

Summary

We want to create a recipe creating/sharing and grocery list app. You'll be planning out what tables we'll need, what information they'll store, and how the data will relate to each other.

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

BRAINSTORM

- User login information (username)
- Auth information (email/password)
- User's recipes (ingredients/instructions/public vs. private)
- Grocery list(ingredients from recipes)

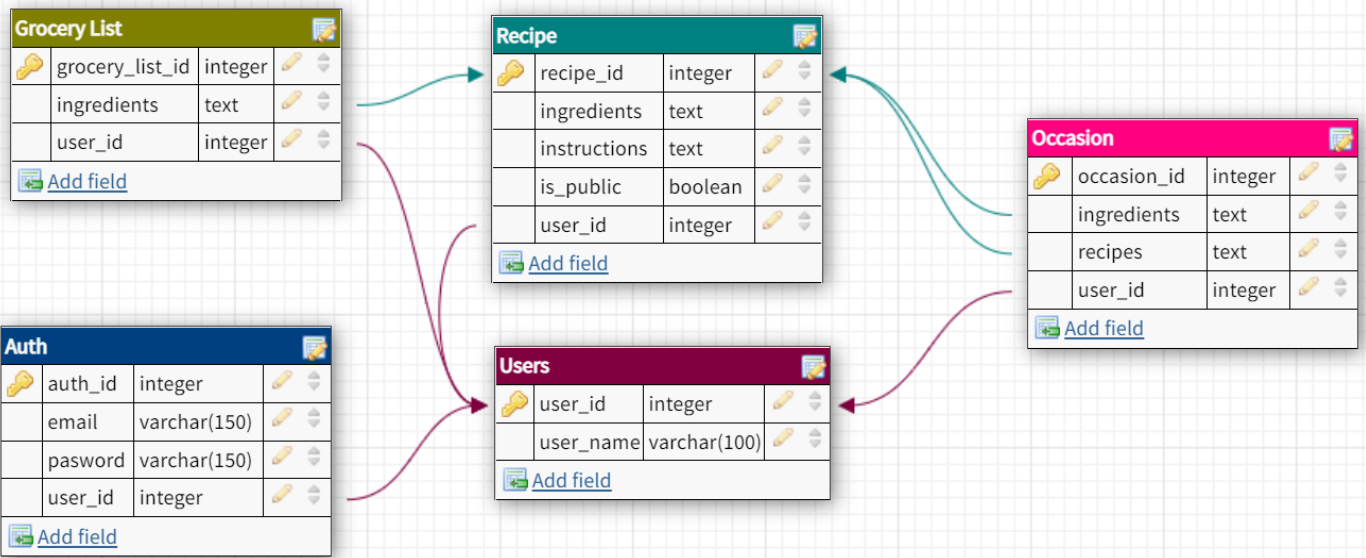
TABLE IDEAS

- Users
 - username
- Auth
 - Email
 - password
- Recipes
 - Ingredients
 - Instructions
 - public/private
- Grocery List
 - Ingredients from recipes/ ingredients to buy
- Occasions
 - recipes

RELATIONSHIPS

- **ONE TO ONE**
 - User to Auth (There is only one user to their unique info)
- **ONE TO MANY**
 - Users to Recipes (One user can have many recipes)
 - User to Grocery List (One user can have many grocery lists)
 - User to Occasions (One user can have many occasions)
- **MANY TO MANY**
 - Recipes to Occasions (There is many recipes for many occasions)
 - Recipes to Grocery List (Many ingredients from different recipes can be added to many grocery lists)

COLUMNS



Users

Users			
🔑	user_id	integer	
	user_name	varchar(100)	
Add field			

- Chose the user id and user name because it relates specifically to the user
 - Chose integer for user_id because that will be the designator of each user
 - Chose varchar for user_name and made it unique so that each user could have a unique name of less than 100 characters













Occasions

Occasion			
🔑	occasion_id	integer	
	ingredients	text	
	recipes	text	
	user_id	integer	
Add field			

- Chose recipes in occasions because users will input what recipes they want for various occasions of their choosing
- Chose ingredients because each recipe will have various ingredients
- Chose user_id because each occasion will be linked to one User









- Chose text because it will already be chosen from recipes
- Chose integer because that is the value of the SERIAL KEY

Recipes

Recipe			
	recipe_id	integer	 
	ingredients	text	 
	instructions	text	 
	is_public	boolean	 
	user_id	integer	 
 Add field			







- Ingredients will represent the ingredients needed for the recipe as text with no limit because it can require a lot of ingredients
- Instructions will represent the instructions to cook the recipe as text with no limit because it can require a lot of ingredients
- Is_public will determine if it is available to the public or if its private as bool because it will either be public = true or false
- User_id is attached to the user that made the recipe

Grocery List

Grocery List			
	grocery_list_id	integer	 
	ingredients	text	 
	user_id	integer	 
 Add field			

- Ingredients represents the ingredients needed to buy from the grocery that were used in 1 of the recipes and in text with no limit because some can require a lot
- User_id is linked to the users table so we know which user uploaded the recipe

Auth

Auth			
	auth_id	integer	
	email	varchar(150)	
	password	varchar(150)	
	user_id	integer	
 Add field			

- We chose email as the storing data because it is a must to make an account and we chose varchar to limit how long it is down to 150 characters.
- Password to store their password and varchar for the same reason as the email
- User id from the users table to link them together

POSTGRES TABLE CODE

```
CREATE TABLE recipe_user (
  user_id SERIAL PRIMARY KEY,
  User_name VARCHAR(100)
);
```

```
CREATE TABLE auth (
  auth_id SERIAL KEY,
  email VARCHAR(150),
  password VARCHAR(150),
  user_id INT REFERENCES recipe_user(user_id)
);
```

```
CREATE TABLE recipe (
  recipe_id SERIAL,
  Ingredients TEXT,
  Instructions TEXT,
  Is_public BOOL,
  user_id INT REFERENCES recipe_user(user_id)
);
```

```
CREATE TABLE grocery (
  grocery_id SERIAL,
  recipe_id INT REFERENCES recipe(recipe_id)
  user_id INT REFERENCES recipe_user(user_id)
);
```

```
CREATE TABLE occasions(  
occasions_id SERIAL,  
recipes TEXT NOT NULL REFERENCES recipe(instructions),  
recipe_id INT REFERENCES recipe(recipe_id)  
user_id INT REFERENCES recipe_user(user_id)  
);
```