



Frontend Developer

ReactJS Vite: Membangun Web Modern
Bootcamp JuaraCoding 2025



Pengantar: Struktur Proyek Modern

Memahami struktur proyek adalah langkah pertama untuk bekerja dengan efisien.

Proyek React yang dibuat dengan Vite dan TypeScript memiliki struktur yang ringkas, bersih, dan sangat terorganisir. Pendekatan ini memisahkan file konfigurasi dari kode sumber dan membuat alur kerja lebih intuitif.



Struktur Folder Utama

Berikut adalah gambaran umum struktur folder utama yang akan Anda temukan setelah menjalankan perintah `npm create vite@latest`. Setiap folder memiliki peran yang jelas.

- `node_modules/`: Berisi semua library dan dependensi yang diperlukan proyek Anda.
- `public/`: Folder untuk aset statis yang tidak perlu diolah oleh Vite, seperti gambar atau favicon.
- `src/`: Folder utama yang berisi semua kode sumber aplikasi Anda.
- `.eslintrc.cjs`, `.gitignore`, `index.html`, `package.json`, `README.md`, `tsconfig.json`, `vite.config.ts`: File-file konfigurasi proyek.



Inti Aplikasi: Folder src/

Folder src/ adalah jantung dari proyek Anda. Di sinilah semua kode aplikasi yang sebenarnya berada. Struktur di dalamnya sangat penting untuk menjaga proyek tetap terorganisir.

- assets/: Tempat untuk menyimpan aset-aset seperti gambar, ikon, atau file CSS global.
- components/: Folder untuk komponen-komponen yang dapat digunakan kembali, seperti tombol, navigasi, atau kartu produk.
- pages/: Folder untuk komponen yang merepresentasikan halaman lengkap, seperti HomePage.tsx atau AboutPage.tsx.
- App.tsx: Komponen utama yang menjadi wadah bagi seluruh aplikasi Anda.
- main.tsx: Titik masuk utama (entry point) aplikasi Anda, di mana aplikasi React di-render ke DOM.



Titik Masuk: index.html

File index.html adalah halaman HTML tunggal yang akan dimuat oleh browser.

Ia berfungsi sebagai "wadah" untuk seluruh aplikasi React Anda. File ini hanya berisi satu elemen `<div id="root"></div>` di dalam `<body>`. Skrip main.tsx akan secara otomatis dimasukkan oleh Vite dan bertanggung jawab untuk merender komponen App ke dalam elemen root ini.



Konfigurasi Penting: package.json & tsconfig.json

Dua file konfigurasi ini sangat vital untuk mengelola proyek.

- **package.json:** Mengelola dependensi proyek, skrip yang dapat dijalankan (scripts), dan metadata proyek. Anda akan sering menggunakan skrip seperti `npm run dev` atau `npm run build`.
- **tsconfig.json:** File konfigurasi TypeScript. Di sinilah Anda menentukan aturan kompilasi, seperti versi JavaScript target dan pengaturan strictness tipe data. Vite secara otomatis membaca konfigurasi ini.



Alur Kerja Sederhana

Memahami alur kerja membantu Anda menempatkan file dengan benar.

- Pengembangan: Anda menulis kode di file `.tsx` di dalam folder `src/`.
- Vite: Saat Anda menjalankan `npm run dev`, Vite memproses kode Anda. Vite menggunakan konfigurasi `tsconfig.json` untuk memeriksa tipe TypeScript dan mengubah kode menjadi JavaScript.
- Browser: Kode JavaScript yang telah diproses kemudian disajikan ke browser yang memuat `index.html`.
- React: React mengambil alih dan merender komponen `App.tsx` ke dalam elemen `<div id="root">`.



Memulai dengan React Hooks

React Hooks adalah fitur yang diperkenalkan di React 16.8 yang memungkinkan Anda "terhubung" ke state dan fitur siklus hidup React dari komponen fungsional.

Sebelum Hooks, fitur ini hanya tersedia di komponen kelas. Hooks memungkinkan penulisan kode yang lebih bersih, lebih ringkas, dan lebih mudah dipahami. Ini adalah standar modern dalam pengembangan React.



useState: Mengelola State Komponen

useState adalah hook paling dasar dan paling sering digunakan.

Ia memungkinkan Anda menambahkan state ke komponen fungsional. **useState** **mengembalikan** sepasang nilai: variabel state saat ini dan fungsi untuk **memperbarui** state tersebut. Ketika fungsi pembaruan dipanggil, React akan secara otomatis merender ulang komponen dengan nilai state yang baru.

