ReactJS Global State Mangement Redux Toolkit

1. Global State Mangement, why!
2. How redux works, explanation!
3. How redux works, let's start coding!
4. Redux/ToolKit with React
5. Shopping Cart (E-Commerce System)
6. Redux bad practice

Kimz Code on YouTube

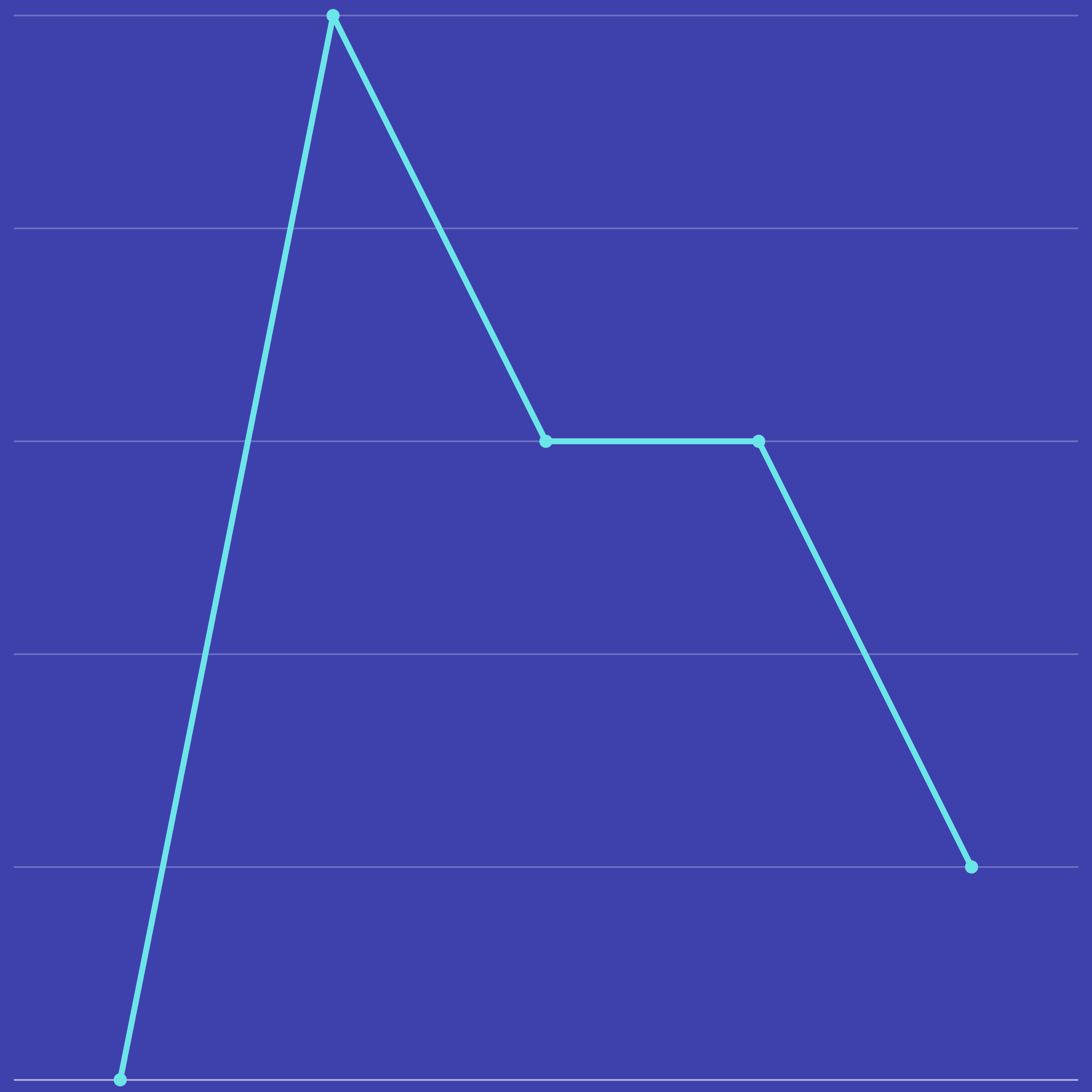
Learning Curve

Easy

Hard

Normal

Ready to start!



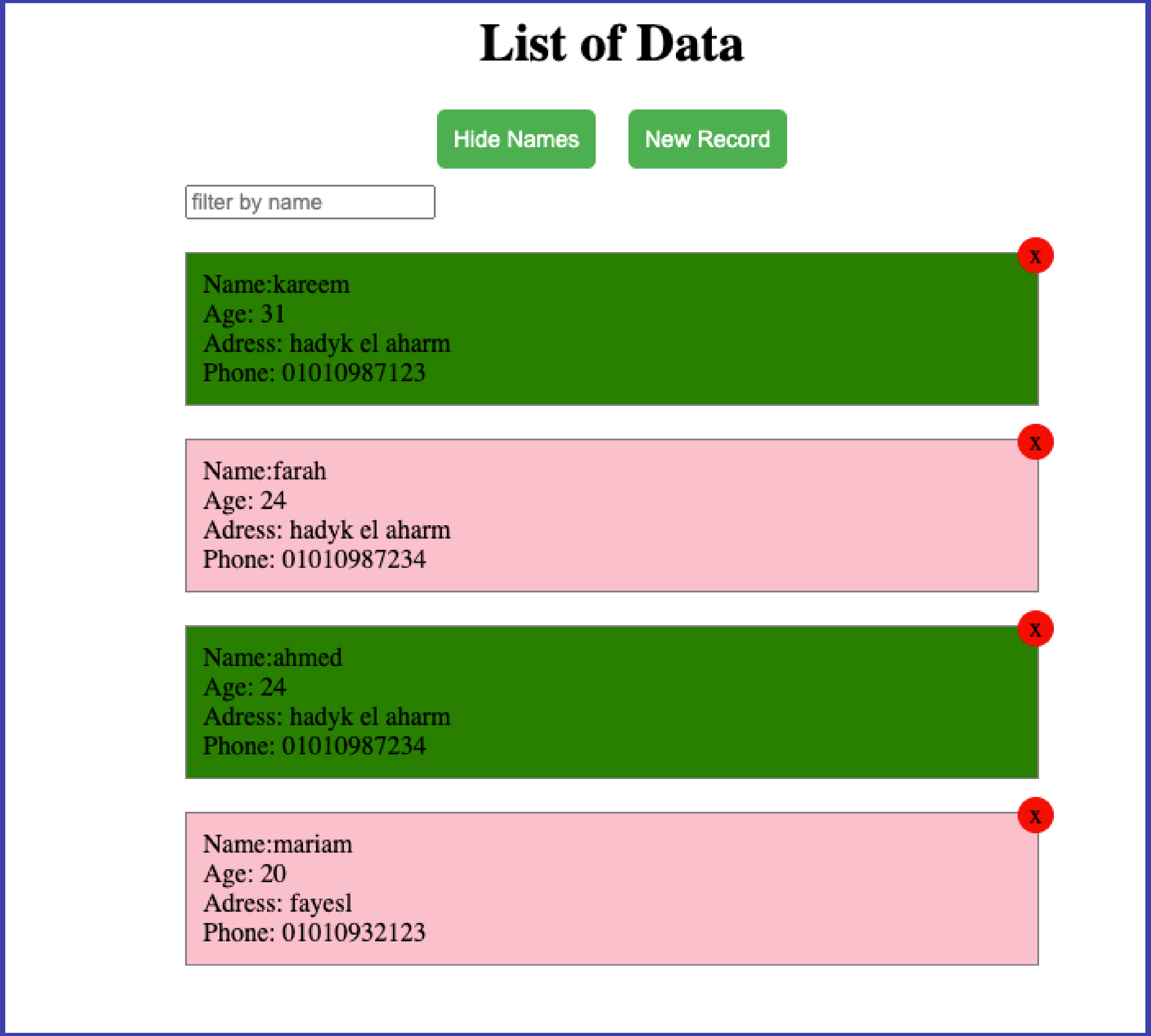
How to study and understand?
Always take notice

By Kimz Code YouTube Channel

Before global state

By Kimz Code YouTube Channel

Our Old App (local state AND Props Drilling)

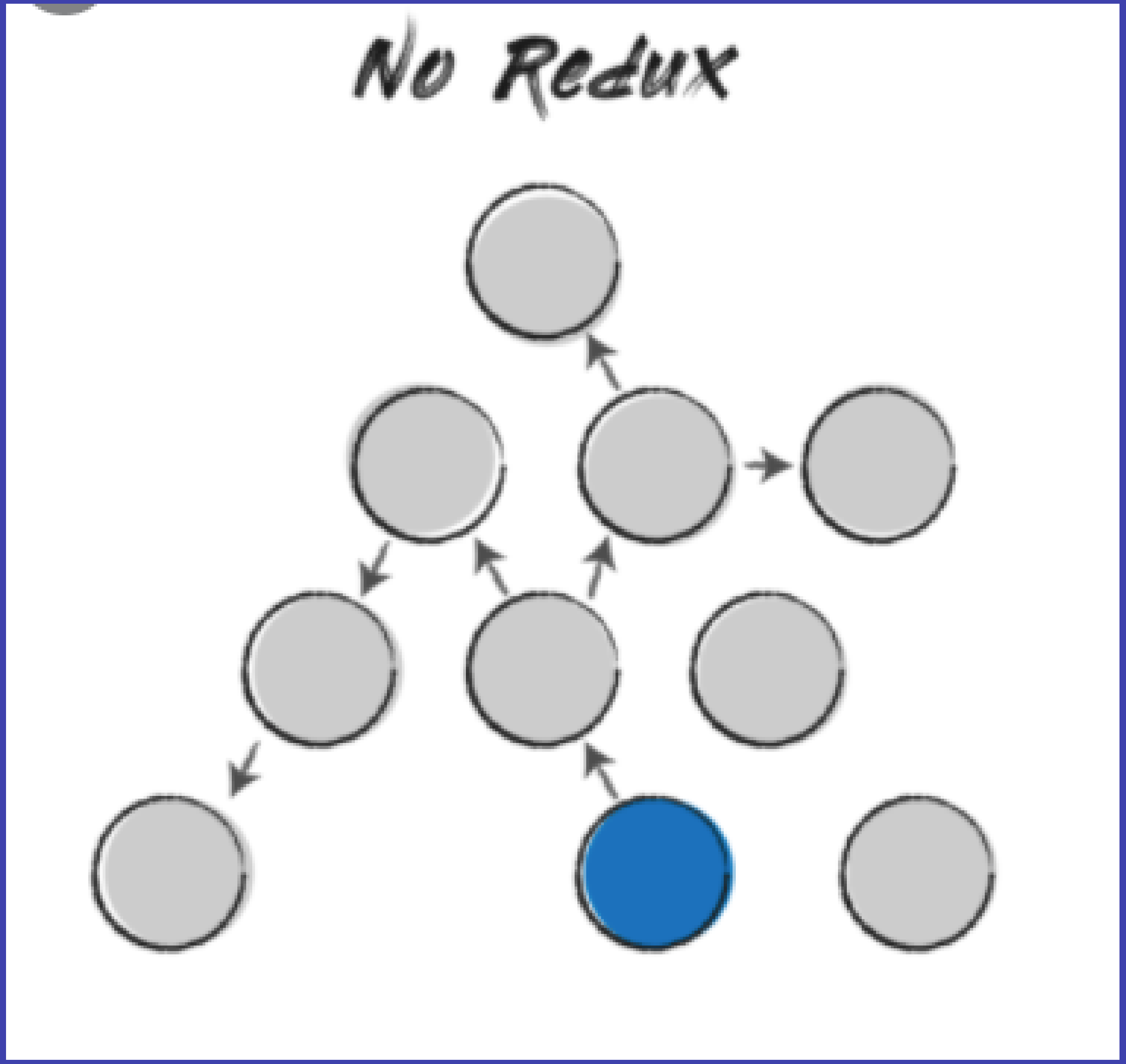


Name:kareem
Age: 31
Adress: hadyk el aharm
Phone: 01010987123

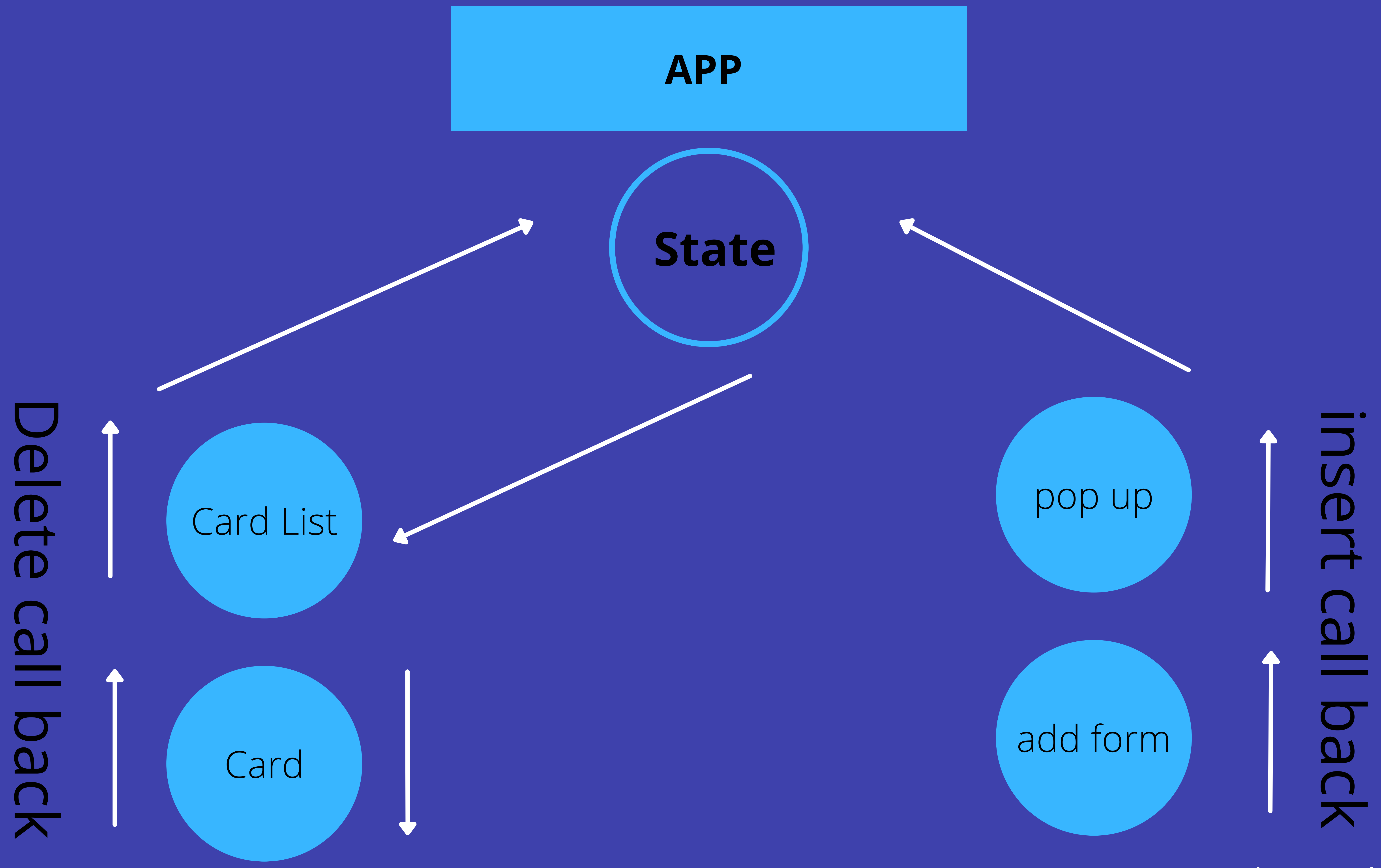
Name:farah
Age: 24
Adress: hadyk el aharm
Phone: 01010987234

Name:ahmed
Age: 24
Adress: hadyk el aharm
Phone: 01010987234

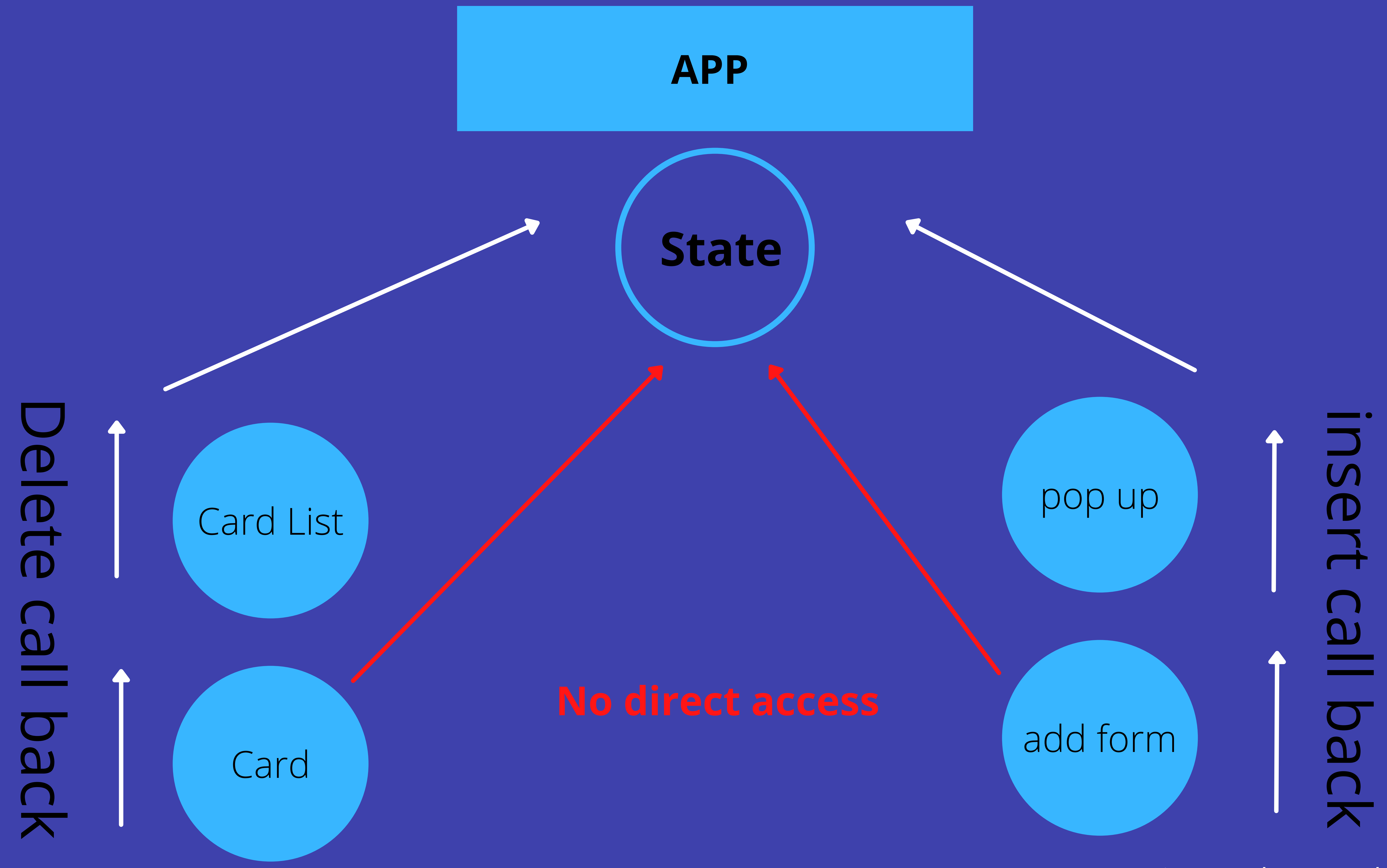
Name:mariam
Age: 20
Adress: fayesl
Phone: 01010932123



By Kimz Code YouTube Channel

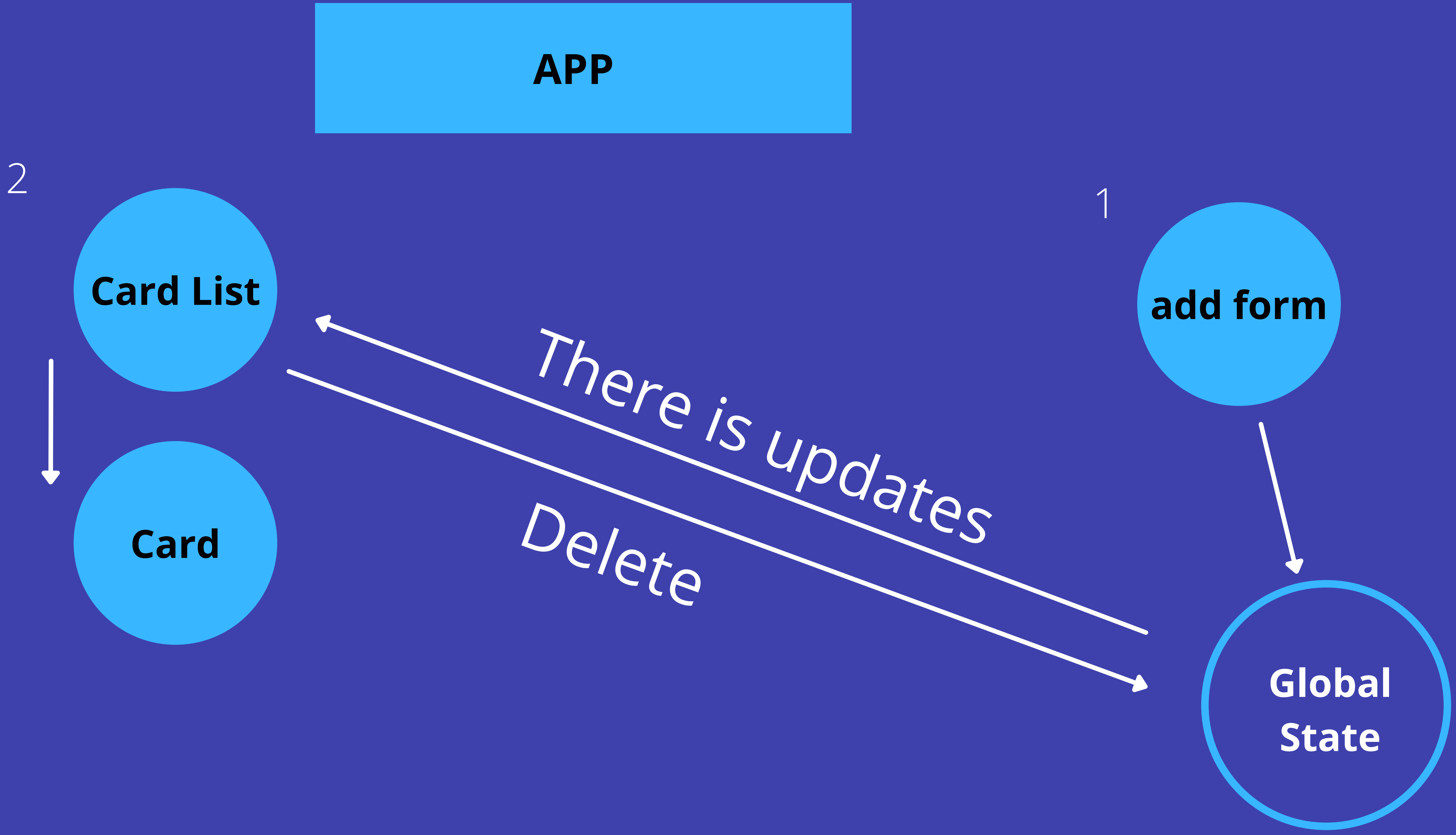


By Kimz Code YouTube Channel



By Kimz Code YouTube Channel

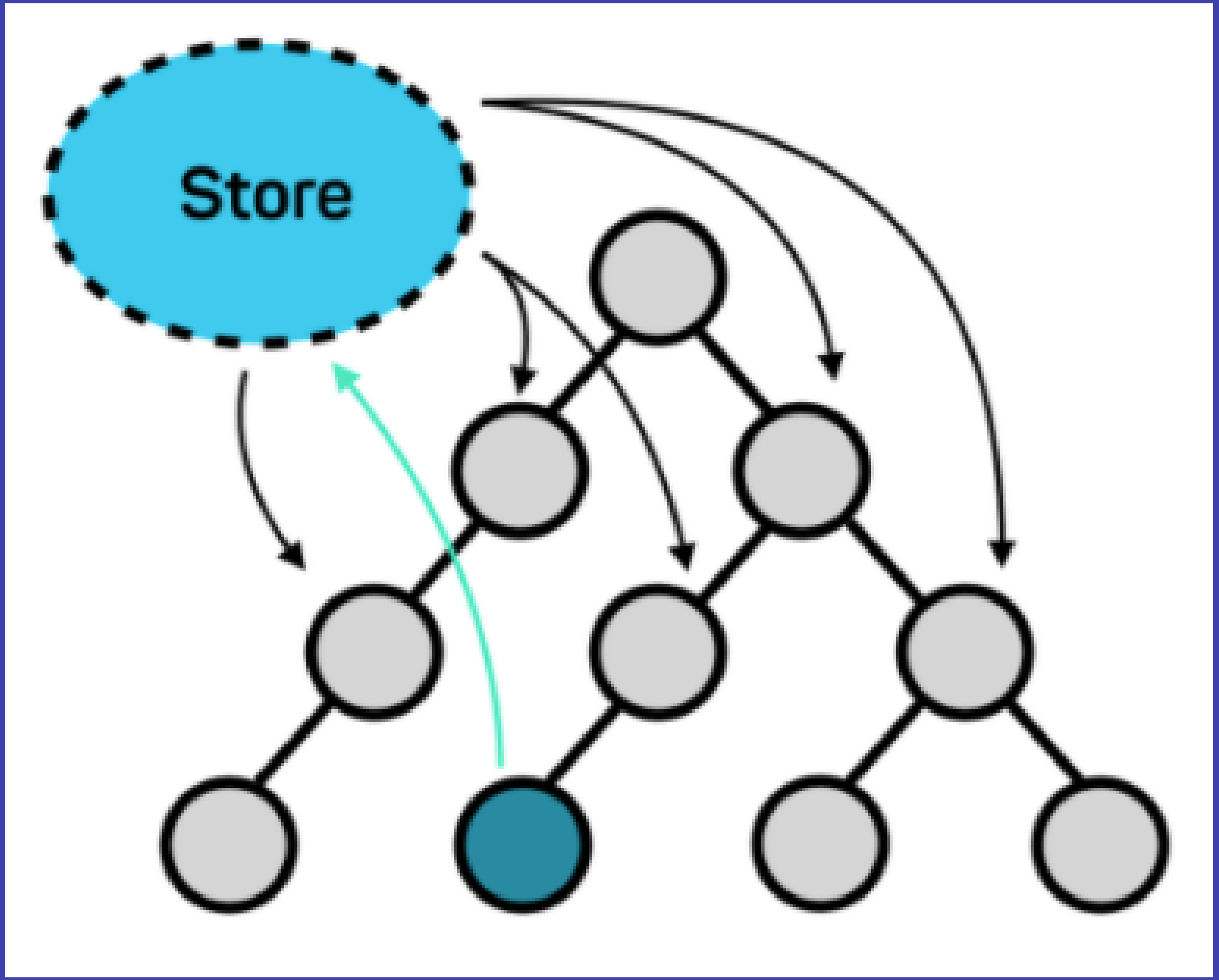
With global state



By Kimz Code YouTube Channel

With global state

Store (global state)



Conclusion

- Local state is scoped and we have drilling props (pass props & call back function) between components to share data
- Global state allows to access state from anywhere without do more drilling

Redux

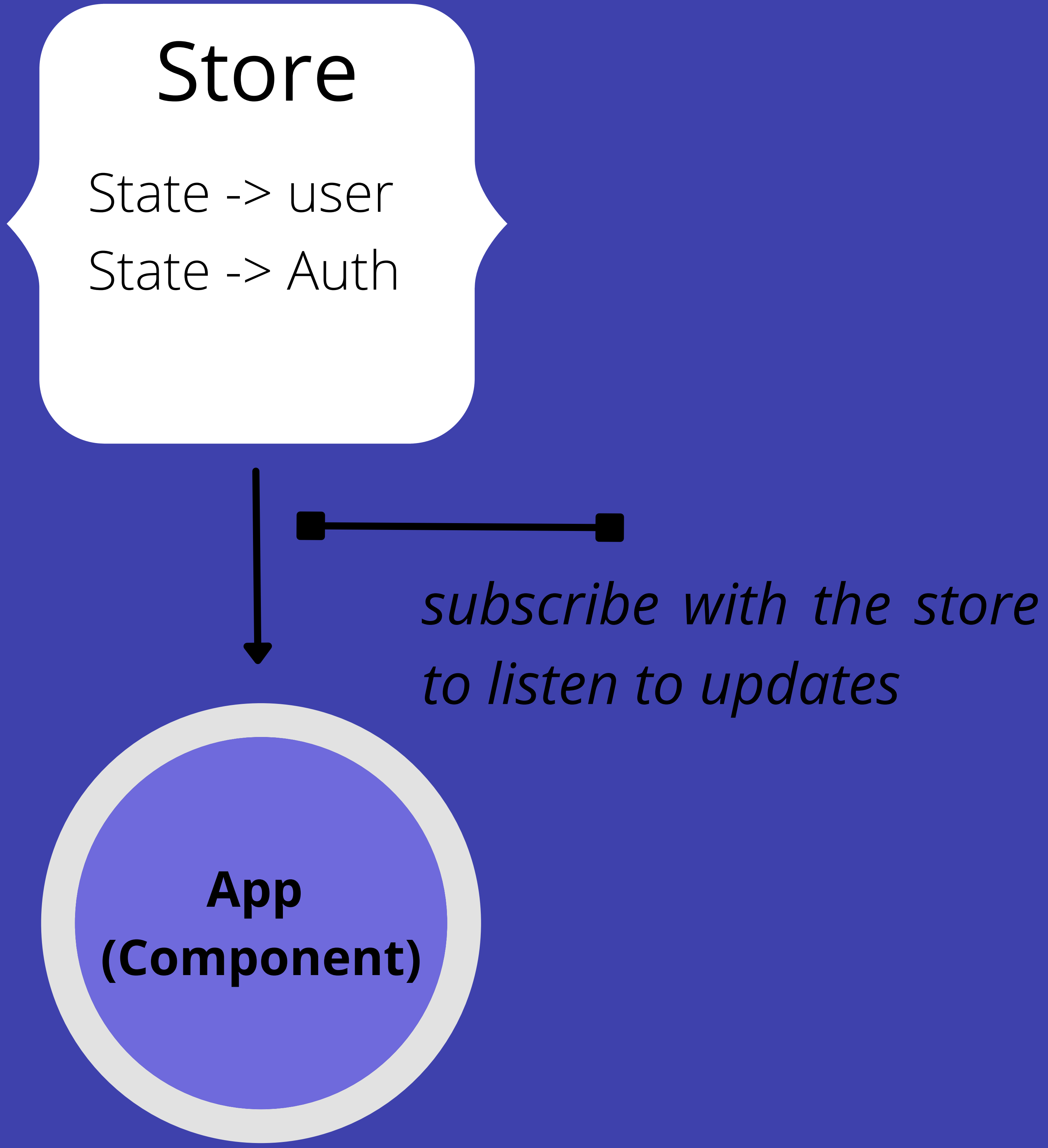
By Kimz Code YouTube Channel

Store

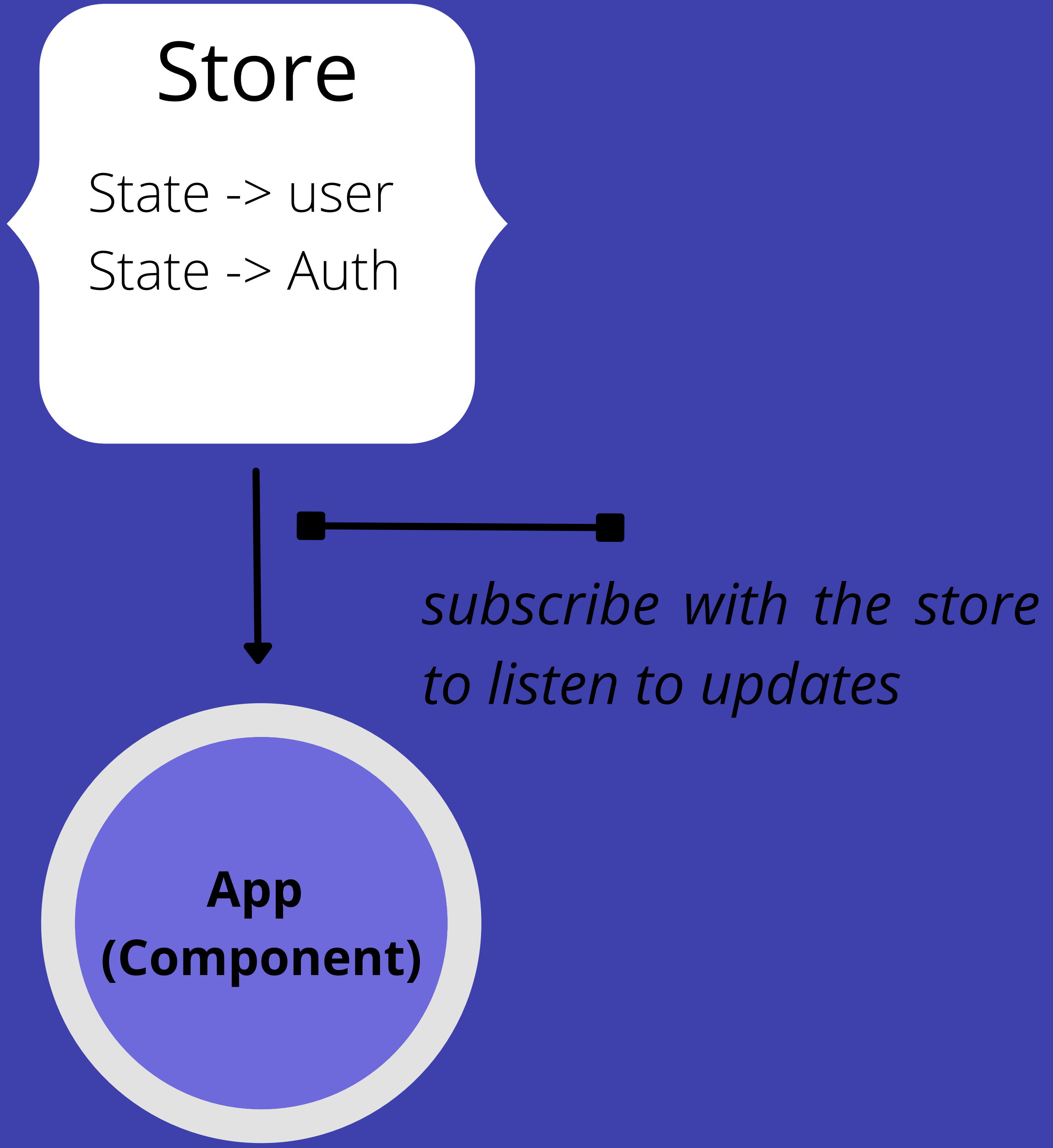
State -> user

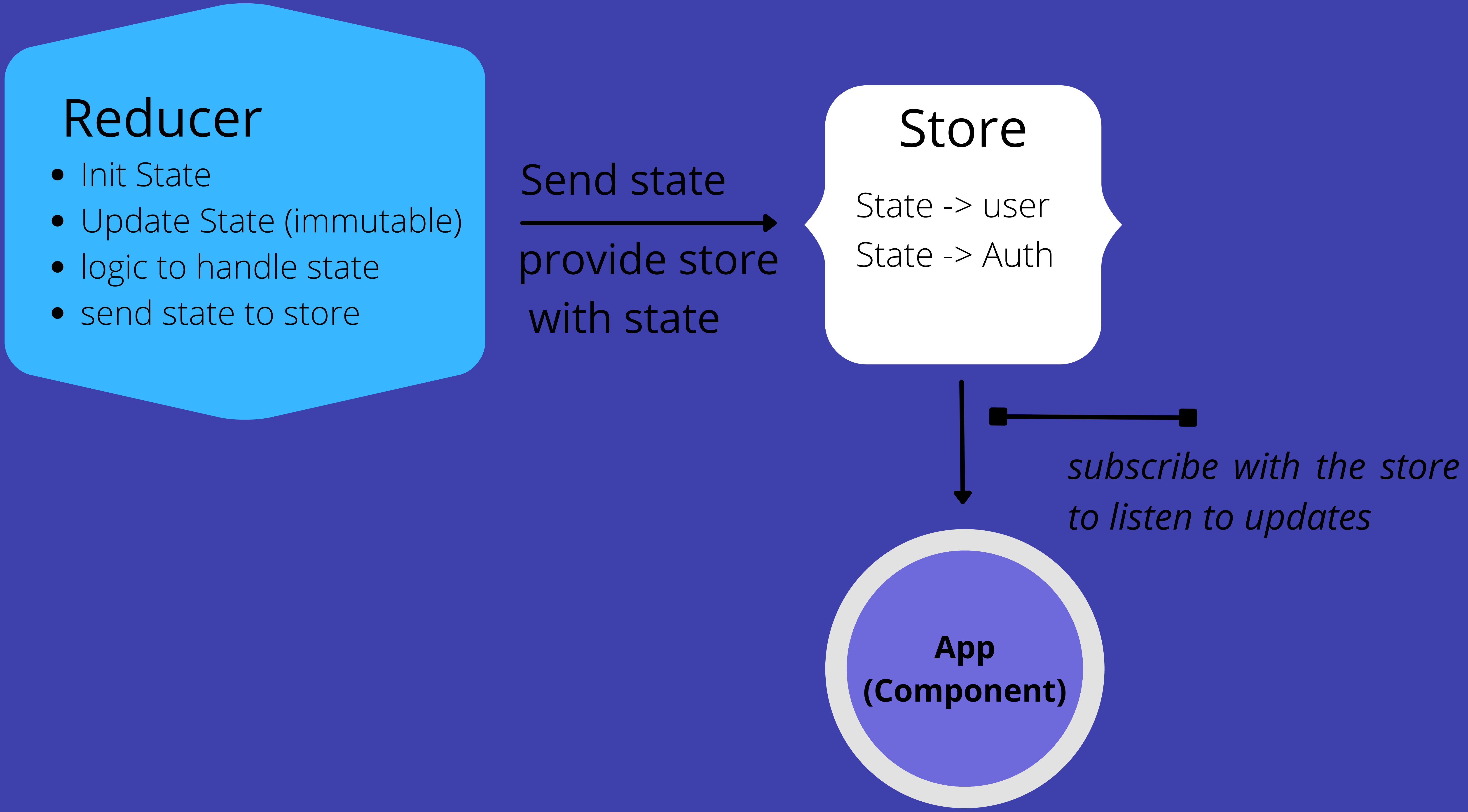
State -> Auth

How read state? Subscribe

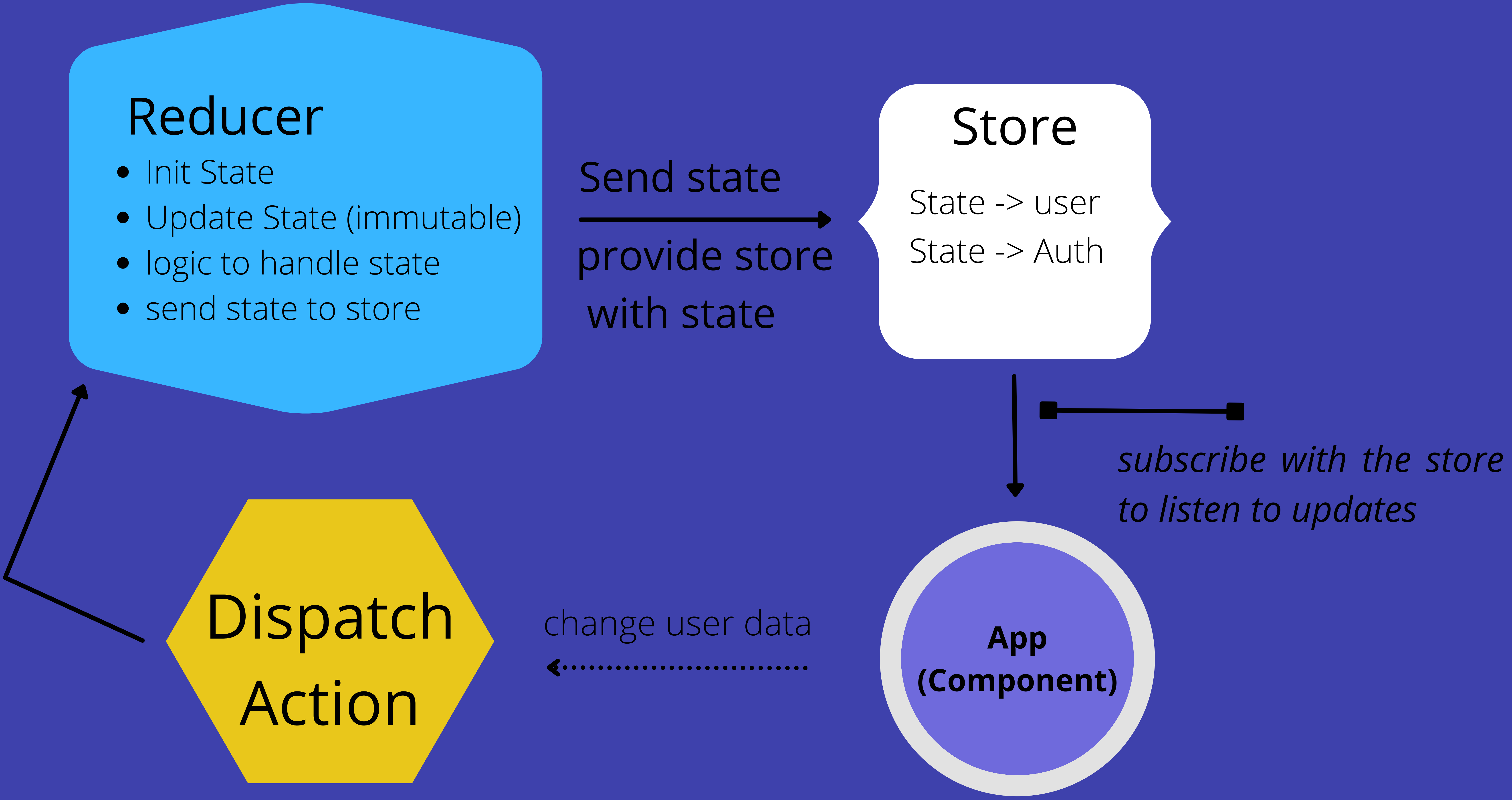


How is the state created in-store?
How the state will be updated?





By Kimz Code YouTube Channel



By Kimz Code YouTube Channel

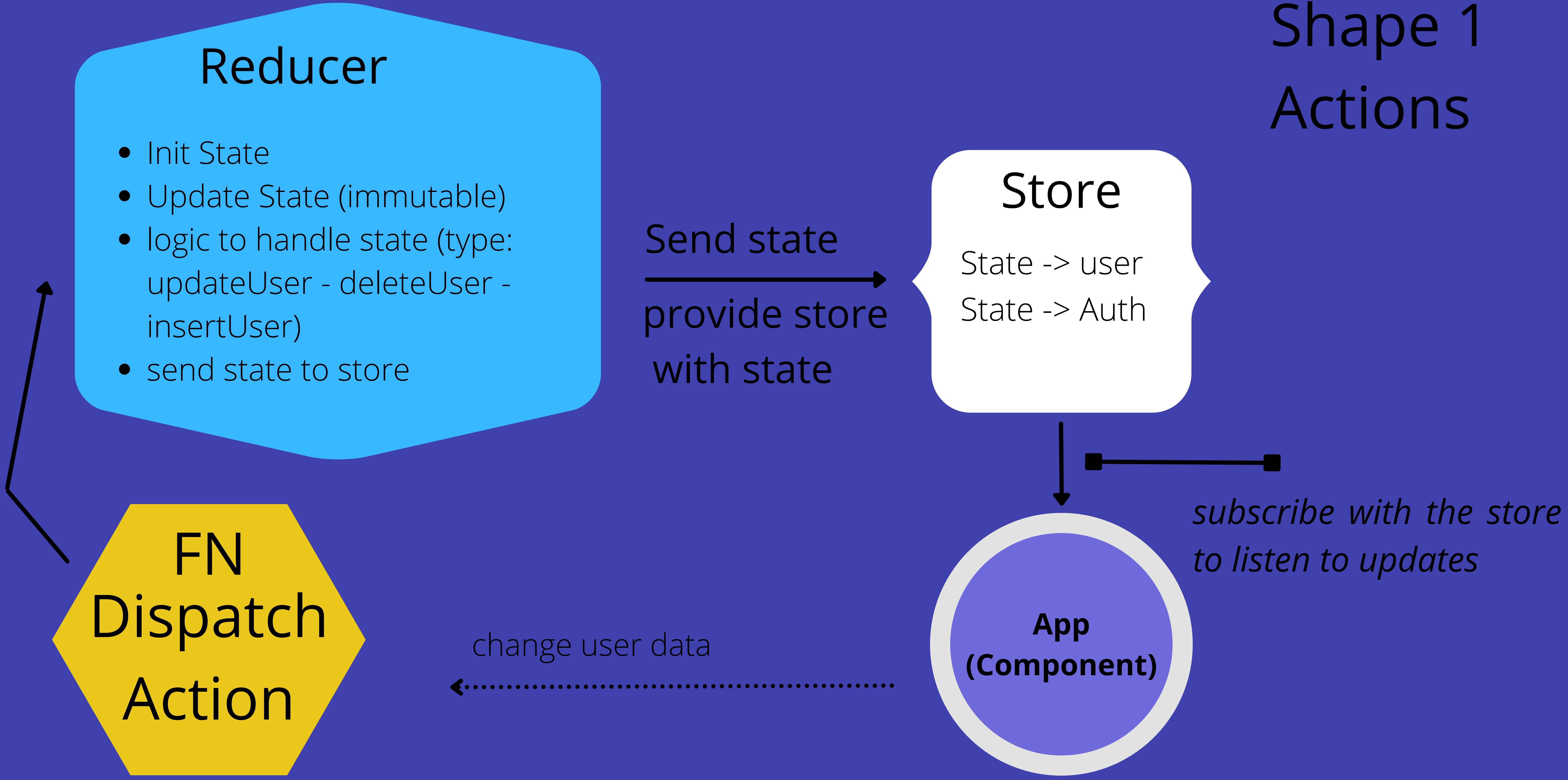
Conclusion

- Only reducers can create/update state
- The store is the place that contains all states
- (Dispatch function) send a trigger to force reducer do update
- Subscribe with the store to listen to updates
- Wihtout subscribe there is no re-render happen and component will not see updates

Lets go deep, step by step!

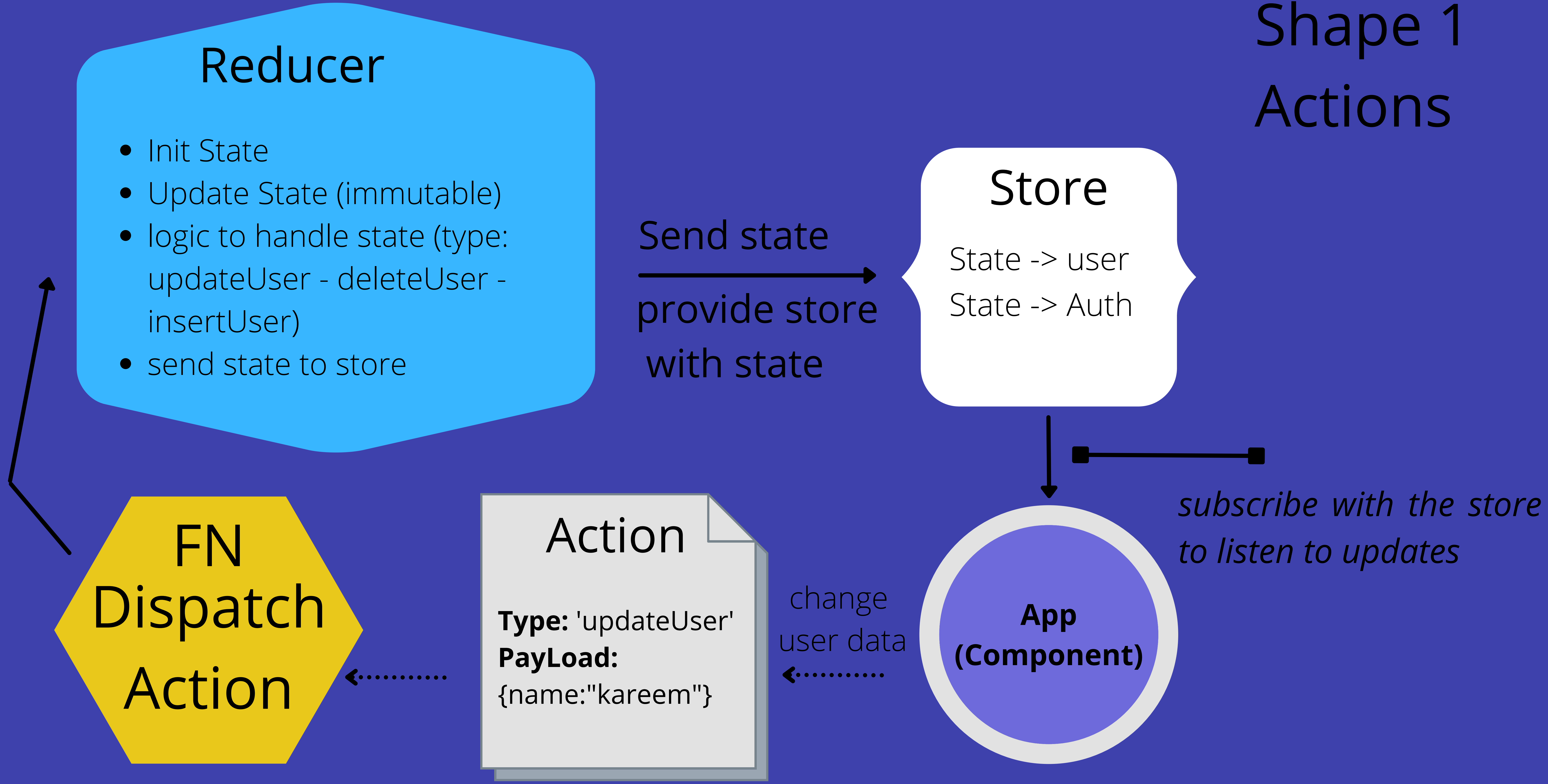
Action & Dispatch fn

Shape 1 Actions



By Kimz Code YouTube Channel

Shape 1 Actions



By Kimz Code YouTube Channel

Conclusion

- Action is something like a contract!
- Dispatch function: is a function that take Action and sends it to the reducer to make updates

Reducer

Reducer

```
const initState = {username: "kareem"}

const user = (state = initState, action) =>
{
  some logic to handle actions
  return state
}
```

- Is function
- Should be Pure
- Init the state and send it to the store
- State is object
- Don't mutate
- Always return
- Can have many reducers

Reducers

Reducer

```
const initState = {username: "kareem"}
const user = (state=initState, action) => {
    Return state

}
```

Reducer

```
const initState = {name: "dark"}
const theme = (state=initState, action)
=> {
    Return state

}
```

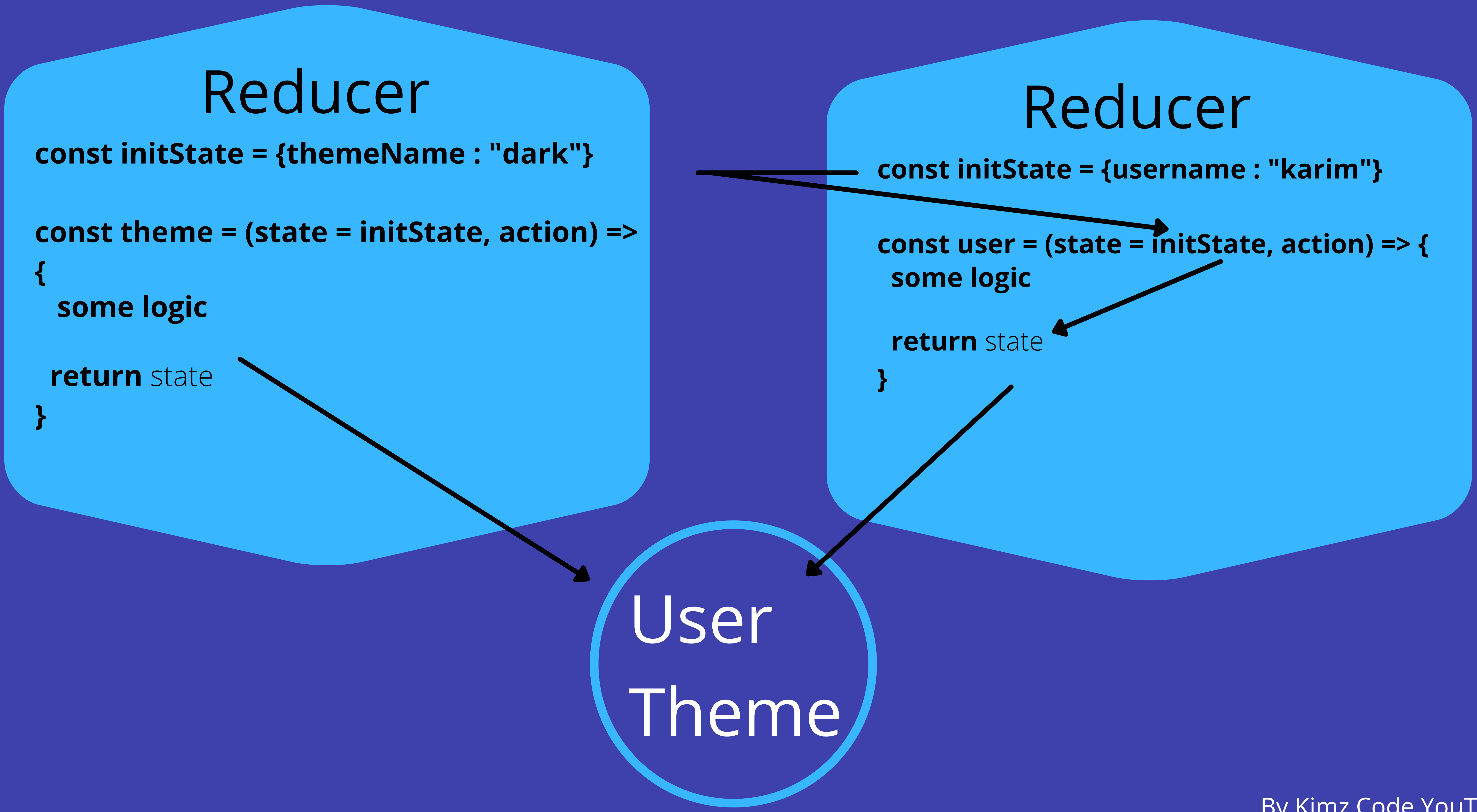
Reducer

```
const auth = {login: "false"}
const auth = (state=initState, action) => {
    Return state

}
```

Reducer

By Kimz Code YouTube Channel

Init state and send to store



```
const initState = {themeName : "dark"}

const theme = (state = initState, action) =>
{
  some logic

 return state
}
```

```
const initState = {username : "karim"}

const user = (state = initState, action) => {
  some logic

  return state
}
```

By Kimz Code YouTube Channel

Reducer (logic) - if Vs Action type

Reducer

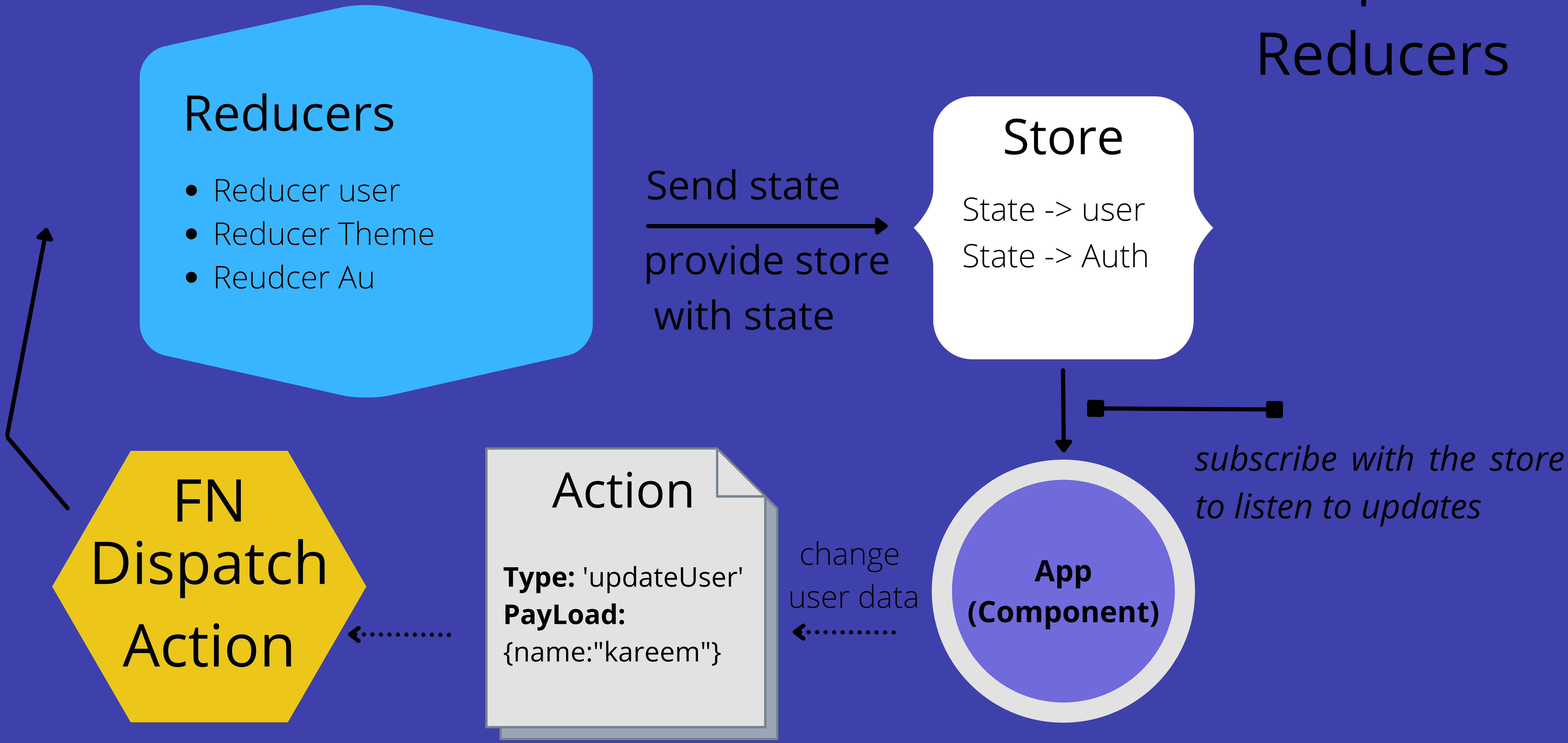
```
const initState = {themeName : "dark"}

const theme = (state = initState, action){
if(action.type === "changeTheme"){
  return state with new state + action.payload
}

  return state
}
```

Reducer

```
const initState = {username : "karim"}

const user = (state = initState, action){
if(action.type === "updateUser"){
  return state with new state + action.payload
}
if(action.type === "deleteUser"){
  return state with new state + action.payload
}
  return state
}
```

Dont worry, I will tell you how this logic will fire

Shape 2
Reducers



By Kimz Code YouTube Channel

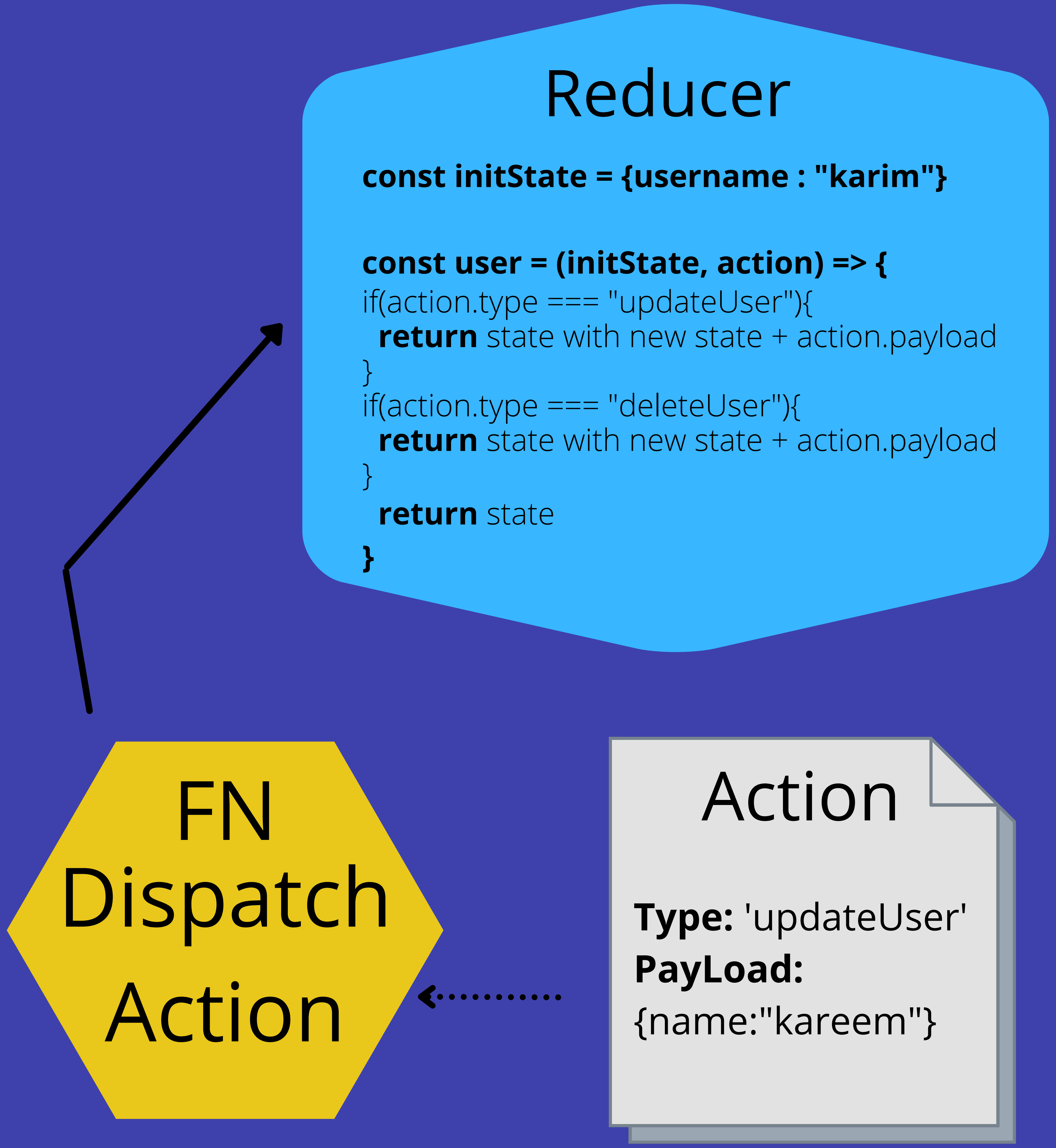
Conclusion

- Can have many reducers
- reducer return state
- returned state will take place in store

So how & when reducer function will fire?

Answer:

```
if(action.type === "updateUser"){
  return state with new state + action.payload
}
```



```
const initState = {username : "karim"}

const user = (initState, action) => {
if(action.type === "updateUser"){
  return state with new state + action.payload
}
if(action.type === "deleteUser"){
  return state with new state + action.payload
}
  return state
}
```

Type: 'updateUser'
PayLoad:
{name:"kareem"}

By Kimz Code YouTube Channel

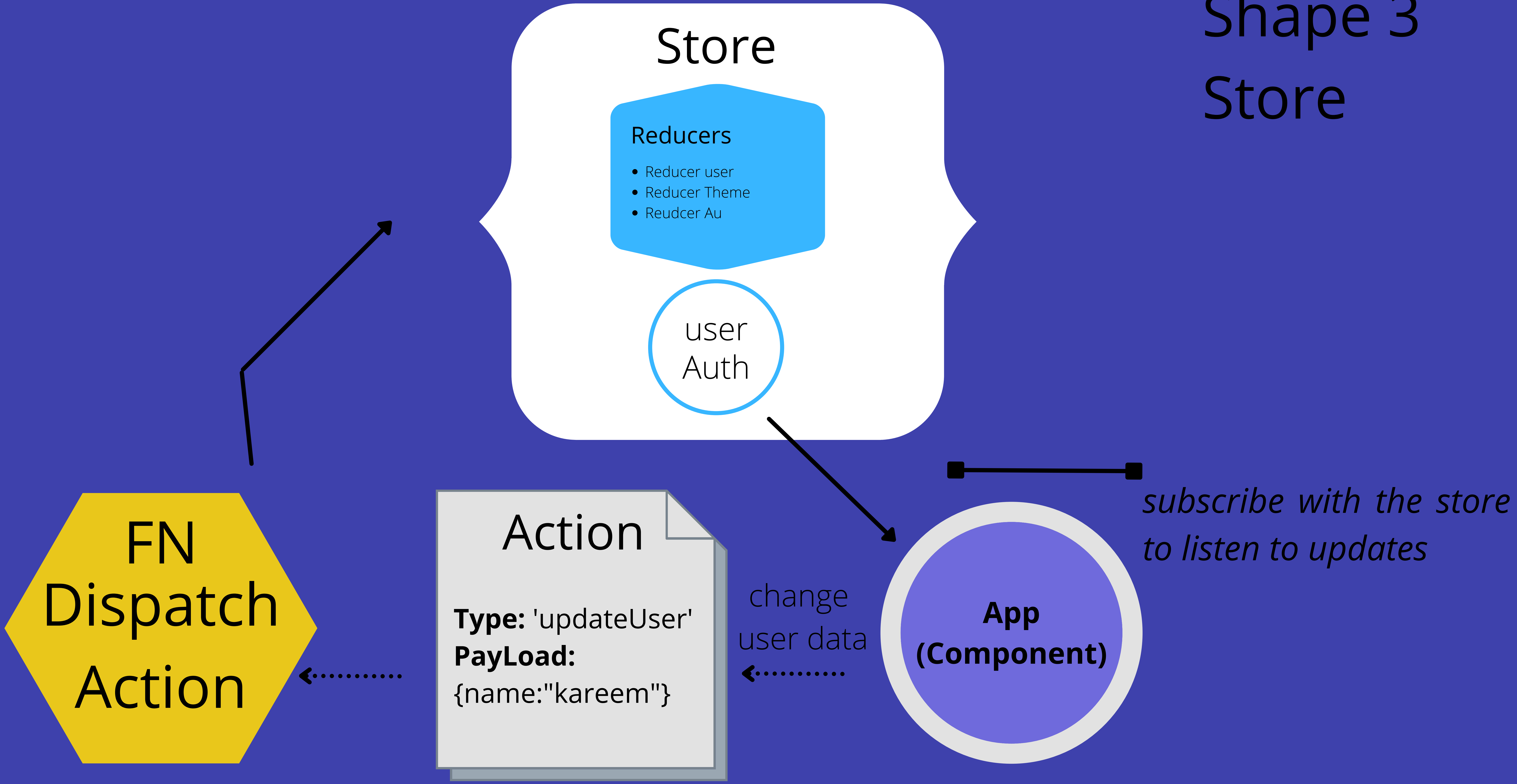
But what if i have many reducers, how i will naviagte action to related reducers?

to answer we need to:

- understand where is the reducer take place?
- need to look up again to the final design pattern!

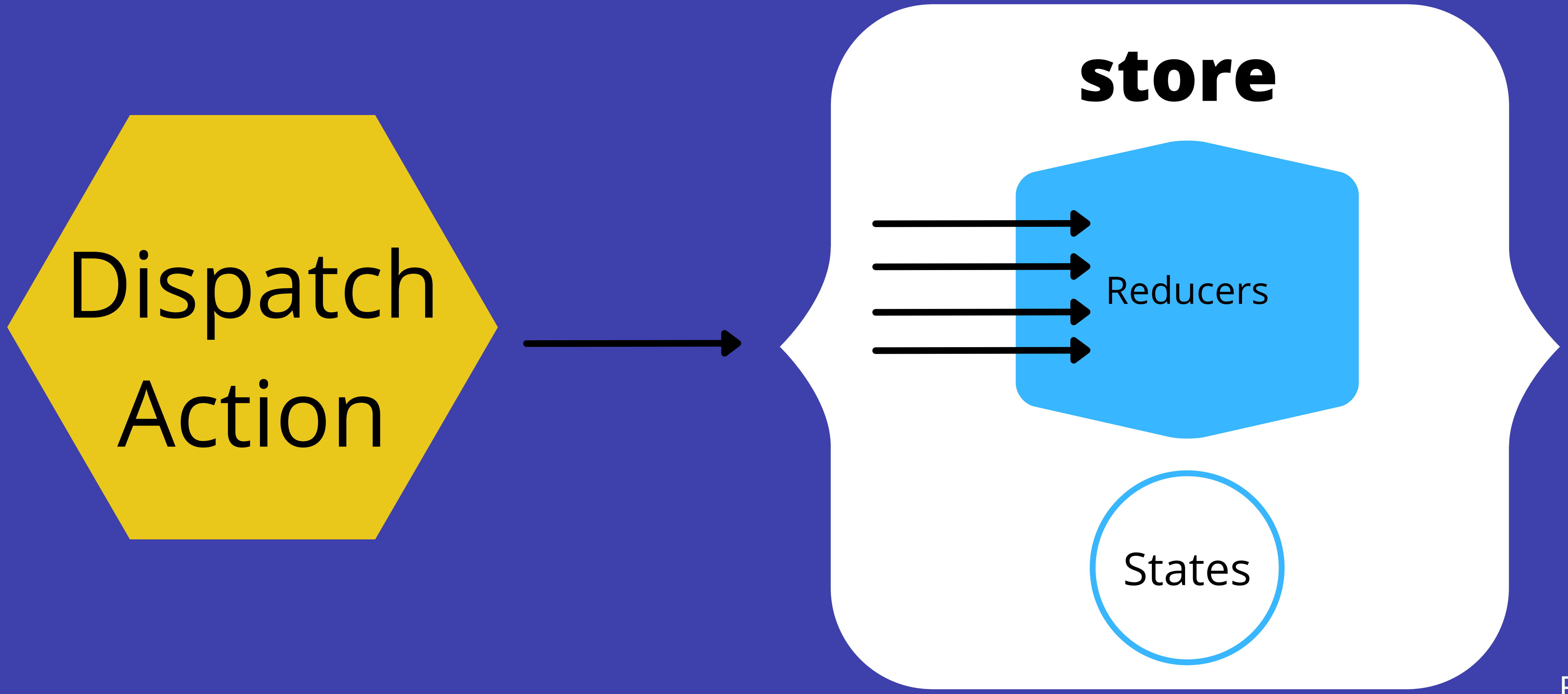
Need to understand first new shape & process on details

Shape 3 Store



Important note

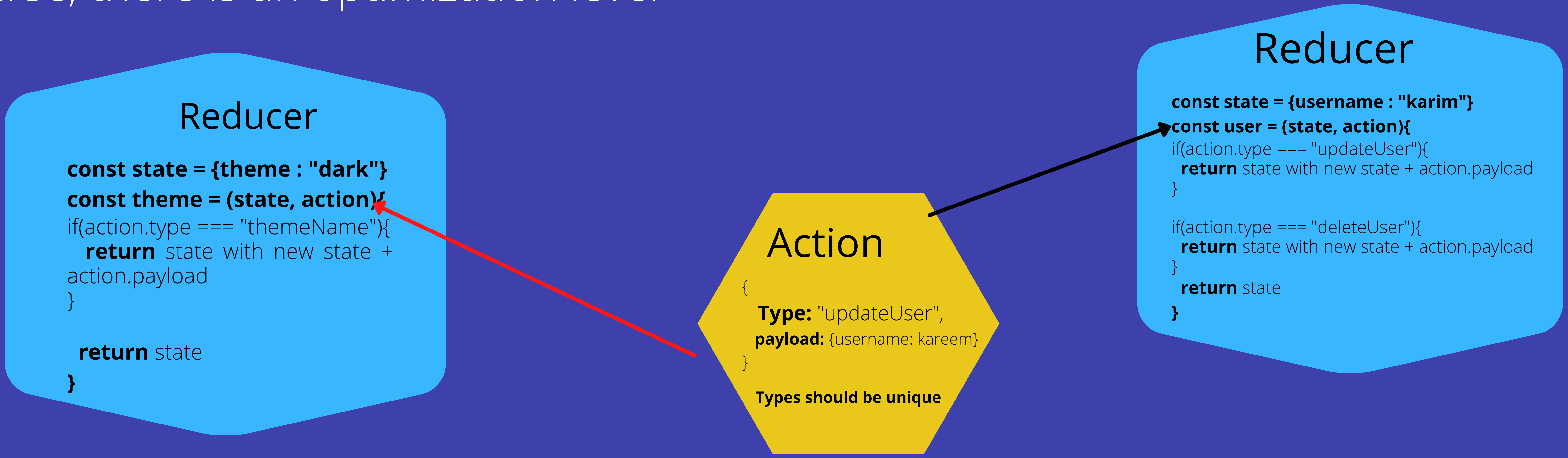
- Store contains reducer and states
- when dispatch action fire it goes to store first, then store send it to the reducer
- The store is the key,it takes action and lookup for related reducer



By Kimz Code YouTube Channel

Store & Dispatching Steps

- Store take incoming dispatch, fetch throw all reducers and assign action as a param for each reducer
- The reducer will be executed one by one - the reducer will recive action as param, and by using if condition with "action.type" reducer will check do I have this type? if yes stop looping and execute the logic
- of course, there is an optimization level



```
const state = {theme : "dark"}
const theme = (state, action){
if(action.type === "themeName"){
  return state with new state + action.payload
}

  return state
}
```

```
{
  Type: "updateUser",
  payload: {username: kareem}
}
```

```
const state = {username : "karim"}
const user = (state, action){
if(action.type === "updateUser"){
  return state with new state + action.payload
}

if(action.type === "deleteUser"){
  return state with new state + action.payload
}
  return state
}
```

By Kimz Code YouTube Channel

What about state as param? why

when dispatch is sent to store and while store does fetch throw reducers, store actually path two params to reducers:

- Action: we already know why.
- State: to make sure this reducer will have the latest version from the state

State Life cycle inside reducer:

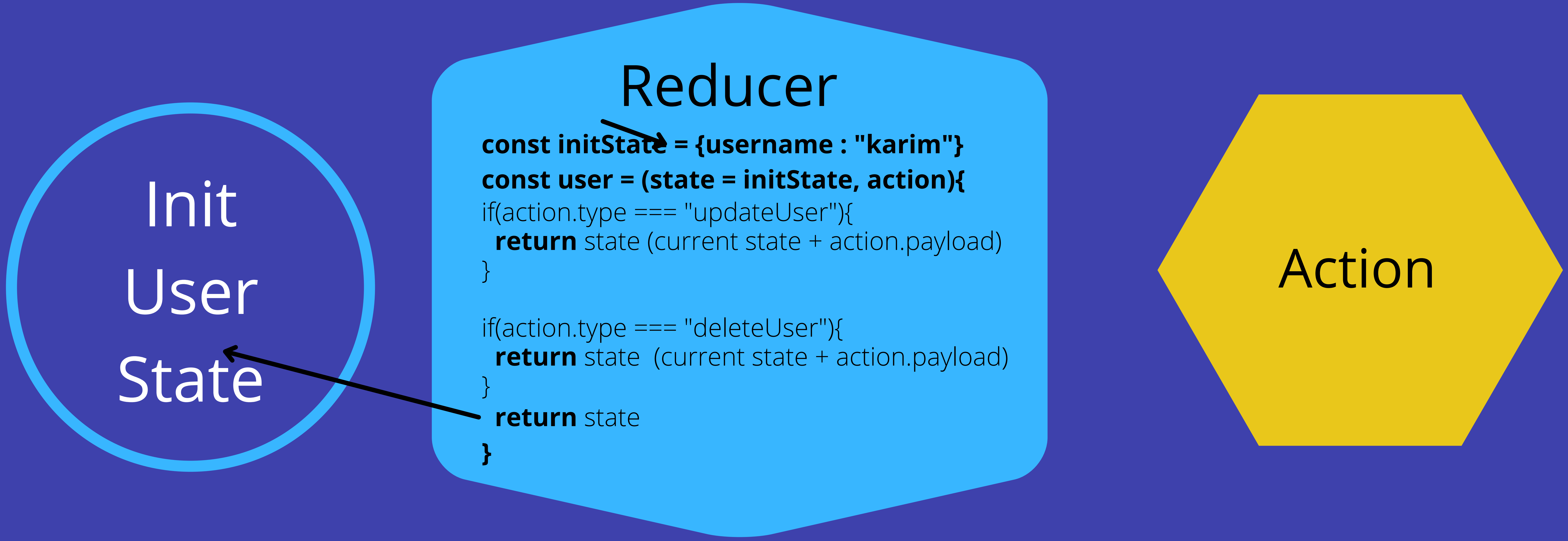
- when the app inits, the reducer will depend on the state you created **const state = {username: "karim"}** and will return it as default, why! because there no "action.type" match with what reducer have
- then, every time there is action come to the store, the store will pass the last state it has from the previous return

First time to load app

- **Store will start**
- **Action status:**

  Normal: no action

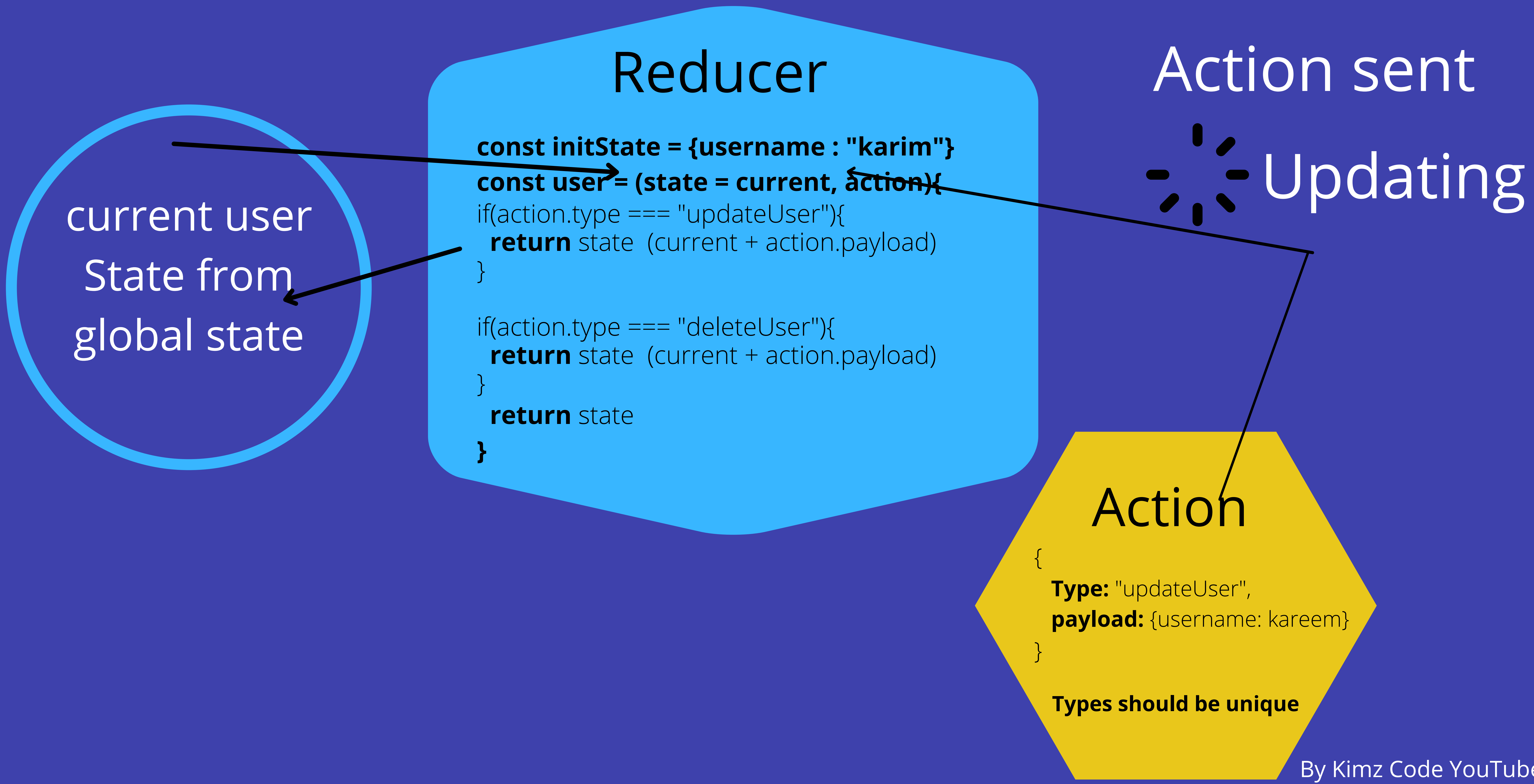
  unless you sent action when app init for the first time



```
const initState = {username : "karim"}
const user = (state = initState, action){
if(action.type === "updateUser"){
  return state (current state + action.payload)
}

if(action.type === "deleteUser"){
  return state  (current state + action.payload)
}
  return state
}
```

By Kimz Code YouTube Channel

After first time to load app



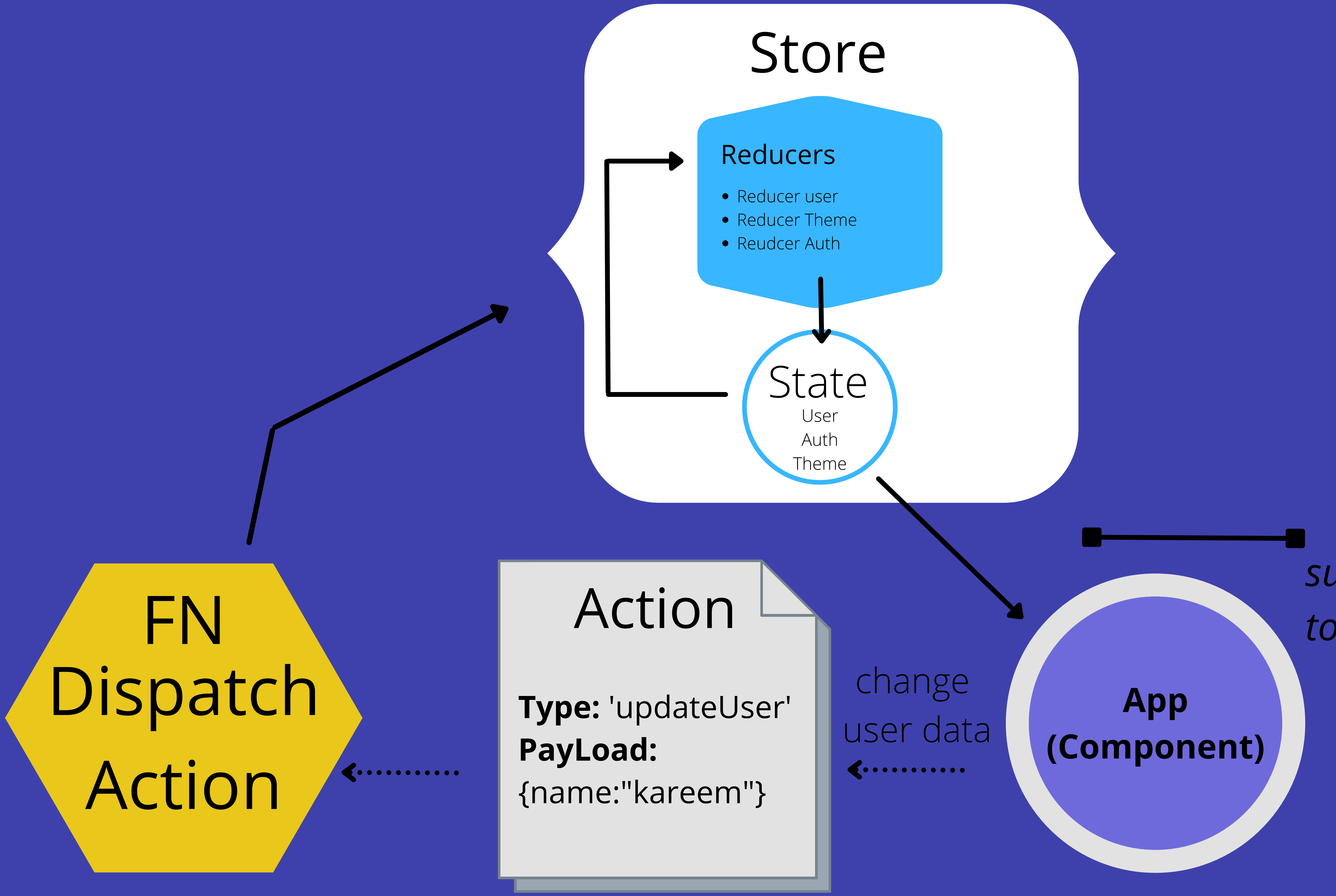
```
const initState = {username : "karim"}
const user = (state = current, action){
if(action.type === "updateUser"){
  return state  (current + action.payload)
}

if(action.type === "deleteUser"){
  return state  (current + action.payload)
}
  return state
}
```

```
{
  Type: "updateUser",
  payload: {username: kareem}
}
```

Types should be unique

By Kimz Code YouTube Channel

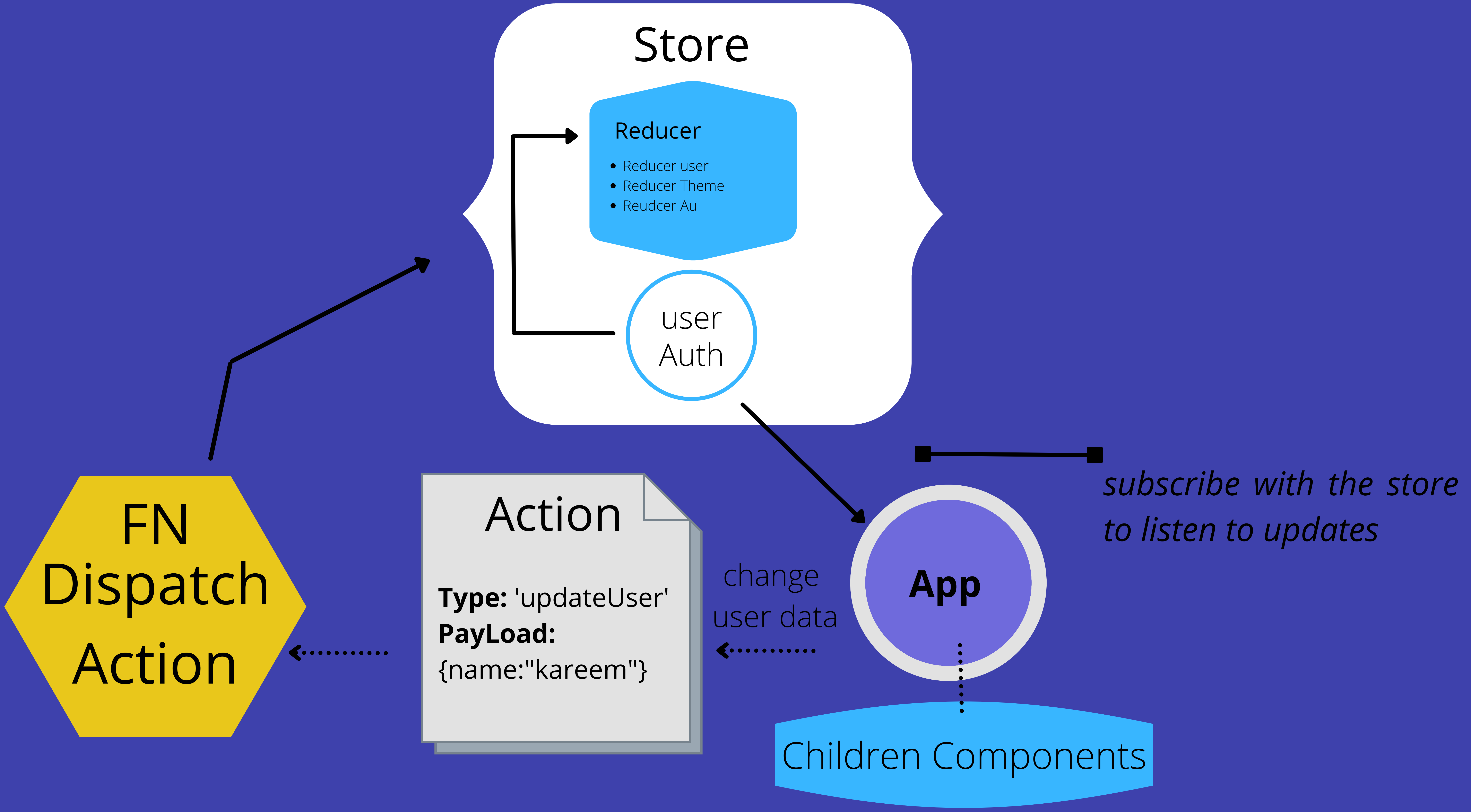
Shape 4
Final Store



subscribe with the store to listen to updates

React & Redux Subscription

- Prefer to subscribe with the main App or main parent for groups of component
- Once subscribn enabled all children will be able to see updates too



By Kimz Code YouTube Channel

The End of story

By Kimz Code YouTube Channel