

**DR. BABASAHEB AMBEDKAR TECHNOLOGICAL
UNIVERSITY, LONERE 402103**



PROJECT REPORT

on

“SORTING VISUALIZER USING JAVASCRIPT”

PRESENTED BY

Sr No	Student Name	PRN
1.	Aishwarya Anand Kumbhar	2067571242015
2.	Rutuja Arjun Mane	2067571242013
3.	Aarti Madhukar Palande	2067571242020
4.	Muskan Hanif Shaikh	2067571242007

UNDER THE GUIDANCE OF

Dr. Balshetwar S. V.



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

YASHODA TECHNICAL CAMPUS, WADHE, SATARA



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
Certificate

This is to certify that the project entitled “SORTING VISUALIZER USING JAVASCRIPT” is prepared and submitted by

1. Aishwarya Anand Kumbhar	2067571242015
2. Rutuja Arjun Mane	2067571242013
3. Aarti Madhukar Palande	2067571242020
4. Muskan Hanif Shaikh	2067571242007

Students of B.Tech (COMPUTER SCIENCE AND ENGINEERING), in partial fulfillment of the requirements for the award of Bachelor of Technology in Computer Science and Engineering as prescribed by the Dr. Babasaheb Ambedkar Technological University, Lonere at Yashoda Technical Campus during academic year 2022-2023.

PROJECT GUIDE

Dr. Balshetwar S. V.

HOD

Dr. Balshetwar S. V.

PROJECT CO-ORDINATOR

Prof. Jagtap K. P.

PRINCIPAL

Dr. Badadpure P. R.

DECLARATION

We hereby declare that the project work entitled **SORTING VISUALIZER USING JAVASCRIPT** submitted to the Yashoda Technical Campus's College of Engineering, Satara is a record of an original work done by us under the guidance of our project guide **Dr. Balshetwar S. V.** and this project work is submitted in the partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Science & Engineering. This project report has not been copied, duplicated or plagiarized from any other paper, journal, document or book and has not been submitted to any educational institute or otherwise for the award of any certificate, diploma, degree or recognition.

ACKNOWLEDGEMENT

We take this opportunity to express our deep sense of gratitude to our guide, **Dr. Balshetwar S. V.** for her continuous guidance and encouragement during this study. Without her valuable suggestion and encouragement this would not have been possible. It is because of her experience and wonderful knowledge we can fulfill the partial requirement of completing the Field Training report within the stipulated time.

We express our deep gratitude to **Dr. Balshetwar S. V.** Head of Computer Engineering Department, for his valuable guidance and constant encouragement. We are very much thankful to **Dr. Badadpure P. R.** Principal, YSPM's Yashoda Technical Campus, Faculty of Engineering, Satara. We find our self- spellbound to acknowledge thanks to our father and mother for their silent support, patience, encouragement and affection without which this work would never have been possible and we thank all my colleagues for their valuable co- operation and coordinate which was available time to time.

Project Index

INDEX		
Chapter No	Description	Page No.
	CERTIFICATE	I
	DECLARATION	Ii
	ACKNOWLEDGEMENT	Iii
	PAPER PUBLICATIONS	Iv
	ABSTRACT	V
	INDEX	Vi
	LIST OF TABLES	Vii
	LIST OF FIGURES	Viii
1	INTRODUCTION	9
1.1	Motivation	
1.2	Project Overview	
1.3	Need of project	
1.4	Literature survey (at least 5)	10
2	PROBLEM DEFINITION & SCOPE	12
2.1	Problem statement	
2.2	Scope	
2.3	Area of project	
2.4	Goals & objectives	
3	SOFTWARE REQUIREMENT SPECIFICATION	13
3.1	Software requirement	
3.2	Hardware requirement	
4	PROJECT PLAN	14
4.1	Project schedule	
4.2	Project cost estimation	
4.3	SDLC	
4.4	Feasibility Study	
5	SOFTWARE DESIGN	17
5.2	Flow chart	
5.4	Use Case Diagram	
5.5	Activity Diagram	
5.6	System Architecture	

6	IMPLEMENTATION DETAILS	22
6.1	Modules and Their Functionalities	
7	SYSTEM TESTING	23
7.1	Interface Testing	
7.2	Performance Testing	
7.3	Compatibility testing	
8	SNAPSHOTS/ GUI	24
9	CONCLUSION	31
10	BIBLIOGRAPHY	32

LIST OF TABLES

Table No.	Title	Page No.
1.4	Literature Survey	10
4.1	Project Plan	14
4.2	Project Cost Estimate	14

LIST OF FIGURES

Figure No.	Title	Page No.
4.1	Project Schedule	14
5.1	Data Flow Diagram 5.1.1 : DFD Level 0 5.1.2 : DFD Level 1	17
5.2	Flow Chart	18
5.3	Use-Case Diagram	19
5.4	Activity Diagram	20
5.5	System Architecture	21
8.1	Home Page	24
8.2	Bubble Sort 8.2.1 Bubble Sort 8.2.2 Enter Elements Of Array 8.2.3 Generated Bar 8.2.4 Comparison Between Elements 8.2.5 Swapping Between Elements 8.2.6 Previous Sort 8.2.7 Successfully Sorted	25
8.3	Insertion Sort	28
8.4	Selection Sort	29
8.5	Quick Sort	29
8.6	Merge Sort	30

1. INTRODUCTION

1.1 Motivation:-

The motivation behind this project is to study how the operations on data structure are performed. So that students can learn various algorithms through animation. To get a clear knowledge about various data structures and their operations on it. It will make data structures learning more interesting. The main goal of this project is to implement a system for various sorting algorithms. User runs sorting algorithm by giving custom array. During the demonstration run, the application visualizes pseudo code and current information about some variables.

1.2 Project Overview:-

This project is a study that tested the benefits of animated sorting algorithms for teaching. To visualize sorting algorithms, a web-based animation application was constructed. Aim behind implementation of this project to make a clear understanding ability of various algorithms of data structure.

Using a web page this will simulate the Sorting algorithm of data structure such as Bubble Sort, selection sort, radix sort. And so on. Thus our web page provides effective and efficient knowledge of data structures.

It is a web-based application for visualizing different sorting algorithms with their functionality of:

- Input array size – Here user can take input array
- Stop Button - To Stop Sorted Visualized algorithm
- Resume - To Resume Sorting Visualized algorithm
- Reset - To Reset Sorted Visualized algorithm
- Clear - To Clear Sorted Visualized algorithm

This also provides some theoretical knowledge regarding the data structure.

1.3 Need Of Project:-

As we know Data structure and algorithms (DSA) are one of the important topics in technology field and along with that one more crystal truth is that each and every student goes through each concept of DSA but students get confused and DSA getting hard to understand for students because of everyone does programming but didn't know how it will be working so In this project we will take one concept that is sorting algorithm and of course we implement it and by using animations and visualization it will be helpful for students to understand that how actually sorting methods like bubble sort, selection sort, insertion sort, merge sort and so on work.

1.4 Literature Survey:

Sr. No.	Paper	Year	Author	Review
1.	A system for algorithms' animation	1999	D. Merlini, S. Petruzzi, R. Sprugnoli and M. C. Verri	Data structure related complicated concept are difficult to understand for student. So, visualization technique helps teacher to teach them very easily Sorting algorithm show user input data through animation. This tool shows sorted data and unsorted data with different color. It helps student to understand, how data can be sorted dynamically.
2.	DAVE: A Dynamic Algorithm Visualization Environment for Novice Learners	2008	E. Vrachnos and A. Jimoyiannis	Sorting algorithm represent sorted array through animation. This tool shows sorted data and unsorted data with different color. It helps student to see how sorting function actually work to sort data at backend
3.	Sorting Algorithm visualizer	2022	Thakkar, Kavita, S. Dash and S. K. Joshi	It is E-learning platform which helps to improve theoretical concept regarding Data structure and algorithm. Data can be represented by Bar graph, and then sorting algorithm may be apply on that. In Sorting Visualizer, take input data from user and show that data as bar graph. Then choose animation tool and after that algorithm can be apply on it.
4.	Algorithm Visualizer: Its features and working	2021	Goswami, A. Dhar, A. Gupta and A. Gupta	Some learners can't understand theory with clarity. From this work student can visualize several algorithms and learn new concepts. This visualizer is easy to operate and implement. It contain stepwise representation of visualization of algorithm which makes it easy to understand.
5.	Creating Engaging Online Learning Material with the JSAV JavaScript Algorithm Visualization Library	2016	V. Karavirta and C. A. Shaffer	DSA is complicated to learn for most of the students. Learners can improve their DSA topics with JavaScript Algorithm Visualization. In this contain JavaScript Algorithm Visualization library. This library helps to visualize data structures algorithms. Learners can see previous step of visualization with actual current step.

6.	Visualizing Sequence of Algorithms for Searching and Sorting	2009	Bremananth R.,Radhika V.Thenmozhi S.	Visualizing sequence of algorithm for searching and sorting in this paper. It help to understand how perform the sorting method in easy way. The main pros of algorithm visualization is acquire the knowledge through performing with set of data, technique of manage time and use of memory. It shows ninetype of sorting algorithm form this one of animataion system is BALSA(Brown Algorithm and animator).
7.	Interactive visualization of high dimensional marketing data	2015	Alfa Yohannis Yulius Prabowo	Interactive visualization, it is used to associate all types of variable and also describe visualization of huge amount of data for financial organization. We use matrix visualization as a selection tool, its simple for find the data. The client select data of selected variable and start a analysis by using 5 variable. Using this we give good customer behavior knowledge.
8.	Sorting visualization using ViSA (visualization of sorting algorithm)	2012	Tihomir Orehovački	In this paper, ViSA (Visualization of sorting algorithm) it discribe a tool and different type of technique for visualization of algorithm. The ViSA is simple and self operating visualization system, it explain and compare step by step every sorting algorithm which is more useful to every educational student for analyze thesorting algorithm also for faculty member for teaching.
9.	Sort Attack: Visualization and Gamification of Sorting Algorithm Learning	2015	Alfa Yohannis, Yulius Prabowo	Sorting algorithm is perform fundamental role in the field of computer science, which are not understandable toeveryone to defeat this problem it has been moving toward the game visualization. The combining algorithm gaming tool and instructional or graphical design is an minor task, so it require attentive design, because of that it developed conceptual model for learning instruction and improve visualization learning.
10.	Comparative of Advanced Sorting Algorithms Based on Time and Memory Usage	2021	Marcellino Marcellino, Davin William Pratama Kristien Margi	In this paper, it compare different type of advance sorting algorithm such as Quick sort, Heap sort, Merge sort, introspective sort. A algorithm compare on the bases of time and memory require to perform data sorting. Application write using VS code and satisfy using python language. In this it show introspective sort good at time and heap sort is good for memory.

1.4: Literature Review

2.PROBLEM DEFINATION AND SCOPE

2.1 Problem Statement:-

Aim behind implementation of this project to make a clear understand ability of various algorithms of data structures. Using a web page this will simulates the Sorting algorithm of data structure such as sorting such as bubble sort, selection sort, radix sort. Thus our web page provides effective and efficient knowledge of data structures.

It is web based application for visualize different sorting algorithm with their functionality of speed control, Array size control. This also provide some theoretical knowledge regarding the data structure.

2.2 Scope: -

The main goal of this project is to create teaching support software with visualization of the most known sorting algorithms and their variations. The application supports a graphic visualization of selected algorithms on randomly generated or manually created array, step-by-step execution possibility, pseudo code and current state of variables.

2.3 Area Of Project: -

Web based application

A web application is application software that is accessed using a web browser. Web applications are delivered on the World Wide Web to users with an active network connection.

A Web application (Web app) is an application program that is stored on a remote server and delivered over the Internet through a browser interface.

2.4 Goals and Objectives: -

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.

- Creativity in making the visualizations even better and attractive.
- This project is for educational purpose.

3.SOFTWARE REQUIREMENT SPECIFICATION

3.1 Hardware Requirements

Processor: core i3

Hard disk: 5 GB

Memory: 1 GB RAM

3.2Software Requirements

Browser: Chrome, Mozilla Firefox

IDE: Visual Studio Code

Frontend: HTML, CSS

4. PROJECT PLAN

4.1 Project Schedule

	Aug	Sept	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	Jun
Discussion With Guide											
Planning											
Research											
Synopsis											
Discussion With Guide											
Implementation											
Testing											
Submission											

Fig 4.1 Project Schedule

4.2 Project Cost Estimation

Sr. No	Project Stage	Formulae used	Effort in Days
1	Planning	20 percent of Project Life Cycle effort	20 Days
2	Requirement Gathering Requirement Analysis Design	80 percent of Project Life Cycle effort	90 days
3	Development Or Coding Effort	25percent of Project Life Cycle effort	30 days
4	Testing	20 percent of Project Life Cycle effort	23 days
5	Project Management Effort	25 percent of Project Life Cycle effort	29
6	User Testing	20 percent of Project Life Cycle effort	23
Total No of days Effort Required			215days

Fig4.2: Project Cost Estimation

Number of Hours Per Day: 3 Hours
Totals Hours Required: $\text{Hours} * \text{Days}$
 $3 * 215$
Totals Hours Required: 645 Hours

Cost per hour: 81Rs
Total cost: 645×81
Total cost: 52,245Rs

4.3 SDLC

Software Development Life Cycle (SDLC) is a process used by the software industry to design, develop and test high quality software. The SDLC aims to produce high-quality software that meets or exceeds customer expectations, reaches completion within times and cost estimates.

- SDLC is the acronym of Software Development Life Cycle.
- It is also called as Software Development Process.
- SDLC is a framework defining tasks performed at each step in the software development process.
- ISO/IEC 12207 is an international standard for software life-cycle processes. It aims to be the standard that defines all the tasks required for developing and maintaining software.

SDLC: -Agile Model:

Agile SDLC model is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product. Agile Methods break the product into small incremental builds. These builds are provided in iterations. Each iteration typically lasts from about one to three weeks. Every iteration involves cross functional teams working simultaneously on various areas like –

- Planning
- Requirements Analysis
- Design
- Coding
- Unit Testing and Acceptance Testing.

At the end of the iteration, a working product is displayed to the customer and important stakeholders.

4.4:Feasibility Study: -

A feasibility study evaluates a project's or system's practicality. As part of a feasibility study, the objective and rational analysis of a potential business or venture is conducted to determine its strengths and weaknesses, potential opportunities and threats, resources required to carry out, and ultimate success prospects. Two criteria should be considered when judging feasibility: the required cost and expected value.

A feasibility study is a comprehensive evaluation of a proposed project that evaluates all factors critical to its success in order to assess its likelihood of success. Business success can be defined primarily in terms of ROI, which is the amount of profits that will be generated by the project.

- **Operational Feasibility**

This assessment involves undertaking a study to analyze and determine whether—and how well—the organization's needs can be met by completing the project. Operational feasibility studies also examine how a project plan satisfies the requirements identified in the requirements analysis phase of system development.

- **Scheduling Feasibility**

This assessment is the most important for project success; after all, a project will fail if not completed on time. In scheduling feasibility, an organization estimates how much time the project will take to complete.

5.SOFTWARE DESIGN

5.1 Data Flow Diagram:

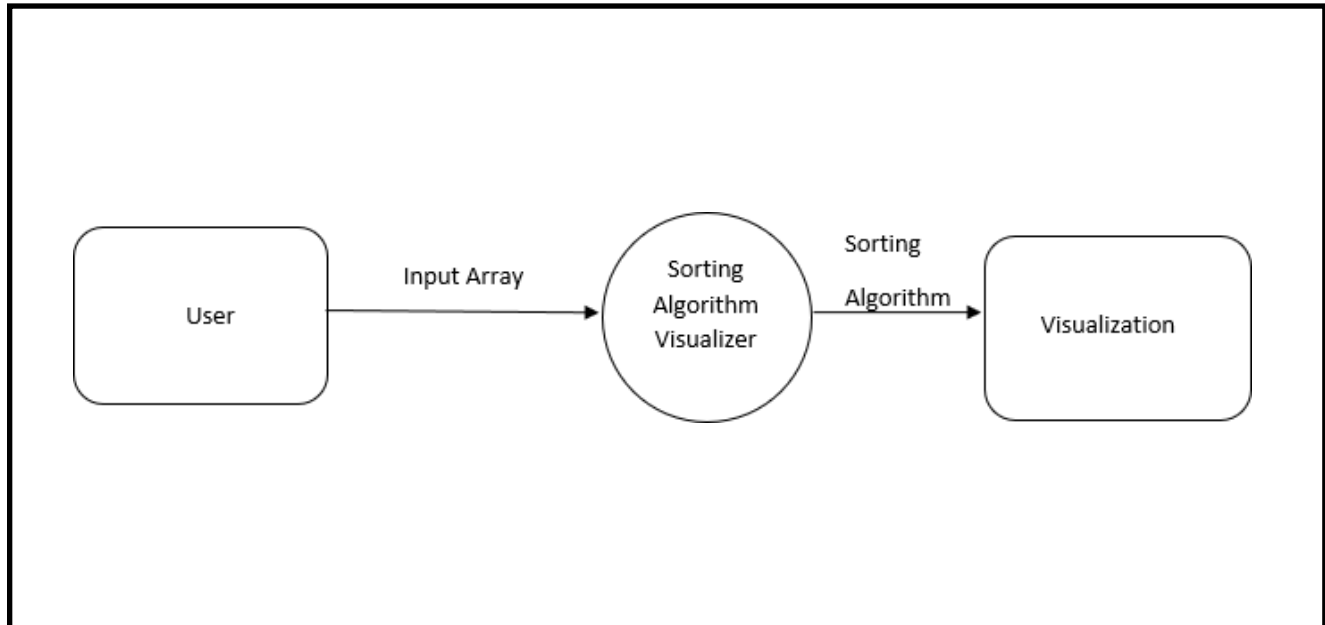


Fig 5.1.1: DFD Level 0

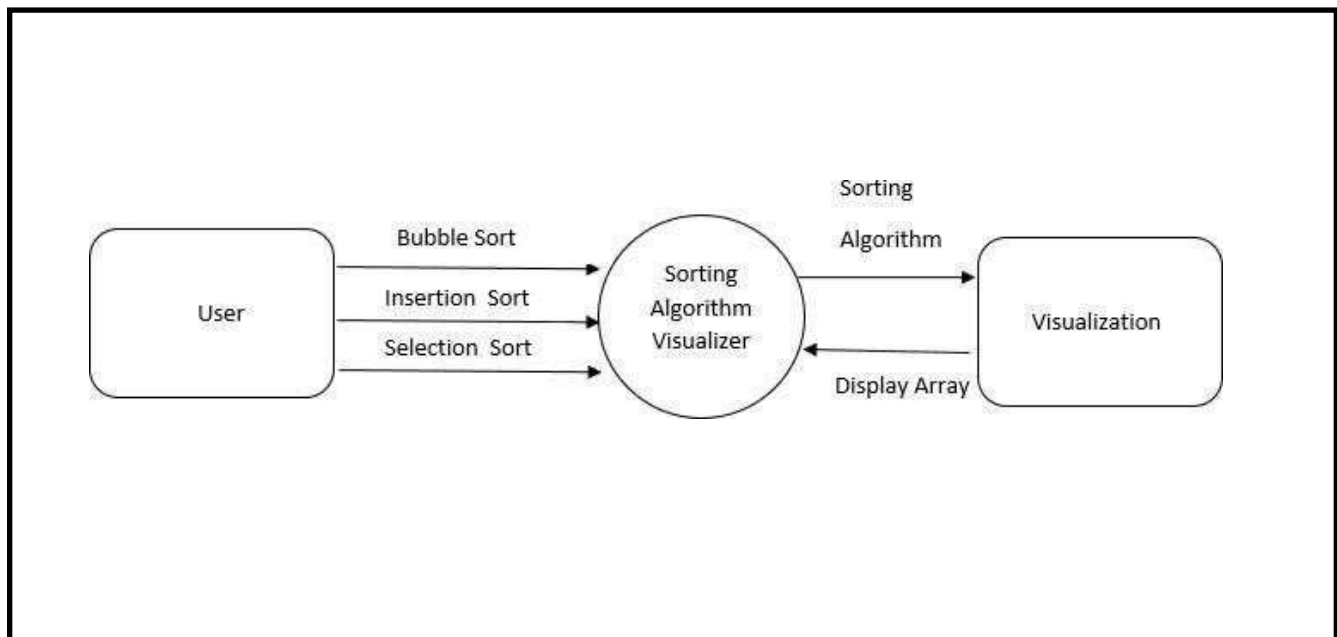


Fig 5.1.2: DFD Level 1

5.2 : Flow Char

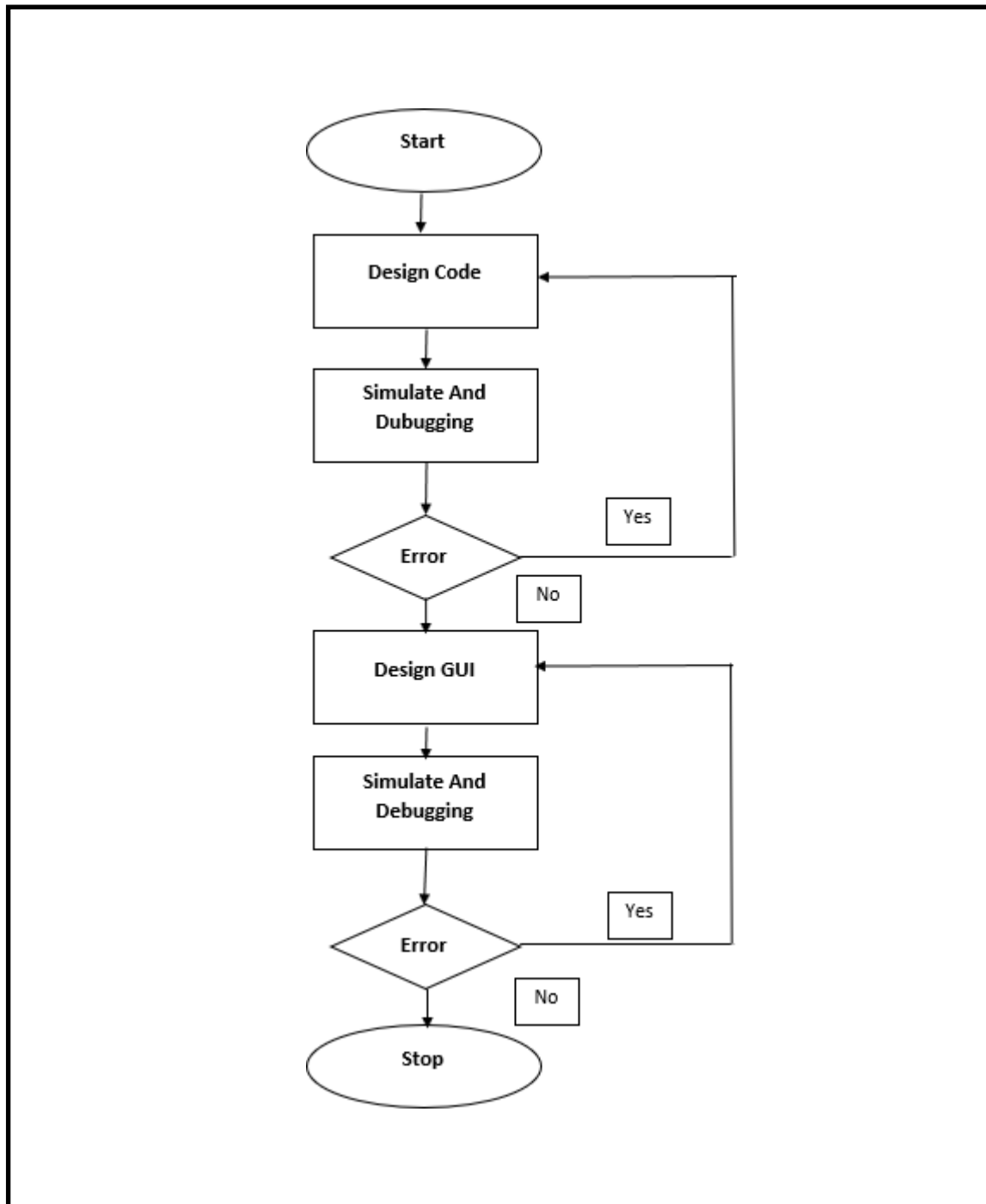


Fig 5.2: Flow Chart

5.3: Use Case Diagram:

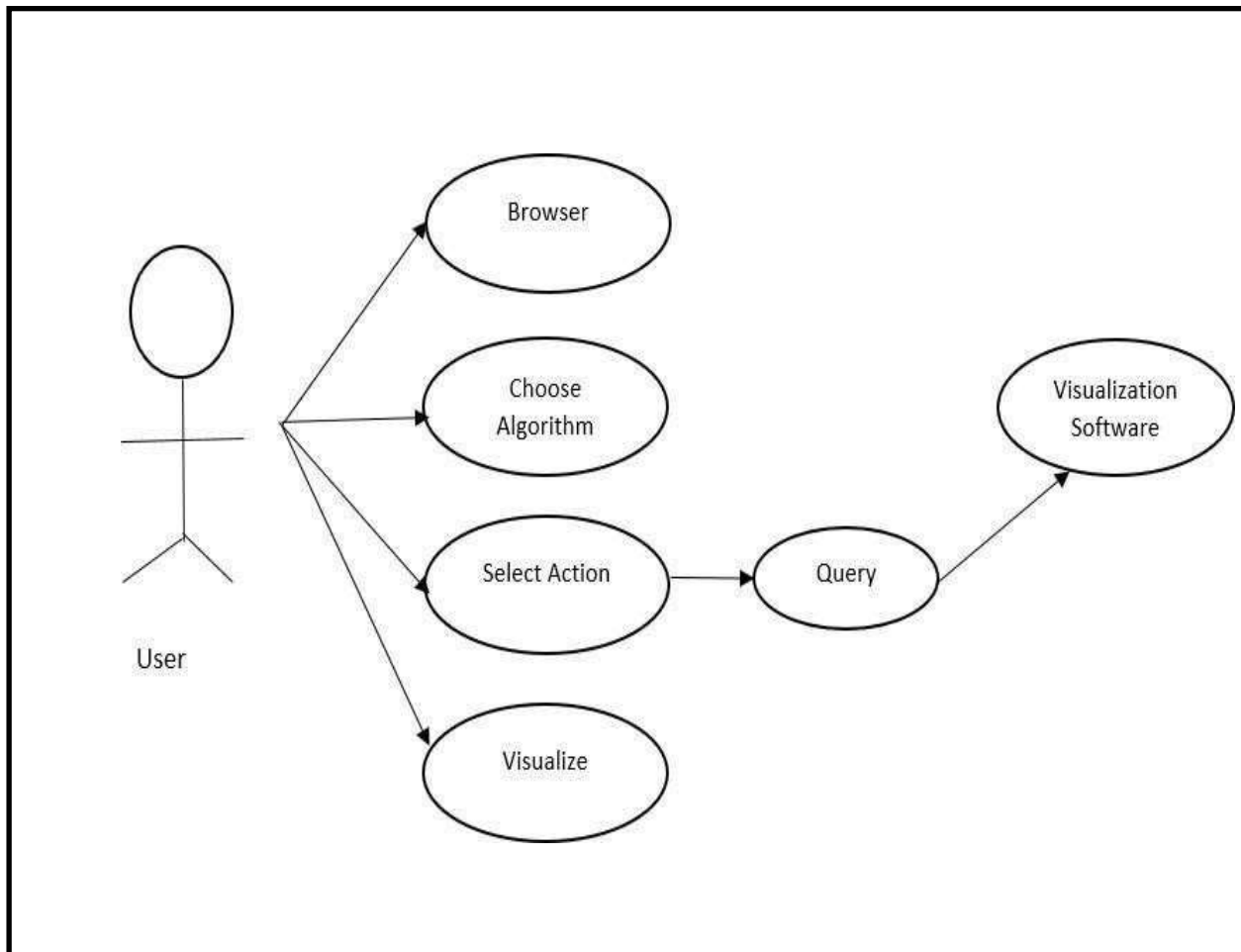


Fig 5.3: Use Case Diagram

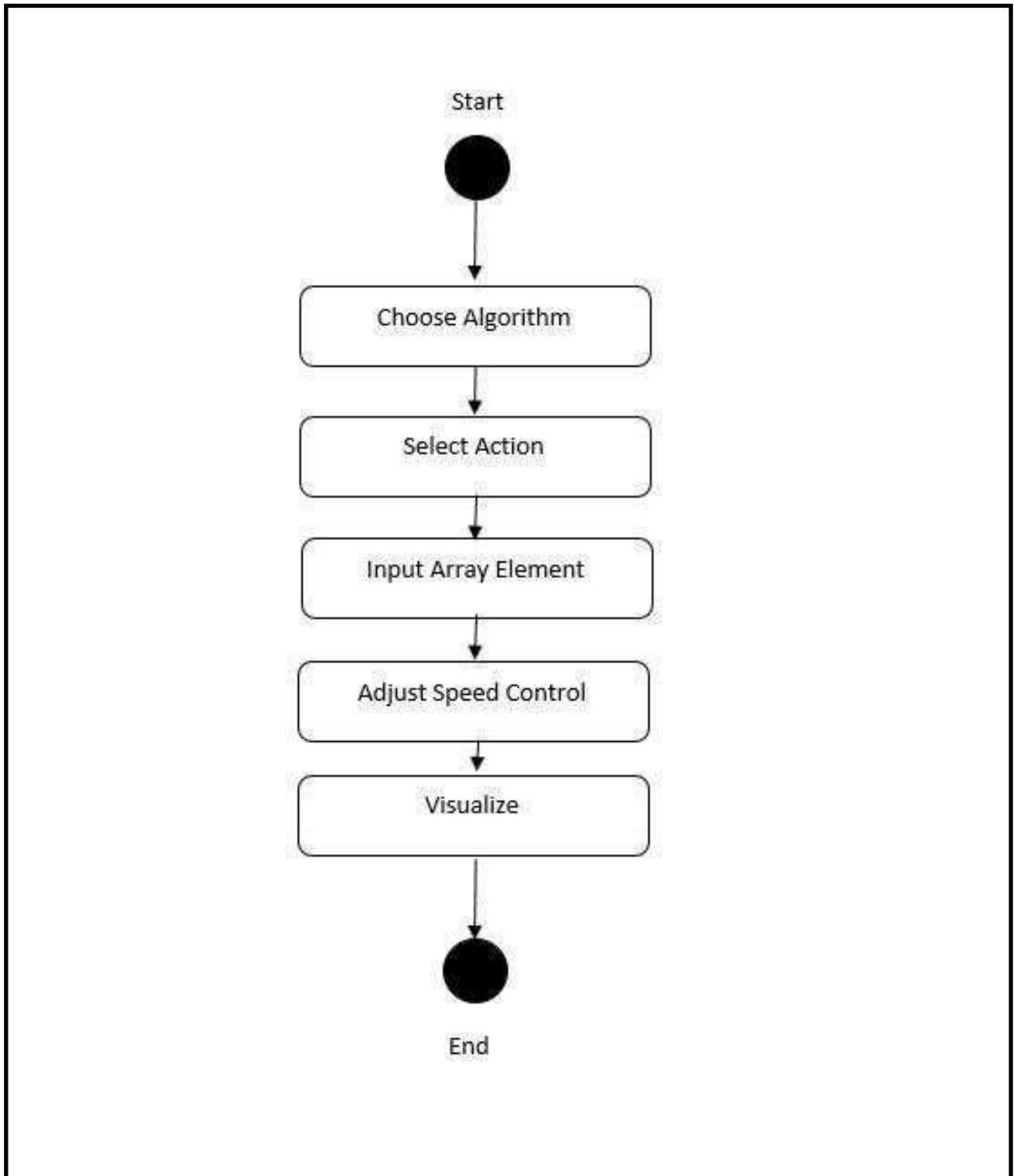
5.4: Activity Diagram:

Fig 5.4: Activity Diagram

5.5: System Architecture

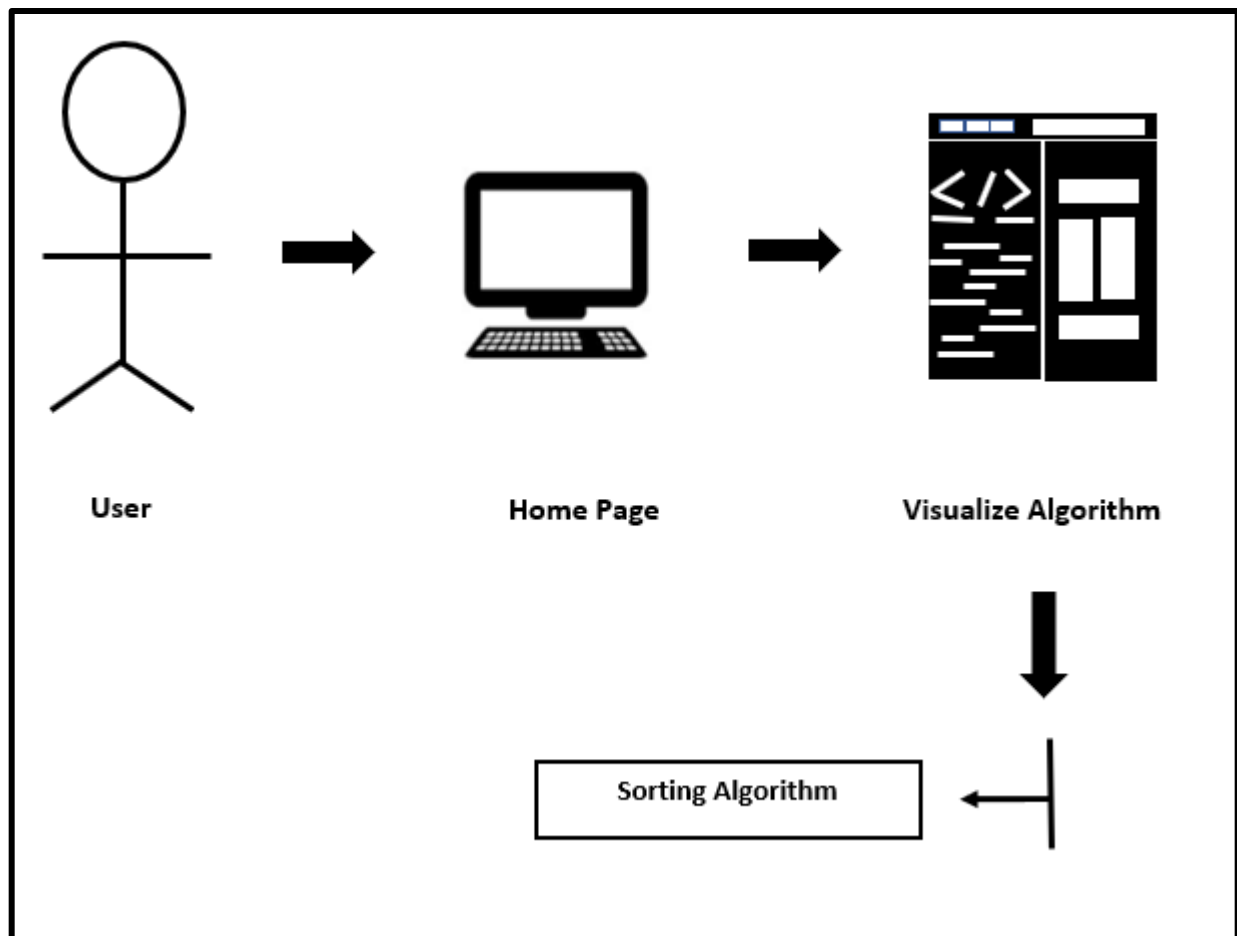


Fig 5.5: System Architecture

6. IMPLEMENTATION DETAILS

6.1 Modules and their functions:

❖ Home Page:

User can see various sorting algorithm and its abstract information.

❖ Bubble Sort Page:

- Input array size – Here user can take input array
- Stop Button - To Stop the bubble sort visualized algorithm
- Resume - To Resume bubble sort visualized algorithm
- Reset - To Reset bubble sort visualized algorithm
- Clear - To Clear bubble sort visualized algorithm

❖ Insertion Sort Page:

- Input array size – Here user can take input array
- Stop Button - To Stop the insertion sort Visualized algorithm
- Resume - To Resume the insertion sort Visualized algorithm
- Reset - To Reset the insertion sort Visualized algorithm
- Clear - To Clear the insertion sort Visualized algorithm

Designing:

For proper visualization we have used different colours to differentiate between the sorted bars and unsorted bars and even for bars which are currently in the process of comparison and sorting. After sorting the colours of bars will change to same colour which will be different from initial colours of array and array elements will be arranged in ascending order.

7.SYSTEM TESTING

What is Web Testing?

Web Testing, or website testing is checking your web application or website for potential bugs before its made live and is accessible to general public. Web Testing checks for functionality, usability, security, compatibility, performance of the web application or website.

a. Interface Testing:

Three areas to be tested here are – Application, Web and Database Server

Application: Test requests are sent correctly to the Database and output at the client side is displayed correctly. Errors if any must be caught by the application and must be only shown to the administrator and not the end user.

Web Server: Test Web server is handling all application requests without any service denial.

Database Server: Make sure queries sent to the database give expected results.

b. Performance Testing:

This will ensure your site works under all loads. Software Testing activities will include but not limited to –

Website application response times at different connection speeds

Load test your web application to determine its behaviour under normal and peak loads.

Stress test your web site to determine its break point when pushed to beyond normal loads at peak time.

c. Compatibility testing:


Compatibility tests ensures that your web application displays correctly across different devices.

This would include-

Browser Compatibility Test: Same website in different browsers will display differently. You need to test if your web application is being displayed correctly across browsers, JavaScript, AJAX and authentication is working fine. You may also check for Mobile Browser Compatibility.

8. SNAPSHOTS / GUI

8.1 Home Page:




YSPM's Yashoda Technical Campus, Satara.
Final Year Project [CSE]

Visualize the sorting algorithms and performs the operation

SORTING VISUALIZER

Developed By:
Aishwarya Kumbhar
Rutuja Mane
Aarti Palande
Muskan Shaikh


Bubble Sort



Bubble sort is a data sorting algorithm that works by randomly copying elements from the first array into a smaller second array, and then reversing the order of these arrays. After this process has been repeated multiple times, the sorted data will be located in the middle of the larger array.

Bubble Sort


Insertion Sort



An insertion sort compares values in turn, starting with the second value in the list. If this value is greater than the value to the left of it, no changes are made. Otherwise this value is repeatedly moved left until it meets a value that is less than it.

Insertion Sort


Selection Sort



Selection sort finds the smallest element in the array and place it on the first place on the list, then it finds the second smallest element in the array and place it on the second place. This process continues until all the elements are moved to their correct ordering.

Selection Sort


Quick Sort



Quicksort is a divide-and-conquer algorithm. It works by selecting a 'pivot' element from the array and partitioning the other elements into two sub-arrays, according to whether they are less than or greater than the pivot.

Quick Sort

Merge Sort



Merge sort follows divide and conquer approach in which, the list is first divided into the sets of equal elements and then each half of the list is sorted by using merge sort. The sorted list is combined again to form an elementary sorted array.

Merge Sort

Fig 8.1. Home Page

8.2.1 Bubble Sort:

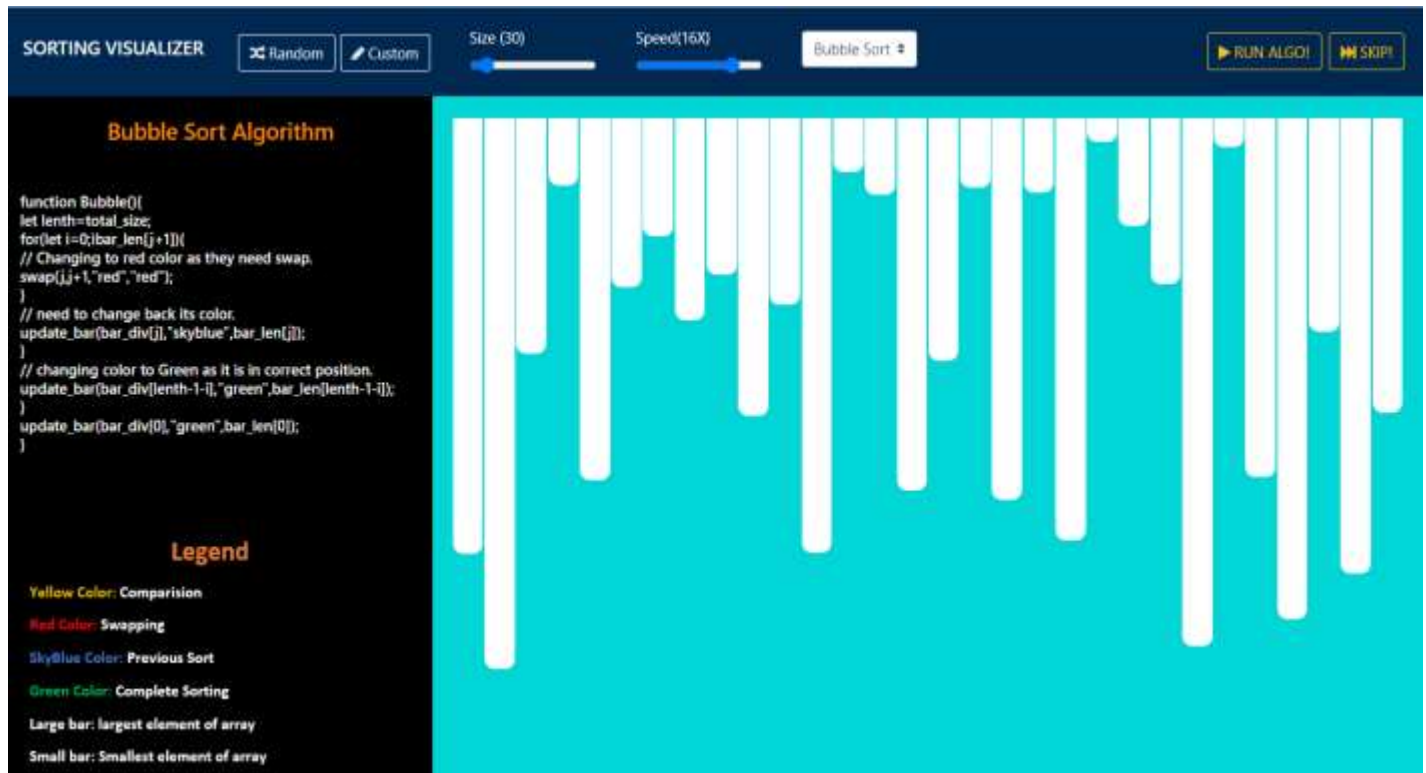


Fig 8.2.1 Bubble Sort

8.2.2 Enter Elements of Array:

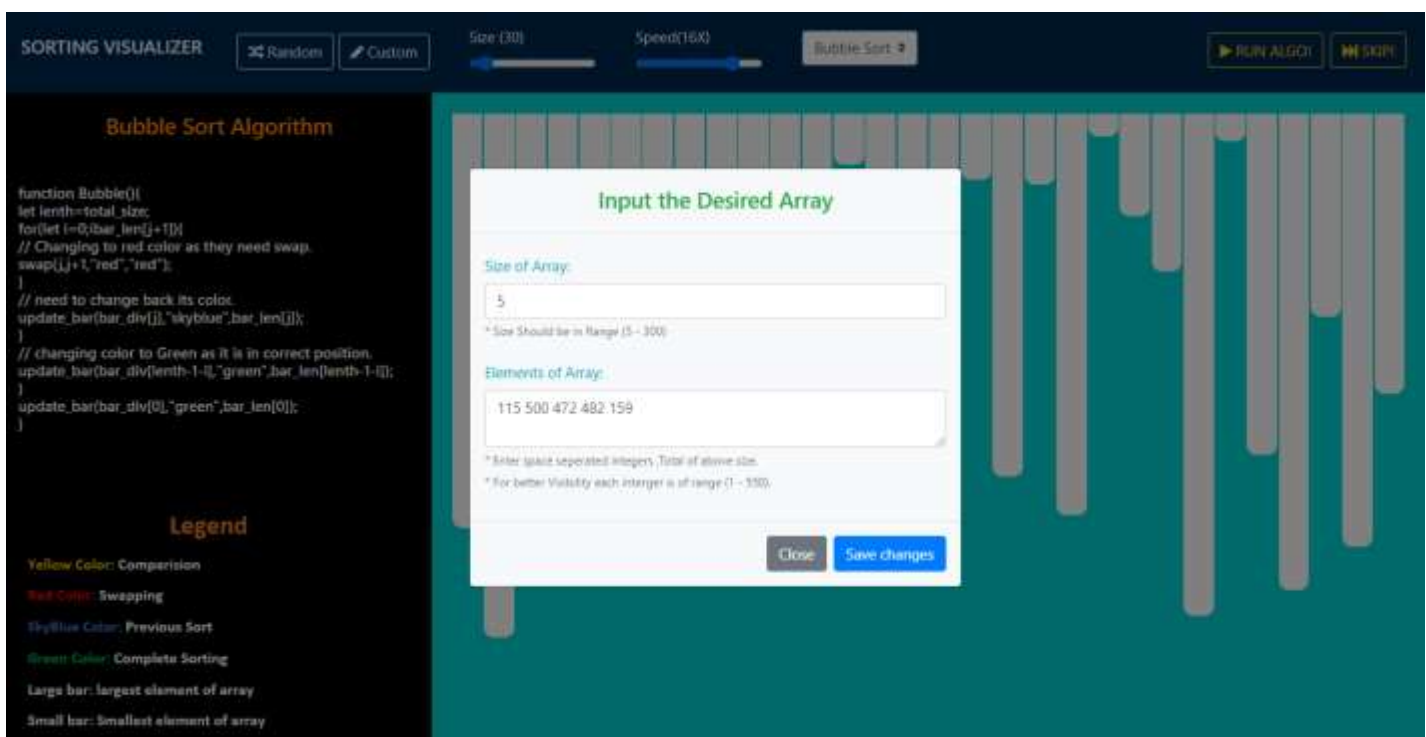


Fig 8.2.2 Enter Bubble Sort Elements

8.2.3 Generated Bars:

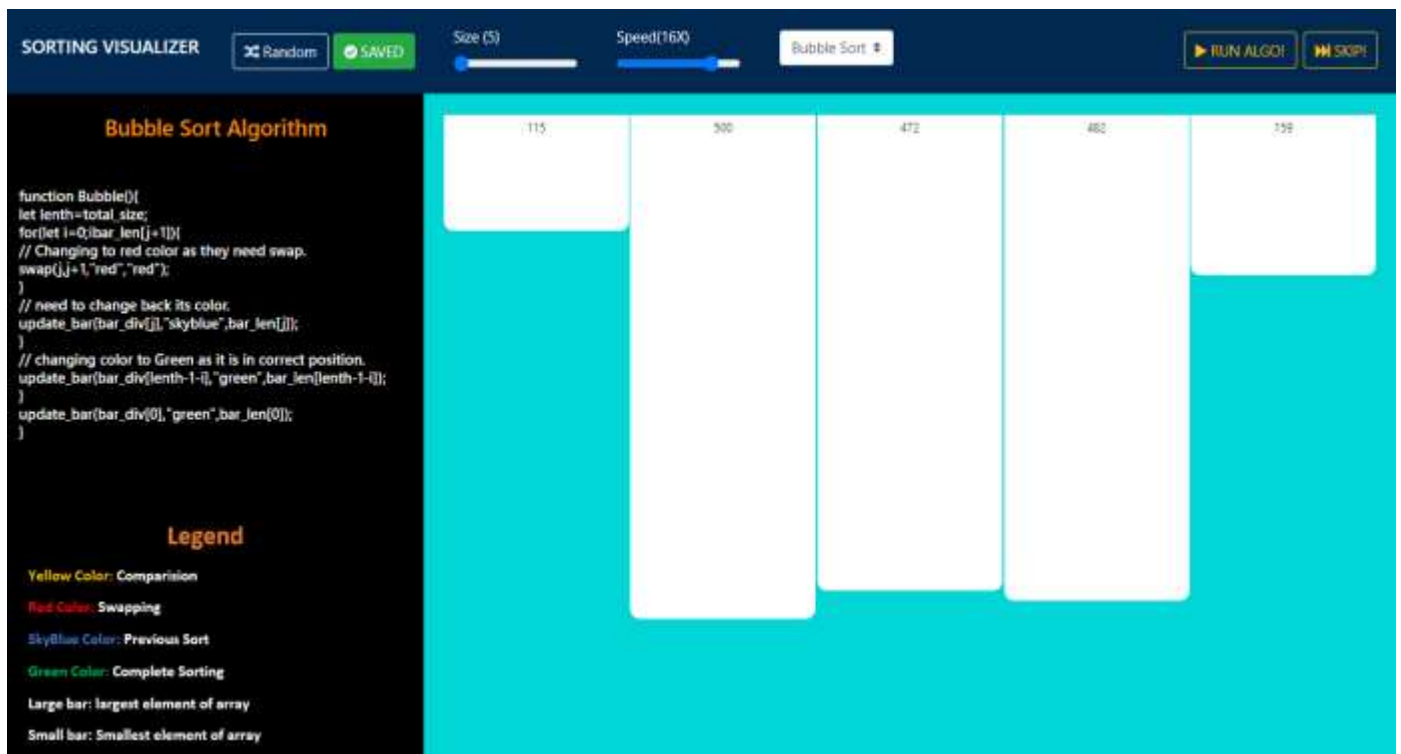


Fig 8.2.3 Generated Bars

Output

8.2.4 Comparison Between Elements:

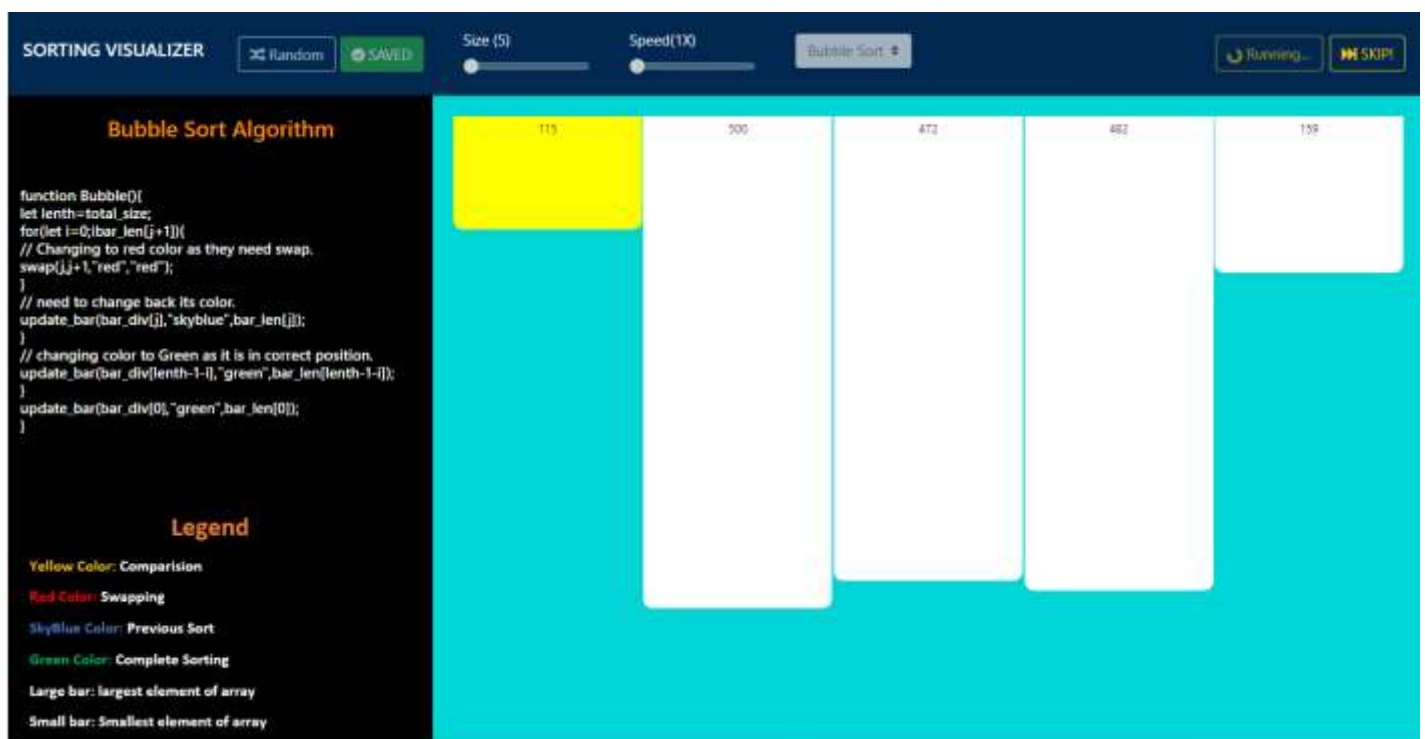


Fig 8.2.4 Comparison Between Elements

8.2.5 Swapping Between Elements:

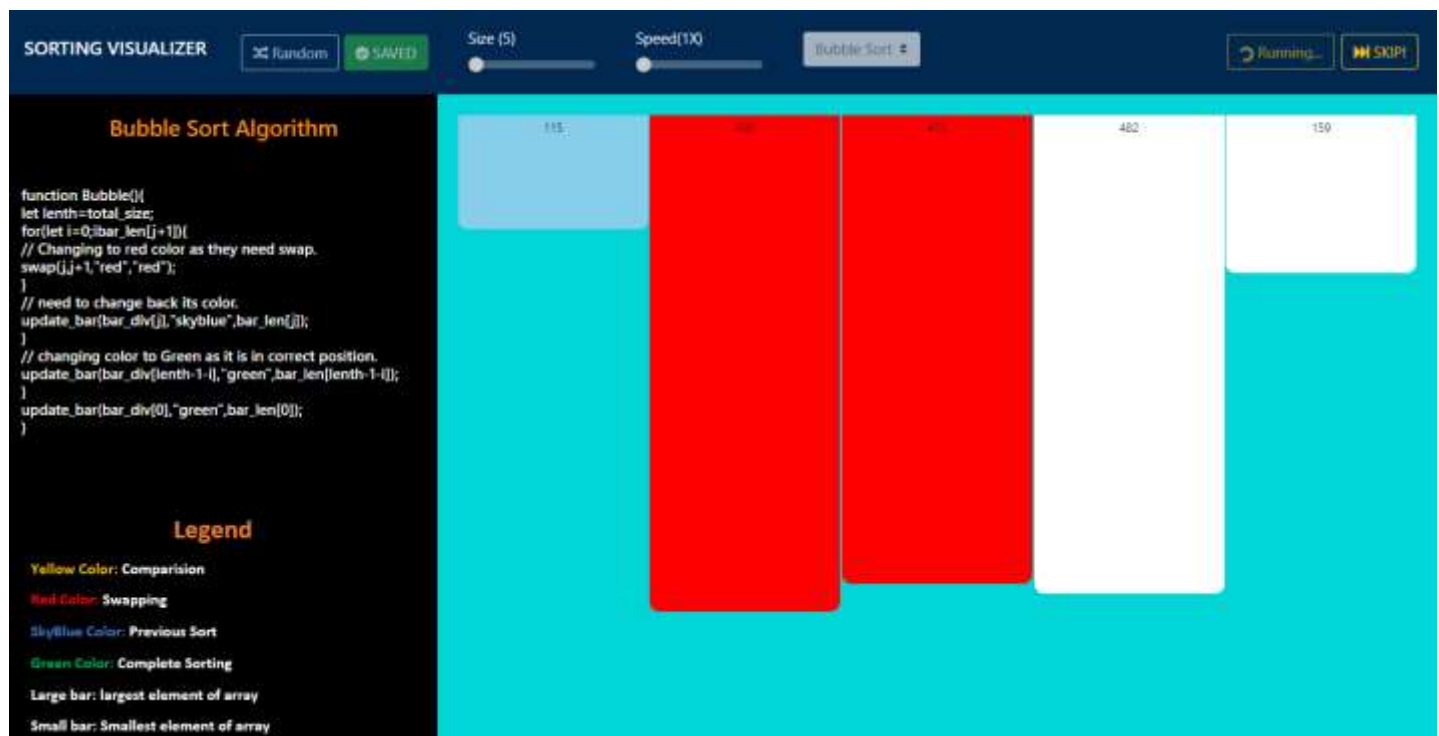


Fig 8.2.5 Swapping Between Elements

8.2.6 Previous Sort:

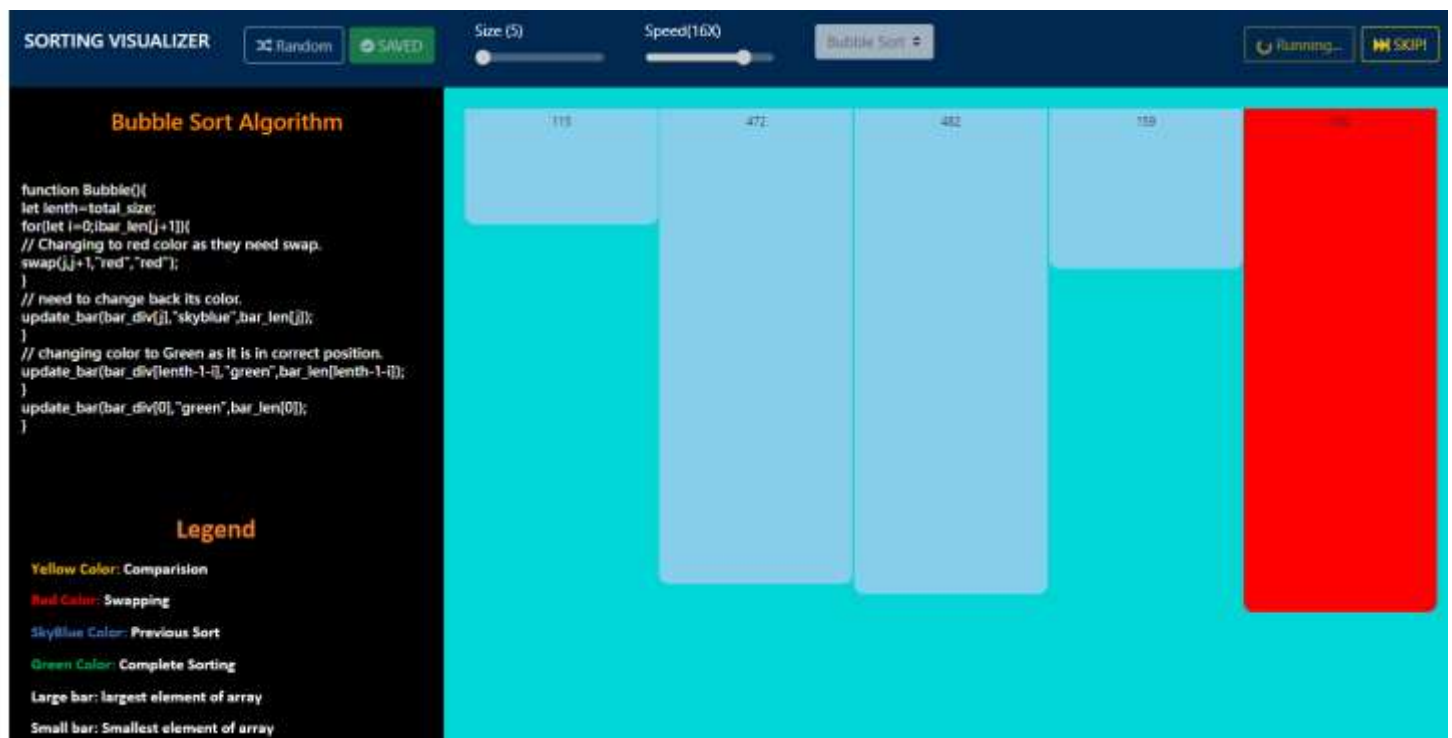


Fig 8.2.6 Previous Sort

8.2.7 Successfully Sorted:

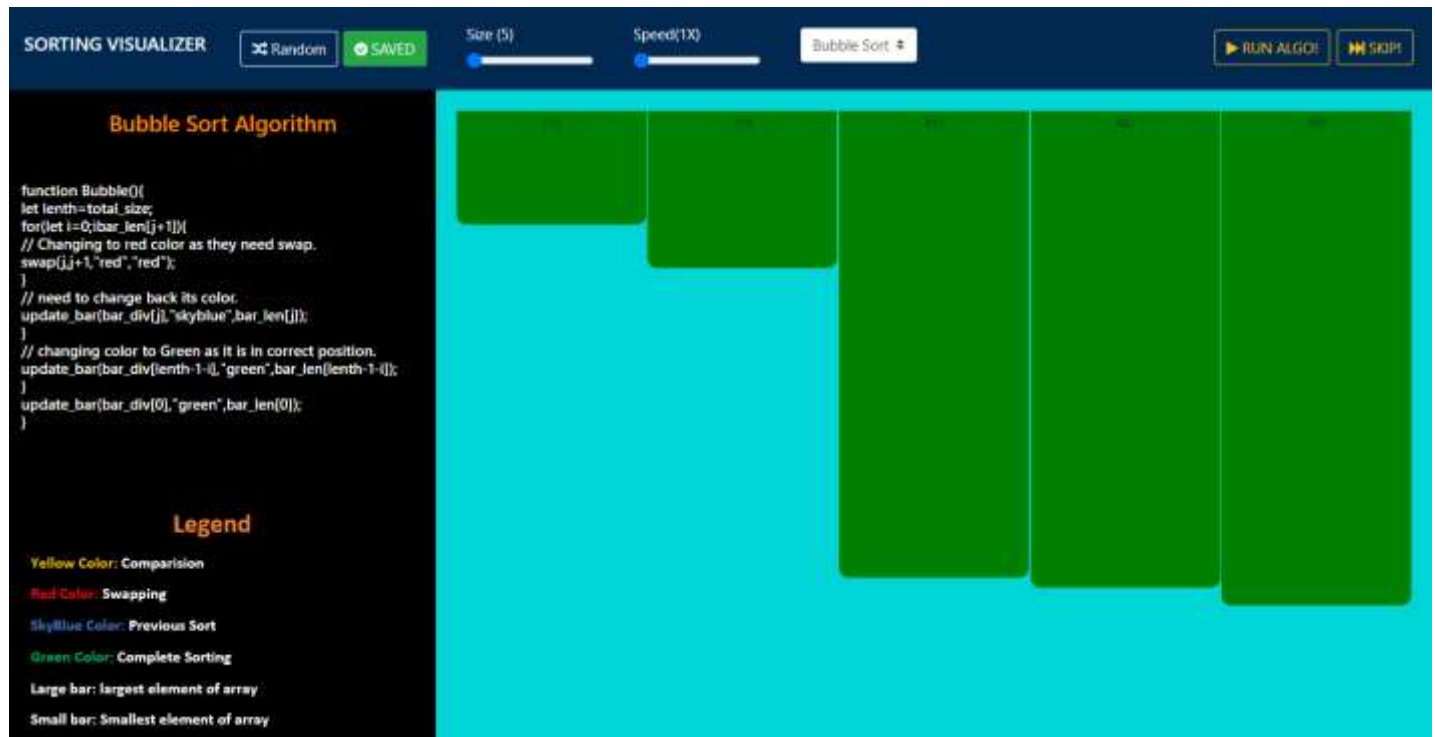


Fig 8.2.7 Successfully Sorted

8.3 Insertion Sort



Fig 8.3 Insertion Sort

8.4 Selection Sort

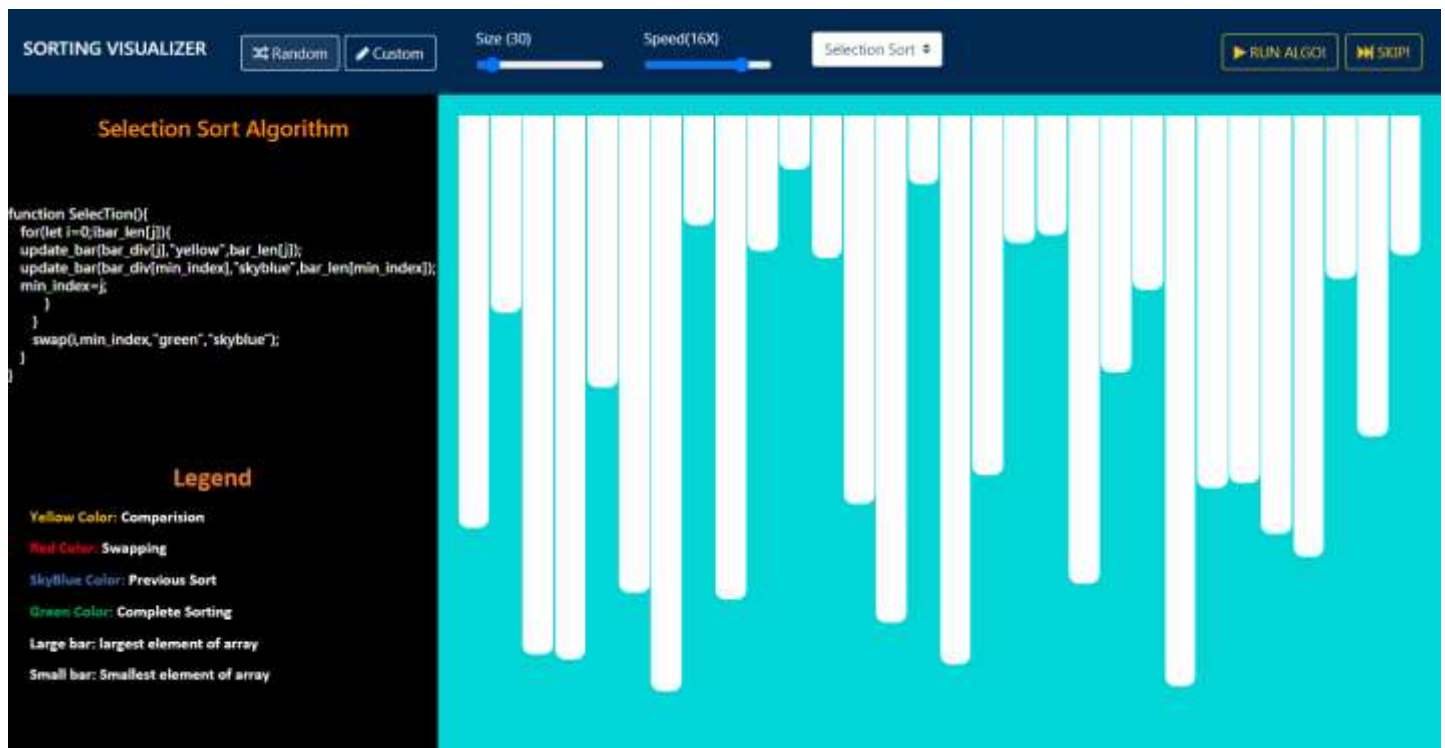


Fig 8.4 Selection Sort

8.5 Quick Sort

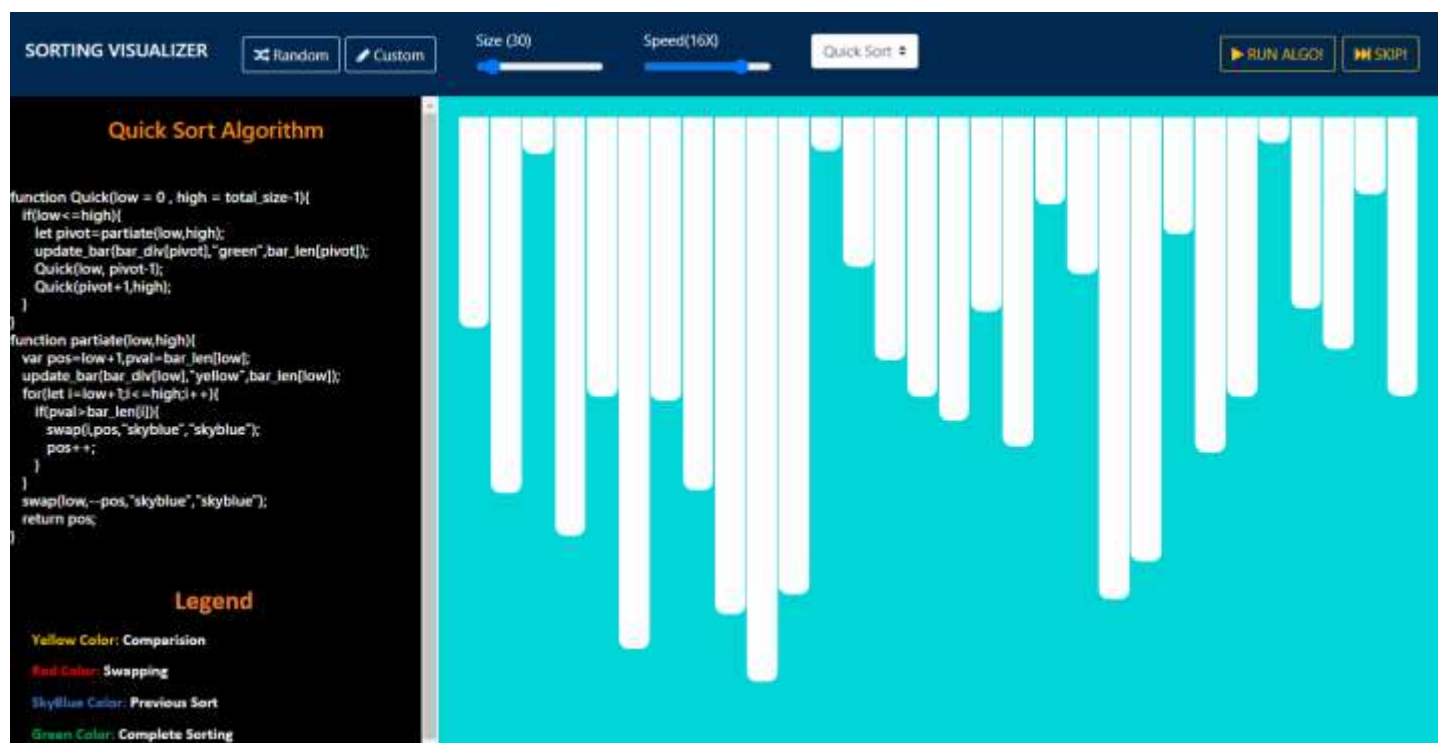


Fig 8.5 Quick Sort

8.6 Merge Sort



Fig 8.6 Merge Sort

9. CONCLUSION

It is teaching support application which visualizes the most known sorting algorithms. User run sorting algorithm by giving custom array. During the demonstration run, the application visualizes pseudo code and current information about some variables.

We tried to create high-quality software with a user-friendly and easy-to-use interface, which could be used by lecturers, tutors, and students. Possible next improvement of the applications is extension it by other algorithms.

10.BIBLIOGRAPHY

- [1] D. Merlini, S. Petruzzi, R. Sprugnoli and M. C. Verri, "A system for algorithms' animation," Proceedings IEEE International Conference on Multimedia Computing and Systems, 1999, pp. 1033-1034 vol.2, doi: 10.1109/MMCS.1999.778649.
<https://ieeexplore.ieee.org/document/778649>
- [2] E. Vrachnos and A. Jimoyiannis, "DAVE: A Dynamic Algorithm Visualization Environment for Novice Learners," 2008 Eighth IEEE International Conference on Advanced Learning Technologies, 2008, pp. 319-323, doi: 10.1109/ICALT.2008.148.
<https://ieeexplore.ieee.org/document/4561697>
- [3] Thakkar, Kavita, S. Dash and S. K. Joshi, "Sorting Algorithm visualizer," 2022 International Conference on Cyber Resilience (ICCR), 2022, pp. 1-5, doi: 10.1109/ICCR56254.2022.9996059.
<https://ieeexplore.ieee.org/document/9996059>
- [4] Goswami, A. Dhar, A. Gupta and A. Gupta, "Algorithm Visualizer: Its features and working," 2021 IEEE 8th Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON), 2021, pp. 1-5, doi: 10.1109/UPCON52273.2021.9667586.
<https://ieeexplore.ieee.org/document/9667586>
- [5] V. Karavirta and C. A. Shaffer, "Creating Engaging Online Learning Material with the JSAV JavaScript Algorithm Visualization Library," in IEEE Transactions on Learning Technologies, vol. 9, no. 2, pp. 171-183, 1 April-June 2016, doi: 10.1109/TLT.2015.2490673.
<https://ieeexplore.ieee.org/document/7298430>
- [6] B. R., R. V. and T. S., "Visualizing Sequence of Algorithms for Searching and Sorting," 2009 International Conference on Advances in Recent Technologies in Communication and Computing, 2009, pp. 647-649, doi: 10.1109/ARTCom.2009.20.
<https://ieeexplore.ieee.org/document/5329052/authors>
- [7] R. Smeulders and A. Heijs, "Interactive visualization of high dimensional marketing data in the financial industry," Ninth International Conference on Information Visualisation (IV'05), 2005, pp. 814-817, doi: 10.1109/IV.2005.66.
<https://ieeexplore.ieee.org/document/1509166>
- [8] I. Reif and T. Orehovacki, "ViSA: Visualization of sorting algorithms," 2012 Proceedings of the 35th International Convention MIPRO, 2012, pp. 1146-1151
<https://ieeexplore.ieee.org/document/6240816>
- [9] A. Yohannis and Y. Prabowo, "Sort Attack: Visualization and Gamification of Sorting Algorithm Learning," 2015 7th International Conference on Games and Virtual Worlds for Serious Applications (VS-GAMES), 2015, pp. 1-8, doi: 10.1109/VS-GAMES.2015.7295785.
<https://ieeexplore.ieee.org/document/7295785>
- [10] M. Marcellino, D. W. Pratama, S. S. Suntiarko and K. Margi, "Comparative of Advanced Sorting Algorithms (Quick Sort, Heap Sort, Merge Sort, Intro Sort, Radix Sort) Based on Time and Memory Usage," 2021 1st International Conference on Computer Science and Artificial Intelligence (ICCSAI), 2021, pp. 154-160, doi: 10.1109/ICCSAI53272.2021.9609715.
<https://ieeexplore.ieee.org/document/9609715>