NextAuth:

`npm i next-auth`

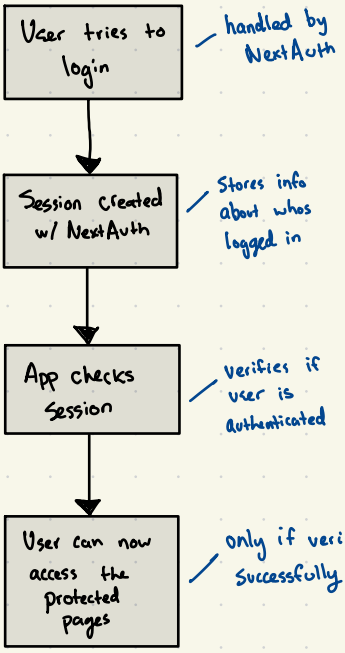
NEXT_PUBLIC_CLIENT_ID
↑ visible to browser and server

JWT_SECRET = ~~~
↑ used to encrypt data, protects the secret.

- Authentication is a process that verifies the user of the software.
- Common features:
  - Sign up
  - Sign in / login
  - password reset
  - sign out / logout
  - Social login (google, github, etc.)

- NextAuth handles complex authentication logic and prevents you from creating a complex system from scratch (which is very risky)
  - password hashing
  - session mgmt
  - OAuth flows
  - DB connections
  - token mgmt
  - email verification

`import { signIn, signOut, useSession } from 'next-auth/react'`



- Once logged in, the API provides an access token which is needed to use the providers features.
- Access tokens expire, but a refresh token can be used to extend the sessions length

- The 'api' route is where all authentication logic happens

`app/api/auth/[...nextAuth]/route.ts`

a dynamic route that handles:
- app/api/auth/signin
- /signout
- /callback/google
- /session

Creating API Route

```
import NextAuth from 'next-auth'
import GoogleProvider from 'next-auth/providers/google'

const handler = NextAuth({
  providers: [
    GoogleProvider({
      clientId: process.env.GOOGLE_CLIENT_ID!,
      clientSecret: process.env.GOOGLE_CLIENT_SECRET!,
    })
  ],
})

export { handler as GET, handler as POST }
```