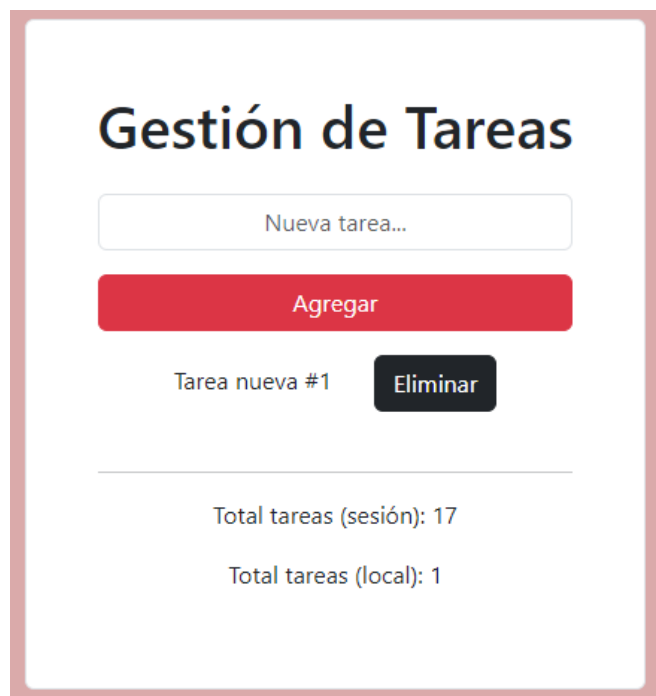


## Evidencia de DESAFIO #3: Realizado con REACT



The screenshot shows a web application titled "Gestión de Tareas". It features a text input field with the placeholder "Nueva tarea..." and a red "Agregar" button below it. A horizontal line separates the input area from the summary area, which displays "Total tareas (sesión): 16" and "Total tareas (local): 0".

Diseño de Gestor de tareas



The screenshot shows the same "Gestión de Tareas" application after a new task has been added. The text input field now contains "Nueva tarea...". The red "Agregar" button is still present. Below it, the text "Tarea nueva #1" is displayed next to a dark grey "Eliminar" button. A horizontal line separates this area from the summary area, which now displays "Total tareas (sesión): 17" and "Total tareas (local): 1".

Agregado de tareas al Gestor de tareas

# Gestión de Tareas

Agregar

Tarea nueva #1

Eliminar

Tarea #2

Eliminar

Tarea adquirida #3

Eliminar

---

Total tareas (sesión): 19

Total tareas (local): 3

Ingresando varias tareas en el gestor de tareas.

# Gestión de Tareas

Agregar

Tarea adquirida #3

Eliminar

---

Total tareas (sesión): 19

Total tareas (local): 1

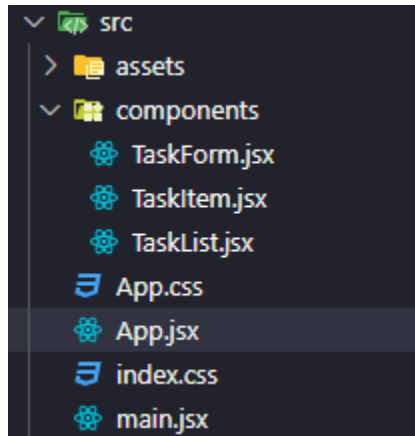
Eliminando tareas (Se eliminaron las primeras 2 tareas).

---

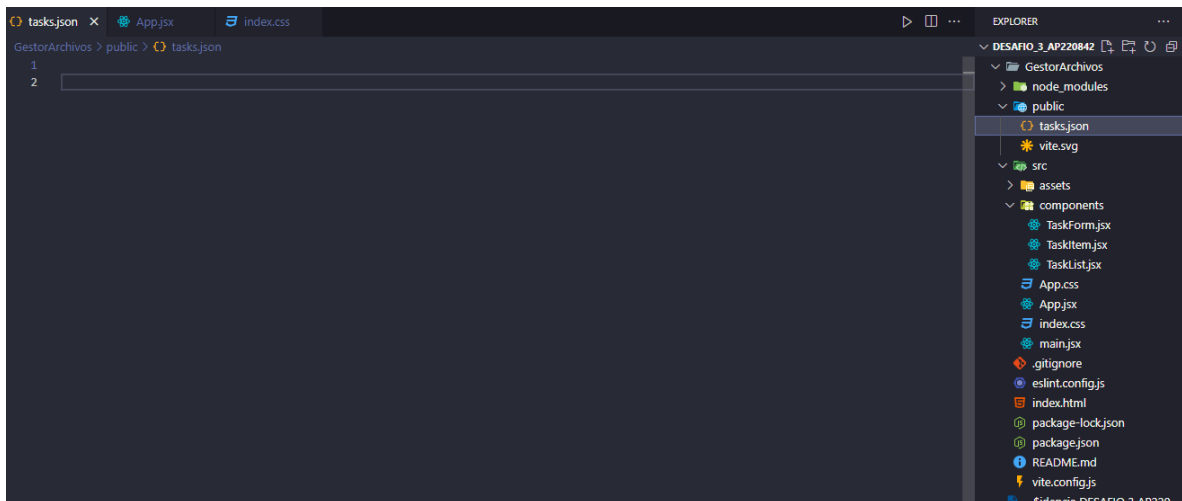
Total tareas (sesión): 19

Total tareas (local): 1

Parte inferior que muestra cuantas tareas hay en el LocalStorage y cuantas se han ingresado en el SessionStorage (Aparecen 19, ya que como prueba se han hecho ingreso y borrado de 19 tareas)



Carpeta SRC que contiene a la carpeta con los componentes usados con REACT y los archivos principales App.jsx y main.jsx.



Archivo “Tasks.json” que contiene la simulación de agregación y eliminación de tareas cuando se ejecuta el Fetch.

## CODIGO DESCRITO

```
tasks.json TaskForm.jsx 1 App.jsx index.css
GestorArchivos > src > components > TaskForm.jsx > TaskForm
1 import { useState } from "react";
2 import "bootstrap/dist/css/bootstrap.min.css";
3
4 const TaskForm = ({ addTask }) => { 'addTask' is missing in props validation
5   const [taskText, setTaskText] = useState("");
6
7   const handleSubmit = (e) => {
8     e.preventDefault();
9     if (taskText.trim()) {
10       addTask({ text: taskText, completed: false });
11       setTaskText("");
12     }
13   };
14
15   return (
16     <form onSubmit={handleSubmit} className="d-flex flex-column justify-content-center">
17       <input
18         className="form-control text-center mt-3"
19         type="text"
20         value={taskText}
21         onChange={(e) => setTaskText(e.target.value)}
22         placeholder="Nueva tarea..." />
23       <button type="submit" className="btn btn-danger mt-3">Agregar</button>
24     </form>
25   );
26 };
```

### Componente TaskForm.jsx

```
File Edit Selection View Go Run ... DESAFIO_3_AP220842
tasks.json TaskItem.jsx 3 App.jsx index.css
GestorArchivos > src > components > TaskItem.jsx > default
1 import "bootstrap/dist/css/bootstrap.min.css";
2
3 const TaskItem = ({ task, deleteTask }) => { 'task' is missing in props validation
4   return (
5     <li className="mt-3 text-center">
6       {task.text} <button className="btn btn-dark ms-4" onClick={deleteTask}>Eliminar</button>
7     </li>
8   );
9 };
10
11 export default TaskItem;
12
```

### Componente TaskItem.jsx

```
tasks.json x TaskList.jsx 3 x App.jsx index.css
GestorArchivos > src > components > TaskList.jsx > ...
1 import TaskItem from "../TaskItem";
2 import "bootstrap/dist/css/bootstrap.min.css";
3
4 const TaskList = ({ tasks, deleteTask }) => { 'tasks' is missing in props validation
5   return (
6     <ul className="list-unstyled">
7       {tasks.map((task, index) => ( 'tasks.map' is missing in props validation
8         <TaskItem key={index} task={task} deleteTask={() => deleteTask(index)} />
9       ))}
10    </ul>
11  );
12 };
13
14 export default TaskList;
15
```

## Componente TaskList.jsx

```
tasks.json TaskList.jsx 3 App.jsx x index.css
GestorArchivos > src > App.jsx > App
1 import { useEffect, useState } from "react";
2 import TaskForm from "../components/TaskForm";
3 import TaskList from "../components/TaskList";
4 import "bootstrap/dist/css/bootstrap.min.css";
5
6 const App = () => {
7   const [tasks, setTasks] = useState([]);
8   const [sessionTaskCount, setSessionTaskCount] = useState(0);
9
10   // Carga de las tareas
11   useEffect(() => {
12     const savedTasks = JSON.parse(localStorage.getItem("tasks")) || [];
13     setTasks(savedTasks);
14
15     // Uso de AJAX para cargar tareas
16     fetch("/tasks.json")
17       .then((response) => response.json())
18       .then((data) => {
19         setTasks((prevTasks) => [...prevTasks, ...data.tasks]);
20       });
21
22     // cONTADOR DE TAREAS EN SESIÓN
23     const sessionCount = sessionStorage.getItem("taskCount") || 0;
24     setSessionTaskCount(parseInt(sessionCount));
25   }, []);
26
27   const addTask = (task) => {
```

## Archivo principal de renderizado REACT

## **INDICACIONES DE EJECUCION**

- 1) Clonar el repositorio con “git clone” desde una terminal o powershell en Window (LINK EN ARCHIVO README DEL REPOSITORIO)
- 2) Una vez clonado, entrar a la carpeta con “cd GestorArchivos”
- 3) Una vez en la carpeta, instalar las dependencias de React con “npm install”
- 4) Terminada la instalación de dependencia, ejecutar comando “npm run dev” para abrirlo con un URL de navegador del entorno de trabajo de VITE