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

Users

- User_id
- Username
- Email address
- Public recipes
- Private recipes
- Grocery lists

User_password

- User_password_id
- user_id(from user table)
- password

Ingredients

- Ingredients for recipes

Recipes

- Instructions
- Store ingredients
- Public or private (bool)
- rating

Groceries

- Hold lists from users

Events

- Create events
- Put recipes on certain occasions

Relationships

One-to-one

- User id => password (one user to one password)
- User_email => user
- Username => user
- Instructions => recipe (one recipe would have one set of instruction)

One-to-many

- User => recipes (user have multiple recipes)
- User => events (user can have multiple events)

- Recipes => ingredients (recipes can have multiple ingredients)
- Events => users (many users can attend one event)

Many-to-many

- Users => rating (many users can rate many recipes)
- Users => ingredients (many users can buy many ingredients)

LAB:

```
CREATE TABLE users(
user_id SERIAL PRIMARY KEY,
username VARCHAR(30),
email_address VARCHAR (50)
);

CREATE TABLE user_password(
user_password_id SERIAL PRIMARY KEY,
user_id INT NOT NULL REFERENCES users(user_id),
password VARCHAR(40)
);

CREATE TABLE ingredients(
ingredients_id SERIAL PRIMARY KEY,
ingredients_name VARCHAR(50),
ingredients_price FLOAT
);

CREATE TABLE recipes(
recipes_id SERIAL PRIMARY KEY,
recipes_name VARCHAR(100),
recipes_private BOOLEAN,
user_id INT REFERENCES users(user_id),
ingredients_id INT REFERENCES ingredients(ingredients_id)
);

CREATE TABLE groceries(
groceries_id SERIAL PRIMARY KEY,
groceries_name VARCHAR(50),
user_id INT REFERENCES users(user_id),
ingredients_id INT REFERENCES ingredients(ingredients_id)
);

CREATE TABLE events(
events_id SERIAL PRIMARY KEY,
user_id INT REFERENCES users(user_id),
recipes_id INT REFERENCES recipes(recipes_id),
event_name VARCHAR(50),
```

```
event_time VARCHAR(5)
);
```

LAB EXPLANATION:

```
CREATE TABLE users(
user_id SERIAL PRIMARY KEY, -Creates user id
username VARCHAR(30), -limits username to 30 characters
email_address VARCHAR (50) -limits email to 50 characters
);

CREATE TABLE user_password(
user_password_id SERIAL PRIMARY KEY, -creates password id
user_id INT NOT NULL REFERENCES users(user_id), - references user_id to get correct pw
password VARCHAR(40) -limits pw to 40 characters
);

CREATE TABLE ingredients(
ingredients_id SERIAL PRIMARY KEY, -creates ingredients id
ingredients_name VARCHAR(50), - limits ingredient names to 50 characters
ingredients_price FLOAT - gives ingredients price that can be decimals
);

CREATE TABLE recipes(
recipes_id SERIAL PRIMARY KEY, -creates recipe id
recipes_name VARCHAR(100), -limits recipe name to 100 characters
recipes_private BOOLEAN, - can make recipes private or public by switching between T & F
user_id INT REFERENCES users(user_id), - references user id to save recipe to users
ingredients_id INT REFERENCES ingredients(ingredients_id) - references ingredients to save
ingredients to recipe
);

CREATE TABLE groceries(
groceries_id SERIAL PRIMARY KEY, - creates groceries id
groceries_name VARCHAR(50), - limits groceries name to 50 characters
user_id INT REFERENCES users(user_id), - references user id to save groceries to user
ingredients_id INT REFERENCES ingredients(ingredients_id) -references ingredients id to save
it to groceries
);

CREATE TABLE events(
events_id SERIAL PRIMARY KEY, - create event id
user_id INT REFERENCES users(user_id), -saves events to user
recipes_id INT REFERENCES recipes(recipes_id), -saves recipes to events
event_name VARCHAR(50), -limit event names to 50 characters
event_time VARCHAR(5) - limits event time to 5 characters
```

);