## **Objectives**

* **Explain about React forms**

In React, forms are used to collect user input. React provides a way to create and manage form elements using state. Unlike traditional HTML where form data is handled by the DOM, React forms are often controlled by the component's state.

React forms allow you to:

* Capture input using various elements (input, textarea, select, etc.)
* Maintain form state using useState or other state management
* Handle user input changes with event handlers like onChange
* Validate and submit data programmatically
* **Define controlled components**

A controlled component in React is a form element (like <input>, <textarea>, or <select>) whose value is controlled by the React component state. It means that the component maintains full control over the input's value.

**Example:**

const [name, setName] = useState('');

<input type="text" value={name} onChange={(e) => setName(e.target.value)} />

In this example:

* value={name} makes the input controlled.
* setName updates the state on every change.

Controlled components ensure that React is the single source of truth for form data.

* **Explain about various input controls**

React supports all standard HTML input types:

|  |  |  |
| --- | --- | --- |
| Input Type | Text Element Used | Description |
| Text Input | <input type="text" /> | For single-line text |
| Textarea | <textarea /> | For multi-line text |
| Checkbox | <input type="checkbox" /> | For boolean values |
| Radio Button | <input type="radio" /> | For selecting one option from many |
| Select/Dropdown | <select> and <option> | For dropdown lists |
| File Input | <input type="file" /> | For file uploads |

Each input type uses onChange to capture changes and update the state accordingly.

* **Explain about handling forms**

Form handling in React typically involves:

1. **Setting initial state** using useState for form fields.
2. **Capturing user input** via onChange events.
3. **Updating state** based on user input.
4. **Using state values** to control input fields (controlled components).

* **Explain about submitting forms**

In React, form submission is handled by defining an onSubmit function and attaching it to a <form> element.

Steps:

1. Prevent default browser behavior using event.preventDefault().
2. Access and validate input values from state.
3. Send data to a server or process it.

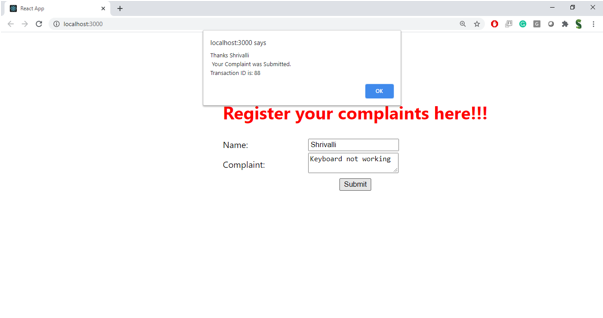
## **Notes**

Estimated time to complete this lab: **60 minutes.**

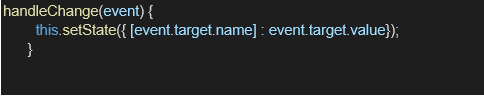
Create a React App named “ticketraisingapp” which will help to raise a complaint and get it resolved.

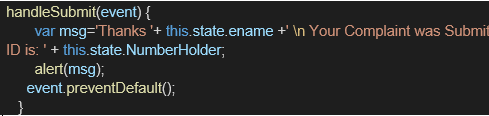
Create a component named “ComplaintRegister” with a form containing a textbox to enter the employee name and a textarea to enter the complaint. Use “handleSubmit” event of the button to submit the complaint and generate a Reference number for further follow ups in the alert box.

**Output Expected:**

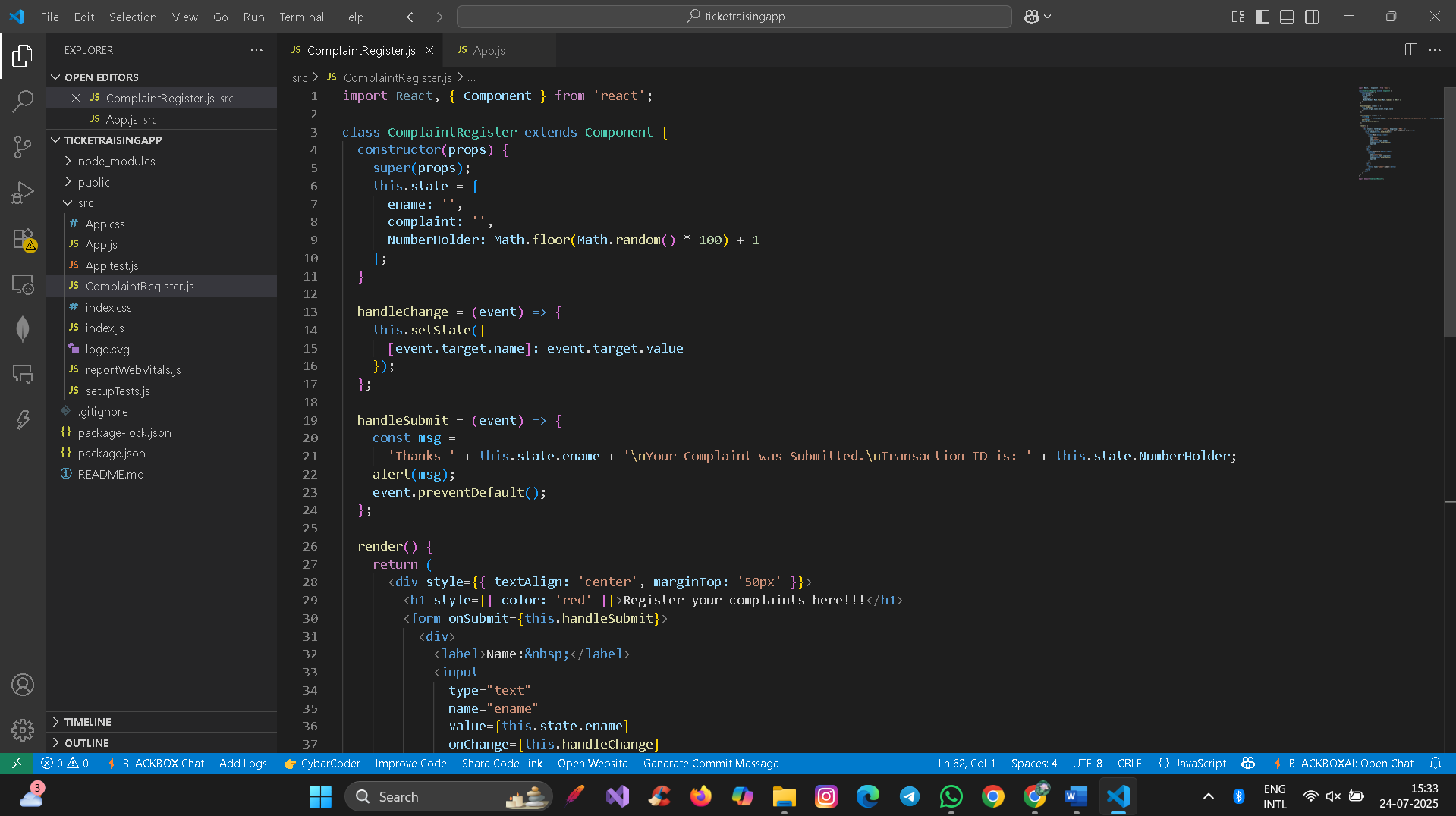
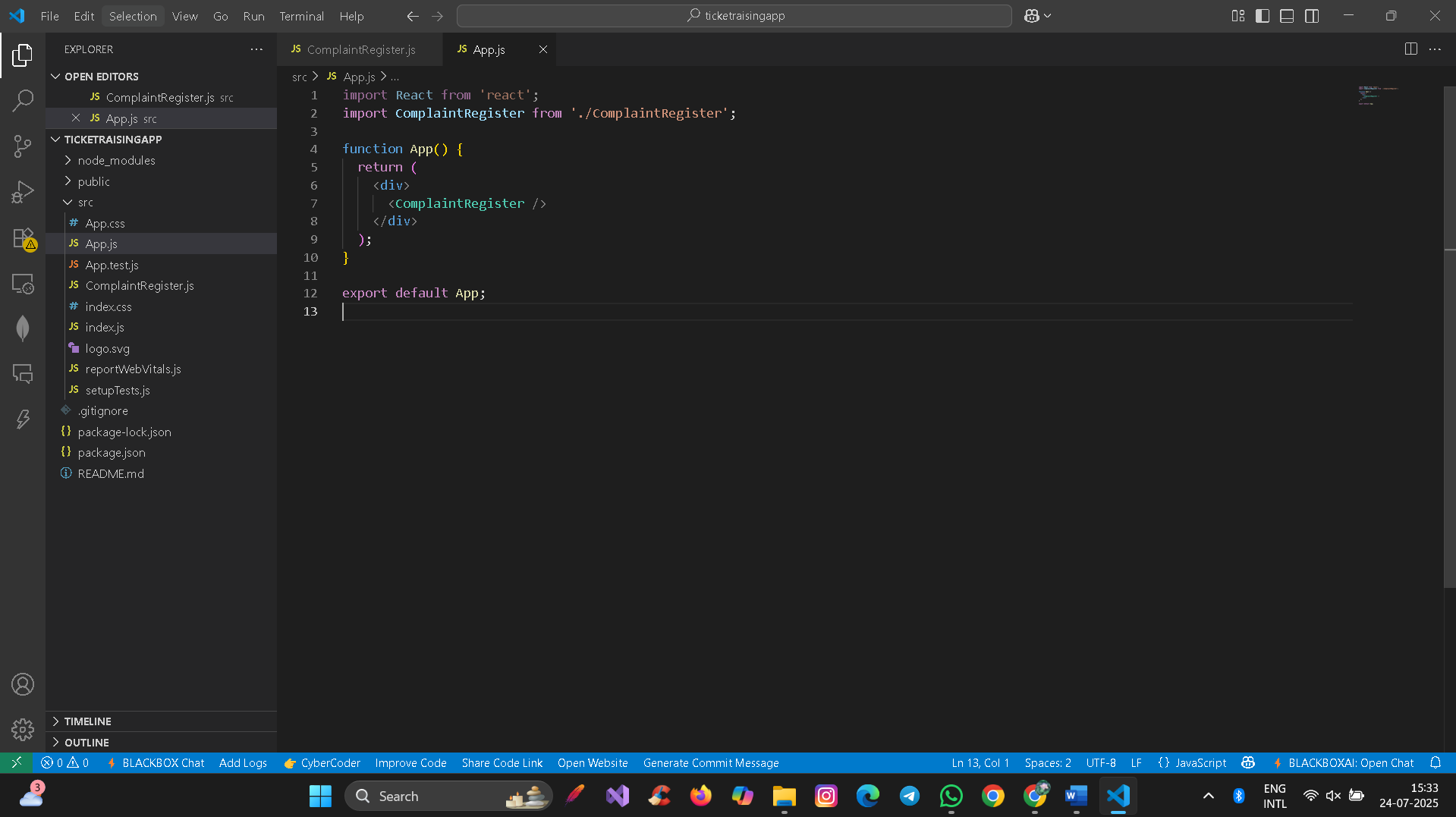
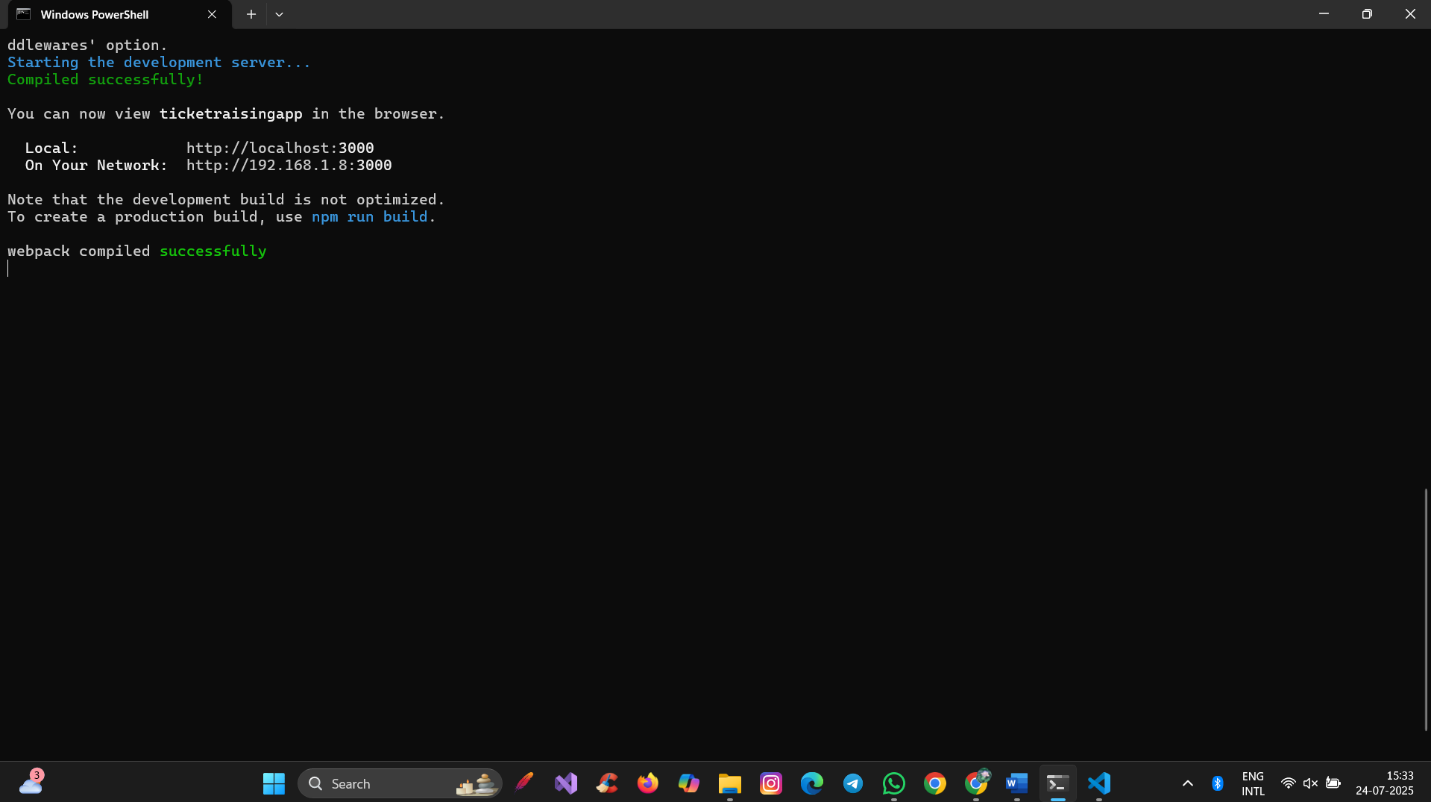
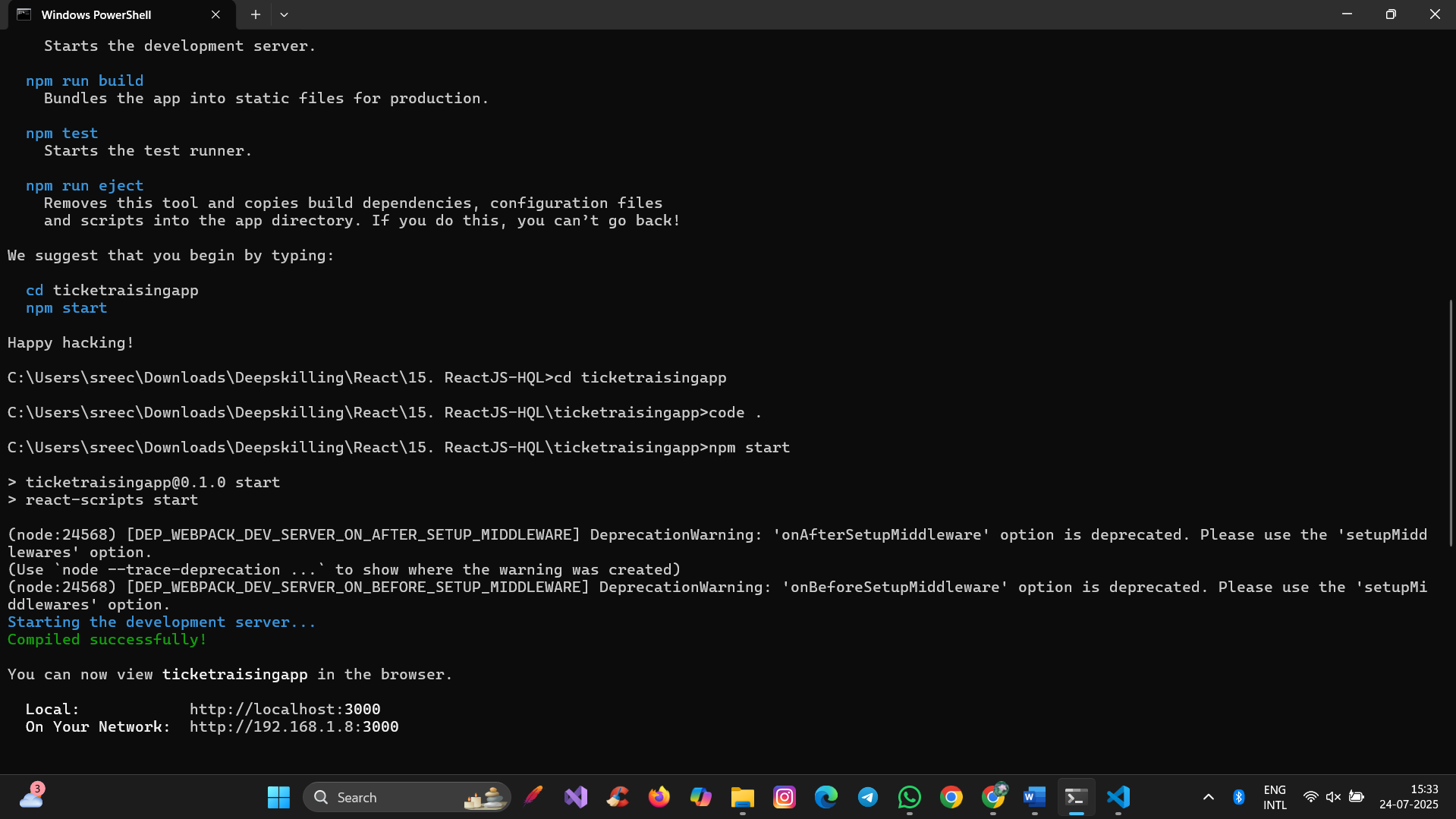
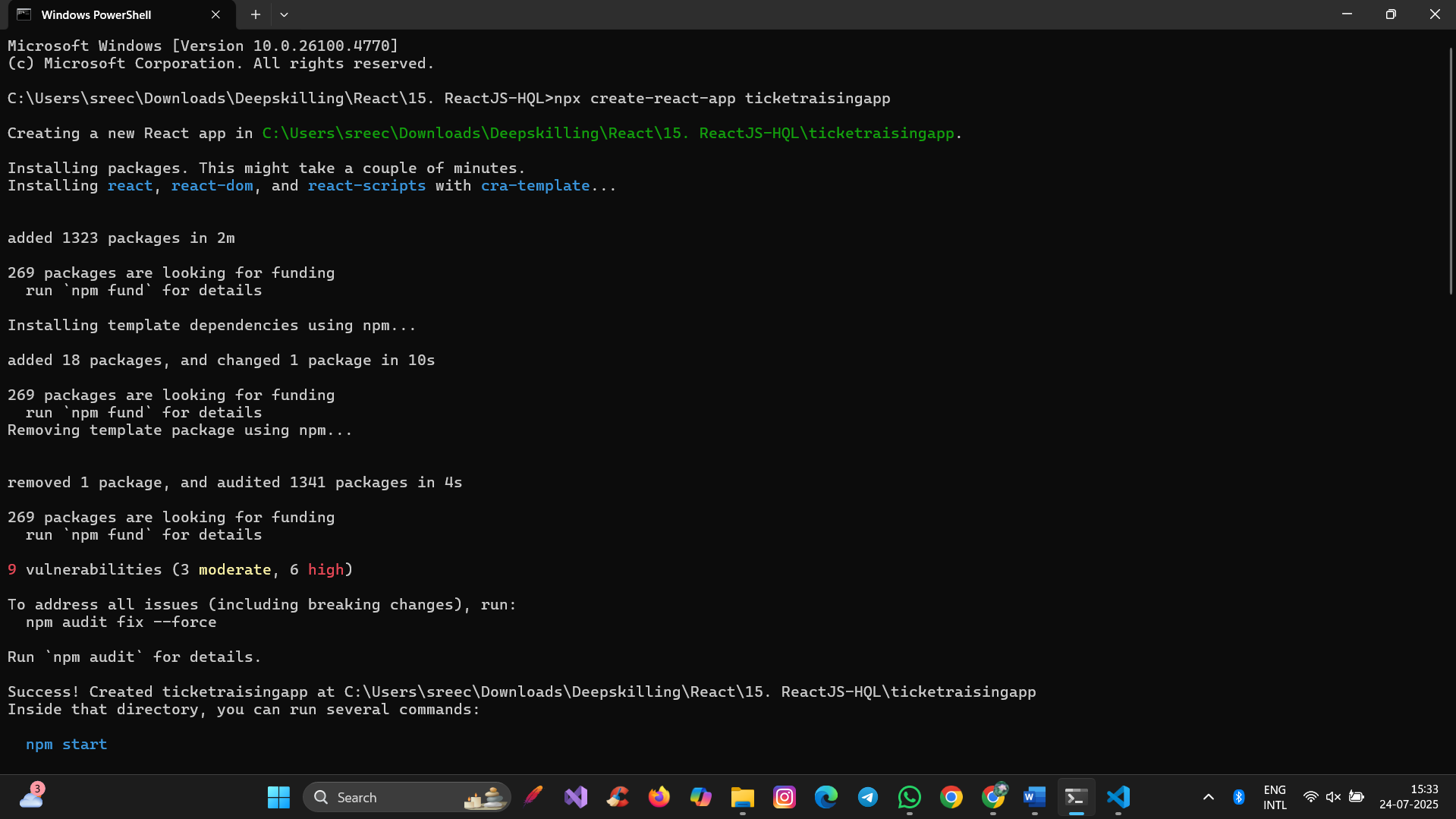


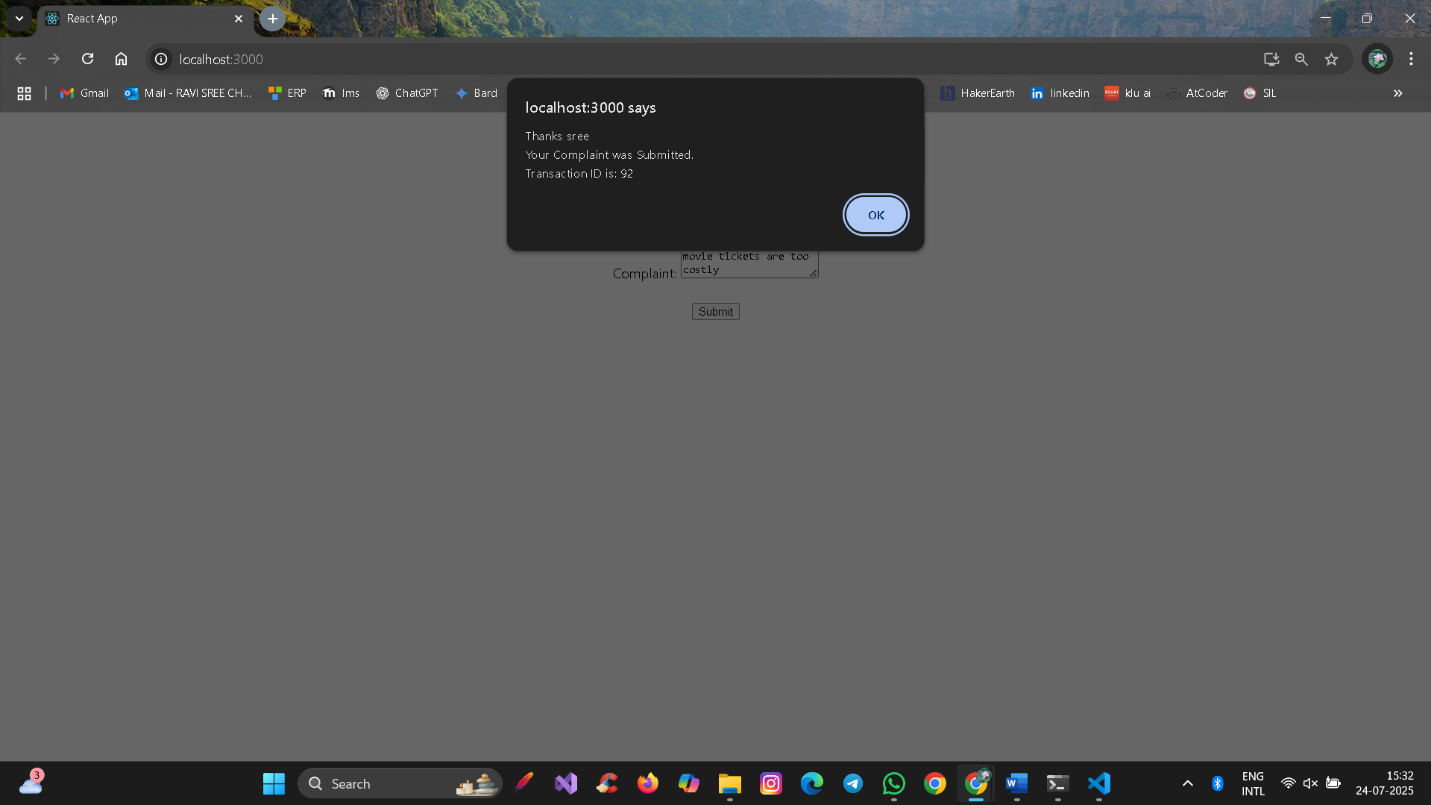
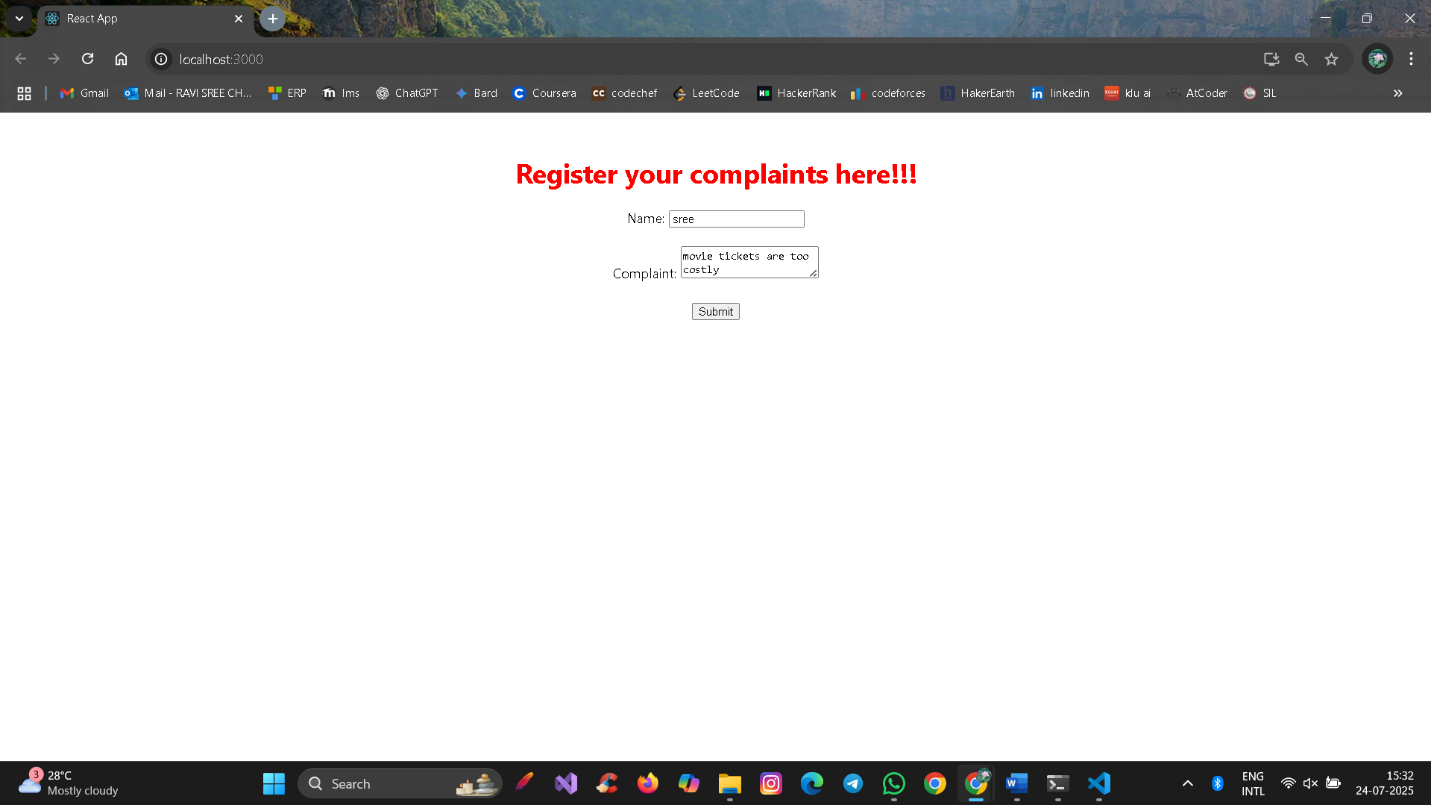
**Hint:**





**HandsOn:**

****

****