

## **Guía 2 Ingeniería de Software**

**Bryan Stiven Failach Durango**

**Laura Valentina González Rojas**

**Valentina Blanco Alvis**

**Julián Mauricio Montoya Rodríguez**

**Bryan Andres Puentes Vargas**

**Ingeniería Web**

**Diana Marcela Toquica**

**Universidad Manuela Beltrán**

**Ingeniería de Software**

**Bogotá D.C**

**4 de febrero de 2025**

**Tabla de contenido**

Estructura y diseño .....	3
Acceso .....	3
Catálogo .....	3
Principal .....	3
Compras .....	3
Tecnologías.....	4
Consumo de APIS .....	4
API de Pago (PayU).....	4
Conexión Frontend - Backend.....	4
Manejo de Estados y Errores .....	4
Interactividad y Validación .....	4
Interactividad en el Frontend (React.js) .....	4
Validaciones en Formularios .....	4
Inicio de Sesión / Registro:.....	4
Carrito de Compra y Pagos:.....	5
Manejo de Estados de Carga y Errores .....	5
Experiencia de Usuario Mejorada.....	5
Adaptabilidad .....	5
Diseño Responsive con Media Queries .....	5
Compatibilidad en Distintos Dispositivos.....	5
Referencias.....	6

## Estructura y diseño

El proyecto se organizará en las siguientes carpetas:

### Acceso

Contiene las páginas relacionadas con el inicio de sesión y el registro de usuarios.

- » Iniciar Sesión: Formulario para ingresar con credenciales.
- » Registro: Formulario de creación de cuenta.
- » Conexión a base de datos: Gestión de autenticación y almacenamiento de usuarios.

### Catálogo

Maneja la visualización de libros y sus detalles.

- » Catálogo principal: Listado de libros disponibles.
- » Descripción del libro: Información detallada de cada libro.
- » Reseñas: Opiniones de los usuarios sobre los libros.
- » Recomendaciones: Sugerencias personalizadas basadas en intereses y compras previas.

### Principal

Incluye la funcionalidad de búsqueda y organización del catálogo.

- » Búsqueda: Barra de búsqueda para encontrar libros específicos.
- » Categorías: Organización de libros por géneros o temas.
- » Filtros: Opciones para refinar la búsqueda según precio, autor, editorial, etc.

### Compras

Gestiona todo el proceso de compra.

- » Carrito de compra: Vista y gestión de los libros agregados al carrito.
- » Confirmación de pago: Página de revisión y confirmación antes de pagar.
- » Detalle de compra: Información de la transacción y estado del pedido.

- » API de pago: Integración con pasarelas de pago seguras.

### **Tecnologías**

- » Frontend: React.js para la construcción de la interfaz de usuario.
- » Estilos: CSS para el diseño y la presentación visual.

### **Consumo de APIS**

#### **API de Pago (PayU)**

Se integrará PayU para procesar pagos en línea de manera segura.

Implementación de flujo de pago con validaciones y respuesta del estado de la transacción.

#### **Conexión Frontend - Backend**

Comunicación mediante Fetch API para interactuar con el backend.

Métodos HTTP como GET, POST, PUT, DELETE según sea necesario.

#### **Manejo de Estados y Errores**

Se usará React.js para gestionar los estados de carga y errores.

Mensajes de error y feedback visual en caso de fallos en la API.

### **Interactividad y Validación**

#### **Interactividad en el Frontend (React.js)**

Implementación de eventos dinámicos (clicks, desplazamiento, cambios en formularios).

Animaciones y transiciones suaves para mejorar la experiencia del usuario.

#### **Validaciones en Formularios**

*Inicio de Sesión / Registro:*

Verificación de campos obligatorios.

Validación de correos electrónicos y contraseñas seguras.

Mensajes de error y confirmación en tiempo real.

*Carrito de Compra y Pagos:*

Validación de cantidad de productos disponibles.

Verificación de datos antes de proceder al pago.

Confirmación de transacción exitosa o manejo de errores.

**Manejo de Estados de Carga y Errores**

Indicadores de carga al realizar peticiones a la API.

Mensajes de error y reintentos en caso de fallos en la conexión con el backend.

Alertas o notificaciones en la UI cuando una operación se completa exitosamente.

**Experiencia de Usuario Mejorada**

Autocompletado en búsqueda para facilitar la selección de libros.

Sugerencias en reseñas y recomendaciones personalizadas.

Notificaciones de estado del pedido tras la compra.

**Adaptabilidad**

**Diseño Responsive con Media Queries**

Uso de CSS Media Queries para adaptar la UI a distintos tamaños de pantalla.

Implementación de un diseño flexible con flexbox y grid para mejorar la disposición de los elementos.

Ajuste de fuentes, botones e imágenes para dispositivos móviles, tablets y desktops.

**Compatibilidad en Distintos Dispositivos**

Pruebas en navegadores como Chrome

Verificación en dispositivos móviles y tablets para garantizar una experiencia fluida.

### **Referencias**

Diana Marcela Toquica Rodríguez. 2025. Material de clase. Universidad Manuela Beltrán, ingeniería Web.