**Nombre:** Mario Villamar

**Fecha:** 12/09/2023

**React**

React es una biblioteca de JavaScript para la construcción de interfaces de usuario (UI) interactivas y reactivas.

**Filosofía:** "Aprende una vez, escribe en cualquier lugar." React se enfoca en la reutilización de componentes y el manejo eficiente del estado para crear aplicaciones rápidas y escalables.

**Arquitectura:**

React sigue una arquitectura basada en componentes. Cada componente es una unidad independiente que encapsula su propia lógica y presentación.

El componente principal se llama "App" en la mayoría de los casos y sirve como punto de entrada.

Los datos fluyen de arriba hacia abajo (flujo unidireccional), lo que significa que los cambios en el estado del componente padre pueden afectar a los componentes hijos, pero no al revés.

El React Router se utiliza para gestionar la navegación en aplicaciones de una sola página.

**Cómo Funciona:**

**Creación de Componentes:** Diseñas componentes que representan partes de tu aplicación. Estos componentes pueden tener su propio estado interno y propiedades.

**Estado y Props:** Utilizas el estado para almacenar datos locales a un componente y las props para pasar datos de un componente padre a un hijo.

**Renderización:** React renderiza los componentes en función de su estado y props. Cada vez que el estado o las props cambian, React vuelve a renderizar los componentes relevantes.

**Virtual DOM:** React mantiene una copia virtual del DOM que utiliza para comparar con el DOM real. Esto permite actualizaciones eficientes, ya que React solo actualiza las partes que han cambiado.

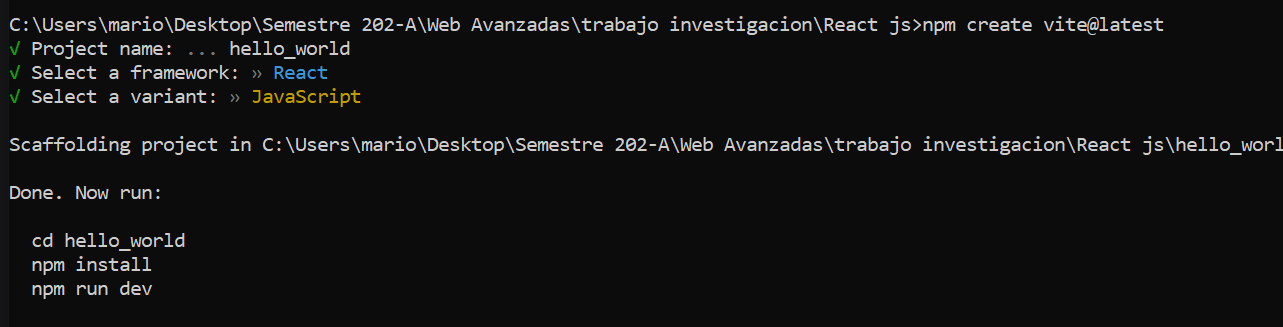
**Gestión de Eventos:** Puedes agregar interacción a tus componentes mediante la gestión de eventos, como clics o cambios de formulario.

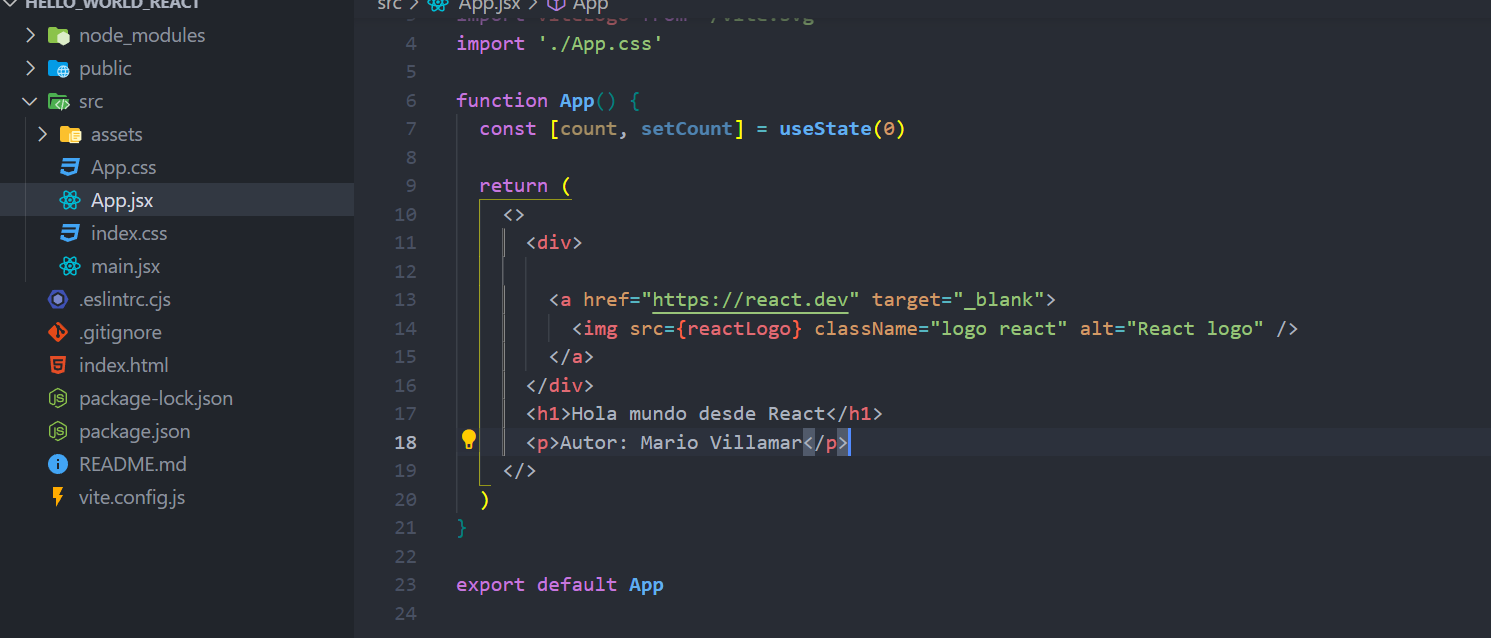
**Crear un Proyecto con Vite.js:**

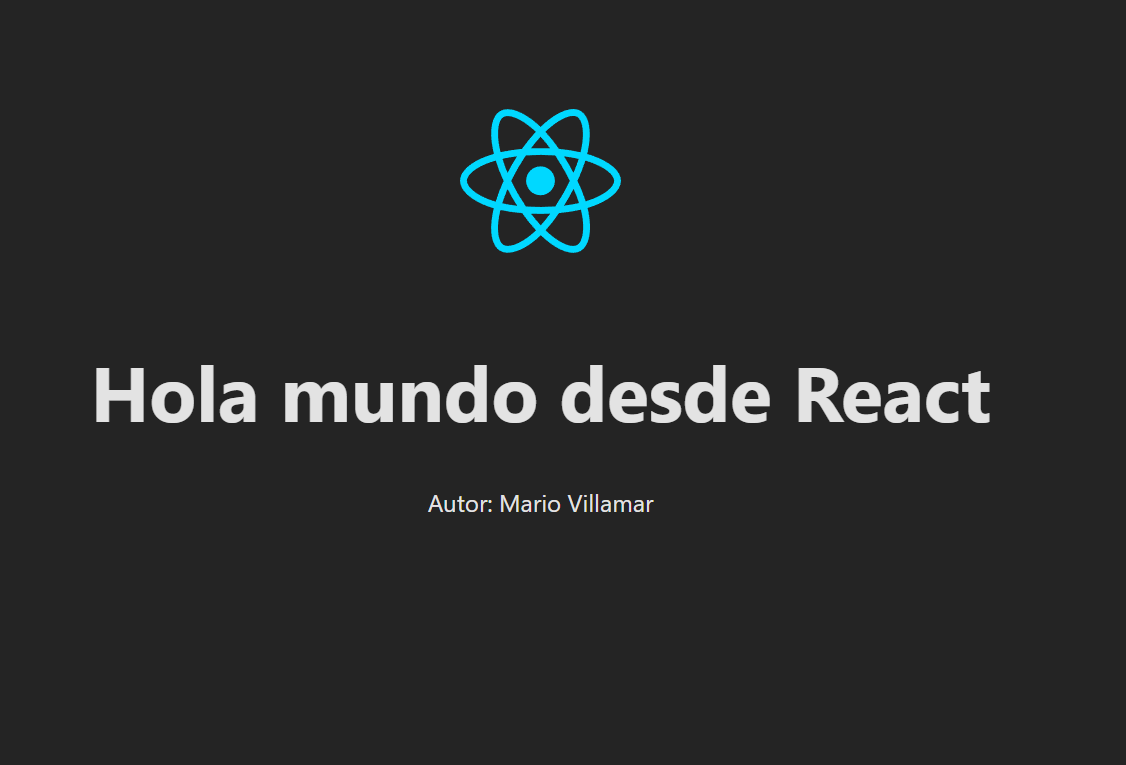
Para crear un proyecto React con Vite.js, primero se debe instalar Vite globalmente:

A black screen with white text

Description automatically generated

****





**Vue**

Vue.js es un framework de JavaScript para desarrollar aplicaciones web interactivas y basadas en componentes.

"Hacerlo simple." Vue se centra en la simplicidad y la facilidad de uso, permitiendo a los desarrolladores crear aplicaciones de manera rápida y eficiente mediante la declaración de la interfaz de usuario y la manipulación declarativa del DOM.

**Arquitectura:**

Vue sigue una arquitectura basada en componentes. Cada componente es una unidad autocontenida con su propia lógica y plantilla.

Utiliza un sistema de reactividad para rastrear y reaccionar a los cambios en los datos.

Vue Router se usa para manejar la navegación en aplicaciones de una sola página al igual que en React.

* **Componentes**: Vue utiliza componentes para organizar la interfaz de usuario y la lógica.
* **Directivas:** Las directivas como v-if y v-for simplifican la manipulación del DOM de manera declarativa.
* **Vue Router:** Proporciona enrutamiento para aplicaciones de una sola página.

**Cómo Funciona:**

**Creación de Componentes:** Diseñas componentes reutilizables, que consisten en una plantilla HTML y una sección de lógica JavaScript.

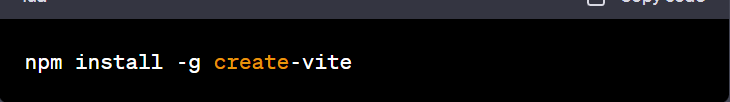
**Data Binding:** Utilizas la directiva v-model y otras directivas para enlazar datos y vistas de manera declarativa.

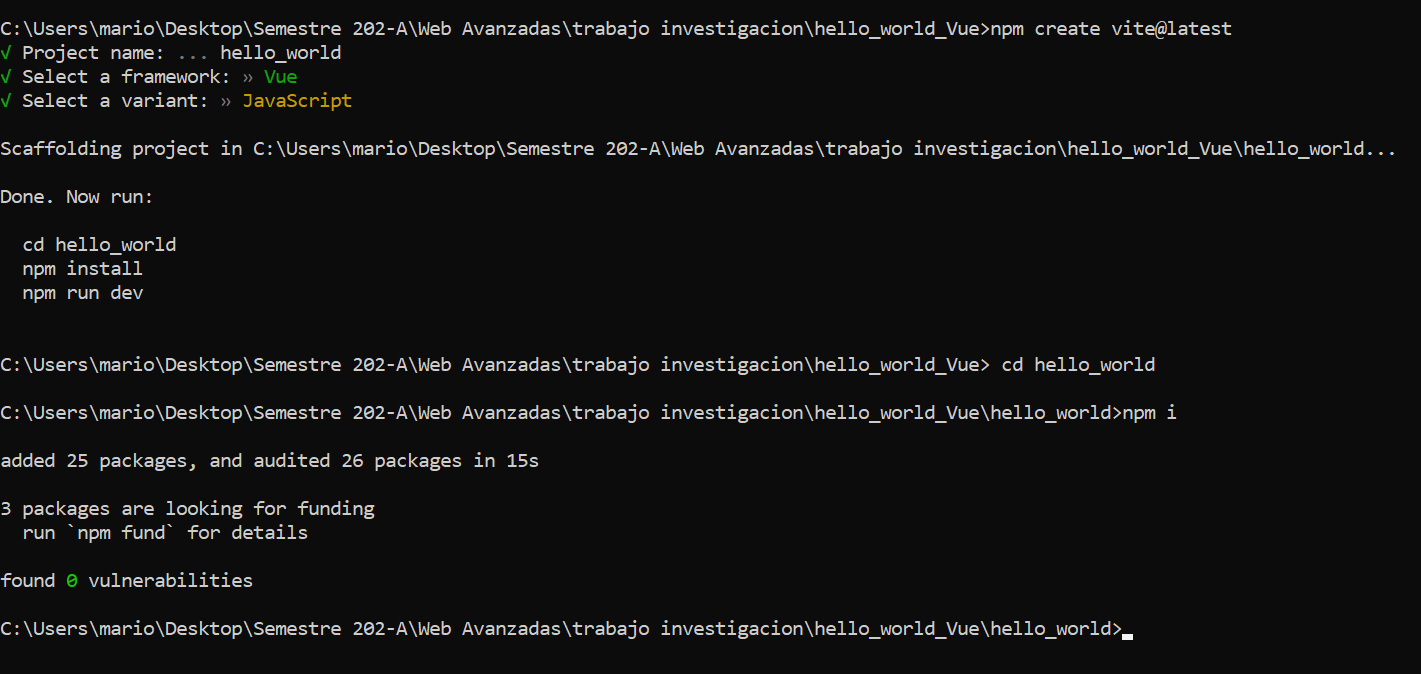
**Reactividad:** Vue rastrea automáticamente los cambios en los datos y actualiza la vista de manera eficiente. Esto se logra mediante el uso de objetos reactivos y getters y setters.

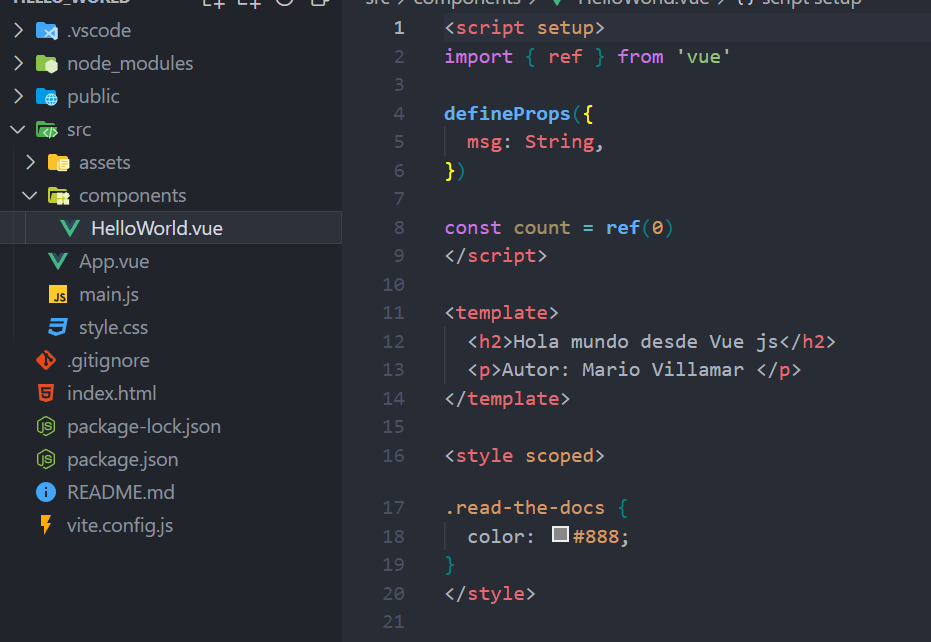
**Directivas:** Vue utiliza directivas como v-if, v-for y v-bind para manipular el DOM de manera declarativa y dinámica.

**Eventos y Métodos:** Puedes manejar eventos y definir métodos en tus componentes para agregar interacción.

**Creación de proyecto con vite:**



****

****

**A black background with a green and blue triangle

Description automatically generated**