- 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/Table Ideas

Users table
- User_name
- User_email
- User_password

Recipes table
- Recipe_name
- Ingredients_ids
- Instructions
- Public_Boolean
- Author
- Recipe_type(breakfast, lunch, dinner)
- Diet_tag

Ingredients table
- Ingredients_id
- Price_per_unit
- Unit

Quantity table
- Quantity_id

Groceries Table
- Ingredients_id
- Quantity
- Recipe_id
- User_id
- price

Occasions table
- Occasion_name
- Occasion_date
- Recipie_name
- user_id

Relationship Types
- One to one
  - Quantity => ingredient
- One to many
  - user=>recipes
  - recipe=>ingredients
  - Grocery trip => recipes
  - Grocery Trip => ingredients
  - user=> grocery trips
  - recipes=>quantities
- Many to Many
  - Recipes => occasions

```
CREATE TABLE users(
 user_id SERIAL PRIMARY KEY,
 user_name VARCHAR(40) NOT NULL,
 user_email VARCHAR(50) NOT NULL,
 user_password VARCHAR(500) NOT NULL
);

CREATE TABLE recipes(
 recipe_id SERIAL PRIMARY KEY,
 is_public BOOLEAN DEFAULT true,
 recipe_name VARCHAR(100) NOT NULL,
 recipe_type VARCHAR(25) NOT NULL,
 diet_tag VARCHAR(50),
 author_id INTEGER NOT NULL REFERENCES users(user_id),
 instructions VARCHAR(25000) NOT NULL
);

CREATE TABLE recipeingredients(
 recipeingredient_id SERIAL PRIMARY KEY,
 recipe_id INTEGER NOT NULL REFERENCES recipes(recipe_id),
 ingredient_id INTEGER NOT NULL REFERENCES ingredients(ingredient_id),
 quantity INTEGER NOT NULL,
 price FLOAT DEFAULT (ingredients(price_per_unit)*recipeingredients(quantity))
);

CREATE TABLE ingredients(
 ingredient_id SERIAL PRIMARY KEY,
 ingredient_name VARCHAR(50) NOT NULL,
 unit_of_measurement VARCHAR(20) DEFAULT 'Whole',
```

```sql
   price_per_unit FLOAT NOT NULL
);

CREATE TABLE groceries(
  trip_id SERIAL PRIMARY KEY,
  user_id INTEGER NOT NULL REFERENCES users(user_id),
  recipe_id INTEGER NOT NULL REFERENCES recipes(recipe_id),
  price FLOAT DEFAULT (SELECT SUM(price) FROM recipeingredients WHERE
groceries(recipe_id)=recipeingredients(recipe_id))
);

CREATE TABLE occasions(
  occasion_id SERIAL PRIMARY KEY,
  occasion_name VARCHAR(50) NOT NULL,
  occasion_date DATE NOT NULL,
  user_id INTEGER NOT NULL REFERENCES users(user_id)
);

CREATE TABLE occasionrecipes(
  occasionrecipes_id SERIAL PRIMARY KEY,
  occasion_id INTEGER NOT NULL REFERENCES occasions(occasion_id),
  recipe_id INTEGER NOT NULL REFERENCES recipes(recipe_id)
);
```