

Specification

Below is a list of all of the requirements for this project, which have been split into the following sections:

- Overview
- Data collection
- Data processing and storage
- Output
- Graphs
- Other

Overview (based on the project description)

- An extremely dynamic data source must be used.
- The data must be processed in a meaningful way.
- A visualization or multiple visualizations must be produced.
- The visualizations must maintain a view of the current data while also updating to show the newly collected data.

Data collection

- Twitter's API will be used as the data source for this project.
- A library will be used to connect to the API to reduce risk of issues connecting to the API both initially and if the API were to change.
- Any filters used when collecting the data from twitter will be easily modifiable so that the content of the dashboard output can be easily modified when required.

Data processing and storage

- Data collection and processing will be handled separately to ensure that data collection is not impacted by data processing.
- The vast amount of data gathered will be processed to reduce the size of data storage whilst still managing to contain all data required for all visualizations that can be produced.

- The data will be stored in an efficient way, to ensure that excessive storage is not required to keep the application running.

Output

- All data used for the visualization will be retrieved from the data store used within the application to ensure that the data collection from Twitter is not impacted by running of the application if many dashboards are running at once.
- Each graph should be able to be added to a cell and updated using a single function call.
- It should be possible to remotely access the dashboard.
- It should be possible to have multiple dashboards running at once.
- A visualization container library would be created so that the output of the application would be easily and highly configurable using a JSON file. A possible structure is shown below.

```
{
  "cols": 3,
  "rows": 2,
  "cells": [{
    "height": 1,
    "width": 2,
    "function": "cellOneFunction()"
  }, {
    "height": 2,
    "width": 1,
    "function": "cellTwoFunction()"
  }, {
    "height": 1,
    "width": 1,
    "function": "cellThreeFunction()"
  }, {
    "height": 1,
    "width": 1,
    "function": "cellFourFunction()"
  }]
}
```

Graphs

- Number of tweets per second (early deliverable).
- Number of tweets per location, shown on a heat map overlaid on a world map.
- Compare number of tweets about multiple topics at once.

- Attempt to predict the rise and fall of currently trending Twitter topics.
- Create a Venn diagram between the followers, retweets and favorites of a user or tweet.
- Show increase or decrease of followers including current trend, i.e. rising or falling.
- A graph / tree diagram showing the interaction of users on a tweet. The root node being the initial tweet, and sub nodes being retweets / favorites.
- (Other visualizations could be added if interesting ideas are thought of throughout development of the dashboard.)

Other

- The project will be split into multiple modular systems to the point where if required, just a one module could be switched out without having a negative impact on the functionality of the application without any changes being necessary. For example, switching the data store module to use Google Data Store rather than a local database.