**Local and Session storage**

**Title: Build a one component that uses local storage for the simple counter while close the tab repoen the same tab then value of that counter should be last one. Second is a create email and password field alongside button save the value of the both fields on session storage and while close the tab and reopen it it will remove from session storage.**

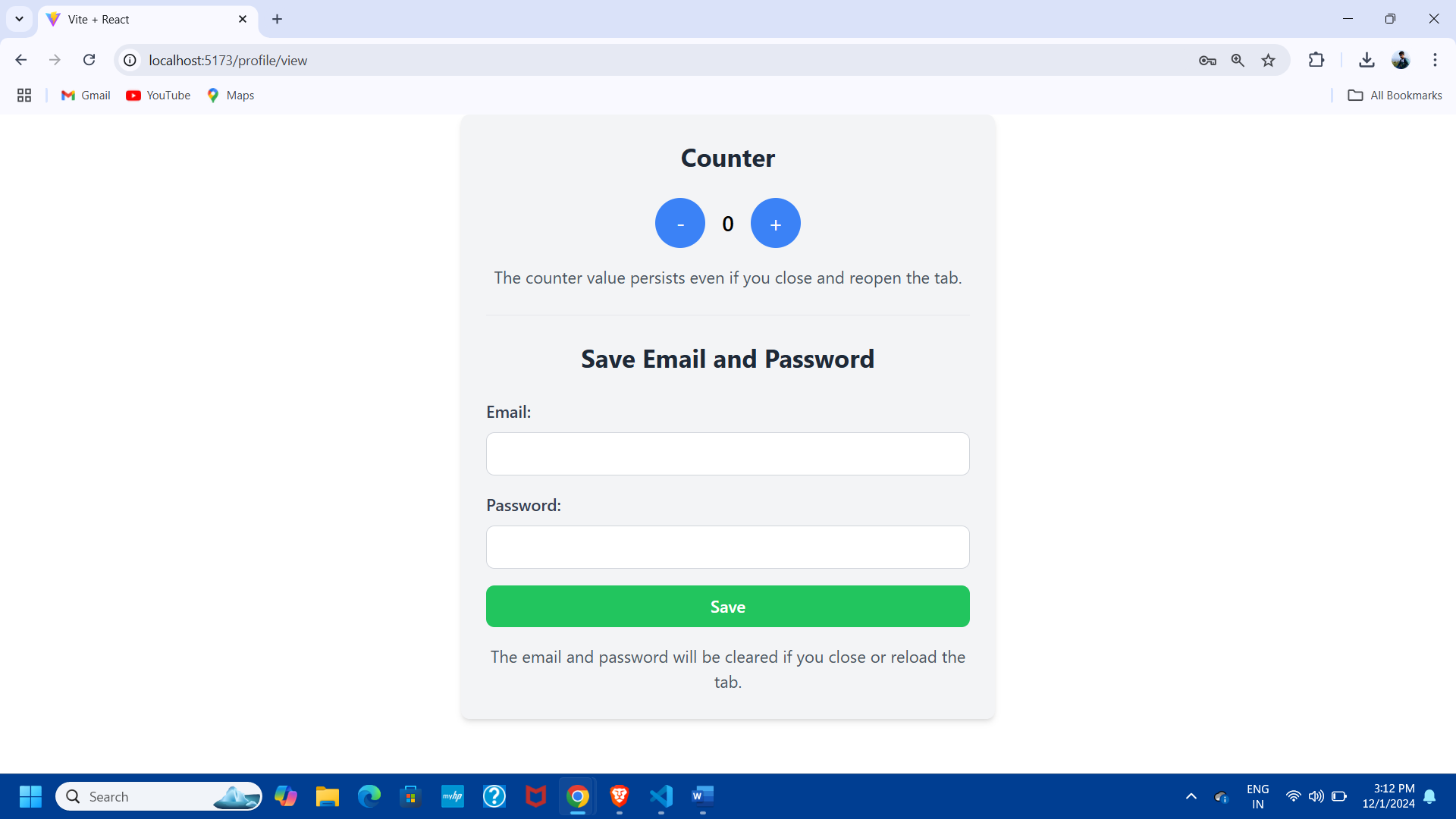
**1. Task Description:**

Create a feature to demonstrate the use of LocalStorage and SessionStorage in React.

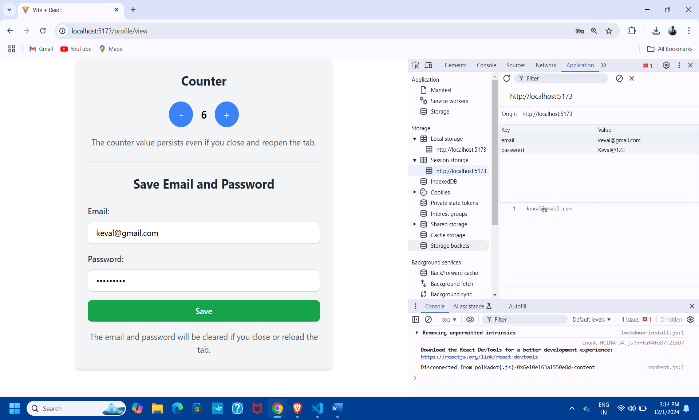
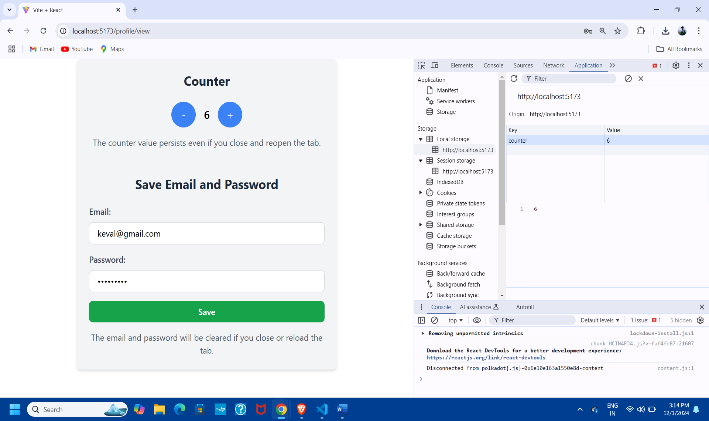
* A form allows users to input data (e.g., name, email).
* Data is saved to LocalStorage or SessionStorage based on user selection.
* Display saved data dynamically, with the option to clear storage or switch between storage modes.

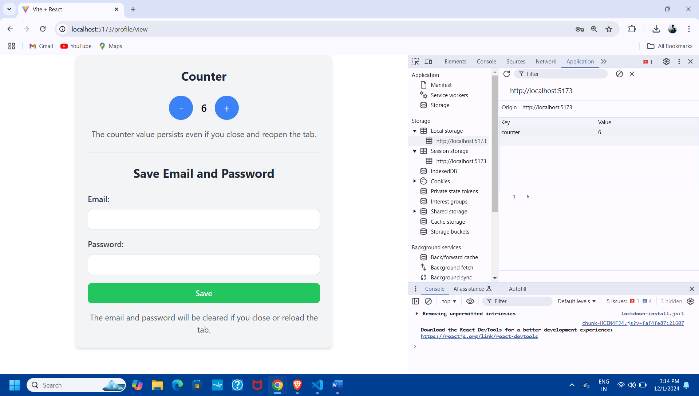
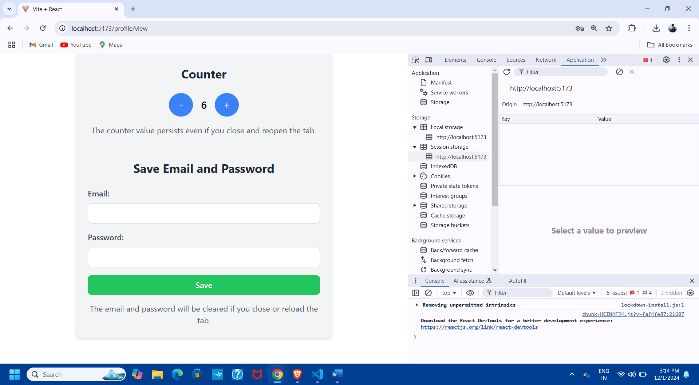
**2. Task Output Screenshot:**

**The dashboard of form is loos like :**



Following image shows how the data (E-mail address and password) stores into local storage and counter value is stored into session storage.



Following image shows , when we reload the page the session storage is erased but local storage(counter value) remain same.

**3. Widget/Algorithm Used in Task**

* **React State Hooks (useState)**:  
  Used for managing user input and toggling between LocalStorage and SessionStorage.
* **LocalStorage and SessionStorage:**
  + **LocalStorage:** Data persists even after the browser is closed.
  + **SessionStorage:** Data is cleared once the browser session ends.
* **HTML Elements Used:**
  + **Input Fields:** Used for taking user input (e.g., name, email).
  + **Radio Buttons:** To choose between LocalStorage and SessionStorage.
  + **Buttons:**
    - Submit: Saves data to the chosen storage.
    - Clear: Clears the storage and updates the view.
* **Validation Logic:**  
  Validates user inputs (e.g., name and email cannot be empty) before saving data.
* **Dynamic Display Logic:**  
  Dynamically retrieves and displays data from the selected storage in real-time.