CommPeak React Practical Task

Task Overview: Inbox Page Implementation

You are required to build an **Inbox page** with the following features:

Data Input

- You are given two JSON files:
    1. **Contacts** – contains contact details (e.g., name, phone, city)
    2. **Conversations** – contains message history

- **Match conversations to contacts** using the **phone number** field.

UI Structure & Behavior

- **Left Sidebar – Contact List**

    - Display a list of contacts with:
        - **Full name**
        - **Last message preview**
        - **Time of last message**

    - Clicking on a contact:
        - Opens the relevant conversation in the **right panel**
        - Highlights the currently selected conversation

◆ **Right Panel – Conversation**

- Show:
    - **Contact's full name**
    - **Avatar with initials**
    - **Total number of messages**

- Display message bubbles:
    - **Outgoing messages** aligned **right**
    - **Incoming messages** aligned **left**

- Allow sending new messages:
    - Appends a message to the conversation thread

Session Logic

- The session is **ephemeral**:
    - When switching between conversations, the previous message history is preserved
    - On browser **refresh**, everything resets
    - **No need to store data** in `localStorage`, cookies, or backend – this is a **frontend-only** task

Sending a Message

- At the bottom of the conversation panel, add a **message input box**
- Support **placeholders** in square brackets inside the message text, e.g., `[first_name]`
    - When sending the message, replace placeholders with actual contact data
    - Example: `[first_name]` → `John`

commpeak

Search Functionality

- Add a **search input field** above the contact list
- Allow real-time filtering of contacts by:
  - **Full name** (first name + last name)
  - **Phone number**
- Search should be **case-insensitive** and update the list dynamically

Design Guidelines

- The UI does **not** need to match the exact design in terms of spacing (padding, margin), but:
  - The **structure and layout** should match
  - Use the following **color variables**:

```css
:root {

  --color-primary: #0066FF;        /* Primary (buttons, highlights) */

  --color-text: #1E254F;           /* Main text */

  --color-divider: #CBD5E0;        /* Borders, separators */

  --color-placeholder: #3D466E;    /* Placeholder text */

  --color-bg: #FFFFFF;             /* Background */

  --color-active: #E5F1FF;         /* Active conversation, outgoing message */

  --color-muted: #718096;          /* Timestamps, secondary text */

  --color-incoming: #F2F6F9;       /* Incoming message background */

}
```

commpeak

commpeak

Writing message:

JD John Doe

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do
Sent: 14:02

30.07.2025

Lorem ipsum
Sent: 14:02

Lorem ipsum
Sent: 14:02

Lorem ipsum
Sent: 14:02

Lorem ipsum
Sent: 14:02

Lorem ipsum
Sent: 14:02

Search

3 Conversations

RL John Doe   00:00
Lorem ipsum dolor sit amet, consectetur adipisc...

RL Jane Smith   00:00
Lorem ipsum dolor sit amet, consectetur adipisc...

RL Michael Levi   00:00
Lorem ipsum dolor sit amet, consectetur adipisc...

Good morning [first_name], we have new offer for people who lives in [city]

first_name   last_name   city   phone

Message sent:

JD John Doe

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do
Sent: 14:02

30.07.2025

Lorem ipsum
Sent: 14:02

Lorem ipsum
Sent: 14:02

Lorem ipsum
Sent: 14:02

Lorem ipsum
Sent: 14:02

Good morning John, we have new offer for people who lives in Tel Aviv
Sent: 19:02

Search

3 Conversations

RL John Doe   00:00
Lorem ipsum dolor sit amet, consectetur adipisc...

RL Jane Smith   00:00
Lorem ipsum dolor sit amet, consectetur adipisc...

RL Michael Levi   00:00
Lorem ipsum dolor sit amet, consectetur adipisc...

Write something...

first_name   last_name   city   phone