import React, { useState } from 'react';

import axios from 'axios';

import '../../styles/Form.css';

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

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

import { GoogleLogin, GoogleOAuthProvider } from '@react-oauth/google';

import { useDispatch } from "react-redux";

import { loginUserAsync } from "../slices/loginSlice";

const clientId = "400425828105-3ml5b9n37ll8ddlghmjcc2e83345edn7.apps.googleusercontent.com";

const Form = () => {

const dispatch = useDispatch();

const [formData, setFormData] = useState({

userName: '',

password: '',

});

const [state, setState] = useState({

emailError: "",

passwordError: "",

});

const [logindata,setloginData]=useState(

localStorage.getItem("loginData")

?JSON.parse(localStorage.getItem("loginData")):null

)

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

const navigate = useNavigate()

const handleChange = (e) => {

const { name, value } = e.target;

setFormData((prevData) => ({

...prevData,

[name]: value,

}));

};

const handleSubmit = (e) => {

e.preventDefault();

const { userName, password } = formData;

setState({

...state,

emailError: userName === "" ? "Please enter your Username" : "",

passwordError: password === "" ? "Please enter your Password" : "",

});

if (userName === "" || password === "") {

return;

}

// Dispatch the async thunk to handle login

dispatch(loginUserAsync(formData)).then((action) => {

if (action.meta.requestStatus === 'fulfilled') {

navigate("/users");

}

});

};

const handleSuccess = async (response) => {

console.log(response , "handleSuccess")

const { credential } = response;

console.log( 'credential---------', credential);

try {

const loginResponse = await axios.post('http://localhost:5000/api/loginwithgoogle', { token: credential });

console.log(loginResponse, 'loginresponse');

const { status, data } = loginResponse;

if (status === 201 && data.token !== undefined && Object.keys(data.token).length) {

// setUser(data);

console.log("Login successful", data);

localStorage.setItem("token", loginResponse.data.token);

localStorage.setItem('user\_data', JSON.stringify(data));

localStorage.setItem("userId", loginResponse.data.userId);

navigate("/users");

}

} catch (error) {

console.error('Error during login:', error);

}

};

const handleError = () => {

console.log("error while login")

}

// end

return (

<div>

<h2>Login</h2>

{error && <p style={{ color: 'red' }}>{error}</p>}

<form onSubmit={handleSubmit}>

<div>

<label>Email:</label>

<input

type="userName"

name="userName"

value={formData.userName}

onChange={handleChange}

required

/>

</div>

<div>

<label>Password:</label>

<input

type="password"

name="password"

value={formData.password}

onChange={handleChange}

required

/>

</div>

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

<button type="button">

<GoogleOAuthProvider clientId={'400425828105-3ml5b9n37ll8ddlghmjcc2e83345edn7.apps.googleusercontent.com'}>

<GoogleLogin

onSuccess={(CredentialResponse)=>{

handleSuccess(CredentialResponse);

}}

onError={handleError}

/>

</GoogleOAuthProvider>

</button>

<div className="text-center">Don't have an account? <Link to="/register">Register</Link></div>

</form>

</div>

);

};

export default Form;