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:

- User email
- User password
- User name
- Phone number
- Gender
- Name
- Unique user-id
- Location
- User recipes
- Recipe Favorites
- Recipe posts/creation
- Recipe public/private boolean
- Individual Recipe ingredient fields
- Recipe instructions field
- Grocery list field
- Occasion id
- Occasion description
- Recipe id
- Usernames
- User email
- Passwords
- Followers
- People following
- Photos
- Bio
- Comments
- Posts
- Groups
- Joined groups
- Posts within groups
- User id

- Gender
- Age
- Phone number
- location
- Recipes
- Level of experience cooking
- cold recipes
- Hot recipes

Table Ideas:
User table:
This table will hold information about the user.

- Unique user-id
- Email
- Password
- Bio
- Gender
- Age
- Phone number
- Profile picture
- Level of experience

Recipe Table:
This table will hold information about recipes that the user has saved, liked, or created.

- Recipe ID
- Favorites
- Ingredients
- Quantities
- Cook time
- Rest Time
- Hot/Cold
- Instructions
- Rating

Post Table:

- Photos
- Content
- Title
- Time
- Date
- Author of Post

Grocery List:

This table will hold information about the ingredients required for each recipe. It will be editable by the user manually. Users will also be able to import ingredient lists to the grocery list from the recipe table.

- Grocery Items
- Grocery List ID

Occasions Table:
This table will hold information about upcoming events (holidays, etc.) and the recipes a user plans to make or share for the occasion.

- Occasion ID SERIAL PRIMARY KEY
- Occasion Description TEXT
- Occasion date TIME
- Occasion time TIME
- Photos IMAGE
- Comments REFERENCE FROM COMMENTS
- Likes INTEGER
- Shares INTEGER
- Invites INTEGER

Comment Table:
This table will hold the data necessary for users to make comments and have their comments posted and interacted with.

- Comment_id SERIAL PRIMARY KEY
- Author of comments `author INT NOT NULL REFERENCES user(user_id),`
- Content TEXT
- Comment post
- Comment Reactions (likes, etc.) INTEGER

Relationships:

- One to one
    - .
- One to many
    - User => posts
        - A single user can make many posts
    - Comments => recipe
        - A single recipe can have many comments
    - User => comments
        - A single user can make many comments
- Many to many
    - User <==> Occasion
        - An occasion can have many users involved.
        - A user can be involved on many occasions.
    - User <==> Recipe
        - A user can have many recipes.

        - A recipe can belong to or be favorited by many users
    - Recipe <==> Occasions
        - An occasion can involve many recipes
        - A recipe can be used for different occasions
    - Grocery List ⇐=> Recipe ⇐> Occasions
        - A grocery list can pull from many recipes and occasions
    - Occasions ⇐=> comments
        - An occasion can have many comments
        - A comment can be made on many occasions
    - Comments <==> posts
        - A post can have many comments
        - A comment can be made on many posts

why you'll be storing that data

and why you chose the data type you did

Columns

User

- User_id
    - This is a unique identifier for each user using the platform. This will be used for occasion invites, comments, likes, posts, etc. This will be a numeric data type to allow for random generation of non-duplicate user id's
- Name
    - This will be used to store the user's name as a string
- Email
    - This will be used to contact the user and send a newsletter and updates. Stored as a string since that's what email addresses are typically used as.
- Password
    - This will be used as a security measure for users as they sign in. Stored as a string.
- Bio
    - Used to give general information about the user. Stored as a string.
- Gender
    - Used to gather data about the user. Stored as a string
- Age
    - Used to gather data about the user. Stored as a numeric value
- Phone_number
    - Used to contact the user or verify their login. Stored as an integer.
- Profile_picture
    - Used to identify the user. Stored as a String (URL)
- Level of Experience

    - Used to share information about the user's cooking/baking experience. Stored as an integer.

Recipe

- Recipe_id
    - Used to identify each unique recipe individually. Stored as a numeric value.
- Favorites
    - Allows users to store their favorite recipes in one place and find them easily. Stored as a boolean. If true it is favorited, if false it is not.
- Ingredients
    - Informs users of the required ingredients for the recipes. Stored as a string.
- Quantity
    - Informs users of the required amount of ingredients. Stored as a float to allow for fractions of standard quantities.
- Cook_time
    - Informs users of the time required for a recipe. Stored as a float.
- Rest_time
    - Informs users of the time required for letting a recipe sit or settle. Stored as a float.
- Hot/Cold
    - Informs the user if the recipe is for a hot dish or cold dish. Stored as booleans
- Instructions
    - General instructions for the users to follow. Stored as a string.
- Rating
    - Feedback for each recipe. Stored as an integer - displayed as stars.

Occasion

- Ocassion_ID
    - Used to uniquely identify each occasion. Stored as a numeric value
- Ocassion_Date
    - Used to indicate the date of the event/occasion. Stored as a date.
- Occasion_Time
    - Used to indicate the time of the event/occasion. Stored as a time.
- Photos
    - Used to display the event. Stored as a string (URL)
- Comments
    - Used to comment on events/occasions. Stored as strings.

Post

- Post
-
- Photos
    Used to store the pictures of the recipes. Datatype is a string to manage the path of the folder where the picture is saved at.
- Content
    - Used to display the content of the comment associated with the recipe being presented. Datatype is text to manage the opinion of the users which is not predetermined.
- Title
    - Used to list the title of the recipe. We used text.
- Time
    - Used to display the time of the post. It is of data type time, which is the perfect format for this type of data.
- Date
    - Used to display the date when the post was made. The data type used was date.
- Author of post
    - Used to display the name of the person who made the comment. Type text with a length of 20 spaces to provide enough space for a standard person name.

Grocery List

- grocery_items
    - Used to count how many items will be used to create the recipe. The data type is an integer to count the items involved.
- grocery_list
    - Used to describe the elements needed to produce the recipe.

Comment

- author_of_comments
    - Used to show the owner of the comment. Stored as text.
- content

    - used to show the content of the comment. Stored as text.
- comment_post
    - used to provide feedback about the comment. stored as text
- comment_reactions
    - used to count the likes of the comment. stored as an integer because it will be counted.

```
CREATE TABLE comments (
 comment_id SERIAL PRIMARY KEY,
 content text,
 author INT NOT NULL REFERENCES user(user_id),
 post_id INT NOT NULL REFERENCES posts(post_id),
 time TIMESTAMP
);

CREATE TABLE following (
  follow_id SERIAL PRIMARY KEY,
  follower_id INT NOT NULL REFERENCES user(user_id),
  following_id INT NOT NULL UNIQUE REFERENCES user(user_id)
);

CREATE TABLE user(
  user_id SERIAL PRIMARY KEY,
  user_name VARCHAR(50),
  user_email VARCHAR(50),
  user_password VARCHAR(500),
  bio VARCHAR(1000),
  gender VARCHAR(10),
  user_phone INT,
  birthday TIMESTAMP,
  profile_pic STRING,
  experience INTEGER
);

CREATE TABLE recipes (
recipe_id SERIAL PRIMARY KEY,
ingredients TEXT,
quantities FLOAT,
```

```
cook TIME,
rest TIME,
hot BOOLEAN,
instructions TEXT,
rating FLOAT
);

CREATE TABLE occasion (
  occasion_id SERIAL PRIMARY KEY,
  occasion_date DATE,
  occasion_time TIME,
  photos VARCHAR(255),
  comments VARCHAR(255),
  likes INTEGER,
  shares INTEGER,
  invites INTEGER
);

CREATE TABLE post (
  post_id SERIAL PRIMARY KEY,
  photos VARCHAR(255),
  content VARCHAR(255),
  title VARCHAR (100),
  post_time TIME,
  post_date DATE,
  post_author INT NOT NULL UNIQUE REFERENCES user(user_id)
);

CREATE TABLE grocery_list (
  grocery_list_id INTEGER,
  grocery_items VARCHAR(255)
);
```