Contents

[Topic: Visualizing Recipe Data 1](#_Toc29632831)

[Proposal 1](#_Toc29632832)

[Questions this data visualization will answer 1](#_Toc29632833)

[Goal 1](#_Toc29632834)

[Requirements 1](#_Toc29632835)

# Topic: Visualizing Recipe Data

Github: <https://github.com/mlaska/Project2.git>

Data Set: <https://api.edamam.com/search?q=&app_id=d6f4ebaf&app>

Inspiration: <https://github.com/mlaska/Project2/blob/master/Homepage%20Visual.vsdx>

Resources:

Caloric Intake Data-<https://www.accessdata.fda.gov/scripts/InteractiveNutritionFactsLabel/factsheets/Whats_On_The_Nutrition_Facts_Label.pdf>

Proposal: Have you ever wanted to make a recipe, but weren’t sure just how good for you that recipe really is for your health? Maybe you have a favorite food, but have trouble determining just how many servings are good for you or you really wish you could figure out maybe that favorite could be healthy with just a few slight alterations.

Our project will focus on utilizing recipe data to help the user visualize and understand how each ingredient contributes to their daily nutrition goals as well as help users to explore ways in which they can modify a recipe to meet their specific needs.

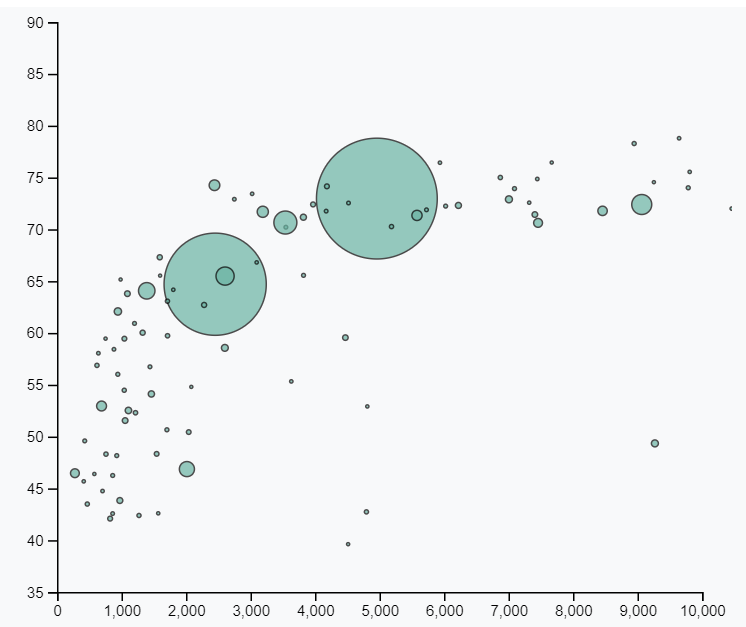
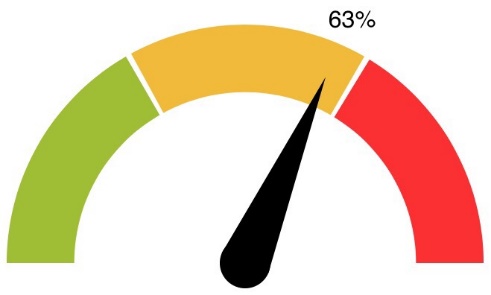
Questions this data visualization will answer:

* Which recipes contain my required nutrient consumption?
* Are there recipes that are vegan/gluten friendly?
* Which ingredients contribute materials that I would consider unhealthy or very healthy?

Goal**:** to combine data from known ingredients and their nutrition information with recipes that are dynamically searched by the user.

Rationale:

First, in our search for data, we have found that there isn’t a single source of truth for information on recipe information as well as the information on the individual ingredients.

* Data:
  + We are going to focus on using an API to identify the ingredients within a recipe. This method will allow a more dynamic search that will provide a wide array of recipes rather than having the database used be static.
    - Note: We are planning on having a static database of recipes meeting the 100 minimum in case there are any issues with the API pulling recipe data.
  + Nutrition information by ingredient will be stored in a static database that will be used to pull nutrition information to fill our visualizations
* Interactions:
  + We hope to have visual providing a list of ingredients for the recipe searched that is interactive on click and hover
    - Upon clicking the ingredient, we hope to have the graphs change to show a nutrition breakdown of that ingredient
  + We hope to have a visual with a cumulative visual of all the ingredients within the recipe showing a breakdown of the nutrition information
    - Upon clicking a nutrient in that graph, we hope to have a new graph provide a breakdown by ingredient of that nutrient
* Visualizations:
  + D3.js Bubble Graphs with ingredients are going to use 3 data points x, y and circle diameter.
    - * <https://www.d3-graph-gallery.com/bubble>
    - X axis will depict the ordered nutrient (order of FDA list) or ingredient (order of Recipe List)
    - Y axis will depict the percent of the total daily recommended (Center of bubble higher means closer to 100% of max total)
    - Bubble size diameter will be determined by the weight of the ingredient or nutrient (Grams?)
    - 
  + D3.js Gauge Graph
    - * <https://hashnode.com/post/d3-gauge-with-needle-in-vuejs-citse28000dd4u553hjfyh9l9>
    - Gauges will be generated to show the percentage of the nutrient total for one serving for all ingredients in the recipe
  + 
* Assumptions:
  + Data will analyzed using FDA recommended Daily Values
  + Data will be referenced by a single serving of the recipe
* Java Scripting languages
  + We hope to utilize Angular, but this is untested. Our expectation is to use this framework with our project but this may change as we explore
  + <https://angular.io/>

# Requirements

1. Your visualization must include a Python Flask–powered RESTful API, HTML/CSS, JavaScript, and at least one database (SQL, MongoDB, SQLite, etc.).
   1. Database type = Object Orient=
      1. Nutrition information for ingredients by weight
   2. HTML type will be sourced in the GitHub repository
   3. API = Freeware to pull recipe lists and information
2. Your project should fall into one of the below four tracks:
   1. A custom “creative” D3.js project (i.e., a nonstandard graph or chart)
   2. A combination of web scraping and Leaflet or Plotly
   3. A dashboard page with multiple charts that update from the same data
   4. A “thick” server that performs multiple manipulations on data in a database prior to visualization (must be approved)
3. Your project should include at least one JS library that we did not cover.
4. Your project must be powered by a data set with at least 100 records.
5. Your project must include some level of user-driven interaction (e.g., menus, dropdowns, textboxes).
6. Your final visualization should ideally include at least three views.

# Meeting Notes

2020/01/11

Cindy is going to start the coding to fill out the list of ingredients pulled from the API

* + Keep in mind that you will be pulling data from a database
  + Thoughts for the list to be dynamically created in the Js file

Ingris is going to be working with Cindy to focus on the ingredient list and also creating a drop down menus

* + Thoughts to start with using a data base already generated by the API with recipe information

David is going to build a function to call the ingredients API based on the recipe String

* + He is going to clean up the recipe API call for Michael to work on
  + He is going to focus on making a reproducible call using a singular ingredient that
  + Working on how to use a search function to create javascritpt load the data to the html file

Michael work on taking API code to pull the recipe and store the data and add it to a location

* Goal to be able to use the API call to then create a document store that can be used as a repository when searched
* Streamline Python into a definition file that can be called by python and java scripting