



Universidad de Margarita  
Alma Mater del Caribe  
Vicerrectorado Académico  
Decanato de Ingeniería de Sistemas  
Asignatura: Programación IV

## **GUIA DE ESTILO DE CÓDIGO: MUSE**

**DOCENTE:**

Ing. Cesar Requena

**ALUMNO:**

Santiago Fermín

Agustin Gil

Gilberto Jiménez

El Valle del Espíritu Santo, noviembre de 2023

## CARPETAS MAS UTILIZADAS

- **js:** Incluye todos los archivos principales de Javascript para el proyecto
- **css:** Contiene todos los archivos de estilo propios de los proyectos. Cada pagina tiene un archivo css que le corresponde, con las siguientes excepciones:
  - login.css es usado tanto para el estilo del login como el registro
  - style.css es un archivo general referenciado en todos los archivos php, contiene variables y otras configuraciones generales del estilo
- **lib:** Esta carpeta existe tanto en la carpeta js como css, dentro de las carpetas lib se encuentran los archivos propios de librerías externas utilizadas dentro del proyecto
- **includes:** Contiene archivos que son utilizados de manera repetida a lo largo de múltiples otros archivos por medio de *include* de php. Como por ejemplo el header y el footer que se repiten tanto en el index como en el inicio de sesión.
- **assets:** Contiene todo lo referente a recursos, principalmente multimedia que son necesarios para la pagina. Esta se encuentra dividida en carpetas que se detallaran mas adelante
- **audio:** Se encuentra toda la música que el sitio web utilizara
- **fonts:** Las fuentes guardadas localmente para la pagina
- **img:** Contiene todos los activos visuales.
- **Img/icons:** En esta se encuentran todos los iconos de la pagina en formato .svg principalmente
- **config:** Contiene los archivos php que sirven para realizar las principales conexiones y validaciones dentro del sitio web
- **email:** Contiene los archivos de phpmailer necesarios para enviar los correos

## Comentarios

- En la medida de lo posible el uso de comentarios debe reducirse al mínimo posible, los nombres de las variables y las funciones deben ser descriptivas de su objetivo
- En caso de utilizar comentarios, en PHP hacer uso de // para los comentarios de línea, o /\* \*/ para los comentarios de bloque. Evitar el uso de múltiples // para simular un comentario de bloque.

```
// Inicializa la sesion
session_start();

// Vacía todas las variables de la sesion
session_unset();

// Destruye la sesion
session_destroy();
```

## Convenciones para clases HTML-CSS:

- Es RECOMENDADO utilizar el sistema de nombres BEM para clases:
  - Los nombres se dividen en bloques (elementos unitarios que sirven por si mismo), elementos (partes de un bloque que no tienen significado por si mismo) y modificadores (estados alternativos para elementos o bloques)
  - Los nombres de bloques u elementos no deben hacer referencia al estilo del mismo, deben describir su utilidad o presencia en la estructura
  - Es recomendado que los nombres de modificadores no incluyan detalles de estilo, en lugar de eso deben describir un estado. Por ejemplo: activo, desplazado, movido
  - La sintaxis del sistema BEM es la siguiente:
    - bloque\_\_elemento—modificador
- Los estilos de CSS se DEBE separar en múltiples archivos en función de su conveniencia para su reutilización. Por lo general creando archivos separados para estilos específicos de partes de la pagina particulares
- NO DEBE utilizarse los nombres de etiquetas para seleccionarse, incluso en casos de etiquetas únicas en la pagina, como *<header>* o *<footer>*, se debe asignar una clase o id para referenciar al mismo

## Conjuntos de reglas para el formato de código PHP

Para este proyecto se utilizará el estándar PSR-12, el cual en si mismo incluye el PSR-1. El PSR-12 se trata de la guía de estilo de código extendido, las cuales fue un remplazo del PSR-2 mas moderno y adaptado su contexto actual. A continuación, se resaltan algunas

- Las funciones en PHP deberán estar en camelCase tal como lo establece el PSR-1. Así mismo las variables también estarán en este formato

```
function conexion()  
{  
    $nombreServidor = "localhost";  
    $baseDatos = "bdmuse";  
    $nombreUsuario = "root";  
    $contraseña = '';  
  
    $pdo = new PDO('mysql:host=' . $nombreServidor . ';dbname=' . $baseDatos,  
$nombreUsuario, $contraseña);  
    return $pdo;  
}
```

- Las constantes DEBEN ser declarados en mayúsculas, utilizando \_ como separadores
- Aquellos archivos que contengan únicamente código PHP DEBEN de omitir la etiqueta de cerrado ?> al final de los mismos
- La indentacion DEBERÍA ser de 4 espacios por cada nivel