

W06-P1: Use DevTools to exam useState values

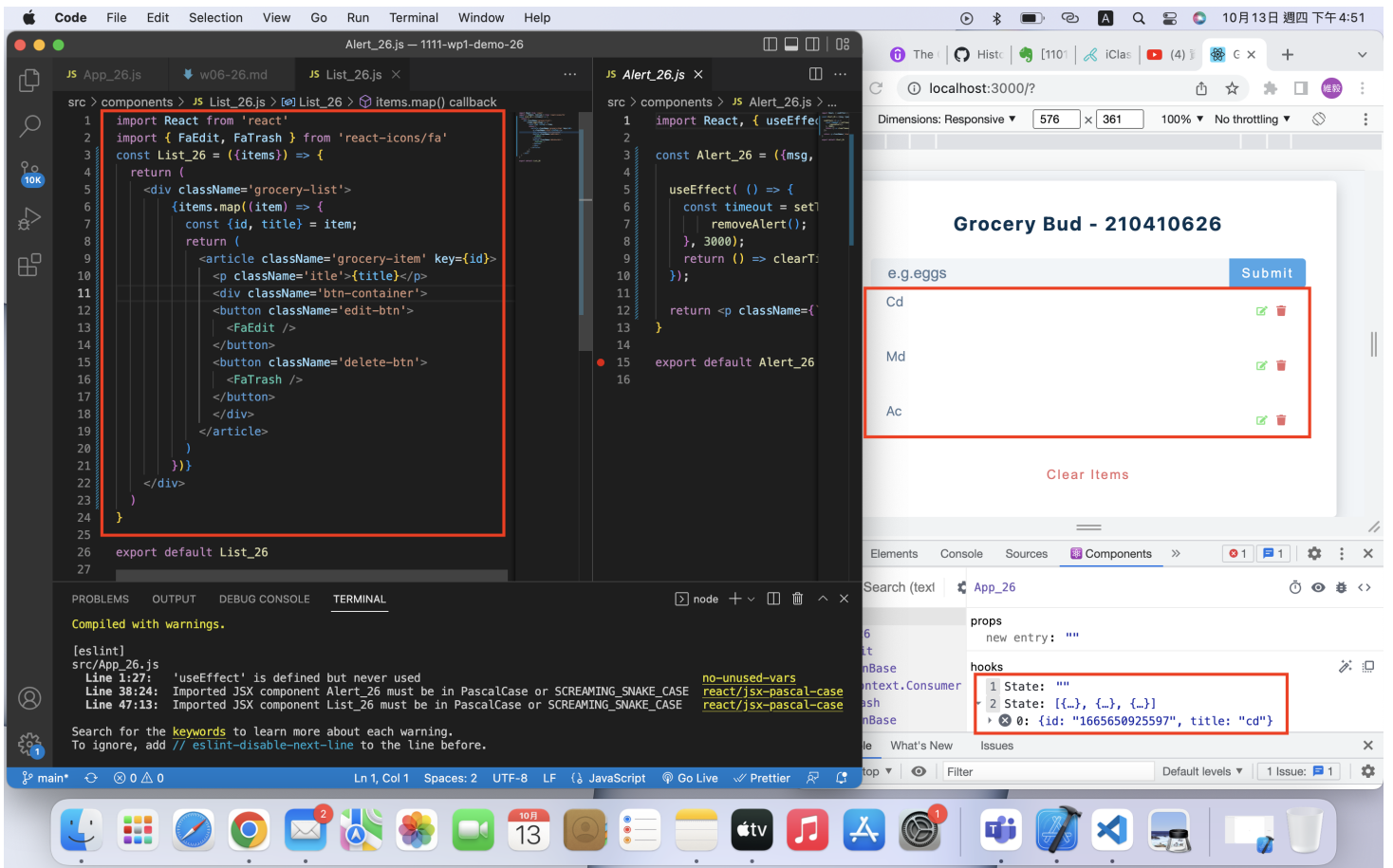
The screenshot displays a web application running in a browser at localhost:3000. The application is titled "Grocery Bud - 210410626" and features a form with an input field containing "e.g.eggs" and a "Submit" button. The background shows the source code of the application in a dark-themed editor. The code defines a component `App_26` using `useState` to manage state. Three sections of the code are highlighted with red boxes:

- The initial state definition for `name`, `list`, and `alert`.
- The `showAlert` function logic.
- The `handleSubmit` function logic, specifically the state update part.

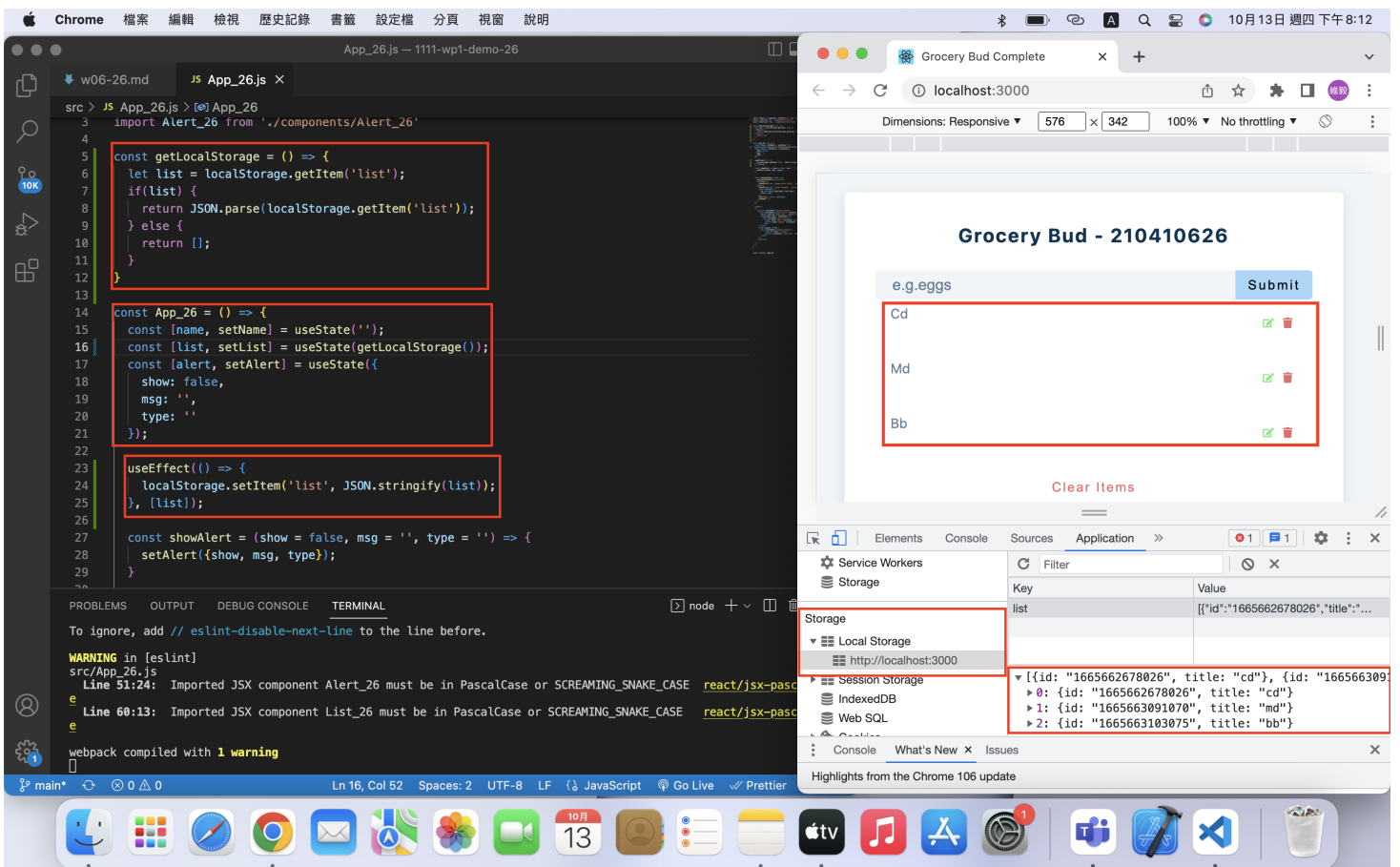
The right sidebar shows the React DevTools component inspector for `App_26`. The "hooks" tab is selected, displaying the state changes for `useState`. The current state is highlighted in red:

```
1 State: ""
2 State: [{...}, {...}, {...}]
3 State: {id: "1665639228805", title: "cd"}
4 State: {id: "1665639461951", title: "me"}
5 State: {id: "1665639467944", title: "yi"}
6 State: {msg: "value changed", show: true, type: "success"}
```

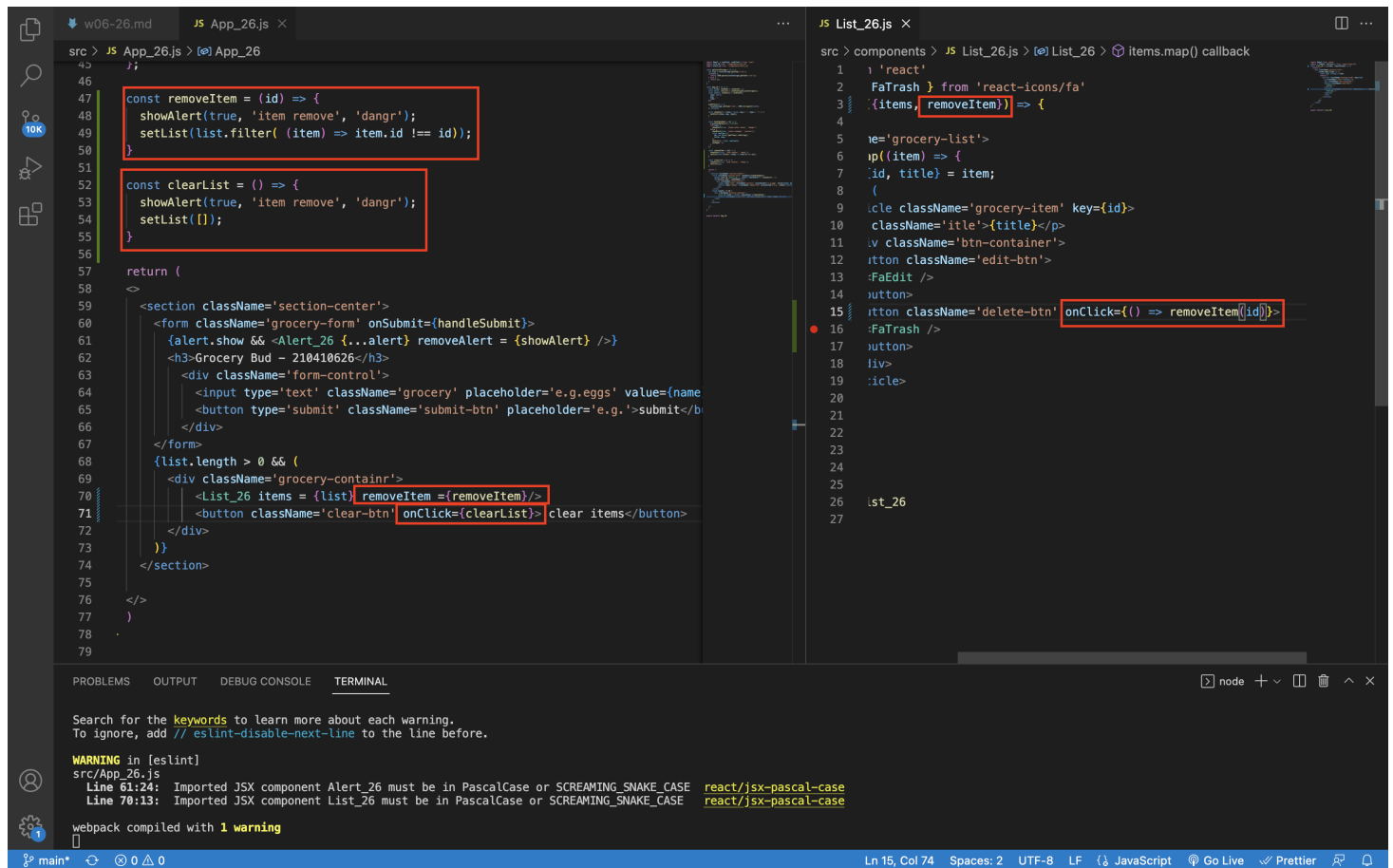
W06-P2: Show <Alert_xx /> and <List_xx />



W06-P3: Using localStorage to save list data



W06-P4: Add function for removing item and clearing list



```
src > JS App_26.js > [App_26]
46
47 const removeItem = (id) => {
48   showAlert(true, 'item remove', 'dangr');
49   setList(list.filter( (item) => item.id !== id));
50 }
51
52 const clearList = () => {
53   showAlert(true, 'item remove', 'dangr');
54   setList([]);
55 }
56
57 return (
58   <section className='section-center'>
59     <form className='grocery-form' onSubmit={handleSubmit}>
60       {alert.show && <Alert_26 {...alert} removeAlert = {showAlert} />}
61       <h3>Grocery Bud - 210410626</h3>
62       <div className='form-control'>
63         <input type='text' className='grocery' placeholder='e.g.eggs' value={name} />
64         <button type='submit' className='submit-btn' placeholder='e.g.'>submit</button>
65       </div>
66     </form>
67     {list.length > 0 && (
68       <div className='grocery-container'>
69         <List_26 items = {list} removeItem={removeItem}/>
70         <button className='clear-btn' onClick={clearList}> clear items</button>
71       </div>
72     )}
73   </section>
74 )
75
76
77
78
79
```

```
src > components > JS List_26.js > [List_26] > items.map() callback
1  'react'
2  FaTrash } from 'react-icons/fa'
3  {items, removeItem} => {
4
5  ne='grocery-list'>
6  ip((item) => {
7    id, title) = item;
8
9    <div className='grocery-item' key={id}>
10     <div className='title'>{title}</div>
11     <div className='btn-container'>
12       <button className='edit-btn'>
13         <FaEdit />
14       </button>
15       <button className='delete-btn' onClick={() => removeItem(id)}>
16         <FaTrash />
17       </button>
18     </div>
19   </div>
20
21
22
23
24
25
26   list_26
27
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Search for the keywords to learn more about each warning.
To ignore, add // eslint-disable-next-line to the line before.

WARNING in [eslint]
src/App_26.js
Line 61:24: Imported JSX component Alert_26 must be in PascalCase or SCREAMING_SNAKE_CASE react/jsx-pascal-case
Line 70:13: Imported JSX component List_26 must be in PascalCase or SCREAMING_SNAKE_CASE react/jsx-pascal-case

webpack compiled with 1 warning

W06-logs: show all four logs done today

```
branch 'main' set up to track 'origin/main'.
(base) huangweiyi@huangweiyideMacBook-Pro 1111-wp1-demo-26 % git log --pretty=format:"%h%x09%an%x09%ad%x09%s" --after="2022-10-04"
11af1bc WeiYiHuangfelix Thu Oct 13 20:37:52 2022 +0800 W06-P4: Add function for removing item and clearing list
a6f5168 WeiYiHuangfelix Thu Oct 13 20:15:55 2022 +0800 W06-P3: Using localStorage to save list data
bb61eb7 WeiYiHuangfelix Thu Oct 13 17:30:26 2022 +0800 W06-P2: Show <Alert_xx /> and <List_xx />
534cf14 WeiYiHuangfelix Thu Oct 13 13:47:39 2022 +0800 W06-P1: Use DevTools to exam useState values
101220f WeiYiHuangfelix Thu Oct 6 20:34:26 2022 +0800 W05-P4: T31_xx: conditional rendering -- 3 cases
5fb75ad WeiYiHuangfelix Thu Oct 6 17:25:57 2022 +0800 W05-P2: useEffect 3 cases demo
4183475 WeiYiHuangfelix Thu Oct 6 16:32:28 2022 +0800 W05-P3: T23_xx: fetch data from Github and display them
041d79b WeiYiHuangfelix Thu Oct 6 13:27:11 2022 +0800 W05-P1: Hooks Tutorial T14_xx
(base) huangweiyi@huangweiyideMacBook-Pro 1111-wp1-demo-26 %
```

```
(base) huangweiyi@huangweiyideMacBook-Pro 1111-wp1-demo-26 % git log --pretty=format:"%h%x09%an%x09%ad%x09%s" --after="2022-10-04"
11af1bc WeiYiHuangfelix Thu Oct 13 20:37:52 2022 +0800 W06-P4: Add function for removing item and clearing list
a6f5168 WeiYiHuangfelix Thu Oct 13 20:15:55 2022 +0800 W06-P3: Using localStorage to save list data
bb61eb7 WeiYiHuangfelix Thu Oct 13 17:30:26 2022 +0800 W06-P2: Show <Alert_xx /> and <List_xx />
534cf14 WeiYiHuangfelix Thu Oct 13 13:47:39 2022 +0800 W06-P1: Use DevTools to exam useState values
```