## App 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

## Brainstorming/Data:

**NEEDED DATA:**

- User ID, Username, Password, Email, First Name, Last Name, Recipe, Recipe Ingredients, Recipe Instructions, Recipe Status(Private or Public), Grocery List, User Occasions

## Table Ideas:

**Users:** User ID, User First Name, User Last Name, User Password, User Email
Users Table: Will hold all of the user information that would serve as credentials and their identifier. This will be a One to Many relationship table because it will feed information to multiple tables.

**User Occasions:** Occasion ID, Occasion Name, User ID, Recipes, Occasion Date
Occasions Table: Will hold a list of occasions that belong to certain users. This will be a One to Many relationship between a user and multiple occasions in this table.

**Recipes:** Recipe ID, Recipe Name, User ID, Status is Private, Ingredients, Instructions
Recipes Table: Will hold the name, ingredients and instructions of a recipe. It will also hold the user id to which the recipe corresponds and if it is private or public.

**User Grocery List:** Grocery List ID, Grocery List Name, User ID, Items
Grocery List Table: Will hold the list of groceries the user needed to buy which will also contain the user that created it.

## Relationships:

- One to One

- One to Many
    - Users ⇒ Recipes
    - Users ⇒ Grocery List
    - Users ⇒ Occasions

- Many to Many
    - Occasions ⇒ Recipes

## Columns:

- Users Table
    - User_id: Storing to identify each user, Using a serial  number to make each number unique.
    - User_first_name: Storing users first name to identify them to their username and password. Using a Varchar (40) to limit length.
    - User_last_name: Storing users last name to identify them to their username and password. Using a Varchar(50) to limit length.
    - User_password: Storing to have their password to check against when they sign in. Using Varchar(500) too keep long enough in case of long hashed passwords.
    - User_email: Storing to identify them and for contract or update information. Using a Varchar(60) to limit length.

- Users Occasions Table
    - Occasion_id: Storing to identify each occasion and using a serial number to make each unique.
    - Ocassion_name: Storing in order to give each occasion a name to reference with user recipes. Using Varchar(50) to limit length.
    - Occasion_user_id: Storing to connect to user that created the occasion. Using a foreign key to identify the user that created it.
    - Occasion_recipes: Storing to know which recipes the user wanted for this occasion. Using a Varchar(200) to have room to store multiple recipe names
    - Ocasion_date: Storing to have a date for the occasion and using a Date type to have an easy format to work with.

- Users Occasions Table
    - Recipe_id: Storing to give each recipe a unique id, Using Serial number to make each different
    - Recipe_name: Storing to give recipe a name, using varchar to limit name
    - Recipe_user_id: storing to link to user that created it, using a foreign key to relate to user id in the users table.

    - Recipe_is_private: Storing to give a status of private or public, using boolean and set to true in order to have each recipe start as private.
    - Recipe_instructions: storing to have instructions on how to create the recipe. Using TEXT to not limit the character count.

- User Grocery List Table
    - Grocery_list_id: Storing to give each list a unique id, Using Serial number to make each different
    - Grocery_list_name: Storing to give list a name, using varchar to limit name
    - List_user_id: storing to link to user that created it, using a foreign key to relate to user id in the users table.
    - List_items: Storing in order to have a list of all the groceries a user needs to buy, using TEXT in order to give them enough room for any amount of items