PES UNIVERSITY,
BANGALORE
Department of Computer Science and Engineering

B-TECH

5TH SEM

UE20CS303

SOFTWAREENGINEERING
PROJECT REPORT
ON
ALGORITHM VISUALIZER

SUBMITTED BY

VISHAL M	PES1UG20CS508
PRASHANTH AKUARTHI	PES1UG20CS525
EKANATHREDDY	PES1UG20CS548
ANUPAMA	PES1UG20CS538

TABLE OF CONTENTS

SI.NO	Topic	Page No.
1	Proposal of the Project	3
2	Software RequirementSpecification	4
3	Project Plan	7
4	Design Diagram	9
5	Test Cases	10
6	Screenshots Of TheOutput	11
7	Conclusion	16
8	Future enhancements	16

1. Proposal of the Project

Algorithms Visualizations contribute to improve computer science education.
The method of teaching and learning algorithms is commonly complex to understand the problem.
Visualization is a helpful technique for learning in any engineering course.

In this report, an e-learning tool for **Pathfinder, Prime Numbers, Sorting Algorithms, N Queen, Convex Hull, Binary Search Game** visualization is described.

For example, In sorting the animation tool would represent information as a bar and once choosing a data-ordering and algorithms,
the user will run an automatic animation or step through it at their own pace.
In path finding making the starting and the end node be able to move around or the user to choose wherever he wants it to start or end.
The developed e-learning tool permits visualizes the algorithm rule steps execution.
It's mean to be used as a supplement to face-to-face instruction or as a complete application.

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 makes Data structures learning more interesting. The main goal of this project is to implement a system for various sorting algorithm , prime number , binary search game , - for investigation and visualization the best and worst case for every implemented algorithmic rule.

2. Software Requirement Specification :

1.Introduction

1.1)Purpose:

Purpose of this project is to create an Algorithm Visualizer web-app

Learning an algorithm gets much easier with visualizing it.

1.2) **Scope:** This application will be useful for:

- Students

students can grasp the intuition of algorithms.

- Lecturers

it would a give a better medium to teach algorithm rather than relying on PPT.

- Programmers

1.3) **Overview:**

- It mainly aims to simplify and deepen the understanding of algorithms operation.
- Possibility of enriching the standard methods of teaching algorithms, with algorithm visualizer.
- It will makes algorithm learning more interesting.

1.4) **References:**

Digitalcommons.edu , IEEE explore and others

2. Overall Description

2.1) **Product Perspective:** The target is an algorithm visualizer

software whose target audience are majorly students and lecturers

2.2) **Product Function:**

- Takes user input on data and constraints
- Displays visualizations of outputs of algorithms
- Algorithmic information: includes description, pseudo-code, complexities, real world applications

2.3) **User Characteristics:**

- Should be easily understood by students, preferably above 10^{th} grade
- Lecturers who want to teach a particular topic in an easy way
- Programmers who want to develop and brush up their skills

2.4) **Constraints:**

- Limited number of important algorithms
- Time taken to implement algorithms

3. Specific Requirements:

3.1) Functional Requirements:

- Should display the output of algorithm step by step on the screen
- Correctness of the output

3.2) Non Functional Requirements:

- Clarity of the output
- Time taken to display the output

3.3) Design Constraints:

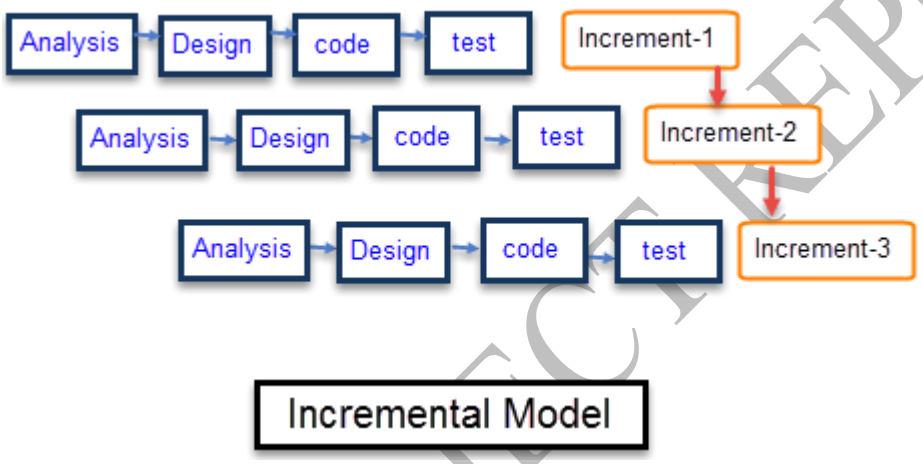
Software language used: The languages that can be used for Algorithm Visualizer are JavaScript , CSS and HTML

Type of Model:

Incremental model:

Incremental Model is a process of software development where requirements divided into multiple standalone modules of the software development cycle. In this model, each module goes through the requirements, design, implementation and testing phases.

Example flowchart:



Incremental Model

We are using an incremental model for this SE project

3.PROJECT PLAN

Every software engineer should have a good understanding of DSA to develop efficient software. Visualizers have a good history of providing effective understanding to the users.

Many algorithm visualizers have been developed over the years.

In 2008, paper "AlCoLab: Architecture of Algorithm Visualization System" concerns the style of script supported algorithm visualization systems for educational purposes, focusing on the support and the improvement that those systems provide in the process of teaching of an conceptual subject such as algorithms.

In 2019, paper “Towards Developing an Effective Algorithm Visualization Tool for Online Learning” reports a work-in-progress research project at Athabasca University on developing an effective algorithm visualization tool for online learning.

In 2019, paper “Open Interactive Algorithm Visualization ” presents a work-in-progress project form developing an open interactive algorithm visualization website.

In 2021, paper “AlgoAssist: Algorithm Visualizer and Coding Platform for Remote Classroom Learning” focuses on "algorithm visualization", which allows a better understanding of its flow and operation. It supports the combination of the lab into a single application dedicated to pre-assessment, algorithm explanation, visualization, coding, and post-assessment.

In 2021, paper “Algorithm Visualizer ” aims to simplify and deepen the understanding of algorithms operation. Within the paper we talk about the possibility of improving the standard methods of teaching algorithms, with the algorithm visualizations

4. DESIGN DIAGRAM

A. System Architecture :

The proposed system involves the simulation of the different type of algorithms codes. As you can see, there are no major components besides the three coding languages. Most websites have tools or scripts that require a server on the back-end (like PHP), but it is not necessary in this case since React JS runs right in the user’s browser

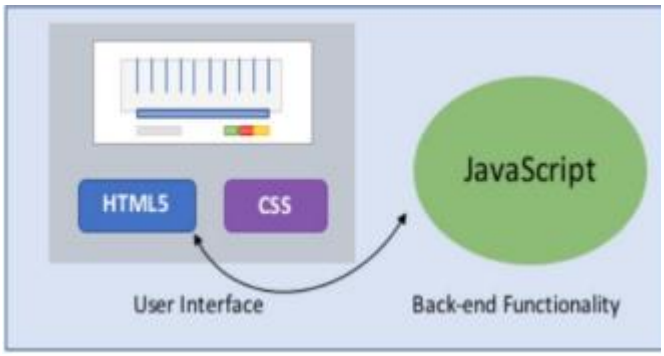


Figure 1: System Architecture 1.

HTML5 and CSS are used for the interface. The HTML5 communicates with the React JS code and vice versa to launch the appropriate algorithms and update the interface accordingly, as seen with a single, bidirectional arrow. As the React JS was modified from a functional programming focus to a more object-oriented one, the parts of the HTML5 that did change were the function calls for each button. All of the back-end interaction is abstracted to the various buttons for selecting algorithms and running the animation.

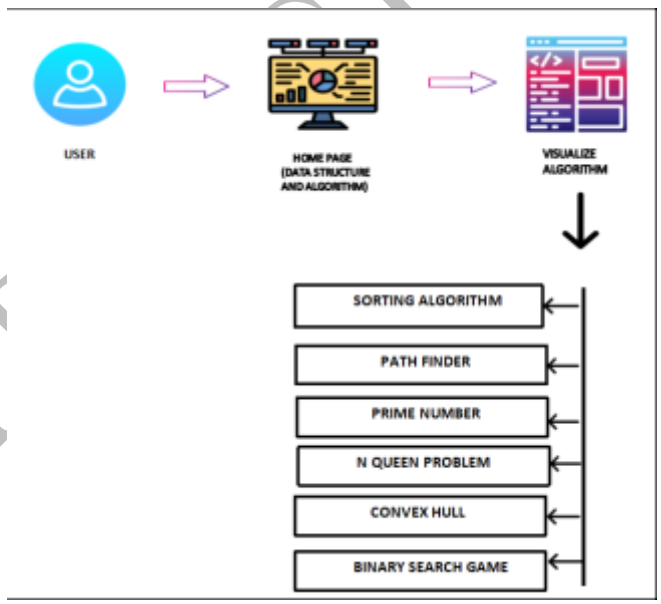
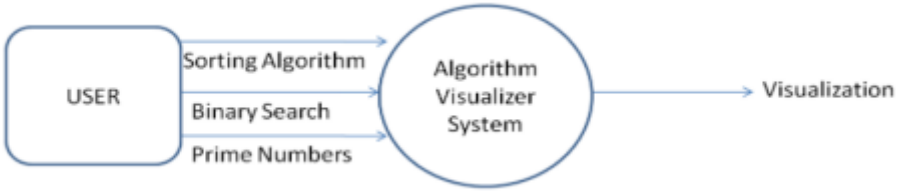


Figure 2: System Architecture 2

B. Data Flow Diagrams:

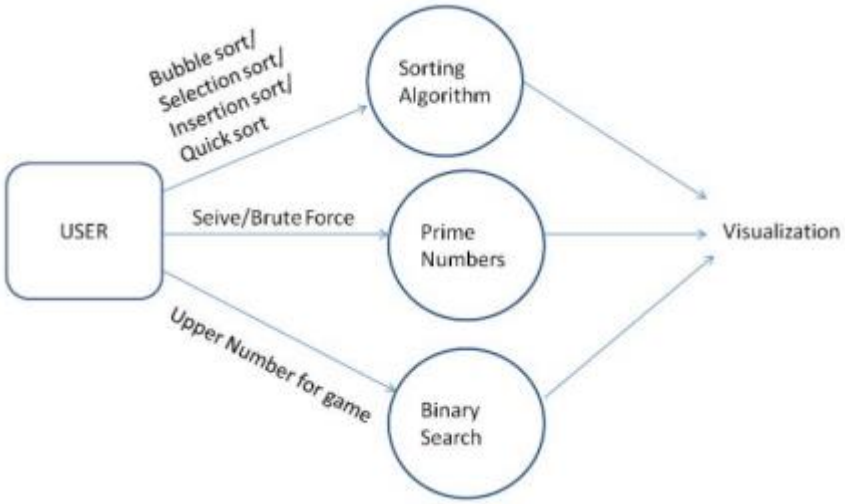
1. DFD LEVEL-0



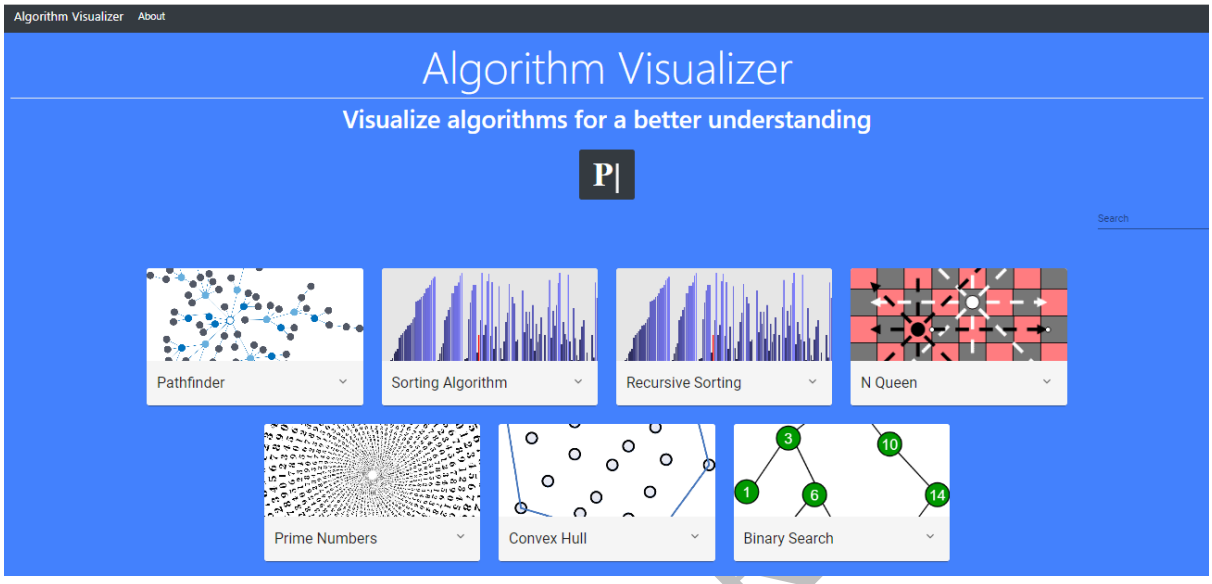
In Data Flow Diagram we Show the flow of data in our system, In DFD0 we show the base DFD in which rectangle present External entity (an outside system that sends or receives data)and circle show a Process(process that changes the data, producing an output).
The arrows towards the process shows input while the arrows away from the process shows output.

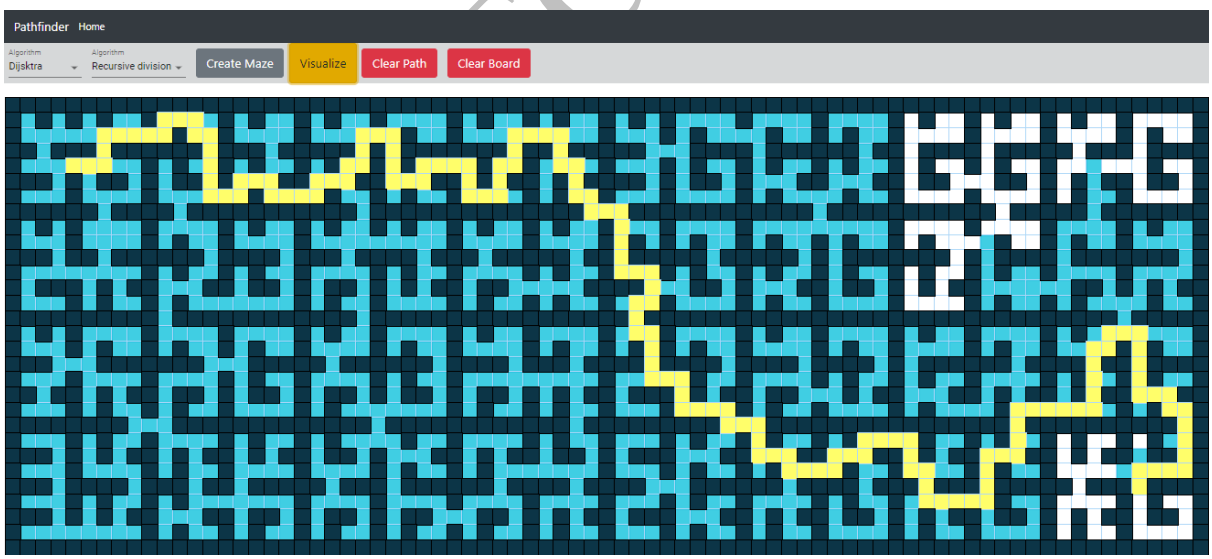
2 .DFD LEVEL-1

DFD1 is the further bifurcation of DFD0.



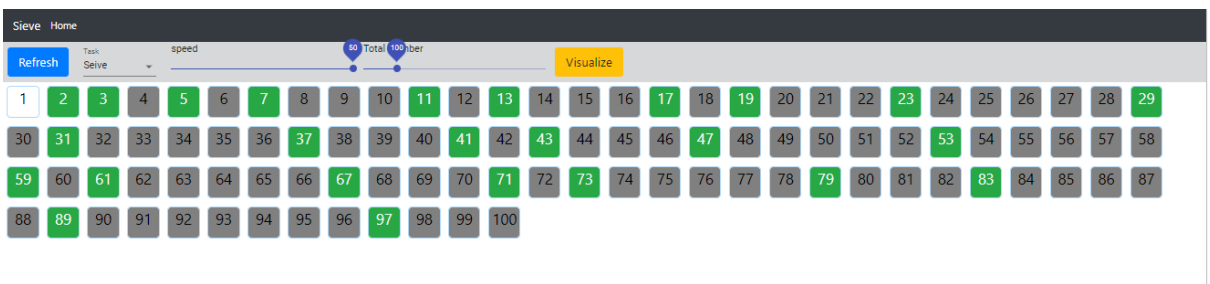
6. SCREENSHOTS OF THE OUTPUT

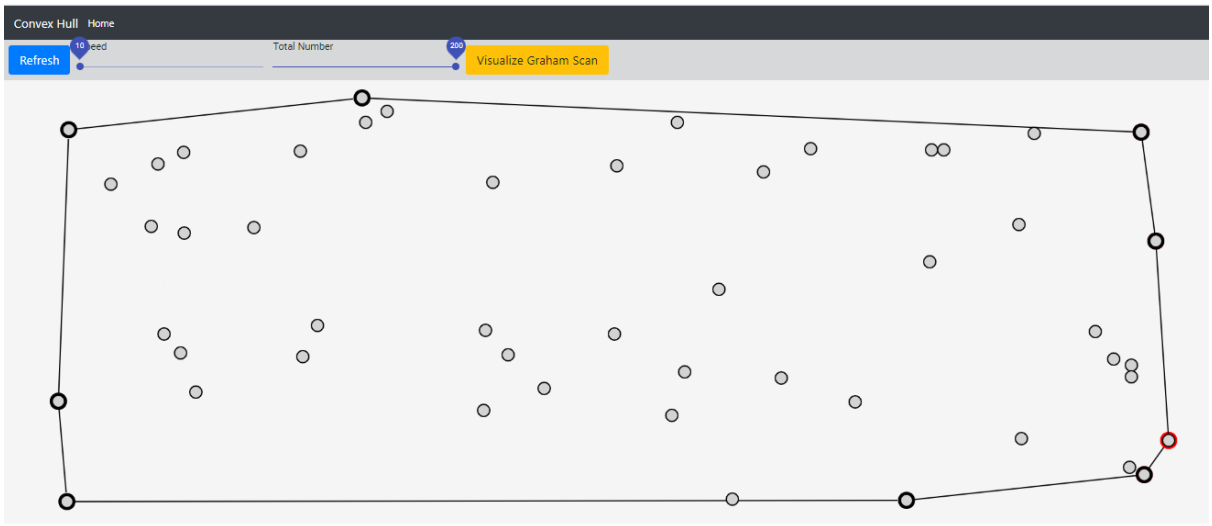


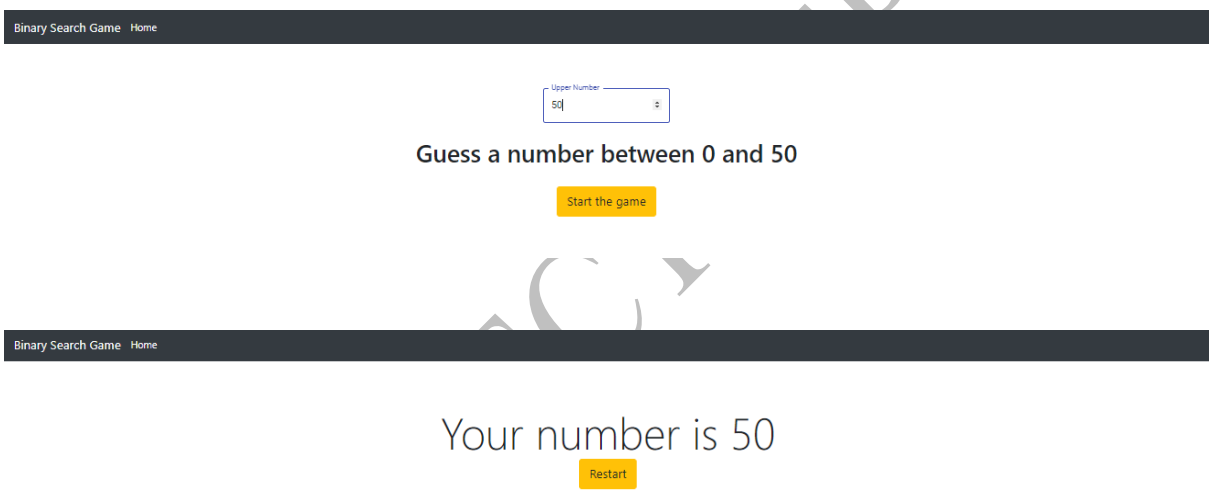



Sorting Visualizer
Home
Randomize
Value range
Numbers
Speed
Algorithm
Bubble Sort
Duo
Algorithm
Selection Sort
Visualize


N Queen
Home
Clear Board
Grid size
Speed
Visualize


Sieve
Home
Refresh
Task
Seive
speed
Visualize
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29
30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58
59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87
88 89 90 91 92 93 94 95 96 97 98 99 100





7.Conclusion

We started our project by studying a number of the well-known algorithm visualizations that are developed over a few years.
According to our findings, algorithmic visualization are often seen as a valuable supporting tool, utilized in addition to straightforward ways of education within the field of computer science.
With the execution of this project, we have got with success attain our objective of our project is to engraft Graph Path Finding with Visualization and differentiate their performance.
As is the case with most other teaching areas, there has been a major gap

between the idea and practical understanding of algorithms realization. This is often true for shortest paths algorithms and in specially for Dijkstra algorithm. The main goal of the project is to use it from research educators and students for teaching and studying the existing known graph algorithms.
The main plan of the system is to provide an associate educational environment for both instructors and students to facilitate the learning process in economical way.

8.Future possible enhancements:

With the shift of remote and digital literacy, a combined platform serving the effective literacy requirements of students is required. Algorithm Visualizer is a combined platform that's a comprehensive result for educators and students to educate and learn online effectively. It substantially focuses on “algorithm visualization", which allows a better understanding of its inflow and operation.