

# Smart To-Do List: A React Hooks PoC App

Kaisser Thomas

June 5, 2025

## Project Overview

**Smart To-Do List** is a lightweight, single-page React application that demonstrates how to use core **React Hooks** to build an interactive, functional task management app. It includes features such as filtering, theming, and task persistence using `localStorage`.

This project serves as a Proof of Concept (PoC) illustrating how to use a modern React-based application framework with hooks, fulfilling the assessment requirement for a framework PoC (option D).

## Technologies Used

- React (via Vite)
- JavaScript (ES6)
- CSS (custom, no framework)
- React Hooks: `useState`, `useEffect`, `useContext`, `useRef`

## Core Features and Hooks

Feature	Hook Used	Purpose
Add/Delete Tasks	<code>useState</code>	Manage list of tasks in component state.
Task Completion Toggle	<code>useState</code>	Toggle completed status and UI updates.
Filter Tasks	<code>useState</code>	Switch between All, Active, Completed views.
Persist to LocalStorage	<code>useEffect</code>	Load and save tasks from/to <code>localStorage</code> .
Autofocus Input	<code>useRef</code>	Automatically focus the input on component mount.
Theme Toggle (Dark/Light)	<code>useContext</code>	Use global context to manage theme state.

Table 1: Features and Associated Hooks

## Example Code Snippets

### useState for Tasks

```
const [tasks, setTasks] = useState([]);
```

### useEffect for Local Storage

```
useEffect(() => {  
  localStorage.setItem('tasks', JSON.stringify(tasks));  
}, [tasks]);
```

### useContext for Theme

```
const { darkMode, toggleTheme } = useTheme();
```

### useRef for Input Focus

```
const inputRef = useRef();  
  
useEffect(() => {  
  inputRef.current.focus();  
}, []);
```

## How to Run the App

1. Clone the repository.
2. Install dependencies:

```
npm install
```

3. Start the development server:

```
npm run dev
```

## Conclusion

This project demonstrates the core capabilities of React Hooks and their role in building modular, modern frontend apps. The Smart To-Do List is simple, clean, and effectively highlights stateful logic and interaction patterns without any third-party dependencies.