Negative15: Documentation

Negative15 is a meal-tracking website for dining hall meals and allows its users to enter what they ate using foods from the HUDS menu, calculate the nutritional profile of the meal, and save their meal history for future reference. The source code for negative15 comes in a .zip file. In order to use negative 15, there are first some steps that the user needs to take before the user can locally host the website.

When coding our project, we stored our files in the localhost folder in the CS50 appliance, however, we're going to have you run it in its own separate folder. First download the negative15.zip file and put it in vhosts in the CS50 appliance. Unzip the file using unzip negative15.zip. Execute cd negative15, followed by ls. You should see three folders: public, templates, and includes, as well as a few other SQL files. Next, in the terminal, execute the command sudo gedit /etc/hosts. This should open gedit and bring you to a file that is normally read-only. Add the line 127.0.0.1 negative 15. This associates that appliance's IP address with negative15, the folder that contains the source code for our website. Now we start the fun process of chmod'ing all the files. In a terminal window, change directories until you are in the negative15 directory in vhosts. Then, chmod 700 includes, templates, chmod 711 public, public/css, public/css, public/fonts, public/img, public/js, chmod 600 includes/*.php, public*.php, and finally chmod 644 public/css/*, public/fonts/*, public/img/*, public/index.html, public/js/*.

Now, if you open Chrome inside of the CS50 appliance and try to go to the address http://negative15/, the browser should be configured to render our webpages using code found in ~/vhosts/negative15/public.

Next, set up the databases needed for the website. Using Chrome inside the appliance, visit negative15/phpmyadmin. Log in using username "jharvard" and password "crimson." Using the GUI, navigate to the SQL tab. Open the SQL file included in the negative15 folder: negative15.sql. Copy the entirety of negative15.sql and paste it into the box in phpmyadmin titled "Run SQL query/queries on server "localhost"": Click the button "go". You should see a green banner that says # "MySQL returned an empty result set (i.e. zero rows)." A new database: negative15, should appear in the left hand sidebar. If it doesn't, refresh the page. Click on the database: you should see two tables, titled "mealhistory," and "users."

You should now be ready to use the website! Open up chrome inside the appliance and visit http://negative15. You should see a login screen. Click the sign up link beneath the log in button and visit the registration page to get registered. Hover your mouse over the left side of the screen to use the side-bar to navigate, or use the home screen buttons. You can also click on the large "NEGATIVE 15" logo at the top of the page to navigate back to the home screen. On the home screen: you have a few options. Click on "Today's menu" if you're only interested in finding out what is available today. Click on "Make a meal" to select a meal time and create a meal out of the available food options (if no HUDS information is available for that specific mealtime, then clicking the Go! Button will not do anything). On that page, you will also be able to select for the number of portions you want to have, and calculate and view a nutritional profile for that meal. Finally, you have the option of saving your meal into your meal history. Back on the home page, you can click on "Meal History" to view a list of all the meals you have recorded. Clicking on a specific date opens up an accordion folder which displays the foods you ate for that specific meal, as well as the nutritional value for the foods you ate.

As a sidenote, small errors may occur infrequently, for example, in the final display of the nutrition facts of a given user-selected meal, a row may appear blank. This is due to errors in the CS50

food API. For example, we know that requesting the nutrition facts for the ham and cheese omelet returns a blank JSON object.

Hopefully, you find this website useful and choose it to keep track of your health over your time at Harvard!