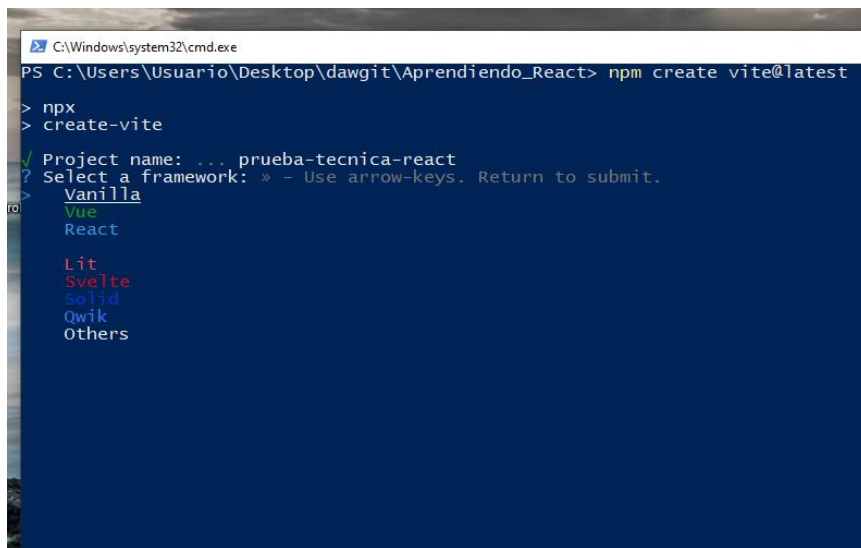


## GUIA PARA INICIAR PROYECTO DE REACT INICIALIZANDO EN VANILLA CREANDO EL PUNTO DE ENTRADA EN LA APP

### 1º Crear el proyecto:

Con el comando ( `npm create vite@latest` ) esto crea el proyecto en la última versión de vite, después de este paso el shell muestra las opciones posibles, para esta guía seleccionaré Vanilla y de este modo configurar manualmente la puerta de entrada a la app.



```
C:\Windows\system32\cmd.exe
PS C:\Users\Usuario\Desktop\dawgit\Aprendiendo_React> npm create vite@latest
> npx
> create-vite
✓ Project name: ... prueba-tecnica-react
? Select a framework: » - Use arrow-keys. Return to submit.
  Vanilla
  Vue
  React
  Lit
  Svelte
  Solid
  Qwik
  Others
```

Una vez nombrado el proyecto y seleccionando Vanilla npm solicita el lenguaje de programación que se desea usar, para esta guía la selección es la de JavaScript.

### 2º INSTALAR PLUGIN REACT

El plugin se instala situado en la carpeta del proyecto:

`npm install @vitejs/plugin-react -E` (el -E otorga la versión exacta).

#### 2.a Configuración del plugin:

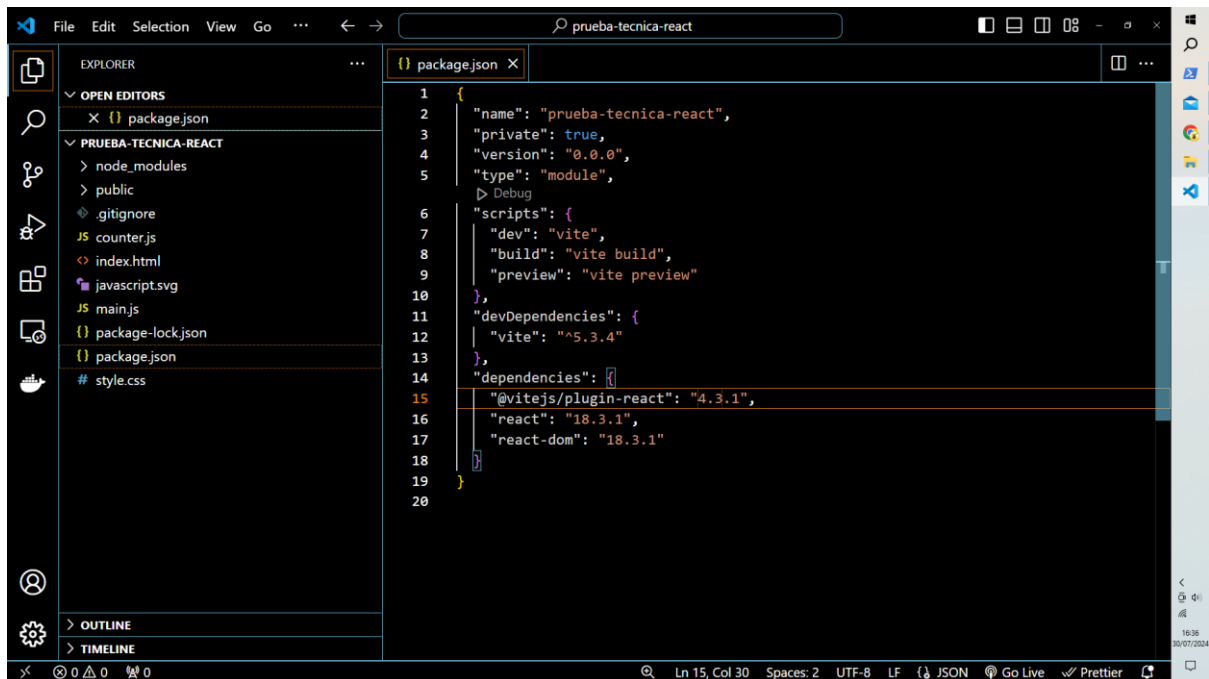
En el proyecto, en el package.json se deben instalar las dependencias necesarias (react y react-dom)

react contiene: la biblioteca de react

react-dom contiene: técnicas y mecanismos que utiliza React para interactuar con el DOM del navegador, incluyendo la renderización de componentes, la gestión de eventos, la manipulación directa del DOM

`npm install react react-dom -E`

Se puede comprobar en el package.json que las dependencias han sido añadidas.



The screenshot shows the Visual Studio Code interface with the 'package.json' file open. The file contains the following JSON structure:

```
1 {
2   "name": "prueba-tecnica-react",
3   "private": true,
4   "version": "0.0.0",
5   "type": "module",
6   "scripts": {
7     "dev": "vite",
8     "build": "vite build",
9     "preview": "vite preview"
10  },
11  "devDependencies": {
12    "vite": "^5.3.4"
13  },
14  "dependencies": {
15    "@vitejs/plugin-react": "4.3.1",
16    "react": "18.3.1",
17    "react-dom": "18.3.1"
18  }
19 }
```

### 3º CONFIGURACION DE VITE

En primer lugar se debe crear el paquete vite.config.js

En el se debe importar el plugin de react, para ello exporto el defineConfig desde vite.



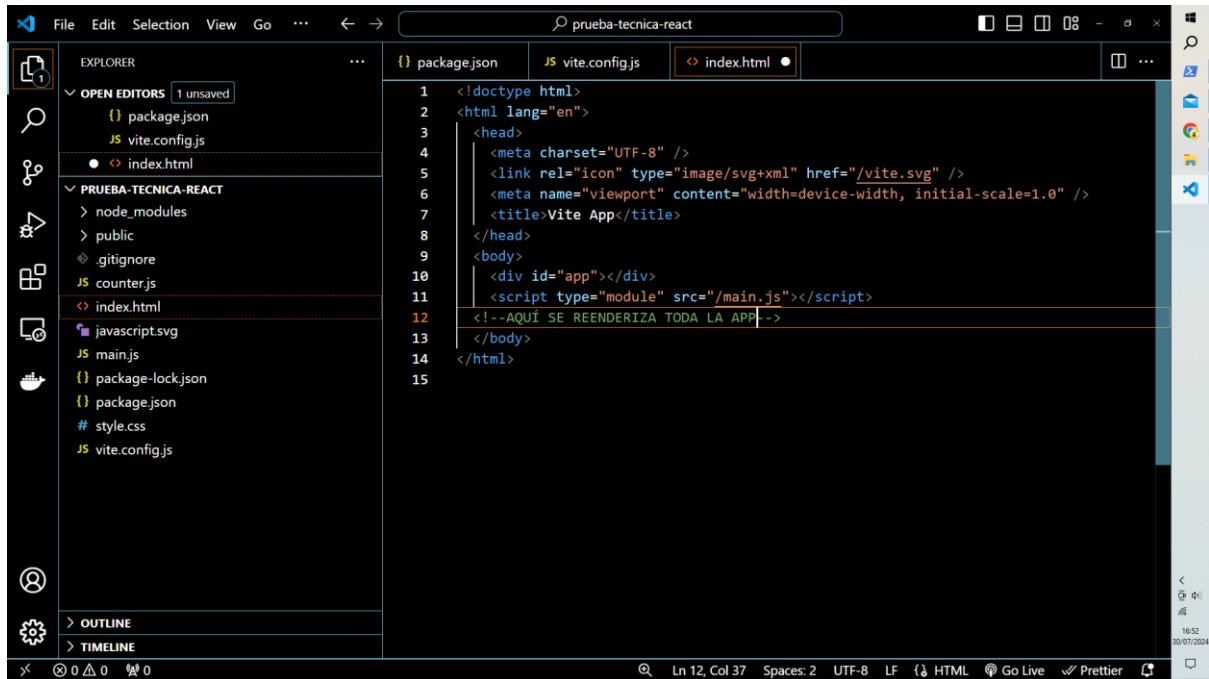
The screenshot shows the Visual Studio Code interface with the 'vite.config.js' file open. The file contains the following JavaScript code:

```
1 import { defineConfig } from 'vite';
2 import react from '@vitejs/plugin-react'
3
4 export default defineConfig(
5   {
6     "plugins": [react()]
7   }
8 )
```

4º PUNTO DE ENTRADA A LA APP.

4.a index.html renderización por main.js

La aplicación se renderiza en el index.html por mediación de un script, concretamente el [main.js](#)



The screenshot shows the Visual Studio Code editor interface. The Explorer panel on the left displays the file structure of the 'prueba-tecnica-react' project. The 'index.html' file is selected and open in the main editor. The code in 'index.html' is as follows:

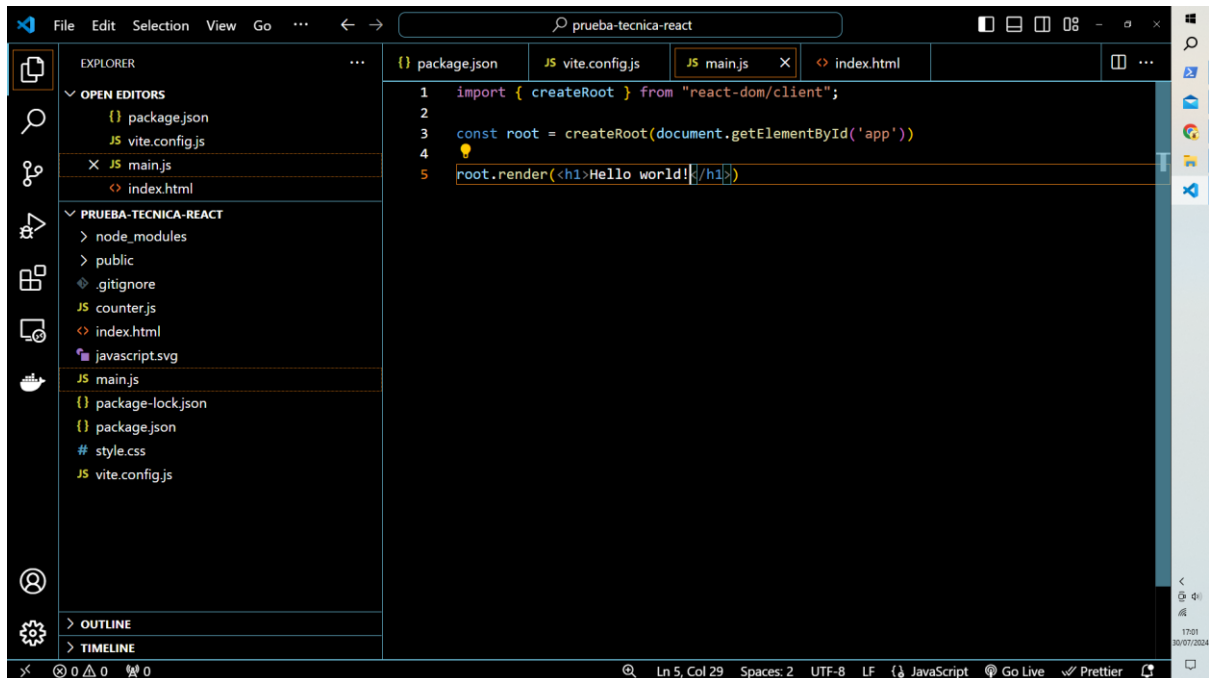
```
1 <!doctype html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8" />
5     <link rel="icon" type="image/svg+xml" href="/vite.svg" />
6     <meta name="viewport" content="width=device-width, initial-scale=1.0" />
7     <title>Vite App</title>
8   </head>
9   <body>
10    <div id="app"></div>
11    <script type="module" src="/main.js"></script>
12    <!-- AQUÍ SE REENDERIZA TODA LA APP -->
13  </body>
14 </html>
15
```

The status bar at the bottom indicates the current position is Line 12, Column 37, with 2 spaces, in UTF-8 encoding, LF line endings, and HTML language mode. It also shows 'Go Live' and 'Prettier' icons.

#### 4.b Configuración del main.js

En el `main.js` se debe importar 'createRoot' desde el react-dom.

Ahora se crea en root y se selecciona el id del div del documento y de esta manera re-renderizando el root actuará sobre el div del index.html



The screenshot shows the Visual Studio Code editor interface. The Explorer sidebar on the left displays the project structure for 'prueba-tecnica-react', including files like package.json, vite.config.js, main.js, and index.html. The main editor window is open to 'main.js', which contains the following JavaScript code:

```
1 import { createRoot } from "react-dom/client";
2
3 const root = createRoot(document.getElementById('app'))
4
5 root.render(<h1>Hello world!</h1>)
```

The status bar at the bottom indicates the current position is Line 5, Column 29, with 2 spaces, UTF-8 encoding, and LF line endings. It also shows the active language as JavaScript and the Prettier formatter.

!!! Para que funcione es necesario cambiar la extensión del main.js a main.jsx tanto en el archivo como en el source de la etiqueta script del index.html, esto se debe a que el plugin de vite necesita .jsx para poder transpilar el código