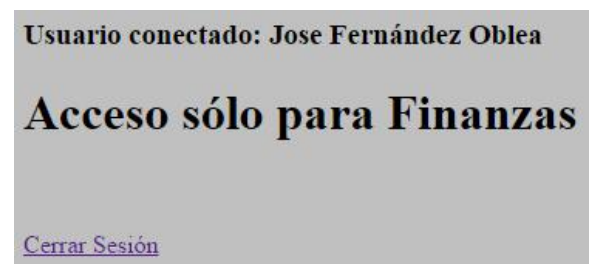
Una vez validado correctamente el usuario y dependiendo de su **rol**, se accederá a una página concreta. Si *rrhh* a la página de RRHH, si *administrador* a la página de Admón., etc. En cualquiera de ellas, se mostrará un mensaje de bienvenida con el **nombre** y **apellidos** del usuario.



Usuario conectado: Ana García González

Acceso sólo para RRHH

[Cerrar Sesión](#)

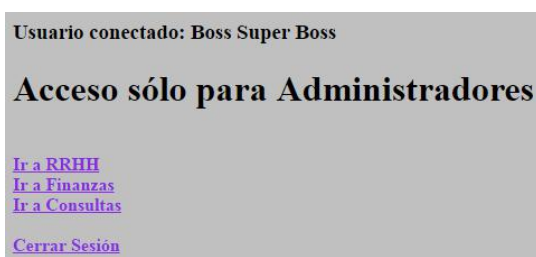


Usuario conectado: Jose Fernández Oblea

Acceso sólo para Finanzas

[Cerrar Sesión](#)

Estas páginas tendrán un enlace para **desconectar la sesión** del usuario. Cuando se pulse sobre dicho enlace se *destruirá por completo la sesión vigente* y se regresará automáticamente a la página inicial de *Control de Accesos*.



Usuario conectado: Boss Super Boss

Acceso sólo para Administradores

[Ir a RRHH](#)
[Ir a Finanzas](#)
[Ir a Consultas](#)
[Cerrar Sesión](#)

El usuario con rol *administrador* accederá a una página donde podrá acceder a las páginas de *RRHH* y de *Finanzas*, así como realizar unas consultas.

Cuando se pulse sobre **Ir a Consultas**, se accederá a una página donde se mostrará la relación de *usuarios* existente en la base de datos.

Usuario conectado: **Boss Super Boss**

Relación de usuarios

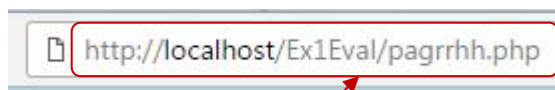
| Usuario | Rol | Nombre | Apellidos |
|---------|-------|--------|-----------------|
| admin | admin | Boss | Super Boss |
| ana | rrhh | Ana | García González |
| jose | fin | Jose | Fernández Oblea |

[Regresar menú anterior](#)

[Cerrar Sesión](#)

La pantalla de consultas tendrá dos enlaces: uno para regresar a la pantalla anterior y otro para cerrar la sesión.

Además, si un usuario malintencionado conociese la URL de cualquier página de la aplicación e intentase acceder a ella de forma directa, se le debe **redigir obligatoria y automáticamente** a la página de *Control de Accesos* y **mostrar un mensaje apropiado**.



Si se intenta acceder directamente **sin autenticarse** a la página de RRHH o a cualquier otra de la aplicación que no sea la de Login, entonces

REDIRECCIONAR



De forma similar, si un usuario malintencionado de *RRHH* intenta acceder a la página de *Consultas*, *Finanzas* o *Admón*, no se le dejará. Igualmente, para el usuario de *Finanzas*.

Un usuario con rol *Administrador*, **sí** que podrá **acceder** a cualquiera de las demás páginas.

Opcionalmente, añadir una columna a la tabla de consultas que tenga un enlace que lleve a una página que tenga formulario de modificación del usuario. En este formulario, solamente se podrán modificar el nombre, apellidos y contraseña del usuario. Y, lógicamente, a esta opción solamente se podrá acceder si se está conectado como usuario con rol de administrador.

Notas: Es conveniente, antes de empezar a codificar, **estructurar** la aplicación. Qué páginas y cuántas se van a necesitar. Qué tipo de variables utilizar para contemplar los diferentes mensajes que se generan en la aplicación, fundamentalmente después de retornar a una página. Cómo limpiar dichos mensajes cuando no son necesarios, etc. Determinar qué páginas son similares. De esta manera se puede codificar y probar solamente una y, una vez dado el visto bueno, hacer una copia de la(s) otra(s).

Se pueden utilizar todos los apuntes que se deseen pero no está permitido copiar de otro compañero ni realizar un copy-paste total o parcial del código de los ejercicios resueltos y entregados en clase.

Entrega del examen: colgar una carpeta comprimida que contenga vuestro nombre y dentro de ésta la carpeta entera del proyecto PHP en el Aula Virtual.

Para la calificación de la prueba se tendrá en cuenta:

- La aplicación deberá poder ejecutarse, en caso contrario, no se evaluará.
 - Cumplir y ajustarse a toda la funcionalidad requerida. No sirve hacer algo distinto aunque sea parecido. Esto incluye el nombre de la BD y el acceso a la misma con los usuarios indicados.
 - No mostrar mensajes por pantalla incontrolados, fundamentalmente *warnings* de variables no inicializadas.
 - Se valorará la claridad del código.
 - Se valorará el uso de funciones.
 - Se valorará el reducir al máximo el código PHP dentro de la sección cliente (HTML).
-
- Login → **1,5 puntos** máximo.
 - Relación de usuarios → **1,5 puntos** máximo.
 - Creación de la Sesión y uso correcto de la misma a lo largo de todas las páginas → **1 punto** máximo.
 - Uso adecuado del rol. Esto incluye el impedir el acceso de un usuario de RRHH o de Finanzas a las otras páginas → **2 puntos** máximo.
 - Redirección automática a Login más mensaje si no se ha establecido la sesión o acceso inapropiado a página que no le corresponde → **1 punto** máximo.
 - Destruir adecuadamente la sesión establecida → **0,5 puntos** máximo.
 - Uso de funciones PHP → **0,5 puntos** máximo.
 - Reducción al máximo del código PHP dentro de la sección cliente (HTML) → **0,5 puntos** máximo.
 - Enlace y página de modificación de datos de usuario → **1,5 puntos** máximo.