***App.js***

import React from 'react';

import { BrowserRouter as Router, Routes, Route } from 'react-router-dom';

import Login from './Login';

import Register from './Register';

import SplashScreen from './SplashScreen';

import AL from './al'; // Import AL component

function App() {

  return (

    <Router>

      <SplashScreen />

      <Routes>

        <Route path="/" element={<Login />} />

        <Route path="/register" element={<Register />} />

        <Route path="/al" element={<AL />} />

      </Routes>

    </Router>

  );

}

export default App;

***App.css***

.login-form {

  width: 300px;

  margin: 100px auto;

  padding: 20px;

  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

  text-align: center;

  display: flex;

  flex-direction: column;

  align-items: center;

}

.login-logo {

  width: 100px;

  /\* Adjust size as needed \*/

  height: 100px;

  border-radius: 50%;

  display: block;

  margin: 0 auto 20px;

  /\* Centering the logo and adding space below \*/

}

.login-form input[type="text"],

.login-form input[type="password"] {

  width: 100%;

  padding: 10px;

  margin: 8px 0;

  box-sizing: border-box;

  border: 1px solid #ccc;

  border-radius: 4px;

}

.login-form button {

  width: 100%;

  padding: 10px;

  background-color: #007bff;

  color: white;

  border: none;

  border-radius: 4px;

  cursor: pointer;

  margin-top: 10px;

}

.login-form button:hover {

  background-color: #0056b3;

}

.login-form a {

  display: block;

  margin-top: 15px;

  text-decoration: none;

  color: blue;

}

***al.js***

import React, { useState, useEffect } from "react";

function BookingConfirmation({ onConfirm }) {

  const [driverName, setDriverName] = useState("");

  const [driverMobile, setDriverMobile] = useState("");

  useEffect(() => {

    // Generate random driver name and mobile number

    const randomDriverName = generateRandomName();

    const randomDriverMobile = generateRandomMobile();

    // Set the generated values

    setDriverName(randomDriverName);

    setDriverMobile(randomDriverMobile);

    // Simulate sending OTP

    setTimeout(() => {

      // Call the onConfirm function to notify the parent component that booking is confirmed

      onConfirm();

    }, 2000);

  }, []);

  const generateRandomName = () => {

    // Random driver names list

    const driverNames = [

      "John Doe",

      "Jane Smith",

      "Michael Johnson",

      "Emily Brown",

    ];

    // Generate random index

    const randomIndex = Math.floor(Math.random() \* driverNames.length);

    // Return the random name

    return driverNames[randomIndex];

  };

  const generateRandomMobile = () => {

    // Generate random mobile number between 1000000000 and 9999999999

    return Math.floor(1000000000 + Math.random() \* 9000000000).toString();

  };

  return (

    <div className="booking-confirmation" style={styles.confirmationContainer}>

      <h2 style={styles.heading}>Booking Confirmed!</h2>

      <p style={styles.text}>OTP sent to your registered mobile number!</p>

      <p style={styles.text}>Your driver is {driverName}</p>

      <p style={styles.text}>Call your driver at {driverMobile}</p>

    </div>

  );

}

function SearchPage({ destination, onBookingConfirmed }) {

  const [cars, setCars] = useState([]);

  const [loading, setLoading] = useState(false);

  useEffect(() => {

    searchCars();

  }, []);

  const searchCars = () => {

    const dummyCars = [

      { id: 1, name: "Scooter", type: "Vespa", price: "Rs.100" },

      { id: 2, name: "Car", type: "Mini", price: "Rs.250" },

      { id: 3, name: "Car", type: "SUV", price: "Rs.450" },

    ];

    setCars(dummyCars);

  };

  const handleConfirmBooking = () => {

    setLoading(true);

    setTimeout(() => {

      setLoading(false);

      onBookingConfirmed();

    }, 4000);

  };

  return (

    <div className="search-page" style={styles.searchPage}>

      <h2 style={styles.heading}>Available Cars for {destination}</h2>

      <div className="cars-list" style={styles.carsList}>

        {cars.map((car) => (

          <div key={car.id} className="car" style={styles.car}>

            <h3 style={styles.carHeading}>{car.name}</h3>

            <p style={styles.carDetail}>Type: {car.type}</p>

            <p style={styles.carDetail}>Price: {car.price}</p>

            <button

              onClick={handleConfirmBooking}

              disabled={loading}

              style={styles.button}

            >

              Select Car

            </button>

          </div>

        ))}

      </div>

      {loading && <p style={styles.loading}>Loading...</p>}

    </div>

  );

}

function AL() {

  const [loggedIn, setLoggedIn] = useState(true);

  const [destination, setDestination] = useState("");

  const [bookingConfirmed, setBookingConfirmed] = useState(false);

  const handleSelectDestination = (selectedDestination) => {

    setDestination(selectedDestination);

  };

  const handleBookingConfirmed = () => {

    setBookingConfirmed(true);

  };

  return (

    <div>

      {loggedIn && !destination && (

        <DestinationPage onSelectDestination={handleSelectDestination} />

      )}

      {loggedIn && destination && !bookingConfirmed && (

        <SearchPage

          destination={destination}

          onBookingConfirmed={handleBookingConfirmed}

        />

      )}

      {bookingConfirmed && (

        <BookingConfirmation onConfirm={() => setBookingConfirmed(true)} />

      )}

    </div>

  );

}

function LocationSelection({ onSelectLocation }) {

  const [location, setLocation] = useState("");

  const handleSelectLocation = () => {

    onSelectLocation(location);

  };

  return (

    <div className="location-selection" style={styles.locationSelection}>

      <h2 style={styles.heading}>Select Pickup Location</h2>

      <input

        type="text"

        placeholder="Enter pickup location"

        value={location}

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

        style={styles.input}

      />

      <button onClick={handleSelectLocation} style={styles.button}>

        Select Pickup Location

      </button>

      <p style={styles.text}>Or</p>

      <button

        onClick={() =>

          navigator.geolocation.getCurrentPosition((position) =>

            onSelectLocation(

              `${position.coords.latitude}, ${position.coords.longitude}`

            )

          )

        }

        style={styles.button}

      >

        Use My Current Location

      </button>

    </div>

  );

}

function DestinationPage({ onSelectDestination }) {

  const [manualDestination, setManualDestination] = useState("");

  const handleSelectManualDestination = () => {

    if (manualDestination.trim() !== "") {

      onSelectDestination(manualDestination.trim());

    }

  };

  return (

    <div className="destination-page" style={styles.destinationPage}>

      <h2 style={styles.heading}>Select Destination</h2>

      <input

        type="text"

        placeholder="Enter destination"

        value={manualDestination}

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

        style={styles.input}

      />

      <button onClick={handleSelectManualDestination} style={styles.button}>

        Select Destination

      </button>

      <LocationSelection onSelectLocation={onSelectDestination} />

      <iframe

        src="https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!1d497513.50684670504!2d79.6179621829818!3d13.04685498122354!2m3!1f0!2f0!3f0!3m2!1i1024!2i768!4f13.1!3m3!1m2!1s0x3a5265ea4f7d3361%3A0x6e61a70b6863d433!2sChennai%2C%20Tamil%20Nadu!5e0!3m2!1sen!2sin!4v1713119314707!5m2!1sen!2sin"

        width="400"

        height="300"

        style={{ border: 0 }}

        allowFullScreen=""

        loading="lazy"

        referrerPolicy="no-referrer-when-downgrade"

      ></iframe>

    </div>

  );

}

const styles = {

  confirmationContainer: {

    backgroundColor: "#f0f0f0",

    padding: "20px",

    borderRadius: "10px",

    textAlign: "center",

  },

  heading: {

    color: "#333",

  },

  text: {

    color: "#666",

    marginBottom: "10px",

  },

  searchPage: {

    backgroundColor: "#f9f9f9",

    padding: "20px",

    borderRadius: "10px",

    textAlign: "center",

  },

  carsList: {

    marginTop: "20px",

  },

  car: {

    backgroundColor: "#fff",

    border: "1px solid #ddd",

    padding: "15px",

    borderRadius: "5px",

    marginBottom: "10px",

  },

  carHeading: {

    color: "#333",

    marginBottom: "5px",

  },

  carDetail: {

    color: "#666",

    marginBottom: "5px",

  },

  button: {

    backgroundColor: "#007bff",

    color: "#fff",

    border: "none",

    padding: "8px 16px",

    borderRadius: "5px",

    cursor: "pointer",

  },

  loading: {

    color: "#777",

    marginTop: "20px",

  },

  locationSelection: {

    backgroundColor: "#f9f9f9",

    padding: "20px",

    borderRadius: "10px",

    textAlign: "center",

  },

  input: {

    padding: "8px",

    borderRadius: "5px",

    marginBottom: "10px",

  },

  destinationPage: {

    textAlign: "center",

  },

};

export default AL;

***Login.jsx***

import React, { useState } from 'react';

import { useNavigate, Link } from 'react-router-dom';

import logo from './logo.png';  // Ensure the path to your logo is correct

function Login() {

    const navigate = useNavigate();

    const [user, setUser] = useState('');

    const [password, setPassword] = useState('');

    const handleSubmit = (event) => {

        event.preventDefault();

        console.log('Login:', user, password);

        // Place your login logic here

        navigate('/al'); // Redirect to AL page upon successful login

    };

    return (

        <div className="login-form">

            <img src={logo} className="login-logo" alt="App Logo" />

            <form onSubmit={handleSubmit}>

                <input type="text" value={user} onChange={(e) => setUser(e.target.value)} placeholder="Username" />

                <input type="password" value={password} onChange={(e) => setPassword(e.target.value)} placeholder="Password" />

                <button type="submit">Login</button>

            </form>

            <Link to="/register">Register</Link>

        </div>

    );

}

export default Login;

***Login.css***

.login-container {

    position: relative;

    padding-top: 40px;

    /\* Space for the logo \*/

    background: #ffffff;

    box-shadow: 0 2px 15px rgba(0, 0, 0, 0.1);

    border-radius: 8px;

    display: flex;

    flex-direction: column;

    align-items: center;

}

.login-logo {

    width: 80px;

    /\* Control the size of your logo \*/

    height: 80px;

    border-radius: 50%;

    /\* Makes the image circular \*/

    position: absolute;

    top: -40px;

    /\* Half of the height to move it up \*/

    /\* Match the background or adjust as needed \*/

    padding: 5px;

    /\* Optional: if you want some spacing around the image \*/

    box-shadow: 0 2px 10px rgba(0, 0, 0, 0.2);

}

.login-form {

    width: 100%;

    max-width: 320px;

    padding: 20px;

}

.login-form h2 {

    text-align: center;

    margin-bottom: 20px;

}

.login-form input {

    margin-bottom: 10px;

    padding: 10px;

    border: 1px solid #ccc;

    border-radius: 4px;

    width: calc(100% - 20px);

    /\* Adjust width for padding \*/

}

.login-button {

    background-color: #007bff;

    color: white;

    padding: 10px;

    border: none;

    border-radius: 4px;

    cursor: pointer;

    width: 100%;

    /\* Full width \*/

}

.login-button:hover {

    background-color: #0056b3;

}

***Logo.png***

******

***Register.jsx***

import React, { useState } from "react";

import { useNavigate } from "react-router-dom";

function Register() {

  const navigate = useNavigate();

  const [email, setEmail] = useState("");

  const [password, setPassword] = useState("");

  const [confirmPassword, setConfirmPassword] = useState("");

  const [errorMessage, setErrorMessage] = useState("");

  const handleSubmit = (event) => {

    event.preventDefault();

    if (password.length < 6) {

      setErrorMessage("Password must be at least 6 characters long.");

      return;

    }

    if (password !== confirmPassword) {

      setErrorMessage("Passwords do not match.");

      return;

    }

    console.log("User Registered:", email, password);

    navigate("/"); // Redirect to the login page

  };

  return (

    <div className="login-form">

      <h1>Register Here</h1>

      <form onSubmit={handleSubmit}>

        <input

          type="email"

          value={email}

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

          placeholder="Enter email"

        />

        <input type="text" placeholder="Enter your full name"></input>

        <input type="text" placeholder="Enter your full address"></input>

        <input type="text" placeholder="Enter your mobileno for OTP communication"></input>

        <input

          type="password"

          value={password}

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

          placeholder="Enter password"

        />

        <input

          type="password"

          value={confirmPassword}

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

          placeholder="Confirm password"

        />

        {errorMessage && <p style={{ color: "red" }}>{errorMessage}</p>}

        <button type="submit">Register</button>

      </form>

    </div>

  );

}

export default Register;

***Register.css***

/\* src/components/Register.css \*/

.register-container {

    display: flex;

    justify-content: center;

    align-items: center;

    height: 100vh;

}

.register-box {

    width: 100%;

    max-width: 360px;

    background: #ffffff;

    padding: 20px;

    box-shadow: 0 2px 15px rgba(0, 0, 0, 0.1);

    border-radius: 8px;

    display: flex;

    flex-direction: column;

}

.register-form input {

    width: 100%;

    padding: 10px;

    margin: 8px 0;

    box-sizing: border-box;

    /\* Includes padding and border in the element's total width and height \*/

    border: 1px solid #ccc;

    border-radius: 4px;

}

.register-button {

    background-color: #4CAF50;

    color: white;

    padding: 14px 20px;

    margin: 8px 0;

    border: none;

    border-radius: 4px;

    cursor: pointer;

    width: 100%;

}

.register-button:hover {

    background-color: #45a049;

}

h1{

    color:blue;

}

***SplashScreen.js***

import React, { useEffect, useState } from 'react';

import logo from './logo.png';  // Adjust the path as needed based on your directory structure

function SplashScreen() {

  const [visible, setVisible] = useState(true);

  useEffect(() => {

    const timer = setTimeout(() => {

      setVisible(false);

    }, 3000);  // Display for 3 seconds

    return () => clearTimeout(timer);

  }, []);

  return visible ? <div style={{ display: 'flex', justifyContent: 'center', alignItems: 'center', height: '100vh' }}>

    <img src={logo} alt="Logo" />

  </div> : null;

}

export default SplashScreen;

***SplashScreen.css***

.splash-container {

    width: 100%;

    height: 100%;

    /\* Full height of the viewport \*/

    display: flex;

    justify-content: center;

    align-items: center;

    background-color: #ffffff;

}

.logo {

    width: 150px;

    /\* Adjust based on your actual logo size \*/

    height: auto;

}