**LAB # 10**

**Frontend development with React js and connect with backend**

**OBJECTIVE:**

To develop a user interface using **React.js** and establish communication with the **backend server** using APIs.

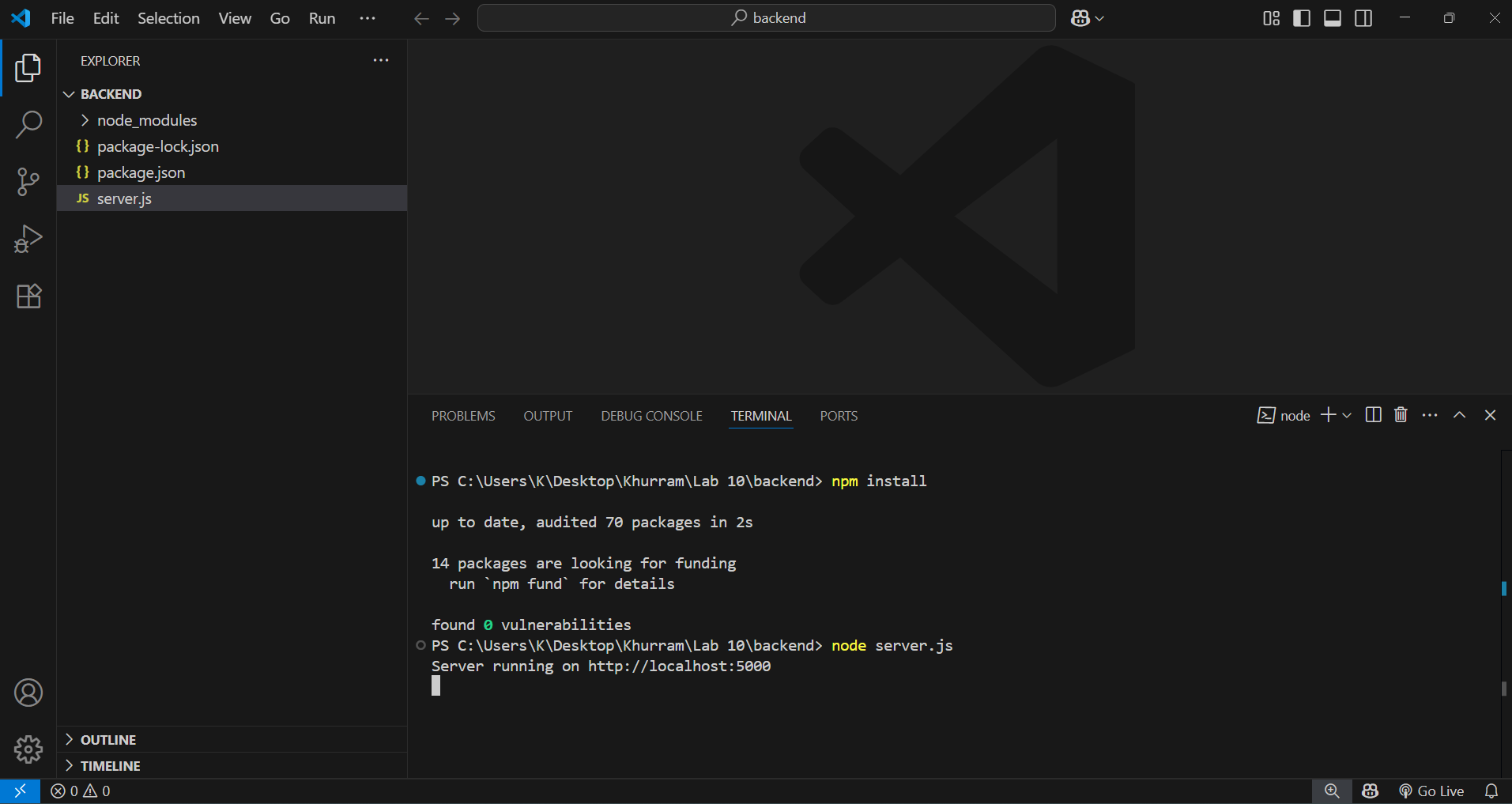
Open “backend” folder in vs code.

Open terminal and run these commands:

**npm install**

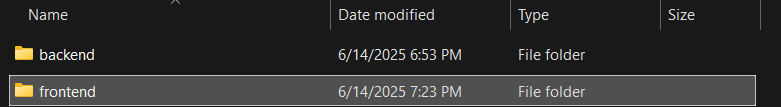
**node server.js**

Your screen will look like this:



Now create another folder “frontend”, outside of the backend folder.

Your screen will look like this:



Now open this “frontend” folder in another window of vs code.

So basically you have two vs code windows open, one for the “backend”, and one for the “frontend” folder.

While backend running, we will develop frontend:

So open the vs code window of the “frontend” folder and type the following command:

**npm create vite@latest**

This command will create vite react project. You will be ask for some project details:

**Project name:**

**.**

**Select a framework:**

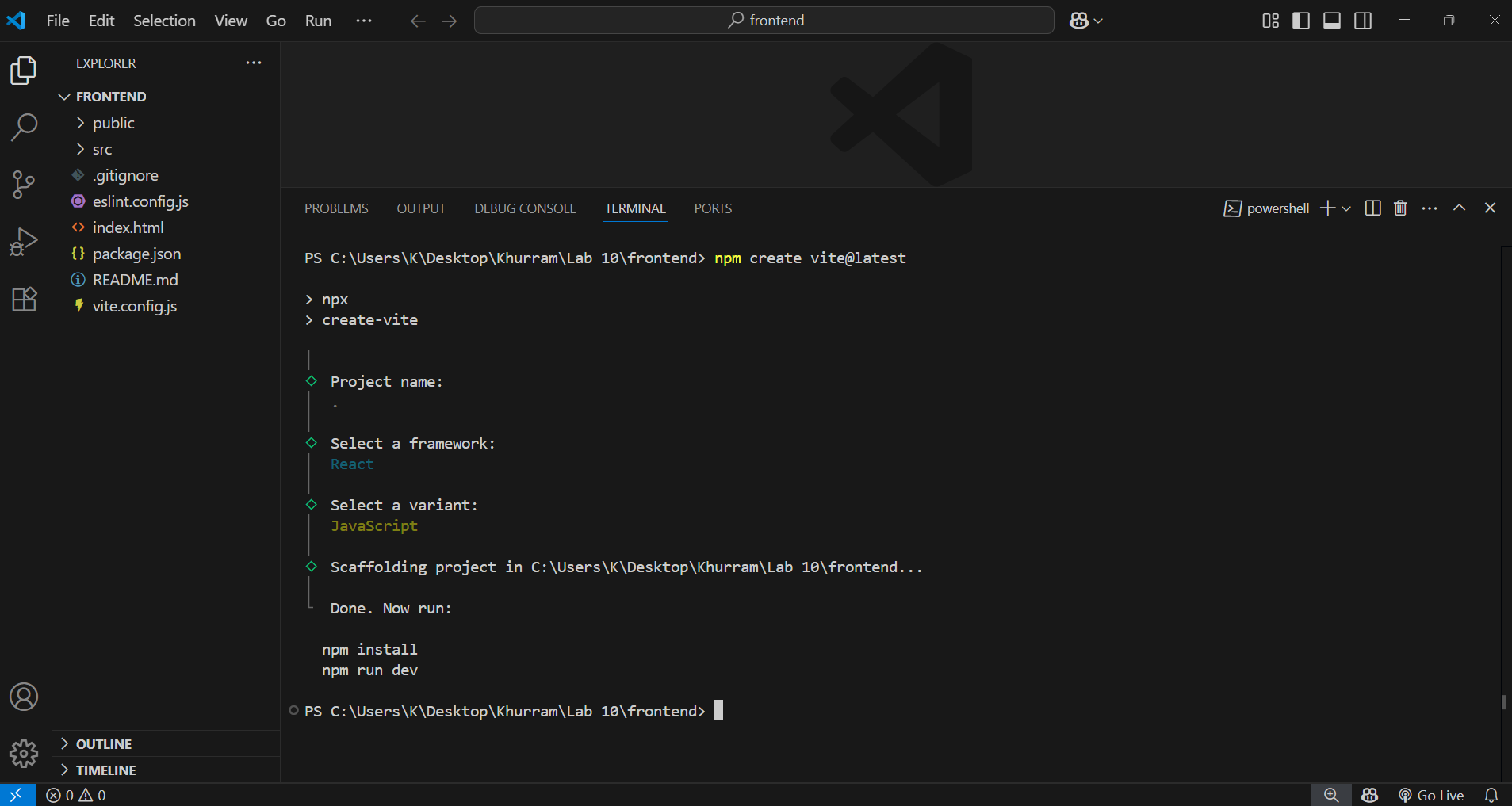
**React**

**Select a variant:**

**JavaScript**

Now your react project is created.

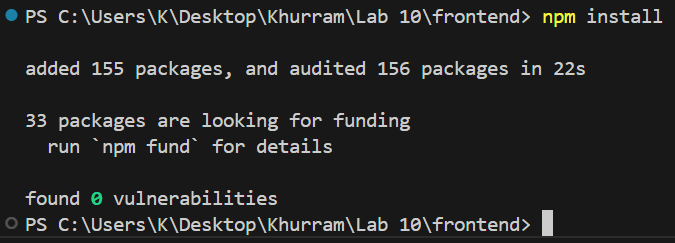
You will see screen like this:



Now run this command:

**npm install**

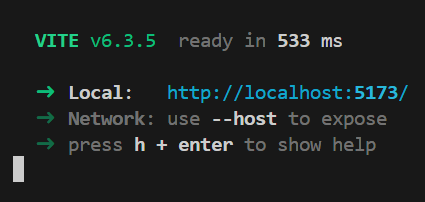
The terminal will look like this:



Now run this command:

**npm run dev**

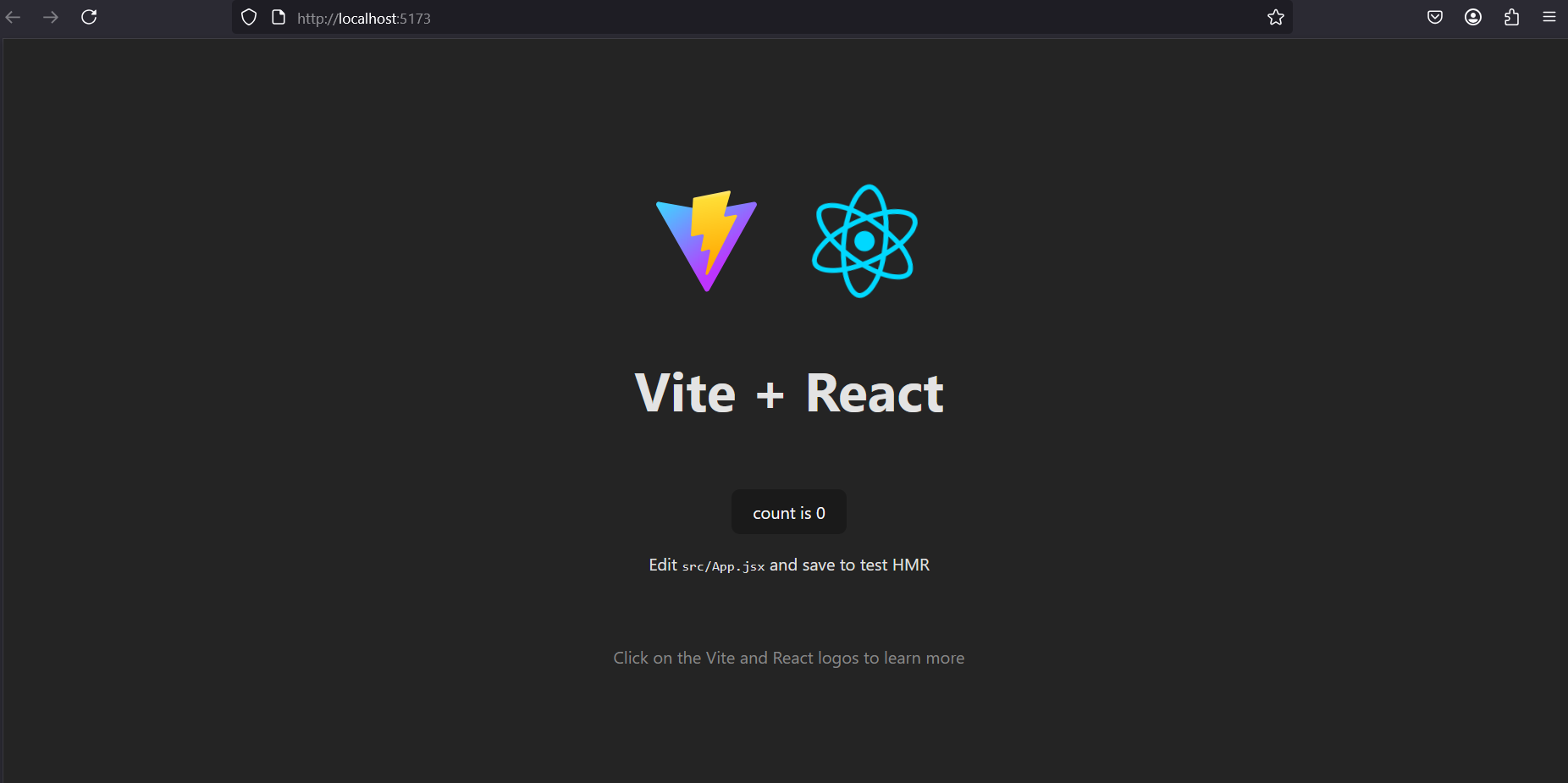
The terminal will look like this:



Open this link in browser:

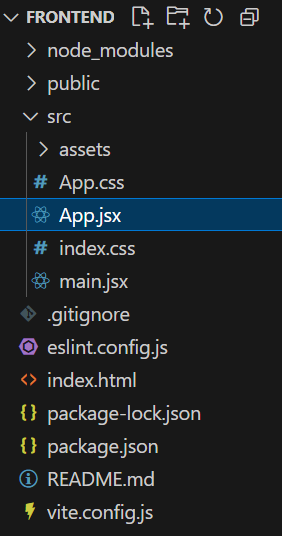
<http://localhost:5173/>

You will see screen like this:



React app is running.

Now open “App.jsx” file:



Now delete old code of App.jsx file,

And paste this new code in “App.jsx” file:

App.jsx:

// React ko import kar rahe hain

// useState or useEffect react k hook hain,

// hook aik tarah k features hain react k

import React, { useState, useEffect } from 'react';

// App component start ho raha hai

function App() {

  // Ye state sab users k data ko store karne k liye

const [users, setUsers] = useState([]);

  // Ye state naya username save karne k liye

  const [username, setUsername] = useState("");

  // Ye state update ke liye selected user ka id save karta hai

  const [editId, setEditId] = useState(null);

  // Component jab first time load ho to data fetch karo

  useEffect(() => {

    fetchUsers();

  }, []);

  // Sab users k data ko backend se fetch karo

  const fetchUsers = async () => {

    const res = await fetch("http://localhost:5000/api/get");

    const data = await res.json();

    setUsers(data); // Data ko state mai save karo

  };

  // New user create karne ka function

  const handleCreate = async () => {

    // Backend ko POST request send kar rahe hain

    const res = await fetch("http://localhost:5000/api/post", {

      method: "POST",

      headers: {

        "Content-Type": "application/json"

      },

      body: JSON.stringify({ username })

    });

    await res.json(); // Response ko ignore kar rahe hain is case mein

    setUsername("");  // Input field clear karo

    fetchUsers();     // Dubara se data fetch karo

  };

  // Kisi user ko update karne ka function

  const handleUpdate = async () => {

    await fetch(`http://localhost:5000/api/put/${editId}`, {

      method: "PUT",

      headers: {

        "Content-Type": "application/json"

      },

      body: JSON.stringify({ username })

    });

    setUsername("");  // Input field clear karo

    setEditId(null);  // Edit mode khatam karo

    fetchUsers();     // Dubara se data fetch karo

  };

  // Delete request bhejne ka function

  const handleDelete = async (id) => {

    await fetch(`http://localhost:5000/api/delete/${id}`, {

      method: "DELETE"

    });

    fetchUsers(); // Data refresh karo

  };

  // Jab edit button pe click ho to form ko update mode mai le aao

  const startEdit = (user) => {

    setUsername(user.username); // Input field mai username dikhaye

    setEditId(user.id);         // Edit mode ke liye id set karo

  };

  // JSX return ho raha hai, jo UI show karega

  return (

    <div>

      <h1>Username Manager</h1>

      {/\* Input field jahan user username type karega \*/}

      <input

        type="text"

        value={username}

        onChange={(e) => setUsername(e.target.value)}

        placeholder="Enter username"

      />

      {/\* Agar edit mode mai hain to update button dikhao \*/}

      {editId ? (

        <button onClick={handleUpdate}>Update</button>

      ) : (

        <button onClick={handleCreate}>Create</button>

      )}

      <h2>All Users</h2>

      <ul>

        {/\* Har user ko list mai show karna \*/}

        {users.map((user) => (

          <li key={user.id}>

            {user.username} (ID: {user.id})

            {/\* Edit button \*/}

            <button onClick={() => startEdit(user)}>Edit</button>

            {/\* Delete button \*/}

            <button onClick={() => handleDelete(user.id)}>Delete</button>

          </li>

        ))}

      </ul>

    </div>

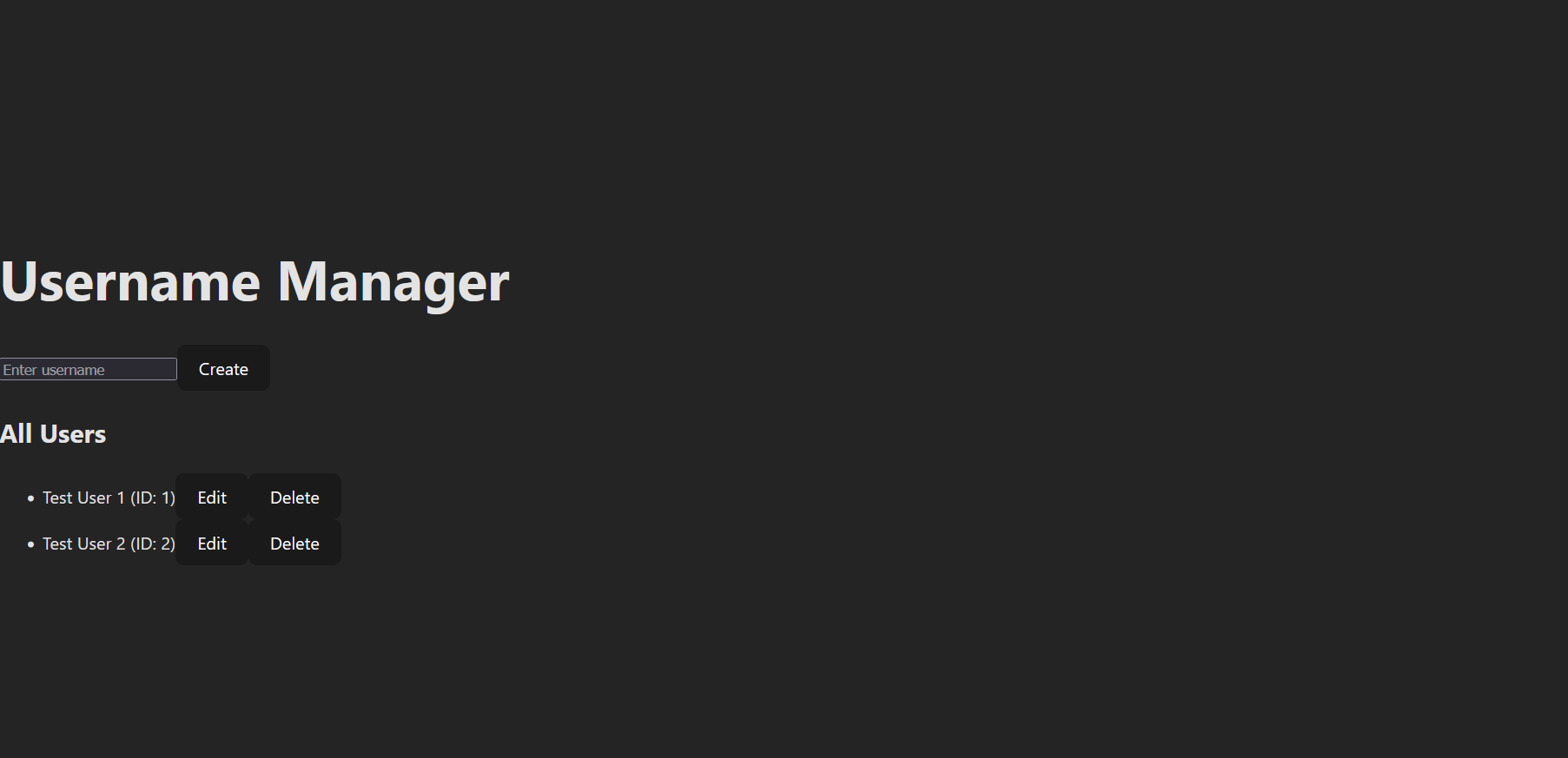
  );

}

// App component ko export kar rahe hain

export default App;

Now web app looks like this:



(Website looks like this due to react css)

Find this lab and its code on my GitHub:

<https://github.com/software-engineer-bse/Introduction-to-Software-Engineering-ITSE-Labs>

**Task:**

1. Create a frontend for calculator by using vite react and connect it with backend.