



Módulo 1: Entorno y Setup moderno (Tailwind v4)



Objetivos

- Crear un proyecto React 19 con Vite.
 - Integrar Tailwind v4 con PostCSS (sin `init`).
 - Configurar ESLint + Prettier para un código limpio.
-



Paso 0: Requisitos

- Node.js 18+ (tú tienes 22, perfecto).
 - npm (incluido con Node).
-



Paso 1: Crear proyecto con Vite

```
npm create vite@latest nombre_proyecto -- --template react
cd nombre_proyecto
npm install
```

Si usarás TypeScript, cambia `--template react` por `--template react-swc-ts` y ajusta extensiones a `.tsx`.



Paso 2: Instalar Tailwind v4 + PostCSS

En la Terminal correr:

```
npm i -D tailwindcss @tailwindcss/postcss @tailwindcss/cli
```

Crea `postcss.config.js` en la raíz:

```
// postcss.config.js
export default {
  plugins: {
    "@tailwindcss/postcss": {},
  }
}
```

```
},  
}
```

Crea/edita **src/index.css**:

```
@import "tailwindcss";  
  
/* (Opcional) Personaliza tokens con v4:  
@theme {  
  --color-brand: #4f46e5;  
  --spacing-xxs: 0.25rem;  
}  
*/
```

Importa el CSS en **src/main.jsx** (Vite ya lo genera, verifica que esté):

```
import ReactDOM from 'react-dom/client'  
import App from './App.jsx'  
import './index.css' // <- Asegúrate de tener esta línea  
import './index.css' // (deja solo una si ya existía)  
ReactDOM.createRoot(document.getElementById('root')).render(<App />)
```



Paso 3: ESLint + Prettier

En la terminal correr:

```
npm i -D eslint prettier eslint-config-prettier eslint-plugin-react eslint-plugin-react-hooks
```

Crea **.eslintrc.json**:

```
{  
  "env": { "browser": true, "es2021": true },  
  "extends": [  
    "eslint:recommended",  
    "plugin:react/recommended",  
    "plugin:react-hooks/recommended",  
    "prettier"  
,  
    "parserOptions": { "ecmaVersion": "latest", "sourceType": "module" },  
    "plugins": ["react", "react-hooks"],  
    "rules": { "react/react-in-jsx-scope": "off" },  
    "settings": { "react": { "version": "detect" } }  
}
```

Crea `.prettierrc`:

```
{  
  "semi": false,  
  "singleQuote": true,  
  "trailingComma": "es5"  
}
```



Paso 4: Prueba rápida de Tailwind

Reemplaza `src/App.jsx` por algo visible:

```
export default function App() {  
  return (  
    <main className="min-h-dvh bg-gray-50 grid place-items-center p-6">  
      <section className="w-full max-w-xl rounded-2xl border bg-white p-8  
shadow">  
        <h1 className="text-3xl font-bold tracking-tight">React 19 + Tailwind  
v4</h1>  
        <p className="mt-2 text-gray-600">  
          Setup moderno funcionando. Edita <code>src/App.jsx</code> y guarda  
para ver cambios.  
        </p>  
  
        <div className="mt-6 grid grid-cols-1 gap-4 sm:grid-cols-3">  
          <div className="rounded-xl border p-4">  
            <p className="text-sm text-gray-500">Estado</p>  
            <p className="text-lg font-semibold">OK </p>  
          </div>  
          <div className="rounded-xl border p-4">  
            <p className="text-sm text-gray-500">Tailwind</p>  
            <p className="text-lg font-semibold">v4 (CSS-first)</p>  
          </div>  
          <div className="rounded-xl border p-4">  
            <p className="text-sm text-gray-500">Build</p>  
            <p className="text-lg font-semibold">Vite</p>  
          </div>  
        </div>  
  
        <button className="mt-6 inline-flex items-center justify-center  
rounded-xl border px-4 py-2 font-medium hover:bg-gray-50 active:scale-[0.98]">  
          Botón Tailwind  
        </button>  
      </section>  
    </main>  
  )  
}
```

Levanta el dev server en la terminal:

```
npm run dev
```

Abre la URL que te muestre Vite (normalmente `http://localhost:5173`).