

Proyecto final de desarrollo web *front-end*

Requisitos:

Para realizar este proyecto son necesarios conocimientos básicos de:

1. HTML, CSS
2. Node.JS y React
3. El uso de VS code

Nivel del ejercicio: Principiante

Duración: 10 horas aproximadamente

Es requisito completar todos los ejercicios de este proyecto. Asegúrate de entregar este documento con tus respuestas a <mail ID> , a más tardar el <date of submission>.

Objetivo:

Tienes un negocio de venta de hamburguesas y quieres crear un sitio web que te ayude a crecer el negocio. Utiliza React.js para crear tu sitio y hacer que tus clientes puedan comprar hamburguesas en línea.

Resumen:

Este proyecto final se divide en 4 ejercicios:

Ejercicio 1:

- Crear el encabezado y el pie de página de la pantalla de inicio.
- Crear el botón «Explorar el menú» para visualizar la información sobre las hamburguesas disponibles.

Ejercicio 2:

- Crear las páginas «Quiénes somos» y «Contacto» para que los visitantes del sitio encuentren la información de contacto.
- Crear las páginas de registro e inicio de sesión.

Ejercicio 3:

- Crear la página de la cesta de compra para seleccionar productos y hacer pedidos.
- Crear la página de envíos para rellenar la información del pedido y confirmar la entrega.

Ejercicio 4:

- Crear la página «Mis pedidos» donde aparecerán los pedidos.
- Crear la página de información del pedido, donde aparecerá la información completa de cada pedido.

Se puede realizar este proyecto en cualquier IDE (Entorno de desarrollo integrado, por sus siglas en inglés), nosotros recomendamos el IDE VS Code.

Si no tienes VS Code, puedes descargarlo de <https://code.visualstudio.com/download> para instalarlo en tu equipo.

Configuración del proyecto:

1. Haz un *fork* del siguiente repositorio que contiene los códigos del proyecto

...

https://github.com/kumar045/Skillup_Burger_Shop.git

...

Ten en cuenta que es necesario crear este proyecto en una cuenta General de Github y compartir en enlace de dicha cuenta general, en lugar del que aparece aquí arriba para garantizar la privacidad y la confidencialidad de todos los demás proyectos de Github presentes en la cuenta de Github de Shivam.

2. Clona este repositorio a VS Code o al IDE que hayas seleccionado para realizar el proyecto.
3. Sigue las instrucciones para añadir el código y crear cada uno de los componentes de tu aplicación de hamburguesas en línea.

Ejercicio 1

1.1 / Tarea 1: Página de inicio

En esta parte, completarás la página de inicio del proyecto de la hamburguesería.

Instrucciones:

1. Ve al archivo `home/Home.jsx` en la carpeta «components».
2. Importa todos los paquetes necesarios

SUGERENCIA:

La declaración de importación es:

...

```
import React from "react";
```

...

SUGERENCIA:

El código completo es el siguiente:

...

```
import React from "react";
import { motion } from "framer-motion";
import Founder from "../Founder";
import Menu from "../Menu";
```

```
const Home = () => {
  const options = {
    initial: {
      x: "-100%",
      opacity: 0,
    },
    whileInView: {
```

```

    { x: 0,
      opacity: 1,
    },
  ];
};

return (
  <>
    <section className="home">
      <div>
        <motion.h1 {...options}>Hamburguesería</motion.h1>
        <motion.p
          {...options}
          transition={
            {
              delay: 0,2,
            }
          }
        >
          Regálate una sabrosa hamburguesa.
        </motion.p>
      </div>

      <motion.a
        href="#menu"
        initial={{
          y: "-100%",
          opacity: 0,
        }}
        whileInView={{
          { y: 0,
            opacity: 1,
          }}
        transition={
          { delay:
            0,4,
          }}
        >
          Explorar el menú
        </motion.a>
      </section>
      <Menu />

      <Founder />

    </>
  );
};

export default Home;

....

```

1.2 / Tarea 2: Encabezado de la página de inicio

En esta parte, completarás la página de inicio del proyecto de la hamburguesería.

Instrucciones:

1. Ve al archivo diseño/Home.jsx en la carpeta «components».
2. Crea una barra de navegación con enlaces a las páginas «Inicio», «Contacto», «Quiénes somos» y «Cesta de compra» y un menú con enlaces a las páginas «Iniciar sesión», «Pedidos» y «Cerrar sesión».

SUGERENCIA:

Con React Router, una barra de navegación con enlaces a la página de Inicio y con un componente de Menú desplegable se ven así:

```
...
```

```
<Link to="/">Inicio</Link>
```

```
<DropdownMenu />
```

```
...
```

SUGERENCIA:

El código completo sería el siguiente:

```
...
```

```
import React from "react";
import { IoFastFoodOutline } from "react-icons/io5";
import { Link } from "react-router-dom";
import { FiShoppingCart, FiLogIn } from "react-icons/fi";
import { FaUser } from "react-icons/fa";
import { motion } from "framer-motion";
import DropdownMenu from './DropdownMenu';

const Header = ({ isAuthenticated = false }) => {
  return (
    <nav>
      <motion.div initial={{ x: "-100%" }} whileInView={{ x: 0 }}>
        <IoFastFoodOutline />
      </motion.div>

      <div>
        <Link to="/">Inicio</Link>
        <Link to="/contact">Contacto</Link>
        <Link to="/about">Quiénes somos</Link>
        <Link to="/cart">
          <FiShoppingCart />
        </Link>

        <DropdownMenu />
      </div>
    </nav>
  );
};

export default Header;
...
```

1.3 / Tarea 3: Pie de la página de inicio

En esta parte, completarás el pie de la página de inicio del proyecto de la Hamburguesería.

Instrucciones:

1. Ve al archivo layout/Footer.jsx en la carpeta «components».
2. Crea una barra de redes sociales con enlaces a YouTube e Instagram, utiliza etiquetas de anclaje e iconos de la biblioteca react- icons/ai.

SUGERENCIA:

Una barra de redes sociales con enlaces a YouTube con iconos de la biblioteca react- icons/ai se ve así:

```

...

<a href="https://youtube.com/xyz">

    <AiFillYoutube />

</a>

...

```

SUGERENCIA:

El código completo sería el siguiente:

```

...

import React from "react";
import { AiFillInstagram, AiFillYoutube, AiFillGithub } from "react-icons/ai";

const Footer = () => {
  return (
    <footer>
      <div>
        <h2>Hamburguesería</h2>

        <p>Intentamos darte el mejor sabor posible.</p>
        <br />

        <em>Prestamos atención a tus comentarios.</em>

        <strong>Todos los derechos reservados @hamburgueseria</strong>
      </div>

      <aside>
        <h4>Síguenos</h4>

        <a href="https://youtube.com/xyz">
          <AiFillYoutube />
        </a>
        <a href="https://instagram.com/xyz">
          <AiFillInstagram />
        </a>
      </aside>
    </footer>
  );
};

export default Footer;
...

```

1.4 / Tarea 4: Explorar el menú

Ahora crearás el menú con algunos de los artículos disponibles en la Hamburguesería, utilizarás el componente MenuCard:

Instrucciones:

1. Ve al archivo home/Menu.jsx en la carpeta «components».
2. Crea una tarjeta de menú para los artículos que incluya el número de artículo, la fuente de la imagen, el precio, el título, un controlador de función para añadir el producto a la cesta y un retraso para la animación.

SUGERENCIA:

La primera tarjeta de menú es:

```
...
```

```
<MenuCard
  itemNum={1}
  burgerSrc={burger1}
  price={3,30}
  title="Hamburguesa con queso"
  handler={addToCartHandler}
  delay={0.1}
/>
...
```

SUGERENCIA:

El código completo sería el siguiente:

```
...
```

```
import React from "react";
import MenuCard from "../MenuCard";
import burger1 from "../../assets/burger1.png";
import burger2 from "../../assets/burger2.png";
import burger3 from "../../assets/burger3.png";

const Menu = () => {
  const addToCartHandler = (itemNum) => {};

  return (
    <section id="menu">
      < h1> MENÚ</h1>

      <div>
        <MenuCard
          itemNum={1}
          burgerSrc={burger1}
          price={3,30}
          title="Hamburguesa con queso"
          handler={addToCartHandler}
          delay={0.1}
        />
        <MenuCard
          itemNum={2}
          burgerSrc={burger2}
          price={4,50}
          title="Hamburguesa vegetariana con queso"
          delay={0.5}
          handler={addToCartHandler}
        />
        <MenuCard
          itemNum={3}
          burgerSrc={burger3}
          price={6,10}
          title="Hamburguesa con queso y patatas fritas"
          delay={0.8}
          handler={addToCartHandler}
        />
      </ div>
    </section>
  );
};

export default Menu;
...
```

Con esto, habrás añadido las primeras secciones a tu solicitud de hamburguesas en línea.

Probar la aplicación:

Abre una nueva terminal:

1. Utiliza el siguiente comando para navegar hasta el directorio Skillup_Burger_Shop:

```
***
```

```
cd .\Skillup_Burger_Shop\
```

```
***
```

2. Ejecuta el siguiente comando para instalar los módulos/paquetes Node necesarios:

```
***
```

```
npm install -s
```

```
--
```

```
-
```

3. Inicia el servidor con el siguiente comando:

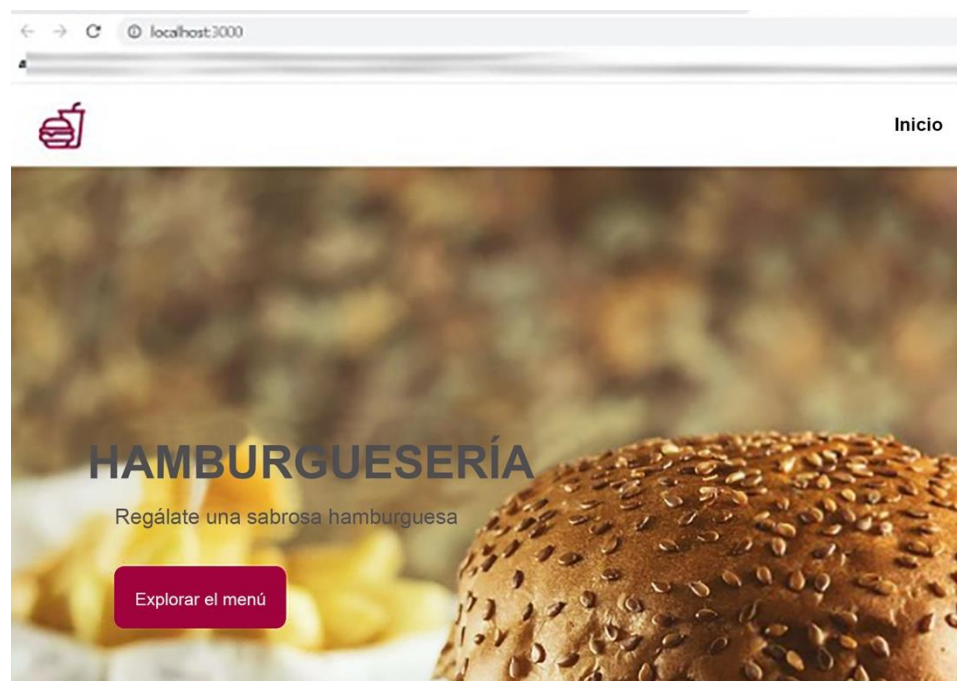
```
***
```

```
npm start
```

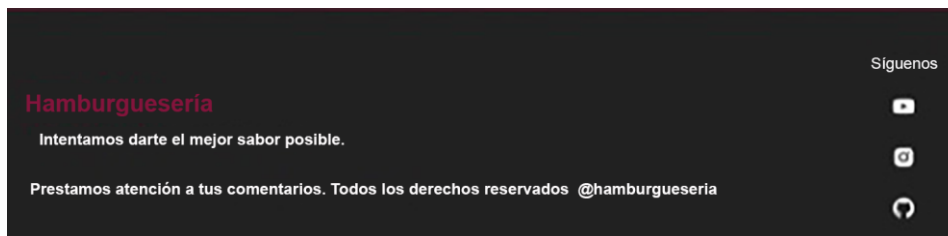
```
***
```

4. Esto iniciará tu aplicación React en **localhost:3000** en una nueva ventana del navegador.

5. La página de inicio y el encabezado deben tener un aspecto similar a:



6. El pie de página debe tener un aspecto similar a:

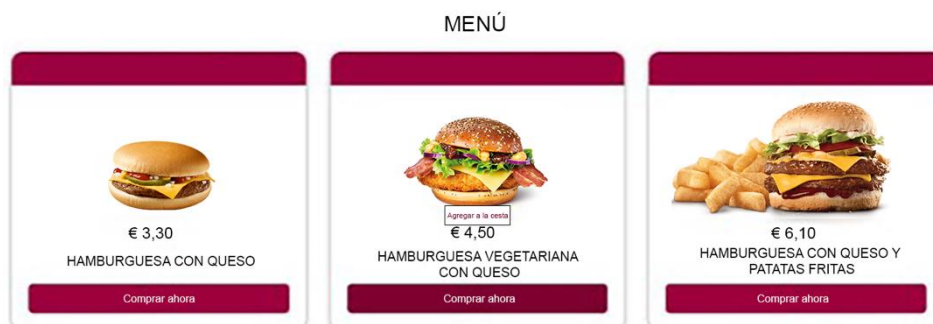


7. El botón "Explorar el menú" debe tener un aspecto similar a:



Al hacer clic en él, accederás al menú, como se muestra a continuación:

Al hacer clic en cualquiera de las hamburguesas (por ejemplo, «Hamburguesa vegetariana con queso») aparecerá una ventana emergente con el mensaje que se muestra a continuación:



Ejercicio 2:

2.1 / Tarea 1: Página «Quiénes somos»

En esta parte, completarás la página «Quiénes somos» del proyecto de la Hamburguesería.

Instrucciones:

1. Ve al archivo home/About.jsx en la carpeta «components».
2. Importa todos los paquetes necesarios y la foto del fundador

SUGERENCIA:

La declaración para importar la foto del fundador es:

```
'''
```

```
import me from "../../assets/skj.jpg";
'''
```

SUGERENCIA:

El código completo sería el siguiente:

```
...

import React from "react";
import { Link } from "react-router-dom";
import { RiFindReplaceLine } from "react-icons/ri";
import me from "../../assets/skj.jpg";

const About = () => {
  return (
    <section className="about">
      <main>
        <h1>Quiénes somos</h1>

        <article>
          <h4>Hamburguesería</h4>
          <p>
            Esto es la Hamburguesería. El lugar con las hamburguesas más
            sabrosas del planeta.
          </p>

          <p>
            Explora nuestra oferta de comida y hamburguesas. Haz clic a
            continuación para ver el menú
          </p>

          <Link to="/">
            <RiFindReplaceLine />
          </Link>
        </article>

        <div>
          <h2>Fundador</h2>
          <article>
            <div>
              <img src={me} alt="Fundador" />
              <h3>Nelson</h3>
            </div>

            <p>
              Soy Nelson, el fundador de la Hamburguesería y afiliado a Good Taste...
            </p>
          </article>
        </div>
      </main>
    </section>
  );
};

export default About;
...
```

2.2 / Tarea 2: Página de contacto

En esta parte, completarás la página de contacto del proyecto de la Hamburguesería.

Instrucciones:

1. Ve al archivo home/Contacto.jsx en la carpeta «components».
2. Crea una ventana emergente que aparezca al pulsar el botón de envío.

SUGERENCIA:

Puedes crear una ventana emergente desde la librería reactjs-popup:

...

```
import Popup from 'reactjs-popup';
<Popup trigger=
  {<button type="button">Enviar</button>}
>
</ Popup>
...
```

SUGERENCIA:

El código completo sería el siguiente:

...

```
import React from "react";
import { motion } from "framer-motion";
import burger from "../../assets/burger2.png";
import Popup from 'reactjs-popup';

const Contact = () => {
  return (
    <section className="contact">
      <motion.form
        initial={{
          x: "-100vw",
          opacity: 0,
        }}
        animate={{
          x: 0,
          opacity: 1,
        }}
        transition={{ delay: 0.2 }}
      >
        <h2>Contáctanos</h2>
        <input type="text" placeholder="Nombre" />
        <input type="email" placeholder="Correo electrónico" />

        <textarea placeholder="Tu mensaje..." cols="30" rows="10"></textarea>
        <Popup trigger=
          {<button type="button">Enviar</button>}
          >
          <div style={{color:"red", transform: 'translate(25%,-700%)',
            backgroundColor: '#fff', padding: '10px', borderRadius: '5px', boxShadow: '0 0 10px
            rgba(0, 0, 0, 0.2)}}>¡Gracias por contactarnos! En breve te responderemos.</div>

        </Popup>

      </motion.form>

      <motion.div
        className="formBorder
        " initial={{
          x: "100vw",
          opacity: 0,
        }}
        animate={{
          x: 0,
```

```

        opacity: 1,
      }}
    transition={{ delay: 0.2 }}
  >
  <motion.div
    initial={{
      y: "-100vh",
      x: "50%",
      opacity: 0,
    }}
    animate={{
      x:
        "50%",
      y: "-50%",
      opacity: 1,
    }}
    transition={
      { delay:
        1,
      }}
  >
    <img src={burger} alt="Hamburguesa" />
  </motion.div>
</motion.div>
</section>
);
};

```

```
export default Contact;
```

```
...
```

2.3 / Tarea 3: Página de registro e inicio de sesión

En esta parte, crearás la página de registro e inicio de sesión de la Hamburguesería.

Instrucciones:

1. Ve al archivo home/Login.jsx en la carpeta «components».
2. Crea pestañas de inicio de sesión y registro para el formulario.

SUGERENCIA:

Puedes crear una pestaña de inicio de sesión así:

```
...
```

```

<MDBTabsPane show={justifyActive === "tab1"}>
  <div className="text-center mb-3">
    <p>Iniciar sesión con:</p>

    <div
      className="d-flex justify-content-between mx-auto"
      style={{ width: "40%" }}
    >
      <MDBBtn tag="a" color="none" className="m-1" style={{ color: "#1266f1" }}>
        <MDBIcon fab icon="facebook-f" size="sm" />
      </MDBBtn>

      <MDBBtn tag="a" color="none" className="m-1" style={{ color: "#1266f1" }}>
        <MDBIcon fab icon="twitter" size="sm" />
      </MDBBtn>
    </div>
  </div>

```

```

        <MDBBtn tag="a" color="none" className="m-1" style={{ color: "#1266f1" }}>
            <MDBIcon fab icon="google" size="sm" />
        </MDBBtn>

        <MDBBtn tag="a" color="none" className="m-1" style={{ color: "#1266f1" }}>
            <MDBIcon fab icon="github" size="sm" />
        </MDBBtn>
    </div>

    <p className="text-center mt-3">o con:</p>
</div>

<MDBInput wrapperClass="mb-4" label="Dirección de correo electrónico" id="form1"
type="email" />
<MDBInput wrapperClass="mb-4" label="Contraseña" id="form2" type="password" />

<div className="d-flex justify-content-between mx-4 mb-4">
    <MDBCheckbox
        name="flexCheck"
        value=""
        id="flexCheckDefault"
        label="Recuérdame"
    />
    <a href="#!>¿Olvidaste tu contraseña?</a>
</div>

<MDBBtn className="mb-4 w-100">Iniciar sesión</MDBBtn>
<p className="text-center">
    ¿No estás registrado? <a href="#!">Regístrate</a>
</p>
</MDBTabsPane>;

```

...

SUGERENCIA:

El código completo será el siguiente:

...

```

import React, { useState } from "react";
import {
    MDBContainer,
    MDBTabs,
    MDBTabsItem,
    MDBTabsLink,
    MDBTabsContent,
    MDBTabsPane,
    MDBBtn,
    MDBIcon,
    MDBInput,
    MDBCheckbox,
} from "mdb-react-ui-kit";

function Login() {
    const [justifyActive, setJustifyActive] = useState("tab1");

    const handleJustifyClick = (value) => {
        if (value === justifyActive) {
            return;
        }

        setJustifyActive(value);
    };

```

```

};

return (
  <MDBContainer className="p-3 my-5 d-flex flex-column w-50">
    <MDBTabs
      pills
      justify
      className="mb-3 d-flex flex-row justify-content-between"
    >
      <MDBTabsItem>
        <MDBTabsLink
          onClick={() => handleJustifyClick("tab1")}
          active={justifyActive === "tab1"}
        >
          Login
        </MDBTabsLink>
      </MDBTabsItem>
      <MDBTabsItem>
        <MDBTabsLink
          onClick={() => handleJustifyClick("tab2")}
          active={justifyActive === "tab2"}
        >
          Regístrate
        </MDBTabsLink>
      </MDBTabsItem>
    </MDBTabs>

    <MDBTabsContent>
      <MDBTabsPane show={justifyActive === "tab1"}>
        <div className="text-center mb-3">
          <p>Iniciar sesión con:</p>

          <div
            className="d-flex justify-content-between mx-auto"
            style={{ width: "40%" }}
          >
            <MDBBtn
              tag="a"
              color="none"
              className="m-1"
              style={{ color: "#1266f1" }}
            >
              <MDBIcon fab icon="facebook-f" size="sm" />
            </MDBBtn>

            <MDBBtn
              tag="a"
              color="none"
              className="m-1"
              style={{ color: "#1266f1" }}
            >
              <MDBIcon fab icon="twitter" size="sm" />
            </MDBBtn>

            <MDBBtn
              tag="a"
              color="none"
              className="m-1"
              style={{ color: "#1266f1" }}
            >
              <MDBIcon fab icon="google" size="sm" />
            </MDBBtn>
          </div>
        </div>
      </MDBTabsPane>
    </MDBTabsContent>
  </MDBContainer>
);

```

```

        <MDBBtn
          tag="a"
          color="none"
          className="m-1"
          style={{ color: "#1266f1" }}
        >
          <MDBIcon fab icon="github" size="sm" />
        </MDBBtn>
      </div>

      <p className="text-center mt-3">or:</p>
    </div>

    <MDBInput
      wrapperClass="mb-4"
      label="Dirección de correo electrónico"
      id="form1"
      type="email"
    />
    <MDBInput
      wrapperClass="mb-4"
      label="Contraseña"
      id="form2"
      type="password"
    />

    <div className="d-flex justify-content-between mx-4 mb-4">
      <MDBCheckbox
        name="flexCheck"
        value=""
        id="flexCheckDefault"
        label="Recuérdame"
      />
      <a href="#">¿Olvidaste tu contraseña?</a>
    </div>

    <MDBBtn className="mb-4 w-100">Iniciar sesión</MDBBtn>
    <p className="text-center">
      ¿No estás registrado? <a href="#">Regístrate</a>
    </p>
  </MDBTabsPane>

  <MDBTabsPane show={justifyActive === "tab2"}>
    <div className="text-center mb-3">
      <p>Regístrate con:</p>

      <div
        className="d-flex justify-content-between mx-auto"
        style={{ width: "40%" }}
      >
        <MDBBtn
          tag="a"
          color="none"
          className="m-1"
          style={{ color: "#1266f1" }}
        >
          <MDBIcon fab icon="14facebook-f" size="sm" />
        </MDBBtn>

        <MDBBtn

```


...

Probar la aplicación:

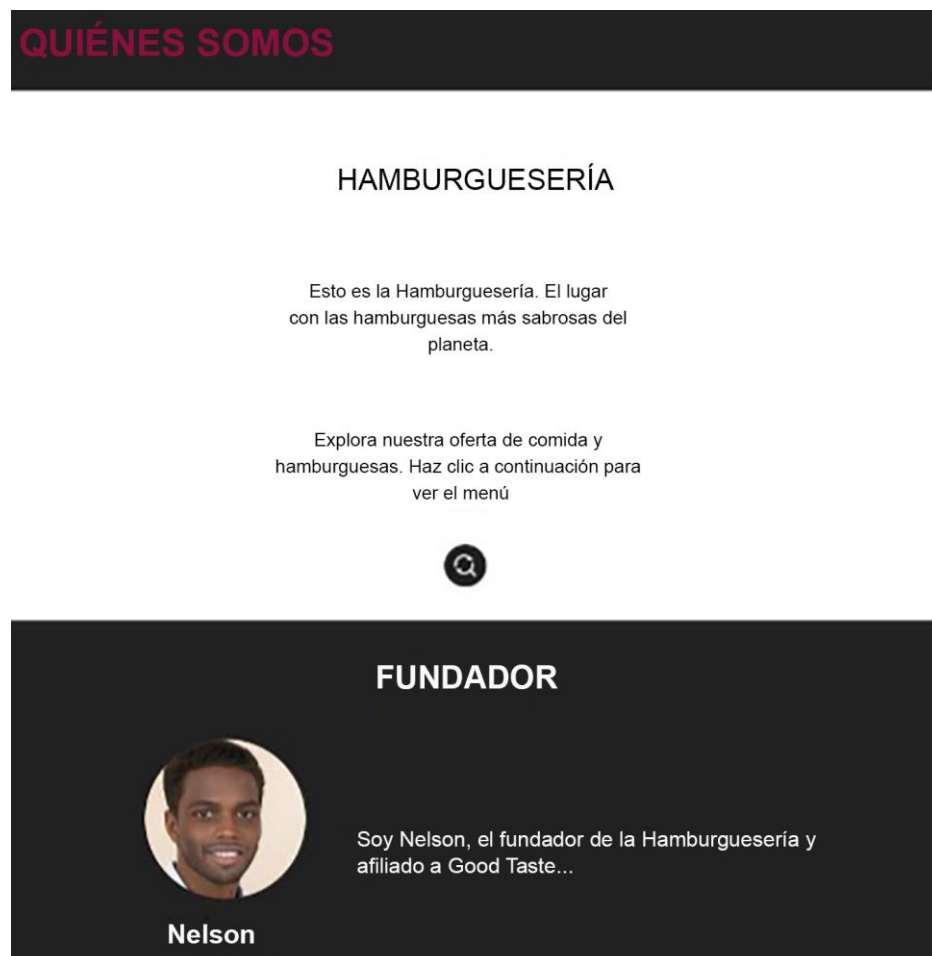
1. Inicia el servidor para probar tu aplicación:

...

```
npm start
```

...

2. El pie de página debe tener un aspecto similar a:







3. La página de inicio de sesión y registro debe tener un aspecto similar a:

INICIAR SESIÓN

REGÍSTRATE

Iniciar sesión con:



o con:

Dirección de correo electrónico:

Contraseña:

☐ Recuérdame

[¿Olvidaste tu contraseña?](#)

INICIAR SESIÓN

[¿No estás registrado? Regístrate](#)

4. La página de contacto y la ventana emergente para los comentarios tendrán un aspecto similar a:

CONTÁCTANOS

¡Gracias por contactarnos! En breve te responderemos.



Ejercicio 3

3.1 / Tarea 1: Crear la página «Cesta de compra»

Ahora completarás el menú de la cesta de compra para seleccionar productos y realizar pedidos.

Ya se han añadido 2 productos, la hamburguesa con queso y la hamburguesa vegetariana con queso. Debes añadir otro artículo, por ejemplo, «Hamburguesa con queso y patatas fritas»

Instrucciones:

1. Ve al archivo `cart/Cart.jsx` en la carpeta «components» y añade un elemento React «`CartItem`».
2. Establece sus atributos como:
 - título: «Hamburguesa con queso y patatas fritas»
 - `img`: `burger3`. Es necesario importar la imagen («`burger3`») de la carpeta «assets» para utilizarla como variable.
 - Fija el valor inicial en 0.

- Añade contadores de incremento y decremento.

SUGERENCIA:

Declaración de importación para burger3.png:

```
...
import burger3 from "../../assets/burger3.png";
...
```

SUGERENCIA:

El código completo de Cart.jsx es:

```
...
import React from "react";
import { Link } from "react-router-dom";
import burger1 from "../../assets/burger1.png";
import burger2 from "../../assets/burger2.png";
import burger3 from "../../assets/burger3.png";

const CartItem = ({ value, title, img, increment, decrement }) => (
  <div className="cartItem">
    <div>
      <h4>{título}/h4>
      <img src={img} alt="Producto" />
    </div>

    <div>
      <button onClick={decrement}>-</button>
      <input type="number" readOnly value={value} />
      <button onClick={decrement}>+</button>
    </div>
  </div>
);

const Cart = () => {
  const increment = (item) => {};

  const decrement = (item) => {};

  return (
    <section className="cart">
      <main>
        <CartItem
          title={"Hamburguesa con queso"} img={burger1}
          value={0}
          increment={() => increment(1)}
          decrement={() => decrement(1)}
        />
        <CartItem
          title={"Hamburguesa vegetariana con queso"}
          img={burger2}
          value={0}
          increment={() => increment(2)}
          decrement={() => decrement(2)}
        />
        <CartItem
          title="Hamburguesa con queso y patatas fritas"
          img={burger3}
          value={0}
          increment={() => increment(3)}
        />
      </main>
    </section>
  );
};
```

```

    decrement={() => decrement(3)}
  />

  <article>
    <div>
      <h4>Subtotal</h4>
      <P>€ {13,90}</p>
    </div>
    <div>
      <h4>IVA</h4>
      <P>€ {13,90 * 0,21}</p>
    </div>
    <div>
      <h4>Costos de envío</h4>
      <P>€ {2,30}</p>
    </div>{ " "}
    <div>
      <h4>Total</h4>
      <P>€ {13,90 + 13,90 * 0,21 + 2,30}</p>
    </div>
    <Link to="/shipping">Envío</Link>
  </article>
</main>
</section>
);
};

export default Cart;

...

```

Ahora añadirás la ruta para la cesta:

3. Ve al archivo `src/App.js` e importa el objeto «Cart» que es un módulo exportado en `Cart.jsx`.

SUGERENCIA:

```
import Cart from "../components/cart/Cart";
```

4. Añade la ruta para «Cart» en el elemento `<Routes>` dentro del método `Return` del elemento `App()`.

SUGERENCIA:

```
<Route path="/cart" element={<Cart />} />
```

5. Navega hasta el archivo `src/styles/cart.scss` y observa el trabajo de estilo CSS que se ha realizado para la sección «Cart». Verás esto importado en `src/App.js` como:

```
import "../styles/cart.scss";
```

3.2 / Tarea 2: Página de envío

Ahora completarás la información del pedido y del envío. El código está parcialmente completo, tu debes hacer el resto.

Instrucciones:

1. Ve al archivo `cart/Shipping.jsx` en la carpeta «components».
2. Ve al elemento `<div>` marcado como DESPLEGABLE DE PAÍSES (*COUNTRY DROPDOWN*)
3. En el siguiente código:
 - etiqueta el elemento con el nombre «País» (*Country*),
 - selecciona el elemento del desplegable y
 - selecciona un elemento de opción para el desplegable, i. e. nombres de países (*country names*).
4. Escribe el código restante como se indica a continuación:
 - Utiliza el método `getAllCountries` para obtener una matriz con los datos de todos los países.
 - Utiliza el método `map()` para recorrer esta matriz, obtener el código ISO de cada país como atributo «value» de la etiqueta de opción y mostrar el nombre del país en el menú desplegable.

SUGERENCIA:

El código completo para el elemento `div` DESPLEGABLE DE PAÍSES será:

```
<div>
  { /* DESPLEGABLE DE PAÍSES */ }
  <label>País</label>

  <select>
    <option value="">País</option>
    {Country && Country.getAllCountries().map((i) => (
      <option value="{i.isoCode}" key="{i.isoCode}">
        {i.name}
      </option>
    ))}
  </select>
</div>
```

5. Del mismo modo, crea un desplegable para seleccionar las provincias de un país.

Será similar al anterior, con las siguientes diferencias:

- utilizarás el objeto «State» de la biblioteca «country-state-city» y
- utilizarás el método `getStatesOfCountry()` con el parámetro «ES» (para seleccionar España como país).

SUGERENCIA:

El código completo para el DESPLEGABLE DE PROVINCIAS será:

```
<div>
  <label>Estado/Provincia</label>

  <select>
    <option value="">Estado/Provincia</option>
    {State && State.getStatesOfCountry("ES").map((i) => (
      <option value="{i.isoCode}" key="{i.isoCode}">
        {i.name}
      </option>
    ))}
  </select>
</div>
```

6. Hay un elemento de entrada para introducir el código postal.
7. Crea un elemento de entrada para el número telefónico.

SUGERENCIA:

El código para el elemento de entrada del número telefónico es:

```
<div>
  //
  <label>N.º de teléfono</label>
  //
  <input type="number" placeholder="Ingresa tu número telefónico" />
  //
</div>
```

8. El código final es un elemento <Popup> que genera un mensaje emergente de «Pedido realizado» al hacer clic en el botón «Confirmar pedido». Observa los elementos CSS que se han utilizado para darle estilo.

9. El código final de Shipping.jsx será:

```
...

import React from "react";
import { State } from "country-state-city";
import Popup from "reactjs-popup";
import { Link } from "react-router-dom";

const Shipping = () => {
  return (
    <section className="shipping">
      <main>
        <h1>Información de envío</h1>
        <form>
          <div>
            <label>Calle y N.º</label>
            <input type="text" placeholder="Ingresa el nombre de la calle y el número" />
          </div>
          <div>
            <label>Ciudad</label>
            <input type="text" placeholder="Ingresa la ciudad" />
          </div>
          <div>
            <label>País</label>

            <select>
              <option value="">España</option>
            </select>
          </div>
          <div>
            <label>Estado/Provincia</label>

            <select>
              <option value="">Estado/Provincia</option>
              {State &&
                State.getStatesOfCountry("ES").map((i) => (
                  <option value={i.isoCode} key={i.isoCode}>
                    {i.name}
                  </option>
                ))}
            </select>
          </div>
        </div>
      </main>
    </section>
  );
}
```

```

        <label>Código postal</label>
        <input type="number" placeholder="Ingresa tu PIN" />
    </div>
    <div>
        <label>N.º de teléfono</label>
        <input type="number" placeholder="Ingresa tu número telefónico" />
    </div>

    <Popup
      trigger={
        <Link className="link" to="/myorders">
          Confirmar pedido
        </Link>
      }
    >
      <div
        style={{
          color: "red",
          transform: "translate(0%, -500%)", backgroundColor: "#fff",
          padding: "10px",
          borderRadius: "5px",
          boxShadow: "0 0 10px rgba(0, 0, 0, 0.2)",
        }}
      >
        ¡Has realizado tu pedido con éxito!
      </div>
    </Popup>
  </form>
</main>
</section>
);
};

export default Shipping;
...

```

10. Importa el objeto «Shipping» (exportado desde Shipping.jsx) en App.js y crea la ruta «Shipping» en el método Return del elemento App().

SUGERENCIA:

Módulo de importación:




```
import Shipping from "../components/cart/Shipping";
```

Ruta:

```
<Route path="/shipping" element={<Shipping />} />
```

Probar la aplicación:

1. Inicia el servidor para probar tu aplicación.
2. El pie de página debe tener un aspecto similar a:

Hamburguesa con queso		<div><div>-</div><div>0</div><div>+</div></div>
Hamburguesa vegetariana con queso		<div><div>-</div><div>0</div><div>+</div></div>
Hamburguesa con queso y patatas fritas		<div><div>-</div><div>0</div><div>+</div></div>
Subtotal		€ 13,90
IVA		€ 2,92
Costos de envío		€ 2,30
Total		€ 19,12
<div>Envío</div>		

- Hacer clic en el botón «Confirmar pedido» te llevará a la página «Envío».
- El pie de página debe tener un aspecto similar a:

INFORMACIÓN DE ENVÍO

Calle y número	<input type="text" value="Gran Vía 123"/>
Ciudad	<input type="text" value="Madrid"/>
País	<input type="text" value="España"/>
Estado/Provincia	<input type="text" value="Madrid"/>
Código postal	<input type="text" value="9876"/>
Teléfono	<input type="text" value="+34 567 890 123"/>

Confirmar pedido

Ejercicio 4

4.1 / Tarea 1: Página «Mis pedidos»

Ahora crearás la página «Mis Pedidos» que mostrará el número de pedido, el estado del pedido, la cantidad de artículos, el importe, la forma de pago y una acción para cada uno de tus pedidos.

Navega hasta `myOrders/myOrders.jsx` en la carpeta `components`. Verás el código de la etiqueta `<thead>`. Ahora añadirás el código de la etiqueta `<tbody>`

1. Utiliza el método «map» para recorrer los números de pedido y mostrar:
 - el estado como «En curso»,
 - la cantidad de artículos como 3,
 - el importe como € 16.00,
 - la forma de pago como «pago contra entrega» y
 - un icono con forma de ojo (*eye icon*) que redirigirá a la página «Información del pedido» al pulsarlo.

SUGERENCIA:

El código completo de `<tbody>` es:

```
<tbody>
  {arr.map((i) => (
    <tr key={i}>
      <td>#ON??????</td>
      <td>En curso</td>
      <td>3</td>
      <Td>€ {16,00}</td>
      <Td>pago contra entrega</td>
      <td>
        <Link to={` /order/${"ON00012345"}`}>
          <AiOutlineEye />
        </Link>
      </td>
    </tr>
  ))}
</tbody>;
```

2. Importa el objeto `MyOrders` (exportado desde `Shipping.jsx`) en `App.js` y crea la ruta «Shipping» en el método `Return` del elemento `App()`

SUGERENCIA:

Módulo de importación:

```
import MyOrders from "../components/myOrders/MyOrders";
```

Ruta:

```
<Route path="/myorders" element={<MyOrders />} />
```

4.2 / Tarea 2: Página «Información del pedido»

En esta parte, completarás la página de información del pedido del proyecto de la Hamburguesería.

Instrucciones:

1. Ve al archivo `myOrders/OrderDetails.jsx` en la carpeta «components».
2. Crea dos secciones en la página de información del pedido, una de estado, que contenga el estado del pedido, la fecha del pedido y la fecha de entrega, y otra de pago, que contenga el método de pago, la referencia de pago y la fecha de pago.

SUGERENCIA:

Sección de estado:

```
'''
```

```
<div>
<h1>Estado</h1>
</div>
```

```
'''
```

SUGERENCIA:

Sección de pagos:

```
'''
```

```
<div>
<h1>Pago</h1>
</div>
```

```
'''
```

SUGERENCIA:

El código completo de ***orderDetails.jsx*** será:

```
'''
```

```
import React from "react";

const OrderDetails = () => {
  return (
    <section className="orderDetails">
      <main>
        <h1>Información del pedido</h1>
        <div>
          <h1>Envío</h1>
          <p>
            <b>Dirección</b>

```

```
        {"Gran Vía 123"}
    </p>
</div>
<div>
    <h1>Contacto</h1>
    <p>
        <b>Nombre</b>
        {"Pedro"}
    </p>
    <p>
        <b>Teléfono</b>
        {"+34 567 890 123"}
    </p>
</div>

<div>
    <h1>Estado</h1>
    <p>
        <b>Estado del pedido</b>
        {"En curso"}
    </p>
    <p>
        <b>Fecha del pedido</b>
        {"23-06-2023"}
    </p>
    <p>
        <b>Fecha de entrega</b>
        {"23-06-2023"}
    </p>
</div>

<div>
    <h1>Pago</h1>
    <p>
        <b>Método de pago</b>
        {"Pago contra entrega"}
```

```
</p>

<p>
  <b>Referencia de pago</b>#{ "RP00012345" }
</p>

<p>
  <b>Fecha de pago</b>
  { "23-06-2023" }
</p>
</div>


<div>
  <h1>Total</h1>
  <p>
    <b>Subtotal</b>€ {13,90}
  </p>
  <p>
    <b>Costos de envío</b>€ {2,30}
  </p>
  <p>
    <b>IVA</b>€ {2,92}
  </p>
  <p>
    <b>Importe total</b>€ {2,92 + 2,30 + 13,90}
  </p>
</div>


<article>
  <h1>Artículos</h1>
  <div>
    <h4>Hamburguesa con queso</h4>
    <div>
      <span>{1}</span> x € <span>{3,30}</span>
    </div>
  </div>
  <div>
    <h4>Hamburguesa vegetariana con queso</h4>
```

```

        <div>
            <span>{1}</span> x € <span>{4,50}</span>
        </div>
    </div>
    <div>
        <h4>Hamburguesa con queso y patatas fritas</h4>
        <div>
            <span>{1}</span> x € <span>{6,10}</span>
        </div>
    </div>

    <div>
        <h4
            style={{
                fontWeight: 800,
            }}
        >
            Subtotal
        </h4>
        <div
            style={{
                fontWeight: 800,
            }}
        >
            € {13,90}
        </div>
    </div>
</article>
</main>
</section>
);
};

export default OrderDetails;
...

```

Probar la aplicación:

1. Inicia el servidor para probar tu aplicación.
2. Al hacer clic en el botón "Confirmar pedido" de la página Envío, se cargará la página «Mis Pedidos» que debe tener un aspecto similar a:

Número de pedido	Estado	Cantidad	Importe	Método de pago	Ver más
#ON???????	En curso	3	€ 16,00	Pago contra entrega	
#ON???????	En curso	3	€ 16,00	Pago contra entrega	
#ON???????	En curso	3	€ 16,00	Pago contra entrega	
#ON???????	En curso	3	€ 16,00	Pago contra entrega	

3. Al hacer clic en el «icono del ojo» en la columna «Acción» accederás a la página «Información del pedido», que debe tener un aspecto similar a:

INFORMACIÓN DEL PEDIDO

ENVÍO

Dirección: Gran Vía 123

CONTACTO

Nombre: Pedro

Teléfono: +34 567 890 123

ESTADO

Estado del pedido: En curso

Fechas del pedido: 23-06-2023

Fecha de entrega: 23-06-2023

PAGO

Método de pago: pago contra entrega

Referencia de pago: RP00012345

Fecha de pago: 23-06-2023

TOTAL

Subtotal: € 13,90

Costos de envío: € 2,30

IVA: € 2,92

Importe total: € 19,12

ARTÍCULOS

Hamburguesa con queso	1 x € 3,30
Hamburguesa vegetariana con queso	1 x € 4,50
Hamburguesa con queso y patatas fritas	1 x € 6,10
Subtotal	€ 13,90

¡Enhorabuena! Ya has agregado todas las secciones a tu aplicación de hamburguesas en línea y has probado su funcionamiento. Con esto, has terminado tu proyecto final.

