


Lucia-auth

New (better?) auth library

About me

Stanisław Synowiec

Currently: Student | IT & Event Technician

 /ssynowiec

 /ssynowiecpl

 /ssynowiec1410



Popularity

↓ Weekly Downloads

21 208



<https://www.npmjs.com/package/lucia> 29.02.2024

Lucia

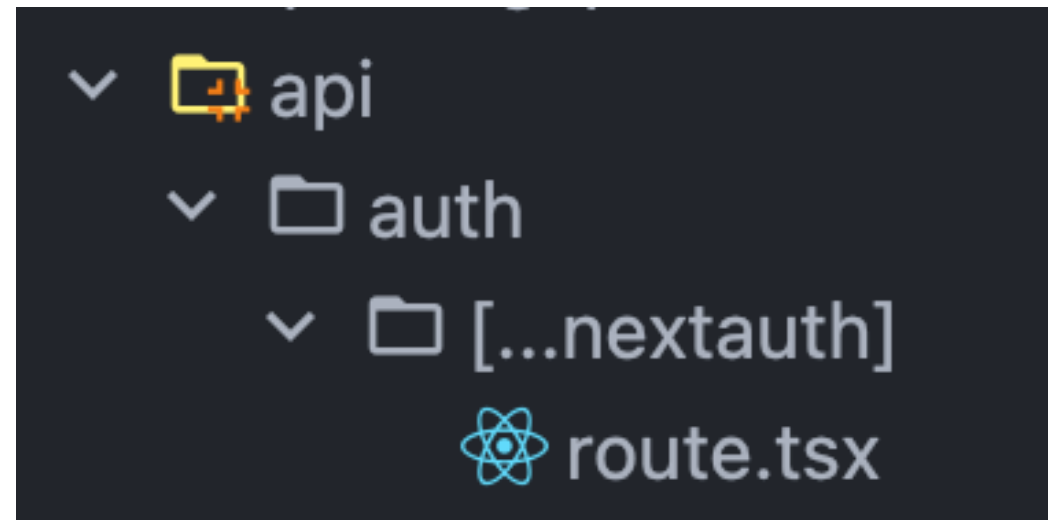
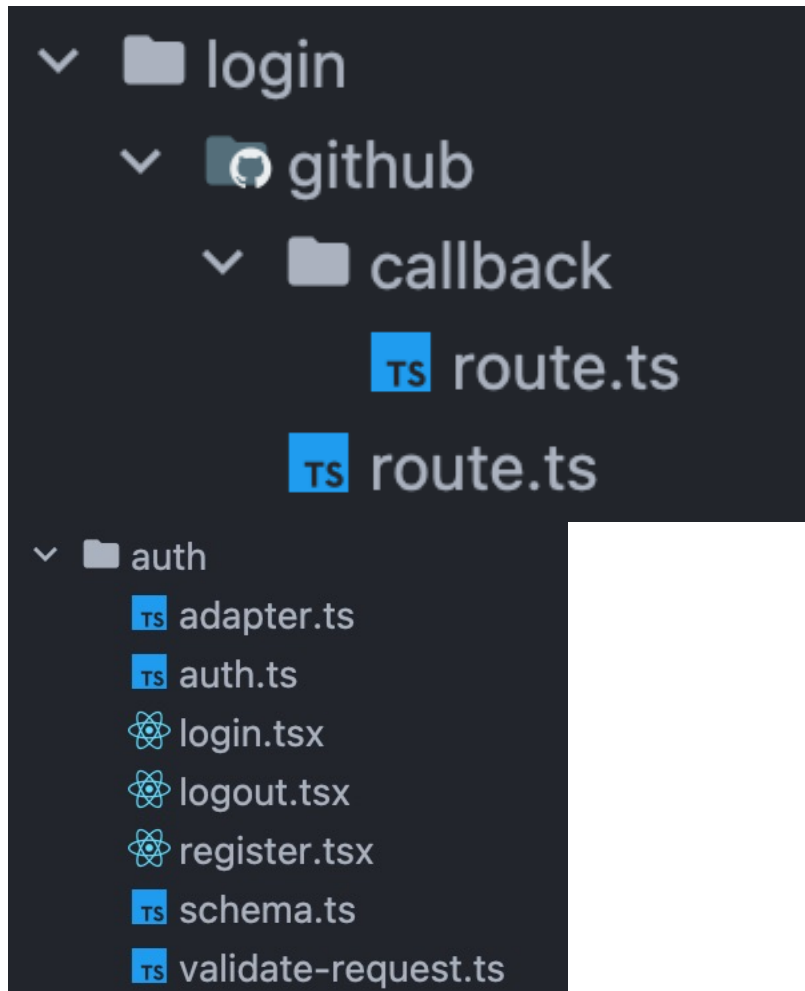
- ✓ TypeScript
- OAuth (GitHub, Google, Facebook, etc.)
(3rd part - arctic)
- ✓ username and password
- ✗ context provider
- ✓ database adapters
- plug-and-play
- ✗ middleware

VS

NextAuth.js

- ✓ TypeScript
- ✓ OAuth (GitHub, Google, Facebook, etc.)
(built-in)
- username and password
- ✓ context provider
- ✓ database adapters
- ✓ plug-and-play
- ✓ middleware

Config files





pilcrowOnPaper on Oct 29, 2023

Maintainer

edited ▾ ⋮

Passwords

Next, password based auth. Unlike Auth.js, I don't want to make it harder to implement it. But I do want to discourage people from implementing it if they're not willing to put the care required to pull it off. Because here's the thing: email/password auth is tedious. It's not *hard*, but there's a lot of steps required to implement it correctly. **And you're not making your password based auth any more secure by using Lucia's keys.** Besides, if you're willing to take time to implement email verification, password reset tokens, and login throttling, keys aren't really making your work significantly easier.

<https://github.com/lucia-auth/lucia/discussions/1231>

app/layout.tsx

```
const metadata: Metadata = {
  title: 'Lucia auth demo',
  description: 'Lucia auth demo app',
};

const RootLayout = ({
  children,
}: Readonly<{
  children: ReactNode;
}>) => {
  return (
    <html lang='en' className={` ${GeistSans.variable} ${GeistMono.variable}`}>
      <body className='flex min-h-screen flex-col'>
        <ThemeProvider
          attribute='class'
          defaultTheme='system'
          enableSystem={true}
          disableTransitionOnChange={true}
        >
          <Nav />
          <main className='flex flex-1'>{children}</main>
          <Footer />
        </ThemeProvider>
      </body>
    </html>
  );
};

export default RootLayout;
```

app/layout.tsx

```
const metadata: Metadata = {
  title: 'Lucia auth demo',
  description: 'Lucia auth demo app',
};

const RootLayout = ({
  children,
}: Readonly<{
  children: ReactNode;
}>) => {
  return (
    <html lang='en' className={` ${GeistSans.variable} ${GeistMono.variable}`}>
      <body className='flex min-h-screen flex-col'>
        <ThemeProvider
          attribute='class'
          defaultTheme='system'
          enableSystem={true}
          disableTransitionOnChange={true}
        >
          <SessionProvider>
            <Nav />
            <main className='flex flex-1'>{children}</main>
            <Footer />
          </SessionProvider>
        </ThemeProvider>
      </body>
    </html>
  );
};

export default RootLayout;
```

Logout

```
import { logout } from '@auth/logout';  
  
<form action={logout}>  
  <Button variant='link'>Logout</Button>  
</form>
```

```
import { signOut } from "next-auth/react"  
  
<button onClick={() => signOut()}>Sign out</button>
```




**Presentation
+ all my links**

Code:

Demo: <https://lucia-auth-demo.vercel.app>

Git-repo: <https://github.com/ssynowiec/lucia-demo>

Sources:

<https://lucia-auth.com>

<https://next-auth.js.org>

THANK YOU!