



Week 8.4

Recap Everything, Build PayTM Frontend

In this lecture, Harkirat guides us through an [end-to-end tutorial](#) on building a comprehensive [full-stack application](#) resembling [Paytm](#). While there are no specific notes provided for this section, a mini guide is outlined below to assist you in navigating through each step of the tutorial. Therefore, it is strongly advised to actively follow along during the lecture for a hands-on learning experience.

It's important to note that this session primarily focuses on the frontend section of the application. For the backend portion, be sure to check out the content covered in the previous 8.2 lecture.

[Recap Everything, Build PayTM Frontend](#)

[Step 1 - Add routing to the react app](#)

[Solution](#)

[Step 2 - Create and hook up Signup page](#)

[Step 3 - Create the signin page](#)

[Step 4 - Dashboard page](#)

[Step 5 - Auth Components](#)

[1. Heading component](#)

[Code](#)

[2. Sub Heading component](#)[Code](#)[3. InputBox component](#)[Code](#)[4. Button Component](#)[Code](#)[5. BottomWarning](#)[Code](#)[Full Signup component](#)[Code](#)[Full Signin component](#)[Code](#)[Step 6 - Signin-ed Comonents](#)[1. Appbar](#)[Code](#)[2. Balance](#)[Code](#)[3. Users component](#)[Code](#)[4. SendMoney Component](#)[Code](#)[Step 7 - Wiring up the backend calls](#)

Step 1 - Add routing to the react app

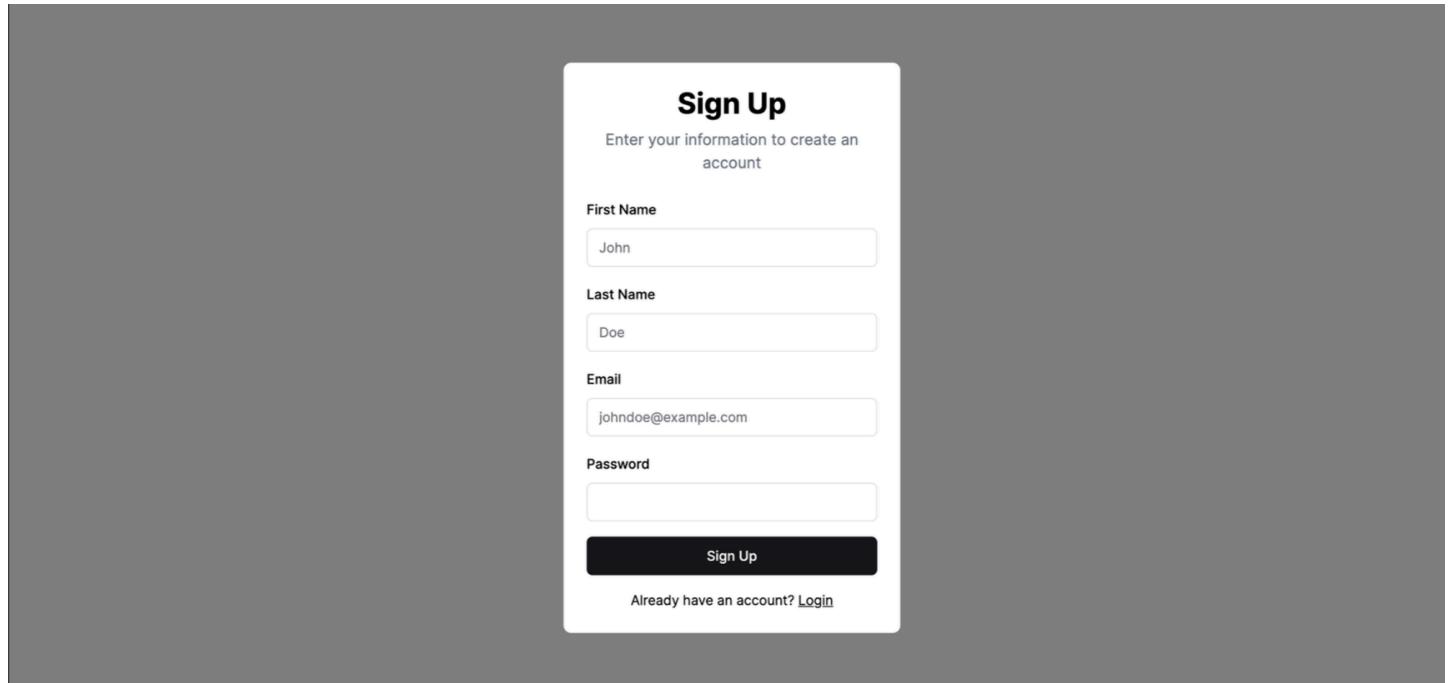
Import `react-router-dom` into your project and add the following routes -

1. `/signup` - The signup page
2. `/signin` - The signin page
3. `/dashboard` - Balances and see other users on the platform.
4. `/send` - Send money to other users

▼ Solution

```
function App() {
  return (
    <>
    <BrowserRouter>
      <Routes>
        <Route path="/signup" element={<Signup />} />
        <Route path="/signin" element={<Signin />} />
        <Route path="/dashboard" element={<Dashboard />} />
        <Route path="/send" element={<SendMoney />} />
      </Routes>
    </BrowserRouter>
  </>
)
}
```

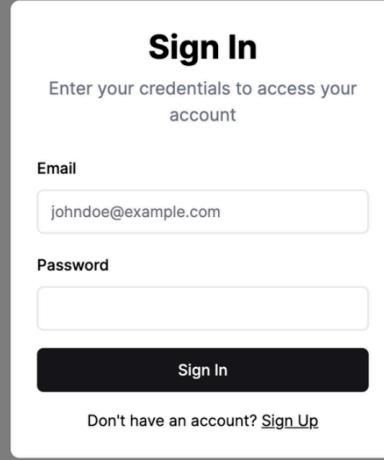
Step 2 - Create and hook up Signup page



If the user signup is successful, take the user to [/dashboard](#)

If not, show them an error message

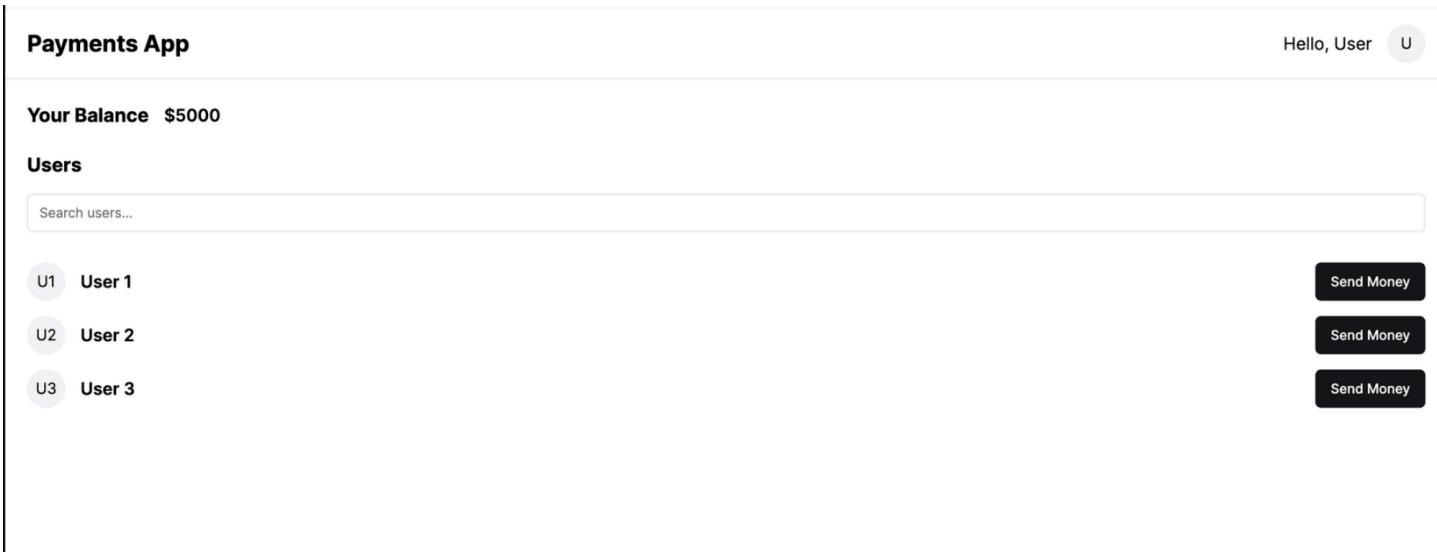
Step 3 - Create the signin page



A screenshot of a sign-in form titled "Sign In". The form instructs the user to "Enter your credentials to access your account". It has fields for "Email" (containing "johndoe@example.com") and "Password". A "Sign In" button is at the bottom, and a link "Don't have an account? [Sign Up](#)" is below it.

If the signin is successful, take the user to </dashboard>

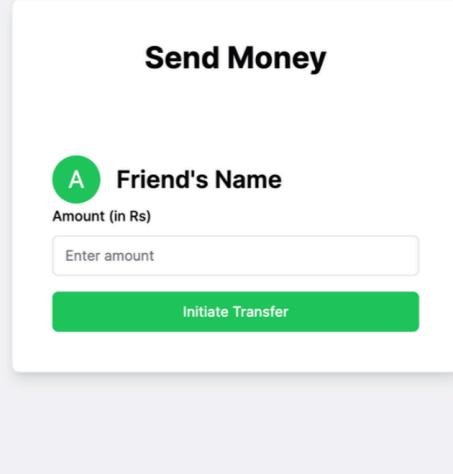
Step 4 - Dashboard page



A screenshot of a dashboard page for a "Payments App". At the top, it says "Hello, User" and shows a user icon. The main area shows "Your Balance \$5000". Below that is a "Users" section with a search bar containing "Search users...". Three users are listed: "User 1", "User 2", and "User 3", each with a "Send Money" button to their right.

Show the user their balance, and a list of users that exist in the database

Clicking on [Send money](#) should open a modal that lets the user send money

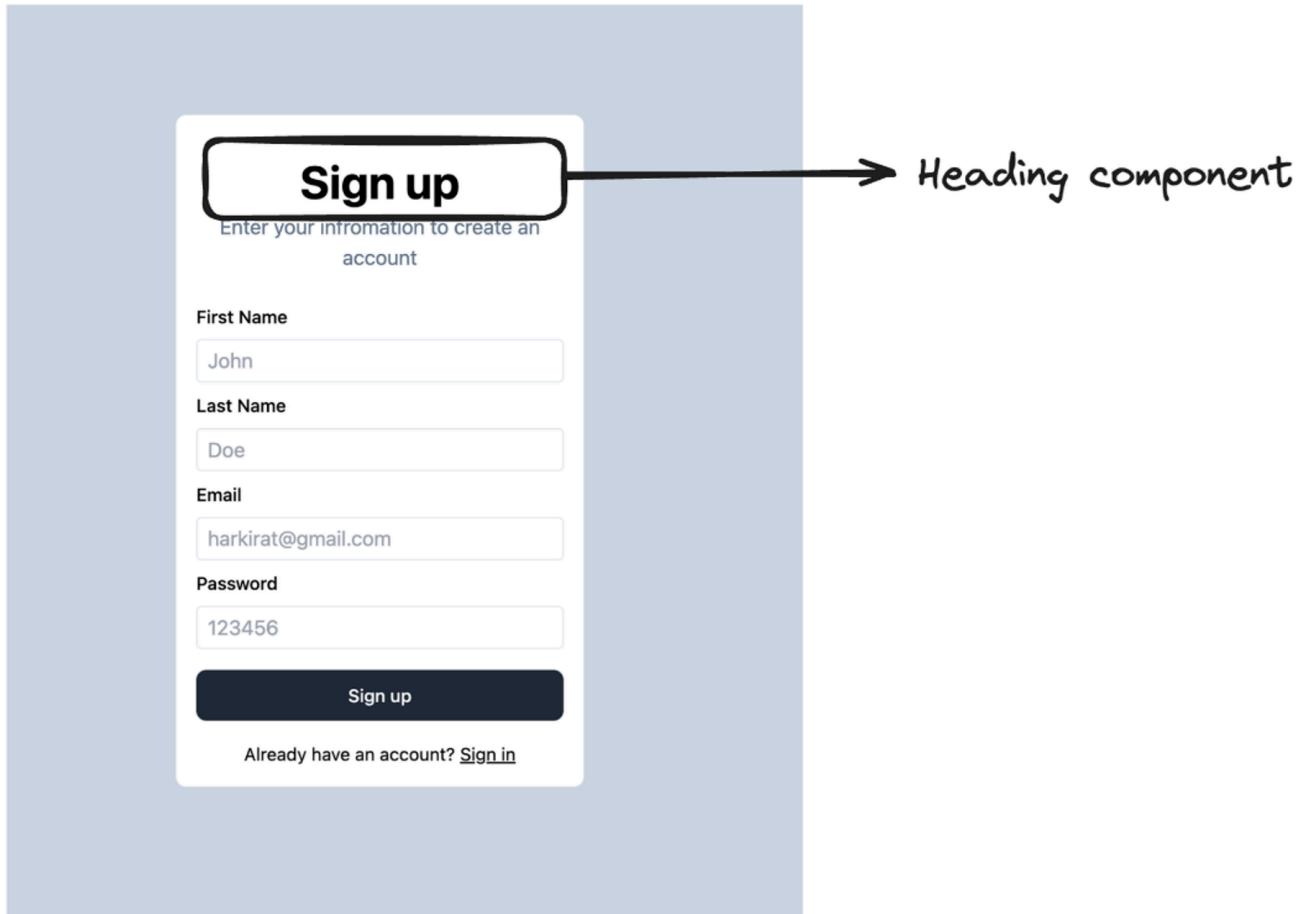


Step 5 - Auth Components

Full Signup component

You can break down the app into a bunch of components. The code only contains the styles of the component, not any onclick functionality.

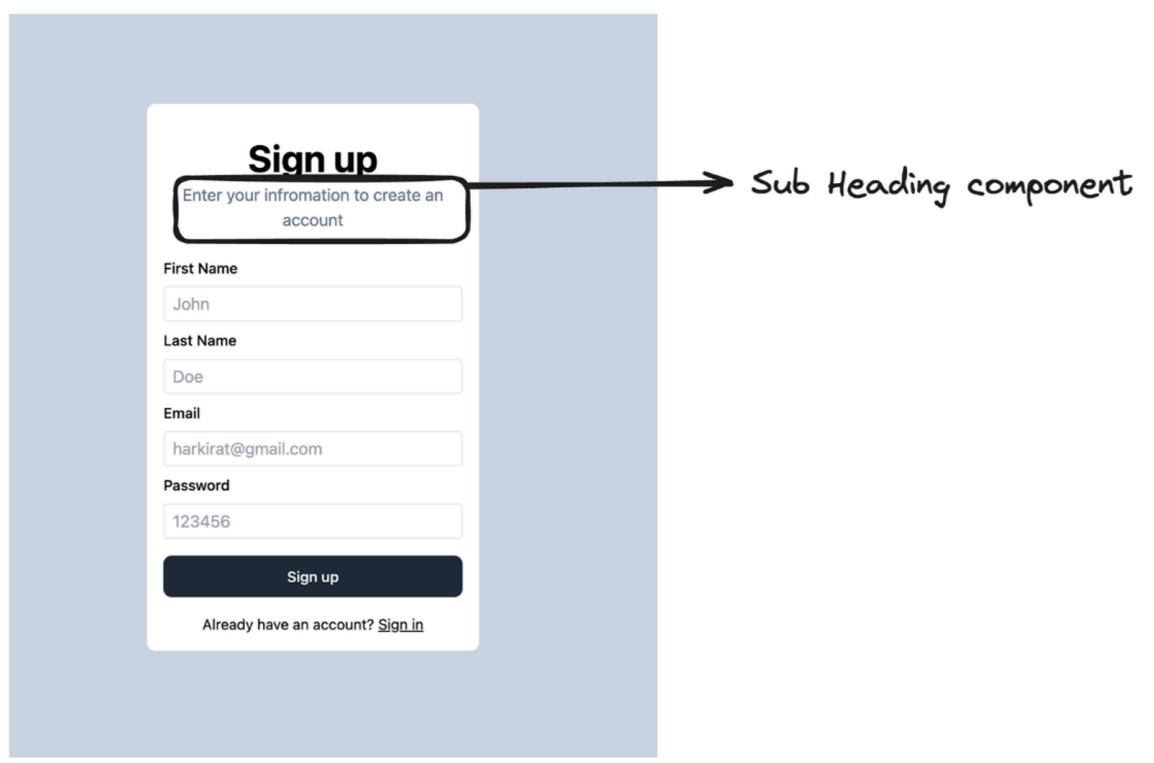
1. Heading component



▼ Code

```
export function Heading({label}) {  
  return <div className="font-bold text-4xl pt-6">  
    {label}  
  </div>  
}
```

2. Sub Heading component

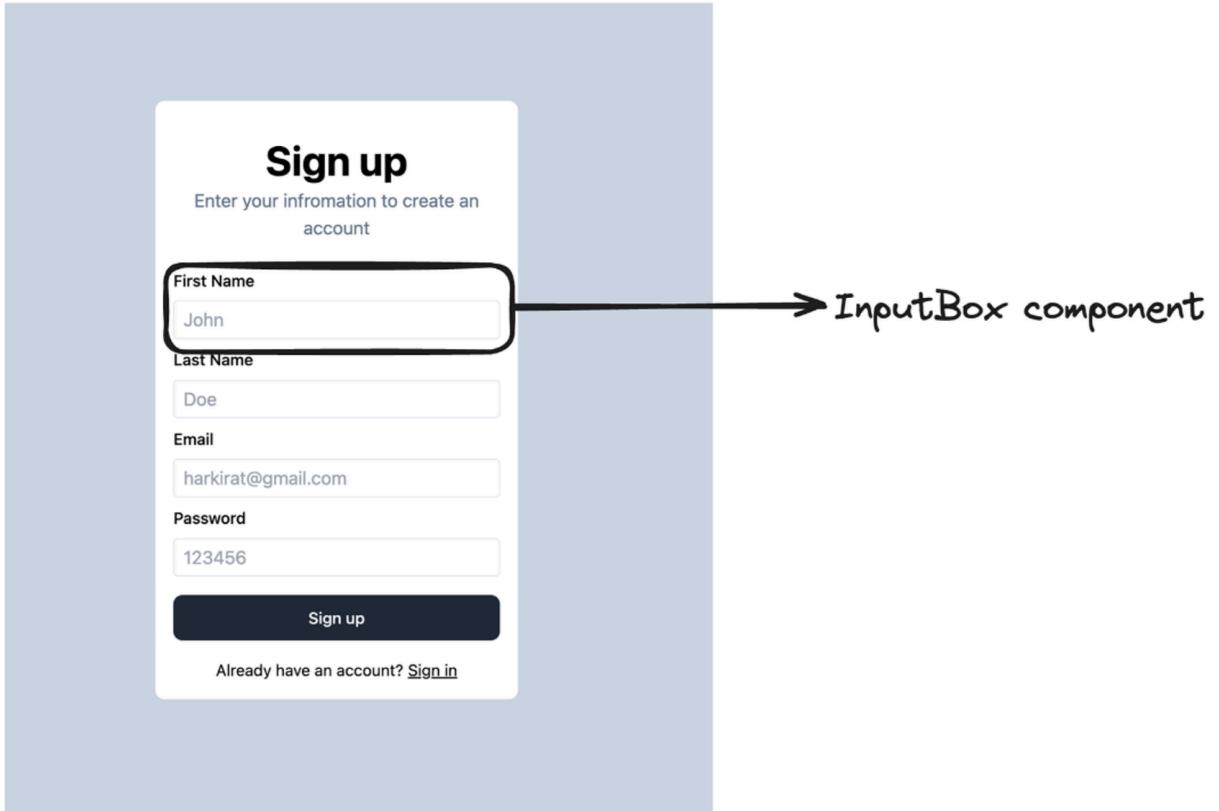


▼ Code

```
export function SubHeading({label}) {  
  return <div className="text-slate-500 text-md pt-1 px-4 pb-4">  
    {label}  
  </div>  
}
```

3. InputBox component

To move canvas, hold mouse wheel or spacebar while dragging, or use the hand tool

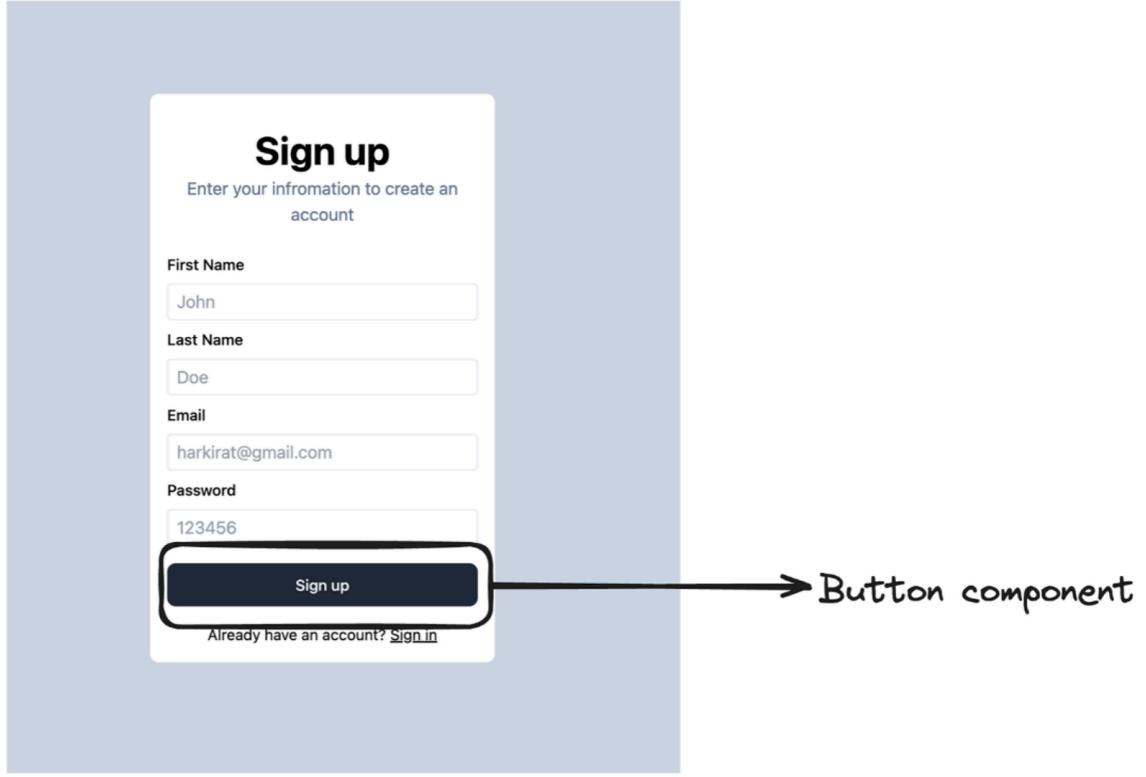


▼ Code

```
export function InputBox({label, placeholder}) {
  return <div>
    <div className="text-sm font-medium text-left py-2">
      {label}
    </div>
    <input placeholder={placeholder} className="w-full px-2 py-1 border rounded border-sla
  </div>
}
```



4. Button Component

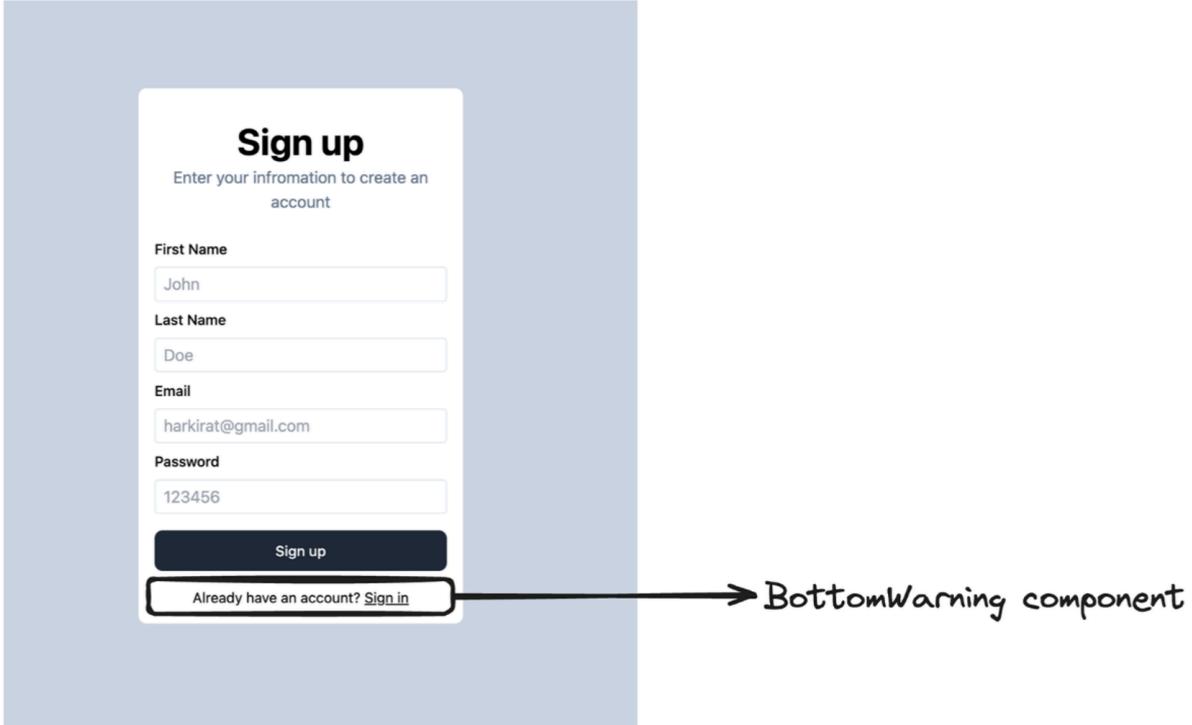


▼ Code

```
export function Button({label, onClick}) {
  return <button onClick={onClick} type="button" class=" w-full text-white bg-gray-800 hov
}
```

This section was blindly copied from <https://flowbite.com/docs/components/buttons/>

5. BottomWarning

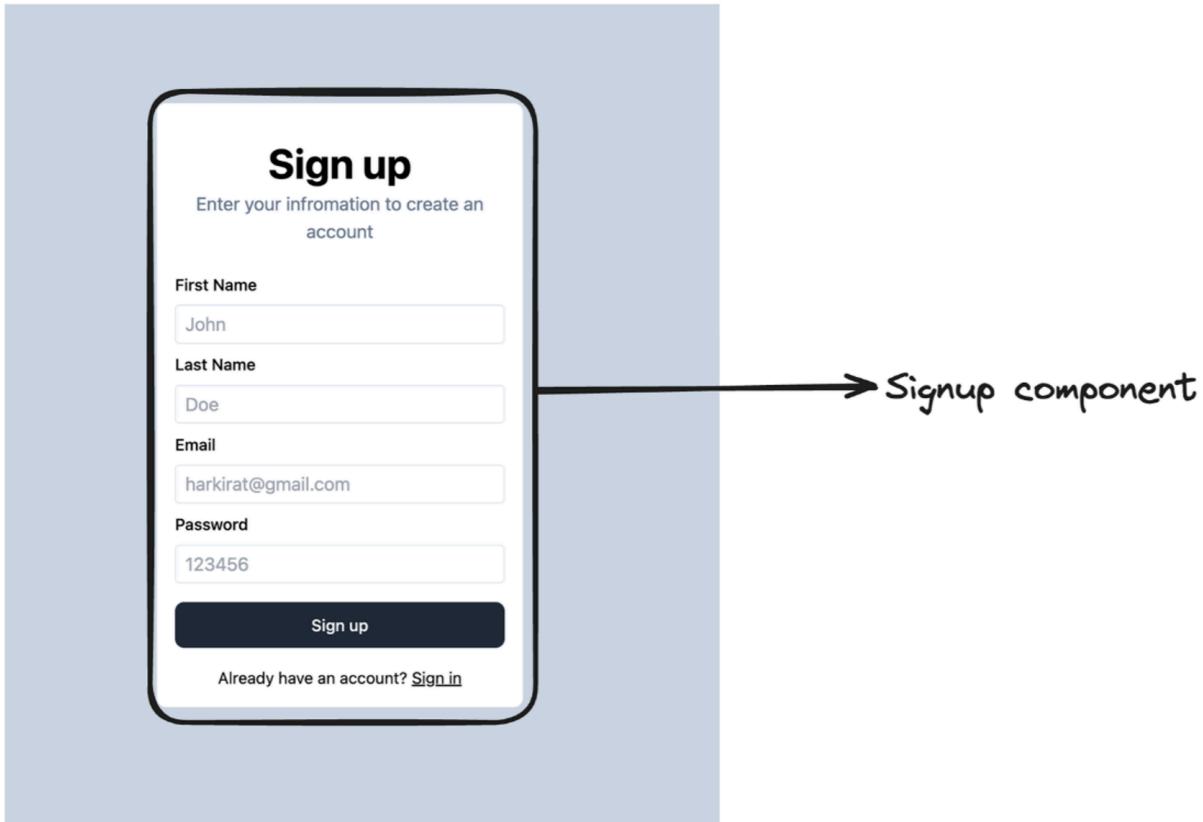


▼ Code

```
import { Link } from "react-router-dom"

export function BottomWarning({label, buttonText, to}) {
  return <div className="py-2 text-sm flex justify-center">
    <div>
      {label}
    </div>
    <Link className="pointer underline pl-1 cursor-pointer" to={to}>
      {buttonText}
    </Link>
  </div>
}
```

Full Signup component



▼ Code

```
import { BottomWarning } from "../components/BottomWarning"
import { Button } from "../components/Button"
import { Heading } from "../components/Heading"
import { InputBox } from "../components/InputBox"
import { SubHeading } from "../components/SubHeading"

export const Signup = () => {
  return <div className="bg-slate-300 h-screen flex justify-center">
    <div className="flex flex-col justify-center">
      <div className="rounded-lg bg-white w-80 text-center p-2 h-max px-4">
        <Heading label={"Sign up"} />
        <SubHeading label={"Enter your information to create an account"} />
        <InputBox placeholder="John" label={"First Name"} />
        <InputBox placeholder="Doe" label={"Last Name"} />
        <InputBox placeholder="harkirat@gmail.com" label={"Email"} />
        <InputBox placeholder="123456" label={"Password"} />
        <div className="pt-4">
          <Button label={"Sign up"} />
        </div>
        <BottomWarning label={"Already have an account?"} buttonText={"Sign in"} to={"/sign-in"} />
      </div>
    </div>
  </div>
```

```
</div>
}
```

Full Signin component

▼ Code

```
import { BottomWarning } from "../components/BottomWarning"
import { Button } from "../components/Button"
import { Heading } from "../components/Heading"
import { InputBox } from "../components/InputBox"
import { SubHeading } from "../components/SubHeading"

export const Signin = () => {

  return <div className="bg-slate-300 h-screen flex justify-center">
    <div className="flex flex-col justify-center">
      <div className="rounded-lg bg-white w-80 text-center p-2 h-max px-4">
        <Heading label={"Sign in"} />
        <SubHeading label={"Enter your credentials to access your account"} />
        <InputBox placeholder="harkirat@gmail.com" label={"Email"} />
        <InputBox placeholder="123456" label={"Password"} />
        <div className="pt-4">
          <Button label={"Sign in"} />
        </div>
        <BottomWarning label={"Don't have an account?"} buttonText={"Sign up"} to={"/signup"} />
      </div>
    </div>
  </div>
}

}
```

Step 6 - Signin-ed Comonents

1. Appbar



▼ Code

```
export const AppBar = () => {
  return <div className="shadow h-14 flex justify-between">
    <div className="flex flex-col justify-center h-full ml-4">
      PayTM App
    </div>
    <div className="flex">
      <div className="flex flex-col justify-center h-full mr-4">
        Hello
      </div>
      <div className="rounded-full h-12 w-12 bg-slate-200 flex justify-center mt-1 mr-2">
        <div className="flex flex-col justify-center h-full text-xl">
          U
        </div>
      </div>
    </div>
  </div>
}
```

2. Balance



▼ Code

```
export const Balance = ({ value }) => {
  return <div className="flex">
    <div className="font-bold text-lg">
      Your balance
    </div>
    <div className="font-semibold ml-4 text-lg">
      Rs {value}
    </div>
  </div>
}
```

3. Users component



▼ Code

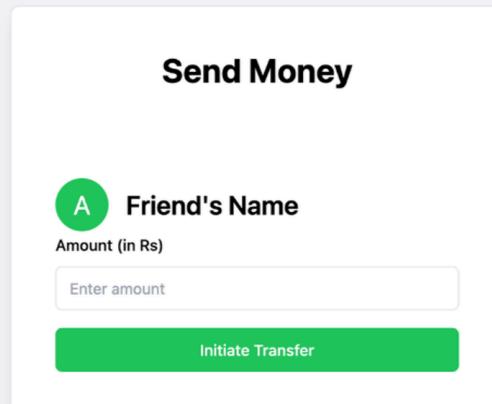
```
import { useState } from "react"
import { Button } from "./Button"

export const Users = () => {
    // Replace with backend call
    const [users, setUsers] = useState([
        {
            firstName: "Harkirat",
            lastName: "Singh",
            _id: 1
        }
    ]);

    return <>
        <div className="font-bold mt-6 text-lg">
            Users
        </div>
        <div className="my-2">
            <input type="text" placeholder="Search users..." className="w-full px-2 py-1 border rounded" />
        </div>
        <div>
            {users.map(user => <User user={user} />)}
        </div>
    </>
}

function User({user}) {
    return <div className="flex justify-between">
        <div className="flex">
            <div className="rounded-full h-12 w-12 bg-slate-200 flex justify-center mt-1 mr-2">
                <div className="flex flex-col justify-center h-full text-xl">
                    {user.firstName[0]}
                </div>
            </div>
            <div className="flex flex-col justify-center h-ful">
                <div>
                    {user.firstName} {user.lastName}
                </div>
            </div>
        </div>
        <div className="flex flex-col justify-center h-ful">
            <Button label={"Send Money"} />
        </div>
    </div>
}
```

4. SendMoney Component



▼ Code

```
export const SendMoney = () => {
  return <div class="flex justify-center h-screen bg-gray-100">
    <div className="h-full flex flex-col justify-center">
      <div
        class="border h-min text-card-foreground max-w-md p-4 space-y-8 w-96 bg-white">
        <div class="flex flex-col space-y-1.5 p-6">
          <h2 class="text-3xl font-bold text-center">Send Money</h2>
          </div>
          <div class="p-6">
            <div class="flex items-center space-x-4">
              <div class="w-12 h-12 rounded-full bg-green-500 flex items-center justify-center">
                <span class="text-2xl text-white">A</span>
              </div>
              <h3 class="text-2xl font-semibold">Friend's Name</h3>
            </div>
            <div class="space-y-4">
              <div class="space-y-2">
                <label
                  class="text-sm font-medium leading-none peer-disabled:cursor-not-allowed"
                  for="amount">
                  Amount (in Rs)
                </label>
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>
```

```
</label>
<input
    type="number"
    class="flex h-10 w-full rounded-md border border-input bg-background
    id="amount"
    placeholder="Enter amount"
/>
</div>
<button class="justify-center rounded-md text-sm font-medium ring-offset
    Initiate Transfer
</button>
</div>
</div>
</div>
</div>
</div>
</div>
}
}
```

Step 7 - Wiring up the backend calls

You can use

1. fetch or
2. axios

to wire up calls to the backend server.

The final code looks something like this -

<https://github.com/100xdevs-cohort-2/paytm/tree/complete-solution> (complete-solution branch on the repo)

The important bits here are -

1. Signup call - <https://github.com/100xdevs-cohort-2/paytm/blob/complete-solution/frontend/src/pages/Signup.jsx#L36>
2. Call to get all the users given the filter - <https://github.com/100xdevs-cohort-2/paytm/blob/complete-solution/frontend/src/components/Users.jsx#L13>
3. Call to transfer money b/w accounts - <https://github.com/100xdevs-cohort-2/paytm/blob/complete-solution/frontend/src/pages/SendMoney.jsx#L45>