Brandon Loveall and Hyrum Sorensen

Brainstorming

USER TABLE:
User id
Username (email)
Password hash

RECIPES:
Recipe id
Creator id
Recipe name
Ingredients id
Recipe instructions
Public or private

INGREDIENTS TABLE:
Ingredient id
Ingredient name

GROCERY LIST TABLE:
Grocery id
Owner id
Ingredients id

OCCASIONS TABLE
Occasion id
Occasion member ids
Occasion recipe id

BRIDGE TABLES:
INGREDIENTS/RECIPES TABLE
INGREDIENTS/GROCERY LIST TABLE
OCCASIONS/USERS TABLE

Relationships:

Occasions ⇐Many⇒ Users
Ingredients ⇐ Many ⇒ Recipes
Ingredients ⇐ Many ⇒ Grocery Lists

Occasions can have multiple users and vice versa. Ingredients can belong to many grocery lists/recipes and recipes/grocery lists can have multiple ingredients.

User ⇒Many⇒ Recipes

User ⇒One⇒ Grocery list

Our SQL Code:

```
CREATE TABLE "users" (
  "user_id" SERIAL PRIMARY KEY,
  "email_address" VARCHAR(50),
  "password_hash" VARCHAR(1000)
);

CREATE TABLE "recipes" (
  "recipe_id" SERIAL PRIMARY KEY,
  "name" VARCHAR(50),
  "creator_id" INT NOT NULL,
  "private" BOOL
);

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

CREATE TABLE "grocery_list" (
  "grocery_list_id" SERIAL PRIMARY KEY,
  "owner" INT NOT NULL
);

CREATE TABLE "occasions" (
  "occasion_id" SERIAL PRIMARY KEY,
  "occasion_date" VARCHAR(20),
  "occasion_location" VARCHAR(100),
  "occasion_description" VARCHAR(2000),
  "owner_id" INT NOT NULL
);

CREATE TABLE "recipe_ingredients" (
  "recipe_ingredients_id" SERIAL PRIMARY KEY,
  "recipe_id" INT NOT NULL,
```

```
  "ingredient_id" INT NOT NULL
);

CREATE TABLE "list_ingredients" (
  "list_ingredients_id" SERIAL PRIMARY KEY,
  "grocery_list_id" INT NOT NULL,
  "ingredient_id" INT NOT NULL
);

CREATE TABLE "user_occasions" (
  "user_occasions_id" SERIAL PRIMARY KEY,
  "occasion_id" INT NOT NULL,
  "user_id" INT NOT NULL
);

ALTER TABLE "recipes" ADD FOREIGN KEY ("creator_id") REFERENCES "users" ("user_id");

ALTER TABLE "grocery_list" ADD FOREIGN KEY ("owner") REFERENCES "users" ("user_id");

ALTER TABLE "occasions" ADD FOREIGN KEY ("owner_id") REFERENCES "users"
("user_id");

ALTER TABLE "recipe_ingredients" ADD FOREIGN KEY ("recipe_id") REFERENCES "recipes"
("recipe_id");

ALTER TABLE "recipe_ingredients" ADD FOREIGN KEY ("ingredient_id") REFERENCES
"ingredients" ("ingredient_id");

ALTER TABLE "list_ingredients" ADD FOREIGN KEY ("grocery_list_id") REFERENCES
"grocery_list" ("grocery_list_id");

ALTER TABLE "list_ingredients" ADD FOREIGN KEY ("ingredient_id") REFERENCES
"ingredients" ("ingredient_id");

ALTER TABLE "user_occasions" ADD FOREIGN KEY ("occasion_id") REFERENCES
"occasions" ("occasion_id");

ALTER TABLE "user_occasions" ADD FOREIGN KEY ("user_id") REFERENCES "users"
("user_id");
```