Report 1

-Testing passes and status code confirms everything is ok

-Testing passes and status code confirms everything is ok.

-No, this is very insecure behavior. I would recommend introducing an input file and traversing through it so that strings can be compared and exceptions can be thrown in such cases and users can be prompted to employ mor secure options for username and password.

Report 2

-all the access token does is authenticate that you have access to the API why we need the refresh token is because we can obtain additional access tokens with it. This means we do not have to ask the user to log in every time.

-No it does not, just like stated earlier we stated that the refresh token allows us to obtain additional tokens. This is after it is confirmed the user has access to the API. So the access token does not go back to the authorization server like the refresh token, but rather the access token goes to the resource server, meaning it can not renew access (give a new refresh token)

Report 3

-Testing has shown that it uses the access token to confirm which user is sending the request. With the help of the refresh token It seems secure as we’ve established earlier.

Learning challenge 2

1. all the appropriate imports are seen

2. We output the website title as a means to confirm it

3. Selenium then clicks the Sign In button on the left by identifying it with its link text

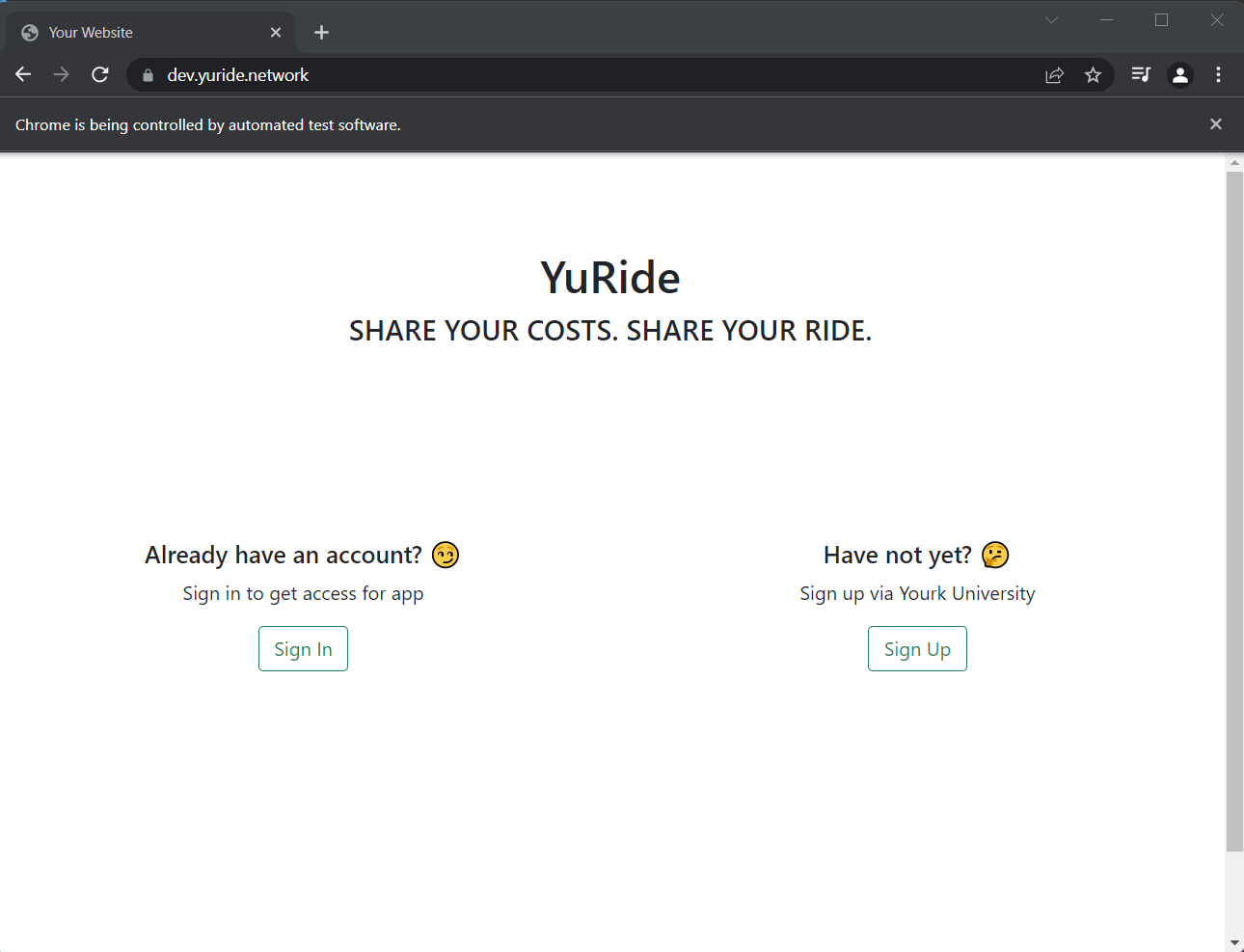
4. we wait until Passport York Login page is loaded this is confirmed by the username textbox

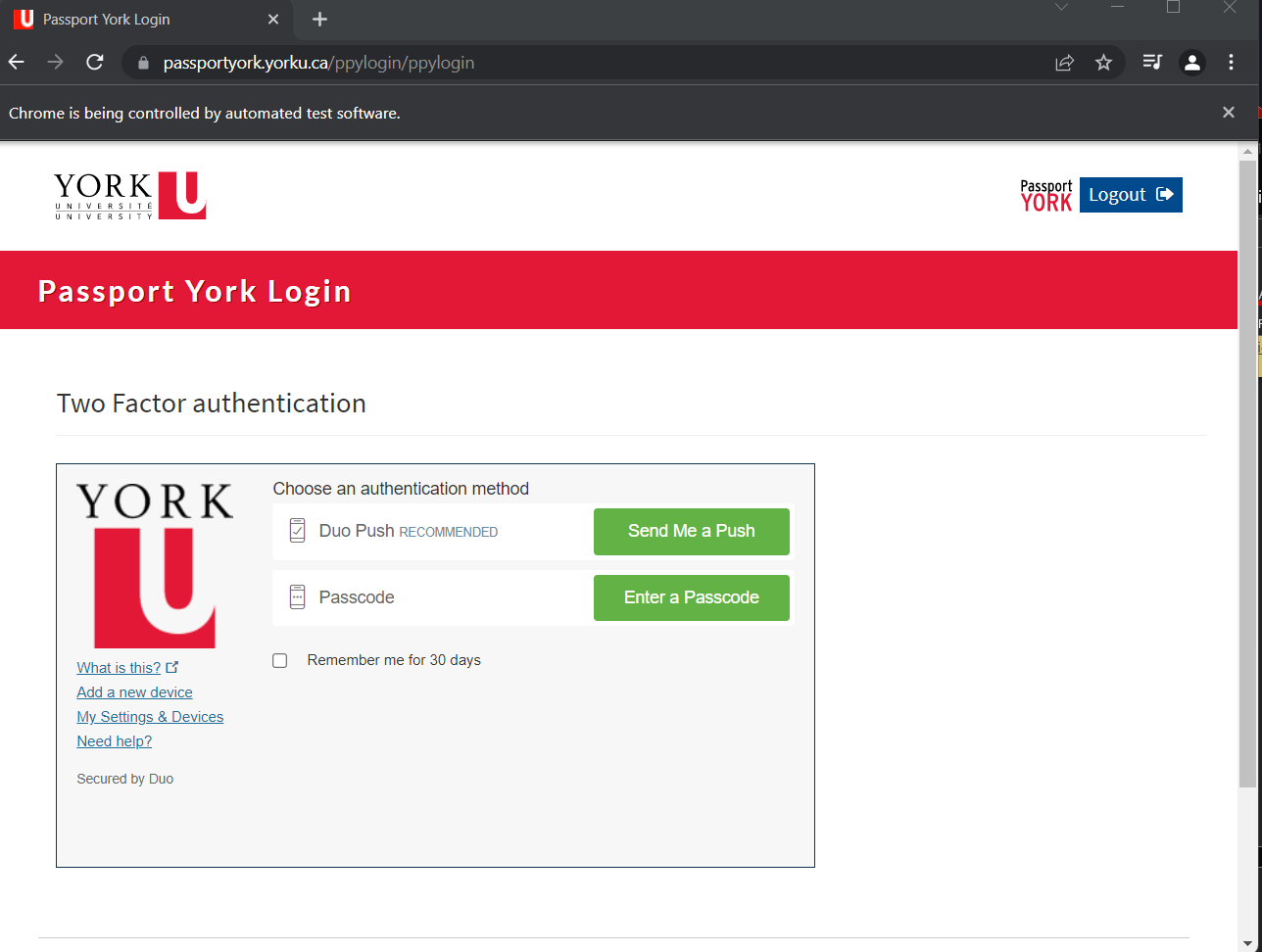
5. Selenium is then used to enter y own Passport York credentials and login

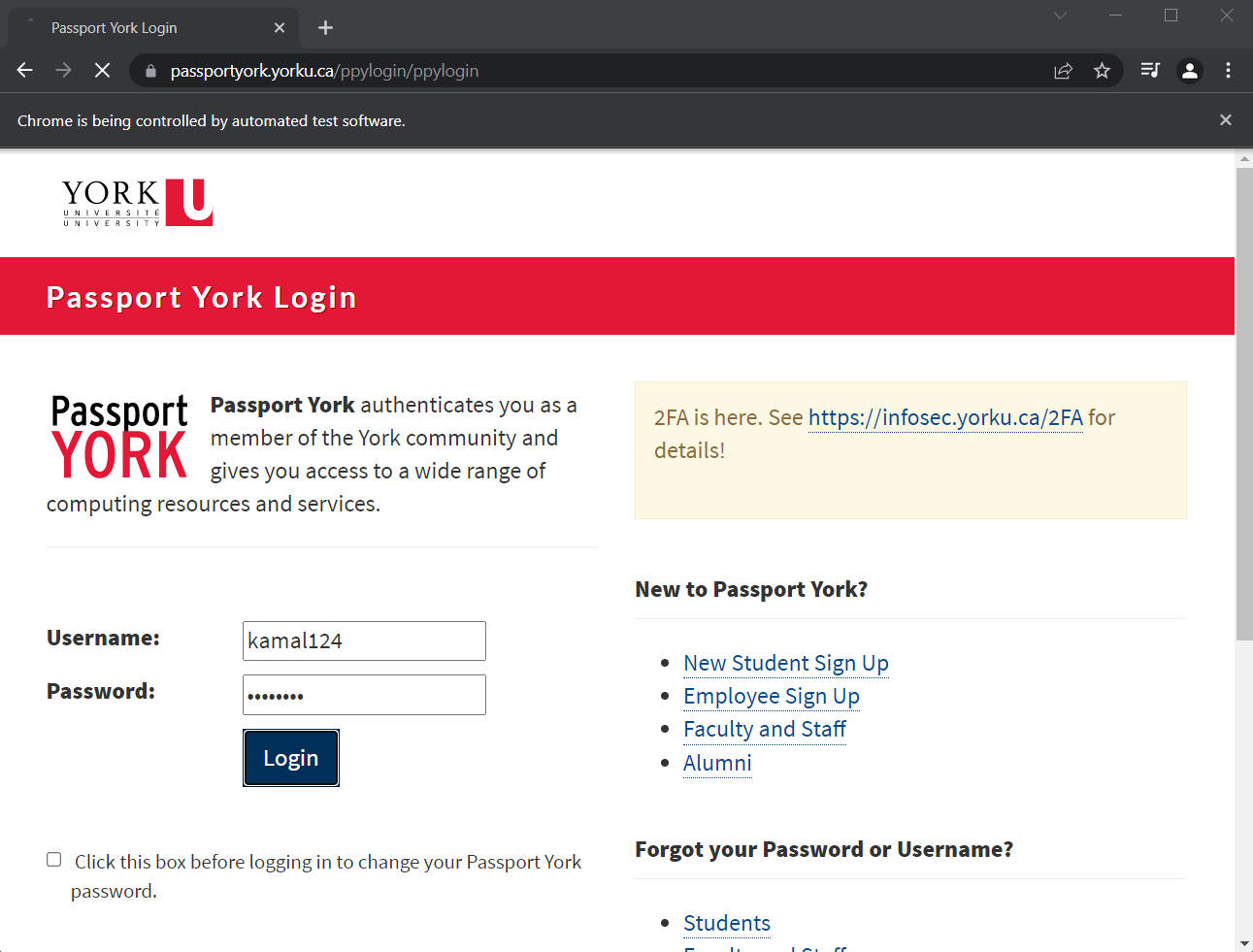
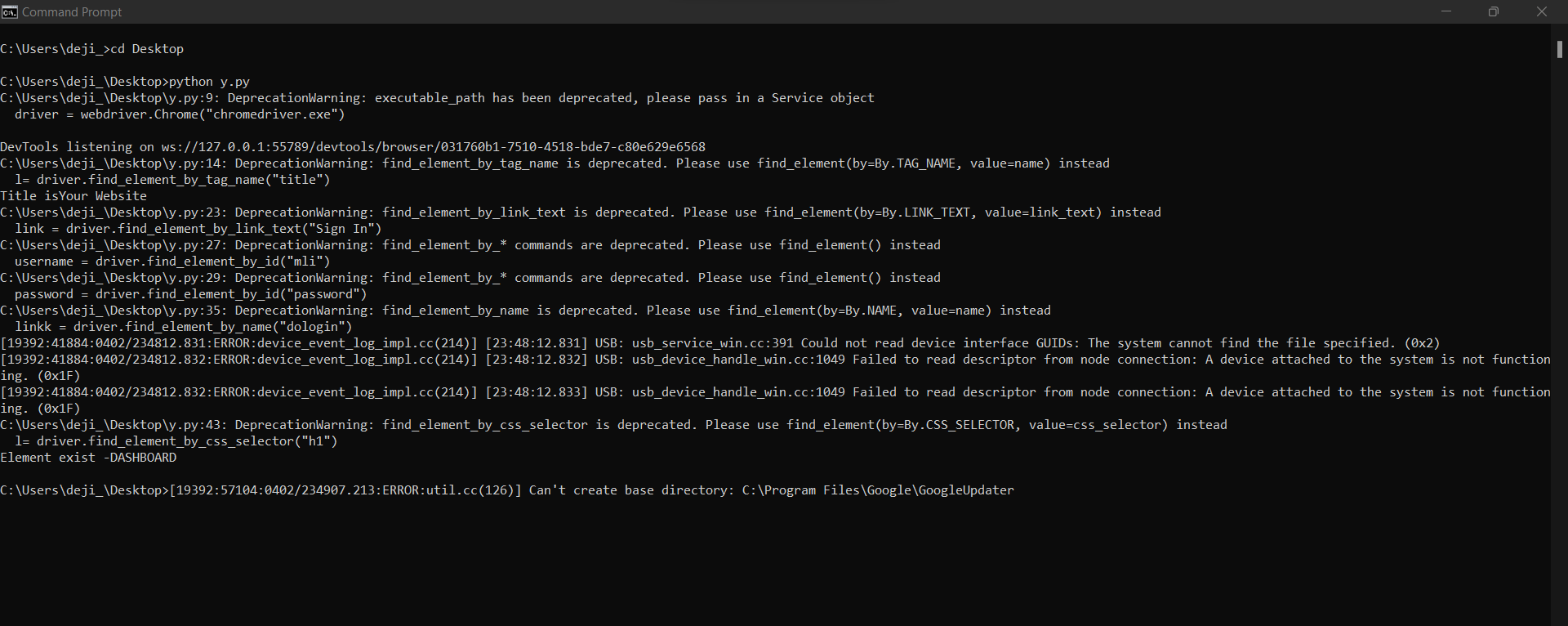
6. “Send Me a Push button “ is then clicked appropriately

7. Waiting is then done until login is confirmed

8. Then we Verify in the body, there's the H1 element with content DASHBOARD (successfully logged in)





Learning Challenge 3

1. It shows the response body of the me using the get method
2. Yes the username has been encrypted as a security measure