JS

Advance React Topics

Saad Irfan
@DevWithSaad

JS

1. Lazy Loading

React puts all components in bundle.js. As the app grows, the bundle.js file grows.

So you should split your codebase.

The **React.lazy** function lets you lazy load a component. So the component only loads in the bundle when it is required. This decreases the bundle size.

🤯

Saad Irfan
@DevWithSaad

JS

1. Lazy loading (Continued)

The lazy-loaded component should be inside React **Suspense** component, as it allows showing a fallback when the component is loading.

You can find the code example link in bio.

Saad Irfan
@DevWithSaad

JS

2. HoC

Sometimes, different components require the same logic. We can do this with Higher Order Components (HoC).

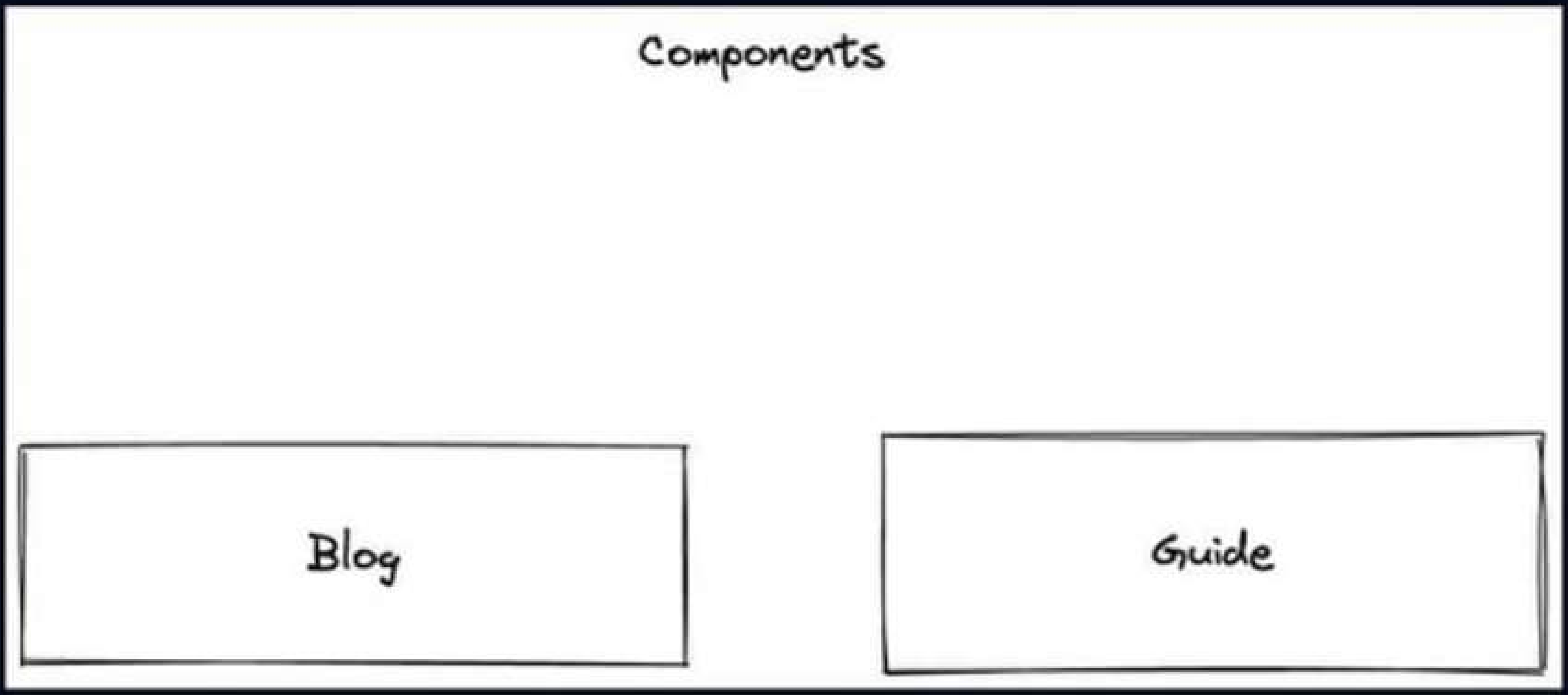
HoC has separate states and functionality but only one code. It is a function that takes a component and returns a new component.

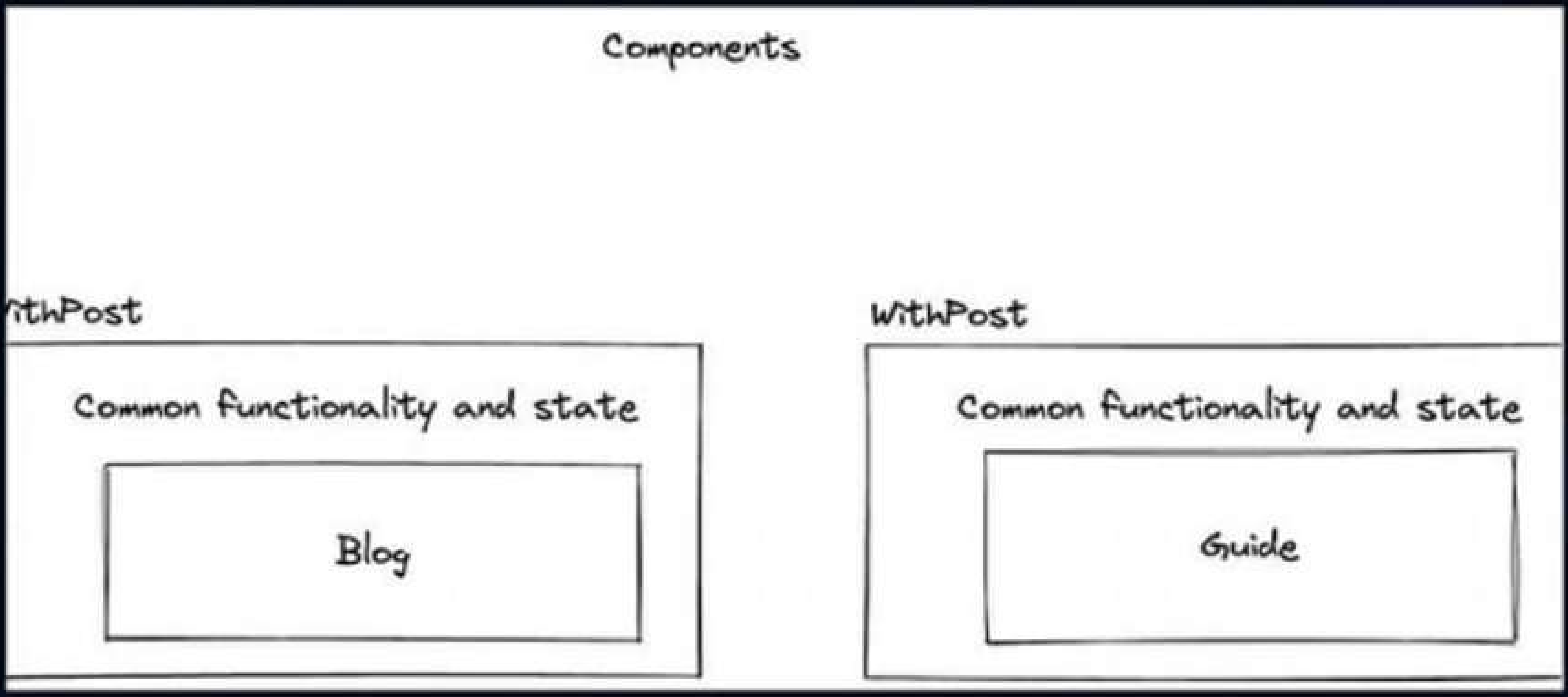
Naming convention: with
E.g., withHover

Saad Irfan
@DevWithSaad

JS

2. HoC (Continued)




Components
ithPost
Common functionality and state
Blog
WithPost
Common functionality and state
Guide

Saad Irfan
@DevWithSaad

JS

3. Optimization

React renders a component and its sub-components (including all the functions) every time the parent component's state or props change.

If a child component calls an API, it will make the API call every time the parent component renders.

Saad Irfan
@DevWithSaad

JS

3. Optimization (Continued)

useMemo is a React hook. A function inside useMemo will only execute if the value/s inside the dependency array changes.

This optimization helps to avoid expensive calculations on every render.

useMemo returns a **value**.

🔥

Saad Irfan
@DevWithSaad

←

JS

3. Optimization (Continued)

useCallback is another React hook. It also takes a function that executes if the value/s inside the dependency array changes.

The difference between useMemo and useCallback is useMemo returns a **value**, whereas useCallback returns a **memoized** version of the provided **callback function**.

Saad Irfan
@DevWithSaad