Features

- users can sign into the app with their email and password
- users can create recipes with ingredients and instructions
- recipes can be marked as public or private
- users can view other people's recipes
- ingredients from recipes can be added to user's grocery lists
- users can create their own occasions and assign recipes to occasions

TABLE IDEAS

User: This holds information about the user. Just standard stuff

Recipes: each row will be a list of ingredients with instructions that a user created

Ingredients: Each row is an ingredient

Grocery list: Each row will contain

- User
  - User ID → SERIAL PRIMARY KEY - because unique user ID
  - User email → VARCHAR because text field
  - User password →VARCHAR because text field
- Recipes
  - Recipe ID → SERIAL PRIMARY KEY so we can call each recipe individually
  - User ID → FOREIGN KEY - derives from other table
  - Instructions → VARCHAR because text field
  - Ingredient ID → FOREIGN KEY - derives from other table
  - Ingredient Amount → INT because it will be an integer
  - Public or private→ BOOLEAN for false or true
- Ingredients
  - ingredient ID→ SERIAL PRIMARY KEY
  - Name → VARCHAR because text field
- Grocery list
  - List item id → SERIAL PRIMARY KEY
  - Ingredient ID → FOREIGN KEY - derives from other table
  - User id → FOREIGN KEY - derives from other table
- Occasions(events)
  - Event ID → SERIAL PRIMARY KEY
  - Event name → VARCHAR because text field
  - Recipe ID → FOREIGN KEY - derives from other table
  - Event location → VARCHAR because text field

Relationships:

- One to one - Only one User ID per grocery list item id

- One to many - These values only feed into one other table respectively
  - User → Recipe
  - User → Grocery List
  - Ingredients → Grocery list
- Many to many - Grocery list is an association table fed by user table and ingredient table
  - User → Ingredients
  - Ingredients → Recipes
  - Recipe ← → Occasion

TABLE CODE:

```
CREATE TABLE "user" (
        user_id SERIAL PRIMARY KEY,
        first_name VARCHAR(50),
        last_name VARCHAR(50),
        user_email VARCHAR (50),
        user_password VARCHAR (100)
);
```

```
CREATE TABLE ingredients (
        ingredient_id SERIAL PRIMARY KEY,
         ingredient_name VARCHAR(40)
);
```

```
CREATE TABLE recipe (
        recipe_id SERIAL PRIMARY KEY,
 instructions text,
 isPublic BOOLEAN,
  user_id INTEGER NOT NULL REFERENCES "user"(user_id),
 ingredient_quantity INT
);
```

```
CREATE TABLE grocery_list (
            list_id SERIAL PRIMARY KEY,
  ingredient_id INTEGER NOT NULL REFERENCES ingredients(ingredient_id),
  user_id INTEGER NOT NULL REFERENCES "user"(user_id)
);


CREATE TABLE occasion (
            event_id SERIAL PRIMARY KEY,
  event_name VARCHAR(30),
  user_id INTEGER NOT NULL REFERENCES "user"(user_id),
  event_location VARCHAR(40),
  event_time DATE
);

CREATE TABLE occasion_recipe (
            occres_id SERIAL PRIMARY KEY,
            Event_id INTEGER NOT NULL REFERENCES occasion(event_id),
            Recipe_id INTEGER NOT NULL REFERENCES recipe(recipe_id)
);

CREATE TABLE recipe_ingredient (
            rec_ingredient_id SERIAL PRIMARY KEY,
            ingredient_id INTEGER NOT NULL REFERENCES ingredients(ingredinet_id),
            recipe_id INTEGER NOT NULL REFERENCES recipe(recipe_id)
);
```

GENERATED FROM DBDESIGNER

```
CREATE TABLE "public.user" (
	"user_id" serial NOT NULL,
	"user_email" varchar(75) NOT NULL,
	"user_password" varchar(75) NOT NULL,
	"first_name" varchar(50) NOT NULL,
	"last_name" varchar(50) NOT NULL,
	CONSTRAINT "user_pk" PRIMARY KEY ("user_id")
) WITH (
  OIDS=FALSE
);



CREATE TABLE "public.recipes" (
	"recipe_id" serial NOT NULL,
	"instructions" varchar(500) NOT NULL,
	"isPublic" BOOLEAN NOT NULL,
	"ingredients_id" int NOT NULL,
	"user_id" int NOT NULL,
	"igredient_quantity" int NOT NULL,
	CONSTRAINT "recipes_pk" PRIMARY KEY ("recipe_id")
) WITH (
  OIDS=FALSE
);



CREATE TABLE "public.grocery list" (
	"list_item_id" serial NOT NULL,
	"ingredient_id" int NOT NULL,
	"user_id" int NOT NULL,
	CONSTRAINT "grocery list_pk" PRIMARY KEY ("list_item_id")
) WITH (
  OIDS=FALSE
);



CREATE TABLE "public.occasions" (
	"event_id" serial NOT NULL,
	"event_name" varchar(255) NOT NULL,
	"event_recipe" varchar(255) NOT NULL,
	"user_id" int NOT NULL,
	"event_location" varchar(255) NOT NULL,
	"event_time" DATE NOT NULL,
	CONSTRAINT "occasions_pk" PRIMARY KEY ("event_id")
) WITH (
  OIDS=FALSE
);



CREATE TABLE "public.ingredients" (
	"ingredient_id" serial NOT NULL,
	"name" varchar(255) NOT NULL,
	CONSTRAINT "ingredients_pk" PRIMARY KEY ("ingredient_id")
) WITH (
  OIDS=FALSE
);




ALTER TABLE "recipes" ADD CONSTRAINT "recipes_fk0" FOREIGN KEY
("ingredients_id") REFERENCES "ingredients"("ingredient_id");
ALTER TABLE "recipes" ADD CONSTRAINT "recipes_fk1" FOREIGN KEY
("user_id") REFERENCES "user"("user_id");

ALTER TABLE "grocery list" ADD CONSTRAINT "grocery list_fk0" FOREIGN
KEY ("ingredient_id") REFERENCES "ingredients"("ingredient_id");
ALTER TABLE "grocery list" ADD CONSTRAINT "grocery list_fk1" FOREIGN
KEY ("user_id") REFERENCES "user"("user_id");

ALTER TABLE "occasions" ADD CONSTRAINT "occasions_fk0" FOREIGN KEY
("event_recipe") REFERENCES "recipes"("recipe_id");
ALTER TABLE "occasions" ADD CONSTRAINT "occasions_fk1" FOREIGN KEY
("user_id") REFERENCES "user"("user_id");
```