General Rules:

Codes:

* 1: Request Successful
* 0: Simple Error
  + Usually print error as popup
* -1: Bad Request, not good

GENERAL CALLS

/getRecipes

This call occurs first right when app starts. Gets first 30 recipes from the database for the home page

* HTTP Type: GET
* Requirements
  + None
* Response output
  + {recipes: [] }
    - a json object with a recipes array

/getRecipeByID

This call occurs on the start of each recipe page being loaded on the browser. It gathers all the recipe data based on the recipe id. To populate the page with.

* HTTP Type: GET
* Requirements
  + ID -> recipe integer id
* Response output
  + {name,prep\_time,cooking\_time,style\_id,image\_location,instruction,ingredients [] }
    - all the data that the recipe holds from the database, including an ingredients array of jsons {ingredient\_name,quantity}

/createUser

This call is when you want to create a new user. This call will handle if the username is taken. Maybe later that the user email is registered.

* HTTP Type: POST
* Requirements
  + user\_name
  + user\_password
  + user\_email -> email of user
* Response output
  + {status, message}
    - ex. status: 1, message: Login Successful
    - a status based on one of the codes, and a message for helpful debugging and output

/login

This call is when the user wants to login. The call will return a session token to use for other calls.

* HTTP Type: POST
* Requirements
  + user\_name
  + user\_password
* Response output
  + {status,message, data: {user\_token: \_\_\_}}
    - the server will generate a session token and pass that as a data json inside the response json

/logoff

This call is when the user wants to log off

* HTTP Type: POST
* Requirements
  + user\_token -> user specific key
* Response output
  + {status, message}