Features

- users can sign into the app with their email and password
- users can create recipes with ingredients and instructions
- recipes can be marked as public or private
- users can view other people's recipes
- ingredients from recipes can be added to user's grocery lists
- users can create their own occasions and assign recipes to occasions

Brainstorming:

- Username
- Email
- Password
- First name
- Last name
- Recipes
- Ingredients
- Instructions
- Is public? Boolean
- Recipe author
- User grocery list
- Occasions

Table ideas:

- Users
  - Will be a table to hold information about users, each row will be a user
- Recipes
  - A table which will hold information about recipes, and will have foreign keys for what ingredients are contained and what instructions are required, each row will be a recipe
- Ingredients
  - A table which will hold all ingredients
- Ingredients list
  - A table which holds lists of ingredients
- Instructions
  - A table which will hold all instructions
- Grocery lists
  - A table which will hold all information for users grocery lists such as a foreign key for the user id to know who the list belongs to
- Occasions
  - It is a table which holds all occasions and which recipes are in those occasions

Relationships:

- One-to-one
- One-to-many
    - Users: I selected users as one to many because the user id can be used as identification in several places, but those places can only use one id
    - Grocery list ingredients: this only uses ingredients to make a list which is referenced in the grocery lists
    - Ingredients: This is one to many because it only gives its id to other tables
- Many-to-many
    - Recipes: This is because recipes can be shown in multiple places, and recipes uses multiple tables to make its own contents
    - Instructions: The reason for this is that instructions are used in recipes and recipes have many ingredients
    - Grocery lists: This is this way because grocery lists uses users info and grocery list ingredients info to pair them as a grocery list owner
    - Occasions: Is many to many because the occasions have a user id and a recipe attached to them

Columns:

- Users
    - Email: I chose this so the front end could pull to allow user to log in, chose varchar so that it could limit email length
    - Password: chose this so the front end could pull to allow user to log in, chose varchar so that it could limit password length
    - User first name: chose this so people can identify the user on the site, chose varchar to limit name length
    - User last name: chose this so people can identify the user on the site, chose varchar to limit name length
- Ingredients
    - Ingredient name: Chose this so each ingredient id also has a name to be identified by, chose varchar to limit name length
- Ingredients lists
    - Ingredient list numt: I did this so different ingredients could be created into lists for recipes, chose INT because you will have to select which list you want the ingredient to be in
    - Ingredient name: this is so you can tell which ingredient is going into a list, chose INT because it uses an ingredient id
- Instructions
    - Instructions name: This is so the instructions can be identified easier, chose varchar to limit string length
    - Instructions ingredients: this is so you know what ingredients you need for these instructions, I chose INT because it is pulling from the ingredients lists id
    - Instructions text: this is for the actual body of instructions, VARCHAR so you can enter text instructions

- Recipes
    - Recipe name: This is so you know what the name of the food is to be made, varchar to limit length
    - Recipe author: This is so you can identify the user who wrote it, INT because it is using the user id
    - Recipe instructions: This is so the recipe knows which instructions belong to it, INT because it is pulling from the instructions id
    - Recipe is public: This is so the front end knows if it can display the recipe to everyone or not, BOOLEAN because it is just true or false if it is public or not
- Grocerylists
    - User_id: This is so you can identify whose grocery list it is, INT because it pulls from user id
    - Grocery list ingredients: So you can know which ingredients are in the grocery list, INT because it pulls from ingredients list id
- Occasions
    - User_id: So you can identify which use created the occasion, INT because it uses the user id
    - Recipe_id: This is to know which recipes are in the occasion, INT because is uses the recipe id