

## **Part 1: Conceptual Planning**

### **Step1: Brainstorming**

#### **Users:**

1. ID (serial primary key)
2. Name (Username)
3. Email
4. Password

#### **Grocery Lists:**

1. ID (serial primary key)
2. Ingredients ID (foreign key references ingredients\_id)

#### **Recipe:**

1. ID (serial primary key)
2. Recipe name
3. Public or Private
4. View (text)

#### **Instructions:**

1. ID(serial primary key)
2. Recipe ID (foreign key references recipe\_id)
3. Ingredient\_id (foreign key references ingredients\_id)

#### **Ingredients:**

1. ID(serial primary key)
2. Recipe ID (foreign key references recipe\_id)
3. Name of ingredients

#### **Reviews:**

1. ID
2. Comments
3. User ID (foreign key references user\_id)

**Occasions:**

1. ID
2. Ingredients ID (foreign key references recipe\_id)

**Image:**

1. ID (serial primary key)
2. Image\_url
3. Recipe ID (foreign key references recipe\_id)

**Step 2: Table Ideas****Users:**

This table will have information about each user.

**Recipe:**

This table will have information (info) about each recipe

**Instructions:**

This table will have info about how to prepare the meal

**Ingredients:**

This table will have all the ingredients for all the recipe

**Reviews:**

This will have user comments about the recipe

**Grocery Lists:**

This will have the selected ingredients pulled for each recipe

**Image:**

Images related to each recipe

**Occasion:**

This will have the list of different occasions for different meal prep.

## Step 3: Relationships

### One to One:

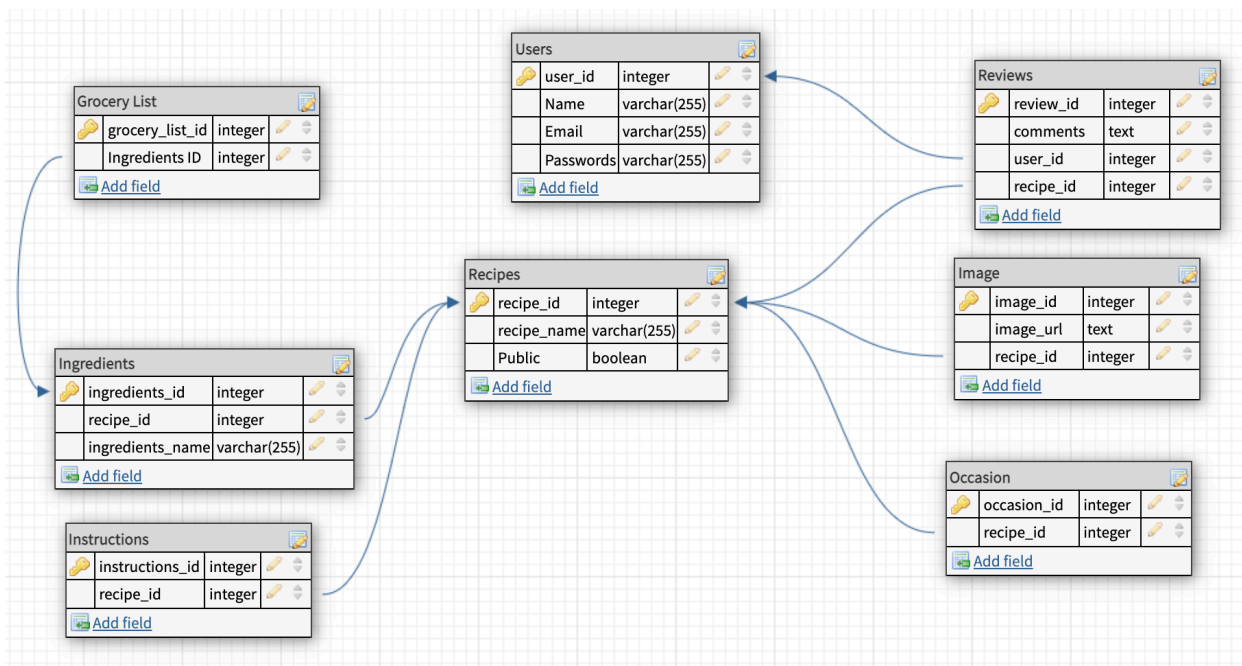
1. User to Grocery List (One user should only have One grocery list)
2. User to Image

### One to Many:

1. Recipe to Ingredients (one recipe has many ingredients)
2. User to reviews (a user can have multiple reviews for multiple recipes)
3. Recipe to Reviews

### Many to Many:

1. Recipes to Occasions (one recipe can be used for multiple occasions and one occasion can have many recipes)



## **Part 2: Table Planning - DB Designer & Pages Doc**

### **Step 1:**

#### **Columns:**

#### **Users:**

1. ID (serial primary key) INTEGER (ID is a integer number)
2. Name (Username) VARCHAR (Name is a string)
3. Email VARCHAR (STRING)
4. Password VARCHAR (STRING)

#### **Grocery Lists:**

1. ID (serial primary key) INTEGER (unique identification)
2. Ingredients ID (foreign key) INTEGER (List can access the ingredients through its ID which is an integer number)

#### **Recipe:**

1. ID (serial primary key) INTEGER
2. Recipe name TEXT
3. Public or Private BOOLEAN (either public or private requires true/false statements)
4. View (text)

#### **Instructions:**

1. ID(serial primary key) INTEGER
2. Recipe ID (foreign key)
3. Ingredient\_id (foreign key)

#### **Ingredients:**

1. ID(serial primary key) INTEGER
2. Recipe ID (foreign key) INTEGER
3. Name of ingredients TEXT (names will be alphabets and letters)

**Reviews:**

1. ID (serial primary key) INTEGER
2. Comments TEXT (needs a lot of space for comments)
3. User ID (foreign key) INTEGER

**Occasions:**

1. ID (serial primary key) INTEGER
2. Ingredients ID (foreign key) INTEGER

**Image:**

1. ID (serial primary key) INTEGER
2. Image\_url TEXT (needs a lot space for url. It is a string datatype)
3. Recipe ID (foreign key) INTEGER

## **Part 3 : Create Tables in SQL - Postgres SandBox & Pages Doc**

### **Users:**

```
CREATE TABLE Users (  
  user_id SERIAL PRIMARY KEY,  
  name VARCHAR(255),  
  email VARCHAR(255),  
  Password VARCHAR(255)  
);
```

### **Recipes:**

```
CREATE TABLE Recipe (  
  recipe_id SERIAL PRIMARY KEY,  
  recipe_name VARCHAR(255),  
  ingredients_id INTEGER NOT NULL REFERENCES  
  ingredients(ingredients_id),  
  instructions_id INTEGER NOT NULL REFERENCES  
  instructions(instructions_id),  
  occasion_id INTEGER NOT NULL REFERENCES  
  occasion(ocasion_id),  
  image_id INTEGER NOT NULL REFERENCES image(image_id),  
  recipe_privacy BOOLEAN default TRUE  
);
```

### **Ingredients:**

```
CREATE TABLE Ingredients (  
  ingredients_id SERIAL PRIMARY KEY,  
  ingredients_name VARCHAR(255)  
);
```

### **Instructions:**

```
CREATE TABLE Instructions (  
instructions_id SERIAL PRIMARY KEY,  
recipe_id INTEGER NOT NULL REFERENCES recipe(recipe_id)  
);
```

### **Reviews:**

```
CREATE TABLE Reviews (  
review_id SERIAL PRIMARY KEY,  
Comments TEXT  
);
```

### **Grocery List:**

```
CREATE TABLE Grocery List (  
grocery_list_id SERIAL PRIMARY KEY,  
ingredients_id INTEGER NOT NULL REFERENCES  
ingredients(ingredients_id)  
);
```

### **Occasion:**

```
CREATE TABLE Occasion (  
occasion_id SERIAL PRIMARY KEY,  
recipe_id INTEGER NOT NULL REFERENCES recipe(recipe_id)  
);
```

**Image:**

```
CREATE TABLE Image (  
  image_id SERIAL PRIMARY KEY,  
  image_url VARCHAR (400);  
  recipe_id INTEGER NOT NULL REFERENCES recipe(recipe_id)  
);
```

**Intermediate:****Users:**

```
INSERT INTO users(name, email, password)  
  VALUES ('Adam Smith', 'adam@yahoo.com', 'password1');
```

```
INSERT INTO users(name, email, password)  
  VALUES ('John Smith', 'john@yahoo.com', 'password2');
```

```
INSERT INTO users(name, email, password)  
  VALUES ('Evans Smith', 'evans@yahoo.com', 'password3');
```

**Recipes:**

```
INSERT INTO recipe(recipe_name,recipe_privacy)  
  VALUES ('Apple Pie', TRUE);
```

```
INSERT INTO recipe(recipe_name,recipe_privacy)  
  VALUES ('Lemon Pie', TRUE);
```



## **Ingredients:**

```
INSERT INTO ingredients(ingredients_name)  
VALUES ('Sugar');
```

```
INSERT INTO ingredients(ingredients_name)  
VALUES ('Salt');
```

```
INSERT INTO ingredients(ingredients_name)  
VALUES ('Spice');
```