Cooking Calendar: Database Design

Overview

I am choosing to use a relational database because I plan to have multiple many-to-many relations. While it's possible to separate recipes into single documents, that data would be very inefficient to search. I plan for users to search for recipes based on ingredients and tags often so having a separate table with proper indices will allow the site to be responsive. In addition, the shopping list functionality is heavily dependent on the chosen recipes and those recipes could be updated by the user frequently.

User Accounts

There will be one type of user account with access to read or write to any table, as the API endpoints allow. There will be a user table that will keep track of all accounts so that all user's data can be kept separate.

Database Design

ER diagram

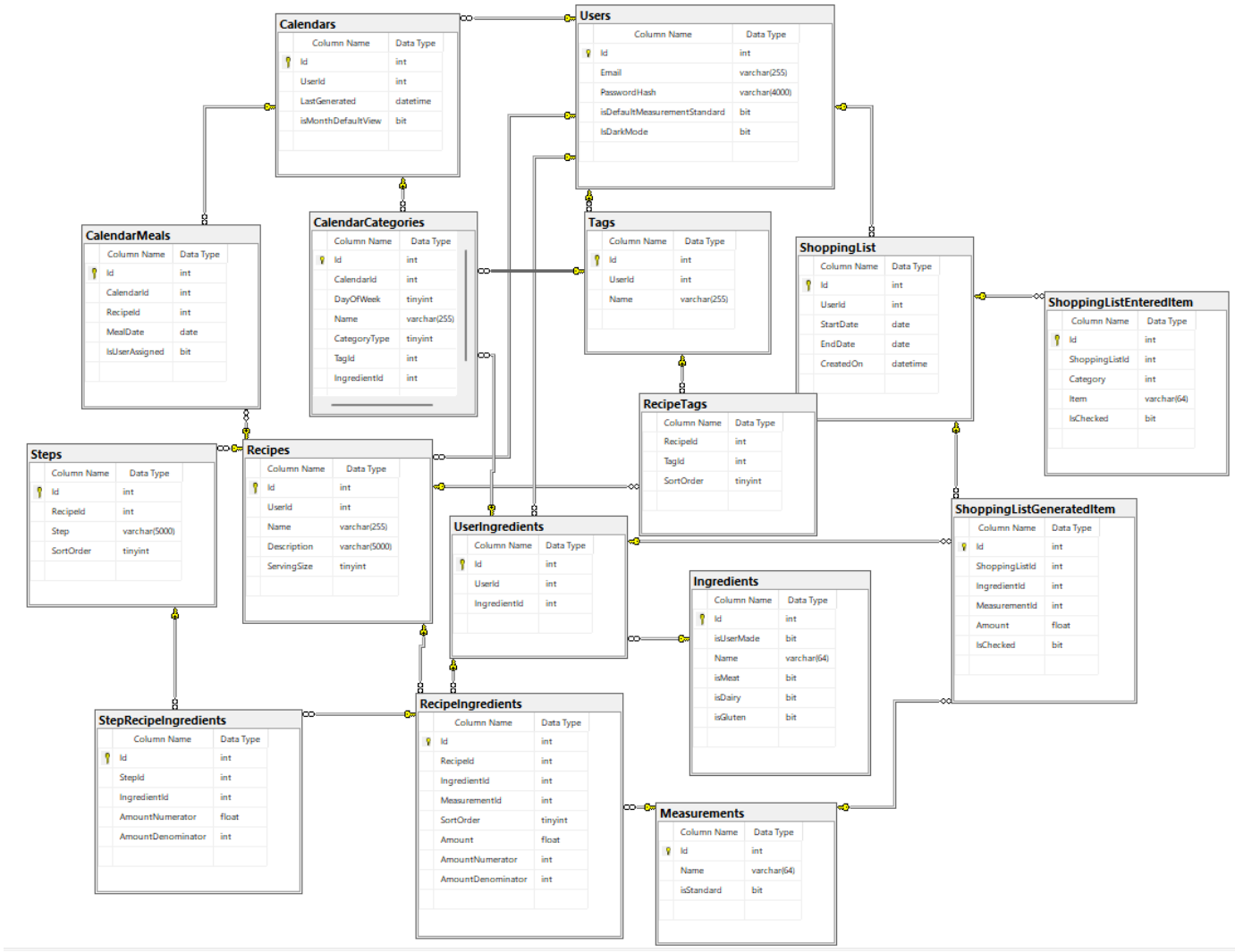

Table Justification

Users

Holds account information and user settings. From this table's "Id" column data is separated so that each user will see only their data.

Measurements

This is a basic lookup table and while the user can not add to or edit the table they will use it indirectly. It will hold the names of all measurement types (Cup, Gram, Lb, Liter…) and a flag to represent whether the measurement type is standard or metric.

Ingredients

This is meant to be a lookup table that gets completed over time. Theoretically, our database would hold every ingredient a user would want as well as basic dietary restriction info. Having a defined list of ingredients means that the user will not have as many typos or use synonyms for the same ingredient. Also having basic allergen info for ingredients will let the application automatically determine what recipes are gluten-free, dairy-free, and vegetarian. Users will be able to add to the table if an ingredient they want to type does not already exist in the system. User-entered values would then be reviewed and either made a permanent addition for all users to select from or remain available only to the user that entered it.

UserIngredients

Links a user and ingredient together. This will allow for faster searching of ingredients, user-defined ingredients, and a more focused list for users to pick from when adding ingredients to recipes. Rows will be unique based on the combination of UserId and IngredientIds.

Recipes

Holds the basic recipe info (name, description, serving size). This table is what will allow a user to add recipes. Other recipe information is gathered through links with the "Steps", "RecipeIngredients" and "RecipeTags"

RecipeIngredients

Links a recipe with its ingredients together. This table will hold the measurement amount for an ingredient in a recipe either expressed as a fraction or a decimal value, depending on user input. Note that the main reason I chose not to just use a float value is that recipes will often express amounts as fractions and the user should see the number exactly as they expect. When adding the feature to change measurements based on the number of servings for a particular day storing the amounts as a numerator and denominator ensures values will not suffer from round-off error when converting.

Tags

For now, tags are just a name that a user can attach to recipes to group them together

RecipeTags

Links tags with specific recipes. Contains a sort order column so that on a recipe page the user can adjust what tag will come first, second, and so on. As a default the sort order will represent the order the user added the tags.

Steps

This table has a many-to-one relationship with recipes. Users will use the table to keep track of each “step” in the process of cooking a recipe.

StepRecipeIngredients

This table will link a step with the ingredients used in a recipe as well as how much. This will let users see the amount of an ingredient to use in both a specific step and in the list of ingredients. In addition when the recipe instructions say “use ⅓ of the ingredient” the application can calculate how much that is for the user. Lastly, when a user changes the serving size the instructions will be updated to match the amount in the ingredients table

CalendarCategories

This field will hold up to 7 records for a user to represent the chosen food category for each day of the week. The fields will let each day have a custom name and choose recipes based on a tag, ingredient, or completely at random. Note in the future there could be different categories for each meal of the day.

Calendars

Keeps track of calendar information for a user. For now, intended to be a one-to-one relationship with users. In the future, a different calendar record could be used for breakfast, lunch, and dinner or to keep track of separate meal plans.

CalendarMeals

This table puts specific recipes on particular dates. The combination of CalendarId and MealDate is unique. The table also keeps track of whether a recipe was assigned by the system or by the user so that when a user would choose to regenerate a plan, the manual changes they made would not get lost.

ShoppingList

The user will need this table to keep track of possible separate shopping lists based on start and end dates. The items on a list are separated between ingredients the system knows the user will need and any other items a user wants to add

ShoppingListGeneratedItem

The table will keep track of the amount of an ingredient the user will need on a shopping list. The generated items can be “checked off” to be marked as completed but will remain visible.

ShoppingListEnteredItem

The table will let the user keep their own shopping list on the site so that all their groceries are in the same place. The category column is meant to link to a specific enum in the code separating drinks, snacks, and any other possible category that might be added later.