Nutrientry: Fasting and Calorie Tracker

Alejandro Martinez

East Los Angeles College

CIS 112: Javascript Programming

Professor Rodriguez

Abstract

Login page has been completed with the logic so you can login and we store the username, email and password after it has been hashed. The login page uses credentials that we grab from the users information that they put in the register page. The data is stored in a postgresql relational database using SQL Alchemy as the ORM.

![Graphical user interface, website

Description automatically generated]()

Figure 1: Login Page

Our sign up page looks similar and has an added field for an email address. I am considering looking into learning how to get functionality in case a user forgets their password. The remember me button still needs some styling but it gives a Boolean value that allows the user to be remembered in storage when they close the browser. This way they don’t have to log back in. I am going to add functionality so that the landing page checks if the user is logged in so they can be directed straight to their dashboard. The code for this portion is below: ![Text

Description automatically generated]()

Figure : Login Code Part 1

![Text

Description automatically generated]()

Figure : Login Code Part 2

The login and sign up form are very similar and both have very similar structures. The fields are being populated by a Python for-loop and we use that to get the different fields automatically based off of my Python form class that is being inherited from WTForms.

![A screen shot of a mountain

Description automatically generated]()

Figure : Signup Page

I was able to get the login to work using Flask-Login which is the first time I learn and use the library, but now I am able to create a user and give them login permission to view certain pages and I can restrict access on other pages if necessary. I am also able to logout a user by sending a POST request to the logout route and this redirects them to the landing page where the user can decide what they want to do from there. The forms are being supplied through Flask-WTForms. I am collecting this data and adding it to the database. The following figure is what the signup and login page look like without any css.   
 ![Graphical user interface, application

Description automatically generated]()![Graphical user interface, application

Description automatically generated]()

Figure 5: Forms without styling

I am using a real login functionality and storing the user passwords with a SHA246 hash through the use of the standard Werkzeug library.   
![Text

Description automatically generated]()

Figure 6: Database User Information

Nutrienty Proposal

# References

(n.d.). Retrieved from Fat Secret: https://www.fatsecret.com/Diary

(n.d.). Retrieved from Lose It!: https://www.loseit.com/

Kruger, J. (2006, July 19). *Dietary and physical activity behaviors among adults successful at weight loss maintenance*. Retrieved from Springer Link: https://link.springer.com/article/10.1186/1479-5868-3-17

UnderArmour. (n.d.). Retrieved from My Fitness Pal: https://www.myfitnesspal.com/food/diary