



## 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

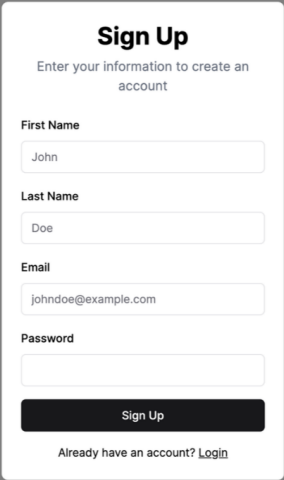
Import `react-router-dom` into your project and add the following routes -

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



**Sign Up**  
Enter your information to create an account

First Name

Last Name

Email

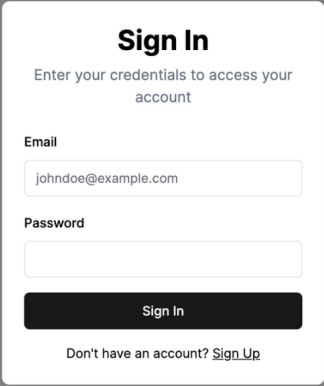
Password

Already have an account? [Login](#)

If the user signup is successful, take the user to `/dashboard`

If not, show them an error message

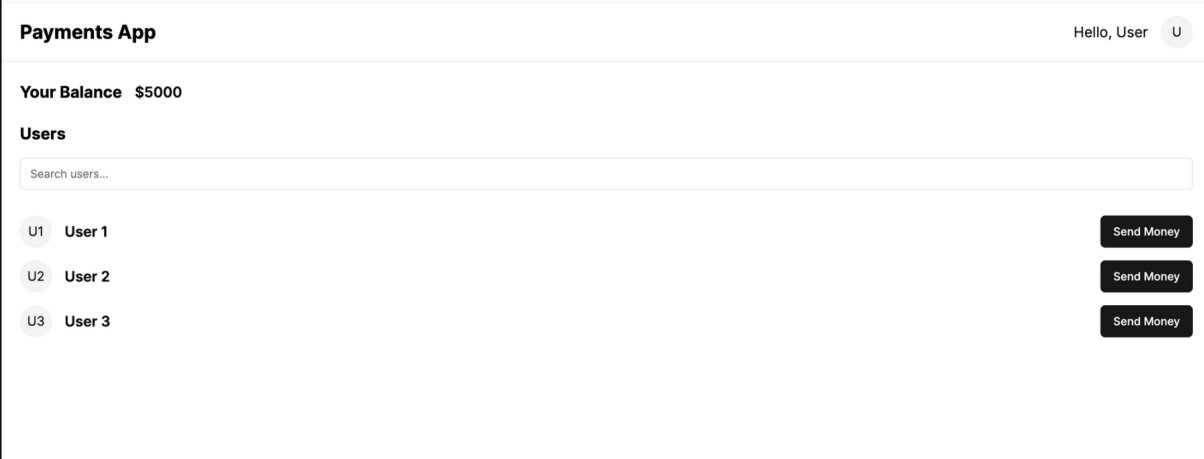
## Step 3 - Create the signin page



The image shows a 'Sign In' form centered on a dark gray background. The form is white and contains the following elements: a title 'Sign In', a subtitle 'Enter your credentials to access your account', an 'Email' label above a text input field containing 'johndoe@example.com', a 'Password' label above an empty password input field, a 'Sign In' button, and a link 'Don't have an account? Sign Up'.

If the signin is successful, take the user to `/dashboard`

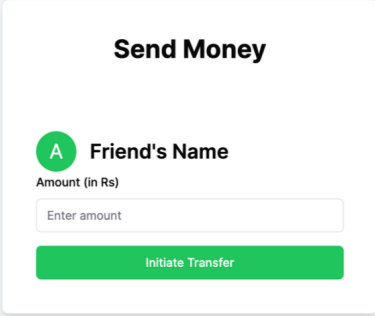
## Step 4 - Dashboard page



The image shows a dashboard page for a 'Payments App'. At the top left is the app name 'Payments App'. At the top right is a greeting 'Hello, User' followed by a user icon labeled 'U'. Below the app name, it says 'Your Balance \$5000'. Underneath, there is a section titled 'Users' with a search bar labeled 'Search users...'. Below the search bar, there is a list of three users: 'U1 User 1', 'U2 User 2', and 'U3 User 3'. To the right of each user name is a 'Send Money' button.

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

A mockup of a 'Send Money' form. The form is a white card with rounded corners and a subtle shadow, centered on a light gray background. At the top, the title 'Send Money' is displayed in a bold, black font. Below the title, there is a green circular profile picture containing the letter 'A', followed by the text 'Friend's Name' in a bold font. Underneath, the label 'Amount (in Rs)' is shown in a smaller font. A text input field with the placeholder 'Enter amount' is positioned below the label. At the bottom of the form, there is a prominent green button with the text 'Initiate Transfer' in white.

## 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

**Sign up**

Enter your information to create an account

First Name  
John

Last Name  
Doe

Email  
harkirat@gmail.com

Password  
123456

Sign up

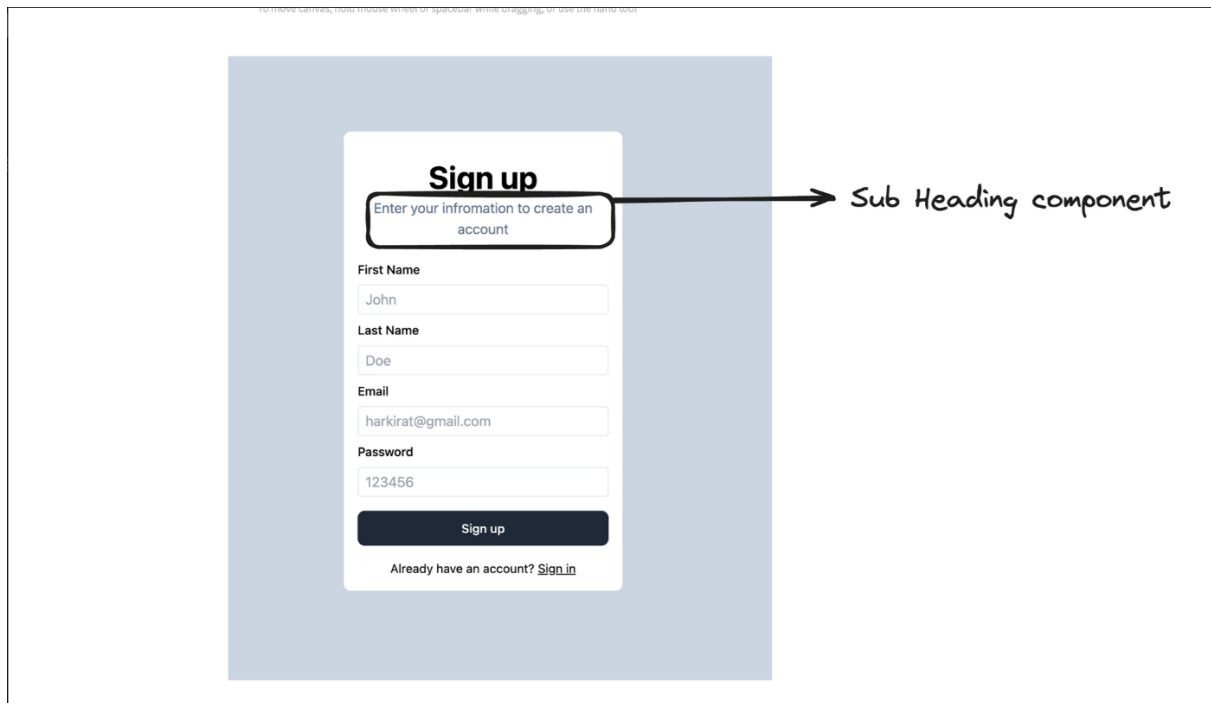
Already have an account? [Sign in](#)

Heading component

## ▼ Code

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

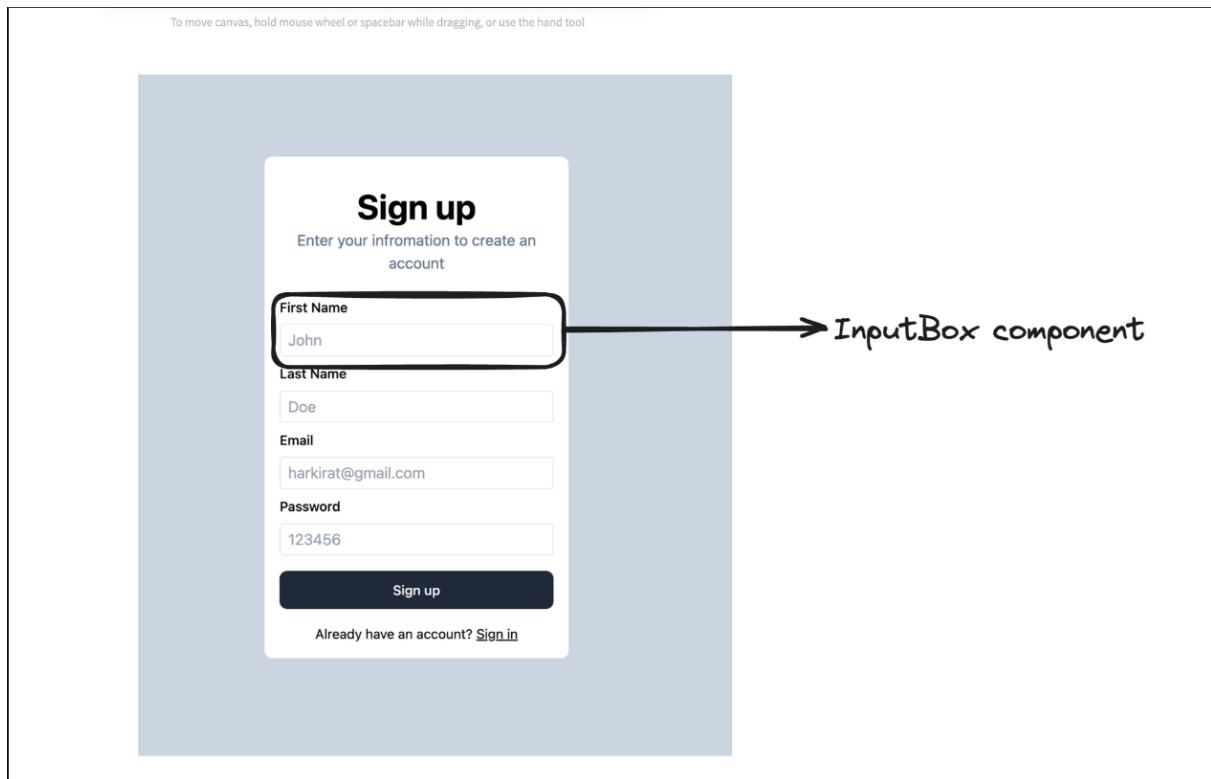
## 2. Sub Heading component



## ▼ Code

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

## 3. InputBox component

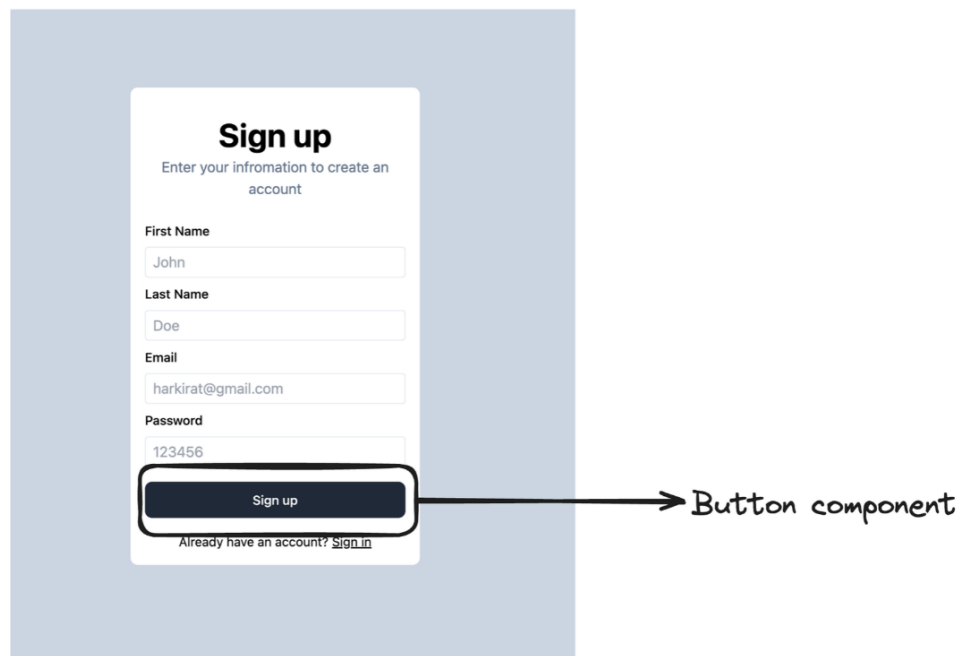


## ▼ Code

```
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-slate-200" />  
  </div>  
}
```

## 4. Button Component





## ▼ Code

```
export function Button({label, onClick}) {  
  return <button onClick={onClick} type="button" class  
    =" w-full text-white bg-gray-800 hover:bg-gray-900 focu  
    s:outline-none focus:ring-4 focus:ring-gray-300 font-med  
    ium rounded-lg text-sm px-5 py-2.5 me-2 mb-2">{label}</b  
    utton>  
}
```

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

## 5. BottomWarning

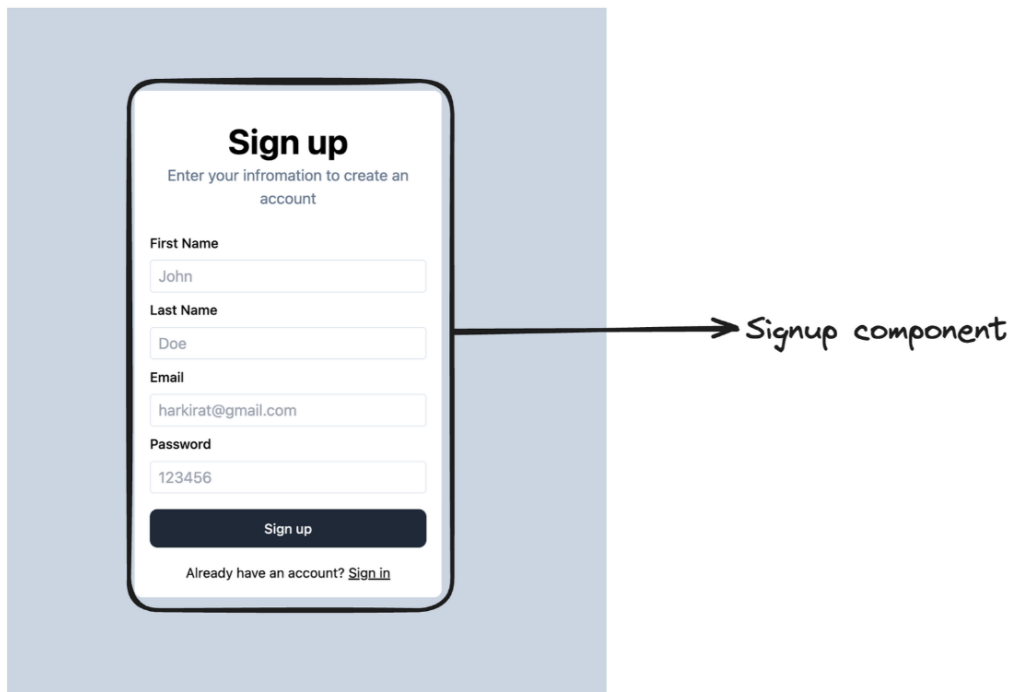


## ▼ Code

```
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= {"Passwor
d"} />
      <div className="pt-4">
        <Button label= {"Sign up"} />
      </div>
      <BottomWarning label= {"Already have an accoun
t?"} buttonText= {"Sign in"} to= {" /signin"} />
    </div>
  </div>
</div>
}

```

## Full Signin component

### ▼ Code

```

import { BottomWarning } from "../components/BottomWarni
ng"
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 ju
stify-center">
    <div className="flex flex-col justify-center">
      <div className="rounded-lg bg-white w-80 text-cent
er p-2 h-max px-4">
        <Heading label= {"Sign in"} />
        <SubHeading label= {"Enter your credentials to ac
cess your account"} />
        <InputBox placeholder="harkirat@gmail.com" label

```

```

    ={"Email"} />
    <InputBox placeholder="123456" label={"Passwor
d"} />
    <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-betw
een">
    <div className="flex flex-col justify-center h-f

```

```

    ull 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-sl
ate-200 flex justify-center mt-1 mr-2">
        <div className="flex flex-col justify-ce
nter h-full text-xl">
          U
        </div>
      </div>
    </div>
  </div>
</div>
}

```

## 2. Balance



### ▼ Code

```

export const Balance = ({ value }) => {
  return <div className="flex">
    <div className="font-bold text-lg">

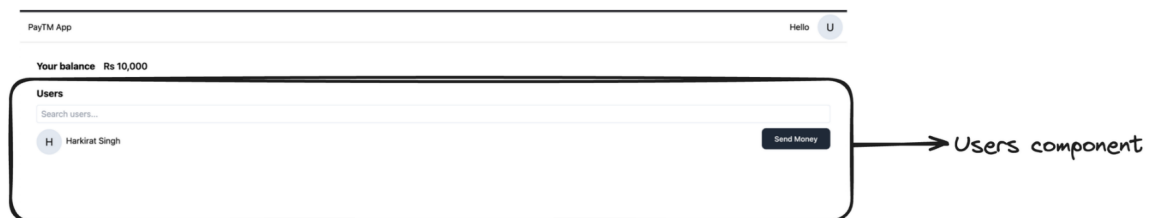
```

```

        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>
    <div className="my-2">
      <input type="text" placeholder="Search user
s..." className="w-full px-2 py-1 border rounded border-
slate-200"></input>
    </div>
    <div>
      {users.map(user => <User user={user} />)}
    </div>
  </>
}

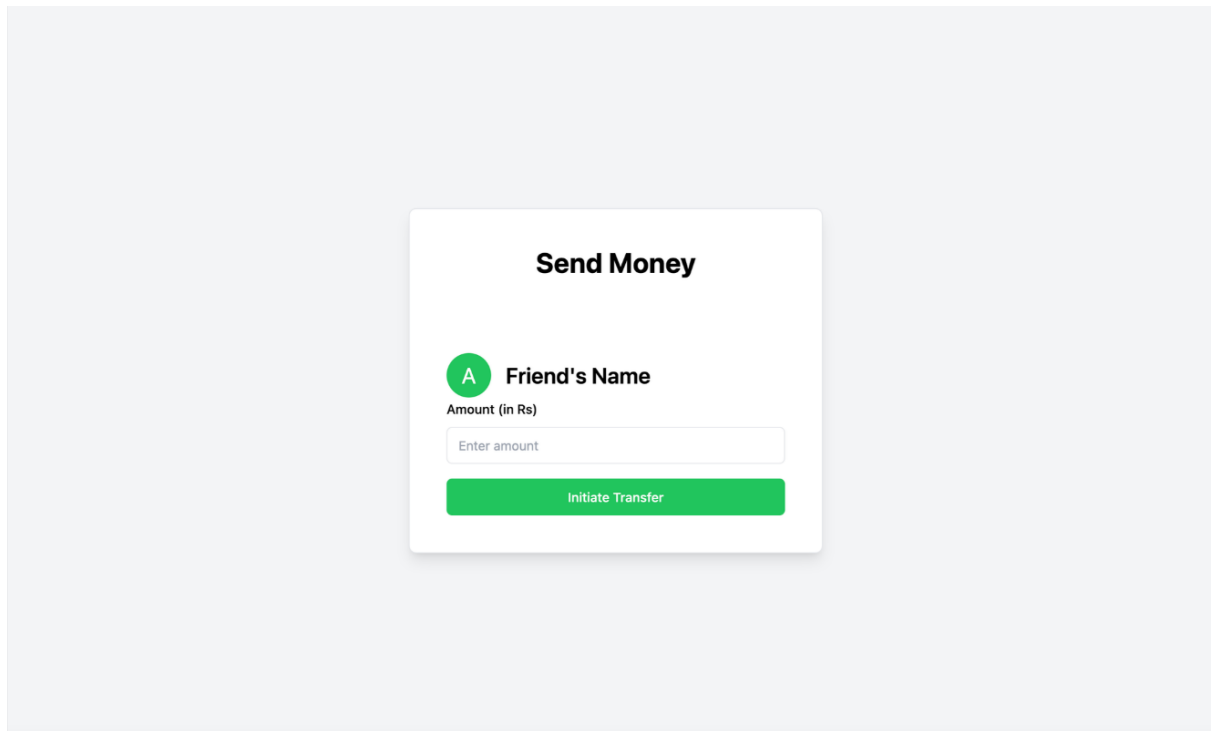
function User({user}) {
  return <div className="flex justify-between">
    <div className="flex">
      <div className="rounded-full h-12 w-12 bg-sl
ate-200 flex justify-center mt-1 mr-2">
        <div className="flex flex-col justify-ce
nter h-full text-xl">
          {user.firstName[0]}
        </div>
      </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-f
ul">
      <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 shadow-lg rounded-lg"
      >
        <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 peer-disabled:opacity-70"
                for="amount"
            >
                Amount (in Rs)
            </label>
            <input
                type="number"
                class="flex h-10 w-full rounded-md border border-input bg-background px-3 py-2 text-sm"
                id="amount"
                placeholder="Enter amount"
            />
        </div>
        <button class="justify-center rounded-md text-sm font-medium ring-offset-background transition-colors h-10 px-4 py-2 w-full bg-green-500 text-white">
            Initiate Transfer
        </button>
    </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>