import React, { useState } from 'react';

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

const roleRoutes = {

    admin: '/admin',

    student: '/student',

    teacher: '/teacher',

    parent: '/parent',

    // Add more roles and their routes here

};

const Login = () => {

    const navigate = useNavigate();

    const [input, setInput] = useState({

        email: "",

        pwd: "",

    });

    const handleLogin = (e) => {

        e.preventDefault();

        const loggeduser = JSON.parse(localStorage.getItem('user'));

        if (loggeduser && loggeduser.email === input.email && loggeduser.pwd === input.pwd) {

            switch (loggeduser.role) {

                case "admin":

                    navigate('/admin');

                    break;

                case "student":

                    navigate('/student');

                    break;

                case "faculty":

                    navigate('/faculty');

                    break;

                case "hod":

                    navigate('/hod');

                    break;

                case "parent":

                    navigate('/parent');

                    break;

                default:

                    alert("Unrecognized role");

            }

        } else {

            alert("Wrong email id or password");

        }

    };

    return (

        <div className='page-height'>

            <div className='login-bg container-fluid d-flex justify-content-center align-items-center'>

                <div className="card border-0 container shadow-lg width-25">

                    <div className="card-body">

                        <div className='text-center'>

                            <img src="https://images.collegedunia.com/public/college\_data/images/logos/16860356593131894094954175992676362983714743865833242n.jpeg?h=71.17&w=71.17&mode=stretch" alt="Logo"/>

                        </div>

                        <div className="pt-4">

                            <form className="row g-3" onSubmit={handleLogin}>

                                <div className="col-12 col-sm-12 col-md-12 col-lg-12">

                                    <label htmlFor="validationDefault01" className="form-label"></label>

                                    <input name='email' value={input.email} onChange={(e) => setInput({...input, [e.target.name]: e.target.value})} type="text" className="form-control rounded-0" id="validationDefault01" placeholder='Email' required />

                                </div>

                                <div className="col-12 col-sm-12 col-md-12 col-lg-12">

                                    <label htmlFor="validationDefault02" className="form-label"></label>

                                    <input name='pwd' value={input.pwd} onChange={(e) => setInput({...input, [e.target.name]: e.target.value})} type="password" className="form-control rounded-0" id="validationDefault02" placeholder='Password' required />

                                </div>

                                <div className="col-12 text-center pt-3">

                                    <button className="btn btn-primary btn-text rounded-0 w-50" type="submit">Login</button>

                                </div>

                                <div className='col-12 text-center pt-3'>

                                    <h6>Please <Link to="/signup">Signup</Link> if you don't have an Account</h6>

                                </div>

                            </form>

                        </div>

                    </div>

                </div>

            </div>

        </div>

    );

}

export default Login;

import React, { useState } from 'react';

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

import axios from 'axios';

const Signup = () => {

    const [input, setInput] = useState({

        fname: "",

        lname: "",

        email: "",

        rollno: "",

        pwd: "",

        role: "student", // Default role

    });

    const [submitted, setSubmitted] = useState(false);

    const handleSubmit = (e) => {

        e.preventDefault();

        axios.post('http://localhost:5000/signup', input)

            .then(response => {

                console.log(response.data);

                setSubmitted(true);

            })

            .catch(error => {

                console.error('There was an error!', error);

            });

    };

    return (

        <div className='page-height'>

            <div className='login-bg container-fluid d-flex justify-content-center align-items-center'>

                <div className="card border-0 container shadow-lg width-25">

                    <div className="card-body">

                        <div className='text-center'>

                            <img className="pt-3" src="https://images.collegedunia.com/public/college\_data/images/logos/16860356593131894094954175992676362983714743865833242n.jpeg?h=71.17&w=71.17&mode=stretch" alt="Logo"/>

                        </div>

                        <form className="row g-3 py-5" onSubmit={handleSubmit}>

                            <div className="col-12 col-sm-6 col-md-6 col-lg-6">

                                <label htmlFor="validationDefault01" className="form-label"></label>

                                <input name='fname' value={input.fname} onChange={(e) => setInput({...input, [e.target.name]: e.target.value})} type="text" className="form-control rounded-0" id="validationDefault01" required placeholder='First Name' />

                            </div>

                            <div className="col-12 col-sm-6 col-md-6 col-lg-6">

                                <label htmlFor="validationDefault02" className="form-label"></label>

                                <input name='lname' value={input.lname} onChange={(e) => setInput({...input, [e.target.name]: e.target.value})} type="text" className="form-control rounded-0" id="validationDefault02" required placeholder='Last Name'/>

                            </div>

                            <div className="col-12 col-sm-6 col-md-6 col-lg-6">

                                <label htmlFor="validationDefault03" className="form-label"></label>

                                <input name='email' value={input.email} onChange={(e) => setInput({...input, [e.target.name]: e.target.value})} type="email" className="form-control rounded-0" id="validationDefault03" required placeholder='Email'/>

                            </div>

                            <div className="col-12 col-sm-6 col-md-6 col-lg-6">

                                <label htmlFor="validationDefault04" className="form-label"></label>

                                <input name='rollno' value={input.rollno} onChange={(e) => setInput({...input, [e.target.name]: e.target.value})} type="text" className="form-control rounded-0" id="validationDefault04" required placeholder='Roll Number'/>

                            </div>

                            <div className="col-12 col-sm-6 col-md-6 col-lg-6">

                                <label htmlFor="validationDefault05" className="form-label"></label>

                                <input name='pwd' value={input.pwd} onChange={(e) => setInput({...input, [e.target.name]: e.target.value})} type="password" className="form-control rounded-0" id="validationDefault05" required placeholder='Password'/>

                            </div>

                            <div className="col-12 col-sm-6 col-md-6 col-lg-6">

                                <label htmlFor="role" className="form-label"></label>

                                <select name='role' value={input.role} onChange={(e) => setInput({...input, [e.target.name]: e.target.value})} className="form-control rounded-0" id="role" required>

                                    <option value='catagory'></option>

                                    <option value="faculty">Faculty</option>

                                    <option value="hod">HOD</option>

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

                                    <option value="parent">Parent</option>

                                    <option value="admin">Admin</option>

                                </select>

                            </div>

                            <div className="col-12 text-center pt-4">

                                    <button className="btn btn-primary btn-text rounded-0 w-50" type="submit">Register</button>

                                </div>

                            {submitted && (

                                <div className="col-12">

                                    <div className="alert alert-success" role="alert">

                                        Form submitted successfully!

                                    </div>

                                </div>

                            )}

                            <div className='col-12 pt-3 text-center'>

                                <h6> <Link to="/">Login</Link> if you have already Registered!</h6>

                            </div>

                        </form>

                    </div>

                </div>

            </div>

        </div>

    );

}

export default Signup;

const express = require('express');

const bodyParser = require('body-parser');

const cors = require('cors');

const mysql = require('mysql');

const bcrypt = require('bcryptjs');

const app = express();

const PORT = process.env.PORT || 5000;

app.use(cors());

app.use(bodyParser.json());

// MySQL connection setup

const db = mysql.createConnection({

    host: 'localhost',

    user: 'root',

    password: 'Admin@123',

    database: 'studentprofile'

});

app.get('/', (req, res) => {

    res.send('Welcome to the API server');

});

// Connect to MySQL

db.connect(err => {

    if (err) {

        console.error('Error connecting to MySQL:', err);

        throw err;

    }

    console.log('MySQL connected');

});

// Route for signup

app.post('/signup', async (req, res) => {

    const { fname, lname, email, rollno, pwd, role } = req.body;

    const hashedPwd = await bcrypt.hash(pwd, 10);

    const user = { fname, lname, email, rollno, pwd: hashedPwd, role };

    db.query('INSERT INTO user SET ?', user, (err, result) => {

        if (err) {

            console.error('Error signing up user:', err);

            return res.status(400).json({ error: err });

        }

        res.json({ message: 'User registered!' });

    });

});

// Login endpoint

app.post('/login', (req, res) => {

    const { email, password } = req.body;

    if (!email || !password) {

        return res.status(400).json({ message: 'Email and password are required' });

    }

    // Query to find user by email

    const query = 'SELECT \* FROM users WHERE email = ?';

    connection.query(query, [email], (err, results) => {

        if (err) {

            console.error('Error executing query:', err);

            return res.status(500).json({ message: 'Internal server error' });

        }

        // Check if user with email exists

        if (results.length === 0) {

            return res.status(404).json({ message: 'User not found' });

        }

        const user = results[0];

        // Compare passwords

        bcrypt.compare(password, user.password, (bcryptErr, bcryptRes) => {

            if (bcryptErr) {

                console.error('Error comparing passwords:', bcryptErr);

                return res.status(500).json({ message: 'Internal server error' });

            }

            if (bcryptRes) {

                // Passwords match

                res.json({ message: 'Login successful!' });

            } else {

                // Passwords do not match

                res.status(401).json({ message: 'Invalid credentials' });

            }

        });

    });

});

// Start the server

app.listen(PORT, () => {

    console.log(`Server running on port ${PORT}`);

});

import React from 'react';

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

import Login from '../pages/login';

import Signup from '../pages/signup';

import Student from '../components/student';

import Admin from '../components/admin';

import Hod from '../components/hod';

import Parent from '../components/parent';

import Faculty from '../components/faculty';

import ProtectedRoute from '../Services/ProtectedRoute';

function routerApp() {

    return (

        <div>

            <Routes>

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

                <Route path="/signup" element={<Signup/>} />

                <Route path="/student" element={<Student/>} />

                <Route path="/admin" element={<Admin/>} />

                <Route path="/parent" element={<Parent/>} />

                <Route path="/hod" element={<Hod/>} />

                <Route path="/faculty" element={<Faculty/>} />

                {/\*Protect Route\*/}

                <Route path='/' element={<ProtectedRoute/>}>

                    <Route path='/' element={<Parent/>} />

                </Route>

            </Routes>

        </div>

    );

}

export default routerApp;