

```
'use client';
```

```
import { Button } from '@shared/components/ui/Button';
```

```
import Link from 'next/link';
```

```
import { signInWithPassword } from '@shared/utils/auth-helpers/server';
```

```
import { handleRequest } from '@shared/utils/auth-helpers/client';
```

```
import { useRouter } from 'next/navigation';
```

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

```
// Define prop type with allowEmail boolean
```

```
interface PasswordSignInProps {
```

```
  allowEmail: boolean;
```

```
  redirectMethod: string;
```

```
}
```

```
export default function PasswordSignIn({
```

```
  allowEmail,
```

```
  redirectMethod,
```

```
}: PasswordSignInProps) {
```

```
  const router = redirectMethod === 'client' ? useRouter() : null;
```

```
  const [isSubmitting, setIsSubmitting] = useState(false);
```

```
  const handleSubmit = async (e: React.FormEvent<HTMLFormElement>) => {
```

```
    setIsSubmitting(true); // Disable the button while the request is being handled
```

```
    await handleRequest(e, signInWithPassword, router);
```

```
    setIsSubmitting(false);
```

```
};
```

```
return (
```

```
<div className="my-8">
```

```
<form
```

```
  noValidate={true}
```

```
  className="mb-4"
```

```
  onSubmit={(e) => handleSubmit(e)}
```

```
>
```

```
<div className="grid gap-2">
```

```
  <div className="grid gap-1">
```

```
    <label htmlFor="email">Email</label>
```

```
    <input
```

```
      id="email"
```

```
      placeholder="name@example.com"
```

```
      type="email"
```

```
      name="email"
```

```
      autoCapitalize="none"
```

```
      autoComplete="email"
```

```
      autoCorrect="off"
```

```
      className="w-full p-3 rounded-md bg-zinc-800 text-white"
```

```
    />
```

```
    <label htmlFor="password">Password</label>
```

```
    <input
```

```
      id="password"
```

```
      placeholder="Password"
```

type="password"

name="password"

autoComplete="current-password"

className="w-full p-3 rounded-md bg-zinc-800 text-white"

/>

</div>

<Button

variant="outline"

type="submit"

className="mt-1"

loading={isSubmitting}

>

Sign in

</Button>

</div>

</form>

<p>

<Link href="/signin/forgot_password" className="font-light text-sm">

Forgot your password?

</Link>

</p>

{allowEmail && (

<p>

<Link href="/signin/email_signin" className="font-light text-sm">

Sign in via magic link

</Link>

</p>

}}

<p>

<Link href="/signin/signup" className="font-light text-sm">

Don't have an account? Sign up

</Link>

</p>

</div>

);

}