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(data to store):

User id
Username
Password
Recipes
Ingredients
Public/private
Grocery lists info
Occasion info

Table Ideas:

-Users: store user info, each row will be a user
-Recipe: store recipe info, each row will be a recipe
-GroceryList: store recipe and ingredient info, each row will be grocery list of ingredients
-Occasion: store occasion info, each row will be an occasion
-PublicRecipe: stores user info and recipe info if public, each row will be recipe by a user
-UserGroceryList: store grocery list and user info , each row will be grocery list of ingredients for recipe
-Occasion: store user occasion info, each row will be an occasion

Relationships:

One - One:
User id/username - one user id can only have 1 username, 1 username only have 1 user id
User/GroceryList - one user can have 1 grocery list, 1 grocery list belongs to 1 user
instructions/recipe - one recipe will have 1 set of instructions, instructions for 1 recipe

One - Many:
Users/recipes - user can make multiple recipes, but a recipe can only be made by 1 user
user/occasion: user can make multiple occasions, 1 occasion belongs to one user

Many - Many:
recipes/ingredients - recipes have many ingredients, ingredients can be used in many recipes

Columns:

Users:

user_id, integer- for unique num for each user
username, varchar(25) - for string username not too long
password, text - for password hashing no char limit
email, varchar(50) - for string email shorter than 50 char

Recipe:

recipe_id, integer - for unique num for each recipe made
Owner, integer - ref to user_id from users to mark recipe creator
Name, varchar(50) - string to name recipe under 50 char
Ingredients, text - for a long list of ingredients
Steps, text - for long list of steps
Private, boolean - to select private or public listing of recipe

PublicRecipe:

Public_recipe_id, int - new id based on when added to public list
Owner, integer - ref to user_id from users to mark recipe creator
Name, varchar(50) - ref recipe_name string to name recipe under 50 char
Ingredients, text - ref ingredients for a long list of ingredients
Steps, text - ref steps for long list of steps

Occasion:

Occasion_id, int - new id based on when occasion made
Occasion_name, varchar(50) - string name of occasion limit 50 char
People_serving, int - number for how many people recommended to serve
Recipe_id, int - ref recipe_id from recipe to know which recipe
Recipe_name, varchar(50) - ref recipe_name from recipe

UserOccasion

User_occasion, int - ref user_id from users for users personal occasion
Occasion_name, varchar(50) - string name of occasion limit 50 char
People_serving, int - number for how many people recommended to serve
Recipe_id, int - ref recipe_id from recipe to know which recipe
Recipe_name, varchar(50) - ref recipe_name from recipe

GroceryList:

Recipe_id, int - ref recipe id to know what recipe ingredients from
Ingredients, text - ref from recipe to see what ingredients needed

UserGroceryList:

User_list, int - ref to user_id from user to show what users list
Recipe_id, int - ref groceryList to know what recipe ingredients from
Ingredients, text - ref from groceryList to see what ingredients needed

SQL:

```
create table users (
        user_id SERIAL primary key,
        username varchar(50),
        password text,
        email varchar(50)
 );

create table recipe (
        recipe_id serial primary key,
        owner int references users(user_id),
        name varchar(50),
        ingredients text,
        steps text,
        private boolean
 );

create table PublicRecipe (
        public_recipe_id serial primary key,
        owner int references recipe(owner),
        name varchar(50) references recipe(name),
        ingredients text references recipe(ingredients),
        steps text references recipe(steps)
 );

create table occasion (
        occasion_id serial primary key,
        occasion_name varchar(50),
        people_serving int,
        recipe_id int references recipe(recipe_id),
        Recipe_name varchar(50) references recipe(recipe_name)
);

create table UserOccasion (
        user_occasion int references users(user_id),
        occasion_name varchar(50) references occasion(occasion_name),
        people_serving int references occasion(people_serving),
        recipe_id int references recipe(recipe_id),
        recipe_name varchar(50) references recipe(recipe_name)
);
```

```
create table GroceryList (
	recipe_id int references recipe(recipe_id),
	ingredients text references recipe(ingredients)
);


create table UserGroceryList (
	user_list int references users(user_id),
	recipe_id int references recipe(recipe_id),
	ingredients text references recipe(ingredients)
);
```

Intermediate SQL:

```
INSERT INTO users(username, password, email)
	VALUES('EthanDev1', 'abc123!', 'abc123@gmail.com');


INSERT INTO recipe (owner, name, ingredients, steps, private)
	VALUES( 1, 'Baked Chicken', 'Chicken, salt, pepper, onions', 'Season Chicken and bake
at 350 degrees', false);
```