**Taode Ogden**

**IGME430 Project 1 Documentation**

**Purpose:**

A simple cooking recipe API that users can add their own recipes to. Using an ‘Adder’ window, they fill out a form that is compiled into a JSON object that is stored in the server to be retrieved by the Index page, which displays all added recipes. Users can search for specific recipes by name. Users can click on a recipe in the list to access a separate page that formats that recipe’s data in a pleasing format.

**Data:**

When the user clicks on the ‘Submit’ button, they send a JSON object to the server with the following properties:

**Name:** The name of the recipe, which is also the key for the entire object

**authorName:** The name of the user who submitted the recipe, entered manually

**ingredientList:** An array of small JSON objects that each contain the ingredient’s name and quantity.

**instructList:** An array of strings, in order of when to do each step of the recipe.

**Linkname**: the recipe’s name, with all space characters replaced by ‘+’s. This is needed for query.parse to retrieve the correct parameters.

**Above and Beyond:**

* The ability for users to favorite recipes, which would be tracked in local storage
* Each recipe page has a form for users to be able to add comments
* When a user adds a recipe, they can attach images.

**Progress:**

The core of the application is completed. There were some stretch goals planned, but there were time constraints with the project. So far, the user can:

1. Submit recipes to be stored in the server via the form in the ‘Add’ page
2. View all stored recipes in the ‘View Recipes’ page
3. and get more detailed page about a specific recipe by clicking its link.

**Borrowed Code:**

The parseBody method was borrowed from the namesake demo provided in class.

There are other snippets borrowed from Stack Overflow, which are noted in the code comments.

**Endpoints:**

**/**

Supported Methods: GET, HEAD

Desc: equivalent to /adder.html; this is the default page the user enters when starting up the app

Return Type: HTML Doc

**/adder.html**

**Supported Methods:** GET, HEAD

**Desc:** sends users to the ‘Add’ page, where they submit recipes that go to the server

**Return Type:** HTML Doc

**/index.html**

**Supported Methods:** GET, HEAD

**Desc:** sends users to the ‘View Recipes’ page, where they can peruse through every recipe stored in the server

**Return Type:** HTML Doc

**/viewRecipe?name = <recipe name>**

**Supported Methods:** GET, HEAD

**Desc:** Loads a template page that will be filled with data from the specified recipe

**Return Type:** HTML Doc

**/getRecipeList?name= <name input>**

**Supported Methods:** GET

**Desc:** Returns a JSON containing all recipes. If a string was entered for the name parameter, returns all recipes that contain the string (not case sensitive) instead.

**Return Type:** JSON

**/getRecipe?name = <recipe’s linkname>**

**Supported Methods:** GET

**Desc:** Returns the JSON object for the specified recipe. The JSON data will be displayed onto the page.

**Return Type: JSON**