I chose a dataset that consists of the nutritional values of each item on the McDonald's menu. Specifically, my main objective of the visualization was for the nutrient-conscious users to be able to easily see the breakdown of the major nutrients in McDonald's menu. While the official online menu provided information regarding the calories, organizing the distribution of calories into each of the macronutrients and micronutrients can say a lot more about the health benefits and risks of an item.

In my initial design, I wanted to fulfill the user's intent to explore the data by letting the user select their desired y-axis to help compare their chosen nutrition to the relative amount of calories in the menu item. Specifically, this can help them infer that: the more left and up a point is, the higher the value in the y-axis they chose in relation to the amount of calories that is contained in the food. Furthermore, by selecting different categories of items, the users can filter the scatterplot points by the category in the McDonald's menu. This acts to fulfill the user intent of filtering data such that the users can more easily consciously choose their food from their desired category. Furthermore, I want the users to be able to query through item names such that, for example, if a user wishes to get an item with egg, they can get a whole selection of menu items that most likely contain eggs as inferred from the name description. Then, by specifically choosing an item, the pie chart will change the data to show the nutrition distribution accordingly. Lastly, the user can also interact between the charts. By clicking on a point in the scatterplot, the user can view the pie chart distribution of the corresponding menu item. This makes choosing from a category and comparing the nutrition and calories more accessible.

From the objective of showing nutrient distribution for a menu item, my first consideration was the query filter based on a specific item. In this case, a search bar seems to be the best choice to rapidly filter through item names and allow the user to select accordingly. Additionally, to show what item was selected, the search bar can maintain the input when the user clicks on a particular item. On the other hand, the choice of buttons seems to work best for categories and filtering. This is because buttons will more easily allow the reinforcementment of what was selected. Specifically, to indicate the current selection, I wanted the button to change to a more eye-catching color. Hence, to be more conscious of color disabilities while maintaining the greatest eye-catching level, I make the selected category button to have a red border and change to full opacity. Additionally, to further reinforce the selection, changing the category will also change the title of the scatterplot to indicate the corresponding category. Similarly, in the pie chart, the reinforcement also exists where the title will change according to the selection of displayed items. Specifically, when a user clicks on a point in the scatterplot, the search bar will also change to display the name of the selected point in the search bar as well as changing the title of the pie chart.

From the point of view of Dynamic Query Filters principle from the paper review, reversibility was implemented through the ability to undo the filtering of the scatterplot. In particular, by having an "all" button, the user can choose to go back to including all points in the scatterplot after they have filtered based on some choices.

During the design process, I realized that some food items will cause the labels on the pie chart to overlap in a way that may make the words harder to understand. However, since making a legend can mean that the users will have to make the extra effort to match between the legend and the pie chart colors, I chose to allow the overlap in hope for quicker comparisons for the users. Then, to increase the readability of overlaps, I decreased the font size of the labels such that the chance of overlapping can decrease. Furthermore, the original idea for the y-axis of the scatterplot was for the user to choose a nutrient category and compare the number of calories coming from that category. However, I soon realized that the dataset only specified the number of grams coming from a specific nutrient category. While I was going to have the serving size be the x-axis of the scatter plot instead, the lack of uniform units in serving size makes that infeasible. Despite this, since the number of grams of a nutrient can usually be correlated to the number of calories, this data visualization can still serve its purpose.

Throughout development, the parts that were hardest and took the most time were updating the pie chart in accordance with a search bar that is responsive when clicked and that hides and shows the list of possible options from the user's input. Since I have never implemented a search bar nor an animated pie chart, the updates of arc and consideration of hiding and showing the dropdown menu was a process of trials and tests.