



UNIVERSIDAD NACIONAL AUTÓNOMA  
DE MÉXICO



FACULTAD DE INGENIERÍA-UNAM

PROGRAMA DE FORMACIÓN DE BECARIOS

CURSO DE DESARROLLO WEB

PROYECTO FINAL DE DESARROLLO WEB: PÁGINA WEB  
DINÁMICA

INTEGRANTES:

SÁNCHEZ PÉREZ MARCO ANTONIO

VIGI GARDUÑO MARCO ALEJANDRO

UNICA

01 DE ABRIL DE 2022

## **Proyecto final de desarrollo web: Página web dinámica sobre cursos**

### **Objetivo**

Los integrantes del equipo pondrán en práctica los conocimientos adquiridos a lo largo del curso de desarrollo web, aplicando elementos dinámicos a vistas estáticas elaboradas en HTML y CSS para contar con una aplicación web interactiva.

### **Introducción**

En este documento, se colocará una breve documentación sobre el proyecto de desarrollo web. Se colocarán descripciones generales sobre los archivos que componen la página, la lógica empleada, implementación, así como descripciones sobre el qué y el cómo se estableció el proyecto. Este proyecto es una ampliación directa al proyecto del curso anterior, pues en aquel se elaboró un diseño web por medio de herramientas como HTML y CSS para tener únicamente las vistas. Y como continuación de ese proyecto anterior, aquí se elaboraron características como funcionalidades dinámicas en las vistas, con el fin de que los usuarios puedan ingresar e interactuar con la aplicación web. Esta parte del funcionamiento dinámico e interactivo se implementó en mayor medida con el lenguaje de programación web PHP, y un poco de Javascript para complementar.

En el proyecto se tuvo que hacer uso de una base de datos para contar con un manejo y desarrollo más realista de una página web, por lo que se diseñó una base de datos desde su modelo entidad - relación, pasando por un modelo relacional, hasta llegar a la implementación física con ayuda de SQL y el sistema administrador de bases de datos MySQL. Se detallará más adelante cómo es que se conectó la base de datos a la aplicación, así como el manejo que se le dieron a los datos dentro del proyecto. También se explicará cómo es que funciona y cómo es que se pensó la estructura de la página, indicando la descripción de las carpetas, módulos y archivos de esos módulos. Finalmente, se colocarán las conclusiones generales.

### **Desarrollo**

En esta sección del documento, se colocará la descripción de los módulos, carpetas, archivos, y en general, el cómo se ideó el proyecto junto con su estructura.

Primero se optó por mejorar el proyecto del curso pasado, con respecto a algunas vistas e incluso algunos detalles estéticos de la página. Por lo que se decidió mejorar el formulario de comentarios, adecuar las fuentes, homogeneizar el proyecto en general y organizarlo. Esto último fue importante porque se decidió organizar el código en módulos o carpetas bien definidas. Un módulo es el que contiene los archivos de las vistas, es decir, los archivos HTML y CSS encargados de definir tanto la estructura como la estética de la página. Este módulo se llamó **página**.

Se realizaron las debidas correcciones sobre los aspectos a mejorar durante la primera presentación del diseño web, en donde actualizamos el uso de una sola fuente principal, el uso de los colores se mejoró teniendo una vista más homogénea haciendo que el aspecto sea mejor entre las páginas; en el header, se corrigió el margen el cual no permitia usar todo el espacio en cuanto a lo ancho y no se apreciaba de forma centrada; el footer fue actualizado teniendo de igual forma una mejor consistencia de colores añadiendo los enlaces en cada imagen y se reparó el uso de media en la página del formulario.

Otro módulo muy importante fue el encargado de realizar la conexión con la base de datos. La carpeta de esta sección se llamó **Configuración**, y fue aquí en donde se planteó la sintaxis de la conexión a la base de datos en MySQL ingresando los parámetros necesarios como el host, usuario, la contraseña y la base de datos a emplear. Esto se realizó por medio de un archivo PHP que utiliza un método de conexión llamado `mysqli_connect`. Es preciso mencionar que la información de la conexión se almacena en un dato con el nombre '\$db' en PHP, lo cual nos facilita el manejo de las conexiones a la base de datos.

```
//definimos constantes para hacer la conexion con la base de datos
define('DB_SERVER', 'localhost');
define('DB_USERNAME', 'root');
define('DB_PASSWORD', '');
define('DB_DATABASE', 'proyectocursos');

//se abre una nueva conexion con el servidor de MySQL
$db = mysqli_connect(DB_SERVER,DB_USERNAME,DB_PASSWORD,DB_DATABASE);
>>
```

Después, se definió un módulo llamado **ObtenerInfoBD**, en donde se definió principalmente la lógica encargada del manejo de sesiones. Además, en esta carpeta se colocaron diversos archivos para recuperar información de las tablas de la base de datos con el fin de realizar pruebas. Esta parte fue muy importante porque nos permitió comprender cómo interactúan los archivos, datos e información dentro del lenguaje de programación PHP. Para las sesiones, se creó un archivo PHP llamado 'sesion.php', el cual utiliza el archivo de configuración para conectarse a la base de datos y después crear una sesión de la página. Para poder validar que se está en una sesión, se hace uso de los super arreglos globales de PHP, en este caso es el arreglo global `$_SESSION`. Este arreglo almacena el usuario que está activo en la página, después se hace una consulta a la base de datos para corroborar que el usuario vigente en la sesión exista en la base, y una vez que se ha comprobado que es el mismo, entonces

no se hace nada para continuar navegando. En el caso de que no se corrobore el usuario, se redirige a un archivo de login en PHP y se termina la sesión actual.

```
<?php
//se importa el archivo de configuracion de la base de datos
include '../Configuracion/config.php';
session_start();

$usuarioVerificado = $_SESSION['usuario_loggeado'];

$ses_sql = mysqli_query($db,"select nombrePila from alumno where nombrePila = '$usuarioVerificado' ");

$row = mysqli_fetch_array($ses_sql,MYSQLI_ASSOC);

$sesion_loggeada = $row['nombrePila'];

if(!isset($_SESSION['usuario_loggeado'])){
    header("location: ../pagina/acceso/acceso.php");
    die();
}


?>
```

Dentro del mismo módulo de **ObtenerInfoBD**, se tiene un archivo similar a 'sesion.php', pero llamado 'sesionSecundario.php', el cual se encarga de ligar a los archivos que se encuentran más ocultos en la estructura del proyecto con la conexión a la base de datos y a la sesión vigente de la página. Finalmente, dentro de esta carpeta, se tiene un archivo PHP encargado de terminar con una sesión vigente y regresar a la página de inicio de sesión por medio de un header. Este mecanismo nos permite hacer que se termine una sesión de algún usuario y permitirle el acceso a otro usuario.

Otro aspecto importante fue el que se implementó en el archivo 'acceso.php' en el módulo de **pagina**. Este archivo es el que implementa el formulario de acceso hacia la página web y a la vez es el que autoriza, por medio de consultas a la base de datos, si un usuario puede acceder a la página. Este archivo requirió de la combinación de HTML y PHP para poder comunicar la entrada de datos del usuario con la lectura a la base de datos. Primero se obtienen los datos ingresados por el usuario, después se hace la consulta a la base de datos y finalmente se comparan las coincidencias entre los datos para poder darle paso al usuario. Una vez que se han comprobado las credenciales, se inserta el usuario loggeado en el arreglo global de usuarios para crear una sesión, y se redirige a la página de inicio en donde se pueden visualizar los cursos.


```
$sql = "SELECT * FROM alumno WHERE nombrePila = '$myusername' and apellidoPaterno = '$mypassword'";
$result = mysqli_query($db,$sql);
$row = mysqli_fetch_array($result,MYSQLI_ASSOC);

$count = mysqli_num_rows($result);
```



[Inicio](#)[Acceso](#)[Acercas de](#)[Formulario](#)[Libre](#)

Acceso



Usuario

550087412

Contraseña

XXXXXXXXXX

Iniciar Sesión

A grandes rasgos, en todos los demás archivos correspondientes a las vistas, se usaron instrucciones PHP en la parte superior para poder validar una sesión activa con algún usuario, De esta manera, se garantiza un nulo acceso a la página si aún no se ha iniciado sesión.

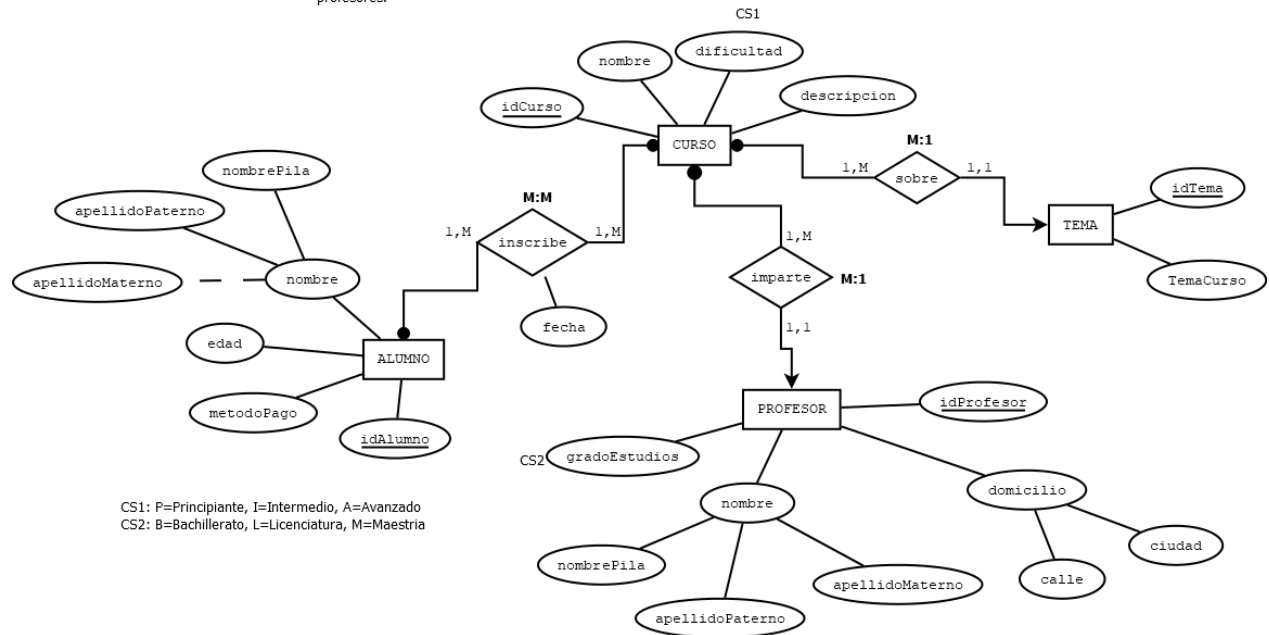
```
<?php
|   include '../ObtenerInfoBD/sesion.php';
?>
```

Gracias a la conexión que se realizó con la base de datos con la página del formulario, fue posible poder guardar los datos que el usuario introducía, tal como lo fué su correo, nombres, apellidos y el comentario con un tamaño máximo de hasta 250 caracteres. Gracias a PHP, es posible almacenar en la base de datos de una manera eficiente, verificando que el correo no contiene espacios, los nombres y apellidos se inicien con mayúsculas y en el comentario la primera letra sea mayúscula. (Uso de Trim, ucwords y ucfirst) Dichas cuatro entradas son necesarias así como también la función de dar click al botón “enviar” para poder almacenarlo en la base de datos, de forma contraria, no se podrá enviar de una manera satisfactoria.

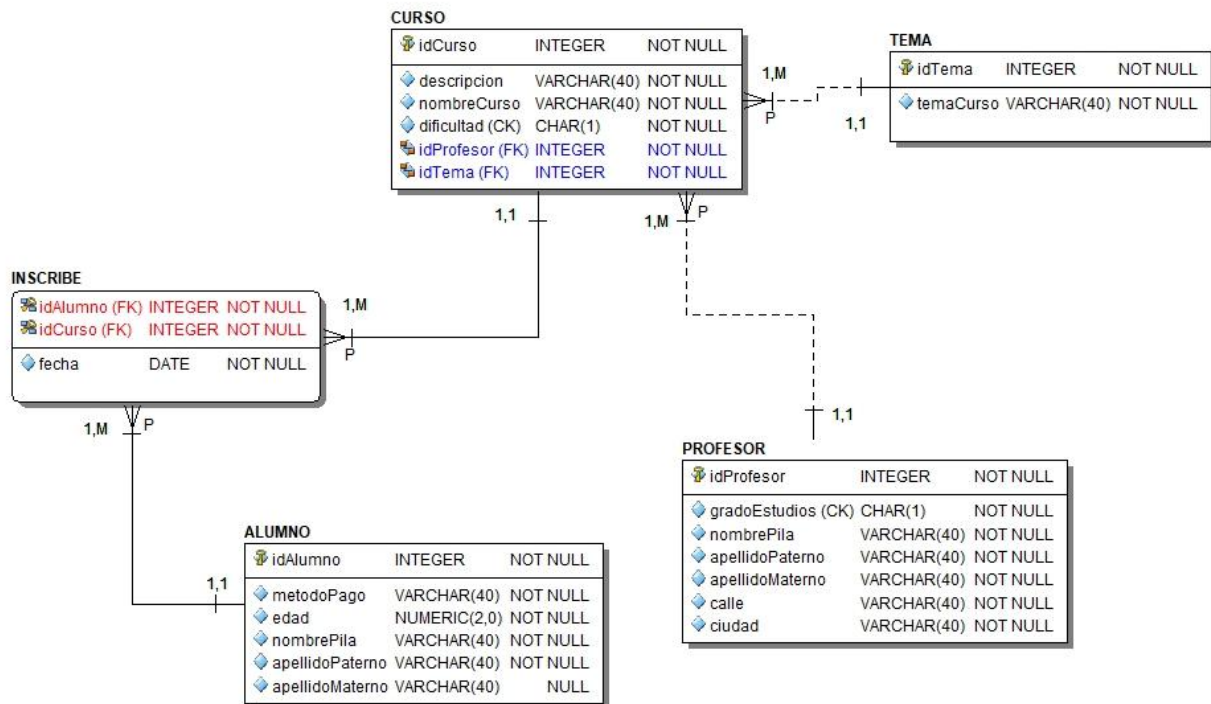
## Base de datos

Un aspecto vital en el proyecto fue el desarrollo, la implementación y utilización de la base de datos. Para su desarrollo, se pensó en definir las entidades más importantes como alumnos, cursos, y profesores que tienen que ver con la temática del proyecto. Después se realizó la elaboración del modelo entidad-relación, el cual define visualmente la estructura de la base de datos a un alto nivel. En el MER, se establecieron las entidades, relaciones entre entidades, cardinalidades, y restricciones en forma de consideraciones semánticas. A continuación, se muestra el diagrama obtenido en esta fase de la elaboración de la base de datos.

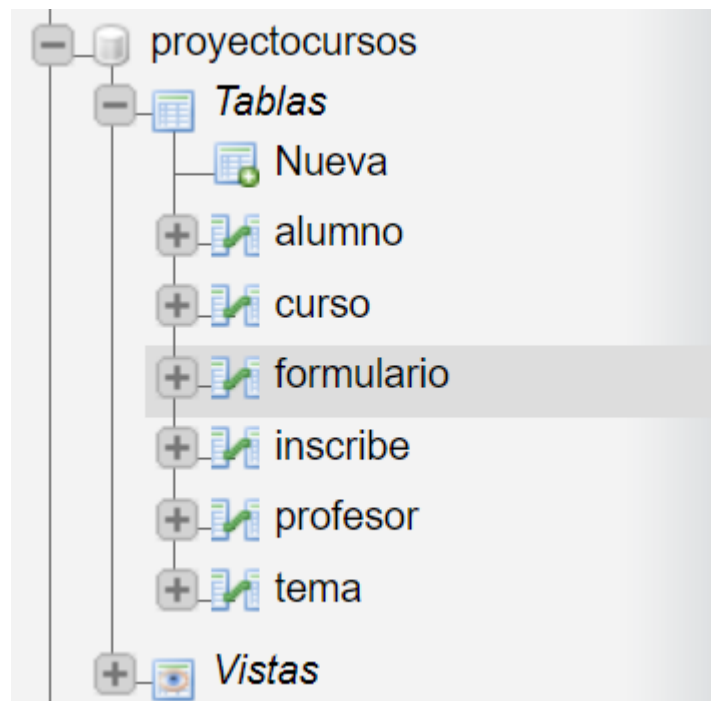
Una micro-empresa quiere crear una base de datos para almacenar a sus usuarios, profesores y cursos para su página web. De los cursos se tienen el identificador, el nombre, la dificultad (solo un caracter entre basico, intermedio y avanzado), tema. De los alumnos se requiere saber el nombre, edad, metodo de pago, y cursos asociados.. Para los profesores, se requiere saber su grado de estudios, su nombre, domicilio y cursos que imparte. Un alumno puede estar inscrito en uno o varios cursos. Un curso está disponible para uno o muchos alumnos. Un profesor imparte un solo curso, y los cursos los imparten uno o muchos profesores.



Después se realizó el modelo relacional, el cual tuvo un nivel de detalle más cercano a una base de datos, pues fue aquí en donde se definieron los tipos de dato, relaciones débiles, relaciones fuertes, llaves primarias, llaves foráneas y restricciones.



Incluso, se procedió con la implementación de la base de datos en un manejador, para esto, se optó por trabajar con MySQL por su gran compatibilidad con el lenguaje de programación PHP. Por medio del modelo relacional se establecieron y definieron las sentencias SQL para definir la estructura de la base, específicamente con el DDL. Y después de haber definido la base, se plantearon las sentencias DML para la inserción de registros a la base de datos, los cuales se emplearon en el proyecto web como se explicó anteriormente.



Finalmente, se colocó una interfaz de eliminación de usuarios dentro del módulo **pagina**. Decidimos colocar en esta sección la descripción de la interfaz web porque tuvo una gran relación con la base de datos. Básicamente en esta interfaz, se tienen consultas a la base en MySQL, se trae la información sobre los usuarios alumnos a la página. Se imprime en pantalla la información de dichos alumnos en forma de tabla, y se tiene una interacción por medio de botones para poder eliminar de la base de datos a los usuarios alumnos. Esta interfaz solo tuvo la intención de trabajar con la información de la base de datos, por medio de consultas, y después editar esa información para hacer eliminaciones en la base de datos. Por esta razón, se tuvo que hacer un query en donde se mandara una eliminación a la base, y los registros en dicha base se quitan para mostrar en pantalla rápidamente cómo es que se actualiza la información. Como en las vistas anteriores, aquí se colocó la restricción de que únicamente se puedan hacer eliminaciones de usuarios, una vez que un usuario haya iniciado sesión, esto para evitar que cualquiera pueda eliminar o modificar la información de la base de datos.