

Evaluación Final Transversal

Encargo de Desarrollo para la Asignatura Desarrollo Fullstack II

Descripción del Proyecto:

Los estudiantes de la asignatura **Desarrollo Fullstack II** están encargados de desarrollar una **tienda online**. Este proyecto integra el uso de **React** para el frontend y el desarrollo del backend utilizando **Spring Boot**, dando continuidad a lo aprendido en la asignatura de Fullstack I.

Formación de equipo: Máximo 3 estudiantes por equipo.

Casos para desarrollar:

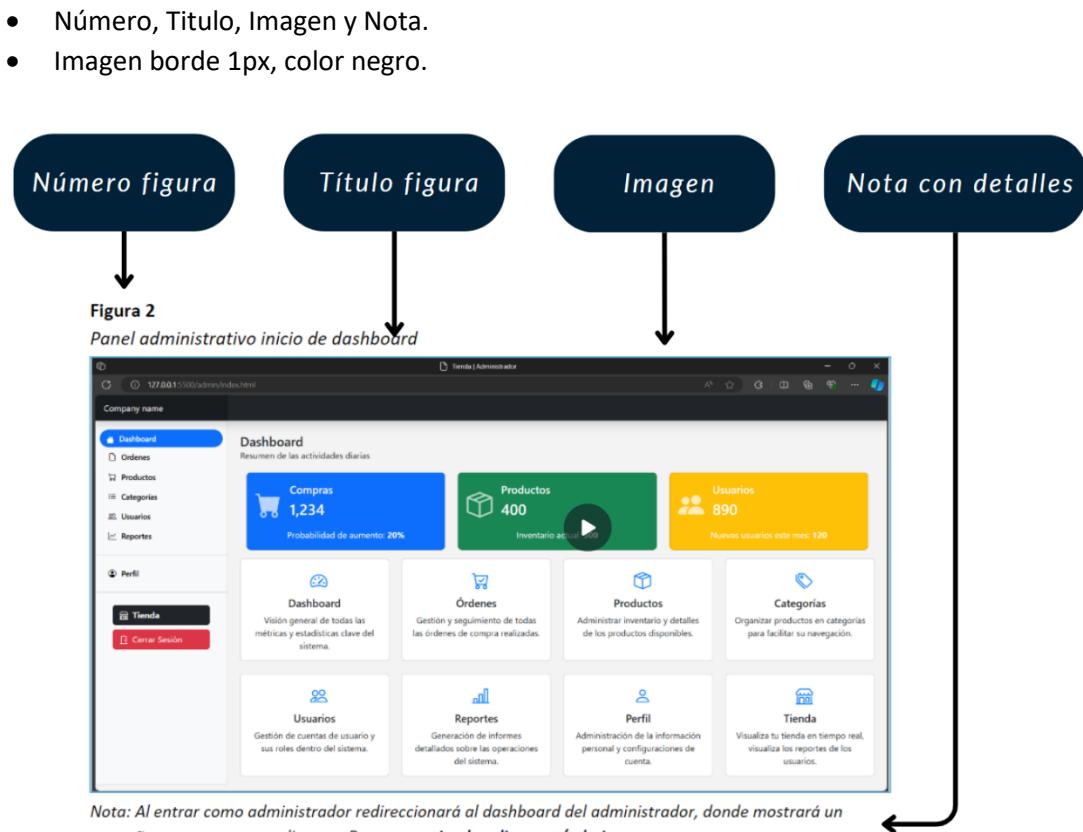
- **Propuestas Sugeridas:** Se ofrecen tres propuestas de desarrollo para la tienda online, cada una con los mismos requisitos técnicos, pero con diferente contenido. Estas propuestas están diseñadas para ofrecer una base estructurada y guiada para el desarrollo del proyecto.



- **Propuesta Personalizada:** Los estudiantes tienen la opción de desarrollar una tienda online basada en un emprendimiento familiar o de algún conocido. Para proceder con esta opción, deberán presentar su propuesta al docente, quien verificará si cumple con los requerimientos necesarios para el examen. Esta opción permite a los estudiantes elegir un tema que les resulte significativo y que los motive durante el desarrollo del proyecto.

Evidencia de la documentación:

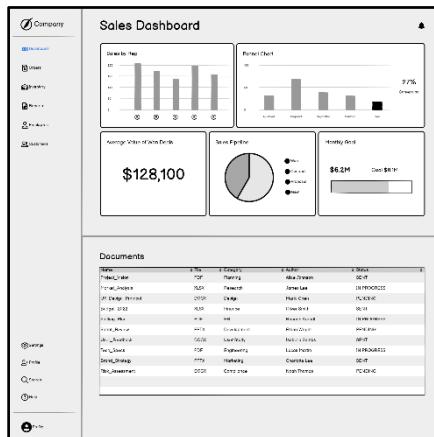
Cada imagen añadida a los documentos se debe documentar de la siguiente manera:



Ejemplos

Figura 1

Mockups dashboard administrativo



Nota: Al entrar como administrador redireccionará al dashboard administrador donde mostrará un pequeño reporte.

Figura 2

Servicios ofrecidos a los estudiantes en la página de Duoc UC para acceder al AVA.



Nota: Acceso al AVA mediante la página de Duoc UC. Ruta: <https://www.duoc.cl/alumnos/ava-blackboard/>.

Instrucciones para el desarrollo de la Evaluación 1 (30%)

Objetivos para la evaluación 1:

Desarrollar una tienda online básica utilizando HTML, CSS y JavaScript. Este proyecto sentará las bases para el desarrollo en futuras evaluaciones. El enfoque de esta evaluación es dominar la estructura y el diseño web básico, estilos propios, validaciones en formulario. y la colaboración efectiva con herramientas de control de versiones.

Requisitos del proyecto:

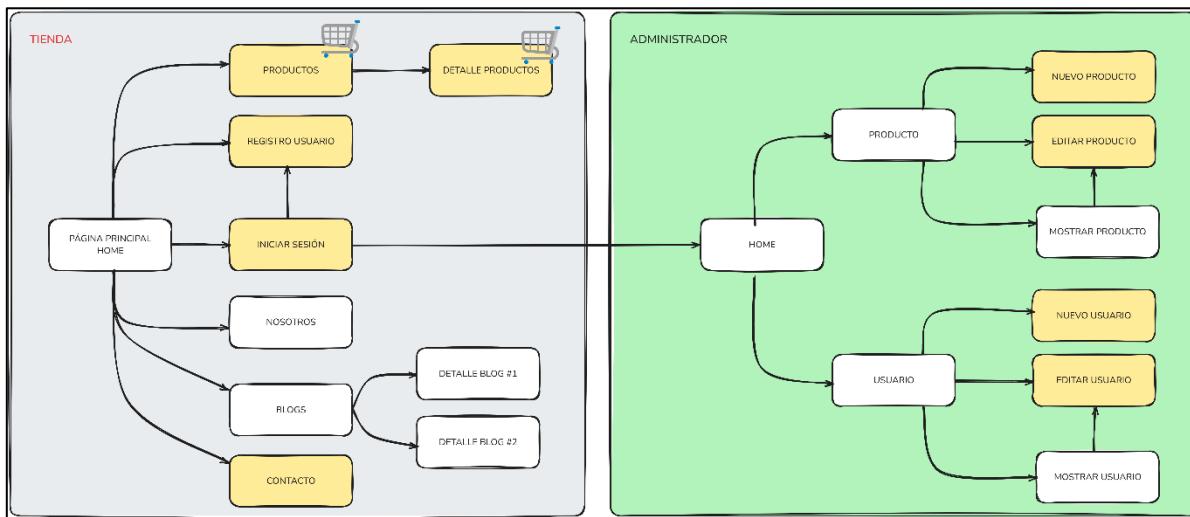
- ❖ **Estructura y etiquetado con HTML:**
 - Crear contenido web utilizando la versión actual de HTML.
 - Implementar una estructura clara y semántica que incluya secciones, encabezados, párrafos y listas donde sea necesario.
- ❖ **Navegación y elementos interactivos:** Implementar una navegación completa que incluya:
 - Hipervínculos para la navegación entre páginas.
 - Imágenes y botones para la interacción del usuario.
 - Elementos de navegación como menús y barras laterales.
 - Formularios para el ingreso de datos.
- ❖ **Diseño con CSS:**
 - Implementar una hoja de estilos CSS personalizada.
 - Incorporar la hoja de estilos de manera externa para facilitar el mantenimiento.
 - Asegurar que el diseño sea consistente y atractivo visualmente.
 - Desarrollar una hoja de estilos externa para aplicar diseño responsive y estético.
 - Usar selectores CSS y propiedades para personalizar el diseño, asegurando que sea consistente en todas las páginas.
- ❖ **Validación de formularios con JavaScript:**
 - Desarrollar validaciones para formularios utilizando JavaScript.
 - Implementar sugerencias y mensajes de error personalizados para mejorar la usabilidad.
 - Implementar funciones de JavaScript para validar formularios en tiempo real.
 - Añadir mensajes de error y sugerencias dinámicas para mejorar la experiencia del usuario.
- ❖ **Colaboración y uso de repositorios:**
 - Configurar un repositorio remoto en GitHub para el proyecto.
 - Realizar cambios en el repositorio con comentarios claros y coherentes.
 - Colaborar efectivamente con los miembros del equipo, distribuyendo tareas de manera equitativa.

El desarrollo de sistema consta de 2 partes, una parte que es la tienda y la otra el sistema administrativo.

En la siguiente imagen representa el flujo de navegación entre páginas y todas las vistas que se deben desarrollar para la entrega 1.

Figura 1

Diagrama de flujo de navegación del proyecto, propuesta entrega 1.



Detalle de los mockups web por cada vista:

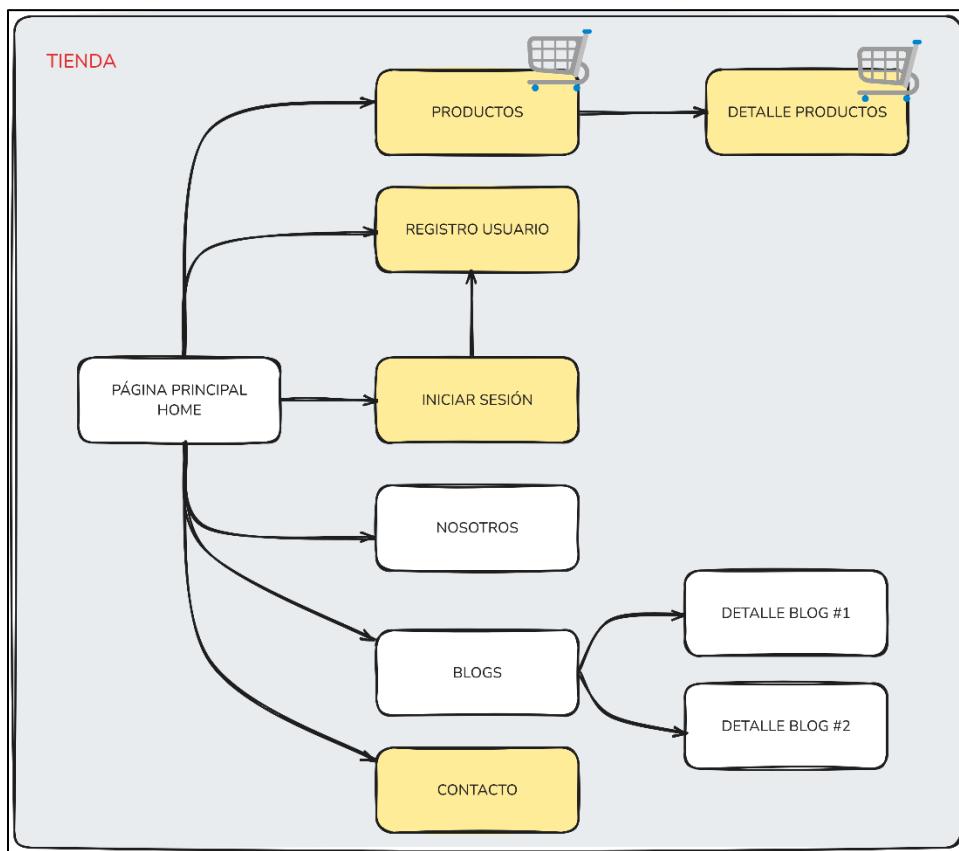
Los contenidos presentados en cada mockup son **requisitos mínimos y sugerencias** para el desarrollo del proyecto.

❖ Vistas tienda

Esta vista es de manera pública y será la cara visible de la tienda.

Figura 2

Diagrama de flujo de navegación de la tienda, propuesta entrega 1.



- **Página principal (HOME):**

- Desarrolla una página que se pueda navegar entre las otras páginas.
- Crea un menú de navegación, incorporando el logo de la tienda y un carrito de compra.
- Implementa un componente principal donde se muestre la información e imagen de la tienda.
- Muestra una lista de productos con su imagen, nombre y precio.
- Añade un footer.

Figura 3

Mockup página principal, propuesta entrega 1.

■ Site Name

[Home](#) | [Productos](#) | [Nosotros](#) | [Blogs](#) | [Contacto](#)

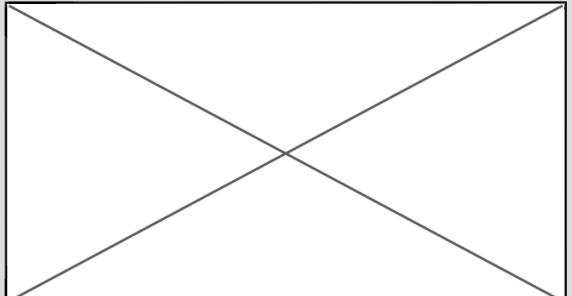
Cart (0)

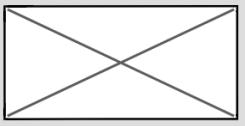
[Iniciar sesión](#) | [Registrar usuario](#)

TIENDA ONLINE

Our North American Field Guides provide tips for identifying birds in all of the regions of the United States and Canada. Download any of our guides for free now!

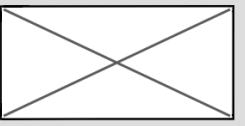
[!\[\]\(313b3b3c8a0c38ad35f0f4cceb5f9abb_img.jpg\) ver productos](#)





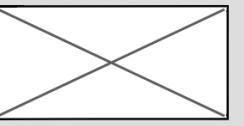
Product Title

Attributes \$00.00



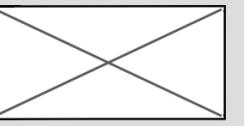
Product Title

Attributes \$00.00



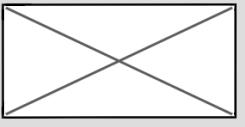
Product Title

Attributes \$00.00



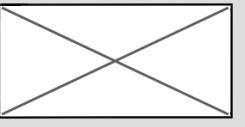
Product Title

Attributes \$00.00



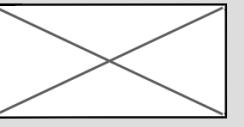
Product Title

Attributes \$00.00



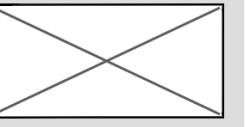
Product Title

Attributes \$00.00



Product Title

Attributes \$00.00



Product Title

Attributes \$00.00

Site Name [Category X](#) | [Category Y](#) | [Category Z](#)

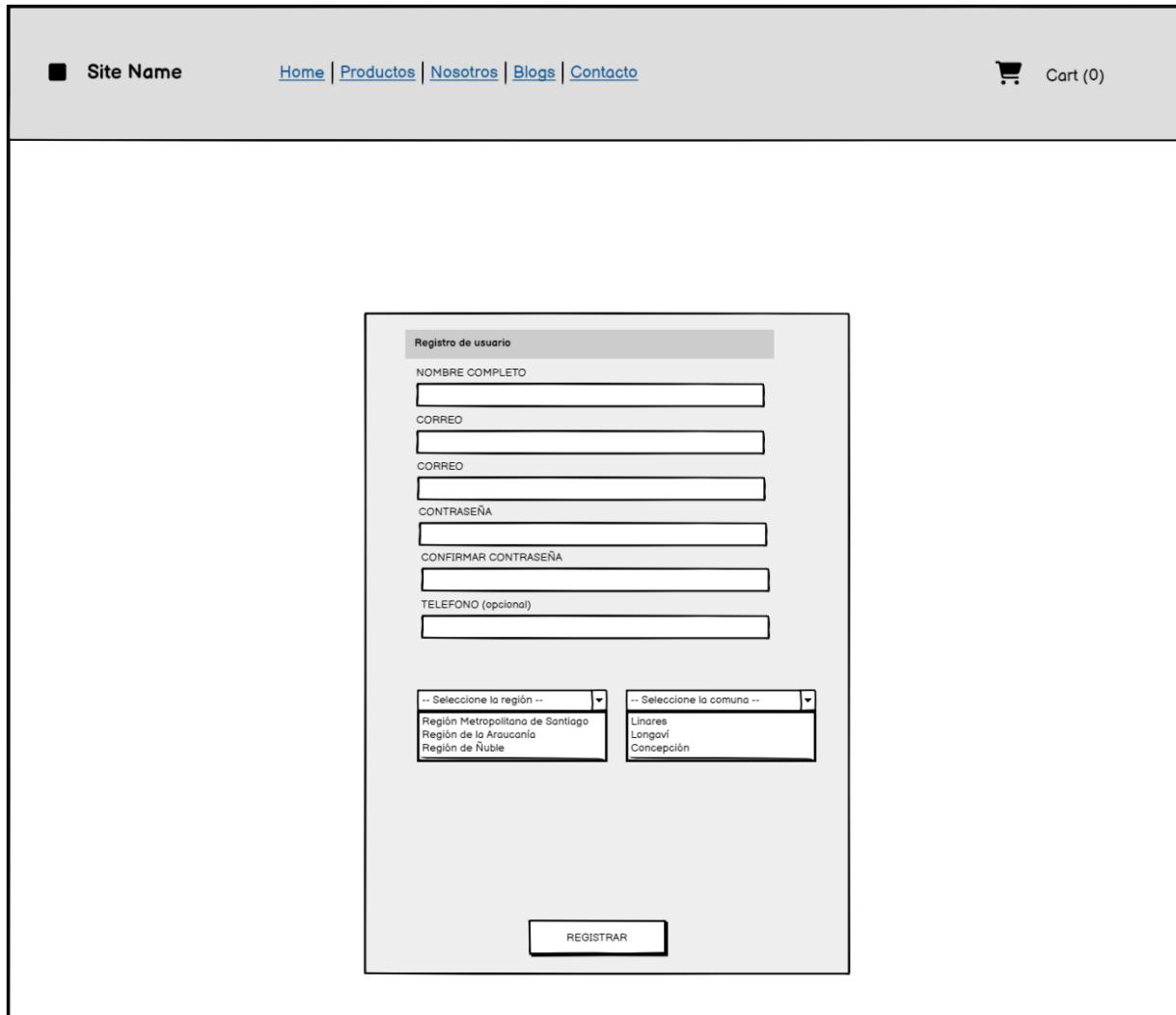
  

Stay In Touch! Join our Newsletter.

- **Registro de usuario:** Diseño de la vista de registro
(La información y reglas de negocios se establecerán en la sección validación de JavaScript).

Figura 4

Mockup registro de usuario, propuesta entrega 1.



The mockup shows a user registration form integrated into a website header. The header includes a 'Site Name' icon, navigation links for Home, Productos, Nosotros, Blogs, and Contacto, and a shopping cart icon indicating 0 items.

The registration form itself has a light gray background and a white input area. It features a title 'Registro de usuario' at the top. Below it are six input fields labeled 'NOMBRE COMPLETO', 'CORREO', 'CONTRASEÑA', 'CONFIRMAR CONTRASEÑA', and 'TELEFONO (opcional)'. Each field has a corresponding label above it and a horizontal input box below it.

At the bottom of the form are two dropdown menus: one for selecting a region ('-- Seleccione la región --') with options like 'Región Metropolitana de Santiago', 'Región de la Araucanía', and 'Región de Nuble'; and another for selecting a commune ('-- Seleccione la comuna --') with options like 'Linares', 'Longaví', and 'Concepción'.

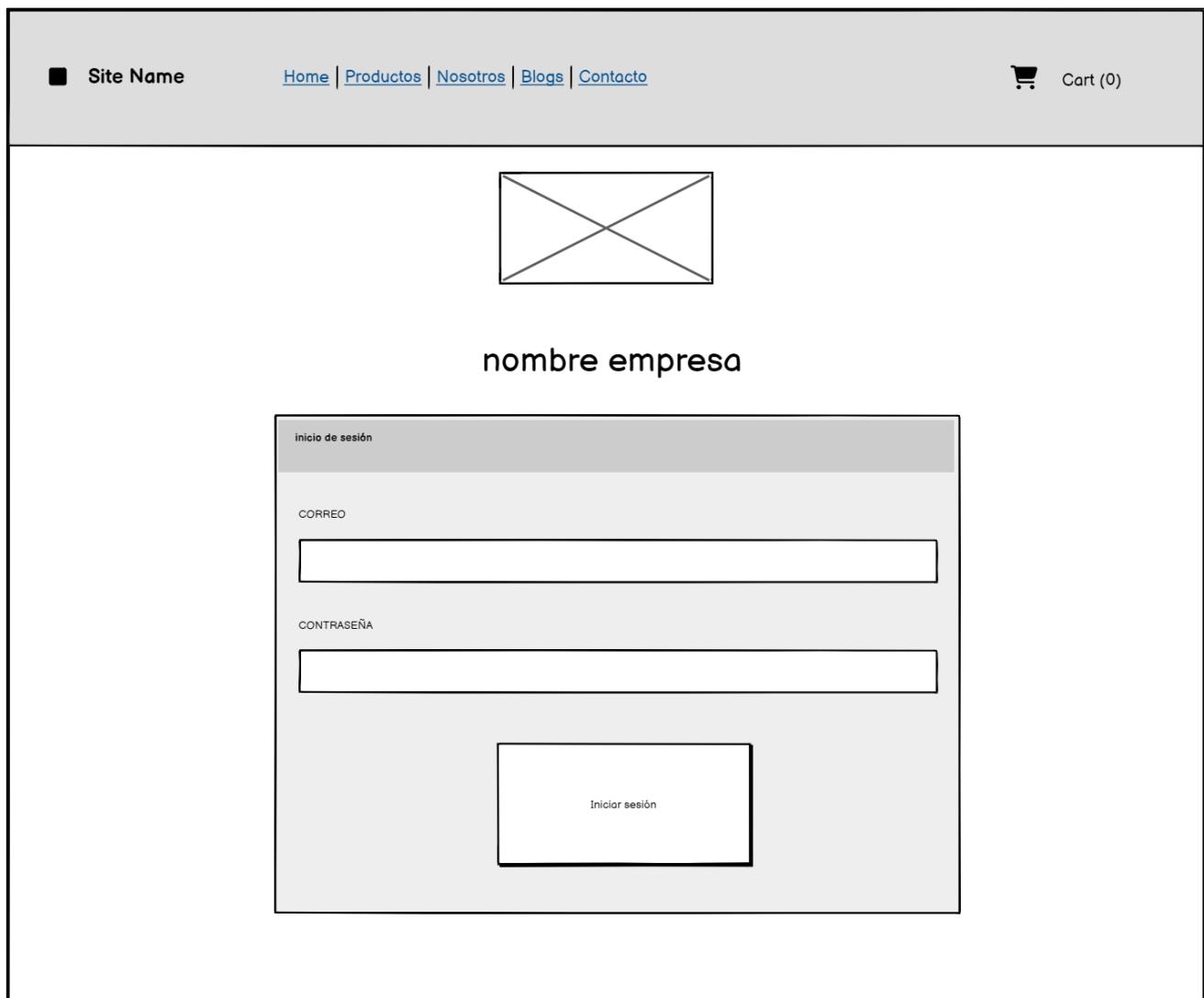
A single 'REGISTRAR' button is located at the bottom center of the form.

- **Inicio de sesión:** Diseño de la vista de ingreso del usuario.
 - Agregar logo de la empresa y nombre.
 - Formulario de inicio de sesión.

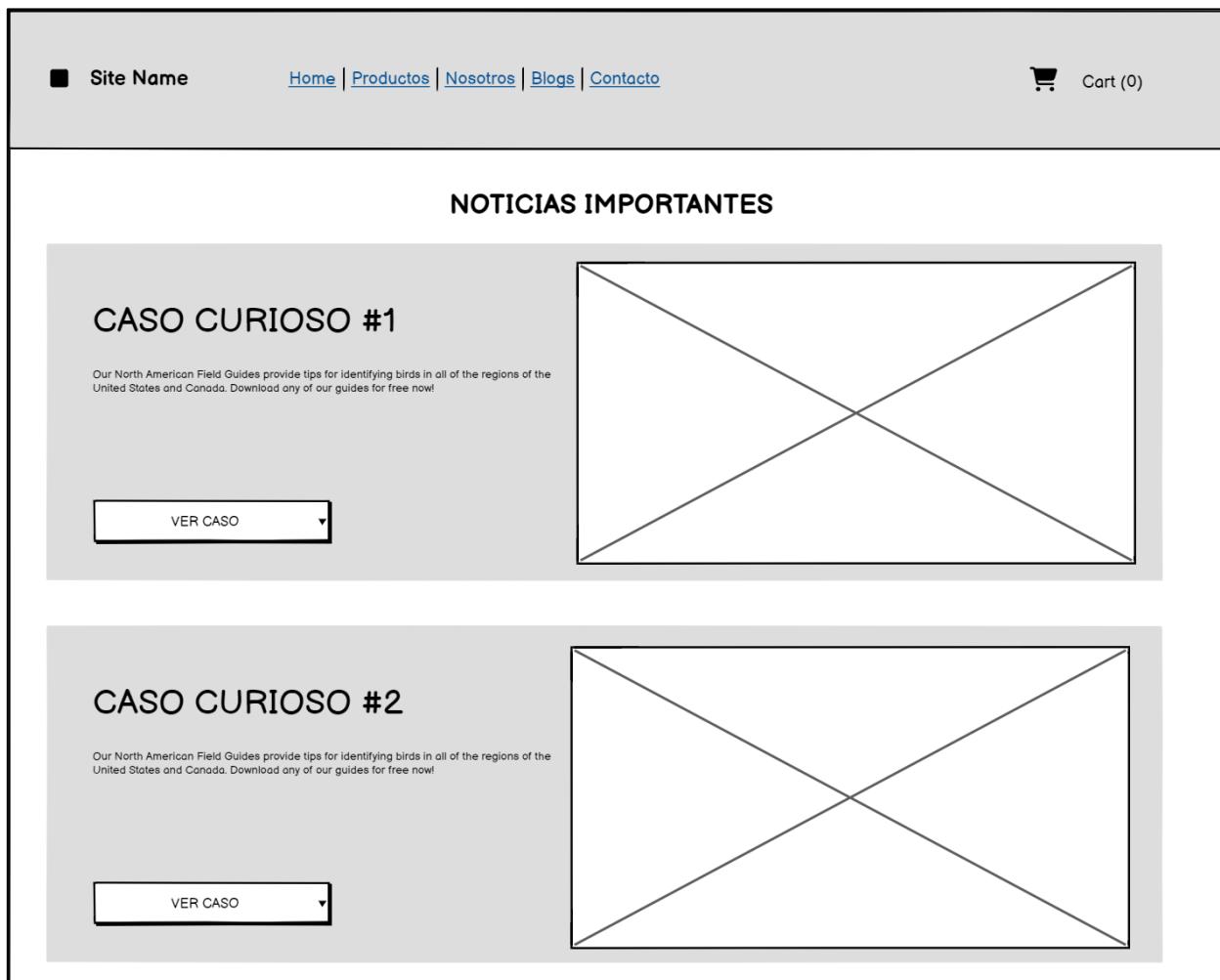
(La información y reglas de negocios se establecerán en la sección validación de JavaScript).

Figura 5

Mockup inicio de sesión (login), propuesta entrega 1.



- **Nosotros:** Una página web en donde explican de que se trata la empresa, su y los desarrolladores.
- **Blogs:** Una vista donde se muestra las noticias referentes a la tienda, como datos curiosos (*contenido del componente: Imagen, título, descripción corta*).
 - **Detalle blogs:** Crea 2 detalles de blogs relacionados a la tienda (*Contenido del componente: Imagen, título, descripción larga – Libre elección*).

Figura 6*Mockup página blogs, propuesta entrega 1.*

The mockup displays a website layout for a blog section. At the top, there is a header bar with a 'Site Name' placeholder, navigation links for Home, Productos, Nosotros, Blogs, and Contacto, and a shopping cart icon indicating 'Cart (0)'. Below the header, the title 'NOTICIAS IMPORTANTES' is centered. Two news items are listed, each enclosed in a gray box:

CASO CURIOSO #1
Our North American Field Guides provide tips for identifying birds in all of the regions of the United States and Canada. Download any of our guides for free now!
[VER CASO](#)

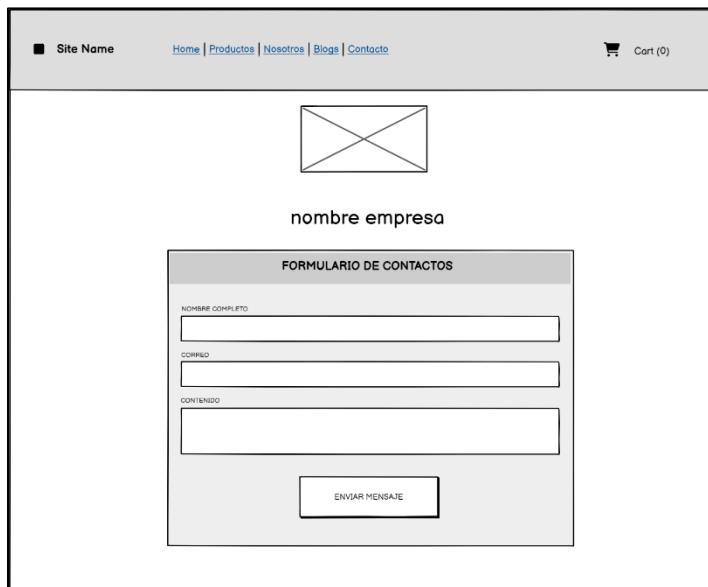
CASO CURIOSO #2
Our North American Field Guides provide tips for identifying birds in all of the regions of the United States and Canada. Download any of our guides for free now!
[VER CASO](#)

Each news item box contains a large 'X' mark in the bottom right corner, likely indicating a placeholder or a redacted area.

- **Contacto:** Vista para que los usuarios puedan enviar mensajes de manera interna.
(La información y reglas de negocios se establecerán en la sección validación de JavaScript).

Figura 7

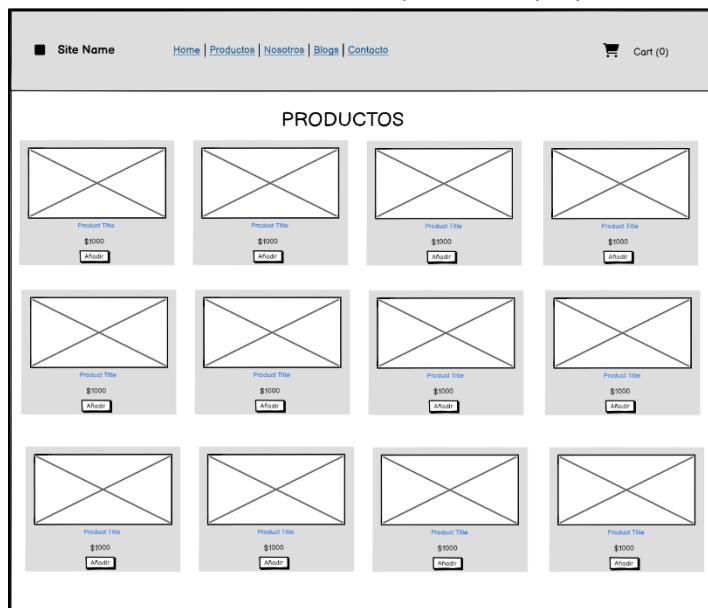
Mockup página de contacto, propuesta entrega 1.



- **Productos:** Lista todos los productos en ventas.
 - Desarrollo de vista de productos (imagen, nombre, precio y botón añadir)
 - Hacer clic en el algún producto nos redireccionara al detalle del producto.
 - Esta ventana tiene un carrito que funciona en esta vista de productos y en el destalle del producto.

Figura 8

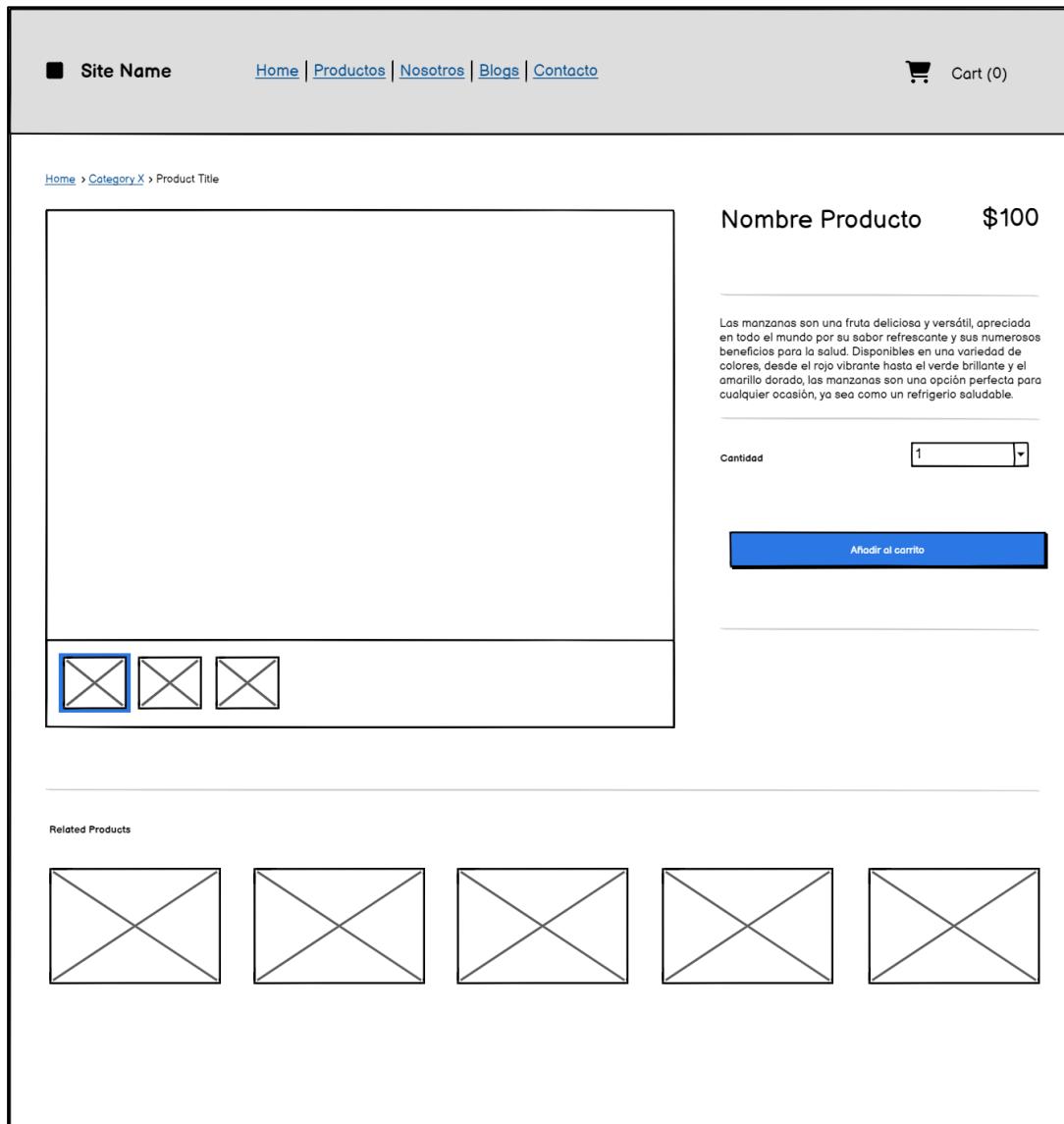
Mockup página productos, muestra un listado de los productos, propuesta entrega 1.



- **Detalle del producto:** Detalla el contenido de cada producto y se puede añadir al carrito de compra.

Figura 9

Mockup detalle de un producto, botón para añadir al carrito de compra, propuesta entrega 1.

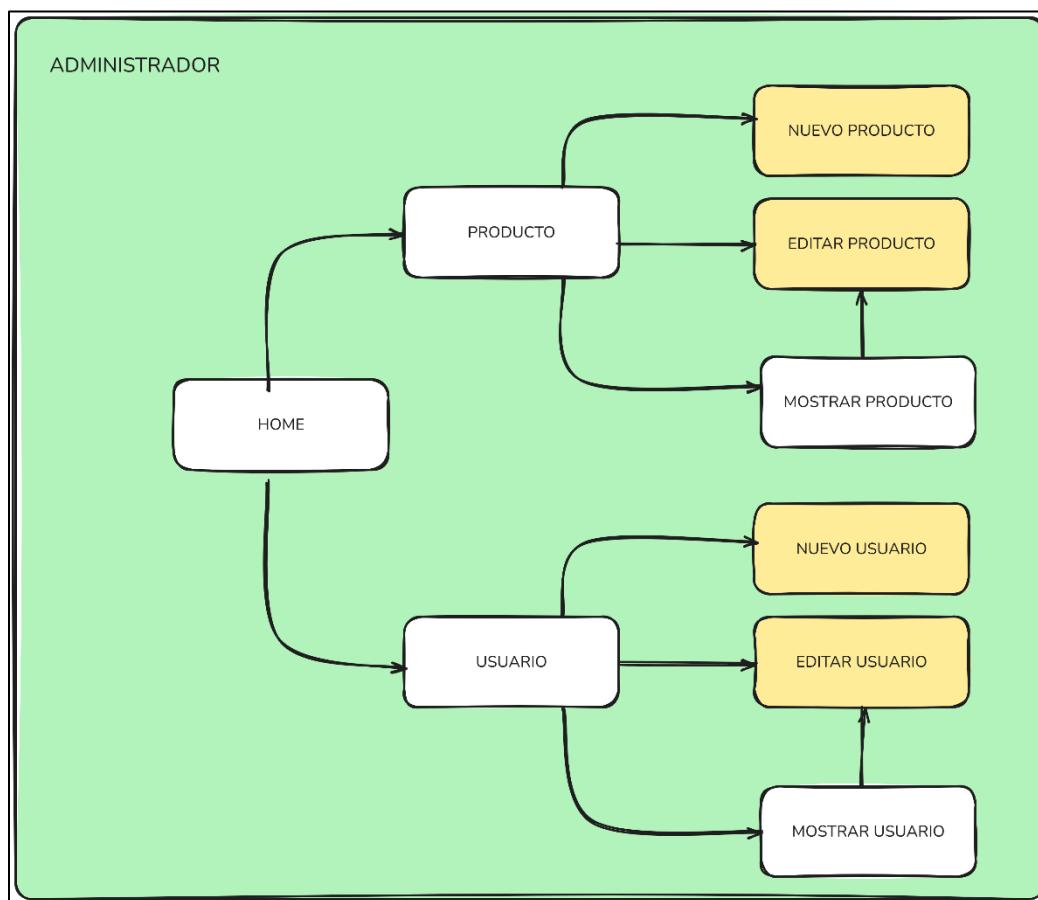


❖ Vista administrador

Todo sistema web necesita un sistema de gestión para configurar y controlar lo que muestra cada tienda, incluyendo el listado de productos disponibles. Este sistema de gestión permite actualizar, añadir, eliminar y organizar productos, asegurando que la información mostrada a los usuarios sea precisa y esté actualizada. Estas vistas siempre están protegidas mediante un sistema de autenticación y de permisos.

Figura 10

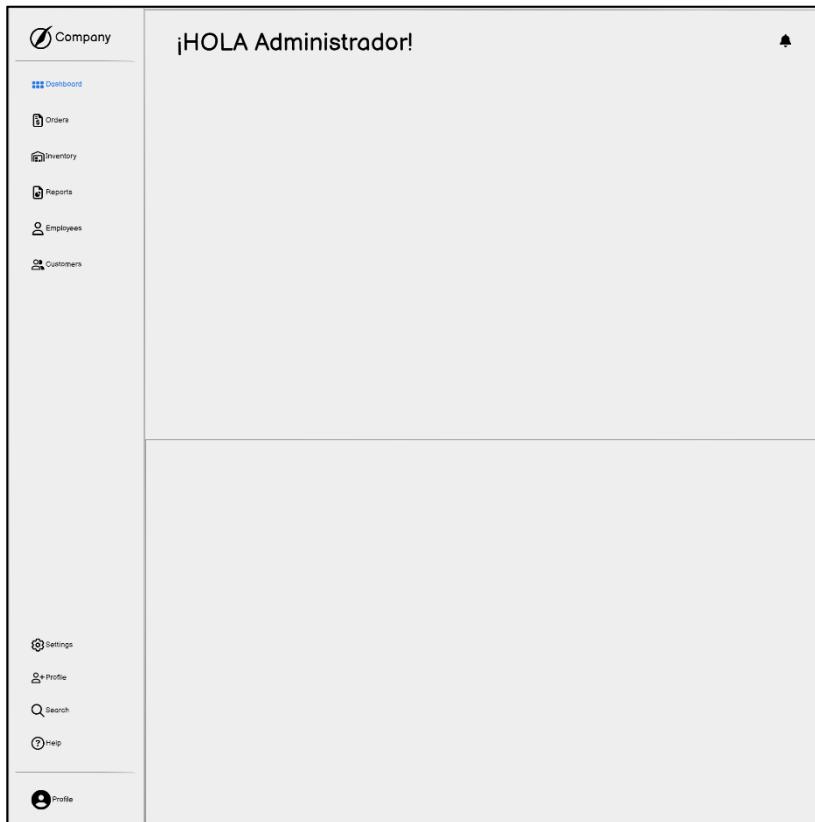
Diagrama de flujo de navegación del administrador, propuesta entrega 1.



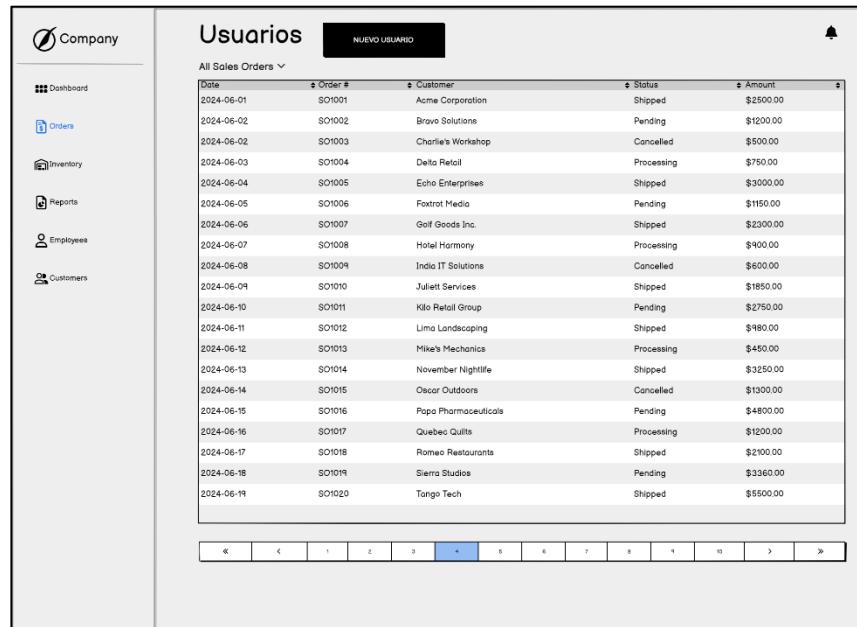
- **Home principal:** Vista con menú vertical y contenido al centro.
 - El diseño del administrador es libre, pero debe tener un menú visible.

Figura 11

Mockup página home del administrador, propuesta entrega 1.



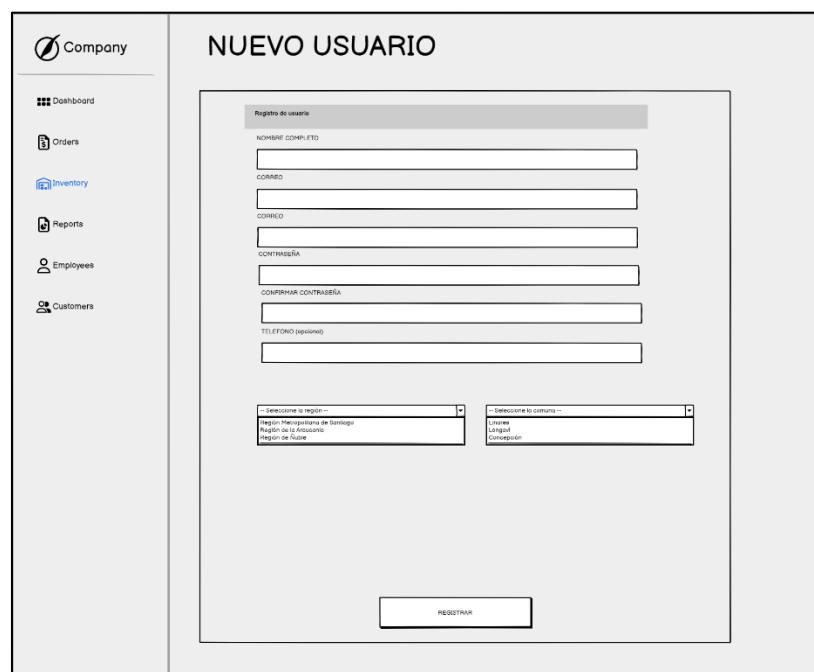
- **Producto o Usuario:** Crear un mantenedor para producto y usuario con la siguiente característica.
 - **Listado** de todos los Productos o Usuarios:

Figura 12
Mockup página listado de usuarios, propuesta entrega 1.


The screenshot shows a web application interface for managing users. On the left is a sidebar with navigation links: Dashboard, Orders, Inventory, Reports, Employees, and Customers. The main content area is titled "Usuarios" and contains a table of user data. The table has columns for Date, Order #, Customer, Status, and Amount. The data shows 20 entries from June 1st to June 19th, 2024. At the top right of the table is a button labeled "NUEVO USUARIO". Below the table is a navigation bar with icons for back, forward, and search.

Date	Order #	Customer	Status	Amount
2024-06-01	SO1001	Acme Corporation	Shipped	\$2500.00
2024-06-02	SO1002	Bravo Solutions	Pending	\$1200.00
2024-06-02	SO1003	Charlie's Workshop	Cancelled	\$500.00
2024-06-03	SO1004	Delta Retail	Processing	\$750.00
2024-06-04	SO1005	Echo Enterprise	Shipped	\$3000.00
2024-06-05	SO1006	Foxtrot Media	Pending	\$1150.00
2024-06-06	SO1007	Golf Goods Inc.	Shipped	\$2300.00
2024-06-07	SO1008	Hotel Harmony	Processing	\$400.00
2024-06-08	SO1009	India IT Solutions	Cancelled	\$600.00
2024-06-09	SO1010	Juliett Services	Shipped	\$1850.00
2024-06-10	SO1011	Kilo Retail Group	Pending	\$2750.00
2024-06-11	SO1012	Lima Landscaping	Shipped	\$180.00
2024-06-12	SO1013	Mike's Mechanics	Processing	\$450.00
2024-06-13	SO1014	November Nightlife	Shipped	\$3250.00
2024-06-14	SO1015	Oscar Outdoors	Cancelled	\$1900.00
2024-06-15	SO1016	Popa Pharmaceuticals	Pending	\$4800.00
2024-06-16	SO1017	Quebec Quilts	Processing	\$1200.00
2024-06-17	SO1018	Romeo Restaurants	Shipped	\$2100.00
2024-06-18	SO1019	Sierra Studios	Pending	\$3360.00
2024-06-19	SO1020	Tango Tech	Shipped	\$500.00

- **Crear un nuevo Producto o Usuario:**

Figura 13
Mockup página nuevo usuario en el administrador, propuesta entrega 1.


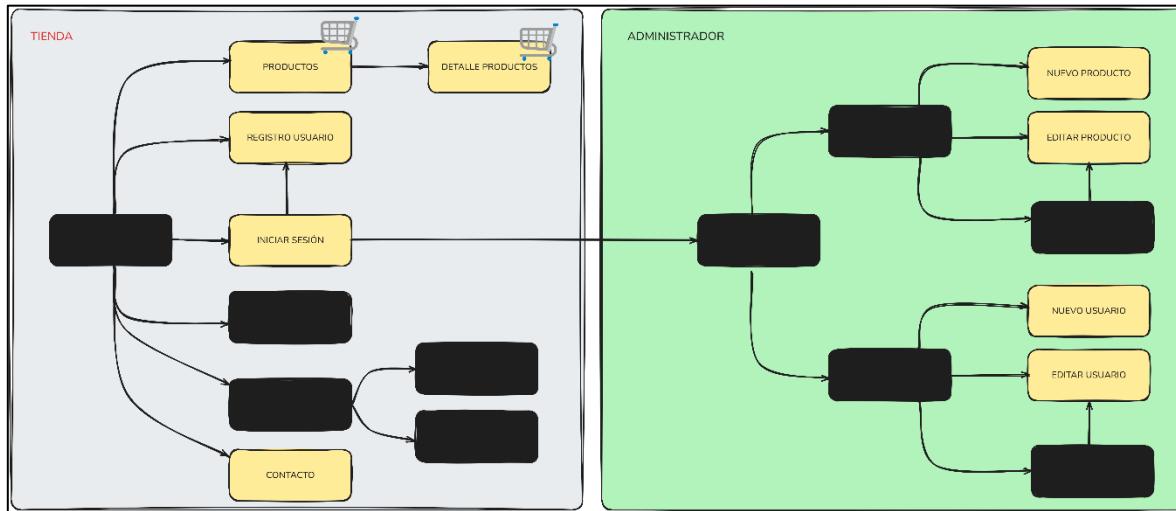
The screenshot shows a registration form titled "NUEVO USUARIO". The form is divided into sections: "Registro de usuario" (User Registration) and "Información adicional" (Additional Information). The "Registro de usuario" section contains fields for Nombre Completo, Correo, Contraseña, Confirmar Contraseña, and Teléfono (optional). The "Información adicional" section includes dropdown menus for "Seleccione la región" (Region) and "Seleccione la comuna" (Comuna), both showing options like "Región Metropolitana de Santiago", "Región de Coquimbo", "Región de Aysén", and "Región de Magallanes". At the bottom is a "REGISTRAR" (Register) button.

Utilización y validación de JavaScript:

En esta fase del proyecto, se definirán las reglas de negocio en las vistas desarrolladas anteriormente. Las vistas que están marcadas en color amarillo indican los lugares donde se deben implementar validaciones utilizando JavaScript. A continuación, se detallarán las validaciones que se harán al sistema como requerimientos del cliente, se permite la adición de reglas y atributos adicionales según sea necesario. Es importante que el proceso de implementación y las decisiones tomadas se documenten de manera detallada, ya que luego se usarán para crear la base de datos.

Figura 14

Diagrama de flujo de navegación del proyecto, donde se destaca las vistas que tendrán formularios con validación en JavaScript, propuesta entrega 1.



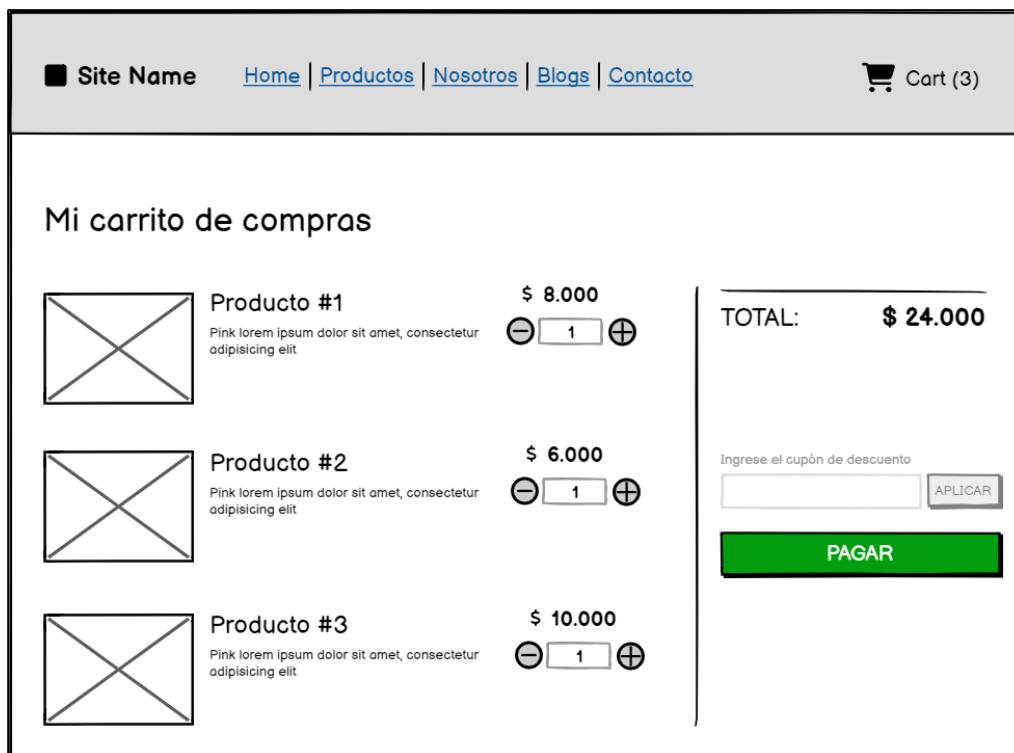
❖ Vista de la tienda

- Registro de usuarios:
 - *El registro de usuarios es lo mismo que el crear un usuario en el administrador (ver requerimientos vista administrador)*
- Inicio de sesión:
 - Correo:
 - Requerido
 - Max: 100 caracteres
 - Solo correos con @duoc.cl, @profesor.duoc.cl y @gmail.com
 - Contraseña:
 - Requerido
 - Entre 4 a 10 caracteres
- Contacto:
 - Nombre:
 - Requerido
 - Max: 100 caracteres
 - Correo:
 - Max: 100 caracteres
 - Solo correos con @duoc.cl, @profesor.duoc.cl y @gmail.com

- Comentario:
 - Requerido
 - Max: 500 caracteres
- **Productos, Detalle del producto (Carrito de compras):**
 - Listar productos mediante JavaScript.
 - Crear un arreglo de productos
 - Mostrar los productos del arreglo
 - Carrito de compra: Implementar un carrito de comprar
 - Añadir un producto al carrito de compras
 - Investigar y definir reglas del carrito de compra mediante lógicas
 - Guardar información del carrito en *LOCALSTORAGE*.
<https://developer.mozilla.org/es/docs/Web/API/Window/localStorage>

Figura 15

Mockup página carrito de compras, propuesta entrega 1.



❖ Vista del administrador

- **Producto**
 - **Nuevo Producto o Editar producto**
 - Código producto:
 - Requerido.
 - Texto
 - Min: 3
 - Max: no tiene límite

- Nombre:
 - Requerido.
 - Max: 100
- Descripción:
 - Opcional.
 - Max: 500
- Precio:
 - Requerido
 - Min: 0 (*se considerará un producto FREE*)
 - Max: No tiene limitaciones
 - Puede ser números decimales
- Stock: Cantidad de productos en el inventario
 - Requerido
 - Min: 0
 - Max: No tiene limitaciones
 - Solo números enteros
- Stock Critico: Mostrar un mensaje de alerta cuando el stock sea igual o inferior.
 - Opcional
 - Min: 0
 - Solo numeros enteros
- Categorías:
 - Requerido
 - Select que muestra la categoría del producto.
- Imagen:
 - Opcional
- Usuario
 - Run:
 - Requerido
 - Validar si el run está correcto
 - Sin puntos ni guion, Ej: 19011022K
 - Min: 7
 - Max: 9
 - Nombre:
 - Requerido
 - Max: 50 caracteres
 - Apellidos:
 - Requerido
 - Max: 100 caracteres
 - Correo:
 - Requerido
 - Max: 100 caracteres
 - Validar Solo correos con @duoc.cl, @profesor.duoc.cl y @gmail.com
 - Fecha Nacimiento:

- Opcional
- Tipo de Usuario: (*implementar solo para la vista administrativa y ver roles asociados al sistema*)
 - Select que muestra el perfil del usuario.
 - Administrador, Cliente y Vendedor.
- Región y Comuna:
 - Mostrar las regiones que están en un arreglo de JS complementario a este entregable.
 - Al momento de cambiar una región, también cambiaria la búsqueda de las comunas.
- Dirección:
 - Requerido.
 - Max: 300 caracteres.

Roles asociados al sistema:

- **Administrador:** tiene acceso total al sistema.
- **Vendedor:**
 - Puede visualizar la lista de los productos y el detalle.
 - Puede visualizar la lista de órdenes y el detalle.
 - Todos los demás accesos no deben aparecer en la vista del vendedor.
- **Cliente:** Solo puede acceder a la tienda.

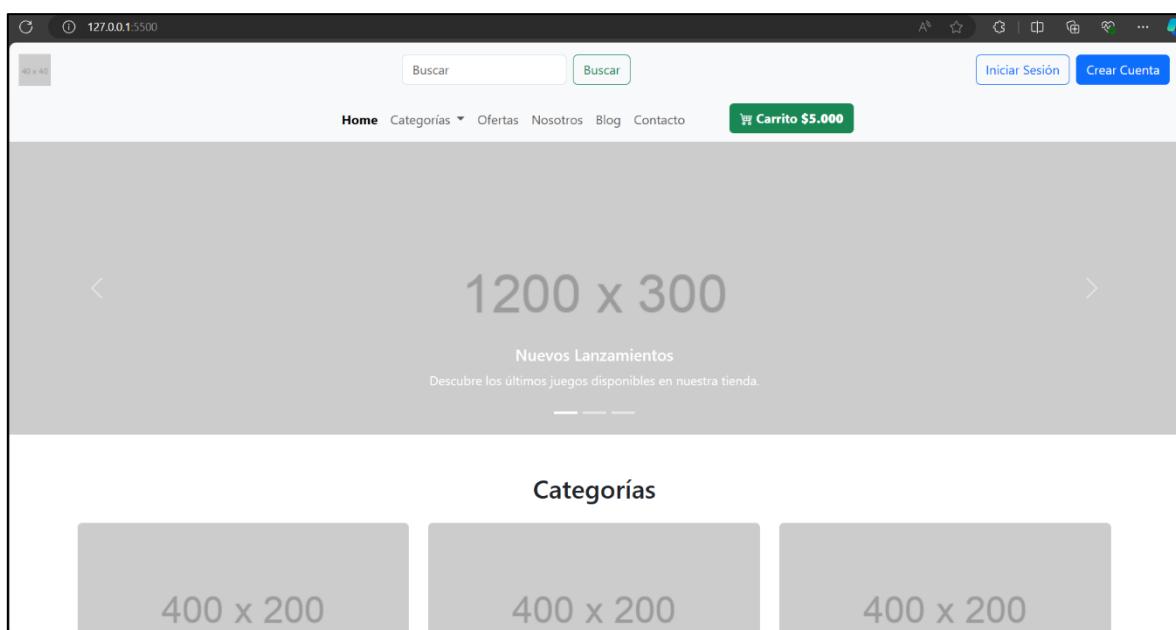
Instrucciones para el desarrollo de la Evaluación 2 (30%)

Objetivos para la evaluación 2:

En esta evaluación, el equipo continuará desarrollando su tienda online con un enfoque en el frontend, utilizando React y añadiendo nuevos flujos de navegación. Se proporcionará un template en HTML como guía para ayudar a los estudiantes a tomar ideas y actualizar sus proyectos con el framework y por último deberán realizar pruebas para asegurar la calidad y el correcto funcionamiento de los componentes.

Figura 1

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



Requisitos del proyecto:

- ❖ **Migración de HTML a React:**
 - Convertir las páginas HTML existentes en componentes de React reutilizables.
 - Asegurarse de que cada componente maneje su propio estado y propiedades de manera eficiente.
- ❖ **Archivo de datos en JavaScript:**
 - Crear un archivo JavaScript que actúe como una fuente de datos simulada (similar a una base de datos) para la aplicación.
 - Implementar funciones para CRUD (Crear, Leer, Actualizar, Eliminar) operaciones sobre los datos dentro del archivo JS.
- ❖ **Integración de persistencia:**
 - Conectar el archivo de persistencia con los componentes de React para gestionar datos en tiempo real y actualizar la interfaz de usuario según sea necesario.

- ❖ **Diseño responsive utilizando Bootstrap:**
 - Incorporar Bootstrap para asegurar que el diseño sea responsive para proporcionar una experiencia de usuario óptima en móviles, tabletas y escritorios.
- ❖ **Vistas adicionales:**
 - Añadir nuevas vistas basadas en los requerimientos actuales del negocio o cliente, como paneles de administración o páginas de detalles de productos.
- ❖ **Interactividad mejorada:**
 - Incorporar características interactivas adicionales utilizando React, como formularios avanzados, filtros de búsqueda y navegación mejorada.
- ❖ **Configuración de pruebas:**
 - Configurar el entorno de pruebas usando Jasmine y Karma para ejecutar pruebas unitarias en componentes de React.
- ❖ **Desarrollo de pruebas unitarias:**
 - Escribir pruebas para verificar la lógica y el comportamiento de los componentes, asegurando que cada parte de la aplicación funcione como se espera.

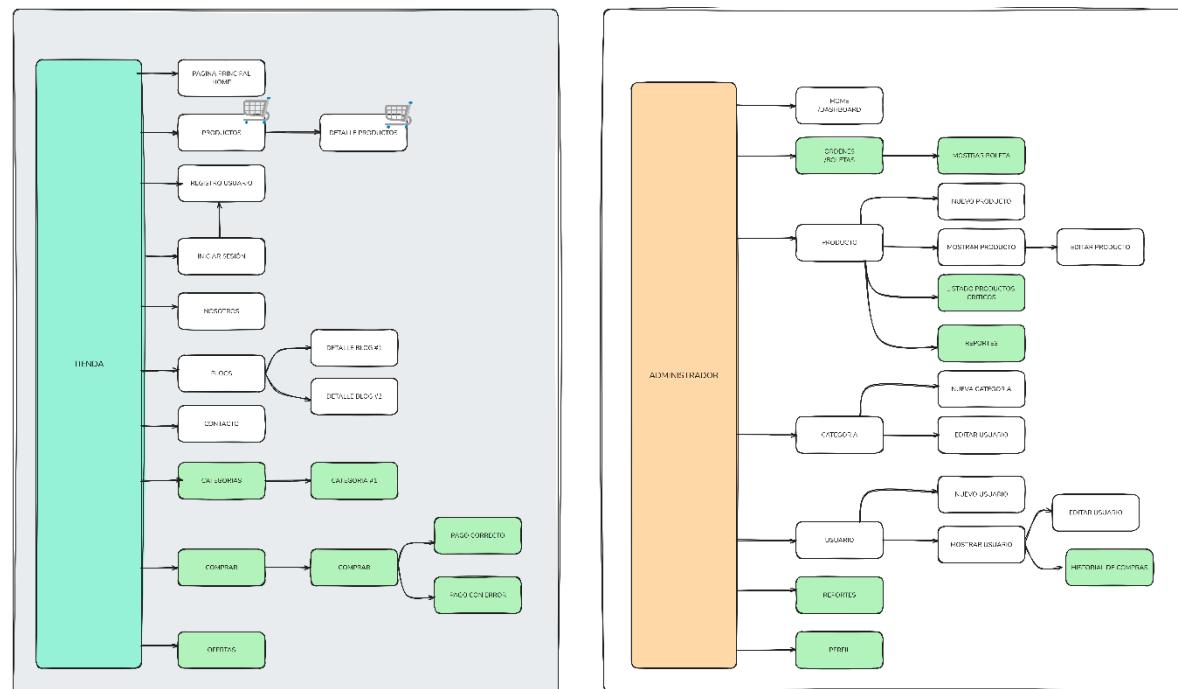
"Al crear un componente en React, sigue el principio de "Single Responsibility". Un componente debe tener una única responsabilidad o propósito en la aplicación. Esto no solo hace que el componente sea más fácil de entender y probar, sino que también mejora la reutilización. Asegúrate de que el componente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

Actualización del flujo entre páginas:

En la siguiente imagen representa el flujo de navegación entre páginas y todas las vistas que se deben desarrollar para la entrega 2.

Figura 2

Diagrama de flujo de navegación del proyecto, propuesta final.



Detalle del nuevo diseño web por cada vista:

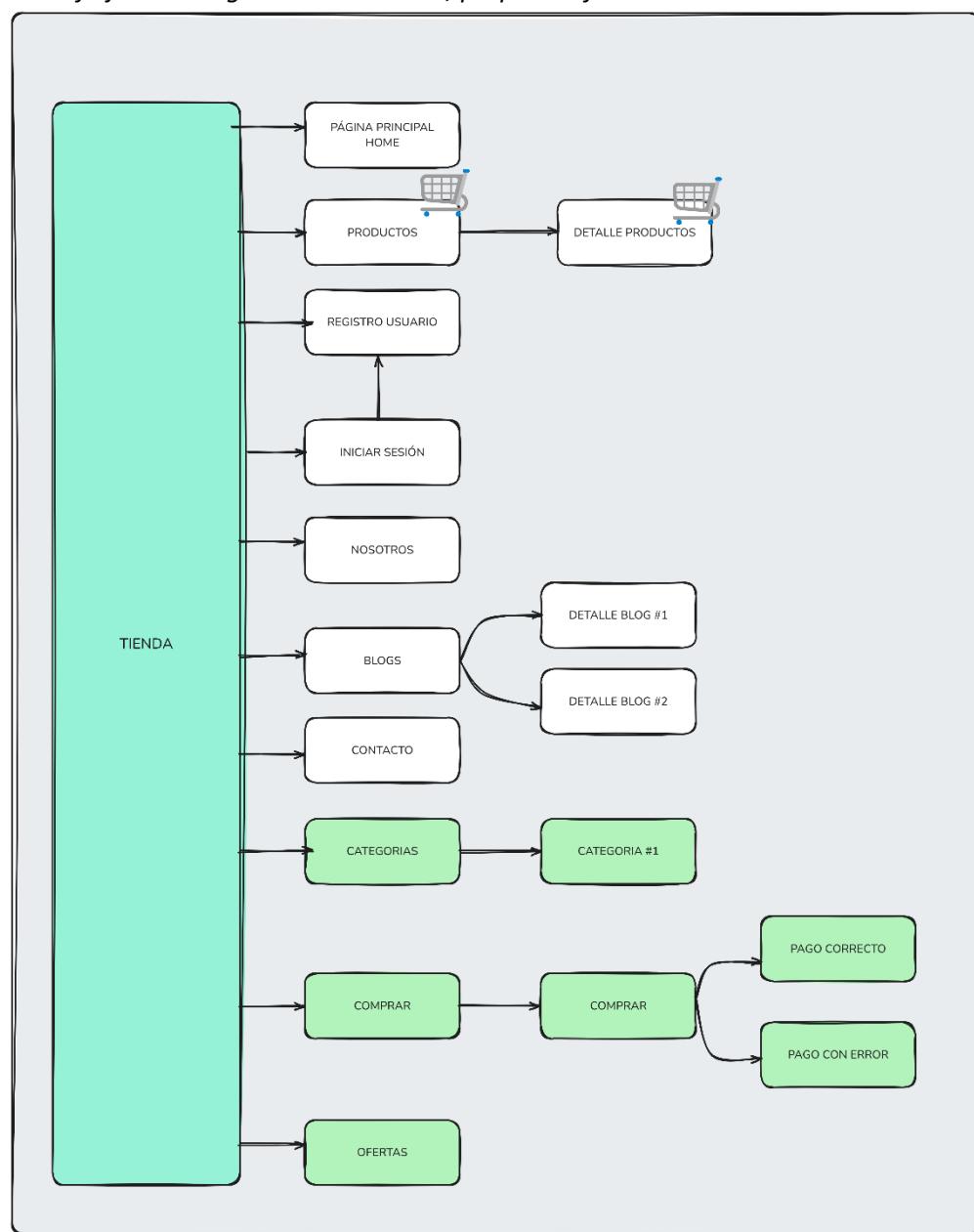
Los contenidos presentados en cada diseño son **sugerencias** para el desarrollo del proyecto.

❖ Vistas de la tienda

Esta vista es de manera pública y será la cara visible de la tienda.

Figura 3

Diagrama de flujo de navegación de la tienda, propuesta final.



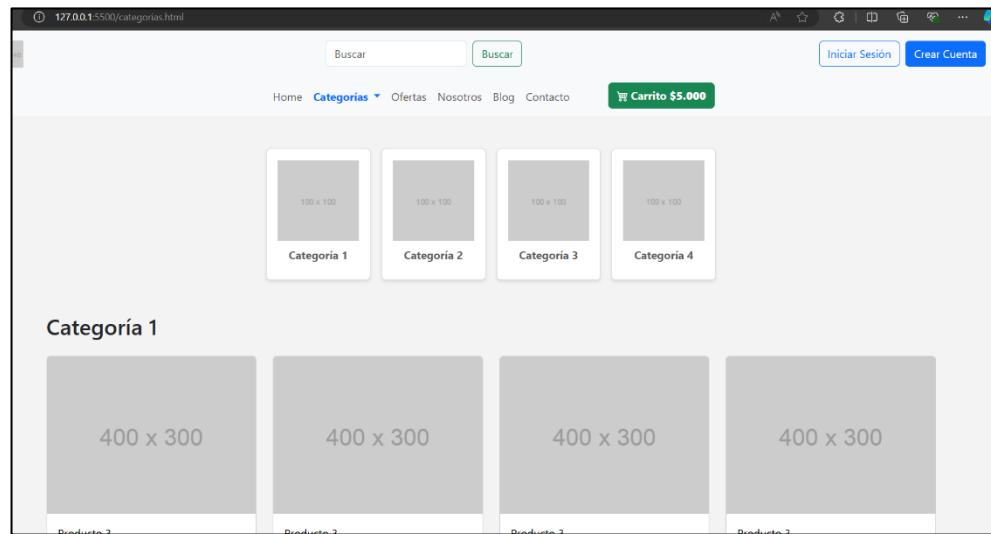
Se reorganizan la estructura de las páginas y se añaden otras que serán complementarias al desarrollo final de este proyecto.

Nuevas vistas:

- **Categorías:**
 - Vista que separa los productos por categorías

Figura 4

Diseño web propuesta vista categorías.



- **Compra:**

- Se incorpora **Checkout** (Paso de Pago): El cliente introduce la dirección de envío y selecciona opciones de entrega.
- Se incorporan nuevas vistas la compra fue exitosa o no y se genera un resumen.

Figura 5

Diseño web propuesta carrito de compra.

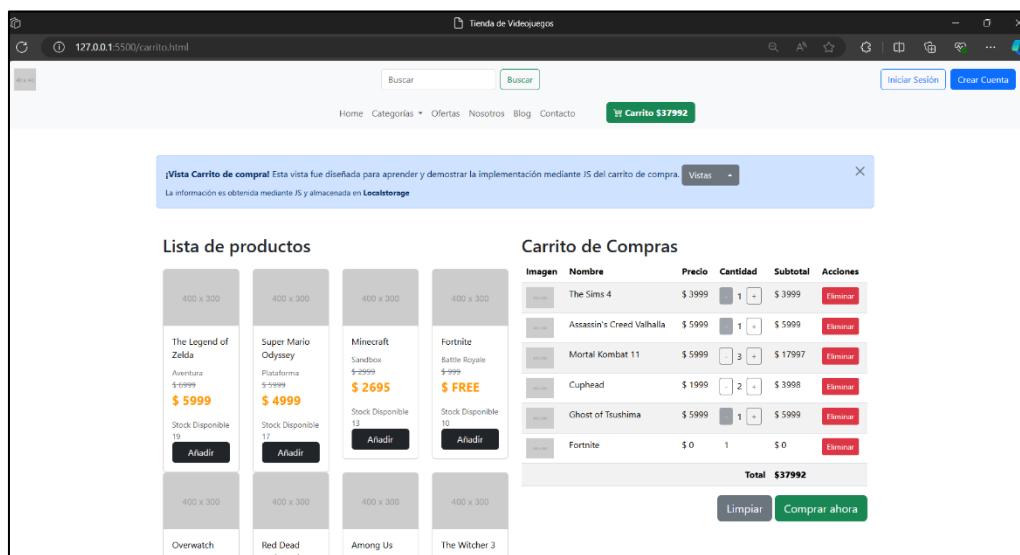


Figura 6

*Diseño web propuesta **checkout**, el cliente debe agregar sus datos y dirección de entrega.*

40 x 40

Home Categorías ▾ Ofertas Nosotros Blog Contacto

Carrito de compra

Completa la siguiente información

Total a pagar: \$ 28775

Imagen	Nombre	Precio	Cantidad	Subtotal
	Fortnite	\$ 0	1	\$ 0
	Minecraft	\$ 2695	4	\$ 10780
	Red Dead Redemption 2	\$ 5999	1	\$ 5999
	Among Us	\$ 499	1	\$ 499
	The Witcher 3	\$ 3999	1	\$ 3999
	Hollow Knight	\$ 1499	1	\$ 1499
	Animal Crossing	\$ 5999	1	\$ 5999

Información del cliente

Completa la siguiente información

Nombre* Apellidos*

Correo*

Dirección de entrega de los productos

Ingrésale dirección de forma detallada

Calle* Departamento (opcional)

Región* Comuna*

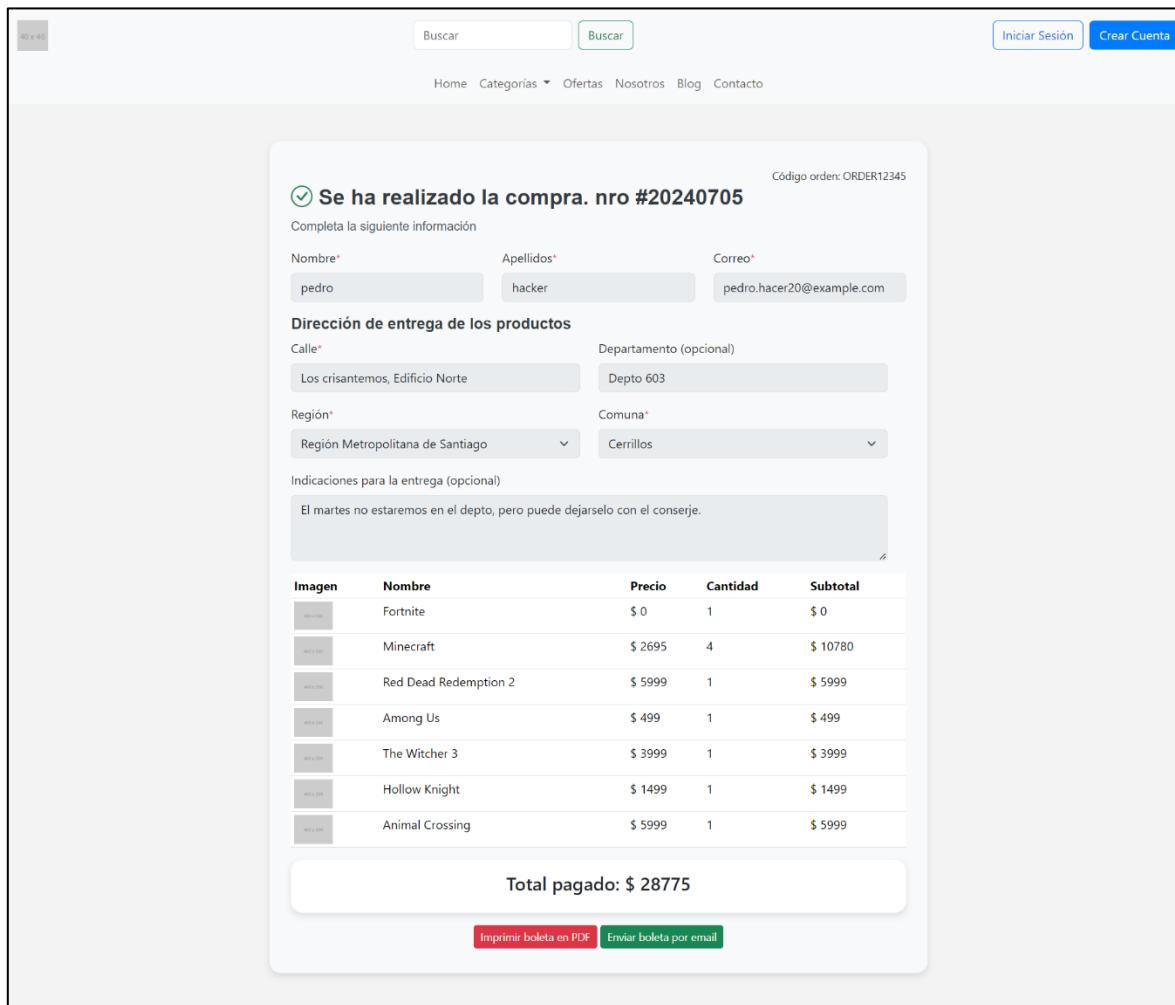
Indicaciones para la entrega (opcional)

Ej.: Entre calles, color del edificio, no tiene timbre.

Nota: Si el usuario ha iniciado sesión toda esta información se añadirá de forma automática.

Figura 7

Diseño web propuesta vista compra exitosa.



40 x 40

Buscar Buscar Iniciar Sesión Crear Cuenta

Home Categorías Ofertas Nosotros Blog Contacto

Código orden: ORDER12345

Se ha realizado la compra. nro #20240705

Completa la siguiente información

Nombre*	Apellidos*	Correo*
pedro	hacker	pedro.hacer20@example.com

Dirección de entrega de los productos

Calle*	Departamento (opcional)
Los crisanteros, Edificio Norte	Deptº 603

Región*	Comuna*
Región Metropolitana de Santiago	Cerrillos

Indicaciones para la entrega (opcional)

El martes no estaremos en el depto, pero puede dejarselo con el conserje.

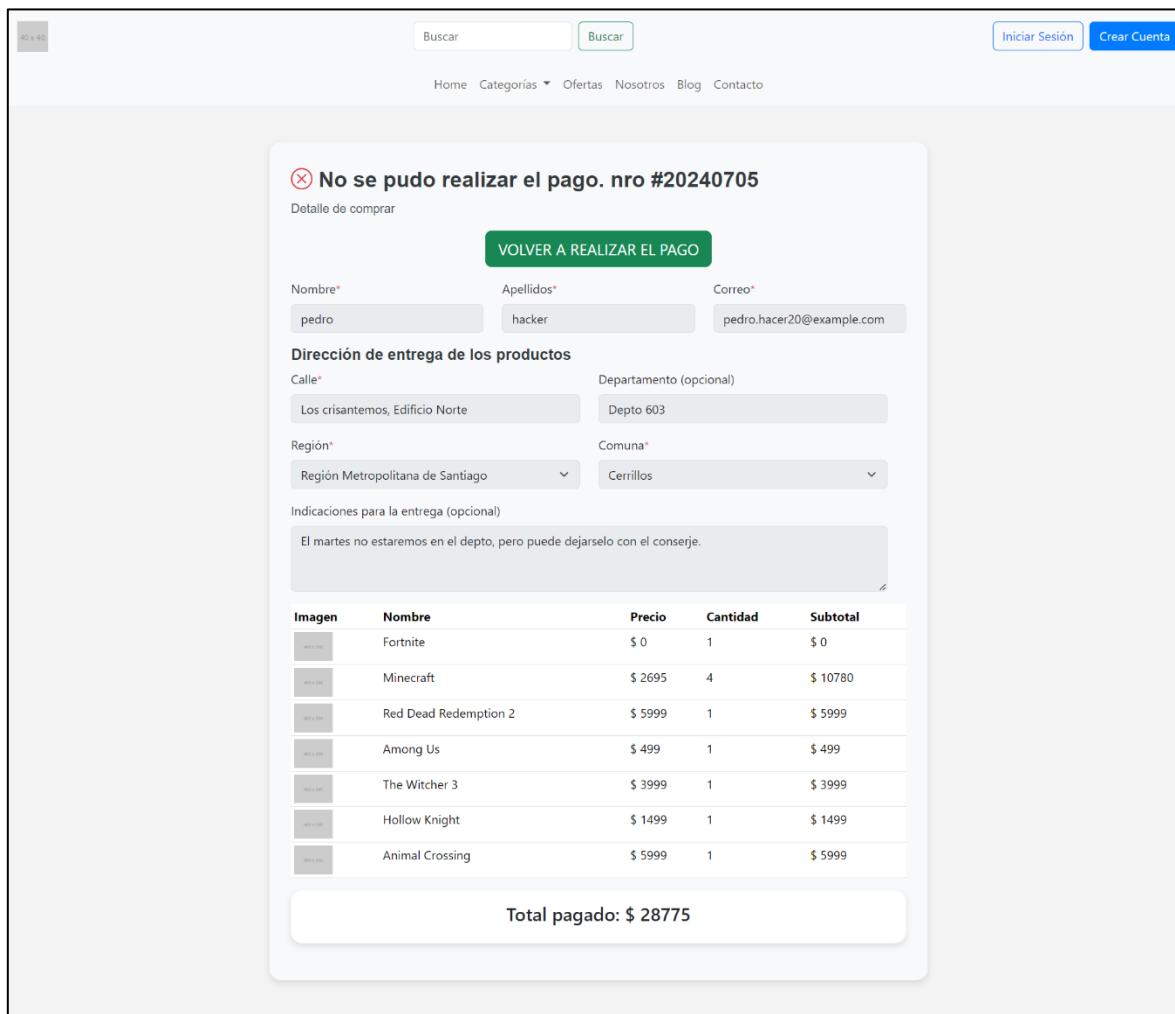
Imagen	Nombre	Precio	Cantidad	Subtotal
[Thumbnail]	Fortnite	\$ 0	1	\$ 0
[Thumbnail]	Minecraft	\$ 2695	4	\$ 10780
[Thumbnail]	Red Dead Redemption 2	\$ 5999	1	\$ 5999
[Thumbnail]	Among Us	\$ 499	1	\$ 499
[Thumbnail]	The Witcher 3	\$ 3999	1	\$ 3999
[Thumbnail]	Hollow Knight	\$ 1499	1	\$ 1499
[Thumbnail]	Animal Crossing	\$ 5999	1	\$ 5999

Total pagado: \$ 28775

Imprimir boleta en PDF Enviar boleta por email

Figura 8

Diseño web propuesta no se pudo realizar el pago.



The screenshot shows a web page with a header containing search bars, navigation links (Home, Categorías, Ofertas, Nosotros, Blog, Contacto), and user account buttons (Iniciar Sesión, Crear Cuenta). A central modal window displays an error message: **No se pudo realizar el pago. nro #20240705**. Below the message, it says "Detalle de compra". A green button labeled "VOLVER A REALIZAR EL PAGO" is visible. Form fields for Name*, Surname*, and Email* contain the values "pedro", "hacker", and "pedro.hacer20@example.com" respectively. Delivery address fields show "Los crísanteros, Edificio Norte" and "Dept 603". Region and Commune dropdowns show "Región Metropolitana de Santiago" and "Cerrillos". An optional note for delivery indicates "El martes no estaremos en el depto, pero puede dejarselo con el conserje." Below this is a table of purchased items:

Imagen	Nombre	Precio	Cantidad	Subtotal
	Fortnite	\$ 0	1	\$ 0
	Minecraft	\$ 2695	4	\$ 10780
	Red Dead Redemption 2	\$ 5999	1	\$ 5999
	Among Us	\$ 499	1	\$ 499
	The Witcher 3	\$ 3999	1	\$ 3999
	Hollow Knight	\$ 1499	1	\$ 1499
	Animal Crossing	\$ 5999	1	\$ 5999

A summary box at the bottom right states "Total pagado: \$ 28775".

- **Ofertas:**

- Vista que muestra todos los productos en ofertas.

❖ Vista del administrador

El desarrollo del sistema administrativo será un proceso colaborativo, donde el docente desempeñará un papel fundamental al proponer y guiar el flujo de trabajo que los estudiantes deben seguir. La propuesta presentada servirá como una hoja de ruta inicial, proporcionando un marco estructurado y destacando los pasos esenciales que los estudiantes deben considerar. Esto permitirá a los estudiantes adaptar y personalizar el proyecto, experimentar con diferentes estrategias y aplicar sus habilidades de resolución de problemas en un contexto práctico. Al seguir esta orientación, los estudiantes tendrán la oportunidad de fortalecer sus competencias técnicas y de gestión de proyectos, fomentar la creatividad y la innovación, y desarrollar un sistema administrativo funcional y eficaz, enriqueciendo así su experiencia de aprendizaje.

Figura 9

Diseño web propuesta para el panel administrativo.

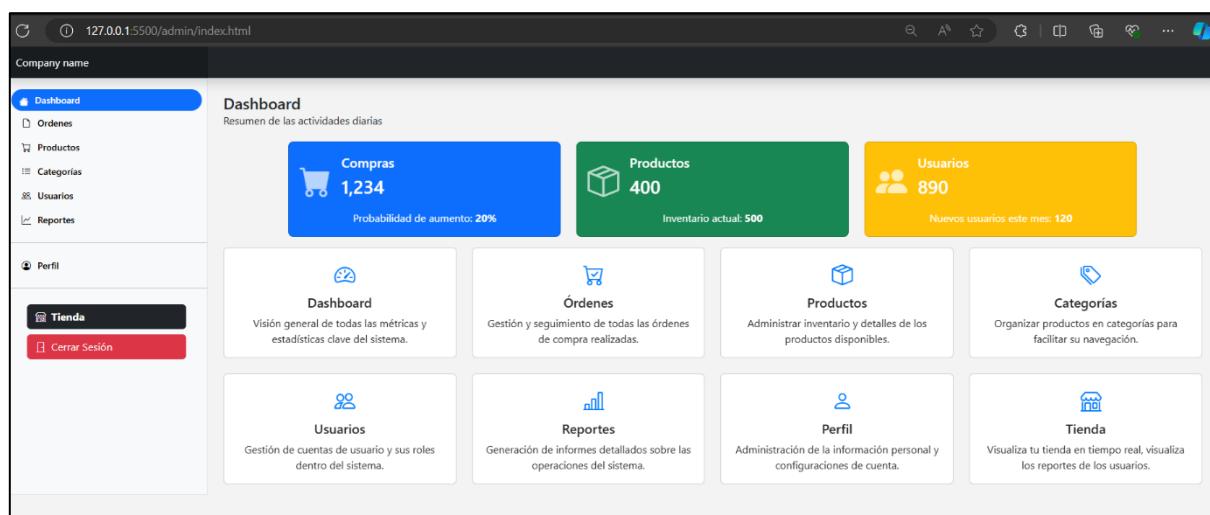
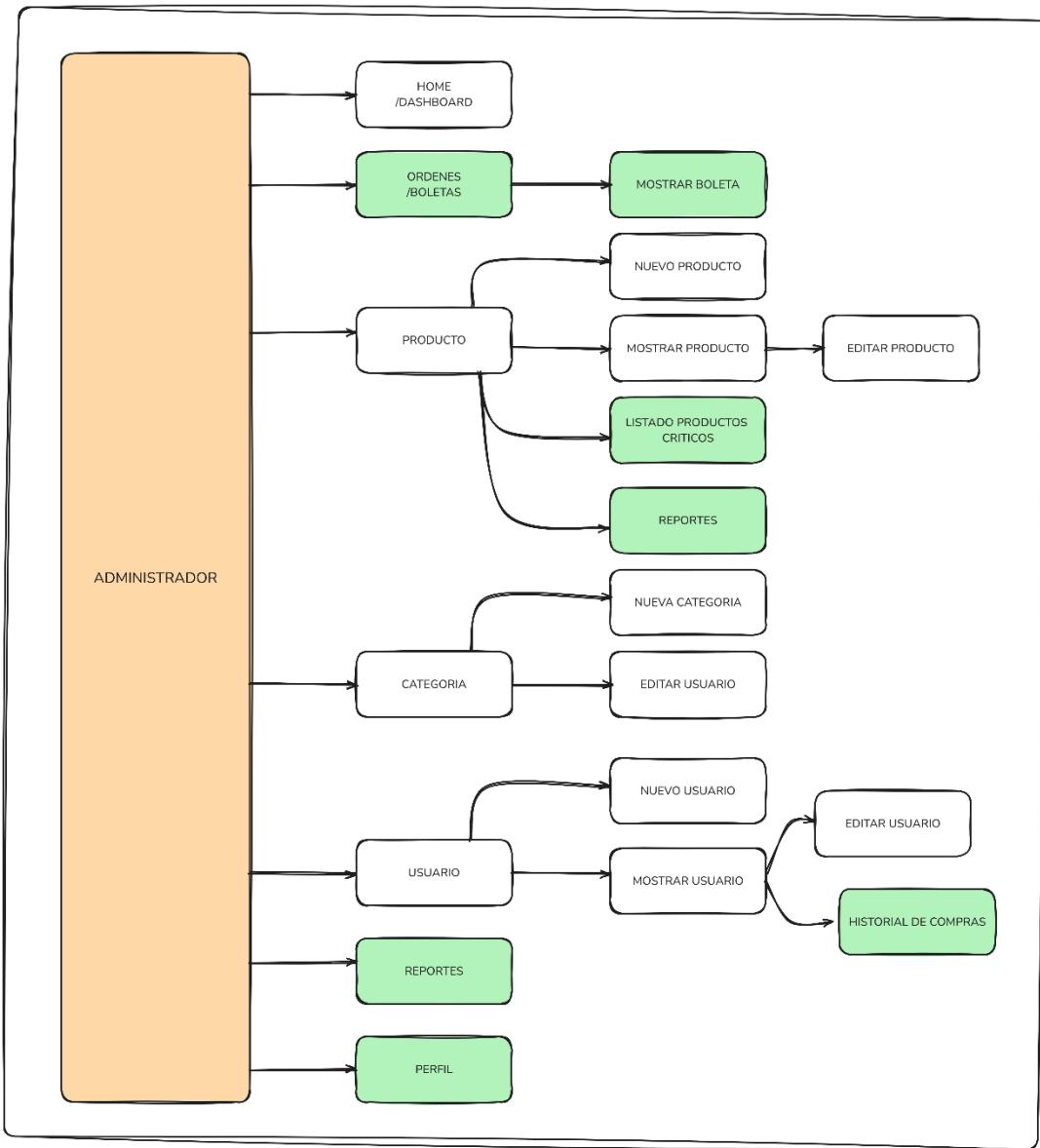


Figura 10

Diagrama de flujo de navegación del sistema administrativo, propuesta final.



Nota: Ver template HTML donde se detalla completamente el este sistema.

Propuesta de pruebas con Jasmine:

 *Este alcance es propuesto por el docente* 

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

❖ Pruebas de renderizado:

- **Renderizado correcto:** Escribe pruebas que verifiquen que los componentes se rendericen con los datos proporcionados.
 - **Tarea:** Asegúrate de que los componentes de lista rendericen todos los elementos de un conjunto de datos.
- **Renderizado condicional:** Crea pruebas para comprobar que los componentes muestren y/o oculten elementos según las condiciones especificadas.
 - **Tarea:** Verifica que un mensaje de error solo se muestre cuando hay un error presente.

❖ Pruebas de propiedades (props):

- **Propiedades Recibidas:** Desarrolla pruebas que confirmen que los componentes reciben y utilizan las propiedades adecuadas.
 - **Tarea:** Asegúrate de que un componente de botón recibe correctamente la etiqueta y la función de evento `onClick`.

❖ Pruebas de Estado (state):

- **Gestión del Estado:** Implementa pruebas que comprueben la lógica de cambios de estado dentro de los componentes.
 - **Tarea:** Verifica que el estado de un formulario cambia correctamente cuando el usuario introduce texto.

❖ Pruebas de Eventos:

- **Simulación de Eventos:** Simula eventos de usuario para verificar que los componentes reaccionen como se espera.
 - **Tarea:** Simula un clic en un botón y comprueba que el estado del componente cambie o que se ejecute una función específica.

Instrucciones para el desarrollo de la Evaluación 3 (40%)

Objetivo del Proyecto:

Desarrollar una aplicación web que integre un backend con Spring Boot y una base de datos, además de implementar la comunicación REST entre el frontend y el backend. Esta evaluación se centrará en la creación de lógica de negocio, la gestión de datos y la seguridad del acceso a la aplicación.

Pasos de Desarrollo:

- **Configuración del Proyecto:**
 - Crear un nuevo proyecto Spring Boot con las dependencias necesarias para la conexión a la base de datos y la seguridad.
 - Configurar una base de datos relacional (MySQL, Oracle cloud o Firebase) y establecer la conexión desde el backend.
- **Modelamiento de Datos:**
 - Definir las entidades y relaciones necesarias en la base de datos, basadas en los requerimientos del cliente.
 - Crear los repositorios y servicios en Spring Boot para interactuar con la base de datos.
- **Implementación de la Lógica de Negocio:**
 - Desarrollar la lógica de negocio necesaria para cumplir con los casos de uso definidos por el cliente.
 - Asegurar que la lógica de negocio esté correctamente encapsulada y testeada.
- **Desarrollo de API REST:**
 - Crear controladores REST en Spring Boot para manejar las operaciones CRUD.
 - Implementa buenas prácticas de API REST, añadiendo un versionado a las rutas de las APIs.
 - Configurar **Swagger** para documentar los endpoints de la API.
- **Integración con el Frontend:**
 - Asegurar que el frontend interactúe con los endpoints del backend a través de llamadas fetch o axios.
 - Validar que la comunicación REST sea efectiva y que los datos se manejen correctamente.
- **Implementación de Autenticación y Autorización:**
 - Configurar Spring Security para gestionar la autenticación de usuarios mediante roles.
 - Implementar JWT para la autenticación basada en tokens.
 - Restringir el acceso a funciones específicas según los permisos del usuario.
- **Colaboración y uso de repositorios:**
 - Configurar un repositorio remoto en GitHub para el proyecto backend.
 - Realizar cambios en el repositorio con comentarios claros y coherentes.
 - Colaborar efectivamente con los miembros del equipo, distribuyendo tareas de manera equitativa.

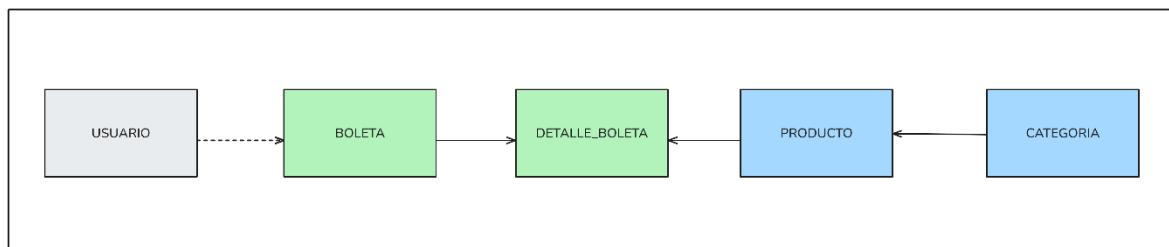
Roles asociados al sistema:

- **Administrador:** tiene acceso total al sistema.
- **Vendedor:**
 - Puede visualizar la lista de los productos y el detalle.
 - Puede visualizar la lista de órdenes y el detalle.
 - Todos los demás accesos no deben aparecer en la vista del vendedor.
- **Cliente:** Solo puede acceder a la tienda.

Propuesta de modelo de base de datos:

Figura 1

Nombre de tablas del Modelo de base de datos relacional, propuesta final.



Criterios de Evaluación:

- **Calidad del Backend:** Estructura, lógica de negocio, y conexión a la base de datos.
- **Integración REST:** Efectividad y documentación de la API.
- **Seguridad:** Implementación adecuada de autenticación y autorización.
- **Colaboración y Documentación:** Uso del repositorio, claridad de la documentación y trabajo en equipo.

Instrucciones para las entregas

Instrucciones para la entrega I

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 del sitio web.

El equipo deberá definir los requerimientos, herramientas y propuestas del proyecto mediante un documento ERS (este documento se irá trabajando en durante el semestre, por lo que debe estar parcialmente terminado).

Entrega del encargo:

-  Enlace GitHub público del proyecto frontend.
-  Proyecto frontend comprimido.
-  Documento ERS (propuesta previa. Versión 1)

Presentación del caso:

- Presentación del caso en 15 minutos cada equipo, más 5 minutos para una ronda de preguntas.
- Presentación del desarrollo funcional.
- Ronda de preguntas abiertas.

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 15 minutos cada equipo, más 5 minutos para una ronda de preguntas.
- Presentación del desarrollo funcional integrado con el backend y frontend.
- Ronda de preguntas abiertas.

Instrucciones para la entrega III

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 del backend e integración entre sistemas.

- ❖ El equipo deberá definir los requerimientos, herramientas y propuestas del proyecto mediante un documento ERS.
- ❖ Entregar un manual de usuario.
- ❖ Documentar la integración, explica modelo de la base de datos, endpoint utilizados, pruebas y resultado comunicación.

Entrega del encargo:

-  Enlace GitHub público del proyecto frontend.
-  Enlace GitHub público del proyecto backend.

-  Proyecto frontend comprimido.
-  Proyecto backend comprimido.

-  Documento ERS (terminado).
-  Manual de usuario (con pantallaos detallados).
-  Documentación de APIs.
-  Documento APIs e Integración.

Presentación del caso:

- Presentación del caso en 15 minutos cada equipo, más 5 minutos para una ronda de preguntas.
- Presentación del desarrollo funcional integrado con el backend y frontend.
- Ronda de preguntas abiertas.

Instrucciones para la entrega del ET

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.

Documentos de entrega:

-  Enlace GitHub público del proyecto frontend.
-  Enlace GitHub público del proyecto backend.
-  Proyecto frontend comprimido.
-  Proyecto backend comprimido.
-  Documento ERS.
-  Manual de usuario.
-  Documento de cobertura del testing.
-  Documentación de APIs.
-  Documento APIs e Integración.

Presentación del caso:

- Presentación del caso en 15 minutos cada equipo, más 5 minutos para una ronda de preguntas.
- Presentación del desarrollo funcional integrado con el backend y frontend.
- Ronda de preguntas abiertas.