



**UNIVERSIDAD AUTÓNOMA
DE CHIHUAHUA**
Facultad de Ingeniería

Fundamentos de Bases de Datos

4.10 Proyecto Final

Johann David Gómez Villalobos 352909

Leonardo Ochoa Pérez 361870

Guillermo Cruz Juárez 352905

Profesor: José Saúl de Lira Miramontes

FECHA DE ELABORACIÓN: 27/05/2024

I. Manual Técnico

1. Sistema/Aplicación

a. Nombre del sistema

SQL gym

b. Descripción y delimitación del sistema

La página web diseñada para el gimnasio está estructurada para proporcionar una experiencia integral y funcional a sus visitantes y usuarios. Esta plataforma digital incluye secciones específicas donde se pueden explorar las diferentes clases disponibles, conocer a los entrenadores, y visualizar una galería con imágenes de las instalaciones. Además, cuenta con un mapa interactivo para facilitar la localización de los gimnasios.

El sitio también integra un blog donde se publicarán artículos relevantes sobre salud y entrenamiento, ofreciendo contenido fresco y de valor añadido para los usuarios. Para facilitar una gestión eficiente y personalizada, el sistema permite la creación de cuentas de usuario con distintos niveles de permisos, adaptándose así a diversas necesidades y roles, como miembros del gimnasio, entrenadores y administradores.

En el núcleo técnico del sitio web, se implementa una conexión con una base de datos SQL. Esta base es esencial para almacenar de manera segura y estructurada todos los registros relacionados con el funcionamiento del sitio, incluyendo datos de usuarios, detalles de las páginas, entradas del blog, información de las clases y perfiles de los entrenadores, entre otros elementos.

c. Objetivo general

Desarrollar y mantener una página web interactiva y segura para el gimnasio, que facilite la interacción efectiva entre el gimnasio y sus usuarios, proporcionando acceso fácil y rápido a información sobre clases, entrenadores, instalaciones y servicios adicionales, a la vez que gestiona de manera eficiente los datos del usuario y del sitio.

d. Objetivos específicos

1. **Implementar una interfaz de usuario intuitiva y accesible** que permita a los visitantes y usuarios navegar fácilmente por las diferentes secciones de la página, como clases, entrenadores, y galería de imágenes.
2. **Desarrollar un sistema de gestión de usuarios** que soporte diferentes niveles de permisos, permitiendo a los administradores, entrenadores y usuarios acceder a funciones específicas acordes a sus roles.
3. **Integrar un blog dentro de la página** para publicar contenido actualizado y relevante que motive a los usuarios a visitar regularmente el sitio web y mantenerse informados sobre temas de salud y entrenamiento.
4. **Crear una conexión segura con una base de datos SQL** para almacenar y gestionar toda la información del sitio web, incluyendo registros de usuarios, páginas, entradas de blog, clases y perfiles de entrenadores, asegurando la integridad y la seguridad de los datos.
5. **Implementar herramientas de localización en el sitio web** que permitan a los usuarios encontrar fácilmente la ubicación de los gimnasios, junto con detalles como horarios de operación y disponibilidad de clases.
6. **Asegurar la compatibilidad de la página web** con diversos dispositivos y navegadores, garantizando una experiencia de usuario coherente y accesible desde smartphones, tabletas y computadoras de escritorio.

e. Descripción de tipos de usuarios

1. Visitantes

- **Descripción:** Los visitantes son individuos que acceden a la página web sin registrarse o iniciar sesión. Tienen el mismo acceso a la información general del sitio, pero no pueden hacer comentarios en el blog.

2. Miembros Registrados

- **Descripción:** Son usuarios que se han registrado en la página web proporcionando información personal básica. Estos usuarios son generalmente miembros activos del gimnasio y pueden hacer comentarios en el blog.

3. Administradores

- **Descripción:** Los administradores son responsables de la gestión y el mantenimiento de la página web. Tienen acceso al backend de la página y pueden agregar, editar o eliminar usuarios, entrenadores, clases, etc. .

f. Entorno operativo del sistema

Front-end

Html, CSS y PHP

Back-end

PHP y Javascript

Base de Datos

MySQL

Sistema de gestión de contenido y Framework

WordPress

2. Especificación de requerimientos

a. Requerimientos funcionales

RF1: El sistema debe permitir el registro de nuevos usuarios proporcionando datos básicos como nombre, correo electrónico, y contraseña.

RF2: El sistema debe permitir a los usuarios iniciar y cerrar sesión utilizando sus credenciales.

RF3: El sistema debe permitir a los administradores y entrenadores crear, modificar y eliminar clases programadas.

RF4: El sistema debe mostrar el horario de las clases, incluyendo día, hora y entrenador asignado.

RF5: El sistema debe proporcionar una sección donde se muestre información detallada de los entrenadores, incluyendo su biografía, especialidades y horarios de clases.

RF6: El sistema debe permitir a los administradores añadir, modificar y eliminar perfiles de entrenadores.

RF7: El sistema debe permitir a los administradores publicar, editar y eliminar entradas en el blog.

RF8: El sistema debe permitir a los usuarios comentar en las entradas del blog, si están registrados y han iniciado sesión.

RF9: El sistema debe mostrar una galería de imágenes de las instalaciones del gimnasio que los administradores puedan actualizar.

RF10: El sistema debe proporcionar un mapa interactivo que muestre la ubicación de los gimnasios.

RF11: El sistema debe incluir un formulario de contacto para que los usuarios puedan enviar mensajes al gimnasio.

RF12: El sistema debe implementar medidas de seguridad para proteger la información personal de los usuarios.

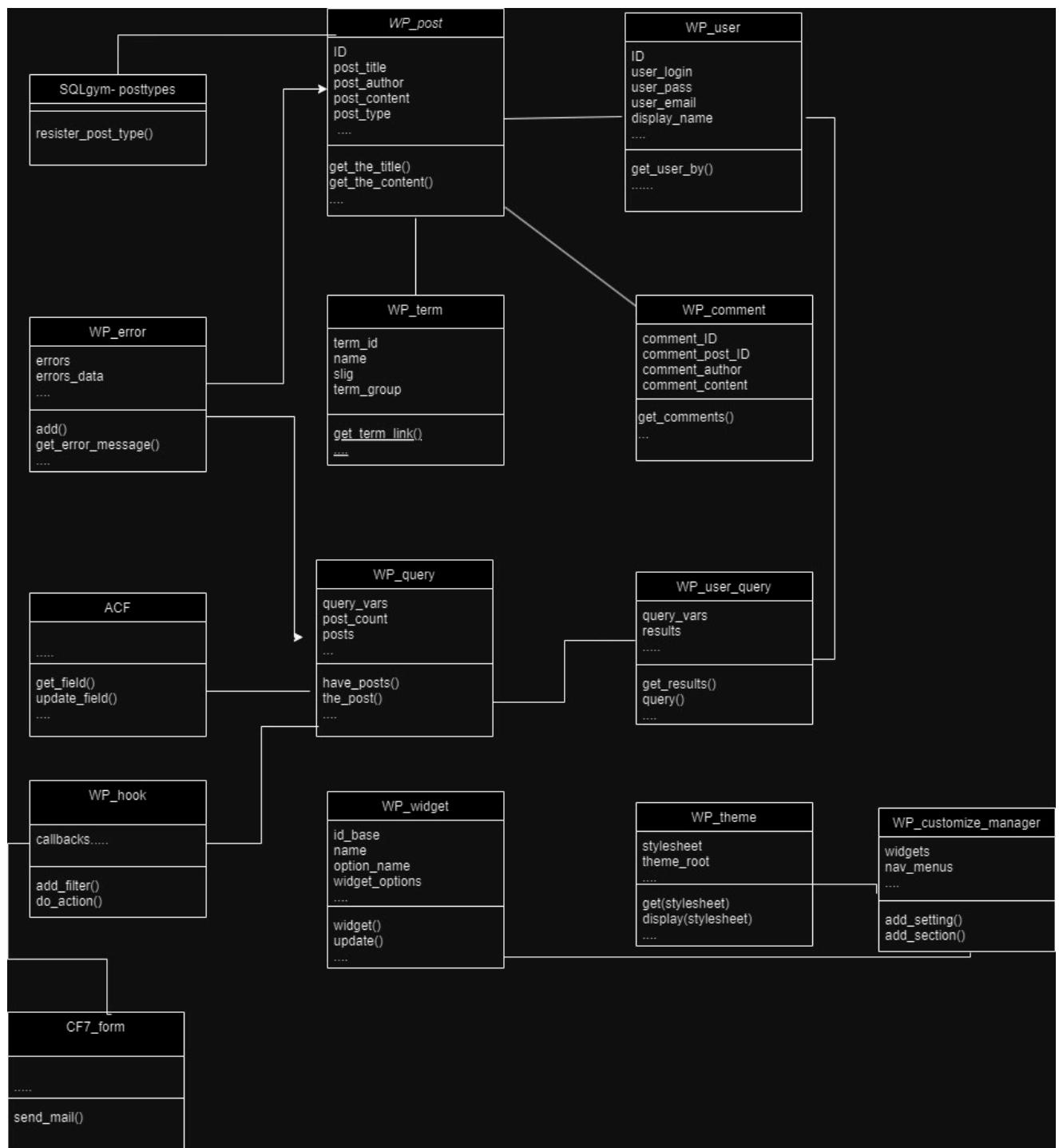
RF13: El sistema debe cifrar las contraseñas de los usuarios en la base de datos.

RF14: El sistema debe ser responsive y compatible con diversos dispositivos y navegadores.

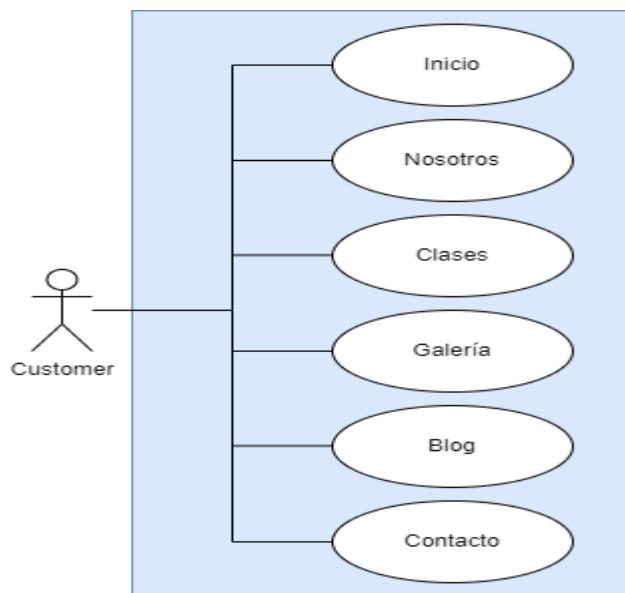
RF15: El sistema debe proporcionar una interfaz intuitiva y fácil de usar para todos los tipos de usuarios.

b. Modelado del sistema

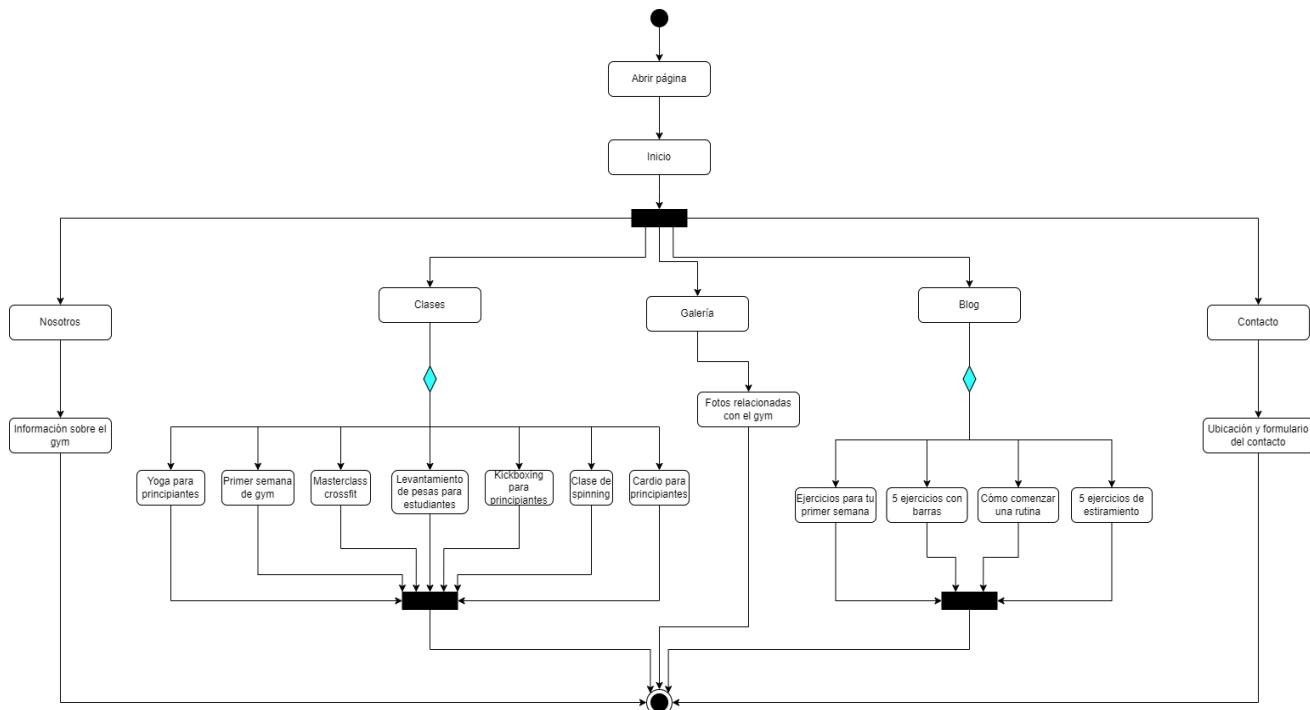
i. Diagrama de clases



ii. Diagrama de casos de uso



iii. Diagrama de actividades



c. Requerimientos no funcionales

RNF1: El sistema debe ser capaz de manejar al menos 100 usuarios concurrentes sin degradación significativa del rendimiento.

RNF2: Todas las páginas web deben cargar en menos de 3 segundos bajo condiciones normales de uso.

RNF3: La página web debe estar disponible el 99.9% del tiempo, excluyendo los períodos programados de mantenimiento.

RNF4: Debe implementarse un sistema de autenticación y autorización robusto para garantizar que solo los usuarios autorizados puedan acceder a funciones específicas.

RNF5: El código del sistema debe ser modular, bien documentado y fácil de mantener.

RNF6: Deben realizarse actualizaciones regulares del software para mejorar la funcionalidad y corregir errores sin afectar la disponibilidad del sistema.

RNF7: La interfaz de usuario debe ser intuitiva y accesible para personas con habilidades tecnológicas limitadas.

RNF8: La página web debe ser compatible con los navegadores más utilizados, incluyendo Google Chrome, Mozilla Firefox, Safari y Microsoft Edge, en sus versiones actuales y las dos anteriores.

RNF9: La página web debe ser compatible y funcional en dispositivos móviles, tabletas y ordenadores de escritorio.

3. Arquitectura del Sistema/Aplicación.

a. Layer/Tier's

1. Presentación (Front-End)

- Este nivel es el que interactúa directamente con los usuarios finales. Incluye toda la interfaz gráfica de usuario (GUI) que se ejecuta en el navegador del cliente. Utiliza tecnologías como HTML, CSS y PHP, e incorpora el framework de WordPress para crear interfaces dinámicas y responsivas.

2. Lógica de Negocio (Back-End)

- Esta capa procesa las solicitudes enviadas desde el front-end, ejecutando la lógica de negocio necesaria para las funcionalidades del sitio web como la gestión de usuarios, reservas de clases, etc. Se implementa con PHP.

3. Datos (Base de Datos)

- En este nivel se almacenan y gestionan todos los datos relacionados con el sitio web. Incluye la base de datos MySQL. Este nivel es responsable de garantizar la integridad, seguridad y disponibilidad de los datos.

b. Frontback/Backend

Frontend

- Responsable de la experiencia del usuario, incluyendo la visualización de la información, la interactividad y la entrada de datos por parte de los usuarios. Aquí se utiliza Html, CSS y PHP así como WordPress como framework.

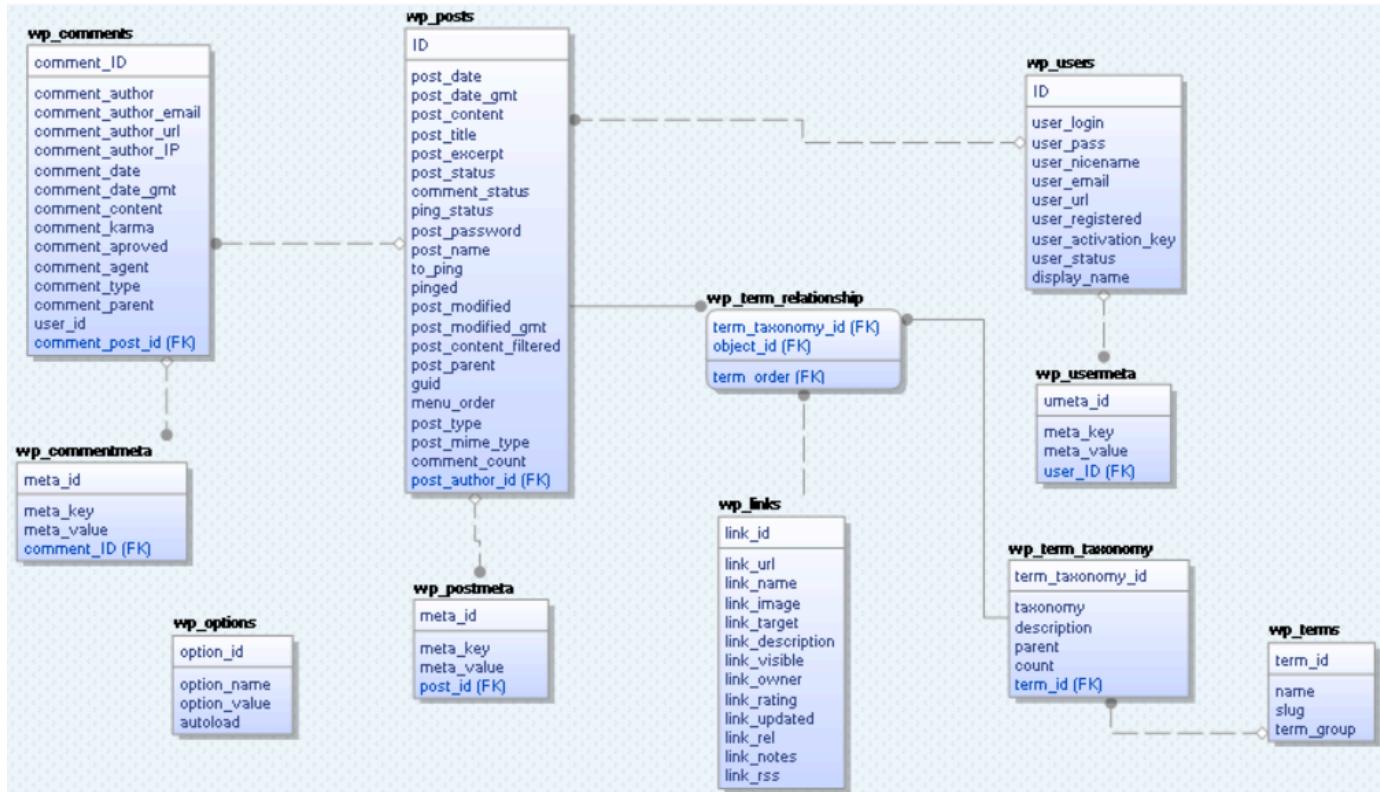
Backend

- Gestiona la lógica de negocio central, el procesamiento de datos y las operaciones de bases de datos. Es el núcleo que procesa las solicitudes del frontend, ejecuta la lógica necesaria y envía los resultados de vuelta al frontend. También maneja la seguridad, como la autenticación de usuarios y la autorización de accesos. Aquí se utiliza PHP y JS así como WordPress como framework..

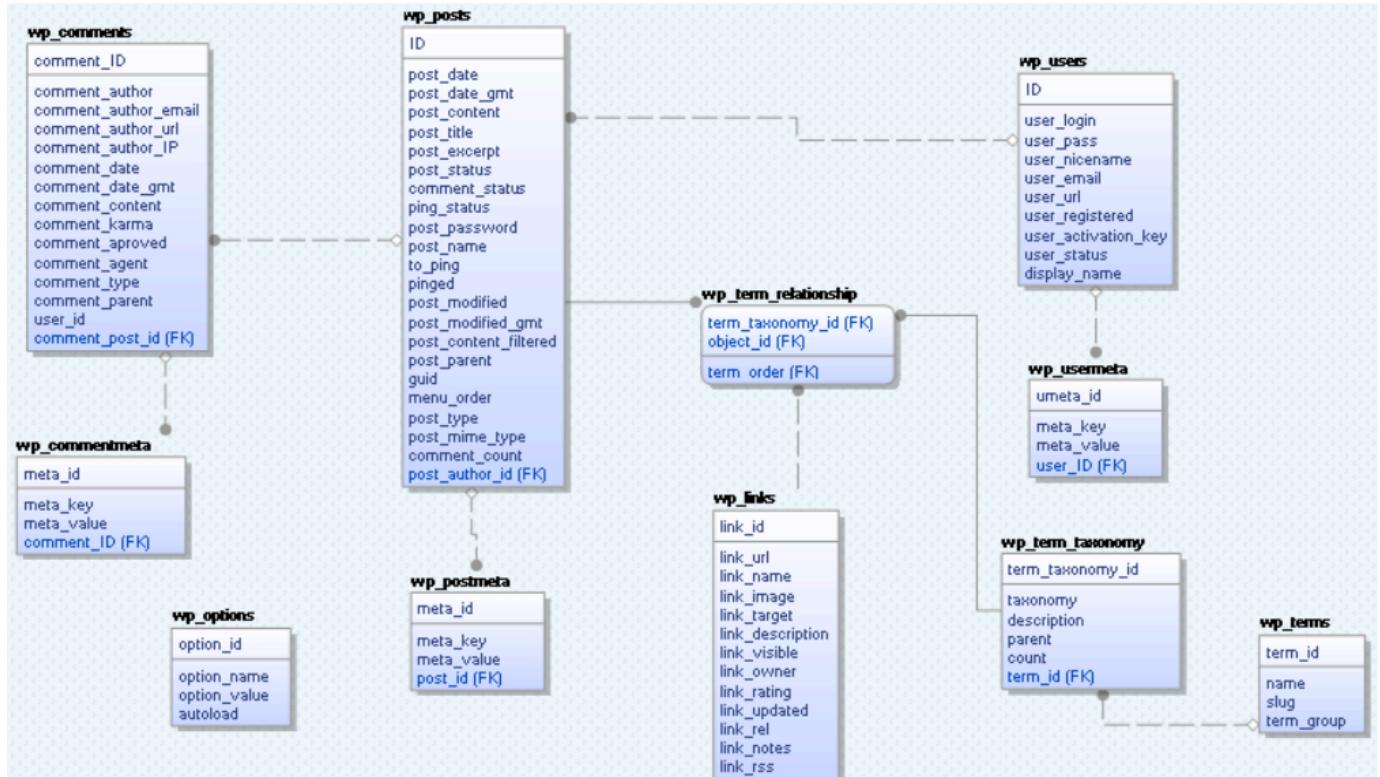
c. Estructura modular del sistema

El sistema se desarrolla en módulos separados para cada función principal, lo que facilita el mantenimiento y la escalabilidad:

1. **Módulo de Autenticación**
 - Maneja el registro de usuarios, inicio de sesión, recuperación de contraseña y gestión de sesiones.
 2. **Módulo de Gestión de Usuarios**
 - Encargado de crear, modificar, y eliminar cuentas de usuario, así como de administrar roles y permisos.
 3. **Módulo de Clases y Reservas**
 - Permite a los usuarios y administradores gestionar las clases, incluyendo horarios, inscripciones y cancelaciones.
 4. **Módulo de Entrenadores**
 - Administra los perfiles de los entrenadores, sus horarios y las clases que ofrecen.
 5. **Módulo de Blog y Contenido**
 - Permite la publicación, edición y eliminación de entradas en el blog, así como la gestión de comentarios y contenido multimedia.
-
4. Base de datos
 - a. Diagrama Conceptual (Entidad/Relación)



b. Esquema Lógico de la Base de Datos



i. Especificación Tablas (Normalizadas hasta 3NF)

1. Tabla `wp_posts`

- **Descripción:** Almacena todos los tipos de contenido como publicaciones, páginas y tipos de contenido personalizados.
- **Columnas:**
 - `ID` (BIGINT, PRIMARY KEY, AUTO_INCREMENT): Identificador único del post.
 - `post_author` (BIGINT, FOREIGN KEY `post_author` REFERENCES `wp_users(ID)`): ID del autor del post.
 - `post_date` (DATETIME): Fecha y hora de publicación del post.
 - `post_date_gmt` (DATETIME): Fecha y hora de publicación en GMT.
 - `post_content` (LONGTEXT): Contenido del post.
 - `post_title` (TEXT): Título del post.
 - `post_excerpt` (TEXT): Extracto del post.
 - `post_status` (VARCHAR(20)): Estado del post (publicado, borrador, etc.).
 - `comment_status` (VARCHAR(20)): Estado de los comentarios (abiertos, cerrados).
 - `ping_status` (VARCHAR(20)): Estado de los pings.
 - `post_password` (VARCHAR(255)): Contraseña del post, si es protegido.
 - `post_name` (VARCHAR(200)): Slug del post.
 - `to_ping` (TEXT): Lista de URLs a las que se enviarán pings.
 - `pinged` (TEXT): Lista de URLs que han sido pingueadas.
 - `post_modified` (DATETIME): Fecha y hora de la última modificación.
 - `post_modified_gmt` (DATETIME): Fecha y hora de la última modificación en GMT.
 - `post_content_filtered` (LONGTEXT): Contenido filtrado.
 - `post_parent` (BIGINT): ID del post principal (si es un post hijo).
 - `guid` (VARCHAR(255)): Enlace permanente al post.
 - `menu_order` (INT): Orden del menú.
 - `post_type` (VARCHAR(20)): Tipo de contenido (post, page, etc.).
 - `post_mime_type` (VARCHAR(100)): Tipo MIME del archivo adjunto.
 - `comment_count` (BIGINT): Número de comentarios en el post.

2. Tabla `wp_users`

- **Descripción:** Almacena información sobre los usuarios registrados.
- **Columnas:**
 - `ID` (BIGINT, PRIMARY KEY, AUTO_INCREMENT): Identificador único del usuario.
 - `user_login` (VARCHAR(60), UNIQUE): Nombre de usuario.
 - `user_pass` (VARCHAR(255)): Contraseña del usuario.
 - `user_nicename` (VARCHAR(50)): Nombre bonito del usuario.
 - `user_email` (VARCHAR(100), UNIQUE): Email del usuario.
 - `user_url` (VARCHAR(100)): URL del usuario.
 - `user_registered` (DATETIME): Fecha y hora de registro.
 - `user_activation_key` (VARCHAR(255)): Clave de activación.
 - `user_status` (INT): Estado del usuario.
 - `display_name` (VARCHAR(250)): Nombre mostrado del usuario.

3. Tabla `wp_postmeta`

- **Descripción:** Almacena metadatos de los posts.
- **Columnas:**
 - `meta_id` (BIGINT, PRIMARY KEY, AUTO_INCREMENT): Identificador único del metadato.
 - `post_id` (BIGINT, FOREIGN KEY `post_id` REFERENCES `wp_posts(ID)`): ID del post al que pertenece el metadato.
 - `meta_key` (VARCHAR(255)): Clave del metadato.
 - `meta_value` (LONGTEXT): Valor del metadato.

4. Tabla `wp_usermeta`

- **Descripción:** Almacena metadatos de los usuarios.
- **Columnas:**
 - `umeta_id` (BIGINT, PRIMARY KEY, AUTO_INCREMENT): Identificador único del metadato.
 - `user_id` (BIGINT, FOREIGN KEY `user_id` REFERENCES `wp_users(ID)`): ID del usuario al que pertenece el metadato.
 - `meta_key` (VARCHAR(255)): Clave del metadato.
 - `meta_value` (LONGTEXT): Valor del metadato.

5. Tabla `wp_comments`

- **Descripción:** Almacena los comentarios realizados en los posts.
- **Columnas:**
 - `comment_ID` (BIGINT, PRIMARY KEY, AUTO_INCREMENT): Identificador único del comentario.
 - `comment_post_ID` (BIGINT, FOREIGN KEY `comment_post_ID` REFERENCES `wp_posts(ID)`): ID del post asociado al comentario.
 - `comment_author` (TINYTEXT): Autor del comentario.
 - `comment_author_email` (VARCHAR(100)): Email del autor del comentario.
 - `comment_author_url` (VARCHAR(200)): URL del autor del comentario.
 - `comment_author_IP` (VARCHAR(100)): IP del autor del comentario.
 - `comment_date` (DATETIME): Fecha y hora del comentario.
 - `comment_date_gmt` (DATETIME): Fecha y hora del comentario en GMT.
 - `comment_content` (TEXT): Contenido del comentario.
 - `comment_karma` (INT): Karma del comentario.
 - `comment_approved` (VARCHAR(20)): Estado de aprobación del comentario.
 - `comment_agent` (VARCHAR(255)): Agente del comentario.
 - `comment_type` (VARCHAR(20)): Tipo de comentario.
 - `comment_parent` (BIGINT): ID del comentario padre (si es una respuesta).
 - `user_id` (BIGINT, FOREIGN KEY `user_id` REFERENCES `wp_users(ID)`): ID del usuario (si es un comentario de usuario registrado).

6. Tabla `wp_commentmeta`

- **Descripción:** Almacena metadatos de los comentarios.
- **Columnas:**
 - `meta_id` (BIGINT, PRIMARY KEY, AUTO_INCREMENT): Identificador único del metadato.
 - `comment_id` (BIGINT, FOREIGN KEY `comment_id` REFERENCES `wp_comments(comment_ID)`): ID del comentario al que pertenece el metadato.
 - `meta_key` (VARCHAR(255)): Clave del metadato.
 - `meta_value` (LONGTEXT): Valor del metadato.

7. Tabla `wp_terms`

- **Descripción:** Almacena términos (categorías, etiquetas).
- **Columnas:**
 - `term_id` (BIGINT, PRIMARY KEY, AUTO_INCREMENT): Identificador único del término.
 - `name` (VARCHAR(200)): Nombre del término.
 - `slug` (VARCHAR(200), UNIQUE): Slug del término.
 - `term_group` (BIGINT): Grupo del término.

8. Tabla `wp_term_taxonomy`

- **Descripción:** Define las taxonomías (categorías, etiquetas).
- **Columnas:**
 - `term_taxonomy_id` (BIGINT, PRIMARY KEY, AUTO_INCREMENT): Identificador único de la taxonomía.
 - `term_id` (BIGINT, FOREIGN KEY `term_id` REFERENCES `wp_terms(term_id)`): ID del término.
 - `taxonomy` (VARCHAR(32)): Nombre de la taxonomía.
 - `description` (LONGTEXT): Descripción de la taxonomía.
 - `parent` (BIGINT): ID del término padre.
 - `count` (BIGINT): Número de objetos asociados a esta taxonomía.

9. Tabla `wp_term_relationships`

- **Descripción:** Relaciona posts con términos (categorías, etiquetas).
- **Columnas:**
 - `object_id` (BIGINT, FOREIGN KEY `object_id` REFERENCES `wp_posts(ID)`): ID del objeto (post).
 - `term_taxonomy_id` (BIGINT, FOREIGN KEY `term_taxonomy_id` REFERENCES `wp_term_taxonomy(term_taxonomy_id)`): ID de la taxonomía del término.
 - `term_order` (INT): Orden del término.

10. Tabla `wp_options`

- **Descripción:** Almacena opciones y configuraciones del sitio.

- **Columnas:**
 - `option_id` (BIGINT, PRIMARY KEY, AUTO_INCREMENT): Identificador único de la opción.
 - `option_name` (VARCHAR(191), UNIQUE): Nombre de la opción.
 - `option_value` (LONGTEXT): Valor de la opción.
 - `autoload` (VARCHAR(20)): Indica si la opción se carga automáticamente.

11. Tabla `wp_links`

- **Descripción:** Almacena los enlaces (blogroll).
- **Columnas:**
 - `link_id` (BIGINT, PRIMARY KEY, AUTO_INCREMENT): Identificador único del enlace.
 - `link_url` (VARCHAR(255)): URL del enlace.
 - `link_name` (VARCHAR(255)): Nombre del enlace.
 - `link_image` (VARCHAR(255)): URL de la imagen del enlace.
 - `link_target` (VARCHAR(25)): Destino del enlace (_blank, _self, etc.).
 - `link_description` (VARCHAR(255)): Descripción del enlace.
 - `link_visible` (VARCHAR(20)): Visibilidad del enlace.
 - `link_owner` (BIGINT): ID del usuario propietario del enlace.
 - `link_rating` (INT): Valoración del enlace.

12. Tabla `wp_termmeta`

- **Descripción:** Almacena metadatos de los términos.
- **Columnas:**
 - `meta_id` (BIGINT, PRIMARY KEY, AUTO_INCREMENT): Identificador único del metadato.
 - `term_id` (BIGINT, FOREIGN KEY `term_id` REFERENCES `wp_terms(term_id)`): ID del término al que pertenece el metadato.
 - `meta_key` (VARCHAR(255)): Clave del metadato.
 - `meta_value` (LONGTEXT): Valor del metadato.

ii. Integridad de Datos (Constraints)

Tabla `wp_posts`

- **Primary Key:** `ID`
- **Foreign Key:** `post_author -> wp_users(ID)`

Tabla `wp_users`

- **Primary Key:** `ID`
- **Unique Constraints:** `user_login, user_email`

Tabla `wp_postmeta`

- **Primary Key:** `meta_id`
- **Foreign Key:** `post_id -> wp_posts(ID)`

Tabla `wp_usermeta`

- **Primary Key:** `umeta_id`
- **Foreign Key:** `user_id -> wp_users(ID)`

Tabla `wp_comments`

- **Primary Key:** `comment_ID`
- **Foreign Key:** `comment_post_ID -> wp_posts(ID), user_id -> wp_users(ID)`

Tabla `wp_commentmeta`

- **Primary Key:** `meta_id`
- **Foreign Key:** `comment_id -> wp_comments(comment_ID)`

Tabla `wp_terms`

- **Primary Key:** `term_id`
- **Unique Constraints:** `slug`

Tabla `wp_term_taxonomy`

- **Primary Key:** `term_taxonomy_id`
- **Foreign Key:** `term_id -> wp_terms(term_id)`

Tabla `wp_term_relationships`

- **Primary Key:** (Composite Key) `object_id, term_taxonomy_id`
- **Foreign Key:** `object_id` -> `wp_posts(ID)`, `term_taxonomy_id` -> `wp_term_taxonomy(term_taxonomy_id)`

Tabla `wp_options`

- **Primary Key:** `option_id`
- **Unique Constraints:** `option_name`

Tabla `wp_links`

- **Primary Key:** `link_id`

Tabla `wp_termmeta`

- **Primary Key:** `meta_id`
- **Foreign Key:** `term_id` -> `wp_terms(term_id)`

5. Lógica/Reglas del Negocio

En el sitio, la lógica del negocio se gestiona principalmente a través de `functions.php` y otros archivos de funciones personalizadas dentro del tema o plugins usados. Esto permite mantener separada la lógica del negocio del resto de la aplicación, facilitando su mantenimiento y mejora. Las reglas específicas de negocio, como las validaciones de datos o decisiones de flujo de proceso, se codifican dentro de estas funciones, aprovechando la rica API de WordPress y su sistema de ganchos (hooks) y filtros para interceptar y modificar el comportamiento estándar de la plataforma.

6. Descripción Interfaz de la aplicación

Página Principal: Bienvenida con hero interactivo, acceso a diferentes secciones como Clases, y Blog, visualización de áreas del gym, testimoniales e instructores.

Header: Barra de navegación responsiva con acceso a inicio, nosotros, clases, galeria, blog y contacto. Incluye logo

Footer: Mismos accesos que el header con un agregado de derechos reservados.

Página nosotros: Muestra información sobre el gym.

Página clases: Muestra listado de clases con las que cuenta el gym, cada clase cuenta con su página de descripción individual.

Página clases: Galería del gimnasio, incorpora plugin lightbox

Página blog: Muestra todas las entradas del blog, cada entrada tiene su página individual con su sección de comentarios.

Inicio de sesión: Formulario para iniciar sesión en el sitio: con correo electrónico , nombre y el comentario.

Contacto: Mapa interactivo con pin editable y con un formulario de contacto con los administradores del gym

Backend: Interfaz para los administradores del sitio, permite agregar, editar y eliminar ya sea entradas de blog, comentarios, páginas, entrenadores, clases, testimoniales, etc. También permite cambiar el tema utilizado, añadir plugins y agregar contenido multimedia para usar en cualquier parte del sitio.

7. Descripción de reglas de seguridad(acceso/operación)

WordPress proporciona un sólido sistema de gestión de usuarios que separa a los usuarios según sus roles y capacidades, lo que facilita la asignación y gestión programática de permisos. Para la seguridad, se pueden implementar prácticas como el uso de HTTPS para todas las transacciones, la protección contra ataques comunes como XSS y SQL Injection mediante la validación adecuada de la entrada del usuario y la limpieza de datos, y el aseguramiento de que los plugins y temas estén actualizados y libres de vulnerabilidades conocidas.

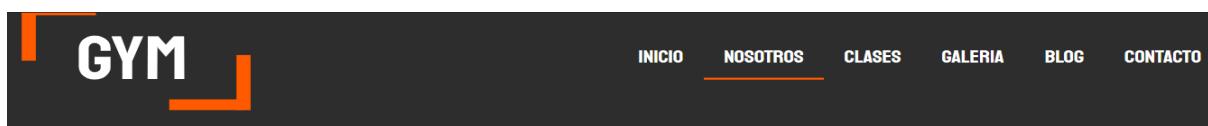
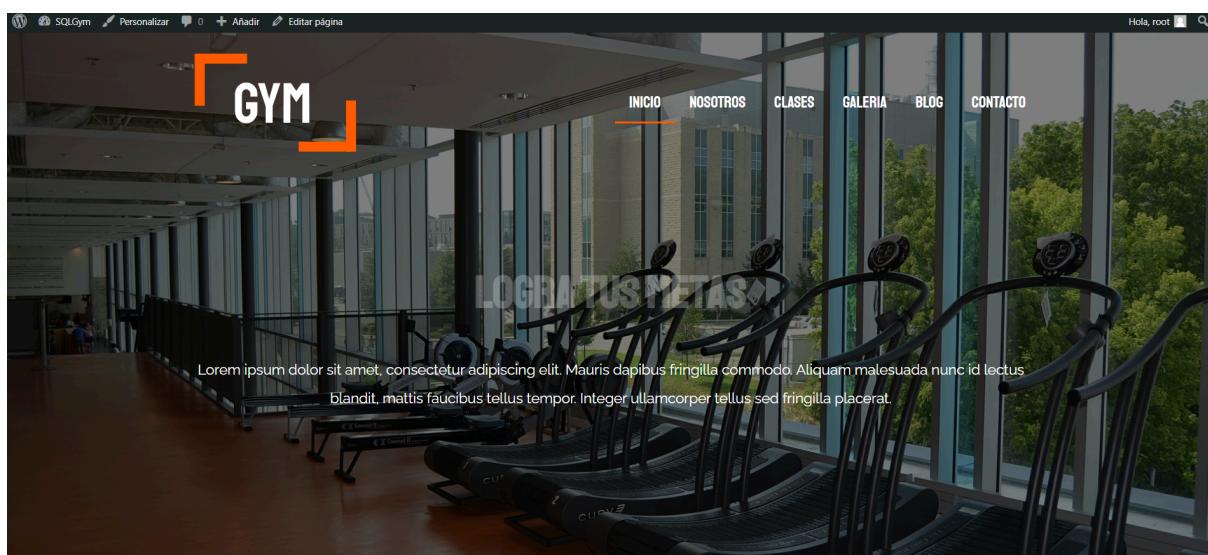
8. Acceso a la Base de Datos mediante la Aplicación

El acceso a la base de datos en el sitio se realiza a través de la clase global `$wpdb`, que proporciona métodos para preparar y ejecutar consultas de manera segura, ayudando a prevenir inyecciones SQL. WordPress también permite la creación de tipos de contenido personalizado (Custom Post Types), que facilitan la modelación de datos sin necesidad de manipular directamente la estructura de la base de datos.

9. Conclusión

El desarrollo de esta página web del gimnasio ha sido diseñado para proporcionar una plataforma robusta, segura y fácil de usar, facilitando una gestión eficaz y mejorando la experiencia del usuario. Con una arquitectura bien definida y reglas de negocio claras, el sistema está preparado para manejar eficientemente las necesidades actuales y futuras del gimnasio. Además fue de utilidad para poder enriquecer más nuestros conocimientos sobre la elaboración de páginas web y bases de datos, así como su conexión.

10. Anexos



SOBRE NOSOTROS



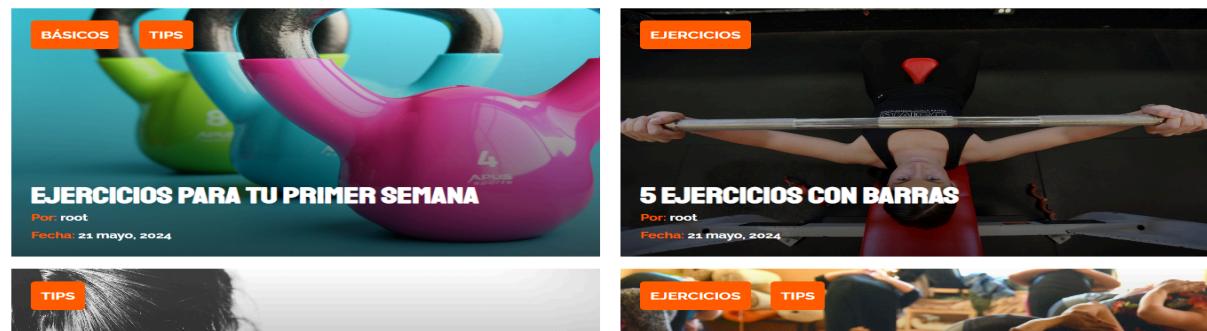
NUESTRAS CLASES

Morbi non diam pulvinar, sagittis mi sed, tincidunt urna. Sed auctor leo nisi, nec sollicitudin quam ultrices non. Phasellus vel est at tortor tincidunt sodales.





GALERIA



CONTACTO



FORMULARIO DE CONTACTO

Tu nombre

11.Bibliografía/Referencias

<https://codex.wordpress.org/es/>

<https://www.php.net/docs.php>

<https://developer.mozilla.org/en-US/docs/Web/HTML>

<https://developer.mozilla.org/es/docs/Web/JavaScript>

<https://dev.mysql.com/doc/>

II. Manual de Usuario

Descripción técnica de la operación del sistema o aplicación

Introducción

El sitio web del gimnasio está diseñado para proporcionar una experiencia completa y dinámica tanto para los visitantes como para los administradores. Permite agregar instructores, testimonios, entradas de blog, clases y páginas adicionales que pueden utilizar estos contenidos. Además, incorpora un sistema de usuarios para interactuar con el blog, un mapa interactivo para mostrar la ubicación del gimnasio y un formulario de contacto.

Requisitos del Sistema

- **Compatibilidad con Navegadores:** El sitio web es compatible con todos los navegadores modernos, incluyendo Google Chrome, Mozilla Firefox, Safari, Edge y Opera.
- **Diseño Responsivo:** El sitio está optimizado para funcionar de manera eficiente en dispositivos móviles, garantizando una experiencia de usuario consistente y agradable en cualquier tamaño de pantalla.

Registro e Inicio de Sesión

- **Registro de Usuarios:**
 - Para registrarse como usuario, dirígete a cualquier entrada del blog y desplázate hacia la parte inferior. Allí encontrarás opciones para registrarte o iniciar sesión, lo que te permitirá dejar comentarios en las entradas del blog.
- **Registro de Administradores:**
 - Para registrar usuarios administradores, accede al backend del sitio ([/wp-admin](#)). Desde la pestaña "Usuarios", puedes añadir nuevos administradores ingresando la información requerida.

Navegación del Visitante

- **Interfaz Principal:**
 - La interfaz principal del sitio web es intuitiva y amigable. Los visitantes pueden navegar fácilmente por el sitio utilizando el menú de navegación ubicado en el header (encabezado) y el footer (pie de página).
- **Acceso a Contenidos:**

- Los visitantes tienen acceso a información sobre instructores, testimonios, entradas del blog, clases y un mapa interactivo que muestra la ubicación del gimnasio. También pueden utilizar el formulario de contacto para comunicarse directamente con el gimnasio.

Navegación del Administrador

- **Panel de Administración:**
 - Los administradores tienen acceso completo al backend de WordPress. Pueden añadir, editar y eliminar contenido, incluyendo entradas del blog, clases, instructores, testimonios e imágenes. Además, pueden gestionar páginas adicionales y plugins.
- **Gestión de Usuarios:**
 - Desde la pestaña "Usuarios", los administradores pueden gestionar todos los usuarios registrados en el sitio, incluyendo la modificación de roles y permisos.
- **Comentarios y Formularios de Contacto:**
 - Los administradores pueden revisar y moderar los comentarios realizados en las entradas del blog. También pueden acceder a los mensajes enviados a través del formulario de contacto para atender las consultas de los visitantes.
- **Personalización del Sitio:**
 - Los administradores tienen la capacidad de modificar la apariencia del sitio web mediante la personalización de temas y widgets, asegurando que el diseño del sitio se alinee con la identidad visual del gimnasio.

Funcionalidades Detalladas

Añadir y Editar Contenido

1. **Entradas del Blog:**
 - **Añadir una Nueva Entrada:** Ve a "Entradas" en el menú del backend y haz clic en "Añadir nueva". Ingresa el título, contenido y cualquier otro dato necesario. Publica o guarda como borrador según sea necesario.
 - **Editar una Entrada:** Selecciona la entrada que deseas editar desde la lista de entradas. Haz los cambios necesarios y actualiza la entrada.
2. **Clases:**

- **Añadir una Nueva Clase:** Ve a "Clases" en el menú del backend y haz clic en "Añadir nueva". Ingresa los detalles de la clase, como el nombre, descripción, horario e instructor asignado. Publica o guarda como borrador.
- **Editar una Clase:** Selecciona la clase que deseas editar desde la lista de clases. Realiza las modificaciones necesarias y actualiza la clase.

3. Instructores:

- **Añadir un Nuevo Instructor:** Ve a "Instructores" en el menú del backend y haz clic en "Añadir nuevo". Ingresa la información del instructor, incluyendo su nombre, biografía y foto. Publica o guarda como borrador.
- **Editar un Instructor:** Selecciona el instructor que deseas editar desde la lista de instructores. Haz los cambios necesarios y actualiza el perfil del instructor.

4. Testimonios:

- **Añadir un Nuevo Testimonio:** Ve a "Testimonios" en el menú del backend y haz clic en "Añadir nuevo". Ingresa el contenido del testimonio y la información del autor. Publica o guarda como borrador.
- **Editar un Testimonio:** Selecciona el testimonio que deseas editar desde la lista de testimonios. Realiza las modificaciones necesarias y actualiza el testimonio.

Gestión de Comentarios

- **Revisar Comentarios:** Ve a "Comentarios" en el menú del backend. Aquí puedes ver, aprobar, responder, editar o eliminar comentarios dejados por los usuarios.
- **Moderar Comentarios:** Utiliza las opciones de moderación para gestionar la calidad y relevancia de los comentarios en tu sitio.

Conclusión

Este manual de usuario proporciona una guía completa sobre cómo interactuar con el sitio web del gimnasio, tanto para visitantes como para administradores. A través de una interfaz intuitiva y funcionalidades robustas, el sitio web facilita la gestión de contenido y la interacción con los usuarios, asegurando una experiencia óptima y eficiente.

Agradecimientos

Queremos expresar nuestro agradecimiento a todos los que han contribuido al desarrollo de este proyecto:

- **Equipo de Desarrollo:** Por su dedicación y habilidades técnicas en la construcción y personalización del sitio.
- **Administradores del Sitio:** Por proporcionar retroalimentación valiosa y pruebas durante el proceso de desarrollo.
- **Clientes y Usuarios del Gimnasio:** Por su paciencia y apoyo durante la fase de implementación.

Cómo Proporcionar Feedback sobre el Sitio

Tu opinión es muy importante para nosotros. Nos ayuda a mejorar y a ofrecer un mejor servicio. Si tienes alguna sugerencia, comentario o encuentras algún problema en el sitio web, por favor sigue estos pasos para proporcionar tu feedback:

1. Formulario de Contacto:

- Dirígete a la página de contacto en nuestro sitio web.
- Llena el formulario con tu nombre, correo electrónico y tu mensaje detallando tu feedback.