# Path Finding Visualizer

**MINI PROJECT – II SYNOPSIS**



Department of Computer Science & Application

## Institute of Engineering & Technology

**SUBMITTED TO: - SUBMITTED BY: -**

**Mr. Ankit Arora Deepak Singh (201500209) Jaideep Gautam (201500310)**

**Srashti Gautam (2115990020)**

**Tushar Sharma (201500746)**

# ACKNOWLEDGEMENT

It gives us a great sense of pleasure to present the synopsis of the B.Tech mini project undertaken during B.Tech III Year.

This project is going to be an acknowledgement to the inspiration, drive and technical assistance will be contributed to it by many individuals.

We owe special debt of gratitude to Mr. Ankit Arora (Technical Trainer), for providing us with an encouraging platform to develop this project, which thus helped us in shaping our abilities towards a constructive goal and for his constant support and guidance to our work.

His sincerity, thoroughness and perseverance have been a constant source of inspiration for us. We believe that he will shower us with all his extensively experienced ideas and insightful comments at different stages of the project & also taught us about the latest industry-oriented technologies.

We also do not like to miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind guidance and co-operation.

# ABSTRACT

Visualization is an efficient way to learn concepts faster than traditional methods. Modern technology enables the creation of e-learning tools, which also greatly contribute to the improvement of computer science education.

The goal of this project is to create the Pathfinding Visualizer, a web-based e-learning tool that can be used to visualize the Shortest Path algorithm. Conceptual applications of the project are illustrated through implementations of algorithms such as Dijkstra’s and DFS. This project aims to perform all these tasks with some knowledge of HTML, CSS, JavaScript, and the React framework. Since the final product is a web application, users can easily see and learn how the algorithm works.

The ease of use of the project provides users with simple operating instructions. First results using the application promise benefits of this here learning tool for students with a good understanding of the shortest path algorithm

# CONTENTS

1. Acknowledgment
2. Abstract
3. Introduction
4. Software Requirement
5. Project Description
6. Working
7. Implementation
8. References

# INTRODUCTION

Currently, e-learning is being promoted very strongly among from different fields. Modern technologies enable the development of visualization tools for topics such as various graph theory algorithms and their descriptions. The implementation of such e-learning tools is one of the most important prerequisites for the successful use of any e-learning system.

Learning by visualization has been shown to help improve learning ability. By providing a visual representation of what the destination node algorithms look like, applications aim to make them easier to understand. A good algorithm visualization tool will bring the algorithm to life by showing the traversal of nodes through the and animating transitions from one node to another.

Moreover, using online learning methods instead of face-to-face lectures has the power to reinforce learning in terms of improving student performance, increasing student satisfaction, and increasing student learning flexibility.

# SOFTWARE AND HARDWARE REQUIREMENTS

* VS Code
* ReactJS
* CSS
* HTML
* Javascript
* 4GB ram
* Window 10 / 11

# PROJECT DESCRIPTION

A pathfinding method searches a graph by starting at one vertex and exploring adjacent nodes until the destination node is reached, generally with the intent of finding the cheapest route. Although graph searching methods such as a breadth-first search would find a route if given enough time, other methods, which "explore" the graph, would tend to reach the destination sooner. An analogy would be a person walking across a room; rather than examining every possible route in advance, the person would generally walk in the direction of the destination and only deviate from the path to avoid an obstruction, and make deviations as minor as possible.

Two primary problems of pathfinding are to find a path between two nodes in a graph and the shortest path problem—to find the optimal shortest path. Basic algorithms such as breadth-first and depth-first search address the first problem by exhausting all possibilities; starting from the given node, they iterate over all potential paths until they reach the destination node. These algorithms run in O(|V|+|E|), or linear time, where V is the number of vertices, and E is the number of edges between vertices.

# WORKING

The starting node is called the initial node, the algorithm assigns initial distance values to all other nodes, and the updates them incrementally.

First, mark all nodes as unvisited and put them into the set. Assigns a tentative distance value of zero to the start node and all other to infinity. Set the current node to display as the first nodes.

For the initial node, tentatively remove unvisited neighbouring nodes, compare the newly found value with the value originally assigned to, and replace it with the smaller.

If all unvisited nodes are considered current nodes, place the current node as a visited node on the grid and draw it from the set of unvisited nodes.

If the destination node is found and marked as visited (while finding the route between the two given points), or the minimum tentative distance between the source and all destinations in the unvisited set is infinite (something such a situation occurs when there is no relationship between the source and target nodes in the unvisited set), then stop. Otherwise, choose the unvisited node with the smallest interim distance, consider this node the current node. There is only one source and multiple destinations. Dijkstra's algorithm helps create the road network and reduces the cost of building roads.

Applicable to network routing protocols. The shortest route will be opened first in the middle and middle system. Applicable to improving the movement of robot systems

# IMPLEMENTATION:

It turns out that there is a big gap between theory and practical understanding of algorithms. This is also true for the shortest path algorithm, especially the Dijkstra algorithm. The main purpose of the e-learning tool is to use it to learn the well-known graph algorithm. Starting with Dijkstra, other shortest-path algorithms have been gradually implemented. The main idea of this system is to provide an integrated educational environment to facilitate the learning process in an efficient way.

The Pathfinding Visualizer tool has three steps.

1. Algorithm Selection

2. Node Placement

3. Visualization

# REFERENCES:

1. Javatpoint
2. The Shortest-Path Problem: Analysis and Comparison of Methods by Hector Ortega-Arranz , Diego R. Llanos , Arturo Gonzalez-Escribano
3. ResearchGate- https://www.researchgate.net › 3643...(PDF) "Path Finding Algorithm Visualization"
4. GeeksforGeeks
5. HTML and CSS- https://www.w3schools.com/

# Faculty Guidelines:

Mr. Ankit Arora (Technical Trainer GLA University)

# GitHub Repository link:

https://github.com/Srashtiiii/pathfinder.git