Index.js

import React from 'react';

import ReactDOM from 'react-dom';

import './styles.css'; // Import the CSS file

import App from './App'; // Import the App component

ReactDOM.render(

  <React.StrictMode>

    <App />

  </React.StrictMode>,

  document.getElementById('root')

);  
  
  
app.js

import React, { useState } from 'react';

const App = () => {

  const [customerName, setCustomerName] = useState('');

  const [customerNumber, setCustomerNumber] = useState('');

  const [selectedItem, setSelectedItem] = useState('1');

  const [quantity, setQuantity] = useState(1);

  const [specialInstructions, setSpecialInstructions] = useState('');

  const [termsAccepted, setTermsAccepted] = useState(false);

  const [orderDetails, setOrderDetails] = useState(null);

  const [errors, setErrors] = useState({});

  const menuItems = [

    { id: '1', name: 'Pizza' },

    { id: '2', name: 'Burger' },

    { id: '3', name: 'Pasta' }

  ];

  const images = [

    '/images/pizza.jpg',

    '/images/burger.jpg',

    '/images/pasta.jpg'

  ];

  const handleSubmit = (e) => {

    e.preventDefault();

    const newErrors = {};

    if (customerNumber.length !== 10) newErrors.customerNumber = 'Contact number must be exactly 10 digits.';

    if (quantity < 1 || quantity > 3) newErrors.quantity = 'Quantity must be between 1 and 3.';

    if (!termsAccepted) newErrors.terms = 'You must accept the terms and conditions.';

    if (Object.keys(newErrors).length > 0) {

      setErrors(newErrors);

      return;

    }

    const order = {

      customerName,

      customerNumber,

      menuItem: menuItems.find(item => item.id === selectedItem).name,

      quantity,

      specialInstructions,

      totalOrder: quantity

    };

    setOrderDetails(order);

    setErrors({});

  };

  const handleReset = () => {

    setCustomerName('');

    setCustomerNumber('');

    setSelectedItem('1');

    setQuantity(1);

    setSpecialInstructions('');

    setTermsAccepted(false);

    setOrderDetails(null);

    setErrors({});

  };

  return (

    <div className="App">

      <header className="App-header">

        <h1>Restaurant Page</h1>

      </header>

      <main className="container">

        <nav>

          <ul>

            <li><a href="#order">Order Now</a></li>

            <li><a href="#gallery">Gallery</a></li>

          </ul>

        </nav>

        <section id="order" className="order-section">

          <div className="form-container">

            <h1>Restaurant Order Form</h1>

            <form onSubmit={handleSubmit}>

              <label>

                Customer Name:

                <input

                  type="text"

                  value={customerName}

                  onChange={(e) => setCustomerName(e.target.value)}

                  placeholder="Enter your name"

                />

              </label>

              <label>

                Customer Number:

                <input

                  type="text"

                  value={customerNumber}

                  onChange={(e) => setCustomerNumber(e.target.value)}

                  maxLength="10"

                  placeholder="Enter your contact number"

                />

                {errors.customerNumber && <p className="error">{errors.customerNumber}</p>}

              </label>

              <label>

                Menu Item:

                <

Styles.css

/\* General Styles \*/

.App {

font-family: 'Arial', sans-serif;

color: #333;

}

.App-header {

background-color: #4caf50;

padding: 20px;

text-align: center;

color: white;

box-shadow: 0 2px 8px rgba(0, 0, 0, 0.1);

}

nav ul {

list-style: none;

text-align: center;

padding: 0;

}

nav ul li {

display: inline;

margin: 0 15px;

}

nav ul li a {

text-decoration: none;

color: #4caf50;

font-weight: bold;

}

.container {

padding: 20px;

}

.order-section {

margin-bottom: 40px;

}

.form-container {

padding: 20px;

background-color: #f9f9f9;

border: 1px solid #ddd;

border-radius: 5px;

box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);

}

label {

display: block;

margin: 10px 0;

font-weight: bold;

}

input,

textarea,

select {

width: 100%;

padding: 10px;

margin-top: 5px;

border: 1px solid #ccc;

border-radius: 4px;

}

textarea {

resize: vertical;

}

button {

padding: 10px 15px;

margin: 10px 5px;

border: none;

border-radius: 4px;

background-color: #4caf50;

color: white;

font-weight: bold;

cursor: pointer;

transition: background-color 0.3s;

}

button:hover {

background-color: #45a049;

}

.button-group {

display: flex;

justify-content: flex-start;

}

.error {

color: red;

font-size: 12px;

margin-top: 5px;

}

.order-summary {

margin-top: 20px;

padding: 15px;

background-color: #eef;

border: 1px solid #bbd;

border-radius: 4px;

}

.gallery {

display: flex;

justify-content: space-around;

flex-wrap: wrap;

}

.gallery img {

width: 200px;

height: auto;

border-radius: 8px;

margin: 10px;

}

footer {

background-color: #f1f1f1;

padding: 15px;

text-align: center;

border-top: 1px solid #ddd;

}