**RECIPE CREATING/SHARING GROCERY LIST APP**

**FEATURES**

* Food groups
* Recipe list
* Recipe instructions
* Ingredients
* Follow
* Comment
* Send Recipe
* Create shopping list
* History
* Recommended
* Tag other users
* Make posts
* Users
* Export
* Time to cook
* Serving size
* Filter by allergies

**BRAINSTORMING**

* Users can sign into the app
  + Username
  + Password
  + Email
* Users can register for the app
  + Username
  + Email
  + Password
  + Name
  + Birthday
  + Gender
  + Age
* Users can follow other users
  + Who is following who
* Users can publish recipes
  + Recipe name
  + Ingredients
  + Food group
  + Author
  + Post ID
  + Shopping list file (needed ingredients)
  + Tags
* Users can comment on published recipes
  + User
  + Text
  + Post ID
  + Timestamp
* Users can favorite recipes
  + User
  + Post ID
  + Timestamp

**TABLE IDEAS**

* Users
  + First Name
  + Last Name
  + Birth Date
  + Username
  + Password
  + User ID
  + Created when?
* Recipes
  + Author
    - Author’s page, bio, recipes posted, and posts.
  + Shopping list/ingredients
    - List of ingredients and warnings for allergies.
  + Instructions
    - Recipe name, directions on how to cook, cooking time, needed cutlery to cook it
  + Serving size
  + Complexity level
  + Cooking time
* Comments
  + User ID
  + Username
  + Timestamp
* Posts
  + Author
  + Timestamp
  + User ID
  + Comments
  + Likes
* Shopping List
  + Ingredients
* Favorites
  + Recipes
* Sort By
  + Allergies
  + Food group
  + Breakfast, lunch, dinner
  + Cooking time
  + Complexity level
  + Serving size
* Messages
  + Username
  + User ID
  + Timestamp
  + Text body
  + Reply
  + Block
  + Report

**RELATIONSHIPS**

* One to one
  + Users to authors
    - Because an author is a user
* One to many
  + Recipe to ingredients
    - Because recipes have more than one ingredient
  + Users to posts
    - Because users can post more than once, but one post can’t have multiple users.
  + Users to comments
    - Because users can comment, but comments do not have more than one user.
  + Posts to comments
    - Because one post can have multiple comments, but one comment cannot be on multiple posts.
  + Recipes to comments
    - See above
  + Shopping list to ingredients
    - One shopping list can have more than one ingredient, but an ingredient is never simultaneously in multiple shopping lists.
  + Recipe to ratings
    - One recipe has multiple ratings. But one rating is not applied multiple times to the same recipe (by a single user).
  + User to ratings
    - A user can rate recipes, but no one can rate a user.
  + User to favorites.
    - One user can have multiple favorites, but it is impossible for a favorite to contain a user.
* Many to many
  + Recipe to ingredients
    - Because one recipe has multiple ingredients, but one ingredient cannot contain multiple recipes within itself.
  + Recipes to authors
    - Recipes can have multiple authors, and an author can be attached to multiple recipes.

CREATE TABLE users (

user\_id SERIAL PRIMARY KEY,

firstname VARCHAR(255) NOT NULL,

lastname VARCHAR(255) NOT NULL,

birthdate DATE NOT NULL,

username VARCHAR(10) NOT NULL,

password VARCHAR(255) NOT NULL

);

CREATE TABLE recipes (

recipe\_id SERIAL PRIMARY KEY,

ingredients VARCHAR(100) NOT NULL,

instructions VARCHAR(10000) NOT NULL,

servingsize INTEGER NOT NULL,

complexitylevel VARCHAR(255) NOT NULL,

cookingtime TIME NOT NULL,

allergies VARCHAR(255) NOT NULL,

foodgroup VARCHAR(255) NOT NULL,

mealtype VARCHAR(255) NOT NULL,

recipename VARCHAR(255) NOT NULL

);

CREATE TABLE comments (

user\_id INTEGER NOT NULL REFERENCES users(user\_id),

username VARCHAR(255) NOT NULL REFERENCES users(username),

timestamp DATE TIME NOT NULL,

post\_id INTEGER NOT NULL REFERENCES posts(post\_id)

);

CREATE TABLE ingredients (

ingredient\_id SERIAL PRIMARY KEY,

ingredient VARCHAR(255)

);

CREATE TABLE shopping\_list (

username VARCHAR (255) NOT NULL REFERENCES users(username),

user\_id INTEGER NOT NULL REFERENCES users(user\_id),

ingredient VARCHAR(255) NOT NULL REFERENCES ingredients(name)

);

CREATE TABLE sort\_by (

servingsize INTEGER NOT NULL REFERENCES recipes(servingsize),

complexitylevel VARCHAR (255) NOT NULL REFERENCES recipes(complexitylevel),

cookingtime TIME NOT NULL REFERENCES recipes(cookingtime),

allergies VARCHAR (255) REFERENCES recipes(allergies),

food\_group VARCHAR(255) REFERENCES recipes(foodgroup)

);

CREATE TABLE favorites (

recipe\_id INTEGER NOT NULL REFERENCES recipes(recipe\_id),

recipes VARCHAR (255) NOT NULL REFERENCES recipes(name)

);

CREATE TABLE posts (

username VARCHAR(255) NOT NULL REFERENCES users(username),

timestamp TIME NOT NULL,

user\_id INTEGER NOT NULL REFERENCES users(user\_id),

likes INTEGER NOT NULL,

text VARCHAR(9999),

post\_id SERIAL PRIMARY KEY

);

CREATE TABLE messages (

user\_id INTEGER NOT NULL REFERENCES users(user\_id),

timestamp TIME NOT NULL,

username VARCHAR(255) NOT NULL REFERENCES users(username),

text VARCHAR(9999)

);