

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
3. cuisine – stores the cuisine type input from the user
4. email – stores the email id, if input by the user
5. flag – stores the value of ‘checked’ of the checkbox on the UI form. If checked equals ‘true’, an email with the list of recipes is sent to the user.

Functions:

1. getRecipeDetails
 - Param - {String} ingredient, cuis, mail, flag
 - Return - none
 - Functionality - This method gets all recipes from given ingredients by querying MongoDB. Sends the data to the ‘/recipes’ api and starts the backend processes.
2. handleSubmit - handles the stream of actions immediately after clicking ‘search recipes’. Calls getRecipeDetails
 - Param - {array} formDict – stores the input details from the form
 - 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

Dict – Dictionary that stores all of the input details from the form

Functions:

1. handleSubmit
 - a. Param – event
 - b. Return - none
 - c. Functionality - This method is called by App.js and returns all of the user inputs through the variables dict
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 and the recipe instructions after clicking search recipes.
4. addHandler
 - a. Param - event
 - b. Return - none
 - c. Functionality - This method updates ingredients set when adding an ingredient
5. removeHandler
 - a. Param - event
 - b. Return - none
 - c. Functionality - This method removes ingredients from the set upon clicking on the ingredient
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 Mongo database

2.getRecipies

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

3.getCuisines

- a . Param - none
- b. Return – RecipeList and totalNumRecipes
- c. Functionality –
 - i Stores data from the filters variable
 - ii Creates regex search string for querying the mongo database based on the user's inputs
 - iii Stores the query results and formats them
 - iv If user checked the 'send email' checkbox, sends an email to them containing their recommended recipes using nodemailer

4. mail_param.js

Stores the authentication password for sending the emails through nodemailer

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 - calls recipeDAO and passes it the filters variable, then retrieves the response containing recipes using getRecipes in recipesDAO.

2.apiGetRecipeCuisines –

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

recipes.route.js

This component creates a new router object to handle the requests