2021

Cookbook Database

KEELEY COLLINS

Introduction

The web application will function as a place where users can create or save recipes into a digital 'cookbook.' Users can create multiple cookbooks under their account, with can contain different recipes. Recipes can be entered manually, or a url can be entered and the application will attempt to extract the relevant information for a recipe from a webpage. Recipes can be edited later, and will be searchable by other users by title, rating, category and/or main ingredient. Cookbooks are searchable by title and category. Recipes can also be reviewed (1 to 5 stars) and users may leave a comment. The website can be found at www.cookbookingfornerds.xyz.

System Architecture

From the home page of the web application, users will be able to either create an account or log in. Once logged in, each user has their own page where, in the form of links, they can view their saved cookbooks, and have an option to create new cookbooks, search the website, or view the user statistics for the website. If a cookbook is selected, it will redirect to a page which will list recipes the cookbook contains, along with links to create a new recipe either manually or from a url. If a recipe is selected, it will redirect to the recipe page which lists information about the recipe, like the title, ingredients, instructions, etc. From the recipe page, there are options to edit the recipe, or leave a rating with a comment.

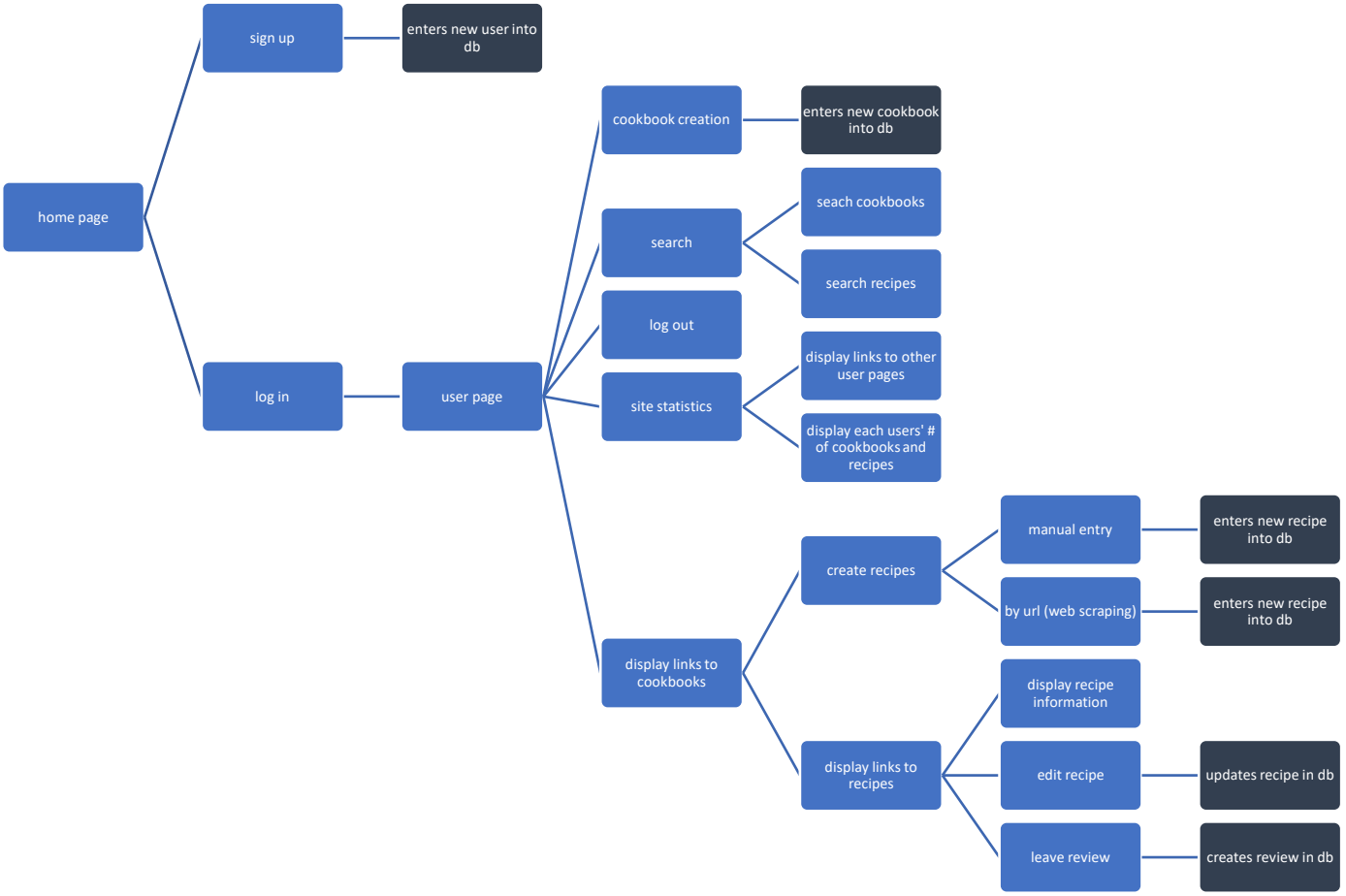
Creation of new users, cookbooks, recipes, and ratings all are done through forms on the website which, once submitted, will enter the data from the fields into their corresponding columns/tables in the database. It is only possible to access cookbook and recipe creation for the current user (users cannot edit other users' data). Recipes are the only object that can be edited after initial creation. For recipe editing, the same form is used as recipe creation, but it will be

prefilled with the current information for that recipe which can then be changed by the user. Upon submission, the entry in the database will be updated with the new information.

The link to the search page will lead to a form where either recipes or cookbooks can be searched. If the "cookbook" option is selected cookbooks will be searchable by title or category. For the search results, positive matches will include a link to the relevant cookbooks by title. If the "recipe" option is selected, recipes will be searchable by title, category, main ingredient, and/or minimum rating. For the search results, positive matches will include a link to relevant recipes by title and will also display a description of the recipe. For recipes, if the minimum rating is set to "0," then it will search all recipes regardless of whether they have ratings or not, and will not display the average rating in the results. If the minimum rating is set to at least "1," only recipes with average ratings above the minimum will be displayed, and the average rating for each recipe will be listed.

The link for "site statistics" on the user page, will redirect to a page listing all users, and each users' number or cookbooks and number of recipes. The users name will link to their user page. These pages will not contain links for cookbook or recipe creation, since users can't edit each other's data, however their cookbooks and recipes can still be viewed, and reviews can be left on their recipes.

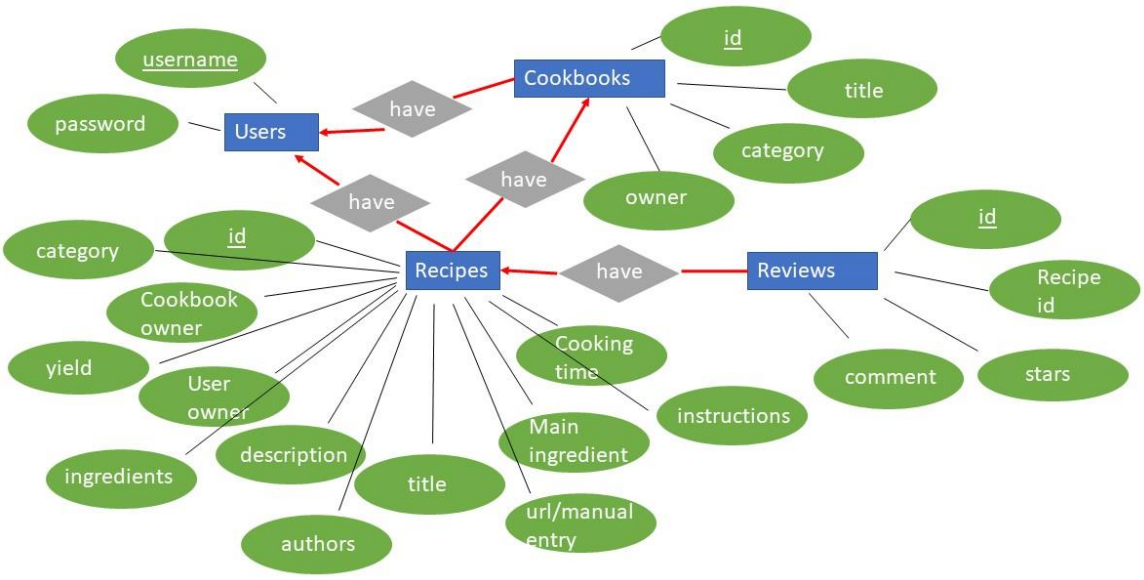
Graphical Overview of Architecture:



Database Design

The database is a mysql database with four tables: users, cookbooks, recipes, and reviews.

ER Diagram:



Tables:

Users (passwords are actually different):

id	name	password
1	KC	jjjjjjjjj
2	veggie_lvr96	jjjjjjjjj

Cookbooks:

id	title	category	owner
1	Homecooking	american	KC
2	Desserts	dessert	KC
3	Salads	salad	KC
4	Junk Food	things I shouldn't eat	veggie_lvr96
5	Lunch	lunch	KC
6	Wraps	wraps	veggie_lvr96

Recipes (only 1 row shown because the table is very wide):

Id	Title	Authors	url	Cooking time	Amount	Ingredients	Instructions	Description
6	BLT	Paul	Manual entry	20 min	1 sandwich	Bacon, lettuce, tomato, bread	Put together sandwich	Great sandwich

Category	Main ingredient	User_owner	Ckbk_owner
Lunch	Bread	KC	5

Reviews:

```
+----+-----------+-------------------------------------------------------+-------+
| id | recipe_id | comment                                               | stars |
+----+-----------+-------------------------------------------------------+-------+
|  1 |         1 | Burned down my kitchen! Terrible!                     |     1 |
|  2 |         1 | Wow, this was really good!                            |     5 |
|  3 |         1 | I added more sugar.                                   |     4 |
|  4 |         2 | Great on a cold day!                                  |     5 |
|  5 |         2 | I burned the roof of my mouth. What a hazard to society! |     3 |
|  6 |         1 | What a great recipe!                                  |     4 |
+----+-----------+-------------------------------------------------------+-------+
```

Implementation Notes:

The web app was implemented using the python library Flask. There is a mysql database on the back end and sqlalchemy is used to connect and run queries to it. For web scraping the python libraries BeautifulSoup and cloudscraper are used. CSS, HTML, and Jinja are used for the front end. Session is used to store information for the current user (reset on logout) and for current cookbook or recipe ids (based on the last page visited). The web app is hosted on dreamhost, and can be found at www.cookbookingfornerds.xyz.

General project structure:

/cookbookingfornerds.xyz (project folder)

passenger_wsgi.py (starts app)

/app

__init__.py

config.py

routes.py (routes/renders webpages, most db queries are here)

models.py (creates db tables)

forms.py (code for forms)

auth.py (routes for user creation/login)

fetch_recipe.py (web scraping)

/static (folder for static files)

/templates (folders for css/html templates)

Test Results:

To display cookbooks on user page:

select title from cookbook where owner = (current_user.name); (this user is ‘KC’)

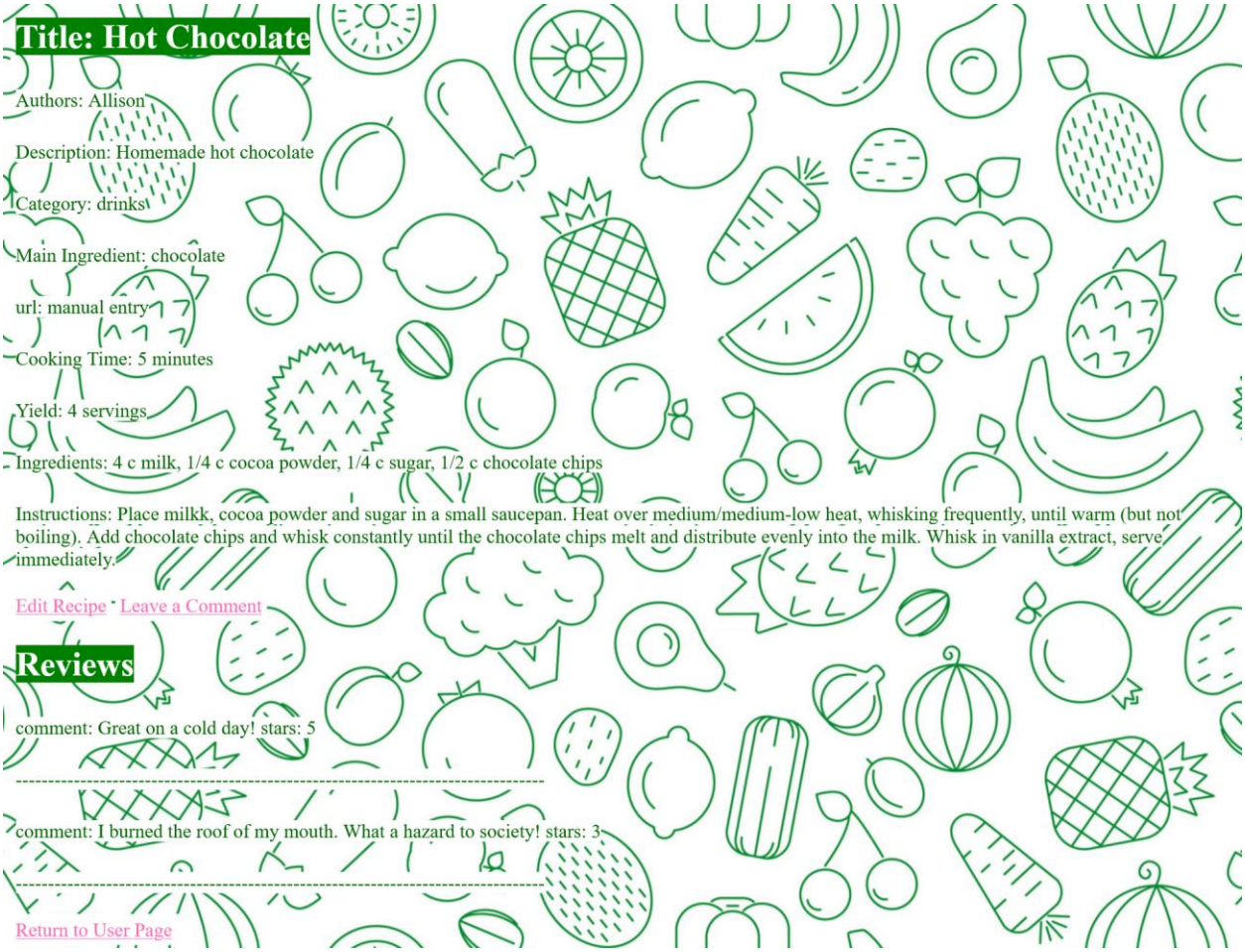
To display recipes on cookbook page:

Select title from recipe where ckbk_owner = (cid); (cid is the cookbook id stored in session)

To display recipe information on recipe page:

Select * from recipe where id = (rid); (rid is recipe id stored in session)

Select * from review where recipe_id = (rid);



To search the website for either cookbooks or recipes:

(The query will depend on what information is passed from the form, below are two possible sample queries)

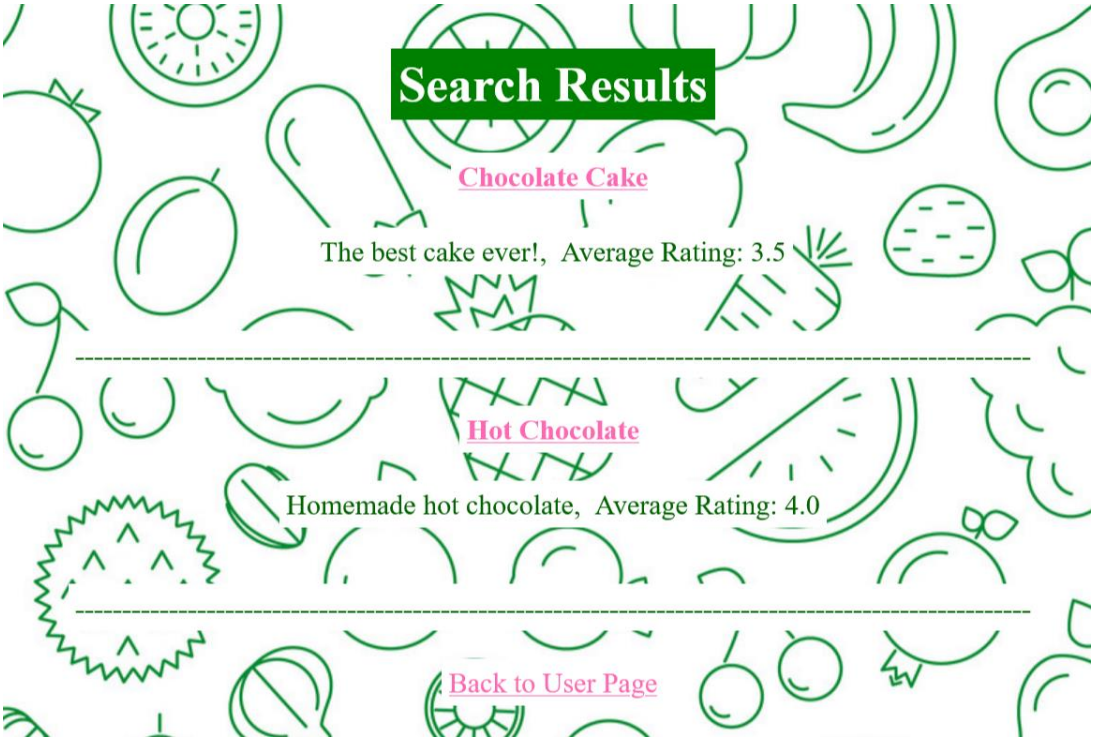
Select * from recipe where main_ingredient like ‘%fruit%’ AND category like ‘%salad%’;

Select * from cookbook where title like ‘%Homecooking%’;

If minimum rating is specified on the seach form (it’s not set to “0”):

(this query is for getting all stars for certain recipes, which are averaged for each recipe and displayed on the search page if the average rating is above the minimum, in this case minimum rating = 2)

Select stars, recipe_id from review, (select * from recipe where title like ‘%Chocolate%’) as searched recipes where searched_recipes.id = recipe_id



To display user statistics:

First the users table is queried for all users:

Select name from users;

For every name selected this query will be run:

Select COUNT(distinct recipe.title) as recipe_number, COUNT(distinct cookbook.title) as ckbk_number, user_owner from recipe, cookbook where user_owner = {name} and owner = {name};