CS416 Software Engineering

Authors/Requirement Engineers:
Ekenechukwu Nwannunu, Adam Richard

Team 8:
Arjun Agrawal, Joe Cullinan, Geonhyuk Im, Brett
Malmquist, Ben Miloshoff, Ekenechukwu Nwannunu,
Adam Richard, Christian Schubert

Undergraduate Project

Software Requirements Specification

Document

Version: 1.1   **Date: (02/17/2021)**

Table of Contents

1. Introduction

1.1 Purpose

This is an overview of the software requirements for the development of a data visualization system. The document is specifically tailored to the development team, maintenance staff, testers, technical writers, support people, customers, and users.

1.2 Scope

This subsection expands upon the specifications of the software products to be produced. These include:

General Outline as follows:

(1) *Software product(s) to be produced by name*
(2) *What the software product(s) will, and, if necessary, will not do*
(3) *Describe the application of the software being specified, including relevant benefits, objectives, and goals.*

Develop a web-based data visualization system. The system shall generate various diagrams and charts out of user-specified JSON data. The system should contain multiple HTML files written in HTML, CSS, and JavaScript (with the Webix UI library and potentially other JavaScript GUI libraries).

1.3 Definitions, Acronyms, and Abbreviations.

contract: A legally binding document agreed upon by the customer and supplier. This includes the technical and organizational requirements, cost, and schedule for a product. A contract may also contain informal but useful information such as the commitments or expectations of the parties involved.

customer: The person, or persons, who pay for the product and usually (but not necessarily) decide the requirements. In the context of this recommended practice the customer and the supplier may be members of the same organization.

JSON (JavaScript Object Notation): a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate.

supplier: The person, or persons, who produce a product for a customer. In the context of this recommended practice, the customer and the supplier may be members of the same organization.

user: The person, or persons, who operate or interact directly with the product. The user(s) and the customer(s) are often not the same person(s).

1.4 References

IEEE Std 830-1998, Software Requirements Specifications.

CS406 Software Engineering Course Project Description

This information can be provided by reference to an appendix or to another document.

1.5 Overview

The rest of the document will contain information regarding the overall descriptions and specifications of the project.   They are to be sectioned into two parts:
Overall description and Specific Requirements.   The Overall description section will cover a description of the interfaces, constrains, functions, and overall characteristics of the project. The Specific Requirements sections will cover the various technologies to be used in the project.

2. The Overall Description

The general factors that affect the product and its requirement is the interactive interface(display), functionality of the website, and the accuracy of the information intended to be displayed to the user.

2.1 Product Perspective

Using the Webix UI library and potentially other JavaScript GUI libraries(I.E Plotly), this product is intended to be a web-based data visualization system. Displaying various diagrams and charts using the user's input JSON data.

2.1.1 System Interfaces

2.1.2 Interfaces

(1) The logical characteristics of each interface between the software product and its
    users.
   - GUI: navigate wireframes
       - Visualization system
       - Charts
       - Diagrams
   -
(2) All the aspects of optimizing the interface with the person who must use the
    system
   - hyperlinks to navigate page to page

The system shall generate various diagrams and charts out of user-specified JSON data.

2.1.3 Hardware Interfaces

The user will input data for the graph and display it back to the user. Ideally, this is
supported on any type of computer, this will be done by the user utilizing the hardware
components to input the data. Examples of this would be the keyboard, monitor, and a
mouse.

2.1.4 Software Interfaces

The user is only required to have a web browser in order to use this application.
Web Browser is used for:
   (1) Search and download the website in order to use the program.

2.1.5 Communications Interfaces

2.1.6 Memory Constraints

2.1.7 Operations

2.1.8 Site Adaptation Requirements

Users should have any working operating system installed to be able to use the program.

2.2 Product Functions

The major functions of this product are:

(1) Building visualization systems
(2) Generate various diagrams and charts out of user-specified JSON data
(3) A series of displayed information to explain the functionality of the program

2.3 User Characteristics

User characteristics may include but are not limited to:

(1) College-educated
(2) Data analysts/scientists
(3) Students

2.4 Constraints

If any file repository services are to be used by the students, such as Github and Google Drive, the access permissions shall be granted only to the students that participate in this project, the course instructor, and BeulahWorks. Github (or other repository services) repositories must be created as private repositories. Any violations of the IP restrictions may lead to legal ramifications.

2.5 Assumptions and Dependencies

The assumption would be that the user has a system and web browser.

2.6 Apportioning of Requirements

Requirements that may be further developed on is functionality, interface, processing of user-input data.

3. Specific Requirements

1. Website
    a) A set of related web pages located under a single domain name, typically produced by a single person or organization.

2. Web-based(HTML, CSS, Javascript)
    a) HyperText Markup Language (HTML), Cascading Style Sheets (CSS), and JavaScript are the languages that run the web.
3. GUI
    a) A GUI (graphical user interface) is a system of interactive visual components for computer software.
4. Wireframes
    a) a two-dimensional skeletal outline of a webpage or app.
5. Data visualization display system
    a) A graphical representation of information and data.

3.1 External Interfaces

The user will input:
1) Menu
    a) Choices for the wanted type of chart for the user to choose from.
2) JSON window
    a) Space where users input JSON code to display information in chart form.

The interface will output:
1) Diagrams
    A. Histogram
    B. Boxplot
    C. Barchart
    D. Scatter Plot
    E. Stacked Bar Chart
    F. Grouped Boxplot
2) Wireframes
    a) Windowed boxes with explanations of the expected functionality of the GUI.

3) Menu
    a) Choices displayed by the system of multiple visualizations.

3.2 Functions

The system shall require valid JSON input from the user for the correct graph/images to be displayed. If invalid input is given, an error message will be displayed.

3.3 Performance Requirements

99 % of the operations shall be processed in less than 1 second

3.4 Logical Database Requirements

The information that is going to be stored for this program is:
1. R dataset CO2 sample

3.5 Design Constraints

Design constraints are limited too:
- Only using the Webix UI library and/or other JavaScript GUI libraries.
- Must be tested through Selenium IDE.

3.5.1 Standards Compliance

This SRS document needed to be written with IEEE format.

3.6 Software System Attributes

The system should be a self relying program that accurately assesses the user's input and displays the output in graphical form. Goals of the program should be informative and educational on displaying CO_2 levels.

3.6.1 Reliability

In order to make sure that the program is reliable, the project will be tested to read user input correctly, build the diagrams, verify the input.

3.6.2 Availability

The system will be available 24/7 since the system is a website.

3.6.3 Security

3.6.4 Maintainability

The program will be maintained through the Selenium IDE.

3.6.5 Portability

The program will be highly portable since it is a website so all the user needs is access to a laptop or computer.

ID	Characteristic	H/M/L	1	2	3	4	5	6	7	8	9	10	11	12
1	Correctness	H												
2	Efficiency	M												
3	Flexibility	L												
4	Integrity/Security	L												
5	Interoperability	L												
6	Maintainability	H												
7	Portability	H												
8	Reliability	H												
9	Reusability	H												
10	Testability	H												
11	Usability	H												
12	Availability	H												

Definitions of the quality characteristics not defined in the paragraphs above follow.

- Correctness - extent to which program satisfies specifications, fulfills user's mission objectives
- Efficiency - amount of computing resources and code required to perform function
- Flexibility - effort needed to modify operational program
- Interoperability - effort needed to couple one system with another
- Reliability - extent to which program performs with required precision
- Reusability - extent to which it can be reused in another application
- Testability - effort needed to test to ensure performs as intended
- Usability - effort required to learn, operate, prepare input, and interpret output

3.7 Organizing the Specific Requirements

3.7.1 System Mode

3.7.2 User Class

The system supports all user classes.

3.7.3 Objects

- MainMenu.html
- Histogram.html

- Boxplot.html
- BarChart.html
- ScatterPlot.html
- StackedBarChart.html
- GroupedBoxplot.html

3.7.4 Feature

1. Navigation buttons to different pages
   a. Buttons with click event handler on each page
2. Accept user input
   a. HTML form with textarea that user can type into
3. Transform input to JSON
   a. drawGraph() function can pull data from textarea and transform to json with parse() function
4. Build graph
   a. Combination of data extracted from the JSON and used as a parameter in the newPlot() function in drawGraph() function
5. Display graph
   a. newPlot() function in drawGraph() function
6. Error handling
   a. Try/catch block in drawGraph() catching exception and displaying the message in HTML

3.7.5 Stimulus

3. 7.6 Response

3.7.7 Functional Hierarchy

3.8 Additional Comments

3.3 Performance Requirements
3.4 Design Constraints
3.5 Software system attributes
3.6 Other requirements

Appendix

Starting the website is the initial step and is achieved by typing in the URL. The application will start from there.

Start website application

When website is launched, we will provide multiple options for which chart you can have visualized sush as a histogram and boxplot

Choose type of chart that is to be visualized

Once the user has selected which chart they want to see the program shall ask the user to input their code in order to convert to a diagram.

Enter JSON code to draw diagrams

Code is designed to make sure that the users input is valid and can be converted properly.

Verify users input

Once the code has been verified, the program will convert the JSON code in order to be visualized for the user.

Build the diagram

Once the code is and all previous conditions have been met, the program shall display the desired diagrams for the user.

Display diagram output

User

System