ALGORITHM VISUALIZER

SYNOPSIS

OF MINOR PROJECT

MASTER OF COMPUTER APPLICATIONS

SUBMITTED BY

TANU JAISWAL
Batch Year– 2019-22
Enrolment No.–U1949014

PROJECT GUIDE- ER. SHREYA AGARWAL

Centre of Computer Education & Training
Institute of Professional Studies
University of Allahabad, Prayagraj
Uttar Pradesh

INTRODUCTION

- Algorithms are a fascinating use case for visualization. To visualise an algorithm, we don't merely fit data to chart; there is no primary dataset. Instead there are logical rules that describe behaviour.

- In mathematical and empirical analyses of algorithms, there is a way to study algorithms. It is called algorithm visualization and can be defined as the use of images to convey some useful information about algorithms.

- Algorithm visualization illustrates how algorithms work in a graphical way. It mainly aims to simplify and deepen the understanding of algorithms operation. Specifically, it demonstrates the instances on the functioning of the algorithms and bring the information about statistics.

- That information can be a visual illustration of an algorithm's operation, of its performance on different kinds of inputs, or of its execution speed versus that of other algorithms for the same problem.

- To accomplish this goal, an algorithm visualization uses graphic elements points, line segments, two- or three-dimensional bars, and so on to represent some "interesting events" in the algorithm's operation.

PROBLEM DEFINITION

- Development of visualizer intent to represent the graphical indentation of algorithms.

MOTIVATION

- Amongst the most popular methods currently being investigated are algorithm visualizations and animations. It is true that students can learn algorithms without using an algorithm visualization. But based on the old-age adage -

  "*A picture speaks more than thousand words"*

  Many researchers and educationists assume that students would learn algorithm faster and more thoroughly using an algorithm visualization.

- There are benefits that are assumed by those creating these systems.
  - A visualization can hope to convey dynamic concepts of an algorithm.
  - Studying graphically is assumed to be easier and more fun for many students than reading "dry" textbooks because an important characteristic of the algorithm visualizations is that they are (seemingly) more like a video game or an animated movie.
  - By making an algorithm visualization more similar to these popular forms of entertainment, instructors can use it to grab students' attention. It can help instructors to cover more material related to a specific algorithm in less time.

OBJECTIVE(S)

1. The main objective of this project is to help beginners to be able to visualize the basic algorithms and get a better understanding of the underlying operations.

2. This aims to simplify and deepen the understanding of algorithms functioning. (Because a deeper understanding of human perception of images will be required before the true potential of algorithm visualization is fulfilled.)

3. This project will act as a tool, made with certain Front-End languages for visualizing sorting algorithms in an educational way. Our purpose is to portray the algorithms so the user can understand how a computer “move some pieces” to achieve his goal, having sorted data at the end!

REQUIREMENT ANALYSIS

- Technology Stack
  - Front-End Languages- HTML, CSS, JAVASCRIPT, BOOTSTARP, REACT
  - Front -End Frameworks- React Using JavaScript

- Software Requirements
  - Operating System:
    - Windows OS-
      Windows XP, Windows 7(Ultimate, & Enterprise), Windows 8(or 8.1), Windows 10 and later on versions.
    - MacOS-
      MacOS X (10.2 -10.15) 32/64-bit PowerPC, MacOS 11(Big Sur, 64-bit intel and ARM), MacOS 11.5 (Air M1, Big Sur, 64-bit intel and ARM), MacOS 12(Monterey, 64-bit intel and ARM) and later on versions.
  - Visual Studio Code
  - Web Browser such as Safari, Google Chrome, Opera, Mozilla Firefox, Microsoft Edge (or Edge Chromium), Internet Explorer, to host on local server.

- Hardware Requirements
  - Windows (minimum preferred)-
    - Processor- i3
    - Hard Disk- 5GB
    - Memory- 4GB Ram
  - MacBook (Preferred)
    - Processor- Apple M1 chip with 8-Core CPU and 8-Core GPU
    - Hard Disk- 5GB
    - Memory- 4/8GB RAM

Software Analysis

The implementation of this project is a combination of HTML5 (Hypertext Markup Language 5), JavaScript, CSS (Cascading Style Sheets), Bootstrap and React.
Following are the description about the factors that are concluded:

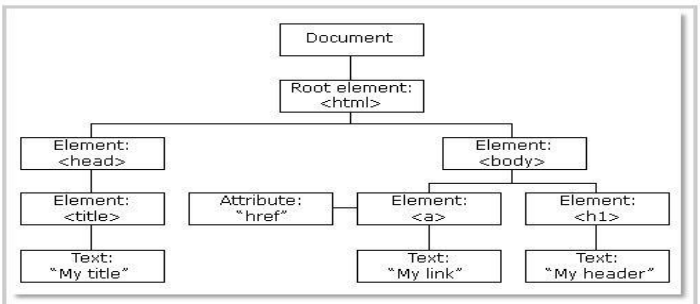
- **HTML** is used for describing the structure of Web pages. It has been used widely and effectively in this project, here are the few factors that are included:
  - Web Pages Development
  - Responsiveness
  - Offline Capability Usage

- **CSS** is used for describing the presentation of Web pages, including colors, layout, and fonts. Few implementing advantages of CSS are given below:
  - Compatibility
  - Image File Handling
  - Dynamic Templates

- **JavaScript** is used to add dynamic behavior to the webpage and add special effects to the webpage. There are certain purposes, due to which dynamic interactivity of web applications interprets full-fledged programming, given below:
  - Validation Purpose
  - Web Server Based
  - Interactive Behavior

- **Bootstrap** is an HTML, CSS & JS Library that focuses on simplifying the development of informative web pages (as opposed to web apps). What basically Bootstrap package contains? List:
  - Scaffolding
  - JavaScript Plugins
  - Components like navbar, iconography, etc.

- **React** is used for handling the view layer and development of web applications. Using the features of react multiple enhancement can be implemented, such as:
  - JSX (JavaScript Syntax eXtension)
  - Virtual DOM
  - Extensions



DOM of a Webpage

Study Design

The implementation of this project is a combination of HTML5 (Hypertext Markup Language 5), JavaScript, CSS (Cascading Style Sheets), Bootstrap and react. There is only one project folder that contains the various index.html and a client folder . The client folder contains the java script bundle of library and modules and some icon file named as .ico. Index.html file has reference to the significant files contained in client folder. As of now, the preferred browser is Google Chrome and Safari using live server in Visual studio code, as I only performed rigorous testing in this environment. Due to quick tests showed possible use in Google Chrome and Safari. High-Level Approach:

- Creating the website's User Interface (UI) using HTML, CSS and enhancing it further using Bootstrap; without actually implementing any of the app's core features.
- Implementation of animations, effects and core functionalities (sorting algorithms) using JavaScript.
- Optional: Publish to GitHub and host your project live using Netlify.

Figure1.

One large refactor later, the code now resembles a Model-View-Controller architecture. Although, due to its functional nature, it has many more individualized functions that update the instance variables and Boolean values, thereby directly updating the View and Controller. A simplified diagram of the Model-View-Controller relationship is below in Figure 2.

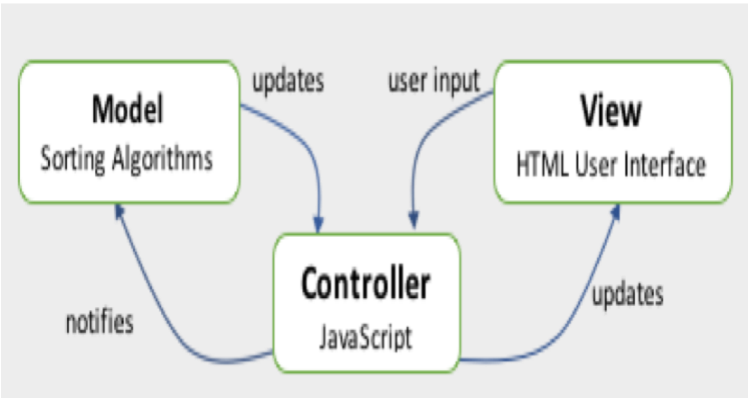


Figure 2: Model-View-Controller diagram of code.

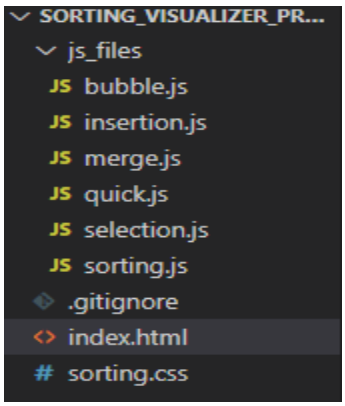


Figure 3.

Also, the class/object relationships between components are shown on the next page in Figure 4. The main module represents the global scope in the html file between the <script></script> tags. The variables and methods listed in it are accessible by all components.

DFDs at least up to second level &/or ER Diagrams &/or Class Diagrams

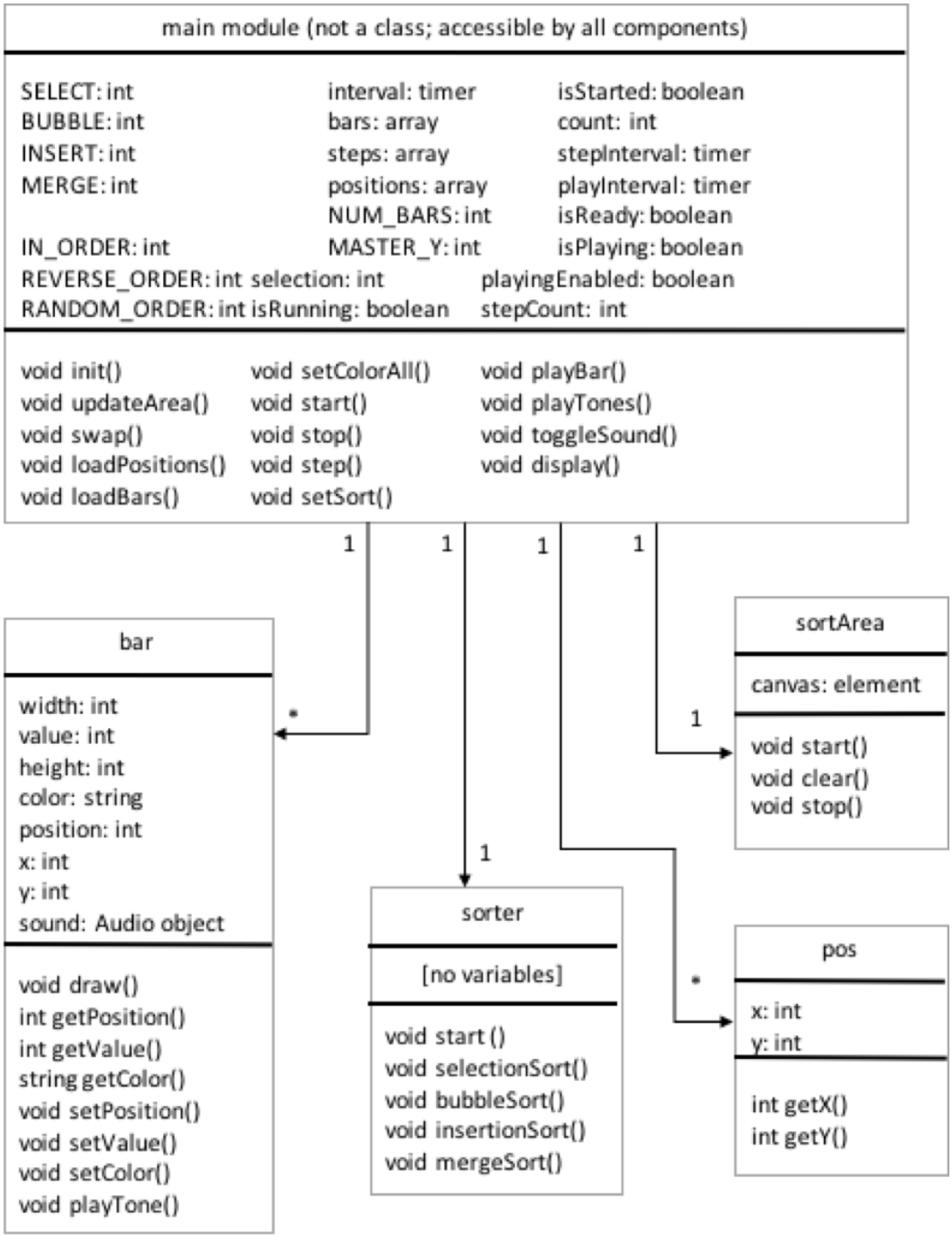


Figure 4: Class/object diagram of instance variables and respective functions in animation tool.

MILESTONE

S.No.	Project Activity	Estimated Start Date	Estimated End Date
1.	**Synopsis submission**	**22-09-2021**	**30-09-2021**

MEETING WITH THE SUPERVISOR

Date of the meet	Mode	Comments by the supervisor	Signature of the Supervisor
6th September, 2021	**Google Meet** (Online)	**Satisfactory**	*Shreya Agarwal*
25th September, 2021	**Google Meet** (Online)	**Good**	*Shreya Agarwal*
29th September, 2021	**Google Meet** (Online)	**Good**	*Shreya Agarwal*

Bibliography & References

1. Reference Book: Algorithm Design: Foundations, Analysis, and Internet Examples, Authored by Roberto Tamassis, published by Wiley.

2. Research Paper on “A Meta-Study of Algorithm Visualization Effectiveness” CHRISTOPHER D. HUNDHAUSENn , SARAH
https://www.cc.gatech.edu/~stasko/papers/jvlc02.pdf

3. Research Paper on Easy Chair Preprint No.5385, Algorithm Visualizer, Ashwani Kumar Singh, Danish Jamal and Pranjal Aggarwal
https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwiZq7jEpKLzAhUZ5nMBHVcCCOoQFnoECBAQAQ&url=https%3A%2F%2Feasychair.org%2Fpublications%2Fpreprint_download%2Ftw6K&usg=AOvVaw0bZunt72S0AKEJFLpws-s4

4. Honours Project Overview, Visualizing Sorting Algorithms, Brian Faria

https://digitalcommons.ric.edu/honors_projects/?utm_source=digitalcommons.ric.edu%2Fhonors_projects%2F127&utm_medium=PDF&utm_campaign=PDFCoverPages