REPASO 1 RICK AND MORTY

npm create vite@latest repaso1

npm i react-router-dom@5.3.0

npm i redux react-redux redux-thunk axios

**VAMOS A UTILIZAR LA API DE RICK AND MORTY, VERSION REST DONDE TENEMOS CHARECTERS, LOCATIONS Y EPISODES.**

**HACEMOS UN GET ALL CHARACTERS**

https://rickandmortyapi.com/api/character

**VAMOS A CREAR UN NAV QUE SE VA A VER EN TODAS LAS PAGINAS Y UNA SECCION PRINCIPAL Q SE VA VIENDO EN UNA RUTA O EN LA OTRA. EN MI HOME VOY A TENER TODO EL LISTADO DE CHARACTERS DE RICK AND MORTIS, OTRA PAGINA NOS LLEVA AL DETALLE DEL PERSONAJE.**

**COMENZAMOS:**

**EN MAIN.JSX AGREGAMOS BROWSERROUTER**

import React from 'react'

import ReactDOM from 'react-dom/client'

import { BrowserRouter } from 'react-router-dom'

import App from './App'

import './index.css'

ReactDOM.createRoot(document.getElementById("root")).render(

  <React.StrictMode>

    <BrowserRouter>

      <App />

    </BrowserRouter>

  </React.StrictMode>

);

**CREO CARPETA VIEWS QUE SON LAS “PAGINAS” QUE VOY A TENER, CON LOS ARCHIVOS:**

**CHARACTERDETAIL.JSX**

**CHARACTERS.JSX**

**HOME.JSX**

**EN CADA ARCHIVO.JSX CREO MIS FUNCIONES X AHORA VACIAS**

export default function CharacterDetail(params) {};

**EN VIEWS/HOME.JSX**

import { Component } from "react";

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

//COMPONENTE FUNCIONAL

export default function Home() {

  return (

    <main>

      <h1>Hola 👋🏻</h1>

      <section>

        <Link to="/characters">Ver los personajes de Rick and Morty</Link>

      </section>

    </main>

  );

}

// COMPONENTE TIPO CLASE

export class HomeClass extends Component {

  render() {

    return (

      <main>

        <h1>Hola 👋🏻</h1>

        <section>

          <Link to="/characters">Ver los personajes de Rick and Morty</Link>

        </section>

      </main>

    );

  }

}

**CREO CARPETA COMPONENTS**

**CREO ARCHIVO COMPONENTS/CHARACTERS.JSX**

import { Component } from "react";

export default function Character(props) {

  const { character, navigate } = props;

  const { id, name, image } = character;

  return (

    <article>

      <img src={image} alt="" />

      <span>{name}</span>

      <button onClick={() => navigate(id)}></button>

    </article>

  );

}

export class CharacterClass extends Component {

  render() {

    const { character, navigate } = props;

    const { id, name, image } = character;

    return (

      <article>

        <img src={image} alt="" />

        <span>{name}</span>

        <button onClick={() => navigate(id)}></button>

      </article>

    );

  }

}

**EN APP.JSX AGREGO LAS RUTAS:**

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

import "./App.css";

import Home from "./views/Home";

import Characters from "./views/Characters";

import CharacterDetail from "./views/CharacterDetail";

function App() {

  return (

    <div className="App">

      <Route exact path="/" component={Home} />

      <Route exact path="/characters" component={Characters} />

      <Route exact path="/characters/:id" component={CharacterDetail} />

    </div>

  );

}

export default App;

**CUANDO INSPECCIONO LAS PROPS QUE ENVIA ROUTE COMPONENT REACT ROUTER MANDA POR PROPS HISTORY, LOCATION Y MATCH. EN HISTORY TENGO UN METODO QUE SE LLAMA PUSH, ASI QUE EN CHARACTERS VOY A PODER SACAR EL METODO PUSH Y ESE METODO PUSH ES EL QUE VOY A ARMAR PARA HANDLENAVIGATE Q RECIBE UN ID Y CUANDO YO RENDERICE LOS CHARACTERS LE VOY A PASER EL HANDLENAVIGATE EN LA PROPIEDAD NAVIGATE Q DIJE Q IBA A RECIBIR.**

**CHARACTERS RENDERIZA LOS CHARACTER CON UN MAP.**

**ESO LO USO EN MI VIEW/CHARACTERS**

import { Component } from "react";

export default function Characters(props) {

  const {

    history: { push }, //viene de route-dom. Me lleva a esa ruta. Trabaja como el Link

  } = props;

  const handleNavigate = (id) => push(`/characters/${id}`);

  return <div></div>;

}

export class CharactersClass extends Component {}

**Armo la estructura de redux**

****

**EN MIS ACTION-TYPES**

export const SET\_CHARACTERS = "characters/set";

export const SET\_APP\_LOADING = "app/loading/set";

**EN ACTIONS/INDEX.JS**

import axios from "axios";

import { SET\_CHARACTERS } from "../actions/action-types";

const setCharacters = (payload) => ({ type: SET\_CHARACTERS, payload });

export const clearCharacters = () => (dispatch) => dispatch(setCharacters([]));

export function getCharactersFromAPI() {

  return async (dispatch) => {

    const response = await axios.get(

      "https://rickandmortyapi.com/api/character"

    );

    const {

      data: { results },

    } = response;

    dispatch(setCharacters(results));

  };

}

**EN REDUCERS/CHARACTERS.JS**

import { SET\_CHARACTERS } from "../actions/action-types";

const initialState = [];

export default function characters(state = initialState, { type, payload }) {

  switch (type) {

    case SET\_CHARACTERS:

      return payload;

    default:

      return state;

  }

}

**EN REDUCERS/APP.JS**

import { SET\_CHARACTERS } from "../actions/action-types";

const initialState = {

  loading: false,

};

export default function app(state = initialState, { type, payload }) {

  switch (type) {

    case SET\_CHARACTERS:

      return payload;

    default:

      return state;

  }

}

**EN REDUCERS/INDEX.JS**

import { combineReducers } from "redux";

import app from "./app";

import characters from "./characters";

const reducer = combineReducers({ app, characters });

export default reducer;

**EN STORE/INDEX.JS**

import { applyMiddleware, compose, createStore } from "redux";

import thunk from "redux-thunk";

import reducer from "./reducers";

const composeEnhancers = window.\_\_REDUX\_DEVTOOLS\_EXTENSION\_COMPOSE\_\_ || compose;

const store = createStore(reducer, composeEnhancers(applyMiddleware(thunk)));

export default store;

**EN MAIN.JSX**

import React from "react";

import ReactDOM from "react-dom/client";

import { Provider } from "react-redux";

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

import App from "./App";

import "./index.css";

import store from "./store";

ReactDOM.createRoot(document.getElementById("root")).render(

  <React.StrictMode>

    <BrowserRouter>

      <Provider store={store}>

        <App />

      </Provider>

    </BrowserRouter>

  </React.StrictMode>

);

**En views-Characters.jsx**

import { useEffect } from "react";

import { useSelector,useDispatch } from "react-redux";

import Character from "../components/Characters";

import store from "../store";

import { getCharactersFromAPI } from "../store/actions";

export default function Characters(props) {

  // const { history } = props;

  // const { push } = history;

  const {

    history: { push },

  } = props;

  const { characters } = useSelector((state) => state);

  const dispatch = useDispatch()

  useEffect(() => {

    dispatch(getCharactersFromAPI())

  },[dispatch])

  const handleNavigate = (id) => push(`/characters/${id}`);

  return (

    <main>

      {characters.map((ch) => (

        <Character

          key={ch.id}

          character={ch}

          navigate={handleNavigate}

        />

      ))}

    </main>

  );

}

**EN COMPONENT/CHARACTERS.JSX y ENTRAR A REACT ICONS Y BUSCAR UN ICONO NAVIGATE**

import { Component } from "react";

import { MdNavigateNext } from "react-icons/md";

export default function Character(props) {

  const { character, navigate } = props;

  const { id, name, image } = character;

  return (

    <article>

      <img src={image} alt="" />

      <span>{name}</span>

      <button onClick={() => navigate(id)}><MdNavigateNext/></button>

    </article>

  );

}

export class CharacterClass extends Component {

  render() {

    const { character, navigate } = props;

    const { id, name, image } = character;

    return (

      <article>

        <img src={image} alt="" />

        <span>{name}</span>

        <button onClick={() => navigate(id)}><MdNavigateNext/></button>

      </article>

    );

  }

}

**APLICO ALGO DE ESTILOS**

import { Component } from "react";

import { MdNavigateNext } from "react-icons/md";

export default function Character(props) {

  const { character, navigate } = props;

  const { id, name, image } = character;

  return (

    <article

      style={{

        display: "flex",

        alignItems: "center",

        justifyContent: "space-between",

        margin: "0.25rem 0",

        gap: "1rem",

        border: "solid 1px #f0f0f0",

        borderRadius: "0.5rem",

        overflow: "hidden",

        width: '100%',

        maxWidth: '50rem',

      }}

    >

      <div style={{ display: "flex", alignItems: "center", gap:'1rem' }}>

        <img src={image} alt="" style={{ width: "5rem", heigh: "5rem" }} />

        <span>{name}</span>

      </div>

      <button onClick={() => navigate(id)} style={{ margin: "1rem" }}>

        <MdNavigateNext />

      </button>

    </article>

  );

}

export class CharacterClass extends Component {

  render() {

    const { character, navigate } = props;

    const { id, name, image } = character;

    return (

      <article>

        <img src={image} alt="" />

        <span>{name}</span>

        <button onClick={() => navigate(id)}>

          <MdNavigateNext />

        </button>

      </article>

    );

  }

}

**En Views/CharacterDetail.jsx**

import axios from "axios";

import { useEffect, useState } from "react";

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

export default function CharacterDetail() {

  const { id } = useParams();

  const [state, setState] = useState();

  useEffect(() => {

    axios

      .get(`https://rickandmortyapi.com/api/character/${id}`)

      .then((response) => setState(response.data));

  }, [id]);

  return (

    <div>

      <img src={state.image} alt="" />

      <h1>{state.name}</h1>

    </div>

  )

}

**Itau -> angelica maria Hernandez 3176352727 angelica.hernandez@itau.co**

**ACA VOY**