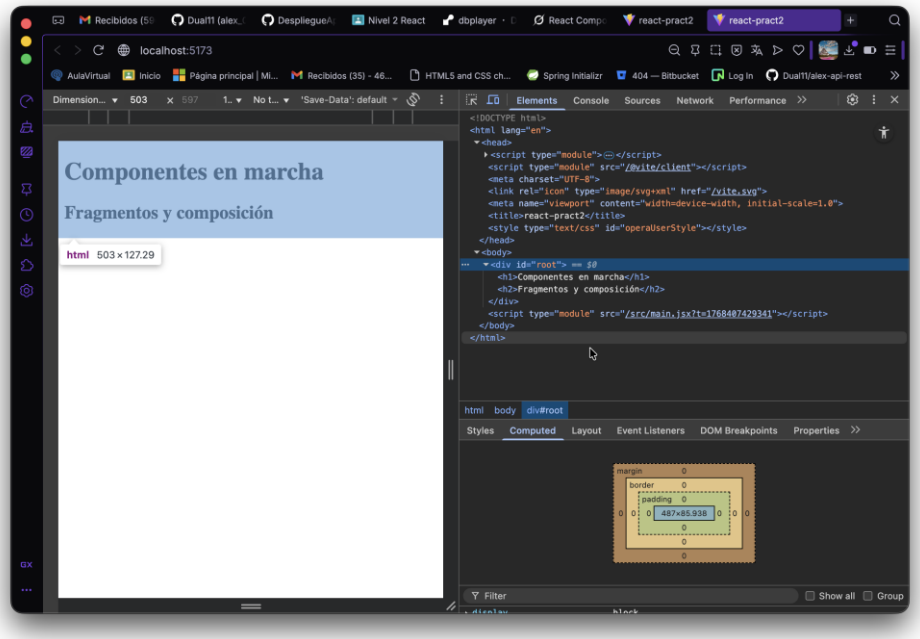
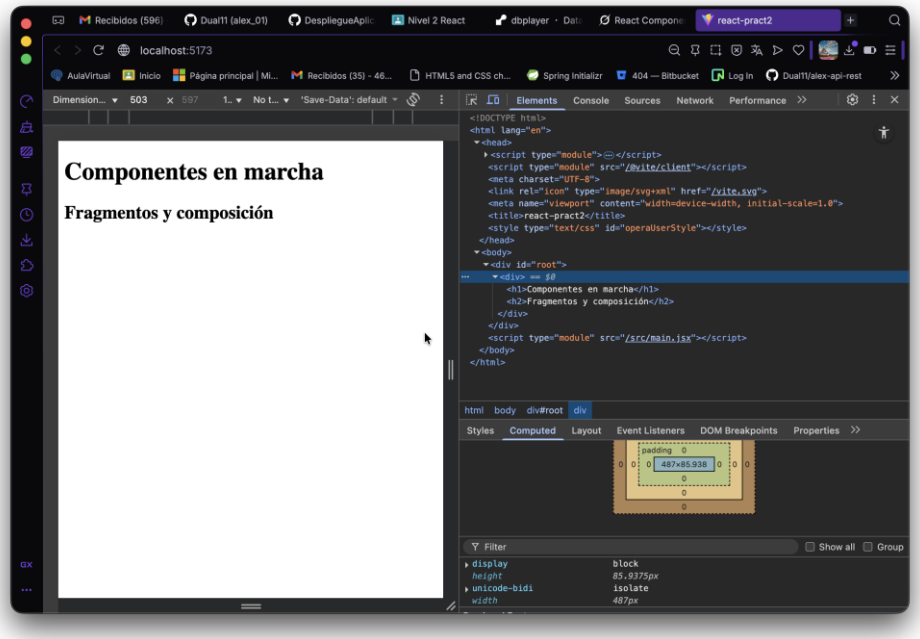
1- Cambiando el texto sin el div

Ha agregado el texto dentro del div root, con un div vacío.



2- He añadido el texto dentro de un div

La diferencia ahora es que, si ha añadido el texto en el div, ya que se lo hemos especificado.

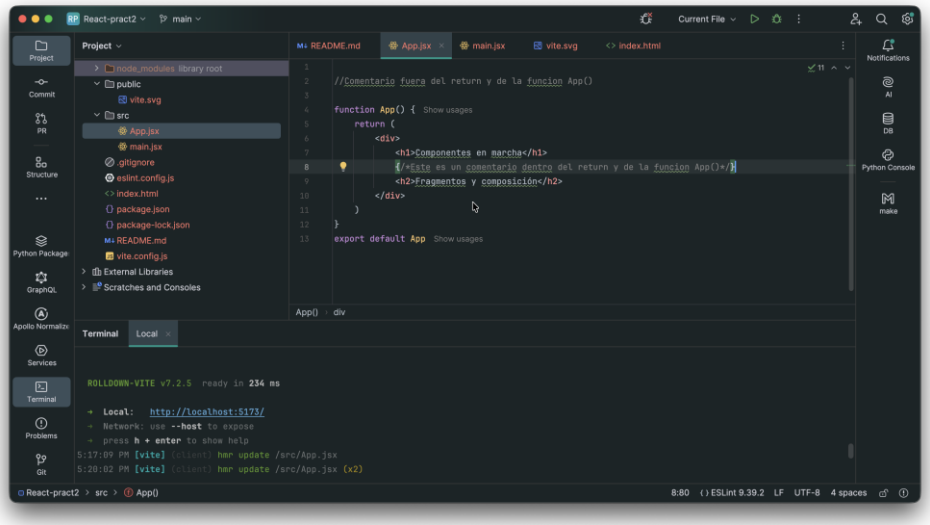


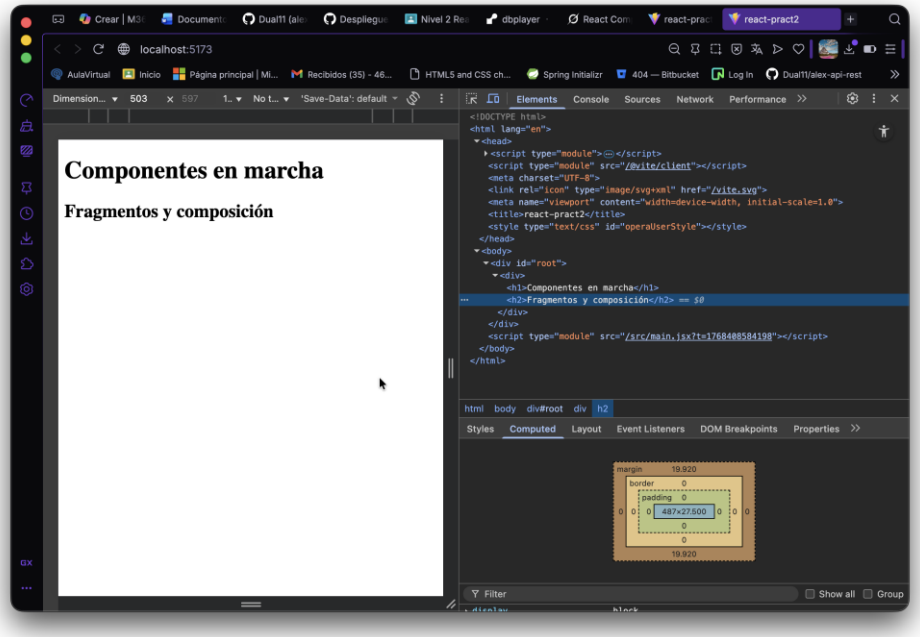
3- Renderizado: ¿dónde se “engancha” React?

document.getElementById('root'). Selecciona el elemento que está en el archivo index.html. Este div es el punto de entrada donde React "engancha" toda la aplicación. Luego, **ReactDOM.createRoot()** crea un root de renderizado en ese elemento y con. **render()** inserta el componente dentro de él, reemplazando el contenido existente.

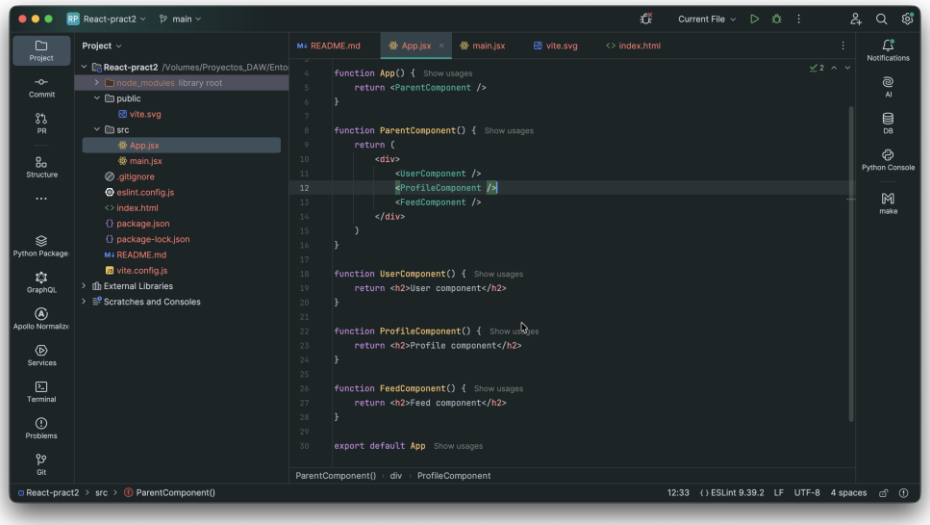
4- Agregando comentarios

He agregado los comentarios y no los muestras





5- He creado los tres componentes



PREGUNTAS EXTRAS

1- ¿Qué es un componente en React?

Un componente en React es una pieza reutilizable e independiente de la interfaz del usuario, esto permite dividir la aplicación en pequeñas partes.

2- ¿Por qué usamos Fragmentos (<>...</>) en lugar de un ?

Los Fragmentos permiten devolver varios elementos hermanos sin añadir un nodo extra al DOM. Un añade un contenedor innecesario que puede romper estilos, semántica o layouts, mientras que <> es invisible en el HTML final.

3- ¿Qué papel tienen index.html y main.jsx en el renderizado?

index.html es la página base que carga el navegador y contiene el dónde React insertará la aplicación.

main.jsx es el punto de entrada de JavaScript: selecciona el elemento root con document.getElementById('root'), crea un root con ReactDOM.createRoot() y renderiza el componente principal () dentro de él.

4- Explica con tus palabras qué significa “componer componentes”.

Componer componentes significa construir la interfaz combinando componentes pequeños dentro de otros más grandes, como si fueran piezas de Lego. Por ejemplo, un componente “Página” puede contener “Header”, “Sidebar” y “Footer”, y cada uno de esos puede tener sus propios subcomponentes. Esto crea un árbol de componentes reutilizable, organizado y fácil de mantener.