

## Laboratorio 8: Sesiones en PHP

### Descripción

Se complementará la aplicación elaborada en el laboratorio pasado para integrar funciones de login y manejo de sesiones.

**Modalidad:** Colaborativa (parejas).

### Objetivos

- Aprender a interactuar con sesiones que registren datos provenientes del usuario u otra fuente de datos.
- Practicar la interacción HTML-PHP por medio de formularios.
- Aprender a interactuar con un modelo de datos en formato JSON para recuperar datos específicos.
- Aprender a manipular información proveniente de un modelo de datos basado en el formato JSON.

### Instrucciones

Utilizando lo realizado en el laboratorio anterior, deberás diseñar y elaborar uno o más archivos en formato JSON que soporten el siguiente modelo de datos:

#### Usuarios

Representan los usuarios registrados en el sistema con sus respectivos niveles de acceso a la funcionalidad.

- **id:** número de identificación del usuario. Solo el sistema lo conoce y utiliza para hacer eficiente el acceso.
- **username:** identificador textual utilizado por el usuario para entrar al sistema.

- **password:** contraseña que utilizará el usuario para entrar al sistema. Puedes almacenarla en texto plano o encriptarla de algún modo si así lo prefieres.
- **nombre:** nombre del usuario.
- **apellidos:** apellidos del usuario.
- **fecha de nacimiento:** fecha de nacimiento del usuario en el formato de tu preferencia (te recomiendo YYYY-MM-DD).
- **tipo de usuario:** identificador del tipo de usuario que le corresponde.

### Tipos de usuarios

Representan los tipos de usuarios existentes en el sistema. Tendrás que definir algún mecanismo para nombrarlos e identificarlos.

- **Id:** identificador numérico del tipo. Solo el sistema lo conoce y utiliza para hacer eficiente el acceso.
- **Nombre:** nombre corto, asociado al tipo, que se mostrará en la interfaz.
- **Descripción:** texto para incluir información adicional y descriptiva sobre el tipo.

### Reglas del modelo de datos

- El id de usuario es único y se genera automáticamente en al insertar un registro.
- El username debe ser único.
- Las fechas se deben guardar en formato “YYY-MM-DD” o como timestamp.
- El id del tipo de usuario es único.

Utilizando como base la aplicación desarrollada en el laboratorio anterior, añadir lo necesario para implementar las siguientes páginas:

- **Login:**
  - Formulario que contendrá un campo para nombre de usuario y otro para contraseña (debe mostrar las letras enmascaradas) y un botón para enviar los datos al servidor.
  - Los datos se enviarán a un script específico que procesará los datos.

- **Registro de Usuarios:**
  - Formulario que contendrá los campos necesarios para registrar un usuario (de acuerdo al modelo de datos), y un botón submit.
  - Los datos se enviarán a un script específico que procesará los datos.
- **Listado de Usuarios:**
  - Página que mostrará los usuarios existentes actualmente en el sistema.
- **Perfil de Usuario:**
  - Página que mostrará la información del usuario autenticado en un formato amigable.
  - Esta página será el index del proyecto.

Las páginas deberán integrarse como aplicación, esto quiere decir que deberás incluir un menú de navegación que te permita ir a las diferentes páginas sin necesidad de escribir las URLs en el navegador. Deberá ser consistente en cuanto al diseño y al layout. También deberá tener un link/botón de logout que cerrará la sesión.

A continuación se describen algunas reglas de negocio generales para la aplicación:

- Todas las páginas deberán tener los datos de los integrantes del equipo, actividad y materia en algún lugar visible que no obstruya el diseño.
- Las únicas páginas que podrán verse sin autenticación son “Registro de Usuarios” y “Login”.
- Debe haber dos tipos de usuarios (al menos):
  - **admin:** tendrá acceso a toda la funcionalidad del sistema y su página “home” será “Listado de Usuarios”.
  - **user:** tiene acceso a toda la funcionalidad menos al “Listado de Usuarios”. Su página “home” será “Perfil de Usuario”.
- Solo se pueden registrar usuarios de tipo “user” mediante la página “Registro de Usuarios”, los administradores deben estar precargados.

Utilizando PHP, desarrollar lo necesario para implementar la siguiente lógica de la aplicación:

- **Login**

1. Verificar si el usuario se encuentra autenticado.
2. Si está autenticado lo envía al “home” correspondiente a su tipo.
3. Si no está autenticado, se muestra el formulario de login.
4. El usuario introduce el usuario y contraseña y presiona “enviar”.
5. El sistema verifica si el usuario existe.
  - a. Si existe lo envía a la página “home” correspondiente su tipo.
    - i. Se deberá almacenar el nombre del usuario, su tipo y hora de login en la sesión.
  - b. Si no existe entonces lo envía a la pantalla “Registro de Usuarios”.

- **Registro de usuarios**

1. El sistema verifica si el usuario ha iniciado sesión.
2. Si ya inició sesión lo envía a la pantalla “home” correspondiente a su tipo.
3. Si no ha iniciado sesión muestra el formulario de registro.
4. El usuario introduce sus datos y los envía.
5. El sistema deberá hacer las validaciones necesarias y después almacenar el usuario.
6. Después de almacenar el usuario lo deberá enviar a la pantalla “Perfil de Usuario”.

- **Listado de usuarios**

1. El sistema verificará que el usuario esté logueado.
2. Si no está logueado lo mandará a la pantalla “Registro de Usuarios”.
3. Si está logueado y es de tipo “admin”, mostrará el listado de usuarios del sistema en un formato presentable.
  - a. La página deberá mostrar el mensaje “Bienvenido <nombre>” en algún lugar visible.

4. Si está logueado y no es de tipo “admin” lo redigirá a la página “Perfil de Usuario”.
- **Perfil de usuario**
    1. El sistema verificará que el usuario esté logueado.
    2. Si no está logueado lo enviará a la página “Registro de Usuarios”.
    3. Si está logueado mostrará sus datos en un formato presentable.
      - a. No se deberá mostrar el ID ni el password.
      - b. El tipo de usuario deberá mostrarse textualmente tomando el nombre del tipo correspondiente.
      - c. La fecha de nacimiento deberá estar en un formato amigable para el usuario.
      - d. La página deberá mostrar el mensaje “Bienvenido <nombre>” en algún lugar visible.
  - **Logout**
    - Será un botón/link que se habilitará en la barra de navegación cuando el usuario esté autenticado en el sistema, al hacer click en él se deberá terminar la sesión y enviar la pantalla “Registro de Usuarios”.

No hay restricciones en cuanto a la cantidad de archivos PHP desarrollados, tienes la libertad de diseñar tu aplicación y su implementación siempre que cumpla con los requisitos planteados.

## **Recursos**

- Suehring, S., & Valade, J. (2013). PHP, MySQL, JavaScript & HTML5 AllinOne For Dummies (1). Somerset, US: For Dummies. Retrieved from <http://0www.ebrary.com.millenium.itesm.mx>.
- W3schools. <http://www.w3schools.com>.

## **Especificaciones de entrega**

1. Crea una carpeta nombrada como tu matrícula en el directorio del curso en drive entregas/laboratorio8/{tu\_matricula}.
2. Sube los archivos fuente generados en una carpeta comprimida.
3. Se debe entregar antes de las **23:30 horas del día lunes 25 de noviembre**.