import React, { useState } from "react";

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

import axios from "axios";

import "./Register.css";

const Register = () => {

  const [step, setStep] = useState(1);

  const [otpVerified, setOtpVerified] = useState(false); // Ensure OTP is verified before registration

  const [formData, setFormData] = useState({

    admissionNumber: "",

    name: "",

    email: "",

    department: "",

    role: "Student",

    password: "",

    confirmPassword: "",

  });

  const [additionalData, setAdditionalData] = useState({});

  const [otp, setOtp] = useState("");

  const [error, setError] = useState("");

  const [files, setFiles] = useState({

    profilePicture: null,

    idCard: null,

    idProof: null,

  });

  const navigate = useNavigate();

  const handleChange = (e) => {

    setFormData({ ...formData, [e.target.name]: e.target.value });

  };

  const handleAdditionalChange = (e) => {

    setAdditionalData({ ...additionalData, [e.target.name]: e.target.value });

  };

  const handleFileChange = (e) => {

    const { name, files } = e.target;

    setFiles((prev) => ({ ...prev, [name]: files[0] }));

  };

  const sendOtp = async () => {

    try {

      const response = await axios.post("http://localhost:5000/api/send-otp", {

        admissionNumber: formData.admissionNumber,

        name: formData.name,

        email: formData.email,

      });

      alert(response.data.message);

      setStep(2);

    } catch (error) {

      setError(error.response?.data?.message || "Error sending OTP.");

    }

  };

  const verifyOtp = async () => {

    try {

      const response = await axios.post("http://localhost:5000/api/verify-otp", {

        email: formData.email,

        otp,

      });

      if (response.data.success) {

        setOtpVerified(true);

        setStep(3);

      } else {

        setError("Invalid OTP");

      }

    } catch (error) {

      setError(error.response?.data?.message || "OTP verification failed.");

    }

  };

  const handleSubmit = async (e) => {

    e.preventDefault();

    setError("");

    if (!otpVerified) {

      setError("OTP not verified. Please verify before registration.");

      return;

    }

    if (formData.password !== formData.confirmPassword) {

      setError("Passwords do not match.");

      return;

    }

    const formDataWithFiles = new FormData();

    for (const key in formData) {

      formDataWithFiles.append(key, formData[key]);

    }

    for (const key in additionalData) {

      formDataWithFiles.append(key, additionalData[key]);

    }

    if (files.idCard) formDataWithFiles.append("idCard", files.idCard);

    if (files.profilePicture) formDataWithFiles.append("profilePicture", files.profilePicture);

    if (files.idProof) formDataWithFiles.append("idProof", files.idProof);

    try {

      await axios.post("http://localhost:5000/api/register", formDataWithFiles);

      alert("Registration successful! You can now log in.");

      navigate("/login");

    } catch (error) {

      setError(error.response?.data?.message || "Something went wrong.");

    }

  };

  return (

    <div>

      <h2>Register</h2>

      {error && <p className="error-message">{error}</p>}

      {step === 1 && (

        <form onSubmit={(e) => { e.preventDefault(); sendOtp(); }}>

          <input type="text" name="admissionNumber" placeholder="Admission Number" required onChange={handleChange} />

          <input type="text" name="name" placeholder="Name" required onChange={handleChange} />

          <input type="email" name="email" placeholder="Email" required onChange={handleChange} />

          <input type="text" name="department" placeholder="Department" required onChange={handleChange} />

          <select name="role" value={formData.role} onChange={handleChange} required>

            <option value="Student">Student</option>

            <option value="Alumni">Alumni</option>

          </select>

          <button type="submit">Send OTP</button>

        </form>

      )}

      {step === 2 && (

        <form onSubmit={(e) => { e.preventDefault(); verifyOtp(); }}>

          <input type="text" placeholder="Enter OTP" value={otp} onChange={(e) => setOtp(e.target.value)} required />

          <button type="submit">Verify OTP</button>

        </form>

      )}

      {step === 3 && (

        <form onSubmit={handleSubmit}>

          <input type="password" name="password" placeholder="Create Password" required onChange={handleChange} />

          <input type="password" name="confirmPassword" placeholder="Confirm Password" required onChange={handleChange} />

          <input type="text" name="gender" placeholder="Gender" required onChange={handleAdditionalChange} />

          <input type="date" name="dob" placeholder="Date of Birth" required onChange={handleAdditionalChange} />

          {formData.role === "Alumni" && (

            <>

              <input type="text" name="currentJobTitle" placeholder="Current Job Title" required onChange={handleAdditionalChange} />

              <input type="text" name="companyName" placeholder="Company Name" required onChange={handleAdditionalChange} />

              <input type="text" name="industry" placeholder="Industry" required onChange={handleAdditionalChange} />

              <input type="number" name="yearsOfExperience" placeholder="Years of Experience" required onChange={handleAdditionalChange} />

              <input type="number" name="graduationYear" placeholder="Graduation Year" required onChange={handleAdditionalChange} />

              <input type="file" name="idProof" onChange={handleFileChange} required />

            </>

          )}

          {formData.role === "Student" && (

            <>

              <input type="text" name="ktuId" placeholder="KTU ID" required onChange={handleAdditionalChange} />

              <input type="number" name="yearOfAdmission" placeholder="Year of Admission" required onChange={handleAdditionalChange} />

              <input type="number" name="currentYear" placeholder="Current Year of Study" required onChange={handleAdditionalChange} />

              <input type="text" name="rollNumber" placeholder="Roll Number" required onChange={handleAdditionalChange} />

              <input type="number" name="graduationYear" placeholder="Graduation Year" required onChange={handleAdditionalChange} />

              <input type="file" name="idCard" onChange={handleFileChange} required />

            </>

          )}

          <input type="file" name="profilePicture" onChange={handleFileChange} required />

          <button type="submit">Complete Registration</button>

        </form>

      )}

    </div>

  );

};

export default Register;