**PROYECTO. BIBLIOTECA VIRTUAL**

**PRESENTADO POR:**

**Brayan Tirado  
Jimena Yocue  
Juan Gomez  
Josue Ariza**

**Jose Rojas**

**Fabian Ramirez  
Cristian Jimenez   
Korey Lopez**

**FICHA: 2931558**

**Presentado a**

**Julio Cesar**

**SERVICIO NACIONAL DE APRENDIZAJE**

**02 de Mayo de 202**

**Archivo: assets/js/dashboard.js  
javascript**

**javascript**

**CopiarEditar**

**let sortField = "titulo";**

**let sortReverse = false;**

**let allBooks = [];**

**Este archivo gestiona la funcionalidad del panel de administración, incluyendo ordenamiento, filtrado y operaciones CRUD para libros.  
Maneja la lista de libros, el ordenamiento por diferentes campos (título, autor, género).  
Administra actualizaciones en tiempo real y la funcionalidad de búsqueda.  
Controla las interacciones de la interfaz de administración y la visualización de datos.**

**Archivo: assets/js/upload.js  
javascript**

**javascript**

**CopiarEditar**

**const imageFileInput = document.getElementById('imageFileInput');**

**const formImageInput = document.getElementById('formImageInput');**

**Administra la funcionalidad de carga de archivos tanto para imágenes como para documentos.  
Maneja la funcionalidad de arrastrar y soltar archivos.  
Proporciona vista previa y validación para archivos subidos.  
Gestiona interacciones del modal para cargas de archivos.**

**Archivo: assets/js/filter.js  
javascript**

**javascript**

**CopiarEditar**

**const genreMap = {**

**programacion: 'Programación',**

**desarrollo\_web: 'Desarrollo Web',**

**desarrollo\_aplicaciones: 'Desarrollo de Aplicaciones'**

**// ...**

**};**

**Implementa la funcionalidad de filtrado de libros por género.  
Gestiona la paginación de listados de libros.  
Maneja combinaciones de ordenamiento y filtrado.  
Controla la funcionalidad de favoritos y las interacciones del usuario.**

**Archivo: assets/js/search.js  
javascript**

**javascript**

**CopiarEditar**

**document.addEventListener('DOMContentLoaded', () => {**

**const searchInput = document.getElementById('searchInput');**

**const searchSuggestions = document.getElementById('searchSuggestions');**

**Proporciona sugerencias de búsqueda en tiempo real mientras los usuarios escriben.  
Implementa búsqueda con retardo para optimizar el rendimiento.  
Maneja sugerencias de búsqueda tanto de libros como de autores.  
Administra los envíos del formulario de búsqueda y la navegación.**

**Archivo: index.php  
php**

**php**

**CopiarEditar**

**<?php**

**session\_start();**

**$tiempo\_maximo\_inactividad = 600;**

**Punto de entrada principal de la aplicación.  
Maneja la gestión de sesiones y tiempo de inactividad.  
Implementa carrusel para libros destacados.  
Proporciona diseño de página de inicio y navegación.**

**He documentado estos archivos clave siguiendo el formato existente. ¿Te gustaría que continúe con el análisis de otros archivos del sistema?**

**Continuaré la documentación técnica con el análisis de libro.php y otros archivos principales:**

**Archivo: libro.php  
php**

**php**

**CopiarEditar**

**<?php**

**session\_start();**

**require\_once 'conexion.php';**

**// Obtener ID del libro desde la URL**

**$libro\_id = isset($\_GET['id']) ? (int)$\_GET['id'] : 0;**

**Gestiona las páginas de detalles individuales del libro.  
Maneja la funcionalidad de valoraciones y comentarios de libros.  
Implementa autorización de usuarios para la gestión de comentarios.  
Muestra libros relacionados y funcionalidad de descarga.**

**Características:**

* **Visualización de información del libro**
* **Sistema de puntuación con visualización de estrellas**
* **Sistema de comentarios con operaciones CRUD**
* **Recomendaciones de libros relacionados**
* **Gestión de descargas de archivos**
* **Integración con perfil de usuario**

**Archivo: assets/css/styles.css  
Contiene estilos globales para la aplicación.  
Define diseños responsivos.  
Implementa diseños de tarjetas de libros.  
Gestiona estilos de navegación y pie de página.**

**Archivo: assets/css/dashboard.css  
Estilos específicos para el panel de administración.  
Define diseños de tablas y diseño responsivo.  
Implementa estilos de formularios y modales.  
Maneja estilos de barra lateral y navegación.**

**Archivo: assets/js/index.js  
Gestiona la funcionalidad de la página de inicio.  
Controla carruseles y deslizadores.  
Maneja interacciones del usuario.  
Implementa comportamientos responsivos.**

**Archivo: conexion.php  
php**

**php**

**CopiarEditar**

**$pdo = new PDO('pgsql:host=localhost;port=5432;dbname=web', 'postgres', '1');**

**Gestiona la conexión a la base de datos.  
Implementa PDO para operaciones seguras en base de datos.  
Maneja errores de conexión y configuración.**

**Tablas clave de la base de datos:**

* **documento: Almacena información de libros  
  id, titulo, autor, descripcion, categoria, imagen\_url, publication\_date, enlace**
* **usuarios: Gestiona cuentas de usuario  
  id, nombre, correo, contraseña, rol, foto\_perfil**
* **comentario: Maneja comentarios de libros  
  id, contenido, idusuario, documento\_id, fecha**
* **calificacion: Almacena valoraciones de libros  
  usuario\_id, documento\_id, puntuacion**

**La arquitectura del sistema es modular con separación clara de responsabilidades:**

* **Frontend: HTML, CSS, JavaScript para interfaz de usuario**
* **Backend: PHP para procesamiento del lado del servidor**
* **Base de datos: PostgreSQL para almacenamiento**
* **Almacenamiento de archivos: Directorios separados para documentos e imágenes**

**Archivo: login.php  
php**

**php**

**CopiarEditar**

**require\_once 'conexion.php';**

**session\_start();**

**if ($\_SERVER['REQUEST\_METHOD'] === 'POST') {**

**$correo = trim($\_POST['correo'] ?? '');**

**$password = trim($\_POST['password'] ?? '');**

**Gestiona la funcionalidad de autenticación de usuarios.  
Implementa verificación segura de contraseñas.  
Maneja redirección basada en roles.**

**Características:**

* **Validación de formularios**
* **Verificación con hash de contraseña**
* **Gestión de sesiones**
* **Manejo de errores y mensajes**
* **Control de acceso basado en roles**
* **Interfaz de login responsiva**

**Archivo: register.php  
Gestiona el registro de nuevos usuarios.  
Implementa hash de contraseñas.  
Valida la entrada del usuario.**

**Características:**

* **Validación de correo electrónico**
* **Requisitos de seguridad de contraseña**
* **Verificación de cuenta duplicada**
* **Saneamiento de datos del usuario**

**Archivo: password.php  
Gestiona la funcionalidad de recuperación de contraseña.  
Implementa generación de token de restablecimiento.  
Maneja notificaciones por correo.**

**Características:**

* **Solicitud de restablecimiento de contraseña**
* **Verificación de token**
* **Actualización segura de contraseña**
* **Notificaciones por correo**

**Archivo: logout.php  
php**

**php**

**CopiarEditar**

**session\_start();**

**session\_destroy();**

**header('Location: index.php');**

**Gestiona el cierre de sesión de usuario.  
Maneja la destrucción de la sesión.  
Implementa redirección segura.**

**Flujo de Autenticación:**

* **El usuario envía credenciales de inicio de sesión**
* **El sistema valida los datos**
* **Verifica el hash de la contraseña**
* **Crea sesión con datos del usuario**
* **Redirecciona según el rol del usuario:**
  + **Rol 1: Usuario regular → index.php**
  + **Rol 2: Usuario administrador → dashboard-admin.php**

**Medidas de Seguridad:**

* **Hash de contraseñas con password\_hash() de PHP**
* **Autenticación basada en sesiones**
* **Saneamiento de entradas**
* **Protección CSRF**
* **Control de acceso por roles**
* **Proceso seguro de restablecimiento de contraseña**

**Archivo: settings.php  
php**

**php**

**CopiarEditar**

**// Crear carpeta de uploads si no existe**

**$upload\_dir = 'assets/uploads/user-photo/';**

**if (!is\_dir($upload\_dir)) {**

**mkdir($upload\_dir, 0755, true);**

**}**

**Gestiona configuraciones y ajustes de perfil del usuario.**

**Características:  
Gestión de Información de Perfil**

* **Actualización de nombre y correo**
* **Validación de entrada**
* **Verificación de correos duplicados**

**Gestión de Foto de Perfil**

* **Subida de fotos con límite de tamaño (2MB)**
* **Formatos compatibles: JPG, PNG, WebP**
* **Eliminación automática de fotos antiguas**
* **Funcionalidad de recorte de imagen**

**Gestión de Contraseña**

* **Actualizaciones seguras**
* **Validación de complejidad**
* **Confirmación de contraseña**

**Seguridad de Cuenta**

* **Expiración de sesión (600 segundos)**
* **Opción de eliminación de cuenta**
* **Validación de formularios**

**Archivo: assets/js/passwordValidation.js  
Implementa validación de contraseña del lado del cliente.**

**Características:**

* **Verificación de longitud mínima (10 caracteres)**
* **Requisito de letra mayúscula**
* **Requisito de número**
* **Validación de carácter especial**
* **Indicador de fuerza en tiempo real**
* **Coincidencia de confirmación de contraseña**

**Archivo: assets/css/settings.css  
Define estilos para las páginas de configuración.**

**Implementa:**

* **Diseño responsivo**
* **Estilizado de foto de perfil**
* **Estilizado de elementos de formulario**
* **Indicadores de fuerza de contraseña**
* **Estilo de mensajes de alerta**

**Flujo de Gestión de Configuración:**

**Actualizaciones de Perfil:**

* **Validación de formularios**
* **Actualización en base de datos**
* **Actualización de datos de sesión**

**Gestión de Fotos:**

* **Validación de archivos**
* **Procesamiento de imágenes**
* **Gestión de almacenamiento**

**Medidas de Seguridad:**

* **Reglas de complejidad de contraseñas**
* **Tiempo de expiración de sesión**
* **Formularios con envío seguro**

**Manejo de Errores:**

* **Errores de validación de entrada**
* **Errores en la subida de archivos**
* **Errores en operaciones de base de datos**
* **Errores de gestión de sesión**

**Archivo: assets/js/search.js**

**javascript**

**document.addEventListener('DOMContentLoaded', () => {**

**const searchInput = document.getElementById('searchInput');**

**const searchSuggestions = document.getElementById('searchSuggestions');**

* **Implementa la funcionalidad de búsqueda en tiempo real**
* **Características:**
  + **Sugerencias de búsqueda automáticas**
  + **Filtrado por título y autor**
  + **Optimización con debounce**
  + **Interfaz de usuario interactiva**

**Archivo: settings.php**

**php**

**$upload\_dir = 'assets/uploads/user-photo/';**

**if (!is\_dir($upload\_dir)) {**

**mkdir($upload\_dir, 0755, true);**

**}**

* **Gestiona la configuración del perfil de usuario**
* **Características:**
  + **Gestión de Información Personal**
    - **Actualización de nombre y correo**
    - **Validación de entradas**
    - **Verificación de correos duplicados**
  + **Gestión de Fotos de Perfil**
    - **Límite de tamaño: 2MB**
    - **Formatos soportados: JPG, PNG, WebP**
    - **Eliminación automática de fotos antiguas**
  + **Seguridad de Cuenta**
    - **Actualización segura de contraseñas**
    - **Tiempo de inactividad (600 segundos)**
    - **Opción de eliminación de cuenta**

**Archivo: password.php**

**php**

**$token = bin2hex(random\_bytes(16));**

**date\_default\_timezone\_set('America/Bogota');**

**$expiracion = date('Y-m-d H:i:s', strtotime('+5 minutes'));**

* **Sistema de recuperación de contraseña**
* **Características:**
  + **Generación de Tokens**
    - **Tokens aleatorios seguros**
    - **Expiración en 5 minutos**
    - **Almacenamiento en base de datos**
  + **Notificaciones por Correo**
    - **Plantilla HTML personalizada**
    - **Enlace seguro de restablecimiento**
    - **Mensajes personalizados**

**Archivo: login.php**

**php**

**if ($usuario && password\_verify($password, $usuario['password'])) {**

**$\_SESSION['usuario\_id'] = $usuario['id'];**

**$\_SESSION['usuario\_nombre'] = $usuario['nombre'];**

* **Sistema de autenticación de usuarios**
* **Características:**
  + **Validación de Credenciales**
  + **Gestión de Sesiones**
  + **Redirección según Rol**
  + **Mensajes de Error Personalizados**

**Estructura de Base de Datos:**

1. **Tabla documento:**
   * **id, titulo, autor, descripcion, categoria**
   * **imagen\_url, publication\_date, enlace**
2. **Tabla usuarios:**
   * **id, nombre, correo, contraseña, rol**
   * **foto\_perfil, reset\_token, reset\_token\_expiration**
3. **Tabla comentario:**
   * **id, contenido, idusuario, documento\_id, fecha**
4. **Tabla calificacion:**
   * **usuario\_id, documento\_id, puntuacion**

**Arquitectura del Sistema:**

* **Frontend: HTML, CSS, JavaScript**
* **Backend: PHP**
* **Base de Datos: PostgreSQL**
* **Almacenamiento: Sistema de archivos para documentos e imágenes**

**Thinking**

**2 Steps**

**Continuaré documentando el archivo libro.php:**

**Archivo: libro.php**

* **Gestiona la visualización y funcionalidad detallada de cada libro**
* **Características principales:**

1. **Gestión de Visualización:**

**php**

**// Obtener detalles del libro**

**try {**

**$query = "SELECT id, titulo, autor, descripcion, categoria, imagen\_url, publication\_date, enlace**

**FROM documento**

**WHERE id = :id";**

* **Muestra información completa del libro**
* **Maneja imágenes y archivos descargables**
* **Presenta información estructurada (título, autor, descripción)**

1. **Sistema de Calificaciones:**

**php**

**// Obtener calificación promedio**

**try {**

**$query = "SELECT AVG(puntuacion)::numeric(3,1) as promedio, COUNT(\*) as total**

**FROM calificacion**

**WHERE documento\_id = :documento\_id";**

* **Calificación promedio con decimales**
* **Contador de total de calificaciones**
* **Visualización de estrellas dinámica**

1. **Sistema de Comentarios:**

**php**

**// Obtener comentarios**

**try {**

**$query = "SELECT c.id, c.contenido AS comentario, c.fecha, c.idusuario, u.nombre, u.foto\_perfil**

**FROM comentario c**

**JOIN usuarios u ON c.idusuario = u.id"**

* **Gestión completa de comentarios**
* **Integración con fotos de perfil**
* **Funcionalidad de eliminación para usuarios autorizados**

1. **Libros Relacionados:**

**php**

**// Obtener libros relacionados**

**try {**

**$query = "SELECT id, titulo**

**FROM documento**

**WHERE categoria = :categoria AND id != :id**

**LIMIT 3";**

* **Muestra libros de la misma categoría**
* **Excluye el libro actual**
* **Límite de 3 sugerencias**

1. **Seguridad e Interactividad:**

* **Validación de sesiones de usuario**
* **Protección contra SQL injection**
* **Manejo de transacciones para operaciones múltiples**
* **Interfaz interactiva para calificaciones**

1. **Categorización:**

**php**

**$categorias = [**

**'programacion' => 'Programación',**

**'desarrollo\_web' => 'Desarrollo Web',**

**'desarrollo\_aplicaciones' => 'Desarrollo de Aplicaciones',**

**// ...**

**];**

* **Sistema de categorización estructurado**
* **Traducción de identificadores técnicos a nombres amigables**
* **Facilita la navegación y organización**

1. **Gestión de Errores:**

* **Manejo de excepciones en operaciones de base de datos**
* **Mensajes de error personalizados**
* **Registro de errores para depuración**

1. **Interfaz Responsiva:**

* **Diseño adaptable con Bootstrap**
* **Iconografía con Font Awesome**
* **Estructura clara y organizada del contenido**

**Este archivo es crucial para la experiencia del usuario, combinando funcionalidades de visualización, interacción y gestión de datos.**