Job Simulation: Dynamic User Director

Project Title: Dynamic User Directory Using Props and Conditional Rendering

Role: React Front-End Development Intern

Technology Stack: React, JSX, CSS

Objective

To implement a reusable user directory where user data is passed via props and rendered conditionally based on the user's role. The simulation enhances understanding of **props**, **JSX**, **component hierarchy**, and **conditional UI rendering** in React.

Task Overview

As part of your React learning, you are required to build a **User Directory** that:

- Displays a list of users with details like name, email, age, and role

- Uses **props** to pass user data (as an array of objects) from the parent component

- Displays different badges or colors for different roles (e.g., Admin, Guest, Member)

- Renders a fallback UI when no user data is available

Task Requirements

1. Functionality:

- Create a parent component that holds a users array (with objects containing name, email, age, role)
- Pass this array to a child component using props
- Inside the child component, use .map() to render each user
- Use conditional rendering to show different styles or messages for different roles ○ If the array is empty, show "No users found"

2. User Interface (UI):

- Layout should include: a header, a user list container, and styled user cards ○ Use conditional background or badge color for roles:
  - Admin: Red
  - Member: Blue

  - – Guest: Gray
- ○ Apply responsive design with flexbox or grid

3. Code Structure:

- ○ Modular structure with App.jsx, UserList.jsx, and UserCard.jsx
- ○ Props are passed and destructured appropriately
- ○ Each component contains comments to explain the logic
- ○ Use CSS classes for role-based styling

Bonus (Optional)

- Add a search box to filter users by name
- Add role count (e.g., total Admins: 2)
- Highlight underage users (age < 18)

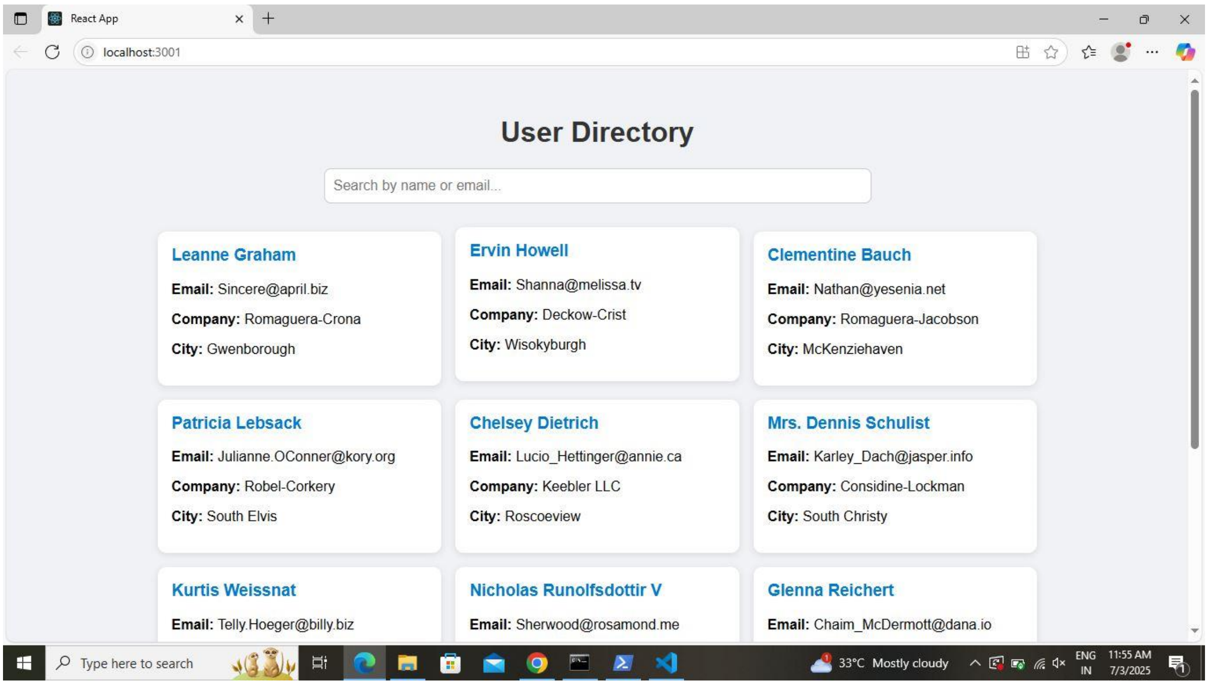
Deliverables

- App.jsx – Holds the data and passes it as props
- UserList.jsx – Maps over users and renders UserCard
- UserCard.jsx – Displays user details with conditional styling
- style.css – Styles for layout and roles

Note: Add a comment in UserList.jsx explaining how conditional rendering works based on the role.

Learning Outcomes

- Using props to manage reusable data
- Performing conditional rendering in React
- Structuring clean and maintainable component hierarchies
- Styling dynamically based on props



User Directory

Search by name or email...

Leanne Graham
Email: Sincere@april.biz
Company: Romaguera-Crona
City: Gwenborough

Ervin Howell
Email: Shanna@melissa.tv
Company: Deckow-Crist
City: Wisokyburgh

Clementine Bauch
Email: Nathan@yesenia.net
Company: Romaguera-Jacobson
City: McKenziehaven

Patricia Lebsack
Email: Julianne.OConner@kory.org
Company: Robel-Corkery
City: South Elvis

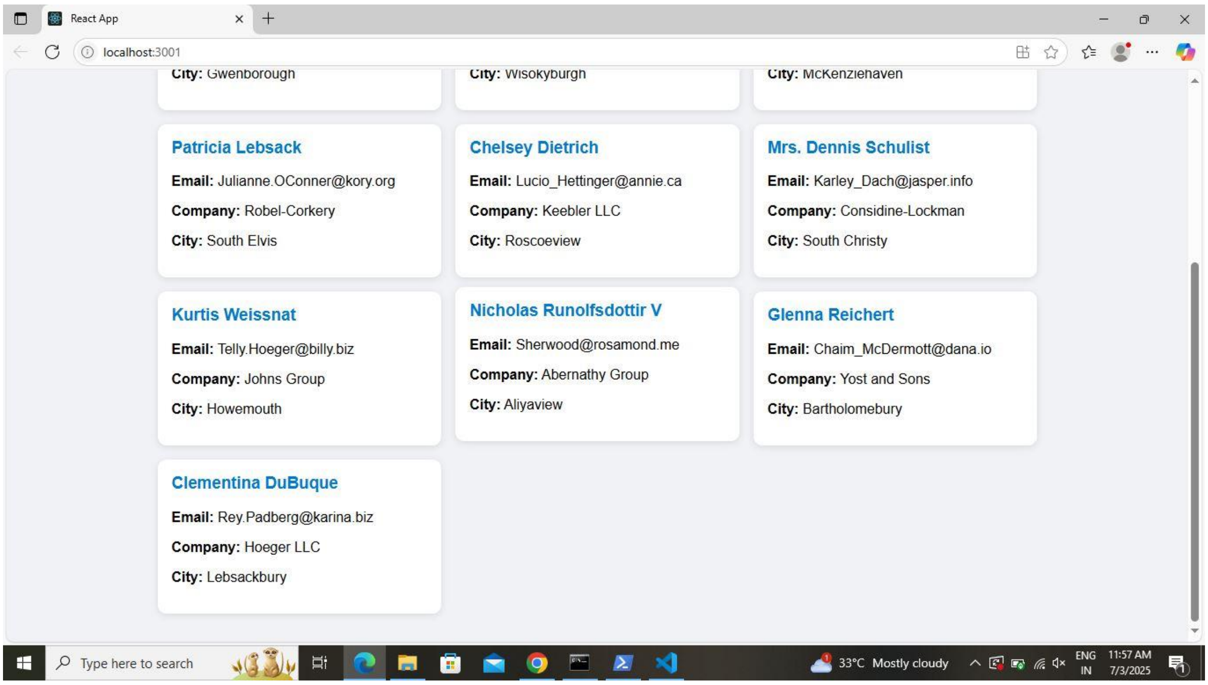
Chelsey Dietrich
Email: Lucio_Hettinger@annie.ca
Company: Keebler LLC
City: Roscoeview

Mrs. Dennis Schulist
Email: Karley_Dach@jasper.info
Company: Considine-Lockman
City: South Christy

Kurtis Weissnat
Email: Telly.Hoeger@billy.biz

Nicholas Runolfsdottir V
Email: Sherwood@rosamond.me

Glenna Reichert
Email: Chaim_McDermott@dana.io



City: Gwenborough

City: Wisokyburgh

City: McKenziehaven

Patricia Lebsack
Email: Julianne.OConner@kory.org
Company: Robel-Corkery
City: South Elvis

Chelsey Dietrich
Email: Lucio_Hettinger@annie.ca
Company: Keebler LLC
City: Roscoeview

Mrs. Dennis Schulist
Email: Karley_Dach@jasper.info
Company: Considine-Lockman
City: South Christy

Kurtis Weissnat
Email: Telly.Hoeger@billy.biz
Company: Johns Group
City: Howemouth

Nicholas Runolfsdottir V
Email: Sherwood@rosamond.me
Company: Abernathy Group
City: Aliyaview

Glenna Reichert
Email: Chaim_McDermott@dana.io
Company: Yost and Sons
City: Bartholomebury

Clementina DuBuque
Email: Rey.Padberg@karina.biz
Company: Hoeger LLC
City: Lebsackbury