F22 - Data Modeling Lab

Brainstorming:

- Username
- Email
- Password
- Ingredients
- Grocery list (contains Ingredients)
- Recipes (public or private, can be for a certain occasion)
- Instructions
- Occasions

Views:

- View recipe containing who made it, ingredients, instructions, which occasions its good for, and an add ingredients to grocery list feature.
- A sign in screen where users enter their emails and password
- A grocery list screen where users can see the ingredients they need to buy, and maybe the recipes they got them from are linked.
- A search feature where users can search by occasion, ingredients, or who authored the recipe.

Table Ideas:

- Users: Stores info about each individual user, username, email, password.
- Recipes: Holds info about each recipe, each has a row.
- Ingredients: Contains all ingredients, each in a row.
- Occasions: Contains each occasion
- GroceryList: Stores what each user has in their grocery list.
- RecipeIngredients: An association table to relate recipes and ingredients.
- OccasionRecipes: An association table that relates occasions and recipes.

Relationships:

One-to-one:

-

One-to-many:

- user/recipes - Each user can create many different recipes.
- user/occasion - Each user can create many different occasions.

Many-to-many:

- ingredients/recipes - Each recipe contains many ingredients, and each ingredient can be used for many different recipes.
- occasions/recipes - Each occasion can call for many recipes, and each recipe can be used for many occasions.
- grocery list/ingredients - Each grocery list contains many ingredients, and each ingredient can be on many people's grocery lists.

Columns:

- Users:
    - user_id integer - assigns unique number to each user used to identify them.
    - username varchar 25 - what users see and can search by. More personal than an id number.
    - email varchar 40 - stores a string of the users email.
    - password text - made text because it will store a hash that shouldn't be limited.
- Recipes:
    - recipe_id integer - unique id to track each recipe.
    - recipe_name varchar 50 - a string of the name of the dish.
    - user_id integer - an number that indicates the user who created the recipe.
    - public boolean - A true false value of whether the recipe is public or not.
    - instructions text - Chose text because the instructions for a recipe could be very long, possibly multiple pages.
- Ingredients:
    - ingredient_id integer - unique id for each ingredient.
    - ingredient_name varchar 50 - string to indicate the name of the ingredient.
- Occasions:
    - occasion_id integer - unique id for each occasion created.
    - occasion_name varchar 50 - string of the title of the occasion.
    - user_id integer - the number id of the user who created the occasion.
- GroceryList:
    - grocer_list_id integer - unique id for each grocery list item
    - user_id integer - id of the user who added the ingredient to their grocery list.
    - ingredient_id integer - id of the ingredient added.
- RecipeIngredients:
    - rec_ing_id integer - unique id of each row comparing recipe and ingredients.
    - recipe_id integer - id of the recipe
    - ingredient_id integer - id of ingredient in each recipe.
- OccasionRecipes:
    - occ_rec_id integer - unique id of each row comparing occasion and recipe.
    - occasion_id integer - id of the occasion.
    - recipe_id integer - id of the recipe.

Part 3: SQL - Create tables

```
CREATE TABLE users (
  user_id SERIAL PRIMARY KEY,
  username VARCHAR(25),
  email VARCHAR(40),
  password TEXT
  );

CREATE TABLE recipes (
  recipe_id SERIAL PRIMARY KEY,
```

```
  recipe_name VARCHAR(50),
  user_id INT UNIQUE REFERENCES users(user_id),
  public BOOLEAN,
  instructions TEXT
  );

CREATE TABLE ingredients (
  ingredient_id SERIAL PRIMARY KEY,
  ingredient_name VARCHAR(50)
  );

CREATE TABLE RecipeIngredients (
  rec_ing_id SERIAL PRIMARY KEY,
  recipe_id INT UNIQUE REFERENCES recipes(recipe_id),
  ingredient_id INT UNIQUE REFERENCES ingredients(ingredient_id)
  );

CREATE TABLE Grocerylist (
  grocery_list_id SERIAL PRIMARY KEY,
  user_id INT UNIQUE REFERENCES users(user_id),
  ingredient_id INT UNIQUE REFERENCES ingredients(ingredient_id)
  );

CREATE TABLE occasions (
  occasion_id SERIAL PRIMARY KEY,
  occasion_name VARCHAR(50),
  user_id INT UNIQUE REFERENCES users(user_id)
  );

CREATE TABLE occasionrecipes (
  occ_rec_id SERIAL PRIMARY KEY,
  occasion_id INT UNIQUE REFERENCES occasions(occasion_id),
  recipe_id INT UNIQUE REFERENCES recipes(recipe_id)
  );
```

Intermediate:

```
INSERT INTO users(username, email, password)
VALUES('Jackie', 'Jackie@gmail.com', '1234');

INSERT INTO recipes(recipe_name, user_id, public, instructions)
VALUES('Soup', 1, true, 'blah blah');
```