import React from 'react';

const validEmailRegex = RegExp(

  /^(([^<>()\[\]\.,;:\s@\"]+(\.[^<>()\[\]\.,;:\s@\"]+)\*)|(\".+\"))@(([^<>()[\]\.,;:\s@\"]+\.)+[^<>()[\]\.,;:\s@\"]{2,})$/i

);

const validateForm = errors => {

  let valid = true;

  Object.values(errors).forEach(val => val.length > 0 && (valid = false));

  return valid;

};

export default class DataForm extends React.Component {

  constructor(props) {

    super(props);

    this.state = {

      fullName: null,

      email: null,

      password: null,

      errors: {

        fullName: '',

        email: '',

        password: '',

      }

    };

  }

  handleChange = (event) => {

    event.preventDefault();

    const { name, value } = event.target;

    let errors = this.state.errors;

    switch (name) {

      case 'fullName':

        errors.fullName =

          value.length < 5

            ? 'Full Name must be at least 5 characters long!'

            : '';

        break;

      case 'email':

        errors.email =

          validEmailRegex.test(value)

            ? ''

            : 'Email is not valid!';

        break;

      case 'password':

        errors.password =

          value.length < 8

            ? 'Password must be at least 8 characters long!'

            : '';

        break;

      default:

        break;

    }

    this.setState({errors, [name]: value});

  }

  handleSubmit = (event) => {

    event.preventDefault();

    if(validateForm(this.state.errors)) {

      console.info('Valid Form')

    }else{

      console.error('Invalid Form')

    }

  }

  render() {

    const {errors} = this.state;

    return (

      <div className='wrapper'>

        <div className='form-wrapper'>

          <h2>Create Account</h2>

          <form onSubmit={this.handleSubmit} noValidate>

            <div className='fullName'>

              <label htmlFor="fullName">Full Name</label>

              <input type='text' name='fullName' onChange={this.handleChange} noValidate />

              {errors.fullName.length > 0 &&

                <span className='error'>{errors.fullName}</span>}

            </div>

            <div className='email'>

              <label htmlFor="email">Email</label>

              <input type='email' name='email' onChange={this.handleChange} noValidate />

              {errors.email.length > 0 &&

                <span className='error'>{errors.email}</span>}

            </div>

            <div className='password'>

              <label htmlFor="password">Password</label>

              <input type='password' name='password' onChange={this.handleChange} noValidate />

              {errors.password.length > 0 &&

                <span className='error'>{errors.password}</span>}

            </div>

            <div className='submit'>

              <button>Create</button>

            </div>

          </form>

        </div>

      </div>

    );

  }

}