

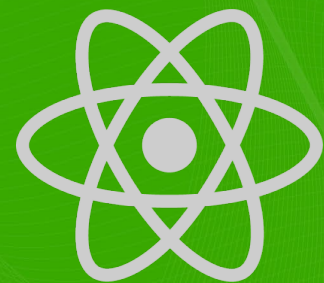


Fundamentos de React

Centro de Servicios y Gestión Empresarial
SENA Regional Antioquia

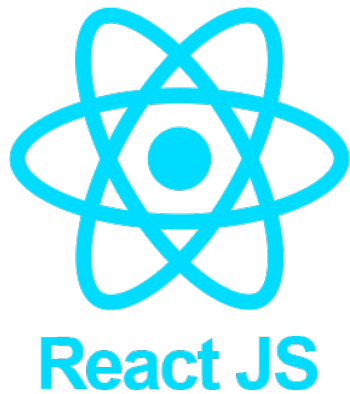


www.sena.edu.co



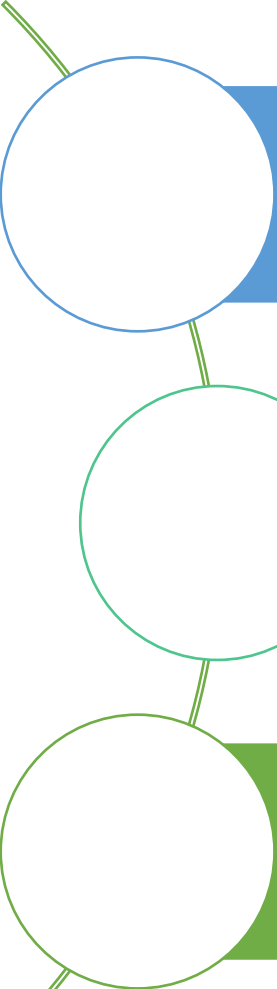
Conceptos

Conceptos - React



React es una biblioteca de JavaScript utilizada para **construir interfaces de usuario**, especialmente aplicaciones de una sola página (SPA, por sus siglas en inglés). Fue desarrollada por Meta e implementada inicialmente en Facebook y se destaca por su capacidad para crear aplicaciones rápidas, interactivas y eficientes.

Características - React

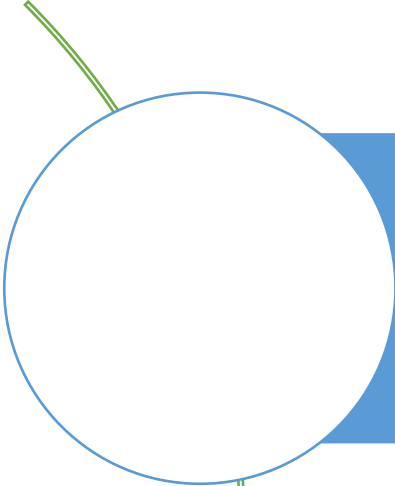
A diagram on the left side of the slide shows three white circles with green outlines, connected by green lines. The circles are arranged vertically, with the top circle connected to the middle one, and the middle one connected to the bottom one. Each circle is positioned to the left of a corresponding text block.

Componentes: React se basa en el uso de componentes reutilizables, lo que facilita la creación y gestión de la interfaz de usuario.

JSX: Es una extensión de la sintaxis de JavaScript que permite escribir código HTML dentro de JavaScript, lo que facilita la creación de interfaces.

Virtual DOM: React utiliza un DOM virtual que mejora el rendimiento al actualizar solo los componentes que han cambiado en lugar de volver a renderizar todo el DOM.

Características - React



Unidirectional Data Flow: Los datos en React fluyen en una sola dirección, desde los componentes padres hacia los hijos, lo que facilita el seguimiento de cambios y el control del estado de la aplicación.

The diagram shows a light blue circle on the left, connected by a thin blue line to a blue rectangular box on the right. The circle has a small line extending from its top-left edge.

React Hooks: Introducidos en React 16.8, permiten utilizar el estado y otras características de React sin escribir clases.

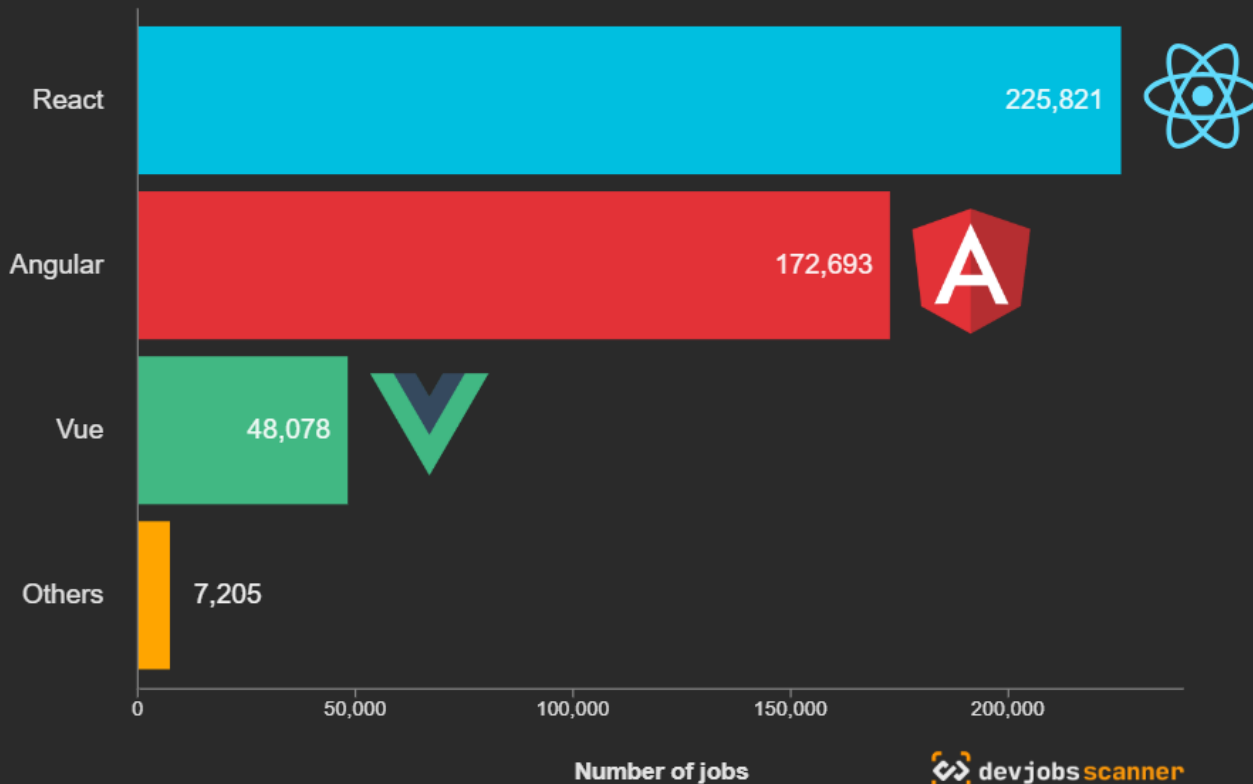
The diagram shows a light green circle on the left, connected by a thin green line to a green rectangular box on the right. The circle has a small line extending from its bottom-left edge.

¿Cuál es el framework frontend más demandado?



Number of jobs by Frontend Framework

From 01-Nov-2022 to 31-Dec-2023

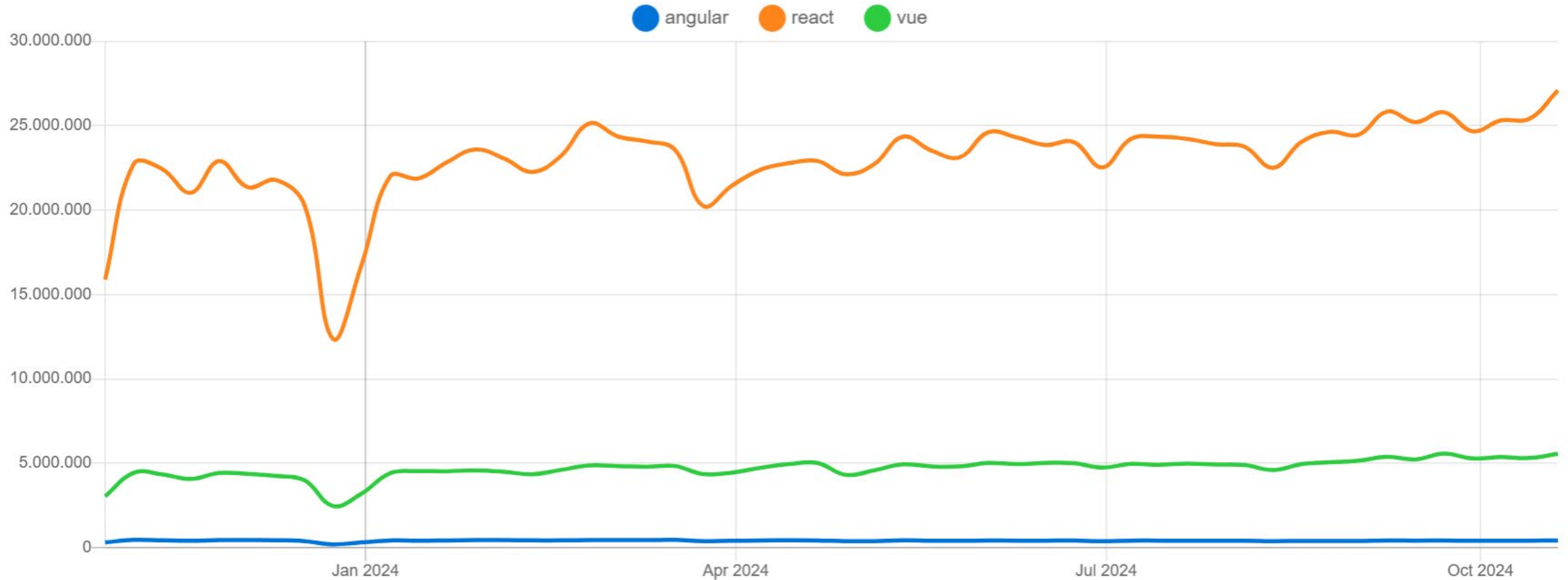


<https://www.devjobsscanner.com/blog/the-most-demanded-frontend-frameworks/>

npm trends → angular vs react vs vue



Downloads in past 1 Year ▾



<https://npmtrends.com/angular-vs-react-vs-vue>

Programación Imperativa

En la **programación imperativa**, le dices al programa *cómo* hacer algo paso a paso, definiendo explícitamente el flujo de control. Te enfocas en describir el proceso de solución, especificando cada instrucción para lograr el resultado deseado.

Enfoque en el *cómo* hacerlo.

- Define una serie de pasos secuenciales.
- Se controla el estado del programa a través de variables y ciclos.
- Es común en lenguajes como C, Java, Python, y otros.

Programación Declarativa

En la **programación declarativa**, en lugar de describir cómo hacer algo paso a paso, te enfocas en **qué** quieres lograr, dejando que el sistema se encargue de los detalles de cómo ejecutarlo. No describes el flujo de control detallado, sino el resultado deseado.

Enfoque en el *qué* se quiere lograr.

- Se omiten los detalles de control de flujo o manipulación de estado.
- Frecuentemente se usa en lenguajes y paradigmas como SQL, HTML, CSS, y lenguajes funcionales como React.



Imperativo VS Declarativo



```
// Gestion de Ventas
const ventas = [100, 300, 450, 600, 700, 850];
```

```
// Imperativo
function sumarVentas(ventas) {
  let total = 0;
  for (let i = 0; i < ventas.length; i++) {
    total += ventas[i];
  }
  return total;
}
console.log(sumarVentas(ventas));
```

```
// Declarativo
const sumarVentasDeclarativo =
ventas.reduce(
  (acumu, venta) => acumu + venta, 0);
console.log(sumarVentasDeclarativo);
```

JSX – React

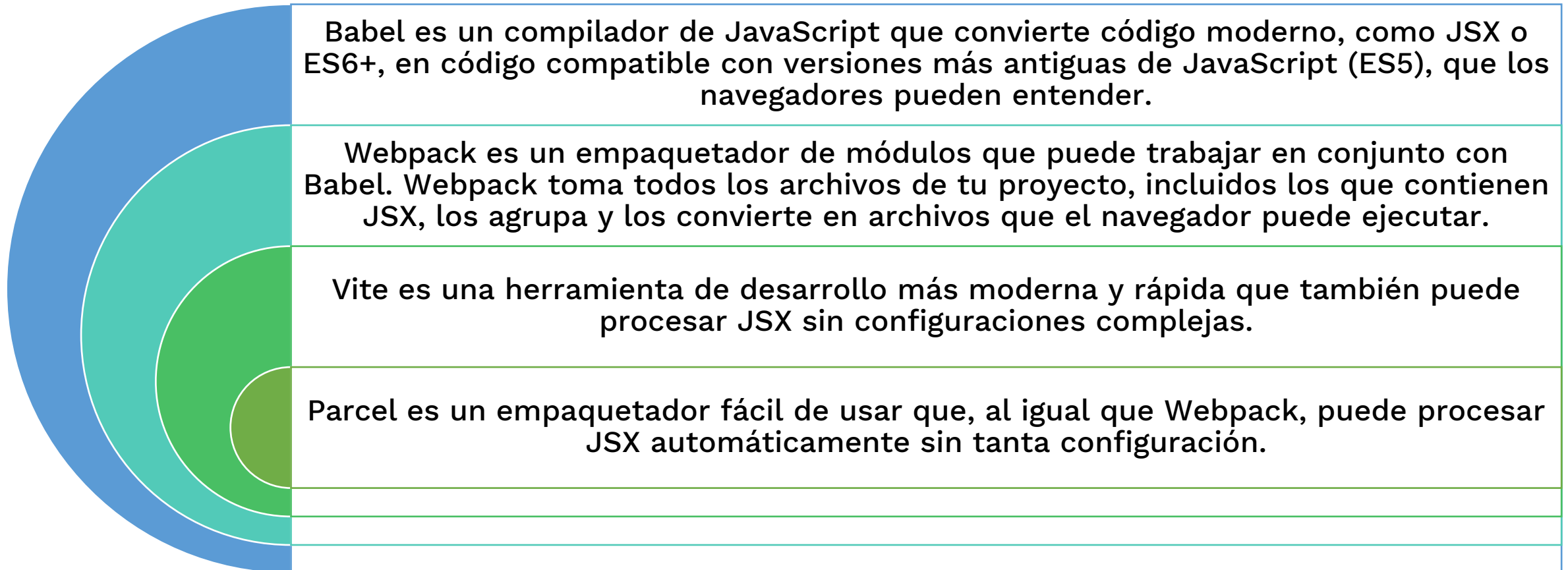
JSX (*JavaScript XML*) es una extensión de la sintaxis de JavaScript utilizada en React para escribir estructuras de la interfaz de usuario de forma similar a HTML, pero dentro de código JavaScript.

Dado que los navegadores no pueden interpretar JSX de forma nativa, se necesita un "**empaquetador**" o **compilador** para transformar este código en JavaScript puro que el navegador pueda ejecutar.



¿Qué empaquetadores se utilizan con JSX?

Los empaquetadores más comunes que se utilizan para procesar JSX son:

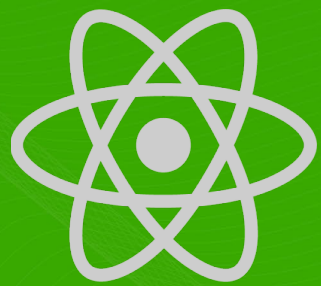


Vite - Empaquetador

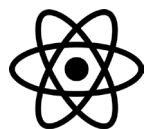
Es una herramienta de desarrollo moderna y rápida para construir aplicaciones web, creada por Evan You, el mismo autor de Vue.js. Vite se ha vuelto muy popular en proyectos de JavaScript debido a su rapidez y facilidad de configuración, especialmente cuando se trabaja con frameworks como React, Vue y otros.

Principales características de Vite:

- **Inicio instantáneo:** Vite puede iniciar el servidor en muy poco tiempo, incluso en proyectos grandes.
- **Hot Module Replacement (HMR):** Vite recarga rápidamente solo los módulos modificados cuando se hace cambios, sin perder el estado actual de la aplicación.
- **Compatibilidad con JSX y TypeScript:** Vite soporta JSX y TypeScript sin configuraciones adicionales.
- **Plugins:** Vite tiene un sistema de plugins basado en Rollup, lo que permite agregar funcionalidades extra o integrar herramientas específicas.



Crear un proyecto de React + Vite



Crear un proyecto de React + Vite

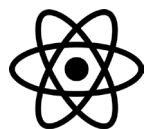
Para crear un proyecto de React con Vite, siga estos pasos:

1. Instalar Vite y configurar el proyecto

- A. Abrir una terminal en la ubicación donde desea crear el proyecto.
- B. Ejecutar el siguiente comando para crear un nuevo proyecto con Vite:

```
npm create vite@latest my-react-app
```

- Para el nombre del proyecto, puede presionar *Enter* para usar el nombre predeterminado (my-react-app) o escribir uno diferente.
- Luego, seleccione **React** como framework, y **JavaScript** o TypeScript según preferencias.



Crear un proyecto de React + Vite

1. Instalar Vite y configurar el proyecto

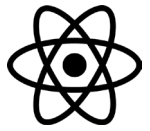
C. Entrar en la carpeta del proyecto:

```
cd my-react-app
```

D. Instalar las dependencias necesarias:

```
npm install
```

npm (*Node Package Manager*) es el **gestor de paquetes predeterminado** para Node.js. Es una herramienta que permite instalar, compartir y gestionar paquetes o librerías de JavaScript, facilitando el trabajo de los desarrolladores al incluir funcionalidades y dependencias de terceros en sus proyectos.



Crear un proyecto de React + Vite

2. Ejecutar el servidor de desarrollo

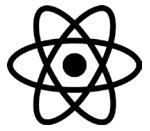
Para iniciar el servidor de desarrollo, usa el siguiente comando:

```
npm run dev
```

Esto iniciará el servidor y, en la terminal, verás un enlace (normalmente **http://localhost:5173**) donde puedes ver tu aplicación en el navegador.

```
VITE v5.3.1  ready in 6053 ms

→ Local:   http://localhost:5173/
→ Network: use --host to expose
→ press h + enter to show help
```



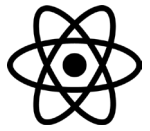
Crear un proyecto de React + Vite

3. Estructura del proyecto

La estructura básica del proyecto generada con Vite para React incluirá lo siguiente:

```
my-react-app/  
├── index.html  
├── package.json  
├── vite.config.js  
└── src/  
    ├── App.jsx  
    ├── main.jsx  
    └── assets/
```

1. **index.html:** Punto de entrada HTML donde se inserta la aplicación React.
2. **vite.config.js:** Archivo de configuración de Vite donde puedes personalizar ajustes (como plugins o alias).
3. **src/:** Carpeta principal para el código fuente.
 - **main.jsx:** Archivo de entrada de la aplicación donde se monta el componente App.
 - **App.jsx:** Componente raíz de tu aplicación.



Crear un proyecto de React + Vite

4. Empaquetar para producción

Para construir el proyecto para producción, ejecute:

```
npm run build
```

Esto generará una carpeta **dist** con los archivos optimizados y listos para desplegar en un servidor.

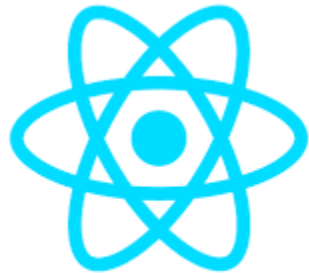
5. Servir la carpeta dist en un servidor estático

La carpeta **dist** ahora contiene todo lo necesario para que la aplicación funcione en un entorno de producción. Puedes usar cualquier servidor estático (como Nginx, Apache, Vercel, Netlify, etc.) para servir estos archivos.

```
npm run preview
```

```
→ Local:  http://localhost:4173/  
→ Network: use --host to expose  
→ press h + enter to show help
```

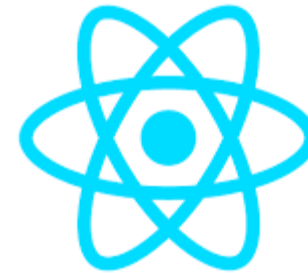

Material de apoyo



React JS

Documentación oficial:

<https://es.react.dev/>



React JS

Comunidad:

<https://www.reactjs.wiki/>



GRACIAS

Presentó: Alvaro Pérez Niño
Instructor Técnico

Correo: aperezn@misena.edu.co

<http://centrodeserviciosygestionempresarial.blogspot.com/>

Línea de atención al ciudadano: 01 8000 910270

Línea de atención al empresario: 01 8000 910682



www.sena.edu.co