Project Part 3
CS-GY 6083 C Fall 2022
Prof Frankl
Due 11:59 pm Sunday Dec 11, 2022
You will hand in your code via GitHub and a short write up (details coming soon). You will also schedule a demo session in which one of the TAs will run through a series of tests with you.

In part 3 of the project, you will use the table definitions we will post (solution to part 2) to implement application code for Cookzilla as a web-based application. You may use Python, PHP, Java, node.js, or Go. If you'd like to use some other language, check with the TAs by 11/28/2022. *You must use prepared statements if your application language supports them.*

I have supplied Python code for a sample application that includes most of the constructs you'll need in order to do this assignment, along with videos that explain the code in detail. If you do not have prior experience with web and/or database application programming, you should study these and you may modify the provided sample code to implement Cookzilla.

Your Cookzilla implementation should allow a user to log in, search for recipes, display relevant info about recipes, and post recipes and related data. In addition, you must add some other features, as described below. (The number of additional features depends on whether you are working alone or with others).

You may continue to work alone or with the one or two others with whom you worked on parts 1 and/or 2 or you may form a team (two or three students, total): Any new teams or additions to a team must be reported to the TAs (by e-mail or on EdStem by Nov 27.) Everyone on a team is expected to understand all the code; you should work collaboratively on the required features; each team member should be the primary developer for two of the additional features.

Assume each user has already registered and created a password, which is stored as an SHA-2 hash. (We will provide Python/Flask code to manage user registration and login).

Cookzilla should support the following **REQUIRED** use cases:

1. **Search for Recipes** : Users should be able to search for recipes that have a particular tag and/or a given number of stars.
2. **Display Recipe Info:** Given a recipeID (possibly selected from a menu based on a search), display relevant information about the recipe, including the description, the steps in order, etc
3. **Login:** The user enters her username and password. Cookzilla will add "salt" to the password, hash it, and check whether the hash of the password matches the stored password for that username. If so, it initiates a session, storing the username and any other relevant data in session variables, then goes to the home page (or provides some

mechanism for the user to select her next action.) If the password does not match the stored password for that username (or no such user exists), Cookzilla informs the user that the login failed and does not initiate the session. **We will supply Python/Flask code for this. If you're using a different implementation language, you'll need to write this yourself.** After successful login, the user should see their profile, the recipes they've posted and the groups they belong to.

The remaining use cases require the user to be logged in.

4. **Post a Recipe:** Post a recipe and related data (steps, tags, etc)

ADDITIONAL FEATURES: If you are working alone, you should also implement at least two more features. If you are working with one partner, you should implement at least four more features. If you are working as a group of three, you should implement at least six more features. The following are suggestions. You are welcome to be creative and implement other features in place of these, but they must be specifically relevant to Cookzilla and must involve non-trivial SQL statements. (Registering users or other generic features of that nature are not permitted.)

Suggestions:

5. Log users' actions and display recipes and/or reviews that they viewed recently
6. Post a review
7. Post an event for a group that user belongs to
8. RSVP to an event that the user belongs to
9. Convert the units in a recipe
10. More complex searches
11. Search for users with similar taste as the given user (e.g. who have given similar ratings to some kinds of recipes)
12. The user can set a preferred unit, and everytime they see the recipe ingredients, they should be able to see it in the previously set unit.

If you want to do any other query apart from the ones suggested above, please check-in with us first by Dec 6.