

**Christian Dominic P. Tabelon**

## **PART 1:**

### **Component in React:**

A component is a self-contained piece of code that represents a UI element or a part of a UI element. The Components in React is where you break down big applications features into smaller parts like the navbar or the sidebar in a big component that is the app components in the file. This will make designing the app easier

### **JSX and Component**

Jsx file is a component in react that compiles faster and make the file size smaller the code is also optimized are all merged in one file to run. The errors can also be seen in the console in the browsers inspect console.

### **Props**

They are used to pass data from a parent component to a child component, this allows us to reuse a code like a Header or Footer that can be in different files. This enables the child component to customize its behavior based on the received props.

### **States**

States is a component where it enables you to design interactions that can respond to the user. The state is where you find the state object is keep component-specific property values stored.

### **Contrast state and Props**

The props are a method for passing data or characteristics from one component to another, while the state is the real-time data available for use within that single component.

### **SPA**

Single-Page Applications (SPAs) offer a consistent user experience and can boost performance by lowering the number of HTTP queries, as they only require a single initial load of the application. This results in faster navigation and a more seamless user experience.

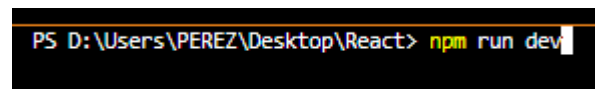
## CSR

Client-Side Rendering (CSR) is a useful approach for boosting React web application performance. By delegating rendering to the client-side, CSR reduces the load on the server, resulting in faster page loads and improved overall application responsiveness.

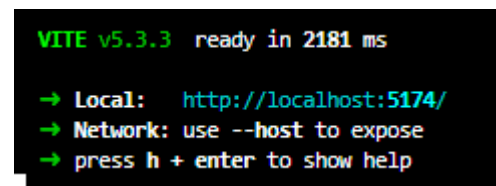
## Part 2:

## SCREENSHOTS

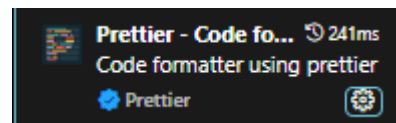
Run the code using npm run dev



```
PS D:\Users\PEREZ\Desktop\React> npm run dev
```



```
VITE v5.3.3 ready in 2181 ms
→ Local:   http://localhost:5174/
→ Network: use --host to expose
→ press h + enter to show help
```



```
Prettier - Code fo... 241ms
Code formatter using prettier
Prettier
```

In the Task list we can see a number of people with their task descriptions and with their names

### Task List

Tasks					
<input type="text" value="Search tasks"/>					
No.	Task Description	Assignee	Deadline	Status	Actions
1	Cleaning	Josh	2023-03-15		<button>Toggle Status</button>
2	Cooking	Dale	2023-03-20		<button>Toggle Status</button>
3	Writing	Leigh	2023-03-12		<button>Toggle Status</button>
4	Coding	Patrice	2023-03-18		<button>Toggle Status</button>
5	Planning	Jeff	2023-03-22		<button>Toggle Status</button>
6	Organizing	Jethan	2023-03-25		<button>Toggle Status</button>
7	Researching	Josh	2023-03-25		<button>Toggle Status</button>
8	Coding	Jacob	2023-03-25		<button>Toggle Status</button>

The Search filter will only search what task is the person doing and cannot search the person.

### Task List

Tasks					
<input type="text" value="Josh"/>					
No.	Task Description	Assignee	Deadline	Status	Actions

The search bar can only search the task given

### Task List

Tasks					
<input type="text" value="Cleaning"/>					
No.	Task Description	Assignee	Deadline	Status	Actions
1	Cleaning	Josh	2023-03-15	Completed	<button>Toggle Status</button>
2	Cleaning	Leigh	2023-03-25		<button>Toggle Status</button>

Toggle the status to mark the assigned individual as pending work and will show when is the deadline

## Task List

Tasks					
<input type="text" value="Search tasks"/>					
No.	Task Description	Assignee	Deadline	Status	Actions
1	Cleaning	Josh	2023-03-15	Pending	<button>Toggle Status</button>
2	Cooking	Dale	2023-03-20	Pending	<button>Toggle Status</button>
3	Writing	Leigh	2023-03-12	Pending	<button>Toggle Status</button>
4	Coding	Patrice	2023-03-18	Completed	<button>Toggle Status</button>
5	Planning	Jeff	2023-03-22	Completed	<button>Toggle Status</button>
6	Organizing	Jethan	2023-03-25	Pending	<button>Toggle Status</button>
7	Researching	Josh	2023-03-25	Pending	<button>Toggle Status</button>
8	Coding	Jacob	2023-03-25		<button>Toggle Status</button>

Toggle the status button again and the assignee will be marked as completed

Task List

Tasks

Search tasks

No.	Task Description	Assignee	Deadline	Status	Actions
1	Cleaning	Josh	2023-03-15	Completed	<div>Toggle Status</div>
2	Cooking	Dale	2023-03-20	Completed	<div>Toggle Status</div>
3	Writing	Leigh	2023-03-12	Completed	<div>Toggle Status</div>
4	Coding	Patrice	2023-03-18	Completed	<div>Toggle Status</div>
5	Planning	Jeff	2023-03-22	Completed	<div>Toggle Status</div>
6	Organizing	Jethan	2023-03-25	Completed	<div>Toggle Status</div>
7	Researching	Josh	2023-03-25	Completed	<div>Toggle Status</div>
8	Coding	Jacob	2023-03-25		<div>Toggle Status</div>