



TIENDA Y BIBLIOTECA

INTEGRANTES

JORGE ANDRÉS NAVARRO RAMOS

DOCENTE

ALEXANDER TOSCANO

TÉCNICAS AVANZADAS DE PROGRAMACIÓN

UNIVERSIDAD DE CÓRDOBA

LICENCIATURA EN INFORMÁTICA CON ÉNFASIS EN MEDIOS AUDIOVISUALES

FACULTAD DE EDUCACIÓN Y CIENCIAS HUMANAS

MONTERÍA – CÓRDOBA

2024

Descripción General:

"Tienda y Biblioteca" es una aplicación diseñada para los amantes de la lectura y la escritura. Su objetivo principal es facilitar la compra y venta de libros nuevos y usados, así como ofrecer una plataforma para la publicación de trabajos y libros, conectando a lectores, autores y vendedores en un solo lugar.

Características Principales:

1. Marketplace de Libros:

- **Compra y Venta:** Los usuarios pueden explorar una amplia variedad de libros nuevos y de segunda mano. Los vendedores particulares y librerías pueden listar sus productos, ofreciendo opciones para todos los presupuestos.
- **Filtrado Avanzado:** La aplicación permite a los usuarios buscar por género, autor, año de publicación y estado del libro (nuevo o usado), facilitando la búsqueda de su próxima lectura.

2. Publicación de Obras:

- **Autopublicación:** Los escritores pueden publicar sus libros directamente en la plataforma, proporcionando herramientas de formato y diseño para que sus obras luzcan profesionales.
- **Revisión y Comentarios:** Los lectores pueden dejar reseñas y calificaciones, lo que ayuda a otros a descubrir nuevos talentos y mejora la visibilidad de los autores.

3. Interfaz Intuitiva:

- **Diseño Amigable:** La aplicación cuenta con una interfaz sencilla y atractiva, diseñada para que tanto lectores como escritores naveguen sin

dificultad. La experiencia de usuario está optimizada para facilitar la compra, venta y publicación de manera fluida.

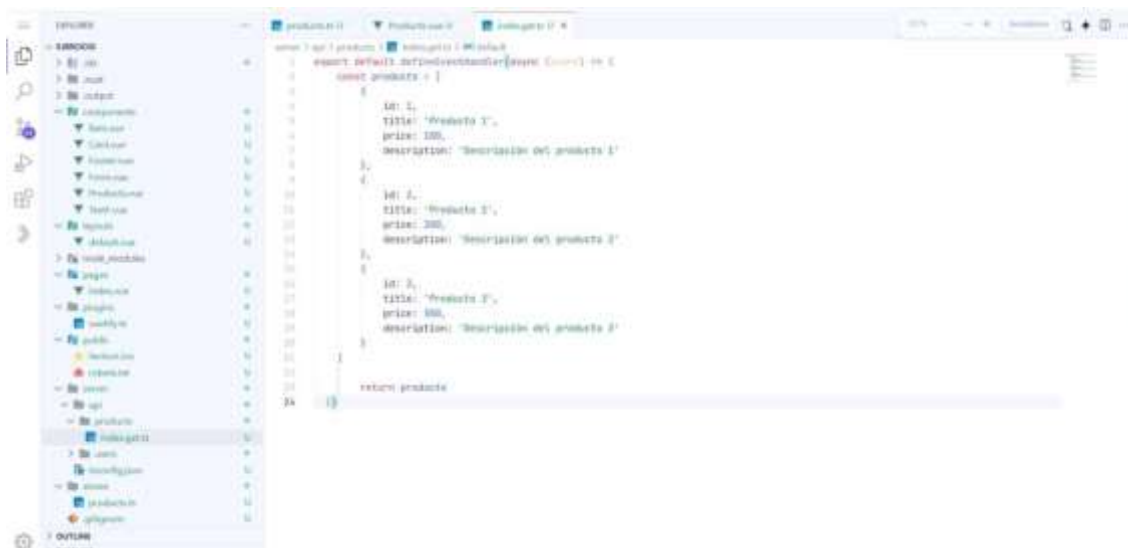
Objetivo:

"Tienda y Biblioteca" aspira a convertirse en un punto de encuentro para aquellos que aman la literatura en todas sus formas. Ya sea comprando un libro que ha estado en su lista de deseos, vendiendo uno que ya no necesitan, o compartiendo su propia historia con el mundo.

Posible vista de la aplicación final:



Server

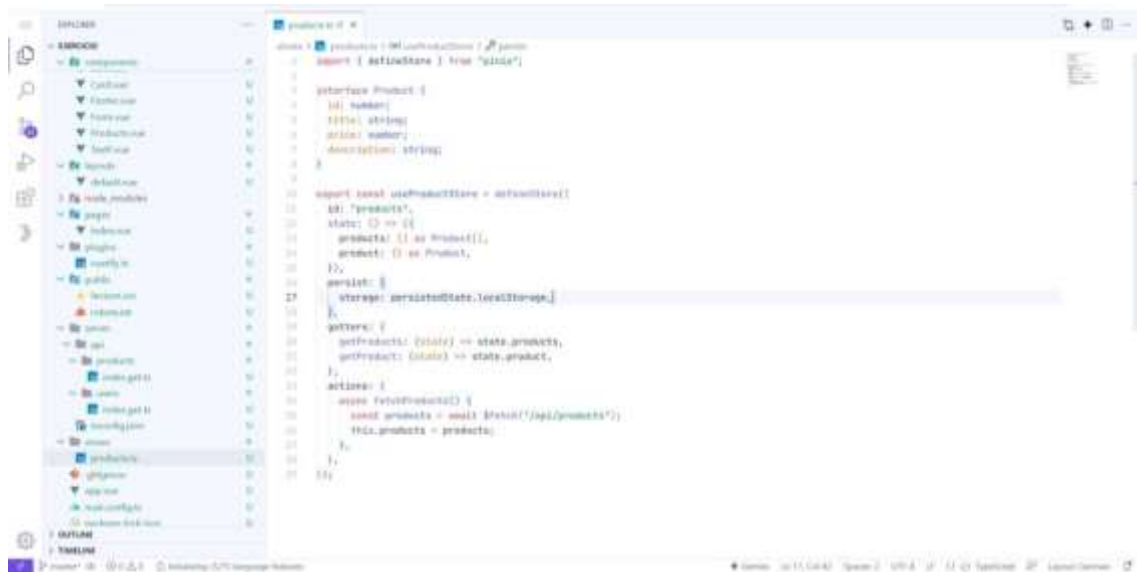


Este código define un manejador de eventos asíncrono en un servidor, que devuelve una lista de productos en formato JSON.

1. `defineEventHandler`: Es una función que define un manejador para una ruta de servidor (por ejemplo, en un framework como Nuxt 3).
2. `products`: Es un arreglo de objetos que representa productos, donde cada objeto tiene:
 - `id`: Un identificador único.
 - `title`: El nombre del producto.
 - `price`: El precio del producto.
 - `description`: Una breve descripción del producto.
3. `return products`: El manejador devuelve el arreglo `products` como respuesta a la solicitud (`request`), lo que significa que la lista de productos será enviada al cliente que haga la petición.

Este código es un simple endpoint de servidor que responde con una lista de productos estática cuando se hace una solicitud a dicha ruta.

State



Este código define un store en Pinia para gestionar el estado de productos en la aplicación.

1. interface Product: Define la estructura de un producto, con las propiedades `id`, `title`, `price` y `description`, todas tipadas correctamente para garantizar que cada producto tenga estos atributos.

2. defineStore: Esta función crea una tienda de estado, que en este caso tiene:

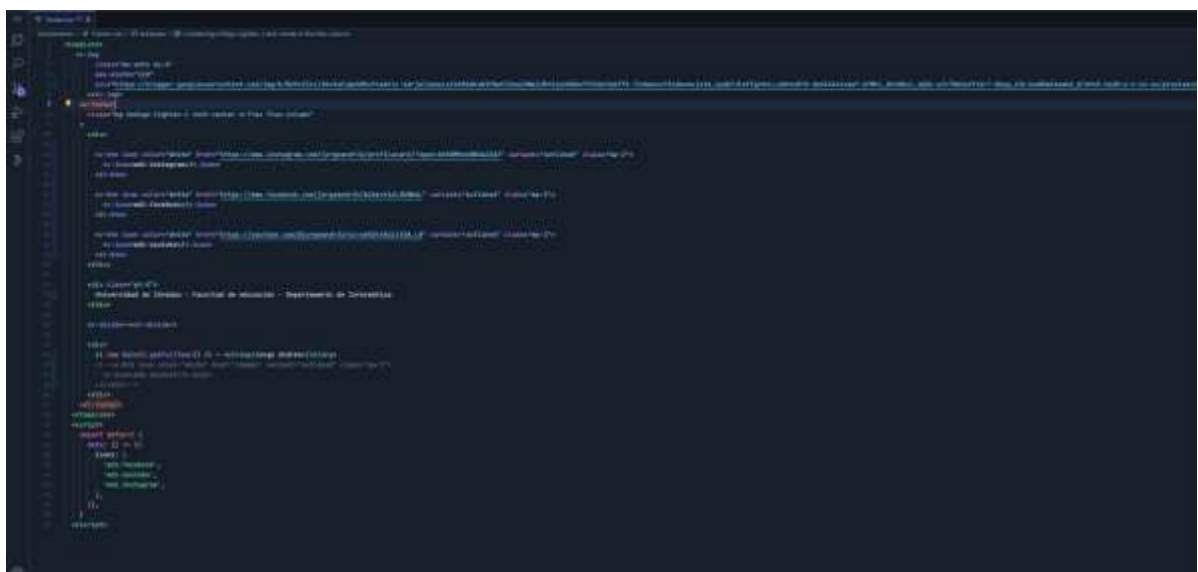
- id: "products": El identificador único de la tienda (en este caso, "products").
- state: Es el estado inicial de la tienda. En este caso, tiene dos propiedades:
 - i) products: Un arreglo vacío que almacenará los productos. ii)
 - product: Un objeto vacío que almacenará un solo producto.
- persist: Utiliza `localStorage` para guardar el estado de la tienda en el almacenamiento local del navegador, asegurando que los datos persistan entre recargas de página.

- getters: Define dos "getters" que permiten acceder de manera fácil a los datos del estado:
- getProducts: Devuelve el arreglo de productos.
- getProduct: Devuelve un solo producto.
- actions: Contiene una acción asíncrona llamada `fetchProducts`, que realiza una solicitud HTTP para obtener los productos desde la API (`/api/products`) y luego los asigna a `this.products` en el estado de la tienda.

Este código define una tienda Pinia para gestionar productos. Permite:

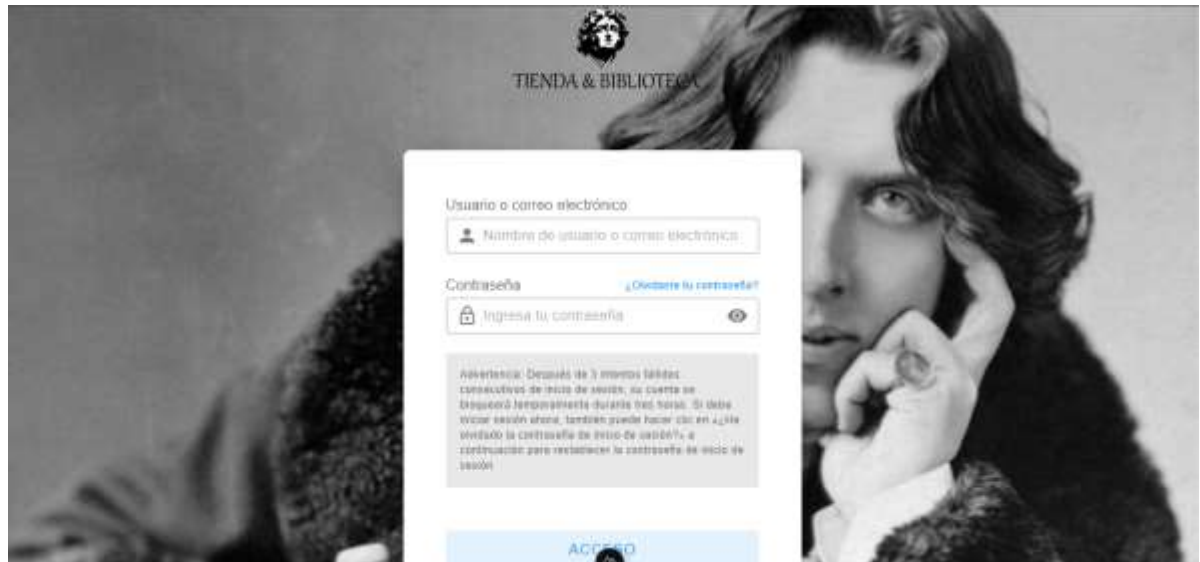
- Obtener todos los productos (`getProducts`).
- Obtener un solo producto (`getProduct`).
- Traer los productos desde un endpoint (server) API (`fetchProducts`) y almacenarlos en el estado de la tienda.
- Los datos de la tienda se mantienen persistentes en el `localStorage` del navegador.

Footer





Login



TIENDA & BIBLIOTECA

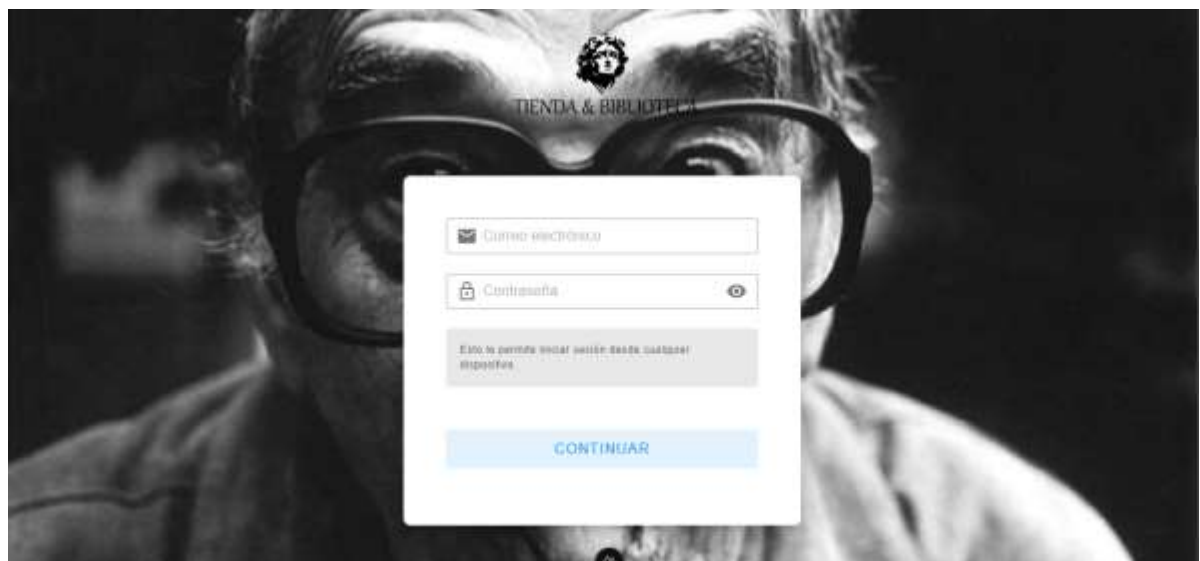
Usuario o correo electrónico

Contraseña [¿Olvidaste tu contraseña?](#)

¡Atención! Después de 5 intentos fallidos consecutivos de inicio de sesión, su cuenta se bloqueará temporalmente durante tres horas. Si debe iniciar sesión ahora, también puede hacer clic en «¿Olvidaste tu contraseña?» para restablecer la contraseña de inicio de sesión.

ACCESO

Register



TIENDA & BIBLIOTECA

Esto no permite iniciar sesión desde cualquier dispositivo.

CONTINUAR