Grocery List Application

Brainstorming:

* table to take in our users
  + Must include a username and password
  + Another table to authenticate their passwords
* table to make recipes
  + Using ingredients or instructions
  + Ability to add ingredients from recipe to your grocery list
  + Make it so it can be public or private
  + Make it able to be viewed by other users
* table to make a grocery list
  + Able to add ingredients from a recipe
* Table for occasion
  + Table for the people invited to the occasion
  + Able to add recipes into the occasion

Table ideas:

* Users table (store info about the user)
* Authorization table (give user ability to login link to users)
* Shopping list table (allow user to create a list of ingredients and are able to get ingredients from recipes)
* Recipe table (allow user to create a public/private recipe with ingredients and instructions)
* Occasion table (allow user to make an occasion by linking occasion to user and link a recipe to it)
* Occasion users table (allow user to invite other people to their occasion by linking users and occasions)
* Occasion and recipes table (that will allow you to link recipes and occasions)

Relationships:

1. One to one
   1. Haven’t utilized this yet to my knowledge
2. Many to many
   1. More specifically an association table, used to connect the occasion and recipe tables
      1. So that the occasion and recipe tables can be linked together
   2. Another association table between occasion and users
      1. To link together their data so a user can make an occasion
3. One to many
   1. The relationship between the users table and the grocery list table
      1. Users can have multiple lists, but the list cannot have multiple users
   2. The relationship between the users table and the recipe table
      1. Users can have multiple recipes, but a recipe cannot have multiple users
   3. The relationship between the users table and the occasion table
      1. Users can make multiple occasions, but one occasion can only be made by one user

Columns:

Grocery List Users Table

**grocerylist\_user\_id**- *integer* – to make a new id for each new user (id is a number so we use integer)

**First\_name** - *varchar(255)* – to allow the user to input their name (could use a text limit so use varchar)

**Last\_name** - *varchar(255)* – to allow the user to input their last name (could use a text limit so use varchar)

**Pronouns** - *varchar*(*255*) – to allow the user to specify their pronouns (could use a text limit so use varchar)

**Profile\_pic** - *varchar*  – to allow the user to upload a profile picture (could use a text limit so use varchar)

Grocery List Authorization Table

**Grocerylist\_auth\_id** -*integer* – to have a unique auth for each user, (id is a number so it will be an integer)

**Email** - *varchar(255)* – to have a unique email for each user to login with (varchar since it is a text that needs limit)

**passwordHash** - *varchar(1000)* – to have a secure password input per user (varchar 1000 because hashed passwords are long)

**Grocerylist\_user\_id** - *varchar(1000*) – to connect to the user table (it will be a text input with a limit)

Grocery List Table

**grocery\_list\_id** - *integer* – to get a new id for each list (id is a number so it will be an integer)

**Grocery\_list\_body** - *varchar* – to make space for the user to add to the list (varchar because it will be written out text, and should have a limit)

**Grocery\_list\_ingredients** - *varchar* – to allow the user to add to the list from an existing recipe’s ingredient list, and a varchar because it will be text and should have a limit)

Grocery List Recipe Table

**Grocerylist\_recipe\_id** - *integer* - to get a new id for each new recipe (id is a number so integer)

**Recipe\_op** - *integer* - to link each recipe created to the user who created it (id since we are linking in the user id here)

**Body\_ingredients** - *text* - place for the ingredients to be listed here (text so no limit on ingredients)

**Body\_instructions** - *text* - place for the instructions to be listed here (text so no limit on instructions)

**Recipe\_private** - *boolean* - so it can be set to true if the user wants the recipe to be private

**recipe\_public** - *boolean* - so it can be set to true if the user wants the recipe to be public

Grocery List Occasion Table

**Grocerylist\_occassion\_id** - *integer* - so each new occasion has its own unique id (id is a number so integer)

**Grocerylist\_ocassion\_name** - *varchar -* so we can name the occasion with words

**grocerylist\_ocassion\_recipe** - *integer -* so we can add a recipe to the occasion (integer because we are linking ids which are numbers)

**Grocerylist\_ocassion\_op** - *integer* - so we can link the user that created the occasion using their id (integer because id is a number)

Grocery List Occasion Users Table

**Occassion\_users\_id** - *integer* - so each time a user make a new occasion it has its own unique is (since id is a number we use integer)

**Grocerylist\_user\_id** - *integer*  - so we can link to the users who are in the occasion (using id which is a number so we use integer)

**Grocerlist\_occassion\_id** - *integer* - so we can link to the occasion (using id which is a number so we use integer)

Grocery List Occasion Recipes Table

**Grocerylist\_occasion\_recipe\_id** - *integer*  - so each new recipe in an occasion has its own unique id (id is a number so we use integer)

**Grocerylist\_occasion\_id** - *integer* - so we can link to the occasion the recipe is in (use integer because we are linking with id which is a number)

**Grocerylist\_recipe\_id** - *integer* - so we can link to the recipe being put in the occasion (using id which is a number so we use integer)

SQL:

CREATE TABLE grocery\_list\_users (

grocery\_list\_user\_id SERIAL PRIMARY KEY,

first\_name VARCHAR(255),

last\_name VARCHAR(255),

pronouns VARCHAR(255),

profile\_pic VARCHAR

);

CREATE TABLE grocery\_list\_auth (

grocery\_list\_auth\_id SERIAL PRIMARY KEY,

email VARCHAR(255),

passwordHash VARCHAR(1000),

grocery\_list\_user\_id INTEGER

);

CREATE TABLE grocerylist\_recipe (

grocerylist\_recipe\_ SERIAL PRIMARY KEY,

recipe\_op INTEGER REFERENCES grocery\_list\_users(grocery\_list\_user\_id),

body\_ingredients TEXT,

body\_instructions TEXT,

recipe\_private BOOLEAN,

recipe\_public BOOLEAN

);

CREATE TABLE grocery\_list (

grocery\_list\_id SERIAL PRIMARY KEY,

grocery\_list\_body VARCHAR,

grocery\_list\_ingredients VARCHAR(1000)

);

CREATE TABLE grocery\_list\_occasion (

grocerylist\_occassion\_id SERIAL PRIMARY KEY,

grocerylist\_ocassion\_name VARCHAR,

grocerylist\_ocassion\_recipe INTEGER,

grocerylist\_ocassion\_op INTEGER REFERENCES grocery\_list\_users(grocery\_list\_user\_id)

);

CREATE TABLE grocery\_list\_occasion\_recipe (

grocerylist\_occassion\_recipe\_id SERIAL PRIMARY KEY,

grocerylist\_ocassion\_id INTEGER REFERENCES grocery\_list\_occasion(grocerylist\_occassion\_id),

grocerylist\_recipe\_id INTEGER REFERENCES grocerylist\_recipe(grocerylist\_recipe\_id)

);

CREATE TABLE grocery\_list\_occasion\_users (

occassion\_users\_id SERIAL PRIMARY KEY,

grocerylist\_user\_id INTEGER REFERENCES grocery\_list\_users(grocery\_list\_user\_id),

grocerylist\_occassion\_id INTEGER REFERENCES grocery\_list\_occasion(grocerylist\_occassion\_id)

);

Insert in data:

INSERT INTO grocery\_list\_users (first\_name, last\_name, pronouns, profile\_pic)

VALUES ('Jillian', 'Salloum', 'She/her', null);

file:///C:/Users/Jillian/Downloads/Data%20model%20lab-dbdesigner%20(1).pdf

