BROCHILL

REDUX TOOLKIT

Student

Hema vardhan reddy

2022 April 24

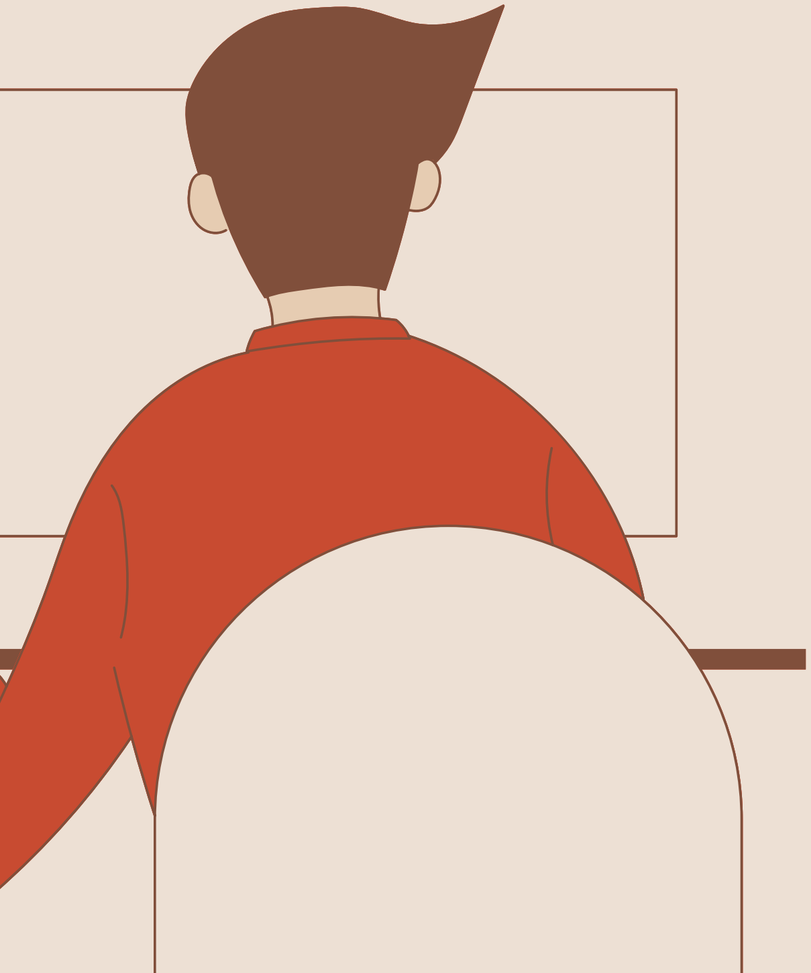
Final Defense

NAME OF
INSTITUTION OR
UNIVERSITY

2022 April 24

Table of Contents

Final Defense



what is redux toolkit?

- redux toolkit is the official,opinionated batteries included toolkit for effecial redux development
- it is a standard way to write redux logic in your application
- it abstracts hard parts of redux and gives development experiance

why not redux?

- configuring redux in our application seems complicated
- a lot of packages need to be installed to get redux functionality
- redux need too much boiler plate code

how to connect react&rkt

react and redux toolkit work both independently but we use react-redux module to combine them both

- react is UI maker
- reduxToolkit is state manager
- react-redux is linking place for both

redux-principles

lets say there is a cake shop->store and shopkeeper->reducer and customer->actionCreator

then customers request -> action

detailed explanation

there is a cake shop here customer cant directly take the cake from shelf -› instead he should emit an action like telling shopkeeper to give the cake now shopkeeper will start to reduce the cake from shelf/store and give it back to customer

- redux store is what holds our application state
- action describes what happened at store
- reducer handles the action and updates the state

three principles

1. global state of your application is stored as object in a single store->maintain a single state object to manage whole application using a store

- eg:

  in case of cake shop state is no,of cakes

  {

  num_of_cakes : 10,

  other states

  }

whole applications state is stored in one obj

2nd principle

the only way to change the state
is to emit/dispatch an action
-› action is a object which
explains what happened

- in order to update a state we need to let redux know about it
- we cannot directly update a state
- to let redux know we use action
- action is an object which tells what to update

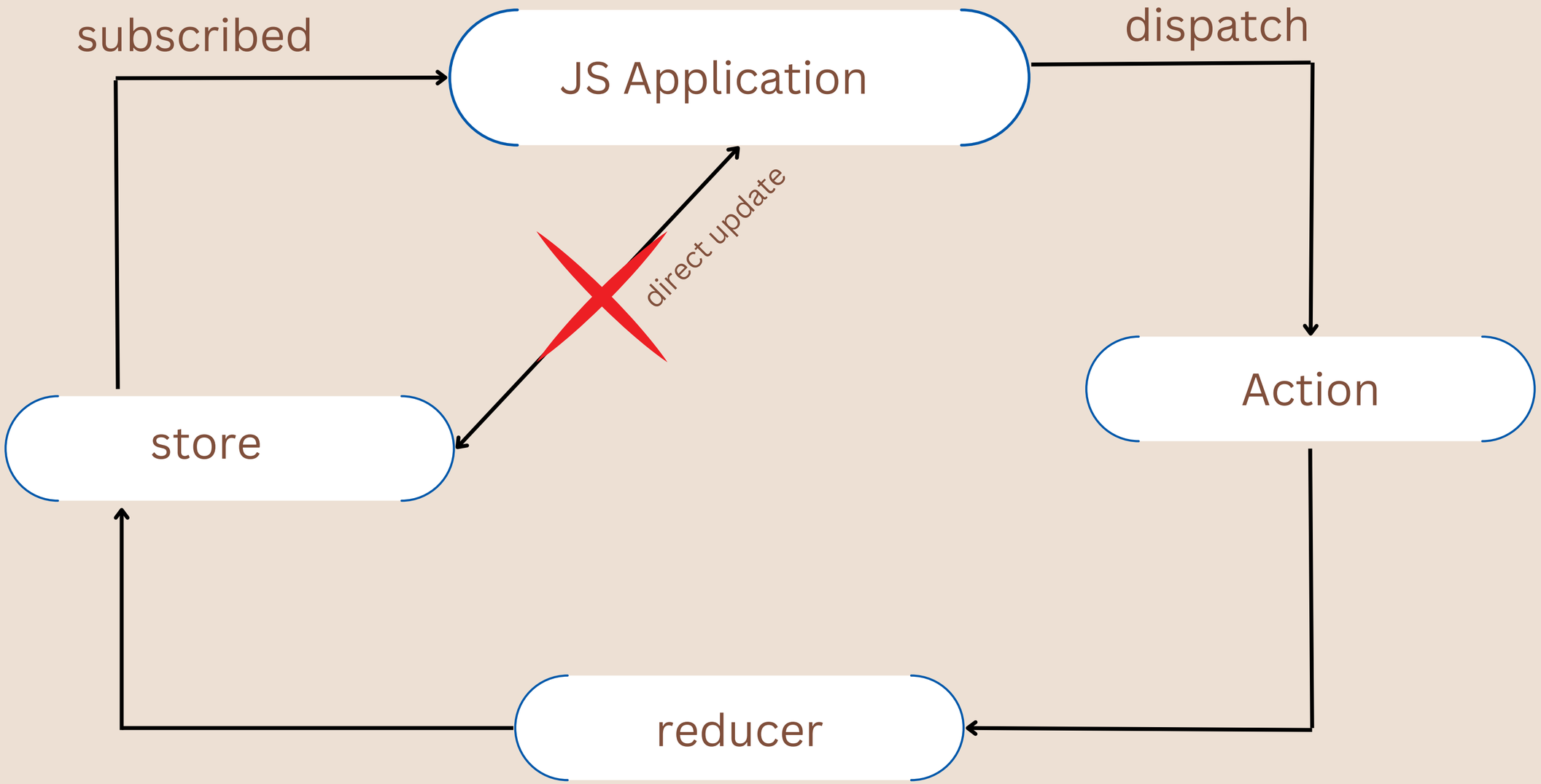
3rd principle

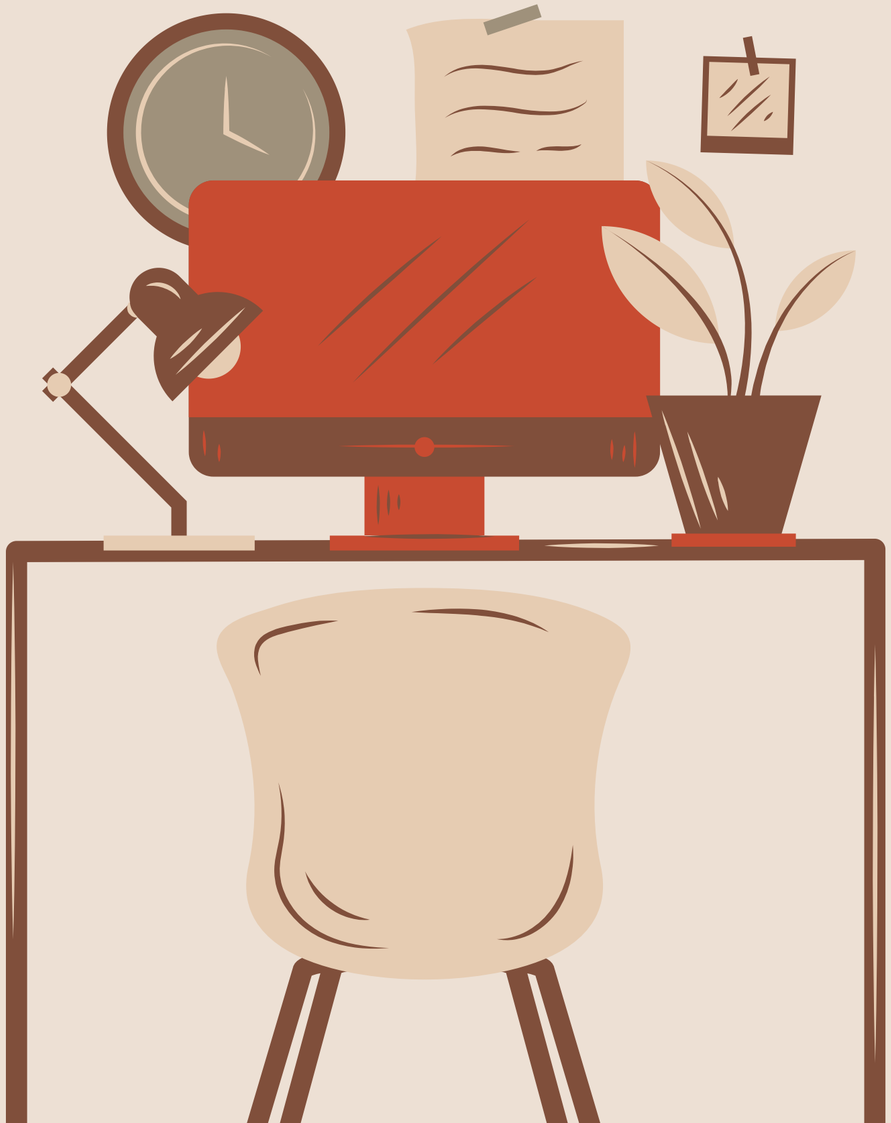
to specify how state tree
is updated on actions
you write pure reducers

- reducer is a function which takes state and action as argument and updates state
- 2nd principle says what to update
- 3rd principle says how to update

REDUCER code

```
const reducer = (state=initialState,action)=>{
        switch(action.type){
        case some_type:
        return updated_state
    }
}
```


subscribed
JS Application
dispatch
direct update
store
Action
reducer



action

the only ay js app can react with store. this carries some info from app to store in form of an object

- type property in action object wxplains what happened in app
- type property will tehnically be a string type

action creator

- action creator is a function which is used to return action object

reducers

action tell what to change in state

- reducer is a function which tells how to change state according to actions sent to store

action creator:

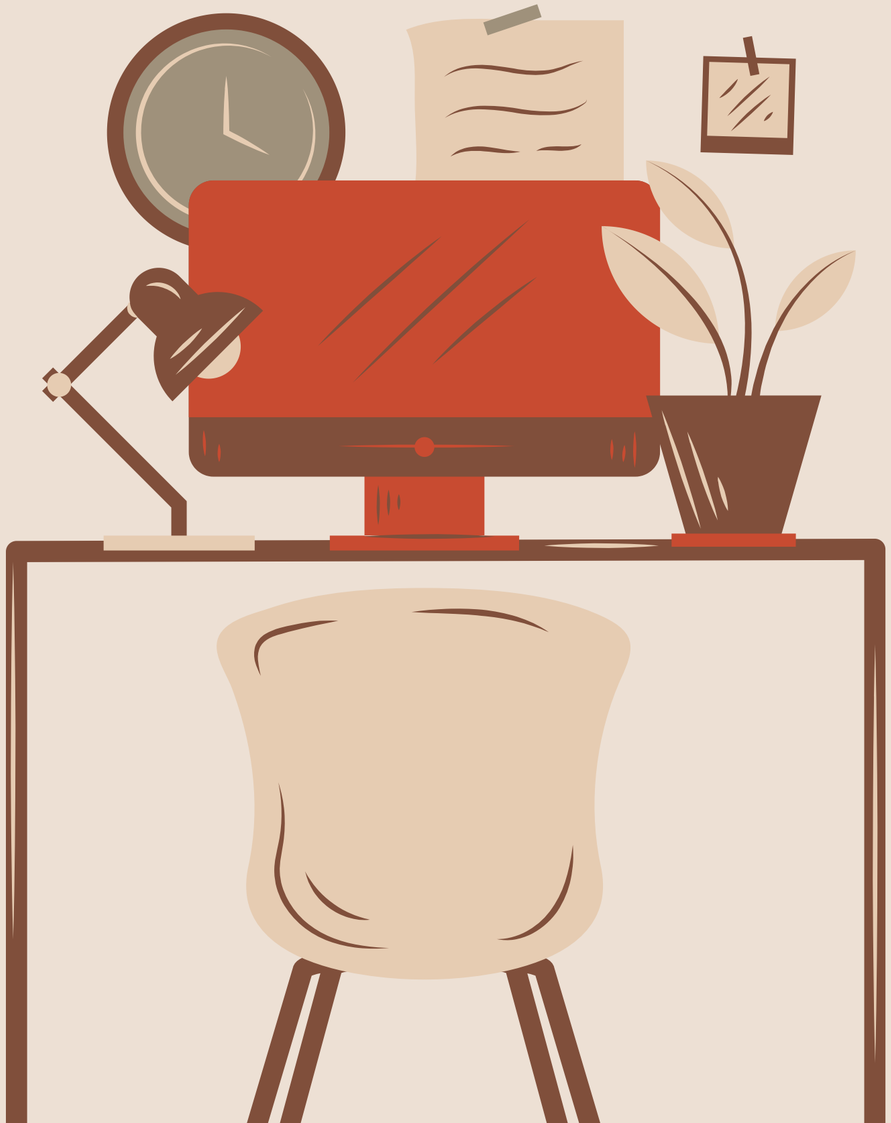
```
const buyCake = ()=>{
    return {
      type: 'BUY_CAKE',
      info: 'i want to buy the cake'
    }
}
```

state:

```
const initialState = {numOfCakes: 10}
```

reducer:

```
const reducer = (state=initialState,action){
        switch(action.type){
            return {
              ...state,
            [noOfCakes]: state.noOfCakes-1
            }
          default: return state
}
```



STORE RESPONSIBILITIES

- holds application state
- allows access to state via getState()
- allows state to be updated via dispatch(action)
- register listners through subscribe(listner)
- unregistering od listners via function returned by subscribe(listner)
  we just store returned fxn in other variable and call it at end of the code
-

responsibility 2:

to get state we use store.getState()

responsiboloty 4:

this responsibility of store is imp

it allows app to subscribe to the changes done in the store

achived using store.subscribe(listner fxn)

whenever the state updates then the subscribe listner fuction executes

store.subscribe(()=›console.log(“updated”))

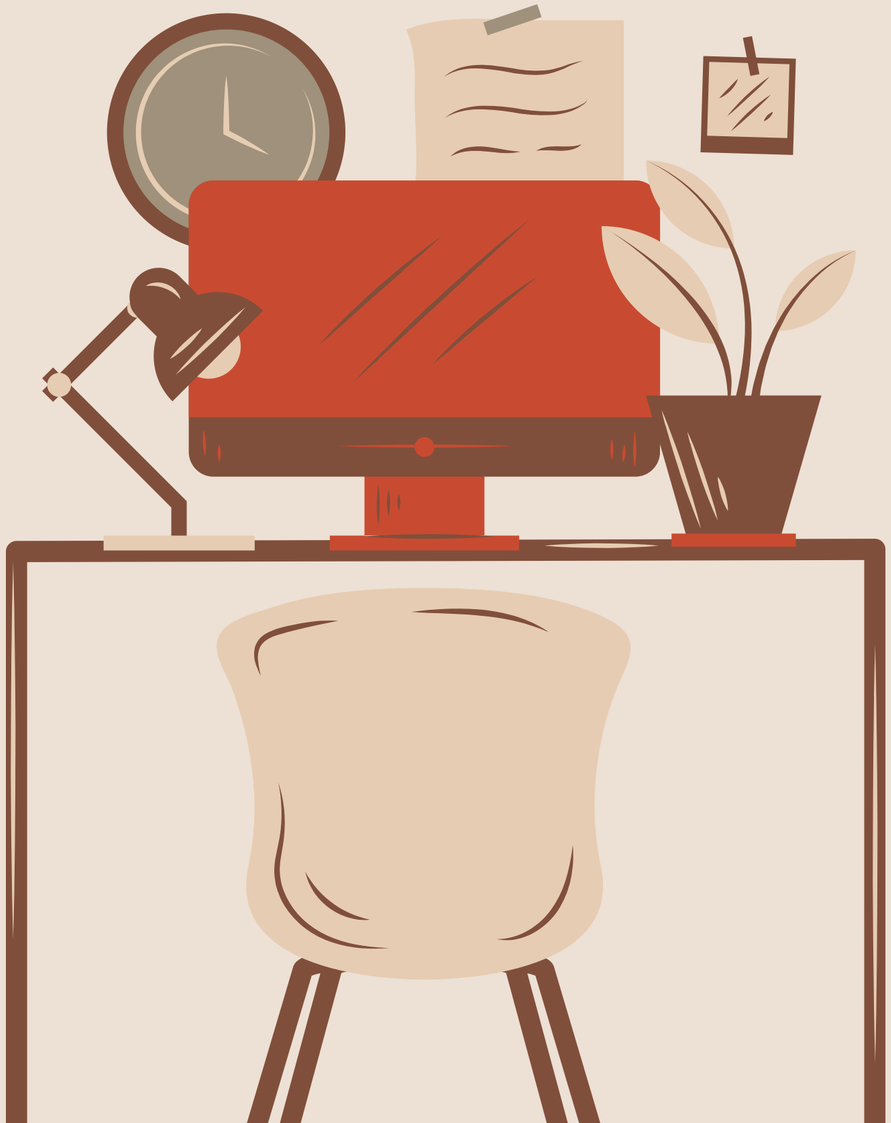
responsiboloty 3:

dispatch method is used to achive dispatch the action from app to redux store

eg: store.dispatch(action/actionCreator)

better to pass actionCreator

any change in creator can be helpful to change action easily



responsibility 5:

inorder to unsubscribe the listner we are iusing right now we use:

const someVariable = store.subscribe(listner fxn)

so just call someVariable() at the end of our code

```
const unsubscribe = store.subscribe(()=>{
        console.log("updated"))
})
```

at the end we unsubscribe and any dispatch after unsubscribe will not run the listerner function console

calling unsubscribe: unsubscribe()