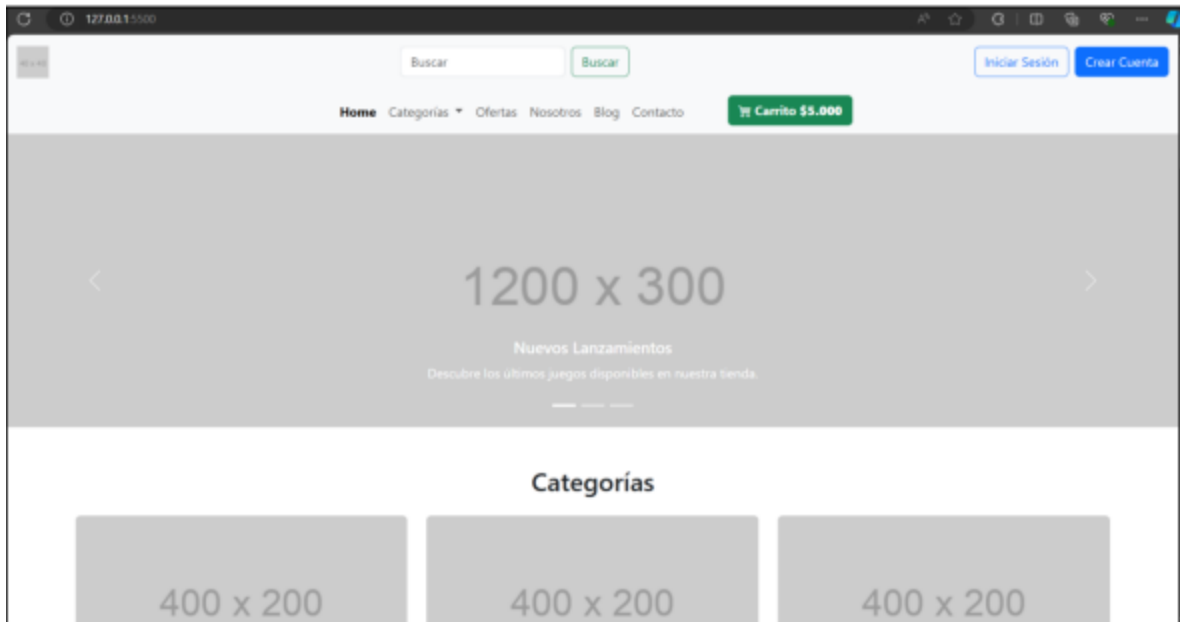


Instrucciones para el desarrollo de la Evaluación 2 (30%)

- ❖ Incorpora frontend basado en tecnología JS para la construcción de una aplicación web, integrando elementos de responsividad para el desarrollo de páginas web simples. Comprueba que el proyecto use un framework moderno de JavaScript, como React y que esté bien estructurado.
- ❖ Migración de HTML a React: o Convertir las páginas HTML existentes en componentes de React reutilizables. o Asegurarse de que cada componente maneje su propio estado y propiedades de manera eficiente.
- ❖ Fuente de datos simulada:
Crear un módulo que actúe como una fuente de datos simulada para la aplicación.
Implementar funciones para realizar operaciones CRUD (Crear, Leer, Actualizar, Eliminar) sobre los datos.
- ❖ Diseño responsivo utilizando Bootstrap: o Incorporar Bootstrap para asegurar que el diseño sea responsivo para proporcionar una experiencia de usuario óptima en móviles, tabletas y escritorios.
- ❖ Vistas adicionales: o Añadir nuevas vistas basadas en los requerimientos actuales del negocio o cliente, como paneles de administración o páginas de detalles de productos.
- ❖ Interactividad mejorada: o Incorporar características interactivas adicionales utilizando React, como formularios avanzados, filtros de búsqueda y navegación mejorada.
- ❖ Configuración de pruebas: o Configurar el entorno de pruebas usando VITEST, React testing para ejecutar pruebas unitarias en componentes de React.
- ❖ Desarrollo de pruebas unitarias: o Escribir pruebas para verificar la lógica y el comportamiento de los componentes, asegurando que cada parte de la aplicación funcione como se espera.

“Al crear un componente en React, sigue el principio de "Single Responsibility". Un componente debe tener una única responsabilidad o propósito en la aplicación. Esto no solo hace que el componente sea más fácil de entender y probar, sino que también mejora la reutilización. Asegúrate de que el componente tenga un nombre claro y descriptivo, utilice props para recibir datos y funciones necesarias, y maneje su propio estado solo cuando

Template web, módulos de vistas para la propuesta entrega 2.



Propuesta de pruebas con Vitest:

Utilizar VITEST, React testing para escribir y ejecutar pruebas unitarias que aseguren la funcionalidad y calidad del código en una aplicación React. A través de estas pruebas, debes verificar el renderizado, el manejo de propiedades (props), los eventos en los componentes React.

- ❖ Pruebas de renderizado: o Renderizado correcto: Escribe pruebas que verifiquen que los componentes se rendericen con los datos proporcionados.
 - Asegúrate de que los componentes de lista rendericen todos los elementos de un conjunto de datos. o Renderizado condicional: Crea pruebas para comprobar que los componentes muestren y/o oculten elementos según las condiciones especificadas.
 - Verifica que un mensaje de error solo se muestre cuando hay un error presente.
- ❖ Pruebas de Eventos: o Simulación de Eventos: Simula eventos de usuario para verificar que los componentes reaccionen como se espera.
 - Simula un clic en un botón y comprueba que el estado del componente cambie o que se ejecute una función específica.





Instrucciones para la entrega II

La entrega de consiste en 2 partes, la primera es la entrega y presentación de la solución de la propuesta y la segunda es la ronda de preguntas abiertas donde cada integrante del grupo deberá responder a preguntas abiertas formuladas por el docente.

Estas preguntas evaluarán la comprensión técnica y los cambios realizados en GitHub durante el desarrollo frontend.

❖ El equipo deberá definir los requerimientos, herramientas y propuestas del proyecto mediante un documento ERS (documento actualizado).

❖ Documento de cobertura del testing: Propuesta y análisis de los test realizados. Entrega del encargo:

-  Enlace GitHub público del proyecto frontend.
-  Proyecto frontend comprimido.
-  Documento ERS (actualizado V2).
-  Documento de cobertura del testing.

Presentación del caso:

- Presentación del caso en 10 minutos cada equipo, más 5 minutos para una ronda de preguntas.
- Ronda de preguntas abiertas.