

José María Segovia Marín

Practica entregable (Enunciado y entregable)

ICA0009-UF1-PR01

Guía para el alumno

El alumno debe de entregar la práctica enunciada en este documento antes del cierre programado en el calendario. La nota y corrección de la práctica se publicará en la plataforma en un plazo no superior a 10 días hábiles a contar desde la fecha límite de entrega.

Los entregables son un fichero zip o rar que contenga:

* Este mismo documento incluyendo las respuestas solicitadas, los dibujos y las capturas de pantallas indicadas en el mismo:
  + [CSnn]: son capturas de pantalla que demuestran el resultado de una práctica. Para realizar la captura se utilizará la tecla “imp pant” o equivalente del teclado, y luego al final de documento y dentro de la página en blanco habilitada para cada captura, se realiza el “pegado” o “paste”. Si fuera necesario se ajustará el tamaño.
  + [DIBnn]: Son esquemas que se solicitan en el enunciado de la práctica y pueden ser realizadas con cualquier herramienta dependiendo del tipo de dibujo, incluso escanear la imagen de un dibujo hecho a mano. También pueden ser completar tablas, escribir texto.

Utilizar los espacios habilitados al final de este documento para la inserción de las capturas y dibujos.

El documento entregado tendrá el siguiente nombre:

|  |
| --- |
| ICA0009-UF1-PR01-“username”.doc  “username” = nombre de usuario del alumno en la plataforma  Ejemplo: ICA0009-UF1-PR01-raulgarciaflores.doc |

* Código fuente de la aplicación con el siguiente nombre:

|  |
| --- |
| ICA0009-UF1-PR01-“username”  “username” = nombre de usuario del alumno en la plataforma  Ejemplo: ICA0009-UF1-PR01-raulgarciaflores |

Ejercicio #1 Creación de una “Aplicación de experiencias LA SALLE”

Los objetivos de este ejercicio es que te familiarices con:

* El lenguaje PHP orientado a servidores
* El uso de formularios HTML
* Interacción de la aplicación con una base de datos

Para resolver este ejercicio se recomienda haber consolidado los conocimientos expuestos en: ICA0009-S01, ICA0009-S02 y ICA0009-S03.

El objetivo de esta práctica es realizar una “Aplicación de experiencias LA SALLE”. Esta aplicación consistirá en un buscador de paquetes de actividades LaSalle-Box. dónde tendrás un listado de actividades (cenas, alojamientos, actividades…), deberás ser capaz de crear un filtro para las actividades y estas aparecerán a modo de tabla de la que podremos manipular.

La aplicación web será alojada en un servidor Apache y accederá a una base de datos MySQL. Los lenguajes a usar para esta aplicación serán PHP para la gestión de la aplicación, y los lenguajes de marcas HTML y CSS.

Al final de esta práctica, añadiremos dos roles distintos de usuarios:

* Usuario “user”: solamente será capaz de visualizar datos y filtrarlos.
* Usuario “admin”: además de las funciones de “user” y además podrá modificar, insertar e incluso borrar los datos.

Este ejercicio se divide en 8 tareas, en cada una de ellas viene especificada su puntuación. Ten en cuenta que es imprescindible que las capturas de pantalla [CSnn] relacionadas estén correctamente realizadas para que se pueda tener en cuenta la puntuación del apartado correspondiente. Esto servirá para demostrar la evolución de la práctica.

Puntuación: 10

Ejercicio #1 - Tarea #1

Crea una base de datos MySQL junto con una tabla llamada “Actividades”. Esta tabla deberá contener como mínimo la siguiente información a modo de columnas (id, nombre, tipo y descripción).

Añade como mínimo 4 tipos distintos y 2 actividades por tipo.

Puedes usar el siguiente modelo de ejemplo:

|  |  |  |  |
| --- | --- | --- | --- |
| ID | NOMBRE | TIPO | DESCRIPCIÓN |
| 1 | Excursión en kayac | AVENTURA | Excursión en kayac de 2h descendiendo el río |
| 2 | Cena restaurante japonés | CENA PARA 2 | Cena para 2 personas en restaurante japonés, menú de noche |
| 3 | Escapada romántica | ALOJAMIENTO | Fin de semana de ensueño con detalle de bienvenida y desayuno incluido |

[CS01] Captura de pantalla de la tabla “Actividades” creada junto con la información añadida

Puntuación: 0,5

(+0,25) Estructura

(+0,25) Datos

Ejercicio #1 - Tarea #2

Gestiona la conexión con la base de datos. Asegúrate antes de nada que puedes conectarte a la base de datos y que la tienes activa.

Puedes gestionar la conexión como en clase (a través un fichero de conexión), cuenta que debes de tener las credenciales siempre parametrizadas y gestionar los posibles errores que pueda ocurrir durante el proceso de conexión.

Crea además un fichero “index.php”. Este fichero contendrá una cabecera con un título de la aplicación, junto con un formulario que servirá para seleccionar el tipo de actividad a filtrar.

En los atributos del formulario deberás detallar que el envío de la información se realizará mediante el método GET, y como acción, se deberá redirigir la aplicación hacia el fichero “read.php” (ver Tarea #3).

[CS02] Captura de pantalla con las pruebas de acceso a la base de datos

[CS03] Captura de pantalla con la vista del formulario

Puntuación 2:

(+0,5) Gestión de la conexión y credenciales

(+0,5) Control de errores

(+1) Formulario

Ejercicio #1 - Tarea #3

En esta tarea deberás crear un fichero llamado “read.php”, este se encargará de visualizar los datos de la tabla “Actividades”.

El fichero deberá conectarse a la base de datos, leer la tabla y posteriormente mostrar su contenido en una tabla con formato HTML, deberás aplicar estilos CSS.

Una vez funcione, aplica un filtro para el tipo de actividad. A través de la url (envío por GET) te debe de llegar el tipo de actividad a filtrar, tu consulta solamente deberá mostrar las actividades correspondientes a dicho filtro.

Fíjate con el ejemplo a continuación:

*p.e. localhost/<nombre proyecto>/read.php?tipo=aventura*

|  |  |  |  |
| --- | --- | --- | --- |
| ID | NOMBRE | TIPO | DESCRIPCIÓN |
| 1 | Excursión en kayac | AVENTURA | Excursión en kayac de 2h descendiendo el río |

Añade también un botón o enlace para volver cómodamente a “index.php”.

Muestra a través de pantallazos como testeas éste código.

[CS04] Captura de pantalla con la visualización de todos los datos

[CS05] Captura de pantalla con las pruebas del envío de datos por GET

[CS06] Captura de pantalla con la visualización con los datos filtrados

Puntuación 2:

(+0,5) Formateo de la tabla con estilos

(+0,5) Recibo de tipo por GET

(+1) Visualización de datos

Ejercicio #1 - Tarea #4

En esta tarea tendrás que añadir la funcionalidad de insertar nuevas actividades. Modifica el fichero “read.php” para que aparezca una fila en blanco junto con un botón de añadir.

Una vez se accione el botón, se llamará a un nuevo fichero “insert.php”.

Este fichero realizará las siguientes acciones:

* Recibirá los datos a insertar
* Establecerá la conexión con la base de datos
* Insertará el nuevo registro
* Llamará de nuevo a “read.php” para actualizar los datos

Muestra a través de pantallazos como testeas éste código.

[CS07] Captura de pantalla con las pruebas de la modificación en “read.php”

[CS08] Captura de pantalla con las pruebas de inserción

Puntuación 1:

(+0,5) Lógica del fichero insert.php

(+0,5) Formulario de inserción

Ejercicio #1 - Tarea #5

En esta tarea tendrás que añadir la funcionalidad de modificación y borrado de actividades. Modifica el fichero “read.php” para que aparezcan dos botones en cada fila, uno para la modificación y otro para el borrado de la actividad en cuestión.

* Funcionalidad de modificación: se deberá poder modificar cada uno de los campos disponibles. Crea un fichero “update.php” que te permita cambiar los valores antiguos por nuevos. Al acabar su función, debemos volver a la vista de la tabla con los nuevos valores.
* Funcionalidad de borrado: se deberá poder borrar la actividad seleccionada. Crea un fichero “delete.php” con la capacidad de borrado del índice elegido.

Muestra a través de pantallazos como testeas éste código.

[CS09] Captura de pantalla con las pruebas de la modificación de actividades

[CS10] Captura de pantalla con las pruebas del borrado de actividades

Puntuación 2:

(+1) Funcionalidad de modificación

(+1) Funcionalidad de borrado

Ejercicio #1 - Tarea #6

Ahora añadiremos a la aplicación la gestión de usuarios.

Vuelve a la base de datos y crea una tabla de “usuarios”. En dicha tabla deberán aparecer como mínimo dos usuarios (admin y user). Guarda en esta tabla el nombre de usuario y una contraseña a tu elección.

Fíjate con el ejemplo a continuación:

|  |  |  |
| --- | --- | --- |
| ID | rol | password |
| 1 | admin | gasED&)76%hladg)& |
| 2 | user | Guoew%(KJH%&877 |

|  |  |
| --- | --- |
|  | **Hint**: como bien sabes, no es una buena práctica guardar un password en una base de datos a no ser que esté encriptado. Aplica el algoritmo MD5 al password antes de almacenarlo.  https://www.php.net/manual/en/function.md5.php |

Muestra a través de pantallazos como testeas éste código.

[CS11] Captura de pantalla con la tabla “usuarios”

[CS12] Captura de pantalla con las pruebas del cifrado de password

Puntuación 0,5:

(+0,25) Estructura y datos de la tabla “usuarios”

(+0,25) Encriptado del password

Ejercicio #1 - Tarea #7

Antes de entrar en la aplicación, el usuario tendrá que abrir su sesión. Crea un nuevo fichero “login.php” para gestionarlo. Dicho fichero será llamado siempre que no exista una sesión abierta.

El fichero “login.php” mostrará primero de todo, un formulario de login. La aplicación tendrá que acceder a la tabla “usuarios” y comprobar si existe el usuario introducido.

|  |  |
| --- | --- |
|  | **Hint**: para comparar la contraseña introducida con la guardada, tendremos que encriptarla de nuevo. Los pasos son los siguientes:   * Capturar el password del usuario por formulario * Encriptarlo con el algoritmo md5 * Comparar el resultado, con el password encriptado de la base de datos |

En caso de login correcto, la aplicación se redirigirá hacia el fichero “index.php”.

Muestra a través de pantallazos como testeas éste código.

[CS13] Captura de pantalla con el formulario de login

[CS14] Captura de pantalla con las pruebas de autentificación de password

Puntuación 1:

(+0,5) Formulario de login

(+0,5) Lógica de autentificación

Ejercicio #1 - Tarea #8

Cada usuario tiene roles distintos con permisos y restricciones propios. Los roles serán los siguientes:

* Usuario “user”: solamente será capaz de visualizar datos y filtrarlos.
* Usuario “admin”: además de las funciones de “user” y además podrá modificar, insertar e incluso borrar los datos.

Para gestionar los permisos de cada usuario, tendremos que controlar cada una de las acciones y mostrar un mensaje de alerta en caso de que el usuario acceda a una acción sin permiso.

Aplica condicionales sobre cada una de las acciones para verificar los permisos. Las acciones a controlar serían las siguientes:

* Visualización: todos los usuarios deben de poder ver los datos
* Filtro: todos los usuarios de deben de poder filtrar
* Inserción: solamente el usuario “admin” podrá insertar nuevos datos
* Modificación: solamente el usuario “admin” podrá modificar los datos
* Borrado: solamente el usuario “admin” podrá borrar los datos

Muestra a través de pantallazos como testeas éste código junto con los mensajes de alerta.

[CS15] Captura de pantalla con las pruebas control de permisos

Puntuación 1:

(+0,5) Lógica de control de permisos

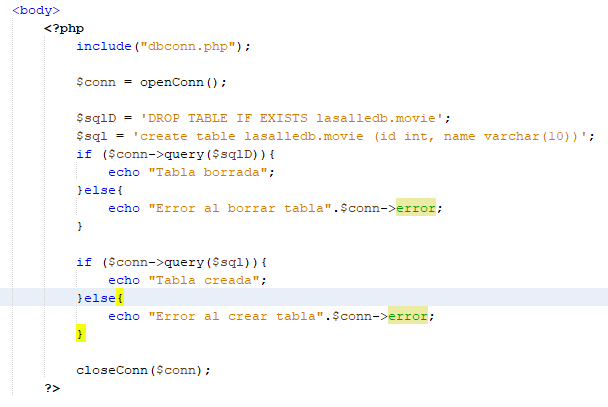
(+0,5) Mensaje de alerta

**Capturas de pantalla**

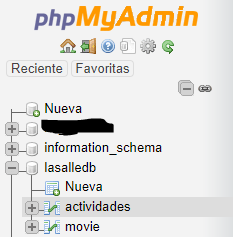
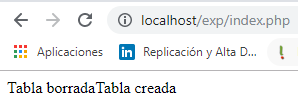
[CS01] Captura de pantalla de la tabla “Actividades” creada junto con la información añadida



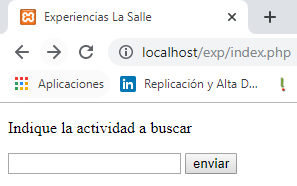
[CS02] Captura de pantalla con las pruebas de acceso a la base de datos







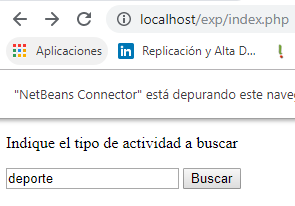
[CS03] Captura de pantalla con la vista del formulario

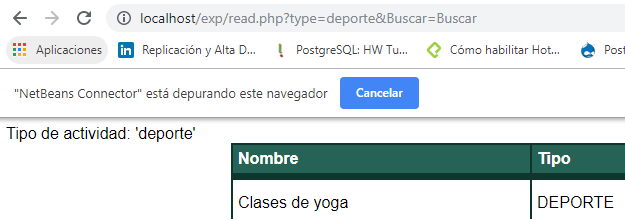


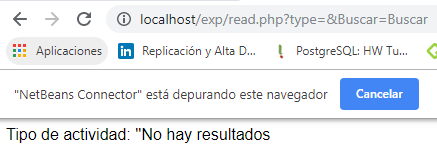
[CS04] Captura de pantalla con la visualización de todos los datos

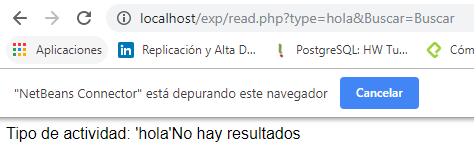


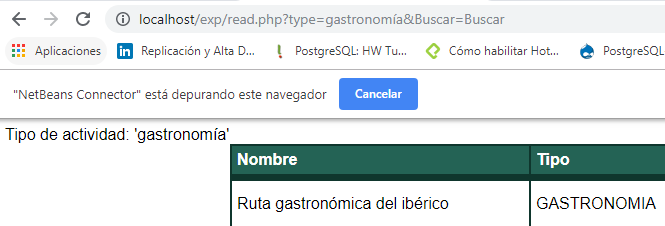
[CS05] Captura de pantalla con las pruebas del envío de datos por GET



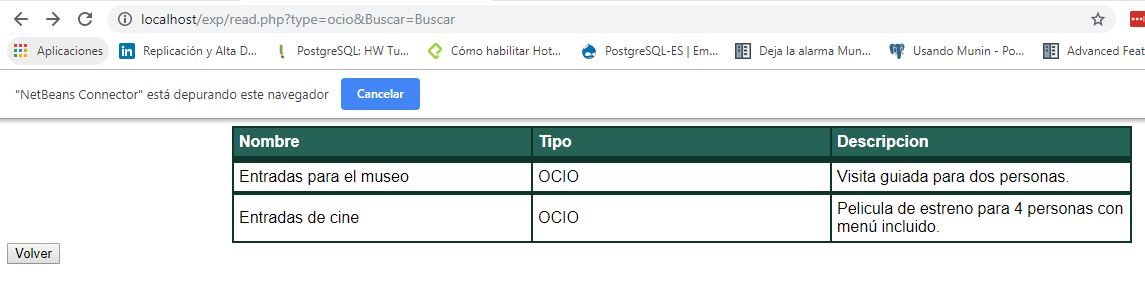




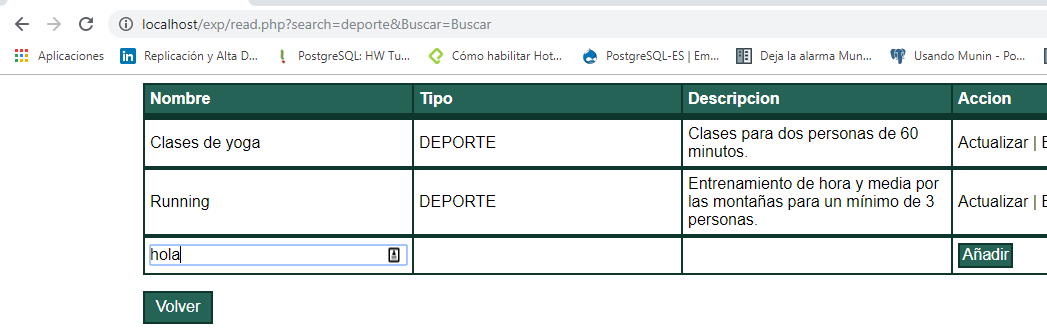


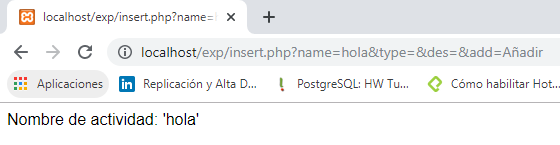


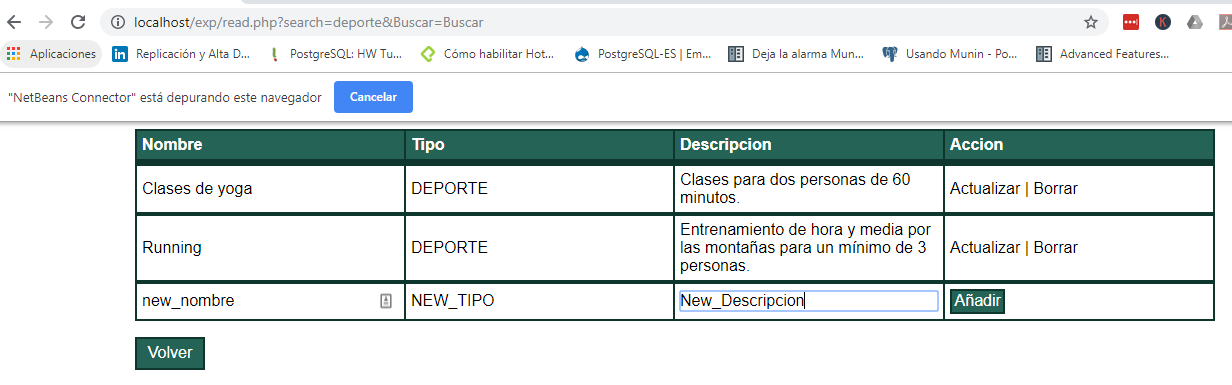
[CS06] Captura de pantalla con la visualización con los datos filtrados

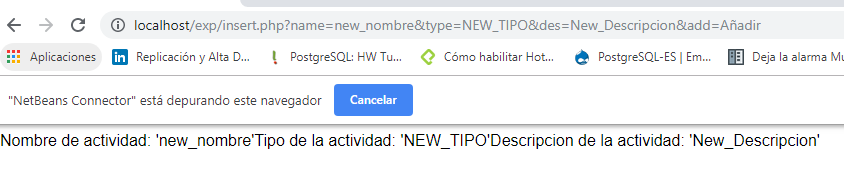


[CS07] Captura de pantalla con las pruebas de la modificación en “read.php”

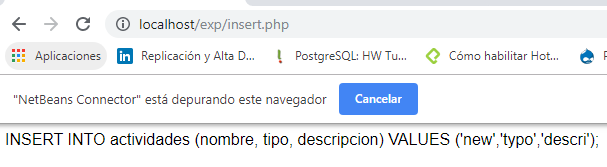


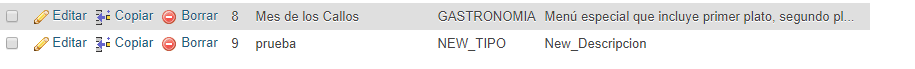




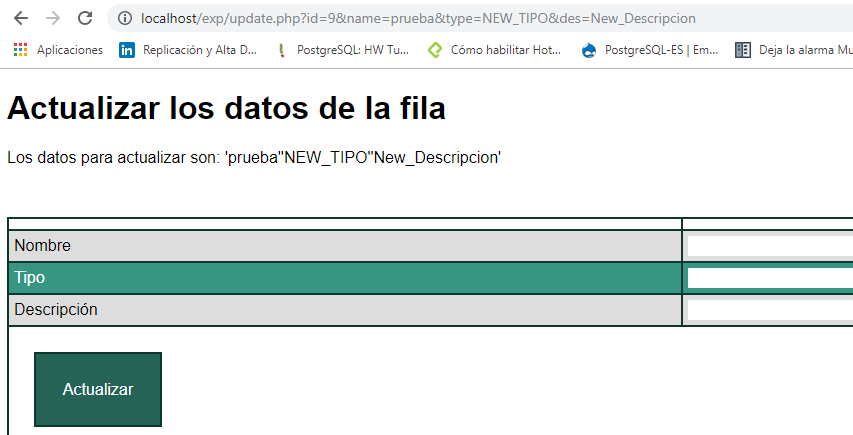


[CS08] Captura de pantalla con las pruebas de inserción

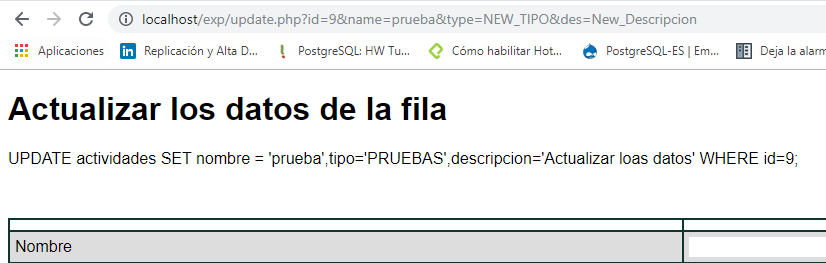




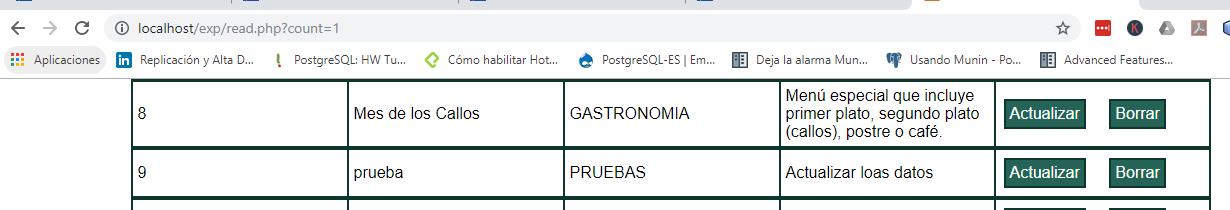
[CS09] Captura de pantalla con las pruebas de la modificación de actividades



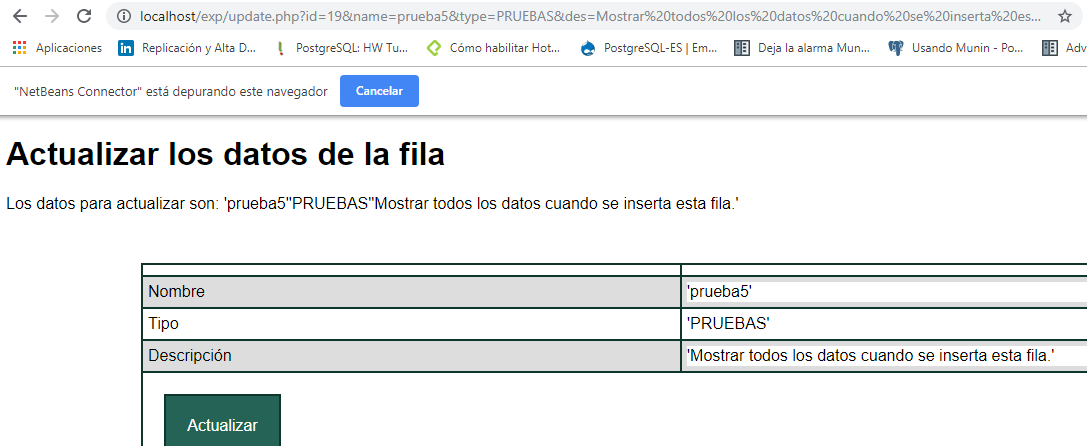




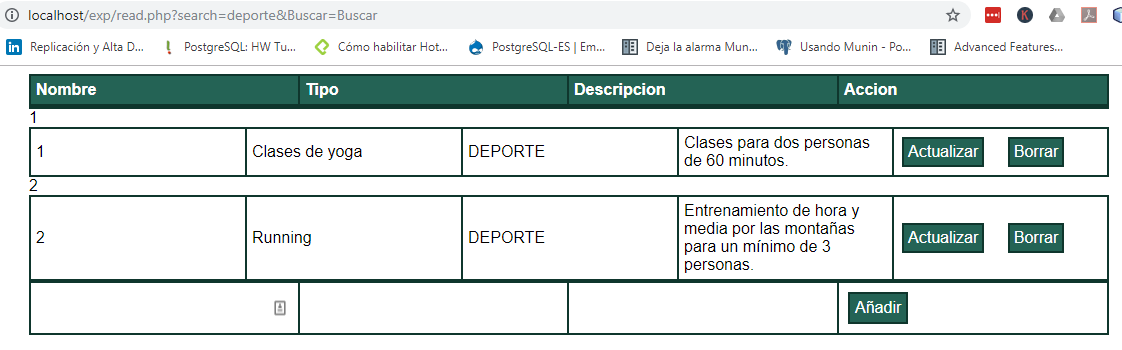


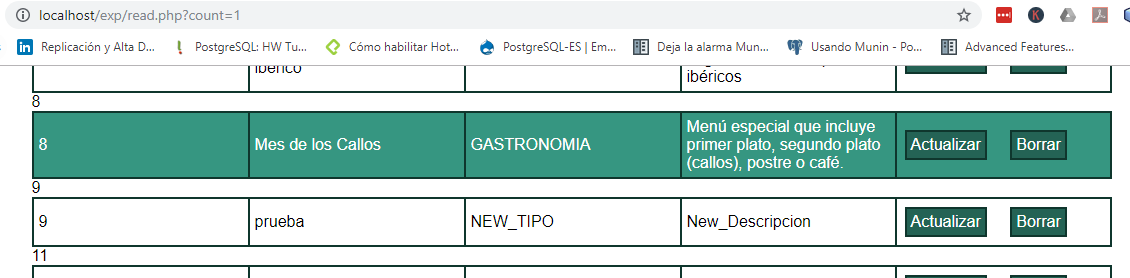


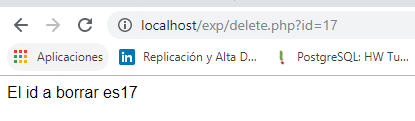
En las pruebas anteriores no rellenaba la tabla con los datos almacenados (sin actualizar). Ahora si lo rellena.

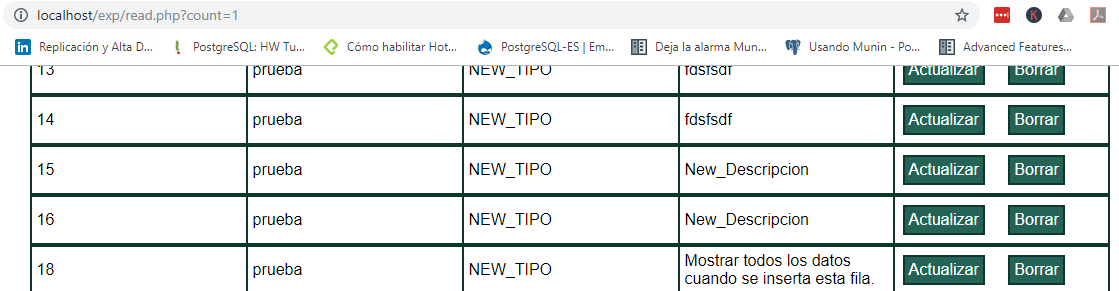


[CS10] Captura de pantalla con las pruebas del borrado de actividades











[CS11] Captura de pantalla con la tabla “usuarios”

[CS12] Captura de pantalla con las pruebas del cifrado de password

[CS13] Captura de pantalla con el formulario de login

[CS14] Captura de pantalla con las pruebas de autentificación de password

[CS15] Captura de pantalla con las pruebas control de permisos