Brainstorming

Ingredient/instructions table connected to recipe table connected to user id
Recipe table is_visible boolean
Grocery list table connected to user and recipe table
Occasions table connected to recipe table and user
Ingredients and instructions would need to be text inputs.

User Table

- User_id
- user_name
- User_password
- User_email

Recipe Table

- User_id
- recipe_id
- Ingredients_text
- instructions_text
- public/private boolean

Grocery List Table

- user_id
- grocery_id
- From recipe ingredients_text

Occasions Table

- User_id
- occasions _id
- Recipe_id
- Occasions_text

Relationships

User Table => Grocery List Table => Occasion Table => Recipe Table: one to many
One user can have many grocery lists, occasions, and recipes

Recipe_id => User Table: many to many
Each recipe_id can be accessed by multiple users. Each individual recipe can only be connected/created by one user.

Grocery List Table => Recipe Table => : one to many
Accessing multiple recipes from the recipe table's ingredients field, associated only to one user, but staying as one grocery list.

Occasion Table => User Table => Recipe Table: many to many
Accesses recipes from the recipe table and is associated with one user.

Columns

User Table

- User_id: A unique user ID, self explanatory
- User_name: The user's name, self explanatory
- User_password: The users password, self explanatory
- User_email: The users email, self explanatory

Recipe Table

- User_id: To associate the creation of a recipe list to a user
- Recipe_id: To create a unique id for the the recipe
- Ingredients_text: The ingredients for the recipe in text
- Instructions_text: The instructions for the recipe in text
- public/private boolean: A check to see whether the user wants the recipe to be private

Grocery List Table

- User_id: To associate the user to their grocery list
- Grocery_id: To create a unique grocery id.
- From recipe ingredients_text: Obtains the ingredients from a recipe. (presumably)

Occasions Table

- User_id: To associate the user to their occasion list
- occasions_id: Creates a unique id for each occasion
- Recipe_id: The fetch the id of chosen recipe for the occasion
- Occasions_text: The unique occasion in question as defined by the user.

```
-- CREATE TABLE users (
--     user_id SERIAL PRIMARY KEY,
--     user_name VARCHAR(20),
--     user_password VARCHAR(100),
--     user_email VARCHAR(50)
-- );
-- -- INSERT INTO users (user_name, user_password, user_email)
-- VALUES ('John', '1234', 'john@john.com');

-- CREATE TABLE recipes (
--     recipe_id SERIAL PRIMARY KEY,
```

```
--     user_id INTEGER NOT NULL REFERENCES users(user_id),
--     Ingredients_text TEXT,
--     instructions_text TEXT,
--     private_bool BOOLEAN
-- );
-- INSERT INTO recipes (user_id, Ingredients_text, instructions_text, private_bool)
-- VALUES (1,'egg, egg, bacon', 'cook it', 'TRUE');

-- CREATE TABLE groceries (
--     grocery_id SERIAL PRIMARY KEY,
--     user_id INTEGER NOT NULL REFERENCES users(user_id),
-- --     recipe_id INTEGER NOT NULL REFERENCES recipes(recipe_id),fff
--     Ingredients_text INTEGER NOT NULL REFERENCES recipes(Ingredients_text)
-- );

-- CREATE TABLE occasions (
--     occasions_id SERIAL PRIMARY KEY,
--     user_id INTEGER NOT NULL REFERENCES users(user_id),
--     Ingredients_text INTEGER NOT NULL REFERENCES recipes(Ingredients_text),
--     instructions_text INTEGER NOT NULL REFERENCES recipes(instructions_text),
--     occasions_text TEXT
-- );

-- SELECT * FROM recipes
```