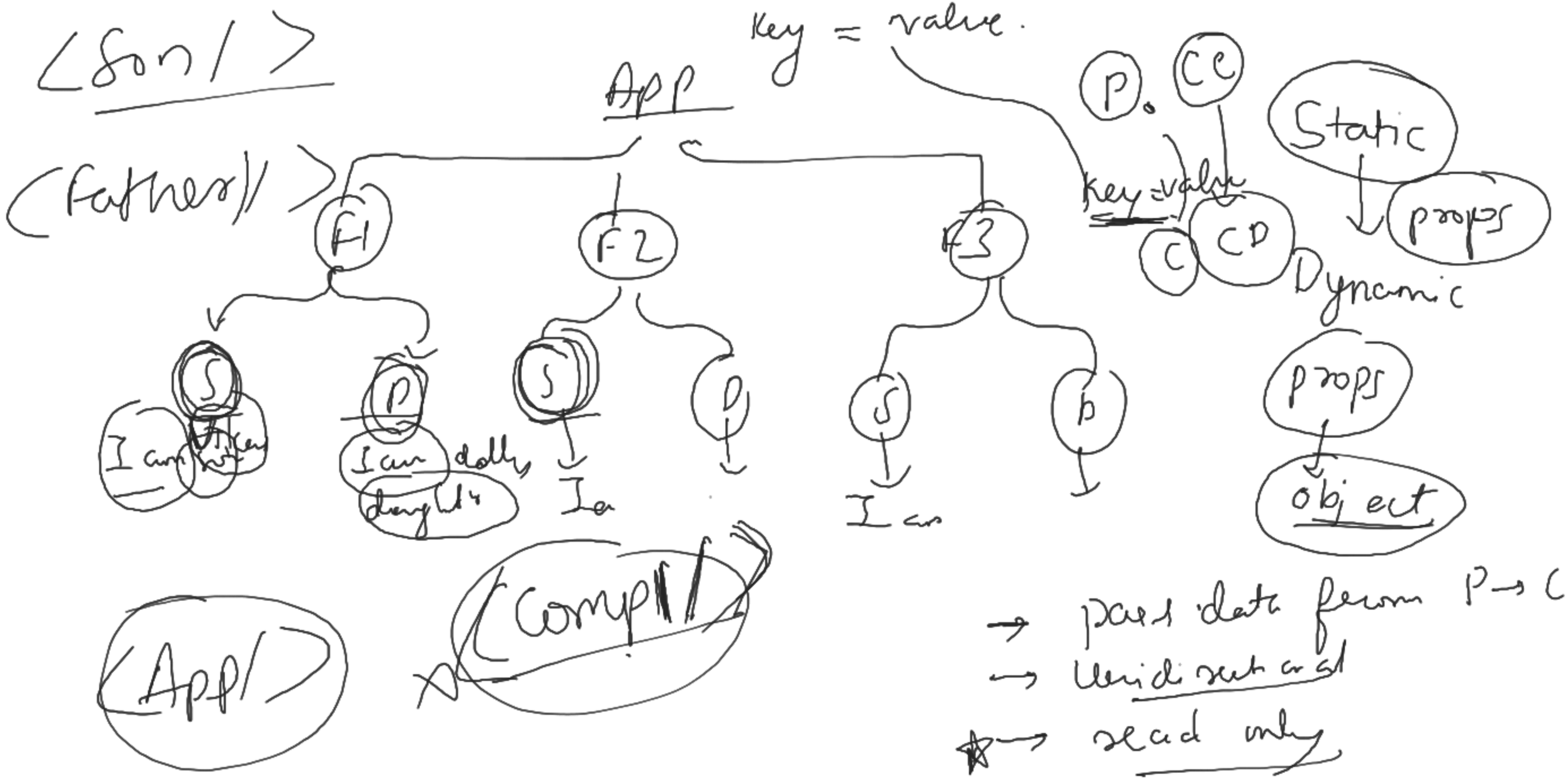

Son
App
Key = value.
Father
F1
F2
F3
S
P
I am
I am daddy
daughter
I a
I am
b
P.
CC
Key = value
C
CD
Static
Dynamic
props
object
Comp
App
→ pass data from P → C
↔ Unidirectional
★ → read only

STATE Management

State matters

↓

forms

What is State ?

Ans → No change

→ something fixed

→ Something in order

→ Conditional.

State

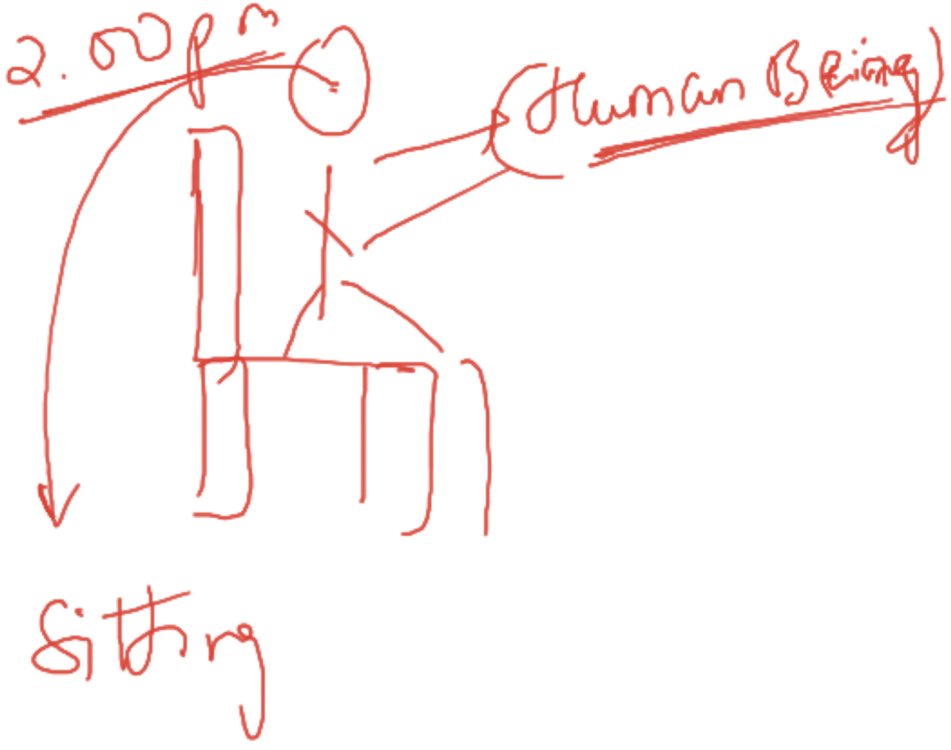
↓

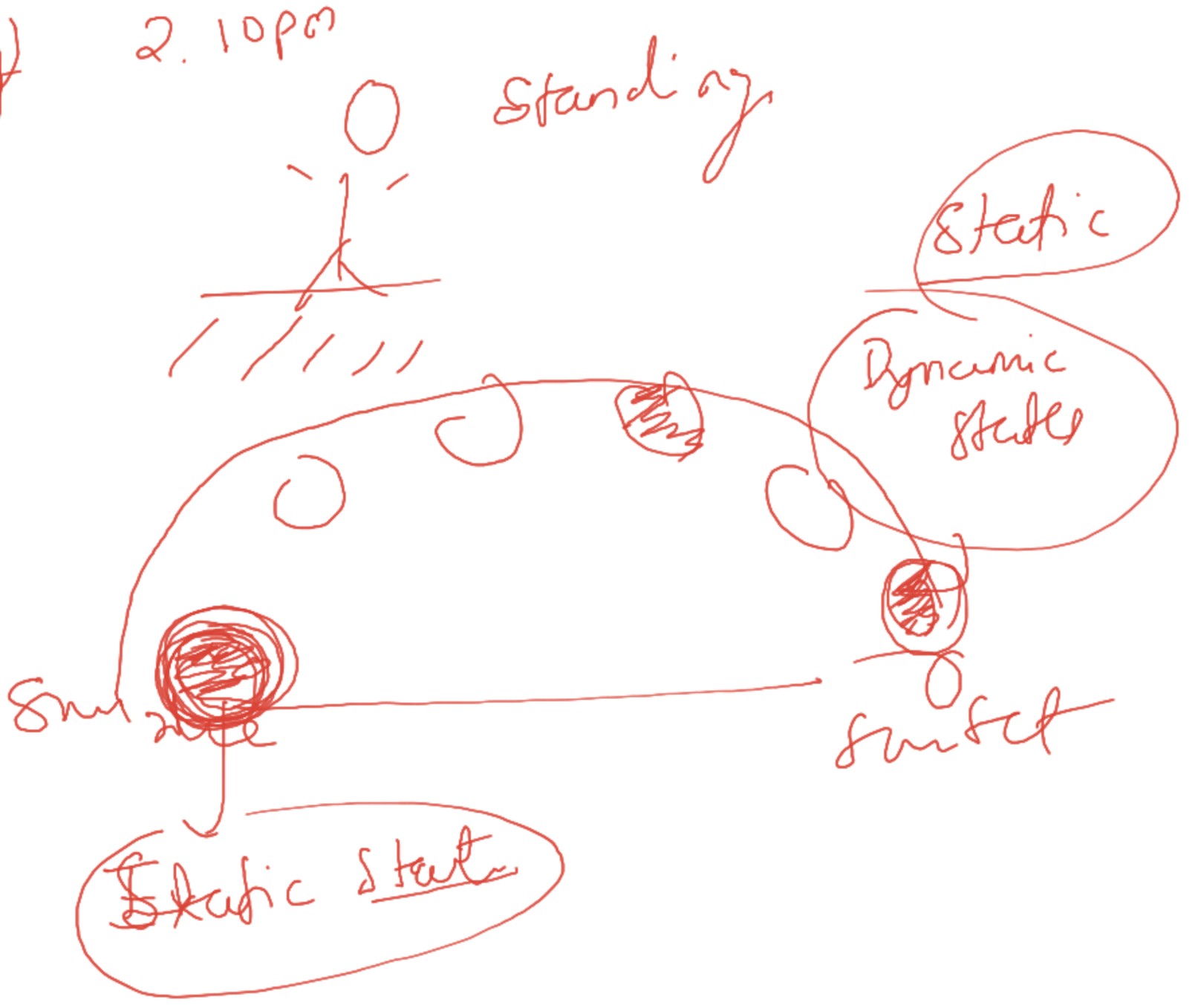
Border (only in React)

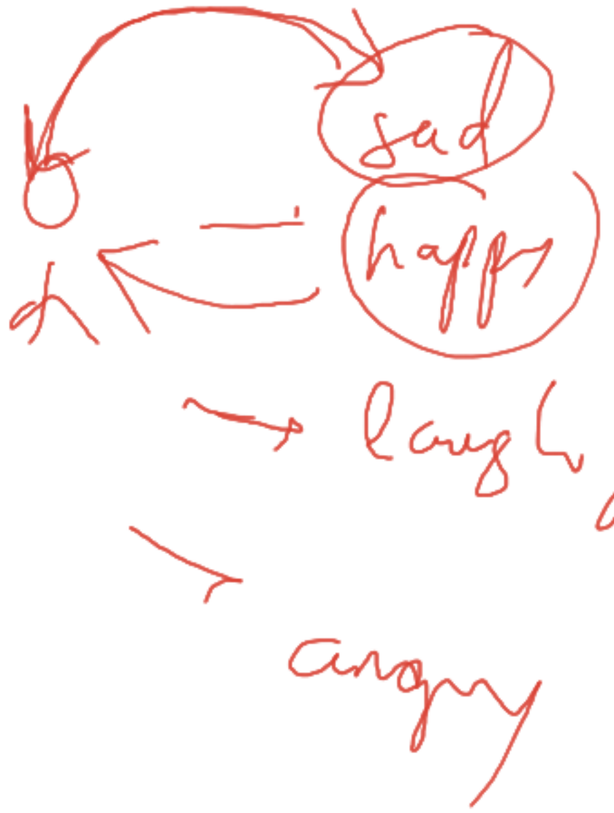
State

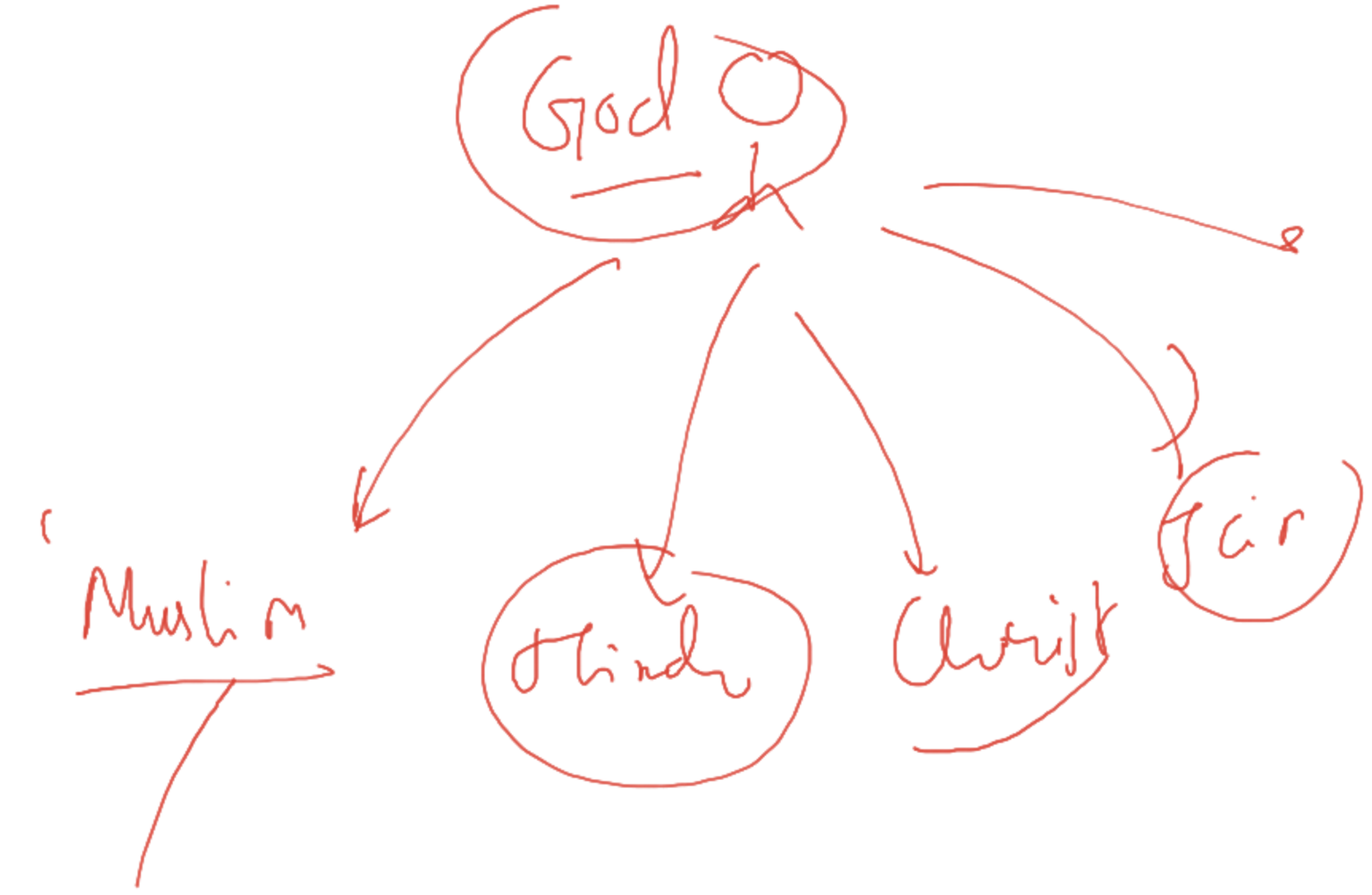
↓

anything that exists right now in this moment.


2.00pm
Human Being
Sitting


2.10pm
Standing
Static
Dynamic
Static state


sad
happy
laughing
angry


God
Muslim
Hindu
Christ
Sir

Software Development

App → also has State

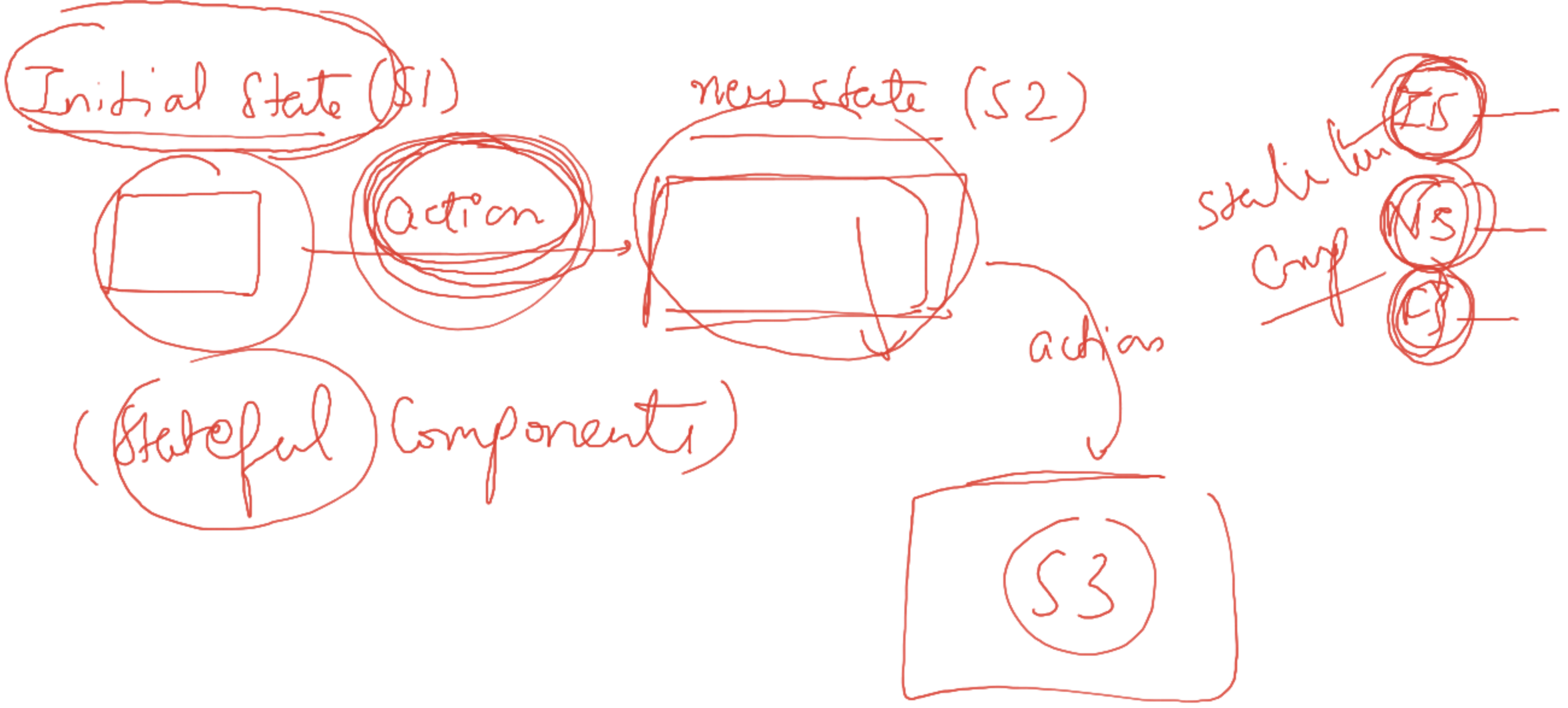
Appbar (S1)

State new

Add Cart

↓

AppBar got updated to new State (S2)


Initial State (S1)
action
new state (S2)
action
S3
(Stateful Components)
Stateless
Comp
F5
N5
F5

* State Management is all about managing all the changes of your application

* It is management of all Stateful Components

Why ?

To update the UI with the new state of the application.

How ?

Mo

Redux - SML for React ✓

React >16 (React Hooks) ✓

↳ Inbuilt functions which helps you to manage state of your application.

React Hooks >= 16

What is a Hook ?

Inbuilt Function

Why Hooks ?

State management

How ?

1) useState()

↳ Used for LOCAL state management.

Implement

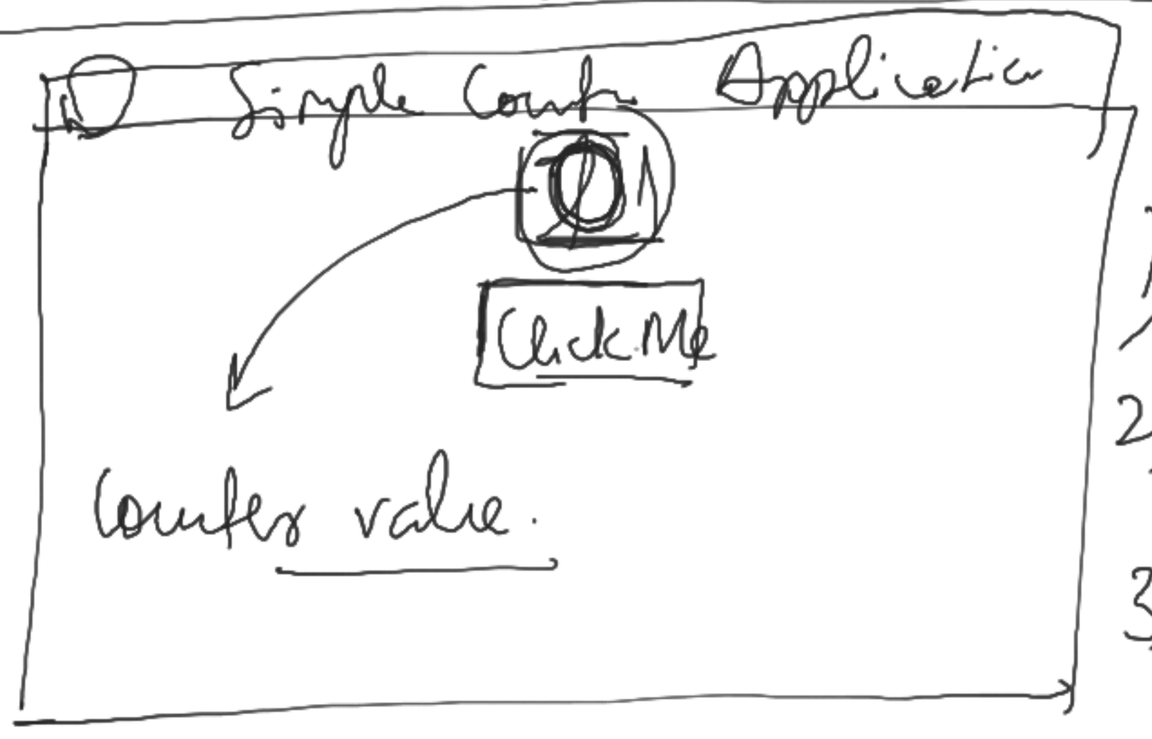
Step (1) Import { useState } from 'React'; ← object destructing

Step (2) Call useState & give it some Initial State

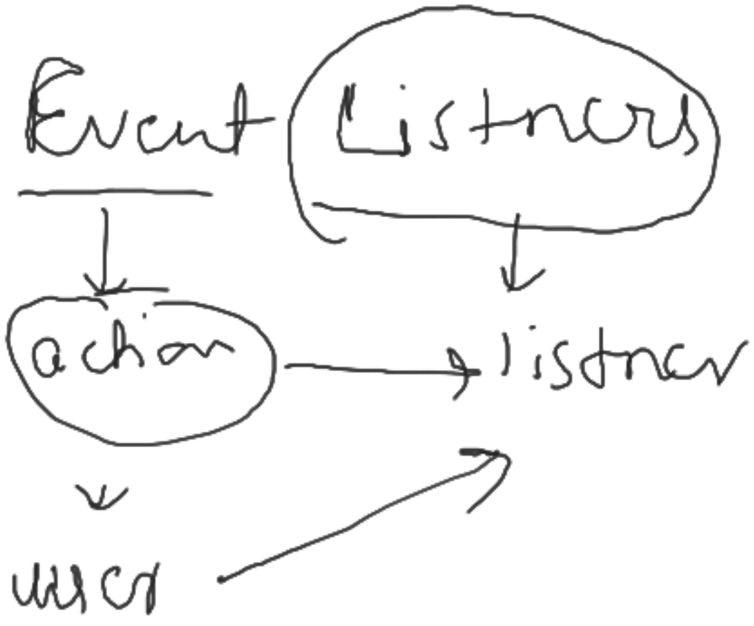
Step (3) use the state where you want.

Project

Simple Counter Application



1) State ?
2) what is that state
3) Implement web 3 step formula



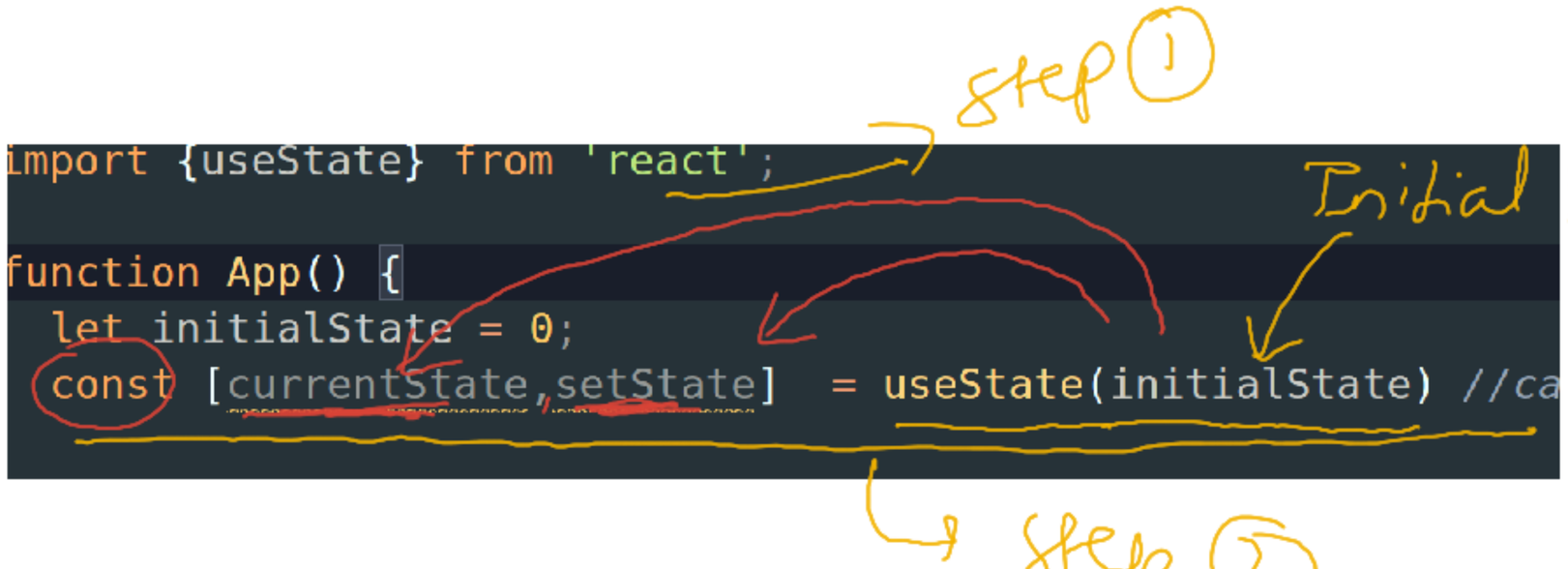
eg) onClick , onHover , onChange

help you to check

if user is performing what kind of action

why is counter value not changing ?

★ React only re-renders the Component if & if you explicitly tell React that in this Comp something is going to change.



useState → current state (variable)

→ function which is used to update the state

* This is the function which tells React "Hey React plz render the component somethings has changed here".

Next State

```
//incremneting counter
setcount(count+1)
```

funcs which is used to update the state
(tells React, please rerender the Comp,
bcz something has changed here")