Grant Roszhart

Meal Master

Re-Design Thoughts

I have not made any additions or removals, nor any changes to existing service layer descriptions. I am comfortable where this document is at currently, although changes will still be possible and likely going forward.

(Please see bottom of document for Stretch Goal service layer description)

Introduction

For this project, I have elected to use Express from Node.js. I feel most comfortable with this backend system, and it works well with PostgreSQL, my database choice. I will be utilizing a RESTful API, and will be hosting the app using the tool Heroku. I have found Heroku to be a common choice for many good reasons. The system will function utilizing routes and controllers to convey data between the front end and PostgreSQL database. Below are the MVP endpoints for this project.

Recipe Endpoints

Retrieve Recipes

Request: GET /api/recipes

Description: Retrieves a list of recipes, displaying them to the user on the front end

Success Example:

Grant Roszhart


```
{
    "status": "success",
    "data": [
      {
        "recipeId": 1,
        "name": "Spaghetti",
        "description": "A classic Italian pasta dish.",
        "category": "italian, pasta",
        "ingredientsID": "23, 64",
        "servings": "2"
      }
    ]
}
```


(This would go on to show all the recipes available)

Fail Example:


```
{
    "status": "error",
    "message": "Recipes not found"
}
```


Service Diagram:

Filter Recipes

Request: GET /api/recipes?category=pasta

Description: Allows users to filter list of recipes by category. In this example, we are filtering to only display recipes with the category of Pasta.

Success Example:

Grant Roszhart


```
{
    "status": "success",
    "data": [
      {
        "recipeId": 1,
        "name": "Spaghetti",
        "description": "A classic Italian pasta dish.",
        "category": "italian, pasta",
        "ingredientsID": "23, 64",
        "servings": "2"
      },
      {
        "recipeId": 42,
        "name": "Macaroni and Cheese",
        "description": "Macaroni noodles with delicious cheesy sauce.",
        "category": "italian, pasta",
        "ingredientsID": "14, 24, 12, 5, 1",
        "servings": "1"
      }
    ]
}
```


Fail Example:


```
  {
    "status": "error",
    "message": "At least one category must be selected."
}
```


Service Diagram:

Retrieve Single Recipe

Request: GET /api/recipes/:recipeid

Description: Allows the front end to retrieve details of one specific recipe when the user clicks on it to view it.

Success Example:

Grant Roszhart


```
{
    "status": "success",
    "data": [
      {
        "recipeId": 1,
        "name": "Spaghetti",
        "description": "A classic Italian pasta dish.",
        "category": "italian, pasta",
        "ingredientsID": "23, 64",
        "servings": "2"
      }
    ]
}
```


Fail Example:


```
{
    "status": "error",
    "message": "Recipe not found"
}
```


Service Diagram:

Add Recipe to Scheduled Meals

Request: POST /api/scheduledmeals/mealid/recipes

Description: Allows the user to add a recipe they selected to their scheduled meals.

Success Example:

Request


```
{
    "recipeId": 1,
    "date": "2023-11-05",
    "servings": 1
}
```


Response

```
{
    "status": "success",
    "message": "Recipe added successfully"
}
```

Fail Example:
Request

```
{
    "recipeId": 1234567890,
    "date": "2023-11-05",
    "servings": 1
}
```

Response

```
{
    "status": "error",
    "message": "Recipe ID not found"
}
```

Service Diagram:

Inventory Endpoints

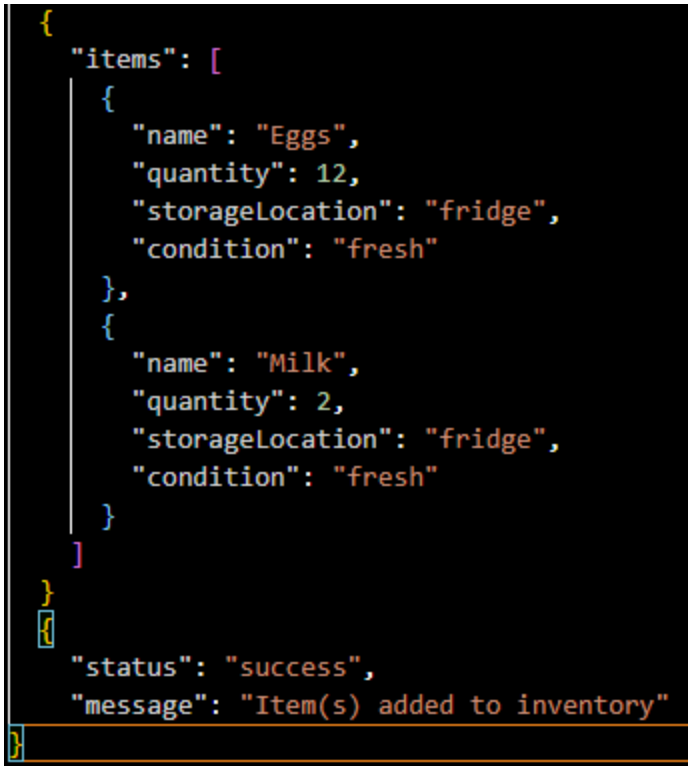
Add Inventory

Request: POST /api/users/userid/inventory

Description: Allows users to add items to their inventory.

Success Example:

Grant Roszhart



Fail Example:

Service Diagram:

Grant Roszhart

Remove Inventory

Request: DELETE /api/users/userid/inventory

Description: Removes item from user inventory.

Success Example:

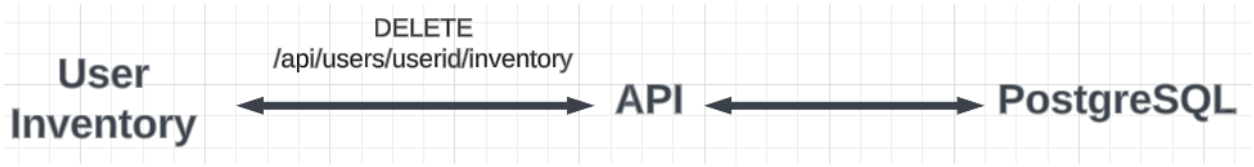
```
{
  "items": [
    {
      "name": "Eggs",
      "quantity": 1,
    }
  ]
}
{
  "status": "success",
  "message": "Item successfully removed"
}
```

Fail Example:

```
{
  "items": [
    {
      "name": "Eggsss",
      "quantity": 1,
    }
  ]
}
{
  "status": "error",
  "message": "Item name not found"
}
```

Service Diagram:

Grant Roszhart



User Endpoints

User Login

Request: POST /api/users/login

Description: Allows users to log into their account with username and password.

Success Example:

```
{
  "email": "user@example.com",
  "password": "password"
}

{
  "status": "success",
  "message": "Login successful",
  "user": {
    "userId": 123,
    "username": "user12",
    "email": "user@example.com"
  }
}
```

Fail Example:

```
{
  "email": "user@example.com",
  "password": "badPassword"
}

{
  "status": "error",
  "message": "Password entered is incorrect."
}
```

Service Diagram:

Grant Roszhart

Register User

Request: POST /api/users/register

Description: Allows users to register a new account if they haven't yet made one or would like to make another.

Success Example:

```
{
  "username": "username",
  "email": "user@example.com",
  "password": "Password"
}
{
  "status": "success",
  "message": "Registration successful",
  "user": {
    "userId": 1,
    "username": "username",
    "email": "user@example.com"
  }
}
```

Fail Example:

```
  {
    "username": "username",
    "email": "user@example.com",
    "password": "Password"
  }
{
    "status": "error",
    "message": "Email already registered to account."
}
```

Service Diagram:

Grant Roszhart

Stretch Goal Endpoint

Upload Recipe

Request: POST /api/recipes/upload

Description: Allows users to upload their own recipes to the database for others to use

Success Example:

```
{
  "name": "Pepperoni Pizza",
  "description": "A classic pizza with pepperoni topping.",
  "ingredients": [
    "Pizza dough",
    "Tomato sauce",
    "Cheese",
    "Pepperoni"
  ],
  "instructions": [
    "Preheat the oven to 450°F.",
    "Roll out the pizza dough on a floured surface.",
    "Spread tomato sauce over the dough.",
    "Sprinkle cheese and add pepperonis and any other desired toppings.",
    "Bake for 15-20 minutes or until the crust is golden brown."
  ],
  "servings": 4,
  "user": "username"
}
```

Grant Roszhart


```
{
  "status": "success",
  "message": "Recipe uploaded successfully",
  "recipe": {
    "recipeId": 12345,
    "name": "Pepperoni Pizza",
    "description": "A classic pizza with pepperoni topping.",
    "user": "username"
  }
}
```


Fail Example:


```
{
  "name": "",
  "description": "A classic pizza with pepperoni topping.",
  "ingredients": [
    "Pizza dough",
    "Tomato sauce",
    "Cheese",
    "Pepperoni"
  ],
  "instructions": [
    "Preheat the oven to 450°F.",
    "Roll out the pizza dough on a floured surface.",
    "Spread tomato sauce over the dough.",
    "Sprinkle cheese and add pepperonis and any other desired toppings.",
    "Bake for 15-20 minutes or until the crust is golden brown."
  ],
  "servings": 4,
  "user": "username"
}
```



```
{
  "status": "error",
  "message": "Recipe upload failed.",
  "errors": {
    "name": "Recipe name is required."
  }
}
```


Service Diagram:

Grant Roszhart