

## Actividades para activar React.js y Node.js

Instalación y configuración del entorno de desarrollo para un proyecto React con Node.js y SQL Server en tu máquina. Nos enfocamos en:

1. **Instalación de Node.js y npm:** Instalamos **Node.js** y **npm** en tu máquina para gestionar dependencias y ejecutar aplicaciones JavaScript.
2. **Creación de un nuevo proyecto React:** Después de algunos problemas con las rutas largas y los espacios en los nombres de las carpetas, finalmente logramos crear un proyecto React en una carpeta con un nombre más simple.
3. **Resolución de errores de módulos:** A lo largo del proceso, eliminamos dependencias innecesarias y reinstalamos los módulos para asegurarnos de que todo estaba configurado correctamente.
4. **Ejecución exitosa de React:** Al final, ejecutamos correctamente el servidor de desarrollo de React y verificamos que la aplicación se estaba ejecutando en el navegador.

---

## Pasos para replicar este proceso:

### 1. Instalar Node.js y npm

Si no lo has hecho aún en una nueva máquina, puedes descargar Node.js desde su sitio oficial [Node.js](https://nodejs.org/). Instala la versión LTS (Long-Term Support) y asegúrate de que también se instala **npm** automáticamente.

- Para verificar la instalación:

```
node -v
```

```
npm -v
```

Esto debería mostrarte las versiones instaladas de **Node.js** y **npm**.

### 2. Crear un nuevo proyecto React:

1. Abre la terminal y navega al directorio donde quieras crear el proyecto. Por ejemplo:

```
cd "C:\Proyectos\ReactJS SQLServ y NodeJS"
```

En mi caso tengo una carpeta llamada Proyectos y estoy guardado todo lo concerniente a este proyecto en la carpeta ReactJS SQLServ y NodeJS.

2. Usa **npm** para crear un proyecto React. Esto descargará automáticamente las dependencias necesarias:

```
npm create-react-app idprojectgaschsoft
```

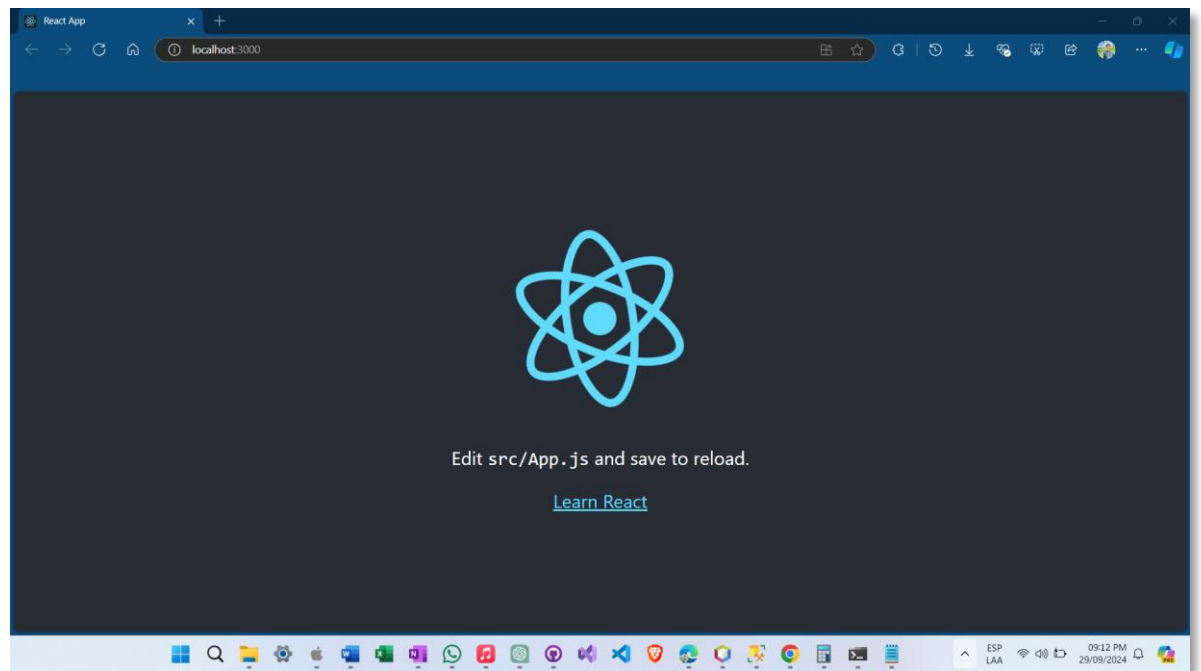
3. Una vez que el proyecto esté creado, entra en la carpeta del proyecto:

```
cd idprojectgaschsoft
```

4. Inicia el servidor de desarrollo para ejecutar la aplicación React:

```
npm start
```

La aplicación se abrirá en el navegador en <http://localhost:3000>. (en este punto pueda que te pida algún permiso de administrador para que se pueda abrir el navegador)



---

## Manejo de errores comunes y soluciones alternativas:

### Errores con rutas largas o caracteres especiales:

- **Error:** No se podía acceder a una carpeta con un nombre largo o caracteres especiales, como en el caso de "3. React JS, SQL Server & Nodejs".
- **Solución:** Renombramos la carpeta a algo más corto y sin caracteres especiales, como **ReactJS SQLServ y NodeJS**.

```
ren "C:\Proyectos\3. React JS, SQL Server & Nodejs"  
"C:\Proyectos\ReactJS SQLServ y NodeJS"
```

Luego navegamos a la nueva carpeta:

```
cd "C:\Proyectos\ReactJS SQLServ y NodeJS\idprojectgaschsoft"
```

### Errores con dependencias o módulos:

- **Error:** Mensajes como "Cannot find module react-scripts" o problemas al iniciar el servidor de desarrollo.
  - **Solución:** Si encuentras problemas con las dependencias o módulos no encontrados, sigue estos pasos:
    1. **Eliminar la carpeta node\_modules:**

```
npm install -g rimraf  
rimraf node_modules
```
    2. **Eliminar el archivo package-lock.json:**

```
del package-lock.json
```
    3. **Reinstalar las dependencias:**

```
npm install
```
    4. **Iniciar el servidor de desarrollo nuevamente:**

```
npm start
```
- 

### Conclusión:

- **Instalamos Node.js y npm.**
- **Creamos un proyecto React** con éxito después de resolver errores relacionados con rutas largas y dependencias.
- **Iniciamos el servidor de desarrollo**, y verificamos que la aplicación se ejecutó correctamente en el navegador.

En mi caso, me encontré con inconvenientes como crear el proyecto dentro de una carpeta con caracteres especiales (por ejemplo, "3. React JS, SQL Server & Nodejs"). Después, lo moví a una carpeta más simple, directamente en el disco local C:, lo que facilitó el proceso y evitó errores.