

# Requirement Statement

Sameer Al Harbi

October 2022

## 1 Introduction

The aim of this project is to create a web-based, multi-dimensional plotter to provide a visualisation tool to allow users to explore massive, multi-dimensional data sets. Visualising large, complex data sets is a common requirement and modern graphics cards can be used to speed up the rendering to provide a fast, visualisation tool with additional effects used to represent different data attributes with web technologies such as WebGL or the higher level three.js. This project would suit a student who was interested in three-dimensional computer graphics, web applications, visualisation of large, complex datasets and programming for graphics cards. The project could extend the plotter to further dimensions by adding additional graphical effects to represent additional information in the data set, obtaining and presenting data in real-time. (From Original Project Brief)

This document serves as an expansion of the original brief introducing the project and aims to serve as a central "client requirement" from which agile techniques act on. The original brief is very brief as can be seen above in its entirety. This document aims to add more guided information as identified by the author on how he specifically aims to approach the originally stated project. This new guide of sorts is not set in stone and does not expect to have any significance beyond helping kick start development and help with documenting the project timeline in the report.

## 2 MVP / Minimum Viable Product

The goal is to create a Web-based viewer that allows a user to import a data source in JSON, which is then used to plot one of multiple 3D graph options select able. The application should automatically change up the vertex and fragment shaders to correctly render the chosen graph or otherwise have general options that work on all. Scale, units and value of render must always be visible within the rendered scene.

## 3 Extensions

On top the MVP mentioned above, there is also potential to extend the project with the following features:

### 3.1 Extra Large

- **External Connection to database** Consider either integrating a local database to allow the user to store and use very large data stores from within the app, or adding support for external cloud services.
- **GPU Accelerated Data Analysis** Research into the potential of using the GPU to speed up the mathematical functions run on data.

### 3.2 Large

- **Moveable Camera** Allow the user to freely move the camera around the scene (or alternatively change to predefined positions) when viewing the generated graph. with all labels being updated to be visible and accurate
- **Advanced Graphs** have the option to show geo spatial data on a map view or to have other advanced data views that don't fit on a standard axis bound graph
- **Layer System** As identified in the research phase, consider the ability to mix and match view layers to create a custom view that shows certain parts of the data from a single or multiple sources.

### 3.3 Medium

- **Data management** Add the ability to filter or otherwise manage and cleanup data before it is rendered

### 3.4 Small

- **Image Taking** Add the ability to take an image of the generated graph and it export it in a set composition