README

This is a simple login system using: Nodejs, JWT auth, Expess and React.

follow the instructions below:

1. Before we start, we need to install the following libraries with npm, yarn , whatever we choose:

Run with npm:

- npm init -y

- npm install express jsonwebtoken

- nodemon index.js

2. Creating the app with Express.

2. Listen port (5000 for example).

3. Creating the JSON file containing all the users data.

- username: Varchar

- password: Varchar

- isAdmin: takes true/false indicating whether the user is an admin or not

5. Login Route { if the obtained infos are valid =⇒ Grant privileges (return users infos using express.json (if we don't use it, we cannot send nor receive json through express)}

   else{ Error Message}.

In our login system this is done using the JWT.

> 💡 What is Token?

Tokens are pieces of data that carry just enough information to facilitate the process of determining a user's identity or authorizing a user to perform an action

> 💡 What is an access Token ?

When a user logins in, the authorization server issues an <u>access token</u>, which is an artifact that client applications can use to make secure calls to an API server.

> 💡 What is a refresh Token?

for security purposes, access tokens may be valid for a short amount of time. Once they expire, client applications can use a refresh token to "refresh" the access token. That is, a <u>refresh token</u> is a credential artifact that lets a client application get new access tokens without having to ask the user to log in again.

> 💡 **for more info**s check out this article: <u>https://auth0.com/blog/refresh-tokens-what-are-they-and-when-to-use-them/</u>

So in short: Our system creates an access token at first, then when it expires, the system takes the refresh token to create another access token :)

6. Delete Route: the system checks if the user is authenticated by checking the access token, then it checks if {the user isAdmin= =True ⇒ ok } else {error}.

7. getId route: this function returns the id of the user authenticated, i only use this function to see if the authentication process if valid.

Client REACT

1- install the necessary packages with npm:

npm install -g create-react-app

2- set the proxy to "http://<u>localhost</u>:5000/api/" (our api port :))

3- in this directory, we can run:

npm start/ yarn start, open <u>localhost:3000</u> in your browser