

React.JS

Introducción

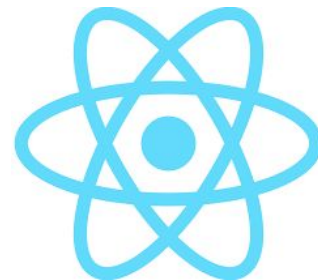
Módulo 04

React Redux Provider

React Redux

Redux es una librería que no depende de ninguna otra. Podemos usar Redux con Angular, JQuery, etc. Para conectarse con React, necesitamos el paquete NPM **react-redux**. Este paquete provee a React dos componentes principales para realizar la conexión con Redux:

- El componente **Provider**, que provee acceso al store a toda la aplicación
- El componente **connect()**, que permite a un componente concreto el acceso al estado actual y al método **dispatch()** de Redux para despachar acciones, a través de sus props.



Provider

```
import React from "react";

import { Provider } from "react-redux";

import store from "../ruta/al/store";

function App() {
  return (
    <div>
      <Provider store={store}>
        <div>
          Aplicacion aqui
        </div>
      </Provider>
    </div>
  );
}

export default App;
```

El primer componente que vamos a ver es el ***Provider***. Este componente es parecido al *BrowserRouter*. El *BrowserRouter* proporcionaba el contexto necesario para que el Router funcionara. De forma análoga, el *Provider* ofrece el contexto necesario a todos sus hijos el acceso al Store pasado en la propiedad 'store'.

Contexto

En React.Js podemos evitar tener que pasar el mismo dato mediante las Props, a través de todo el árbol de componentes mediante la API Context.

Más información en

<https://es.reactjs.org/docs/context.html>

NOTA: La API Context, al igual que los Middleware de Redux son contenidos que se ven en el [Curso de React.Js Avanzado](#)



¡Muchas gracias!

¡Sigamos trabajando!