

# Recipe Recommender Source Documentation

## Frontend Source files

### **App.js - ReactJS Class**

App Component is the main component in React which acts as a container for all other components. This is the root component which calls child component - Form.js & RecipeList.js. Each of these child components return data back to App.js after they process information.

Instance variables :

1. Ingredients - is a set data structure. No duplicate ingredients are allowed
2. recipeList - is array that store all recipe details

Functions:

1. getRecipeDetails
  - Param - {String} recipeName
  - Return - none
  - Functionality - This method gets all recipes from given ingredients by querying MongoDB
2. handleSubmit
  - Param - {array} ingredientInput
  - Return - none
  - Functionality - This method is a callback function of Form component which sets instance variable - Ingredients in App.js
3. constructor
  - Param - none
  - Return - none
  - Functionality - initialize default values to instance variables

### **Form.js - ReactJS Class**

Form Component takes user inputs and sends to back to App/Root component

Instance variables :

Ingredients - is a set data structure. No duplicate ingredients are allowed

Functions:

1. handleSubmit
  - a. Param - event
  - b. Return - none
  - c. Functionality - This method is called by App.js and returns user inputs
2. constructor
  - a. Param - none

- b. Return - none
  - c. Functionality - initialize default values to instance variables
- 3. printHandler
  - a. Param - event
  - b. Return - Html list of ingredients
  - c. Functionality - This method displays the ingredients
- 4. addHandler
  - a. Param - event
  - b. Return - none
  - c. Functionality - This method updates ingredients set
- 5. removeHandler
  - a. Param - event
  - b. Return - none
  - c. Functionality - This method removes ingredients set
- 6. render
  - a. Param - none
  - b. Return - none
  - c. Functionality - constructs UI of Form

### **RecipeList.js - ReactJS functional Component**

This component is parent component which calls Recipe component

params: recipe list returned by mongoDB

return : UI Structure of all recipes to be displayed

### **Recipe.js - ReactJS functional Component**

This component structures UI style of a Recipe

params: recipe

return : UI Structure of a recipe

### **Header.js - ReactJS functional Component**

This component structures title of website

params: none

return : UI Structure of a header of website

### **Backend Source files**

### **index.js**

This component Index.js , establishes the MongoDB connection.

### **server.js**

This component server.js, connects to the server using the uri mentioned.

### **.env**

This component, .env contains the DB Uri and the port number to connect to the MongoDB

### **recipesDAO.js**

This component, recipesDAO contains function to retrieve the recipes based on ingredients.

Functions:

#### **1.injectDB**

- a. Param - conn
- b. Return - none
- c. Functionality - connects to the DB

#### **2.getRecipes**

- a. Param - none
- b. Return – recipesList, totalNumRecipes
- c. Functionality – Retrieve the recipes based on the ingredients

#### **3.getCuisines**

- a . Param - none
- b. Return - cuisines
- c. Functionality – Retrieves the list of cuisines

### **recipes.controller.js**

This component retrieves the result using the functions in recipesDAO

Functions:

#### **1.apiGetRecipes**

- a. Param – req, res, next
- b. Return - none
- c. Functionality - retrieves the response using getRecipes in recipesDAO.

#### **2.apiGetRecipeCuisines –**

- a. Param - req, res, next

- b. Return – none
- c. Functionality – - retrieves the response using getRecipes in recipesDAO.

## **recipes.route.js**

This component creates a new router object to handle the requests