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

Brainstorming

- Email
- Password
- Recipe
- Ingredients
- Instructions
- Public (true/false)
- Grocery list
- Occasion

Table Ideas	**Relationships**

• USERS
  ○ First_name
  ○ Last_name
• AUTH
  ○ Email
  ○ Hash
  ○ fk-users(user_id)
• RECIPE
  ○ Name
  ○ About
  ○ Public
  ○ fk-ingredient(name)
  ○ fk-ingredient(amount)
• INGREDIENT
  ○ Name
  ○ Amount
• GROCERY
  ○ fk-ingredient(name)
  ○ fk-ingredient(amount)
• OCCASION
  ○ fk-recipe(name)
	ONE TO ONE
	• fk-recipe(name)
	• fk-users(user_id)
	  ○ One instance makes a singular call
	
	ONE TO MANY
	• fk-ingredient(name)
	• fk-ingredient(amount)
	  ○ One instance can make multiple calls to ingredient
	
	MANY TO MANY

```
CREATE TABLE "public.users" (
      "user_id" serial NOT NULL,
      "first_name" varchar(255) NOT NULL,
      "last_name" varchar(255) NOT NULL,
      CONSTRAINT "users_pk" PRIMARY KEY ("user_id")
) WITH (
  OIDS=FALSE
);
```

```
CREATE TABLE "public.auth" (
      "auth_id" serial NOT NULL,
      "user_id" integer NOT NULL UNIQUE,
      "email" varchar(255) NOT NULL UNIQUE,
      "hash" TEXT NOT NULL,
      CONSTRAINT "auth_pk" PRIMARY KEY ("auth_id")
) WITH (
  OIDS=FALSE
);
```

```
CREATE TABLE "public.recipe" (
      "recipe_id" serial NOT NULL,
      "name" varchar(255) NOT NULL,
      "about" varchar(255) NOT NULL,
      "public" BOOLEAN NOT NULL,
      "user_id" integer NOT NULL,
      "ingred_amount" integer NOT NULL,
      "ingred_name" varchar(255) NOT NULL,
      CONSTRAINT "recipe_pk" PRIMARY KEY ("recipe_id")
```

```
) WITH (
  OIDS=FALSE
);



CREATE TABLE "public.ingred" (
     "ingred_id" serial NOT NULL,
     "name" varchar(255) NOT NULL,
     "amount" integer NOT NULL,
     CONSTRAINT "ingred_pk" PRIMARY KEY ("ingred_id")
) WITH (
  OIDS=FALSE
);



CREATE TABLE "public.grocey" (
     "grocery_id" serial NOT NULL,
     "ingred_name" varchar(255) NOT NULL,
     "ingred_amount" integer(255) NOT NULL,
     CONSTRAINT "grocey_pk" PRIMARY KEY ("grocery_id")
) WITH (
  OIDS=FALSE
);



CREATE TABLE "public.occasion" (
     "occasion_id" serial NOT NULL,
     "recipe_name" integer NOT NULL,
     CONSTRAINT "occasion_pk" PRIMARY KEY ("occasion_id")
```

```
) WITH (
  OIDS=FALSE
);
```

```
ALTER TABLE "auth" ADD CONSTRAINT "auth_fk0" FOREIGN KEY ("user_id")
REFERENCES "users"("user_id");
```

```
ALTER TABLE "recipe" ADD CONSTRAINT "recipe_fk0" FOREIGN KEY
("user_id") REFERENCES "users"("user_id");

ALTER TABLE "recipe" ADD CONSTRAINT "recipe_fk1" FOREIGN KEY
("ingred_amount") REFERENCES "ingred"("amount");

ALTER TABLE "recipe" ADD CONSTRAINT "recipe_fk2" FOREIGN KEY
("ingred_name") REFERENCES "ingred"("name");
```

```
ALTER TABLE "grocey" ADD CONSTRAINT "grocey_fk0" FOREIGN KEY
("ingred_name") REFERENCES "ingred"("name");

ALTER TABLE "grocey" ADD CONSTRAINT "grocey_fk1" FOREIGN KEY
("ingred_amount") REFERENCES "ingred"("amount");
```

```
ALTER TABLE "occasion" ADD CONSTRAINT "occasion_fk0" FOREIGN KEY
("recipe_name") REFERENCES "recipe"("name");
```