Tutorial : https://www.mongodb.com/languages/mern-stack-tutorial

Create React App :

* npx create-react-app client
* cd client
* npm install bootstrap react-router-dom
* empty src folder
  + rm src/\*\*/\*
* Create new files
  + touch src/index.js src/App.js
* Index.js

import React from "react";

import ReactDOM from "react-dom";

import App from "./App";

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

ReactDOM.render(

<React.StrictMode>

<BrowserRouter>

<App />

</BrowserRouter>

</React.StrictMode>,

document.getElementById("root")

);

* create a *components* folder
  + mkdir src/components
  + (cd src/components &&
  + Create frontend files for each API
    - touch create.js edit.js navbar.js userList.js)
    - create.js
* import React, { useState } from "react";
* import { useNavigate } from "react-router";
* export default function Create() {
* const [form, setForm] = useState({
* name: "",
* position: "",
* level: "",
* });
* const navigate = useNavigate();
* // These methods will update the state properties.
* function updateForm(value) {
* return setForm((prev) => {
* return { ...prev, ...value };
* });
* }
* // This function will handle the submission.
* async function onSubmit(e) {
* e.preventDefault();
* // When a post request is sent to the create url, we'll add a new record to the database.
* const newPerson = { ...form };
* await fetch("http://localhost:8000/users/addUser", {
* method: "POST",
* headers: {
* "Content-Type": "application/json",
* },
* body: JSON.stringify(newPerson),
* })
* .catch(error => {
* window.alert(error);
* return;
* });
* setForm({ first\_name: "", last\_name: "", user\_name: "" ,email:"",password:""});
* navigate("/");
* }
* // This following section will display the form that takes the input from the user.
* return (
* <div>
* <h3>Create New Record</h3>
* <form onSubmit={onSubmit}>
* <div className="form-group">
* <label htmlFor="first\_name">First Name</label>
* <input
* type="text"
* className="form-control"
* id="first\_name"
* value={form.first\_name}
* onChange={(e) => updateForm({ first\_name: e.target.value })}
* />
* </div>
* <div className="form-group">
* <label htmlFor="last\_name">Last Name</label>
* <input
* type="text"
* className="form-control"
* id="last\_name"
* value={form.last\_name}
* onChange={(e) => updateForm({ last\_name: e.target.value })}
* />
* </div>
* <div className="form-group">
* <label htmlFor="user\_name">User Name</label>
* <input
* type="text"
* className="form-control"
* id="user\_name"
* value={form.user\_name}
* onChange={(e) => updateForm({ user\_name: e.target.value })}
* />
* </div>
* <div className="form-group">
* <label htmlFor="email">Email</label>
* <input
* type="email"
* className="form-control"
* id="email"
* value={form.email}
* onChange={(e) => updateForm({ email: e.target.value })}
* />
* </div>
* <div className="form-group">
* <label htmlFor="password">Password</label>
* <input
* type="password"
* className="form-control"
* id="password"
* value={form.password}
* onChange={(e) => updateForm({ password: e.target.value })}
* />
* </div>
* <div className="form-group">
* <input
* type="submit"
* value="Create person"
* className="btn btn-primary"
* />
* </div>
* </form>
* </div>
* );
* }
  + - Edit.js
* import React, { useState, useEffect } from "react";
* import { useParams, useNavigate } from "react-router";
* export default function Edit() {
* const [form, setForm] = useState({
* first\_name: "",
* last\_name: "",
* user\_name: "",
* email: "",
* password: "",
* records: [],
* });
* const params = useParams();
* const navigate = useNavigate();
* useEffect(() => {
* async function fetchData() {
* const id = params.id.toString();
* const response = await fetch(`http://localhost:8000/users/${params.id.toString()}`);
* if (!response.ok) {
* const message = `An error has occurred: ${response.statusText}`;
* window.alert(message);
* return;
* }
* const record = await response.json();
* if (!record) {
* window.alert(`Record with id ${id} not found`);
* navigate("/");
* return;
* }
* setForm(record);
* }
* fetchData();
* return;
* }, [params.id, navigate]);
* // These methods will update the state properties.
* function updateForm(value) {
* return setForm((prev) => {
* return { ...prev, ...value };
* });
* }
* async function onSubmit(e) {
* e.preventDefault();
* const editedPerson = {
* first\_name: form.first\_name,
* last\_name: form.last\_name,
* user\_name: form.user\_name,
* email: form.email,
* password: form.password,
* };
* // This will send a post request to update the data in the database.
* await fetch(`http://localhost:8000/users/updateUser/${params.id}`, {
* method: "PUT",
* body: JSON.stringify(editedPerson),
* headers: {
* 'Content-Type': 'application/json'
* },
* });
* navigate("/");
* }
* // This following section will display the form that takes input from the user to update the data.
* return (
* <div>
* <h3>Update Record</h3>
* <form onSubmit={onSubmit}>
* <div className="form-group">
* <label htmlFor="first\_name">First Name: </label>
* <input
* type="text"
* className="form-control"
* id="first\_name"
* value={form.first\_name}
* onChange={(e) => updateForm({ first\_name: e.target.value })}
* />
* </div>
* <div className="form-group">
* <label htmlFor="last\_name">Position: </label>
* <input
* type="text"
* className="form-control"
* id="last\_name"
* value={form.last\_name}
* onChange={(e) => updateForm({ last\_name: e.target.value })}
* />
* </div>
* <div className="form-group">
* <label htmlFor="user\_name">User Name</label>
* <input
* type="text"
* className="form-control"
* id="user\_name"
* value={form.user\_name}
* onChange={(e) => updateForm({ user\_name: e.target.value })}
* />
* </div>
* <div className="form-group">
* <label htmlFor="email">Email</label>
* <input
* type="email"
* className="form-control"
* id="email"
* value={form.email}
* onChange={(e) => updateForm({ email: e.target.value })}
* />
* </div>
* <div className="form-group">
* <label htmlFor="password">Password</label>
* <input
* type="password"
* className="form-control"
* id="password"
* value={form.password}
* onChange={(e) => updateForm({ password: e.target.value })}
* />
* </div>
* <div className="form-group">
* <input
* type="submit"
* value="Update Record"
* className="btn btn-primary"
* />
* </div>
* </form>
* </div>
* );
* }
  + - UserList
* import React, { useEffect, useState } from "react";
* import { Link } from "react-router-dom";
* const user = (props) => (
* <tr>
* <td>{props.user.name}</td>
* <td>{props.user.position}</td>
* <td>{props.user.level}</td>
* <td>
* <Link className="btn btn-link" to={`/edit/${props.user.\_id}`}>Edit</Link> |
* <button className="btn btn-link"
* onClick={() => {
* props.deleteuser(props.user.\_id);
* }}
* >
* Delete
* </button>
* </td>
* </tr>
* );
* export default function userList() {
* const [users, setusers] = useState([]);
* // This method fetches the users from the database.
* useEffect(() => {
* async function getusers() {
* const response = await fetch(`http://localhost:8000/users/`);
* if (!response.ok) {
* const message = `An error occurred: ${response.statusText}`;
* window.alert(message);
* return;
* }
* const users = await response.json();
* setusers(users);
* }
* getusers();
* return;
* }, [users.length]);
* // This method will delete a user
* async function deleteuser(id) {
* await fetch(`http://localhost:8000/users/deleteUser/${id}`, {
* method: "DELETE"
* });
* const newusers = users.filter((el) => el.\_id !== id);
* setusers(newusers);
* }
* // This method will map out the users on the table
* function userList() {
* return users.map((user) => {
* return (
* <user
* user={user}
* deleteuser={() => deleteuser(user.\_id)}
* key={user.\_id}
* />
* );
* });
* }
* // This following section will display the table with the users of individuals.
* return (
* <div>
* <h3>user List</h3>
* <table className="table table-striped" style={{ marginTop: 20 }}>
* <thead>
* <tr>
* <th>First Name</th>
* <th>Last Name</th>
* <th>Username</th>
* <th>Email</th>
* <th>Password</th>
* <th>Action</th>
* </tr>
* </thead>
* <tbody>{userList()}</tbody>
* </table>
* </div>
* );
* }
  + - Navbar.js
* import React from "react";
* // We import bootstrap to make our application look better.
* import "bootstrap/dist/css/bootstrap.css";
* // We import NavLink to utilize the react router.
* import { NavLink } from "react-router-dom";
* // Here, we display our Navbar
* export default function Navbar() {
* return (
* <div>
* <nav className="navbar navbar-expand-lg navbar-light bg-light">
* <NavLink className="navbar-brand" to="/">
* <img style={{"width" : 25 + '%'}} src="https://d3cy9zhslanhfa.cloudfront.net/media/3800C044-6298-4575-A05D5C6B7623EE37/4B45D0EC-3482-4759-82DA37D8EA07D229/webimage-8A27671A-8A53-45DC-89D7BF8537F15A0D.png"></img>
* </NavLink>
* <button
* className="navbar-toggler"
* type="button"
* data-toggle="collapse"
* data-target="#navbarSupportedContent"
* aria-controls="navbarSupportedContent"
* aria-expanded="false"
* aria-label="Toggle navigation"
* >
* <span className="navbar-toggler-icon"></span>
* </button>
* <div className="collapse navbar-collapse" id="navbarSupportedContent">
* <ul className="navbar-nav ml-auto">
* <li className="nav-item">
* <NavLink className="nav-link" to="/create">
* Create User
* </NavLink>
* </li>
* </ul>
* </div>
* </nav>
* </div>
* );
* }
* App.js
* import React from "react";
* // We use Route in order to define the different routes of our application
* import { Route, Routes } from "react-router-dom";
* // We import all the components we need in our app
* import Navbar from "./components/navbar";
* import RecordList from "./components/recordList";
* import Edit from "./components/edit";
* import Create from "./components/create";
* const App = () => {
* return (
* <div>
* <Navbar />
* <Routes>
* <Route exact path="/" element={<UserList />} />
* <Route path="/edit/:id" element={<Edit />} />
* <Route path="/create" element={<Create />} />
* </Routes>
* </div>
* );
* };
* export default App;