

C/Henrik Høltzer Zealand 24.02.2020

Redux – Vanilla JS (Counter App - Overview )

**1**

**1**

**0**

getState()

notify()

subscribe()

**render()**

Callback function

**count 0**

State

**dispatch(action)**

**countReducer(count, action)**

Reducers

Store

**decrement()**

**Increment()**

Action Creator

**{ type: ‘DECREMENT’}**

**{ type: ‘INCREMENT’}**

Action

<h1>Counter</h1>

    <div>

            <button

                class="ui button primary"

                onClick="store.dispatch(increment())"

            >

                Increment

            </button>

            <button

                class="ui button primary"

                onClick="store.dispatch(decrement())"

            >

                Decrement

            </button>

            Current count: <span id="value">0</span>

        </div>

//Action creators

const increment = () => ({ type: 'INCREMENT'});
const decrement = () => ({ type: 'DECREMENT'});

//Reducers

const countReducer = (count=0, action) => {

    if (action.type === 'INCREMENT') { return count + 1; }

    if (action.type === 'DECREMENT') { return count - 1; }

    return count;

};

//Store

const store = Redux.createStore(countReducer);

//update html

function render(){

   document.getElementById("value").innerHTML = store.getState();

}

//Subscribe on changes, e.g. the callback-function render() is invoked when the store is updated

store.subscribe(render);