Table ideas

User(going to have information about person using the app which includes recipes, their grocery list and occasions they plan for)
- Id serial int key value
- Email varchar unique
- Password varchar
- Recipes fk
- Grocery list int fk
- Occasions int fk

Recipes(will include links to the creator of recipes, it will also be linked to the ingredients needed and a text box with the instructions)
- Id
- Public boolean
- Creator id fk user table
- ingredients(fk ingredient table)
- Instructions text

Ingredients(lists the name of ingredients, nutrition facts, and price of the ingredient)
- Id
- Name varchar
- Nutrition facts text or img
- Price float
- Got it boolean

Grocery list(has the user attached to it and is linked to the recipe)
- Id
- user_id(fk user table)
- recipe_id(fk recipe table)

Occasion
- Id
- Recipes(fk recipes)
- Date text date box
- budget float how much can be spent

Recipe_ingridients
- Id
- Recipe_id fk recipe table
- Ingredients_id fk ingredients table
- Quantity varchar(20) the amount you need
Association table connecting recipe to ingredients

Occasion_recipe
        Id
        Recipe_id fk recipe table
        Occasion_id fk occasion table
        Quantity int

User_recipe
        Id
        User_id fk user table
        Recipe_id fk recipe table

Relationships

One to one
User to grocery list

One to many
User to occasion- it will have just a single grocery list for your sessions and when you buy stuff it comes off your list

User to recipe- there will be many recipes a user will want to follow

grocery to recipe- your only going to have one grocery list and many recipes you'll be shopping for

Many to many
Recipe to ingredients- there are many recipes that will use the same ingredient list.

Occasion to recipe- you might have multiple recipes you'd use for an occasion

Columns

User
        Id serial int key value
        Email varchar unique
        Password varchar
        Recipes fk
        Grocery list int fk
        Occasions int fk

Recipes   Id
        Public boolean
        Creator id fk user table
        ingredients(fk ingredient table)

Instructions text

Ingredients
       Id
       Name varchar
       Nutrition facts text or img
       Price float
       Got it boolean

Grocery list
       Id
       user_id(fk user table)
       recipe_id(fk recipe table)

Occasion
       Id
       Recipes(fk recipes)
       Date text date box
       budget float how much can be spent

Recipe_ingridients
       Id
       Recipe_id fk recipe table
       Ingredients_id fk ingredients table
       Quantity varchar(20) the amount you need

Occasion_recipe
       Id
       Recipe_id fk recipe table
       Occasion_id fk occasion table
       Quantity int

User_recipe
       Id
       User_id fk user table
       Recipe_id fk recipe table

We used the serial key value for id because that is the default that we do. We used text boxes for things that needed a longer explanation. A date box was used to specify a date in the occasion section. Integer was used because that is how we show a number value. Float was used to represent money and prices.

Postgres code

```
-- CREATE TABLE ingridients(
--      id SERIAL PRIMARY KEY,
--   name VARCHAR(255) NOT NULL,
--   nutrition TEXT,
--   price FLOAT NOT NULL,
--   got_it BOOLEAN NOT NULL
-- );

-- CREATE TABLE occasion(
--      id SERIAL PRIMARY KEY,
--   name VARCHAR(50) UNIQUE,
--   occasion_date DATE,
--   budget FLOAT
-- );

-- CREATE TABLE recipe(
--      id SERIAL PRIMARY KEY,
--   public BOOLEAN,
--   instructions TEXT
-- );

-- CREATE TABLE occasion_recipe(
--      id SERIAL PRIMARY KEY,
--   recipe_id INT NOT NULL REFERENCES recipe(id),
--   occasion_id INT NOT NULL REFERENCES occasion(id)
-- );

-- CREATE TABLE grocery_list(
--      id SERIAL PRIMARY KEY,
--   recipe_id INT REFERENCES recipe(id)
-- );

-- CREATE TABLE user_info(
--      id SERIAL PRIMARY KEY,
--   email VARCHAR(50) UNIQUE,
--   password VARCHAR(20),
--   grocrey_id INT REFERENCES grocery_list(id),
--   occasion_id INT REFERENCES occasion(id)
-- );

-- CREATE TABLE recipe_ingridients(
--      id SERIAL PRIMARY KEY,
```

```sql
--   recipe_id INT REFERENCES recipe(id),
--   ingridients_id INT REFERENCES ingridients(id),
--   quantity VARCHAR(80)
-- );

-- CREATE TABLE user_recipe(
--      id SERIAL PRIMARY KEY,
--   user_info_id INT REFERENCES user_info(id),
--   recipe_id INT REFERENCES recipe(id)
-- );
```