e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
Volume:03/Issue:03/March-2021 Impact Factor- 5.354 www.irjmets.com

ALGORITHM VISUALIZER

Barnini Goswami [*1], Anushka Dhar[*2], Akash Gupta[*3], Antriksh Gupta[*4]

[*1]Student, Department of Computer Science and Engineering, Krishna Engineering College, Ghaziabad, Uttar Pradesh, India.

[*2]Student, Department of Computer Science and Engineering, Krishna Engineering College, Ghaziabad, Uttar Pradesh, India.

[*3]Student, Department of Computer Science and Engineering, Krishna Engineering College, Ghaziabad, Uttar Pradesh, India.

[*4]Student, Department of Computer Science and Engineering, Krishna Engineering College, Ghaziabad, Uttar Pradesh, India.

ABSTRACT

Over the years we've observed that algorithms even tough being a complex subject are the foundation of computational thinking and programming skills of a student. So to ease up the hardships of students this idea of the project was formed. Our application Algorithm Visualizer is both interactive and alluring to students.

It gives the students hands on experience of the algorithms' implementation. It feeds into their imagination to help them get a better understanding while also helping teachers to help make their students understand better. Through this project every student can learn at their own pace with our three speeds of learning: slow, average and fast. This interface is designed to make one feel fully engaged and concentrated. The concept of Time Complexity has also been introduced to the user through an interactive game. We have made use of React.js as framework and JavaScript as primary language for our project.

The purpose of this project is to make learning less of a burden and more of an incredible experience which leaves students with the want to learn more.

Keywords: Visualization, Algorithms, e-learning, path finding, sorting

I. INTRODUCTION

When we talk about complex subject topics like Algorithms, it becomes extremely necessary for students to have a strong grip over the topic as it would form the foundation of their computational thinking and programming skills. We had observed that through conventional methods of teaching it becomes a little difficult for students to understand the concept and also for teachers to explain their thoughts.

Motivated by the age-old saying, “a picture speaks more than thousand words”, many researchers and educators assume that students would learn an algorithm faster and more thoroughly using algorithm visualization techniques[1]. So, we developed a method of learning through visualization and hand-on experience over different searching and sorting algorithms which is bound to help the students and teachers. Good visualizations bring algorithms to life by graphically representing their various states and animating the transitions between those states, especially dynamic algorithm visualization which shows a continuous, movie-like presentation of an algorithm’s operations. Visualization allows the human visual system to extend human intellect; we can use it to better understand these important conceptual processes, other things, too.

Also, we are well aware of the fact that the more we do things ourselves and engage the more we tend to learn about a particular topic. Thus, engaging in various game like activities can surely help the users get a hold on the topics.

II. METHODOLOGY

Architectural Design

Our project comprises of 3 pages

1. Welcome page
2. Path finding Algorithm
3. Sorting Algorithm

The following flowchart represents the overall architecture of our application.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
Volume:03/Issue:03/March-2021 Impact Factor- 5.354 www.irjmets.com

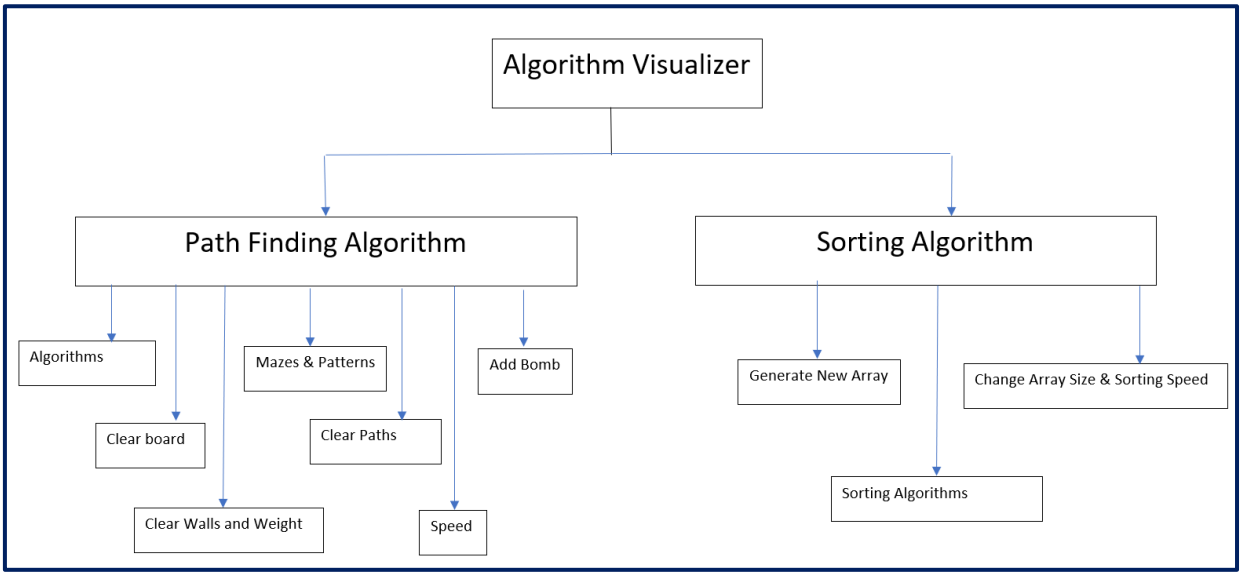

Figure-1: Architectural Design of Algorithm Visualizer

Workflow

User can traverse to two different pages from the welcome page:-
1. Path-finding Algorithm
2. Sorting Algorithm

1. Path-finding Algorithm

The navbar of path-finding algorithm consists of the following options:-

a) Algorithms

We have included:

- Dijkstra's Algorithm
- A*Search
- Greedy Best-First Search
- Swarm Algorithm
- Depth First Search
- Breadth First Search

The algorithms present in the navbar are chosen on the basis of their popularity and difficulty level. Students find it difficult to understand these algorithms theoretically. When they will see the visualization of these algorithms then they will be able to understand it better. User will be able to differentiate between the functionalities of different algorithms on the basis of time complexity after the visualization is over.

b) Mazes and Patterns

Maze and patterns are included to ensure better and clear understanding of algorithms. As there will be walls or obstruction between the starting node and the goal node, user can relate the visualization with real world like situation. Also, user will be able to figure out which algorithm is better based on algorithm time complexity. Especially for users looking for a playful option for understanding these complex topics these fun filled options can turn out to be the appropriate way.

c) Speed

The project contains speed bar for maintaining the speed of visualization, this feature is included because everyone has a different learning rate so the user can vary the speed of visualization according to his/her choice.

Designing

Grid structure will be used to represent each node. Computer generated starting and ending node will be displayed initially. User can change the positions of start and end node according to his/her will.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
Volume:03/Issue:03/March-2021 Impact Factor- 5.354 www.irjmets.com

Structure of Mazes & pattern can also be changed according to user's will (i.e., pattern of addition of new walls).

2. Sorting Algorithm

The navbar of sorting algorithm consists of the following options:-

a) Generate new array

It will generate a new random array. Every time we click on this tab it will generate new random array. Array elements will be displayed in the form of bars with the height of each bar proportional to the numerical value it is representing. While sorting different colored bars would be used to represent the sorted, unsorted and currently sorting numerical values from the array of input numbers.

b) Change array size and sorting speed

A slider will be provided so that user can change the size of array and accordingly the speed of sorting will vary. Size of array will be directly proportional to the sorting speed i.e. (larger the speed of array greater will be the speed of sorting). As, mentioned earlier this feature has been implemented to ensure users are able to learn at their own pace without any haste.

c) Algorithms

- Merge Sort
- Quick Sort
- Heap Sort
- Bubble Sort

User can pick any algorithm from the above list of available algorithms. Algorithms are chosen on the basis of their popularity and importance according to the existing curriculum.

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 arrange in ascending order.

III. MODELING AND ANALYSIS

Our implementation of the project is based on the below ER (entity relationship) model.

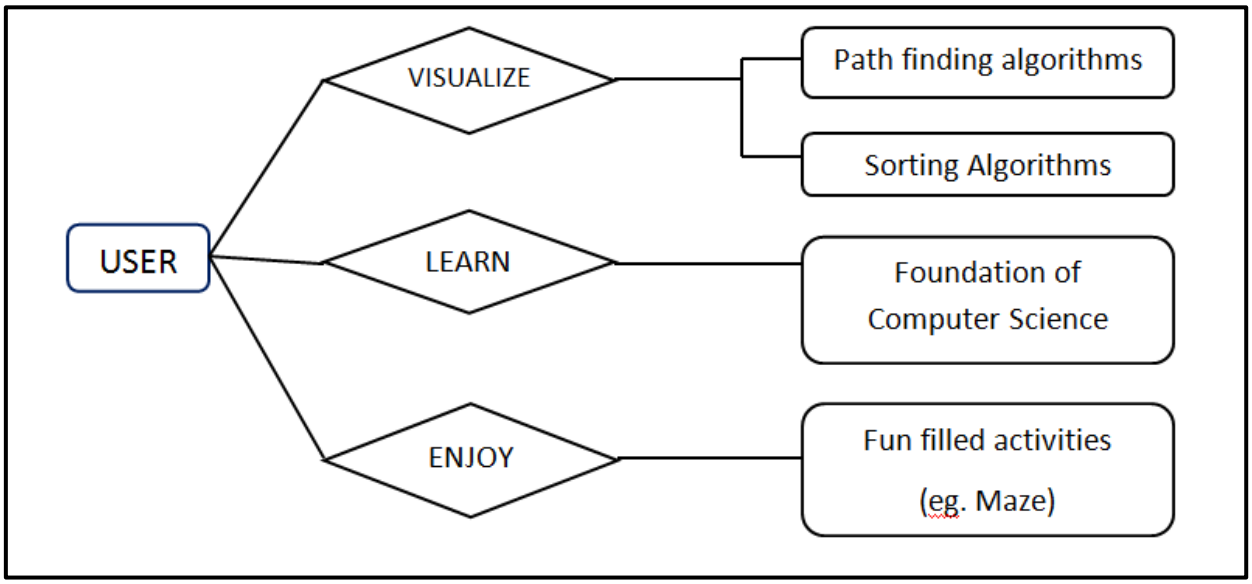

Figure-2: ER model of Algorithm Visualizer

As we are able to see from the above model that the centre of attraction of our application is the user thus, we need to ensure great user experience (UX) which would enhance the overall impact of our application.

Since we did not have much complex relationships to manage in our application we decided to implement our app using some lightweight frameworks and scripting languages. Thus, JavaScript as the base language was an obvious choice owing to its lightweight nature and wide variety of framework options.

We then went through most of the popular JavaScript frameworks. We tested each of them by trying to implement a sample page and came to a unanimous decision that React.js was the best choice due to its features like reusability, easy testing and debugging, and component based approach.

Now, the only thing left was to decide how to structure our application to maximize its effectiveness. For this we analyzed a few existing designs over the internet and we finally decided on an architecture which has already been explained in the methodology section.

IV. RESULTS AND DISCUSSION

- Through a survey conducted by us we inferred that 60% of the students responded better to understanding concepts through visualization rather than their own imagination or the regular teaching methods.
- It's been proved time and again through different experiments and research on masses that any kind of visual aid such as an image, a video or even an animation clip tends to be remembered more by humans.
- Not only will the visualization help but due to features of mazes and patterns in our application, the students can relate the working of the algorithms to real life examples (likes obstructions in the form of walls).
- Often we see teachers struggling to make students understand concepts such as algorithms without it getting monotonous, that's where our project comes into play as a great teaching aid. Because of our user friendly and engaging interface the problem of distraction or losing interest tends to decrease, making it very efficient.
- Our project can easily be incorporated alongside our education system by promoting different ways of learning rather than the age old blackboard method as we just need to access a website hosted on the internet to use the application.
- And with uncertain times like nowadays, we cannot only afford to be dependent only on our teachers and one to one offline teaching to understand different concepts. E-learning is the new age learning technique and our project is a step towards reinforcing this method of learning.

V. CONCLUSION

In a nutshell, we identifies some issues by experiencing them ourselves in the present learning strategies in use and we tried to help better the scenario for aspiring students in this domain through or progressive web application.

When we ourselves were learning the subject of algorithms in our curriculum, we found it a bit difficult to relate and understand the practical implementation of the algorithms owing to the difficulty in communication of the concepts from the teachers to the students. We found that there were no proper means that the teachers could adopt to portray their ideas in a better and easy manner in front of the students.

So, we built an application which could help in the following ways:-

- It has been found that it becomes easier for humans to retain the concepts when learnt through visuals than just textual or speech explanations.
- Application is extremely user friendly so people of any age can engage and start learning new things right away.
- The application would also include various fun filled activities like visualization through mazes and patterns.
- This application will also include a parameter of time complexity which will be displayed after the particular sorting algorithm has completed its execution for better comparison.

- Almost all the famous and important algorithms will be present in the application for visualization with both path-finding and sorting algorithms present in same application, thus making it a one stop destination for the students of this domain.

VI. REFERENCES

[1] "E-learning Tool for Visualization of Shortest Paths Algorithms" by Daniela Borissova and Ivan Mustakerov, ResearchGate, July 2015.
[2] "Algorithm Visualization: The State" of the Field by Clifford A. Shaffer, Matthew L. Cooper, Alexander Joel D. Alon, Monika Akbar, Michael Stewart, Sean Ponce and Stephen H. Edwardsacm Transactions on Computing Education, Vol. 10, No. 3, Article 9, Pub. date: August 2010.
[3] "Visualizing sorting algorithms" by Brian Faria, Rhode Island College, 2017.
[4] "ViSA: Visualization of Sorting Algorithms" by Tihomir Orehovački, ResearchGate, May 2012.
[5] " Finding the shortest path in a graph and its visualization using C# and WPF" by Radoslav Mavrevski, Metodi Traykov, Ivan Trenchev, International Journal of Computers, ISSN: 2367-8895, Volume 5, 2020.
[6] "Visualization of Abstract Algorithmic Ideas" by Ludˇek Kuˇcera, Proceedings of the 10th International Conference on Computer Supported Education (CSEDU 2018).
[7] "Willow: A Tool for Interactive Programming Visualization to Help in the Data Structures and Algorithms Teaching-Learning Process" by Pedro Moraes and Leopoldo Teixeira, SBES 2019, September 23ś27, 2019, Salvador, Brazil.