**React Redux**

**Redux**: A state management library for Javascript apps.

* Centralizes application’s state
* Makes data flow transparent and predictable

|  |  |
| --- | --- |
| **REDUX PROS** | **REDUX CONS** |
| Predictable state changes | Complexity |
| Centralized state | Verbosity |
| Easy debugging |  |
| Preserve page state |  |
| Undo/redo |  |
| Ecosystem of add-ons |  |

**Higher Order Function** is a function which takes function as an argument or return function or both.

**Pure Function** every time it returns same result for the same input.

Pure function cannot change or mutate the arguments

Benefits of Pure Function

* Self-documenting
* Easily testable
* Concurrency (run parallel)
* Cacheable (refer previous result)

**Immutability** Once created, cannot be changed

If want to change object, take a copy then change that copied object.

Use spread operator to update objects (spread operator is a shallow copy).

Libraries for immutability(Immutable, Immer, Mori)

|  |  |
| --- | --- |
| **IMMUTABILITY PROS** | **IMMUTABILITY CONS** |
| Predictability | Performance |
| Faster Change Detection | Memory overhead |
| Concurrency |  |

**Redux Architecture**

Redux is built on functional programming

In redux we store our application state inside single javascript object called **store**. It is accessible by all parts of UI.

We cannot directly modify or mutate the **store**.

**Reducer** is used to update the **store.**

**Action** describes what to update.

State

Store

Reducer

Actions

UI

defines

contains

updates

Sent to

triggers

****STORE:**** A Store is a place where the entire state of your application lists. It manages the status of the application and has a dispatch(action) function.

****ACTION:**** Action is sent or dispatched from the view which are payloads that can be read by Reducers. It is a pure object created to store the information of the user's event. It includes information such as type of action, time of occurrence, location of occurrence, its coordinates, and which state it aims to change.

****REDUCER:**** Reducer read the payloads from the actions and then updates the store via the state accordingly. It is a pure function to return a new state from the initial state.

**Steps To Build Redux App**

* Design the store
* Define the actions
* Create a reducer
* Set up the store

Cake Conatiner

Features:

Maintain global state variable ‘numOfCakes’

Types: BUY\_CAKE

Action: buyCake

Reducer: cakeReducer

Ice Cream Conatiner

Features:

Maintain global state variable ‘numOfIceCreams’

Types: BUY\_ICE\_CREAM

Action: buyIceCream

Reducer: iceCreamReducer

npx create-react-app react-redux-sample

npm install redux react-redux

npm i @reduxjs/toolkit

npm start

Install redux devtools

First add redux devtool extension to browser

npm install --save redux-devtools-extension

Configure store.js file for redux devtools

Github:

<https://github.com/Fakhre-Alam-Hub/react-redux-sample>