DSME

Data Structures Made Easy
Technical Specification

Project Number – P33

Feargal Karney – 10362551
Sean Smith – 10339513

Contents

Chapter 1
Introduction

1.1. Overview

Data Structures Made Easy (DSME) is a website that implements a variety of information based on various data structures and algorithms. DSME is intended to be used as a teaching aid for users who desire to learn data structures and algorithms that are implemented within computer related applications. Sources of information within DSME include online documentation, interactive graphical applications, screencasts, downloadable content and Java programs.

Downloadable content is provided for the user in a PDF format. DSME intends to incorporate a professional standard in regards to the information it provides. Downloadable content is based on the documentation of data structures and algorithms represented within the website.

DSME attempts to implement the optimal teaching aid as satisfactory as possible for the computing community. In cases where explanations of a specific data structure or algorithm is regarded as unclear for the user; in this situation DSME provides a graphical application in a simplistic user-friendly format. Differences of implementation techniques of various animations are documented in this specification.

1.2. Glossary

DSME
: An acronym for the website Data Structures Made Easy.

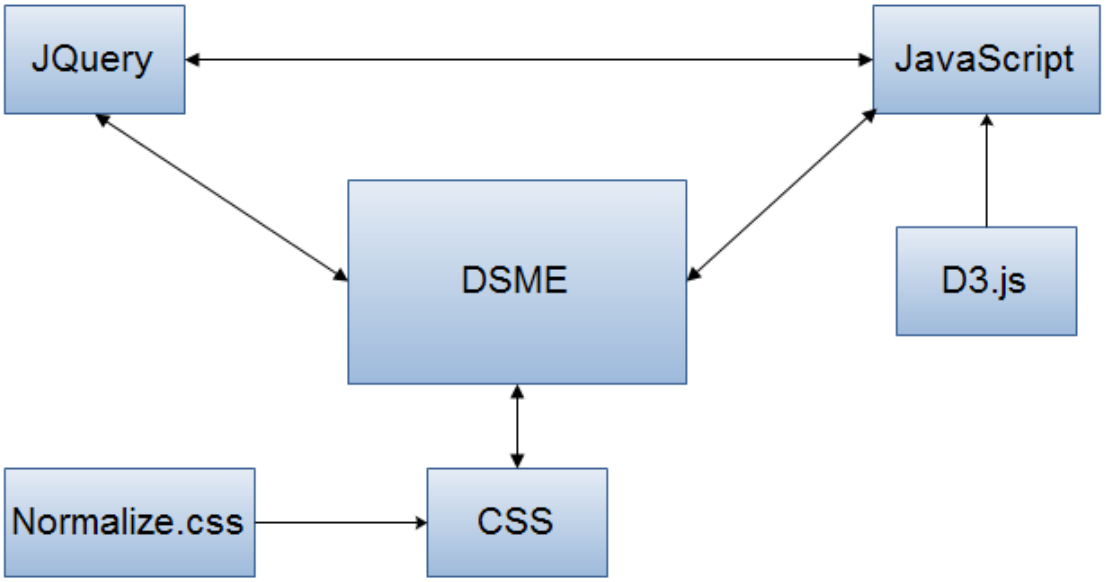
Canvas
: A HTML5 element that is used to draw graphics, on the fly, via scripting.

D3.js
: A JavaScript library for manipulating documents based on data. D3 assists in translating data to graphical format using HTML, SVG and CSS.

Normalize.css

A CSS library that makes browsers render all elements more consistently and in line with modern standards. Normalize is a HTML5-ready alternative to the traditional CSS reset.

Chapter 2
System Architecture



2.2. DSME

The website, DSME, is the front end of the product we are providing to our users. This is what the user will view and interact with in order to accomplish their learning achievements, which is based on the related subject the website offers.

2.2. CSS3

The interaction abilities of the website are highly dependent on the CSS3 coding style which is invoked to full potential. Appropriate colour schemes, textualised formatting, positioning of content and other layout features are incorporated throughout the website, in order to simplify the exploration process of the website for the user. For the purpose of viewing and searching stability, the layout design is a resembling feature within each page of the website. Normalise.css, a third party library, has been attached to our website in order to integrate appropriate design features.

2.3. JavaScript

The functionality of the graphical animations relating to various data structures and algorithms provided within the website was created using the canvas element introduced in HTML5. JavaScript provided the basis of the graphical implementation of various animations within the website. D3.js, a third party library, has been used in correlation with JavaScript in order to provide a flexible, smooth flow control for our algorithm animations. Javascript, in collaboration with jQuery, has been used for data structure animations within the website that rely on user input, which produces graphical output.

2.4. jQuery

jQuery has provided a means of establishing a connection between the designed graphical animation and the user control buttons provided for interaction within our data structures animations. It contributed to the flexibility within our applications, which assisted in enhancing the optionality provided to our users. The latest version of jQuery, v1.9.1, has been incorporated within the website. Access to this downloadable version can be accessed via the reference section of this documentation.

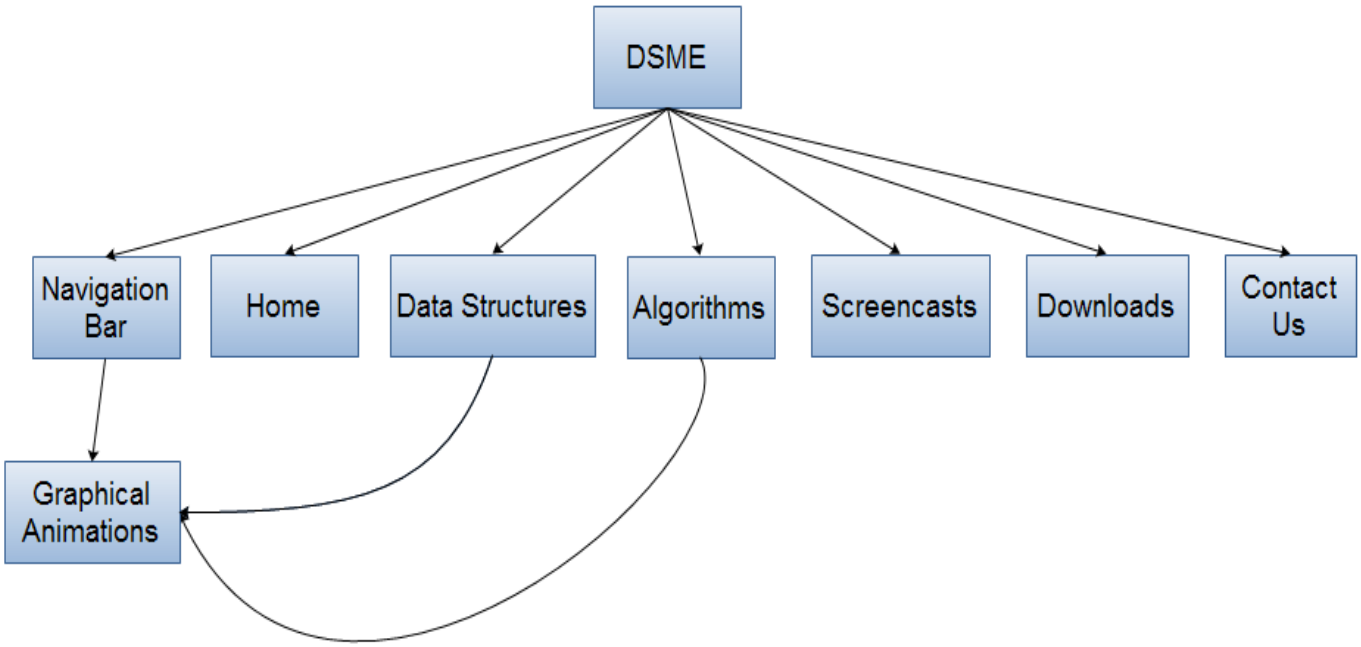
2.5. D3.js

Our algorithm animations required the appearance of a distinct design that completed its intended operations in a satisfactory format. D3.js is a third party library that was incorporated within the design of our algorithm animations, due its efficient manipulation of documents based on data. D3's ability to reduce overhead to a minimum and fast performance, assisted in establishing a complete efficient user-friendly animation. The latest version of d3.js, d3.v3, has been incorporated within the website. Access to this downloadable version can be accessed via the reference section of this documentation.

2.6. Normalize.css

Each browser maintains its own default user style sheet which it uses to make unstyled websites appear more legible. By implementing the normalized CSS library, it ensures our website will render the same way on every browser. Essentially normalize overwrites the default user style sheet provided by browsers. The latest version of normalize.css, v2.1.0, has been incorporated within the website. Access to this downloadable version can be accessed via the reference section of this documentation.

Chapter 3
High-Level Design



3.1. Home

The "home" page is the forefront of our website. Users will be transferred to this page upon clicking the website logo within the top navigational menu. This page contains a brief overview of the purpose of the website and what is has to offer the user.

3.2. Data Structures

The intention of this page is to provide the user with tutorial guide on how to approach learning the content within the website. A definition of a data structure is provided, along with problem areas related to the subject and a suggested approach to maximise the learning capabilities of the user. Links to various data structure animations and content of the subject is provided; these links can also be accessed via the navigational side bar. The guide in which we have provided is based primarily on survey in which we have conducted. The survey is located within the appendix section of this documentation.

3.3. Algorithms

The intention of this page is to provide the user with tutorial guide on how to approach learning the content within the website. A definition of an algorithm is provided, along with problem areas related to the subject and a suggested approach to maximise the learning capabilities of the user. Links to various algorithm animations and content of the subject is provided; these links can also be accessed via the navigational side bar. The guide in which we have provided is based primarily on survey in which we have conducted. The survey is located within the appendix section of this documentation.

3.4. Screencasts

The content of this page contains screencasts relating the data structures and algorithms discussed within the website. All screencasts are conducted by the website administrators and provides an informational overview of the selected topic. Information covered within each screencast includes explanation of the data structure or algorithm, an example of its performance observed by the animated application and guide in which how to code such a program using the Java programming language. Screencasts have been recorded by the system administrators using Camtasia 8.0. Access to this downloadable version can be accessed via the reference section of this documentation.

3.5. Downloads

Content, in which we have represented in previous pages, relating to a variety of data structures and algorithms, is provided in a downloadable PDF format for the user. Included in the downloadable content is an implementation, of the selected data structure or algorithm, in the Java programming language. This is represented in a PDF format, styled to professional format. A PDF converter was used to complete this task. Access to this PDF converter can be accessed via the reference section of this documentation.

3.6. Contact Us

For the purpose of communication with our users, we have constructed a contact page which enables our users to contact the websites administrators via personal emails provided. This will enable us to construct new content for the website, based on user feedback, which will further assist our users.

3.7. Navigational Bar

In order in optimise the accessibility of our website for our users; we have constructed a navigational side bar which links to designated pages relating to various selected data structures and algorithms. The content of these pages are structured formally for viewing purposes. The navigational side bar is present within every page of the website and remains in a fixed position for easy accessibility.

Chapter 4
Problems and Resolution

4.1. Design Problems

Whilst constructing the layout of our website, we intended to design an optimized navigational side bar which would be easily accessible to the user. We sought to have our navigational side bar in a fixed position, on the left side of each webpage. Problems such as overlapping and distortion occurred whilst testing was initiated. By continuous research and future increasing our knowledge of HTML5 and CSS3, we restructured our CSS3 code by inserting z-index and positioning styles to overcome this task.

4.2. Implementation Problems

Upon generating code for our animations within the website, we intended to construct an animation style that would be familiar to our users who wished to learn the functionality of various algorithms. With previous applets available on the web, the majority use rectangles to represent numeric values and switching to represent the functionality of particular algorithm. We intended to inherit this style in an optimal graphical format, by using D3.js. This incorporation succeeded to generate a stylised animation for algorithms. A problem arose when attempting to incorporate the software into data structures that read user input values. This was continuously attempted but with time constraints arising, we decided to source a different solution. But using strict JavaScript in cooperation with JQuery, for button control, we constructed the animation of data structures we originally intended to. Although the smooth layout is absent in the data structures animation, the learning style was incorporated. The resource we used to maximise our programming standards to complete this task was “JavaScript Step By Step” and is referenced within this documentation.

Another implementation problem arose with vast amounts of duplicate code being present within our animations. As various graphical animations are coded similarly in regards to appearance, we created a base class which contains all methods that were required for each program. This reduced time spent debugging certain applications.

4.3. Time Management Problems

Unfortunately, we were unable to keep to our original time management schedule due to unforeseen debugging purposes and time required for learning third party software. We decided to restructure our time schedule in order to meet the deadline with a satisfactory website to produce. This restructure of time management proved successful as we accomplished the original requirements of the project.

Chapter 5
Installation Guide

5.1. Required Software

DSME can be accessed via any popular web browser, such as Google Chrome or Modzilla Firefox.

5.2. Accessing Guidelines

In order to gain access to DSME the user must be using a device connected to the internet. For example, a laptop/desktop computer, smartphone or tablet. Following this, the user enters the website URL into any popular web browser (as mentioned above).

URL: student.computing.dcu.ie/~karneyf2

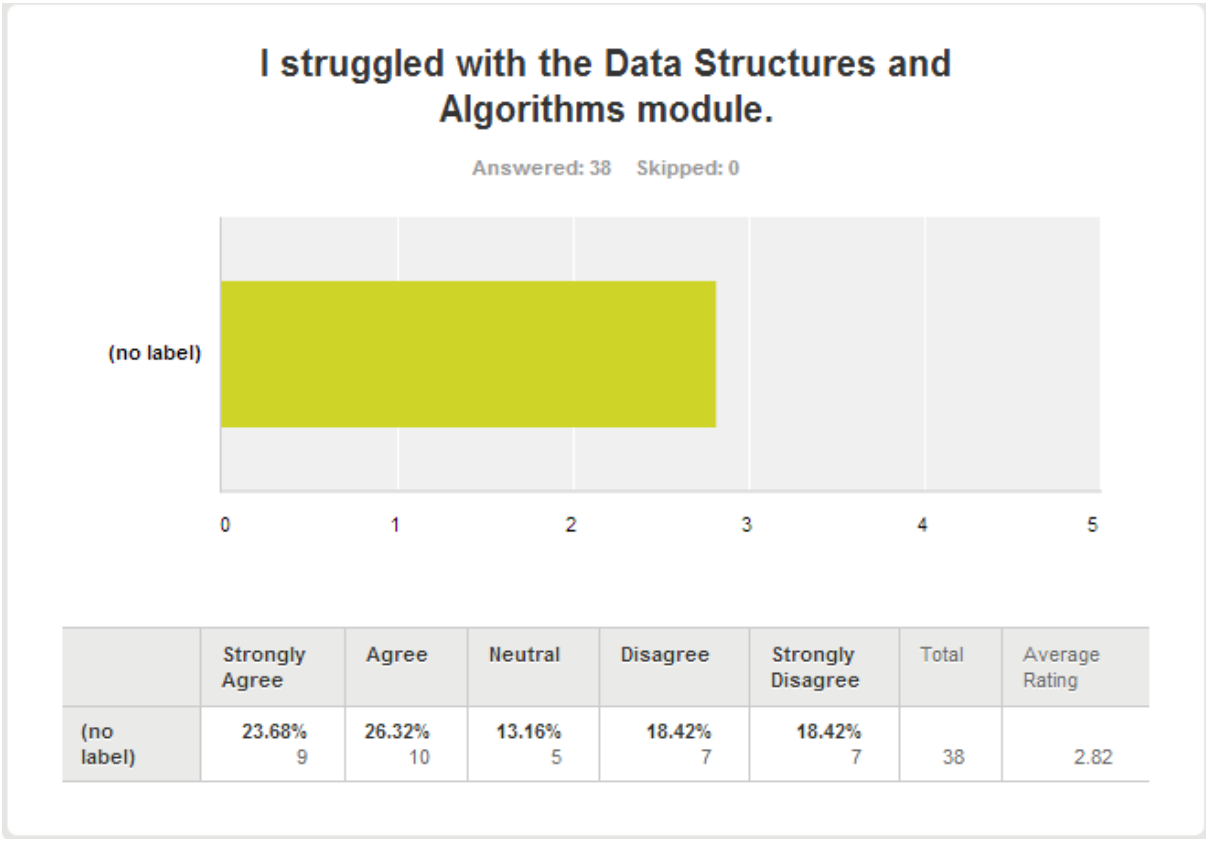
Chapter 6

Appendix

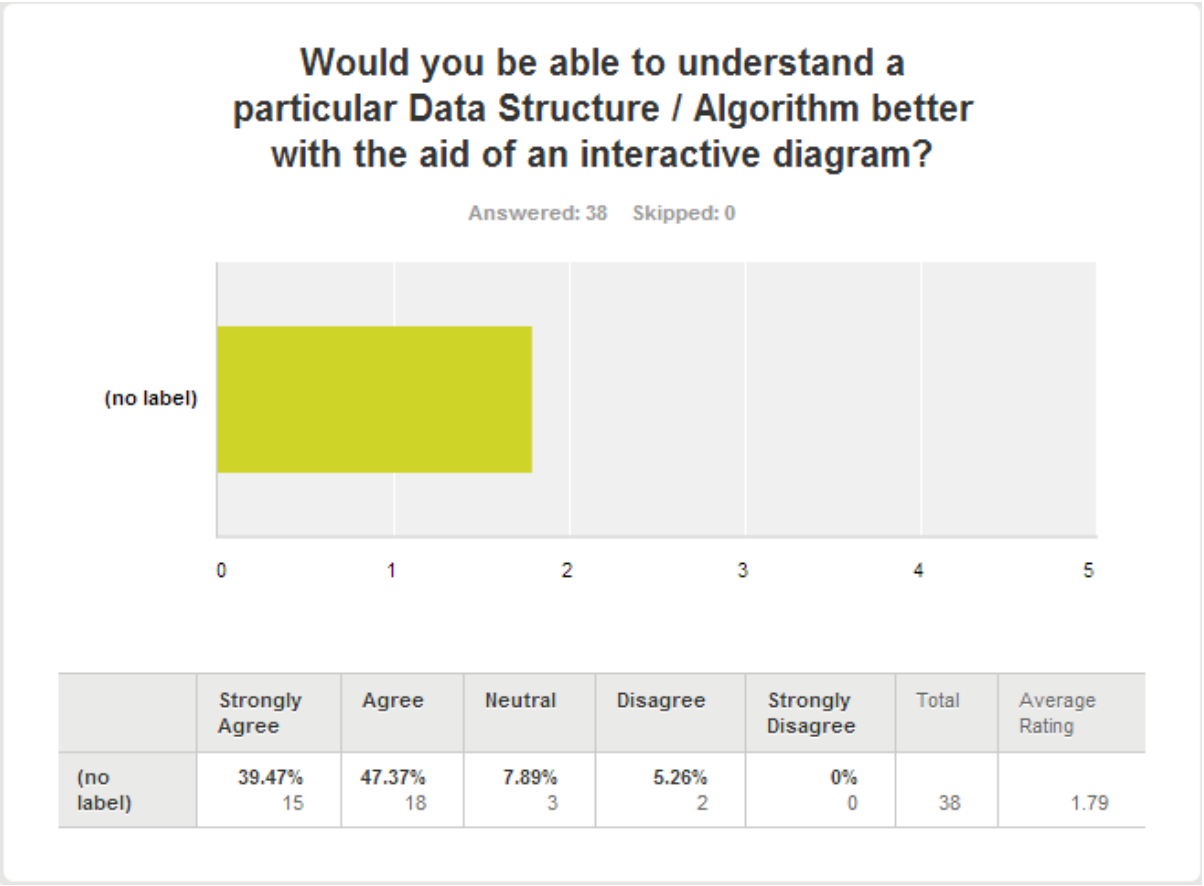
6.1. Survey

We carried out a survey amongst 2nd and 3rd year Computer Science students within Dublin City University, in order to discover the most common problem areas in data structures and algorithms. Based on the results we were able to focus on known areas of difficulty.

The following is a summary of the results yielded by the survey.



	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree	Total	Average Rating
(no label)	23.68% 9	26.32% 10	13.16% 5	18.42% 7	18.42% 7	38	2.82



	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree	Total	Average Rating
(no label)	39.47% 15	47.37% 18	7.89% 3	5.26% 2	0% 0	38	1.79

Common Problem Areas: Binary Search Trees, Doubly Linked Lists, Merge Sort and Quicksort.

Chapter 7
References

Bostock M. 2012. *Data-Drive Documents* [Online]. Available from: http://d3js.org/

Normalize.css 2012. *A modern, HTML5-ready alternative to CSS resets* [Online]. Available from: http://necolas.github.com/normalize.css/

Gallagher N. 2012. *About normalize.css* [Online]. Available from: http://nicolasgallagher.com/about-normalize-css/

jQuery 2012. *Download jQuery* [Online]. Available from: http://jquery.com/

TechSmith 2013. *Camtasia Studio* [Online]. Available from: http://www.techsmith.com/camtasia.html

Word to PDF Online Coverter. *Convert DOC to PDF* [Online]. Available from: http://convertonlinefree.com/

Suehring S. 2010. JavaScript Step By Step 2nd Edition. ISBN: 978-0-7356-4552-3