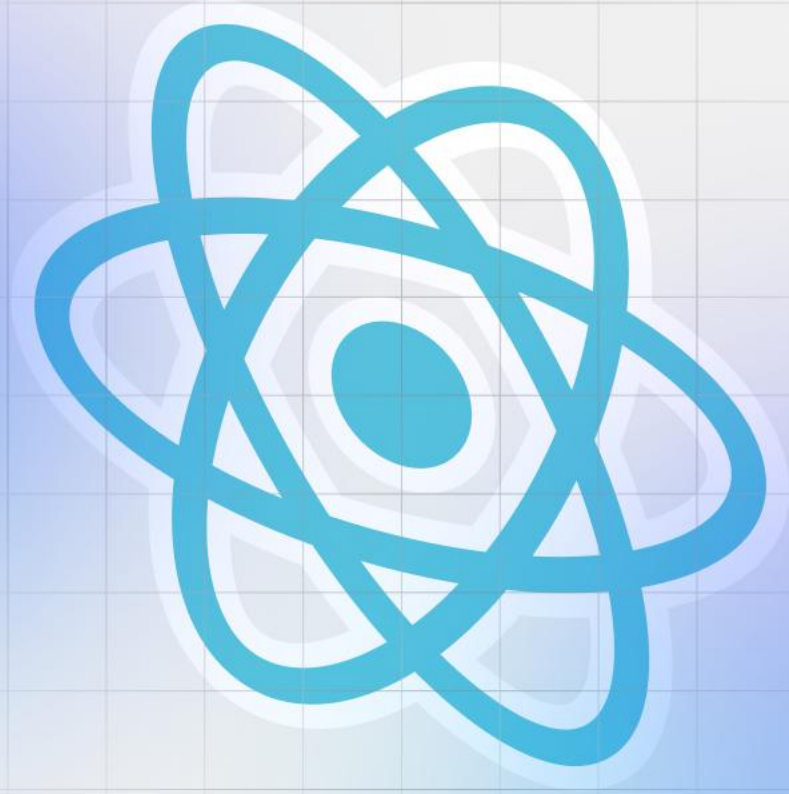


ReactJS

Introduction

Day 1





Hi!, I'm **Ninno**

I'm a **Fullstack Engineer**

Mohammad Khusnul Khuluq

At Keisuugiken, Hayago, Sencra, Obayan



Software Engineer

At Keisuugiken keisuu.co.jp



CTO, Fullstack Engineer

At PT Hayago Robotika Indonesia hayago.id



Founder, Fullstack Engineer

At Obayan obayan.id



Founder, Fullstack Engineer

At Sencra sencra.io

3D Design

UI/UX

Software Engineer

AI/ML

Graphic Design

Electronics / IoT

Robotics Developer

@ [khulqu.py](https://github.com/khulqu.py)

@ [yorozuya.ninno](https://twitter.com/yorozuya.ninno)

Pengenalan

1. Pengenalan React Js
2. Fundamental Refresh
3. React Component
4. Setup View UI

5. Mini Challenge

Tujuan

1. Memahami ReactJs
2. Membuat Proyek Portofolio
3. Create JSX, props, UI
4. Create Simple Component

5. Gokzs Reward

Agenda & Alokasi Waktu

07.00-07.20

Kickoff & konteks React di industri.

07.20-08.00

ES6 Refresh Reactjs.

08.00-08.40

Konsep Reactjs.

08.40-09.00

QnA Sesi 1

09.00-09.40

Setup proyek, Vite + Tailwind.

09.40-10.20

**Live coding: komponen
Button & Badge.**

10.20-10.50

Mini challenge individual.

10.50-11.00

Preview & Appendix



ReactJs ?



Dipakai luas di
web modern

Ekosistem kuat



Deklaratif,
State → UI

lebih mudah diprediksi.



Reusable komponen
scalable & rapi.

Cocok untuk
landing page
hingga **SPA**
System.

NEXT.JS



ES6 Refresh



`const`

nilai yang tidak di-reassign.



arrow function →

singkat, lexical **this**



Hello, `${name}`

Template literal



es6.jsx

```
const greet = (name) => `Hi, ${name}!`;  
let counter = 0; counter += 1;
```

ES6 Refresh



Destructuring

memudahkan akses properti.



Spread (...arr)

gabung/kloning koleksi.



Rest (...rest)

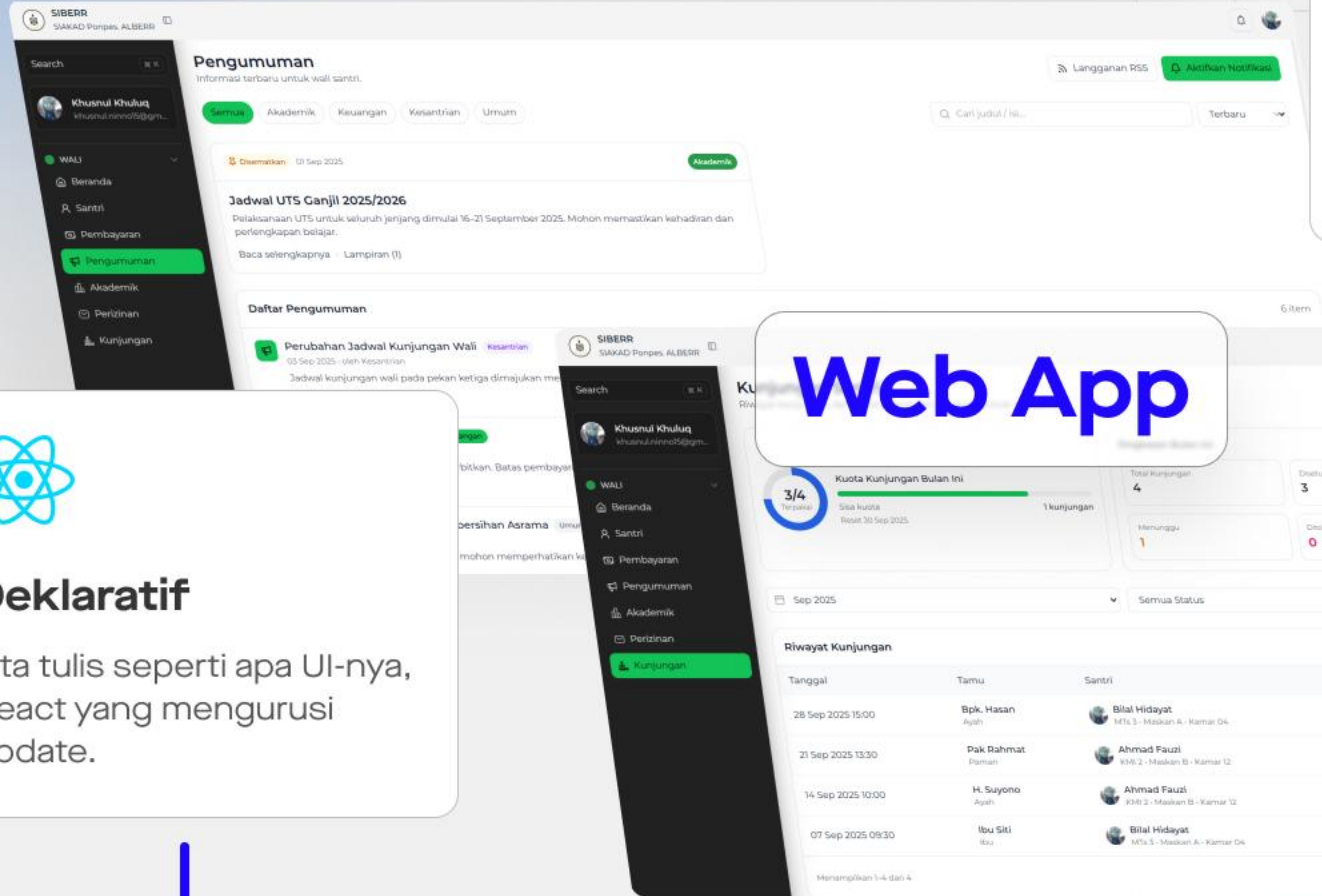
menampung sisa properti/argumen.



es6.jsx

```
const user = { name: 'Nino', kelas: 'XII RPL', score: 92 };  
const { name, ...infoLain } = user;  
  
const base = ['HTML', 'CSS'];  
const skills = [...base, 'JS', 'React'];
```

Main Concept



Virtual DOM

efisiensi render

es6.jsx

Web App



Deklaratif

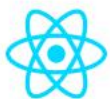
kita tulis seperti apa UI-nya,
React yang mengurus
update.



Komponen

fungsi yang merender JSX,
disusun modular.

ReactJS Rules

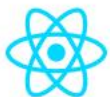


Mirip HTML

tapi di JS → gunakan {...}
untuk ekspresi JS.



Gunakan **className**
bukan **class**



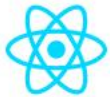
return

hanya boleh punya 1 parent
element.

```
function Hello({ name }) {  
  return <h1 className="text-xl">Hi, {name}!</h1>;  
}
```

HTML KOK DIRETURN ?

Props & Komponen Reusable



Props

Parameter yang membuat komponen fleksibel.



Penamaan komponen

PascalCase.



Komposisi

Komponen bisa memuat komponen lain.

```
function Badge({ text }) {  
  return <span className="badge">{text}</span>;  
}
```

Setup Proyek ReactJs

Requirements:



NodeJs
Latest Version



NPM
Latest Version



install

```
npm create vite@latest my-landing -- --template react
cd my-landing
npm i
npm run dev          # http://localhost:5173
```

Struktur penting



```
src/
  App.jsx
  main.jsx
index.html
```

Tambah Tailwind CSS

install

```
npm i -D tailwindcss postcss autoprefixer  
npx tailwindcss init -p
```

es6.jsx

tailwind.config.js

```
export default {  
  content: ["../index.html","./src/**/*.  
    {js,jsx,ts,tsx}"],  
  theme: { extend: {} },  
  plugins: [],  
}
```

src/index.css

```
@tailwind base;  
@tailwind components;  
@tailwind utilities;
```



Struktur Awal App

es6.jsx

Components/Container.jsx

```
export default function Container({ className = "", children }) {  
  return <div className={`mx-auto max-w-6xl px-4 sm:px-6 ${className}`}>{children}</div>;  
}
```

Button

Components/Button.jsx

badge

Components/Badge.jsx

Mini Challenge

1. Buatlah components untuk Input
(**Components/Input.jsx**)
2. Buatlah components untuk button primary dan outline
(**Components/ButtonVariant.jsx**)
3. Tampilkan di App.jsx

Ringkasan

github.com/khulqu15/reactjs-learn

1. Memahami dasar React, JSX, props.
2. Setup Vite + Tailwind.
3. Membuat komponen Button & Badge & Input



Next

1. State, event handler reactjs
2. Rancang layout Navbar, Hero, Footer.

rules.jsx