

# Lab 4 Samuel Ahlvin

Github link: <https://github.com/SamuelAhlvin/Lab-4>

I think I successfully solved the tasks in this lab for Grade 4. All the requirements from the description are fulfilled, and I have filled the right columns in the table.

It uses a real database with a table called “users”. You can successfully login with the right login info and if you enter incorrect or no info in either input, you’re shown an error message. When you log in, you’re redirected to the start page and then you can change the path to what you want to access. But only the correct users can access the correct pages thanks to the authentication token, otherwise the user is sent either to the identity/login page or shown a “Unauthorized” message. My “identity” file is called login.ejs.

The hardest part for me in this lab was to work with the jwt.sign and verify, because I thought it was a little bit tricky to learn how to store the token etc. After reading your update announcement It was a little bit clearer and I managed to solve it.

Another hard part was to then authenticate the user and make sure so that only the correct user can access the respective files. But I think that it was educational to learn about this concept.

After all these labs I’m much more comfortable working with JavaScript, the frontend course has helped as well in achieving this. I think it’s fun to work with servers and handling all the user info with a real database. It’s a good layout for the labs to progressively learn about concepts and then using the previous labs to build upon.

All in all this lab was interesting and as with the other labs I think this knowledge will be useful in the real world later on. The most interesting part was making the user only be able to access the correct files of the website.

Task	Grade 3	Grade 4	Grade 5
Create git repository for the lab	✓	✓	✓
Have incremental commits in repository	✓	✓	✓
Create a database with a table named “Users”	✓	✓	✓
Add 4 users to the table	✓	✓	✓
Create the path /admin	✓	✓	✓
Add Roles for users and admin	✓	✓	✓
Add roles for student 1, student 2, teacher		✓	✓
Create path /student1, /student2, /teacher		✓	✓
Render list of users in admin.ejs			✓
Register new users with bcrypt encryption			
Create dynamic route for user profiles			
Add readme.md file			