TASK-26

Building a Real-Time Chat App:

26. Create a Component with Composition:

– *Build a `ChatApp.jsx` that uses components like `ChatHeader`, `MessageList`, and `MessageInput`.*

---

1. Folder Structure:

```
src/
 ├── components/
 │   ├── ChatApp.jsx
 │   ├── ChatHeader.jsx
 │   ├── MessageList.jsx
 │   ├── MessageInput.jsx
 │   ├── ChatApp.css
 ├── App.jsx
 ├── index.js
 ├── index.css
```

---

2. File Contents:

2.1 index.js

– **The entry point for your React app:**

```
import React from "react";
import ReactDOM from "react-dom";
import App from "./App";
import "./index.css";

ReactDOM.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>,
  document.getElementById("root")
);
```

2.2. App.jsx

– ***This file integrates the ChatApp component:***

```
        import React from "react";
        import ChatApp from "./components/ChatApp";

        function App() {
         return (
          <div>
           <ChatApp />
          </div>
         );
        }

export default App;
```

2.3. components/ChatApp.jsx:

– ***The main component of the chat app:***

```
        import React, { useState } from "react";
        import ChatHeader from "./ChatHeader";
        import MessageList from "./MessageList";
        import MessageInput from "./MessageInput";
        import "./ChatApp.css";

        const ChatApp = () => {
         const [messages, setMessages] = useState([]);

         const handleSendMessage = (message) => {
          setMessages([...messages, message]);
         };

         return (
          <div className="chat-app">
           <ChatHeader />
           <MessageList messages={messages} />
           <MessageInput onSendMessage={handleSendMessage} />
          </div>
         );
        };

export default ChatApp;
```

2.4. components/ChatHeader.jsx:

- ***The header component:***

```
import React from "react";

const ChatHeader = () => {
 return (
  <div className="chat-header">
   <h1>Chat Application</h1>
  </div>
 );
};

export default ChatHeader;
```

2.5. components/MessageList.jsx:

- ***The message list component:***

```
import React from "react";

const MessageList = ({ messages }) => {
 return (
  <div className="message-list">
   {messages.length > 0 ? (
    messages.map((message, index) => (
     <div key={index} className="message">
      {message}
     </div>
    ))
   ) : (
    <p>No messages yet. Start chatting!</p>
   )}
  </div>
 );
};

export default MessageList;
```

2.6. components/MessageInput.jsx:

– ***The message input component:***

```
import React, { useState } from "react";

const MessageInput = ({ onSendMessage }) => {
  const [message, setMessage] = useState("");

  const handleInputChange = (e) => setMessage(e.target.value);

  const handleSendClick = () => {
    if (message.trim()) {
      onSendMessage(message);
      setMessage("");
    }
  };

  return (
    <div className="message-input">
      <input
        type="text"
        value={message}
        onChange={handleInputChange}
        placeholder="Type your message here..."
      />
      <button onClick={handleSendClick}>Send</button>
    </div>
  );
};

export default MessageInput;
```

2.7. components/ChatApp.css:

– ***Basic CSS for the chat app:***

```
.chat-app {
  display: flex;
  flex-direction: column;
  height: 100vh;
  max-width: 600px;
  margin: 0 auto;
  border: 1px solid #ccc;
  border-radius: 8px;
  overflow: hidden;
```

```
}

.chat-header {
  background-color: #007bff;
  color: white;
  text-align: center;
  padding: 10px;
  font-size: 1.5rem;
}

.message-list {
  flex: 1;
  padding: 10px;
  overflow-y: auto;
  background-color: #f9f9f9;
}

.message {
  padding: 8px;
  margin: 5px 0;
  background-color: #007bff;
  color: white;
  border-radius: 5px;
  max-width: 70%;
}

.message-input {
  display: flex;
  padding: 10px;
  background-color: #fff;
  border-top: 1px solid #ccc;
}

.message-input input {
  flex: 1;
  padding: 10px;
  border: 1px solid #ccc;
  border-radius: 4px;
  margin-right: 10px;
}

.message-input button {
  background-color: #007bff;
  color: white;
  border: none;
  padding: 10px 20px;
  border-radius: 4px;
```

```
                cursor: pointer;
              }

              .message-input button:hover {
                background-color: #0056b3;
}
```

2.8. index.css:

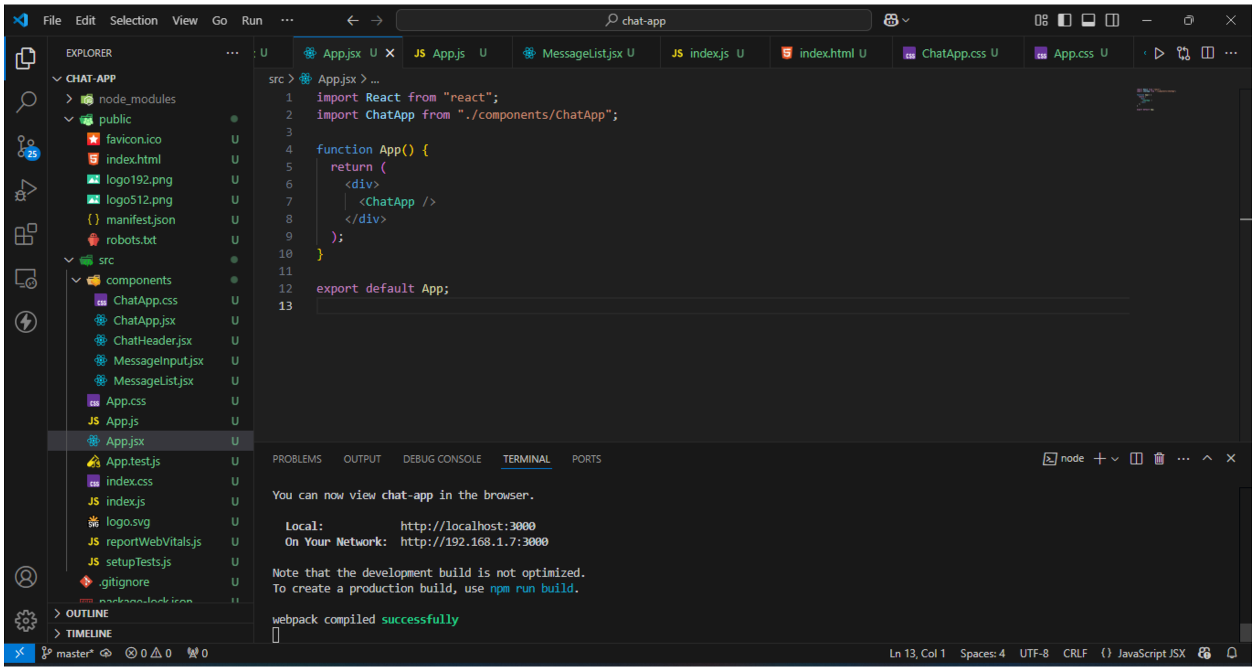
– **Global CSS:**

```
              body {
                margin: 0;
                font-family: Arial, sans-serif;
                background-color: #f0f0f0;
              }

              #root {
                display: flex;
                justify-content: center;
                align-items: center;
                height: 100vh;
}
```

3. Run the Application:

```
npm start
```



```
import React from "react";
import ChatApp from "./components/ChatApp";

function App() {
  return (
    <div>
      <ChatApp />
    </div>
  );
}

export default App;
```

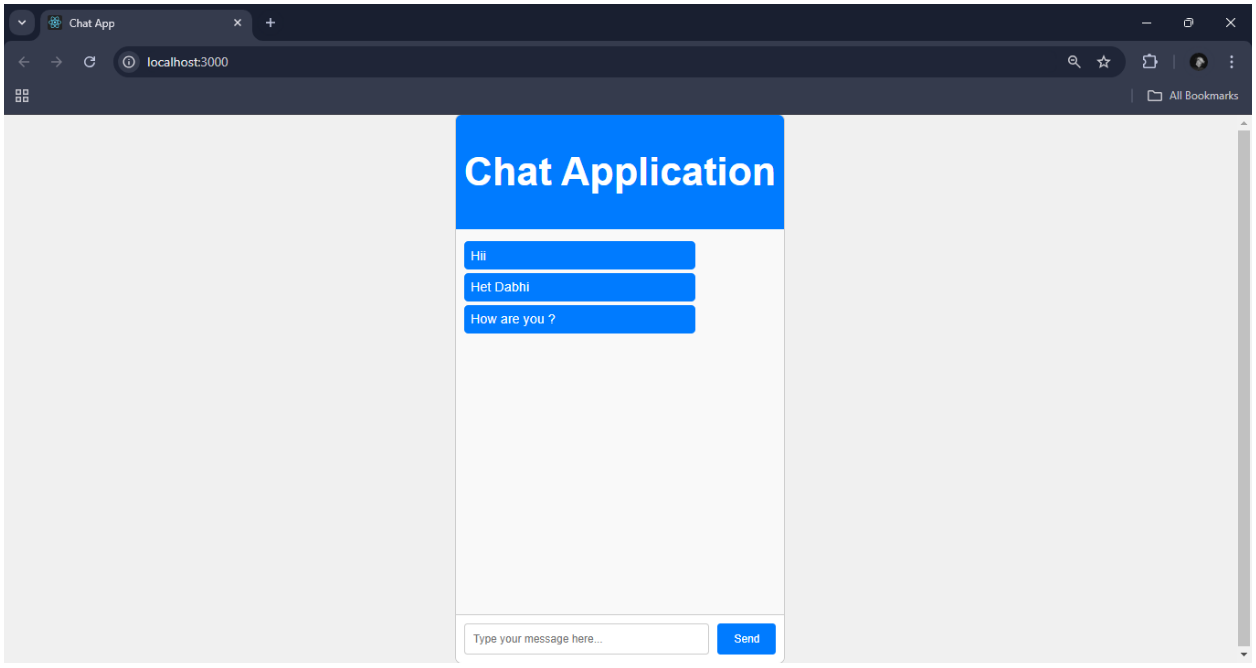
```
You can now view chat-app in the browser.

  Local:            http://localhost:3000
  On Your Network:  http://192.168.1.7:3000

Note that the development build is not optimized.
To create a production build, use npm run build.

webpack compiled successfully
```

Output:



Chat Application

Hii

Het Dabhi

How are you ?