Assignment

Learner Details

- **Name: Shrayanth S**
- **Enrollment Number: SU625MR011**
- **Batch / Class: June 2025 MERN**
- **Assignment: User Directory**
- **Date of Submission: 01/08/2025**

Problem Solving Activity 1.1

1. Program Statement

Build a React-based User Directory that fetches user data from an API and displays it in a responsive card layout using reusable components (UserCard) and dynamic rendering using React hooks.

---

2. Algorithm

- Initialize the React project and required component files.
- In UserDirectory, define a state users using useState.
- Use useEffect to fetch data from https://randomuser.me/api/?results=16 on component mount.
- Store the received data into the users state.
- Loop through each user object and pass it as props to the reusable UserCard component.
- In UserCard, extract the relevant user details (name, picture, email, location) and render them inside a styled card.
- Apply responsive CSS Grid layout using TailwindCSS classes.
- Display the full directory inside UserDirectory, which is rendered from App.js.

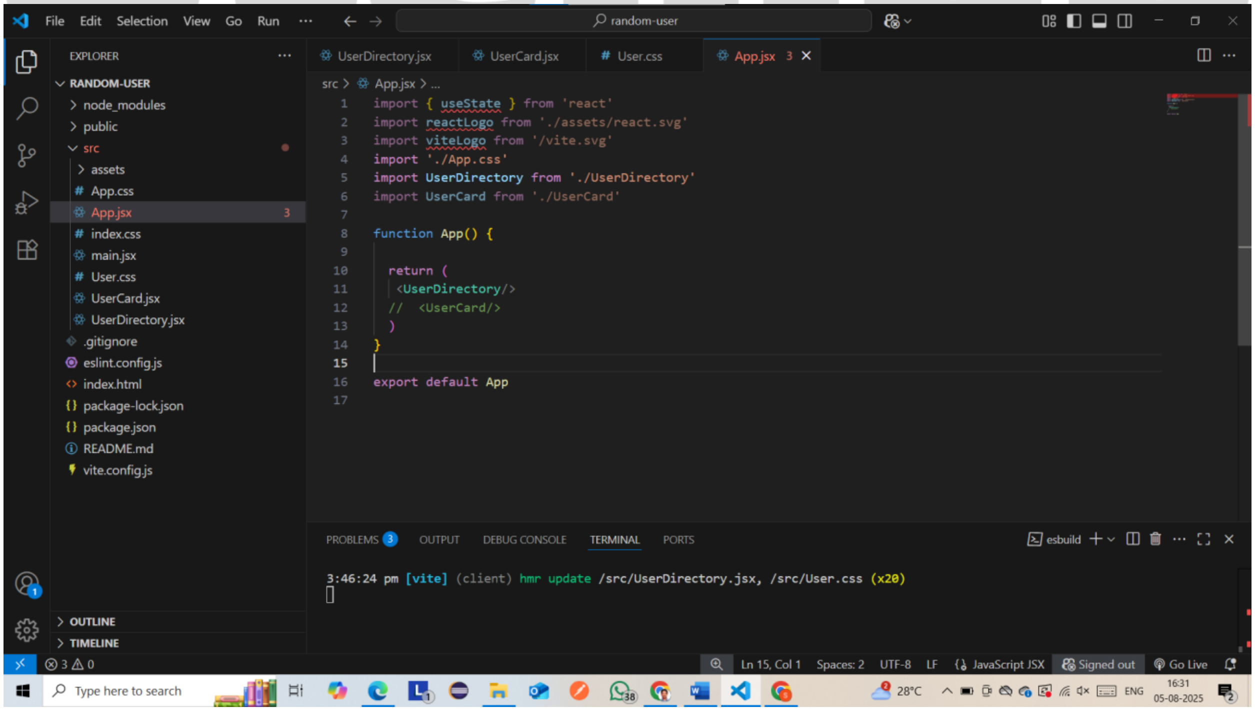
---

3. Pseudocode

```
Component: UserDirectory
  - Initialize users as empty array
  - useEffect:
      - fetch user data from API
      - convert to JSON
      - update users state with the results

  - Render:
      - Heading
      - For each user in users:
          - Render UserCard passing user as prop

Component: UserCard
  - Receive user object as prop
  - Extract name, picture, email, and location
  - Display in a styled card layout
```

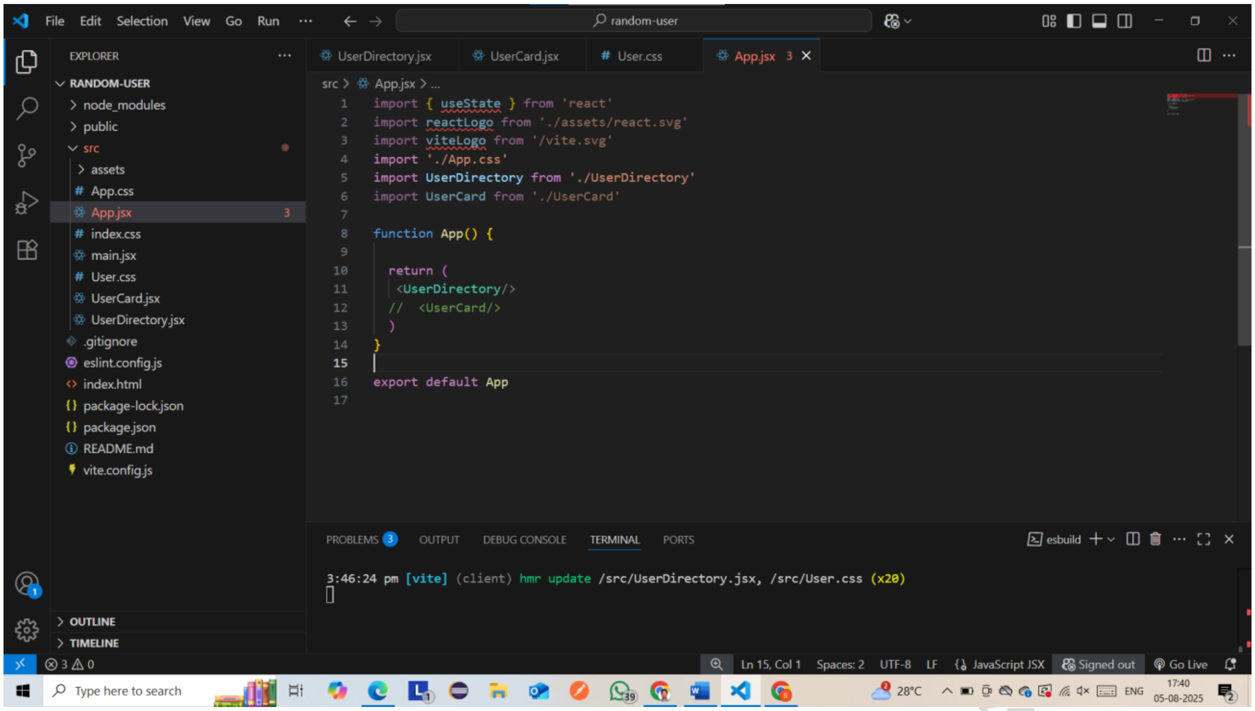
---

4. Program Code



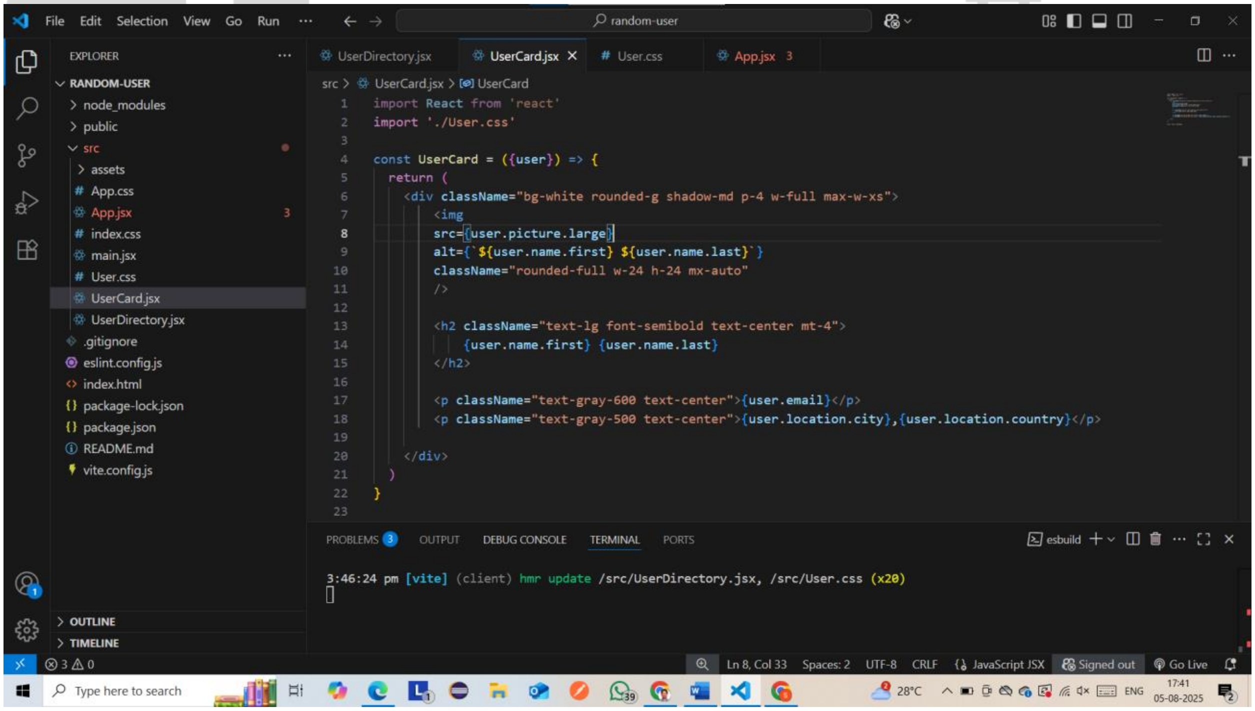
```jsx
import { useState } from 'react'
import reactLogo from './assets/react.svg'
import viteLogo from '/vite.svg'
import './App.css'
import UserDirectory from './UserDirectory'
import UserCard from './UserCard'

function App() {

  return (
   <UserDirectory/>
  //  <UserCard/>
  )
}

export default App

```

```
3:46:24 pm [vite] (client) hmr update /src/UserDirectory.jsx, /src/User.css (x20)
```



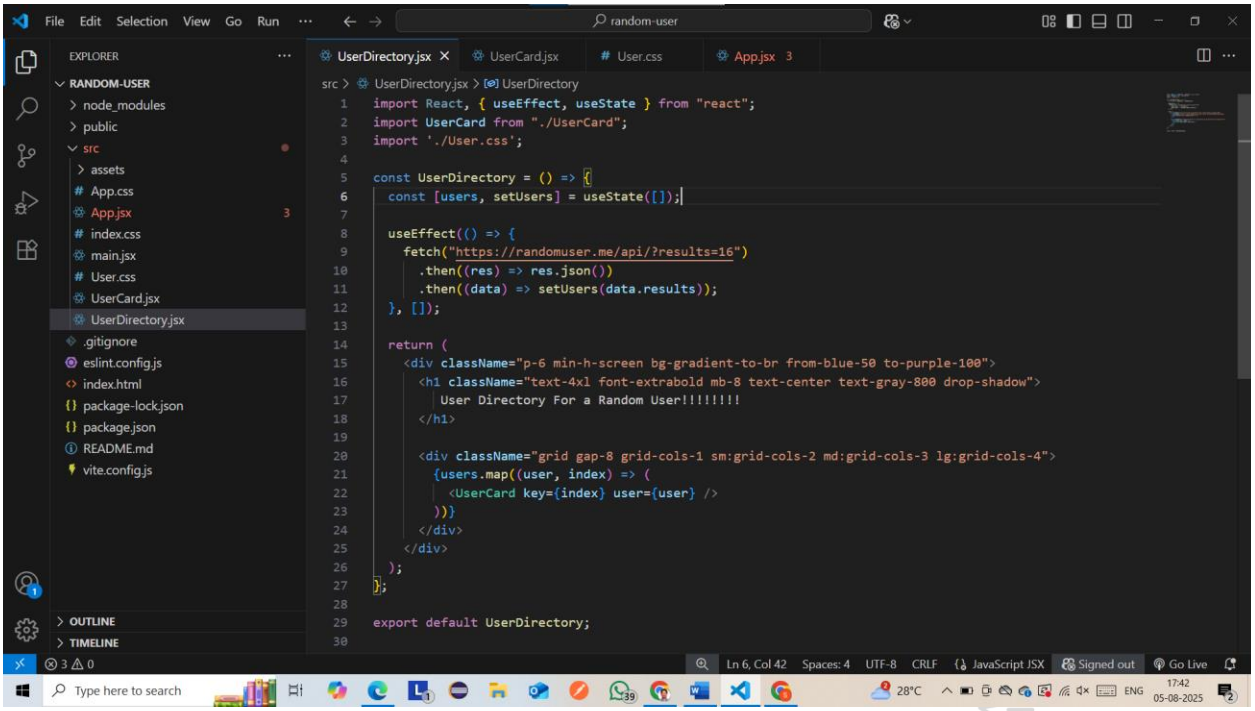
```
import { useState } from 'react'
import reactLogo from './assets/react.svg'
import viteLogo from '/vite.svg'
import './App.css'
import UserDirectory from './UserDirectory'
import UserCard from './UserCard'

function App() {

  return (
   <UserDirectory/>
  //  <UserCard/>
  )
}

export default App

```

```
3:46:24 pm [vite] (client) hmr update /src/UserDirectory.jsx, /src/User.css (x20)
```



```
import React from 'react'
import './User.css'

const UserCard = ({user}) => {
  return (
    <div className="bg-white rounded-g shadow-md p-4 w-full max-w-xs">
        <img
        src={user.picture.large}
        alt={`${user.name.first} ${user.name.last}`}
        className="rounded-full w-24 h-24 mx-auto"
        />

        <h2 className="text-lg font-semibold text-center mt-4">
            {user.name.first} {user.name.last}
        </h2>

        <p className="text-gray-600 text-center">{user.email}</p>
        <p className="text-gray-500 text-center">{user.location.city},{user.location.country}</p>

    </div>
  )
}

```

```
3:46:24 pm [vite] (client) hmr update /src/UserDirectory.jsx, /src/User.css (x20)
```

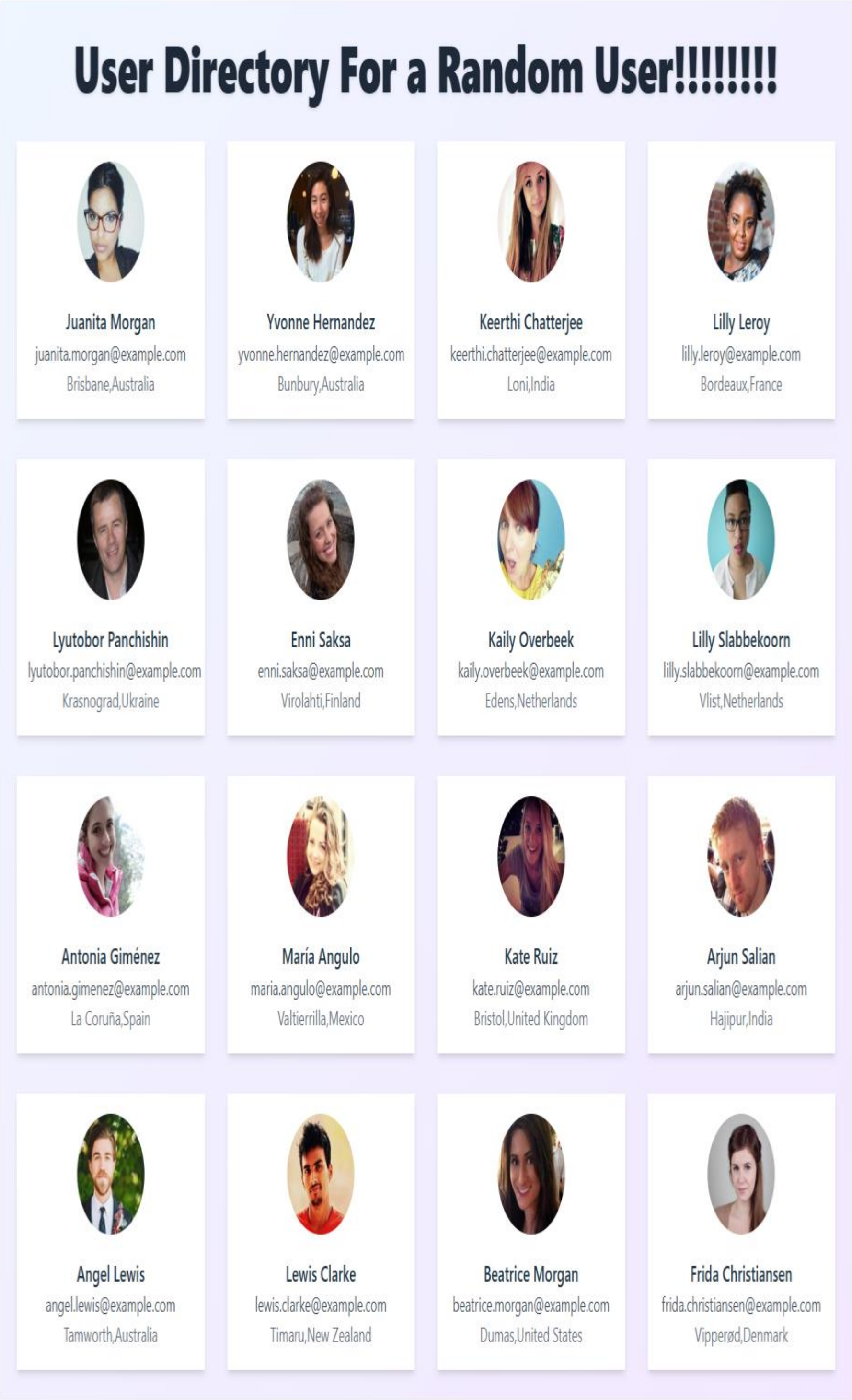


```
import React, { useEffect, useState } from "react";
import UserCard from "./UserCard";
import './User.css';

const UserDirectory = () => {
  const [users, setUsers] = useState([]);

  useEffect(() => {
    fetch("https://randomuser.me/api/?results=16")
      .then((res) => res.json())
      .then((data) => setUsers(data.results));
  }, []);

  return (
    <div className="p-6 min-h-screen bg-gradient-to-br from-blue-50 to-purple-100">
      <h1 className="text-4xl font-extrabold mb-8 text-center text-gray-800 drop-shadow">
        User Directory For a Random User!!!!!!!!
      </h1>

      <div className="grid gap-8 grid-cols-1 sm:grid-cols-2 md:grid-cols-3 lg:grid-cols-4">
        {users.map((user, index) => (
          <UserCard key={index} user={user} />
        ))}
      </div>
    </div>
  );
};

export default UserDirectory;
```

5. Test Cases

Test Case	Description	Expected Output	Result
TC1	Load component with fetched users	16 user cards displayed	Pass
TC2	Ensure card shows user name, email, location	Proper user details visible on card	Pass
TC3	Layout adjusts on different screen sizes	Grid adapts (1, 2, 3, 4 columns)	Pass
TC4	UserCard component reusability	Works with any user object passed as prop	Pass

6. Screenshot of outputs



User Directory For a Random User!!!!!!!!

Juanita Morgan
juanita.morgan@example.com
Brisbane,Australia

Yvonne Hernandez
yvonne.hernandez@example.com
Bunbury,Australia

Keerthi Chatterjee
keerthi.chatterjee@example.com
Loni,India

Lilly Leroy
lilly.leroy@example.com
Bordeaux,France

Lyutobor Panchishin
lyutobor.panchishin@example.com
Krasnograd,Ukraine

Enni Saksa
enni.saksa@example.com
Virolahti,Finland

Kaily Overbeek
kaily.overbeek@example.com
Edens,Netherlands

Lilly Slabbekoorn
lilly.slabbekoorn@example.com
Vlist,Netherlands

Antonia Giménez
antonia.gimenez@example.com
La Coruña,Spain

María Angulo
maria.angulo@example.com
Valtierrilla,Mexico

Kate Ruiz
kate.ruiz@example.com
Bristol,United Kingdom

Arjun Salian
arjun.salian@example.com
Hajipur,India

Angel Lewis
angel.lewis@example.com
Tamworth,Australia

Lewis Clarke
lewis.clarke@example.com
Timaru,New Zealand

Beatrice Morgan
beatrice.morgan@example.com
Dumas,United States

Frida Christiansen
frida.christiansen@example.com
Vipperød,Denmark

7. Observation / Reflection

While working on this assignment, I learned how to use React hooks like useState and useEffect and build reusable components. I practiced fetching API data and displaying it in a responsive layout using Tailwind CSS. A key challenge was handling the layout before the data loaded properly. In the future, I’d like to add loading indicators, search/filter features, and test my components for better performance.