# Nick’s Kitchen, Recipe Database

### Nicholas Skinner

<http://web.engr.oregonstate.edu/~skinnern/FinalCS340/Home.php>

### 1. Introduction

Often when browsing the internet for recipes, one may find a very old recipe with a super cheap cost to make, however, that price is no longer accurate, and you could find that its way out of your price range. To approach deprecated data like this is often frustrating, and you wish that it could just be updated with other things. My website is an approach to fix some of these issues, by allowing users to adjust costs, costs of old recipes will be updated along with the new, preventing data from getting too old or stale. To do this I have costs assigned to ingredients, and recipes are composed of amounts of ingredients, multiply them together and bam! You have the predicted cost at any point in time.

### 2. Detailed Functionality & Requirements

Some detailed functionality of this website includes

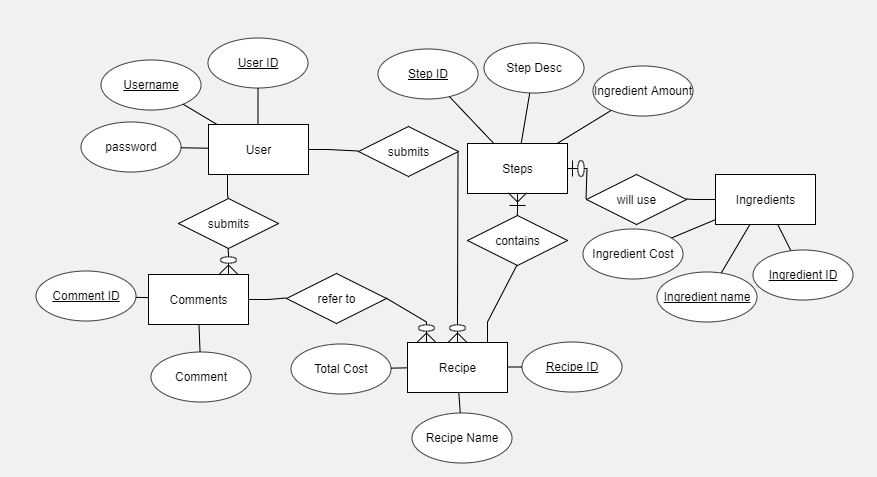
* Users can sign up for accounts with usernames, passwords, and emails
* Users can login, and logout
* A list of recipes will be displayed on the homepage with clickable links
* Users can comment on any recipe that they choose
* Users can search recipes, users, ingredients, and costs
* Users can add or update the cost of ingredients

Some of the business rules that have been adjusted are that I did not end up implementing a rating system, however, I still have the following business rules:

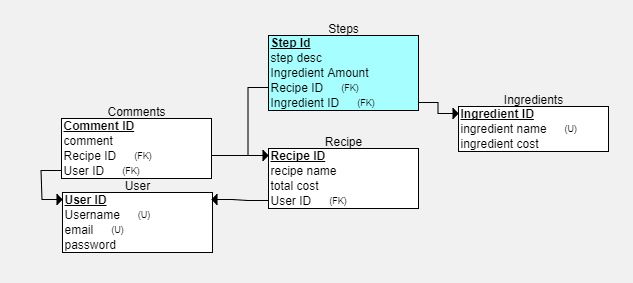
* Recipes may only have one author
* Comments may only have one author
* Email address for account must be unique
* Usernames must be unique

### 3. Database Design

## ER Diagram of Database



## Relation Schema



## Database Tables

My tables are displayed with these schema pictures, and they are also in the TEACH submission. My procedure/trigger will also be in there, but to touch on them for a second.

My trigger and procedure ended up being the same query to update all my recipes, they create a join across 3 tables to get the ingredient cost from ingredients, the ingredient amount from steps, and the recipe id from recipe, it then takes these and creates a sum of the cost to make the recipe, and updates the recipe cost in the recipe table with the cost.

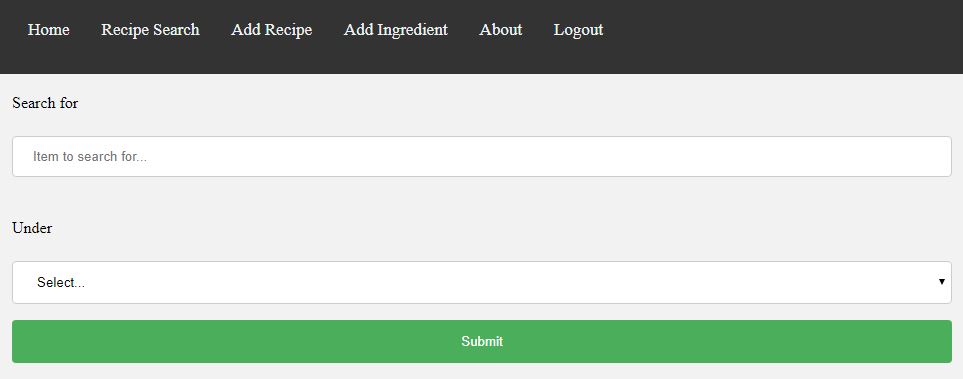
### 4. Website Design

The design of the website simple as far as I can see, 3 or fewer clicks to navigate, and a mostly straightforward navbar.

## Website Layout

This diagram should briefly describe the webpages in your website and show how they are connected.

## User Interface



The general idea of my website is to adhere to the 3 click rule, and to let the navbar be the thing to click to get to where you want to go. You can get to the

## User Manual or Help page

### 5. Application Implementation

* My Implementation of the web side of my project was very basic, nothing outside of php/html/css as I don’t know much about web development, I also had user confirmation pages to tell the user that they either were successful, or failed at what they were trying to do. These pages don’t redirect or anything though.
* My SQL queries were mostly basic joins, I think the worst offender was a three table join for my trigger, but even that’s not too complicated. Outside of that, everything was mostly a select t1.col1, t2.col2 join, or a basic insert/update so nothing had been too confusing to look at.

### 6. Evaluation

I had two others use my website and try to navigate it. They could view the recipes, leave comments, and login and logout as a normal user would. They didn’t delve too much into submitting recipes, which is where I think I could have used the testing, but they said it appeared straightforward enough for a new user to use.

### 7. Future Work & Lessons Learned

What additional functionalities do you plan to implement beyond this class? What challenges have you run into during the design, implementation, and testing of your application? How did you address these issues? What would you do differently next time?

Some functionalities that could be implemented outside of class is creating some form of control to prevent someone from setting the price of one ingredient way to high by creating some algorithm to weed out what is reasonable, and what is not.

Something big for the website that I came into and had a lot of trouble with is input validation, and error checking. After I would make a page display what I wanted under how I know the website to work I thought I was done, but soon after I would realize that I should have set up some form of input validation, sanitization, or something else because many portions of my website can be broken with sql injections into the textboxes.

My approach to fix some of this (that I couldn’t complete in the given time) is to check user input client side before sending, I couldn’t get a strip special characters or anything like that to work, and I was running out of time to implement the rest of my features, so I decided to not implement the validation due to time constraints.

As this was one of my first times creating anything with PHP, I would likely take some form of digital class/overview of the language, and how to use it, that could place me in a much better position knowing the language more than not at all. I could also budget my time more wisely between classes to make sure that the project had gotten more of the time it deserved instead of the excess time I could give it.

***Appendix – Team Report*** *If you worked in a team summarize the division of labor.*

Only one on my team, did all the work. – Nicholas Skinner