Use Case Diagram

This diagram displays the viable actions that the customer can take while interacting with our program. For the stage 2 demo they can both collapse/expand the headers of the data groupings and view the four filters: hours, funding, count, domain, and data (full implementation by final submission. These actions satisfy the customer’s request for the second iteration of our system. Our system present the user with a choice of a pie chart or a bar chart to display the data found in the csv. It also visually maps the data flow from the import of the CSV file to its eventual display. As demonstrated by the diagram, the tree of nodes is created from the parsed. This node is then accessed in the View. The node can be altered by clicking a lower level of the tree – changing the data that will be graphed. The resultant graph is displayed on the right side of the screen. By default, the graph will generate at the highest level of the tree node.

We chose to employ the model-view-controller design pattern. This means our system is split into three main parts: View, Controller, and Model. This modularity separates the display and the data processing which promotes system flexibility. Our design also focuses on encapsulation to hide system complexities in different layers. Our upper layers, such as view, do not see the data processing layers underneath. Encapsulation is critical to Object-Oriented design, it makes debugging and future implementations much easier in addition to making the program more readable.