ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/360243038

QuadTree Visualizer

Presentation · April 2022

DOI: 10.13140/RG.2.2.14957.05609

CITATIONS
0

4 authors, including:

Amey Thakur
University of Mumbai
33 PUBLICATIONS **10** CITATIONS

SEE PROFILE

Mega Satish
University of Mumbai
24 PUBLICATIONS **6** CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Project Neuro-Fuzzy: Artificial Neural Networks & Fuzzy Logic View project

Project QuadTree Visualizer View project

All content following this page was uploaded by Amey Thakur on 28 April 2022.

The user has requested enhancement of the downloaded file.

QuadTree Visualizer

UNIVERSITY OF MUMBAI AFFILIATED INSTITUTE
TERNA ENGINEERING COLLEGE
Department of Computer Engineering

Group ID: TN04CS73

Under the Guidance of :
Prof. Randeep Kaur Kahlon

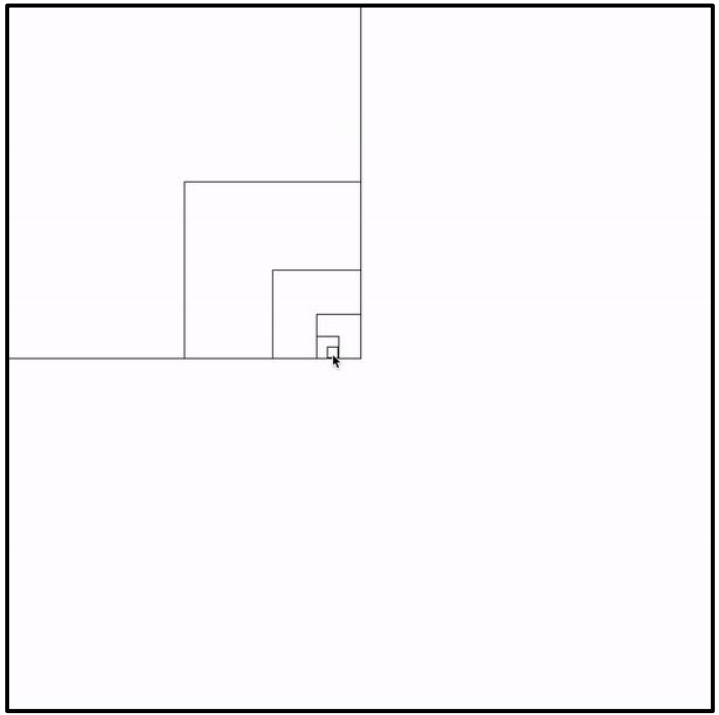
Group Members:

AMEY MAHENDRA THAKUR	TU3F1819127
HASAN MEHDI RIZVI	TU3F1819130
MEGA SATISH	TU3F1819139
AJAY RAMESH DAVARE	TU3F1718006

PROJECT GOAL

An application capable of presenting a view of the QuadTree.

Design and development of QuadTree view and data model.



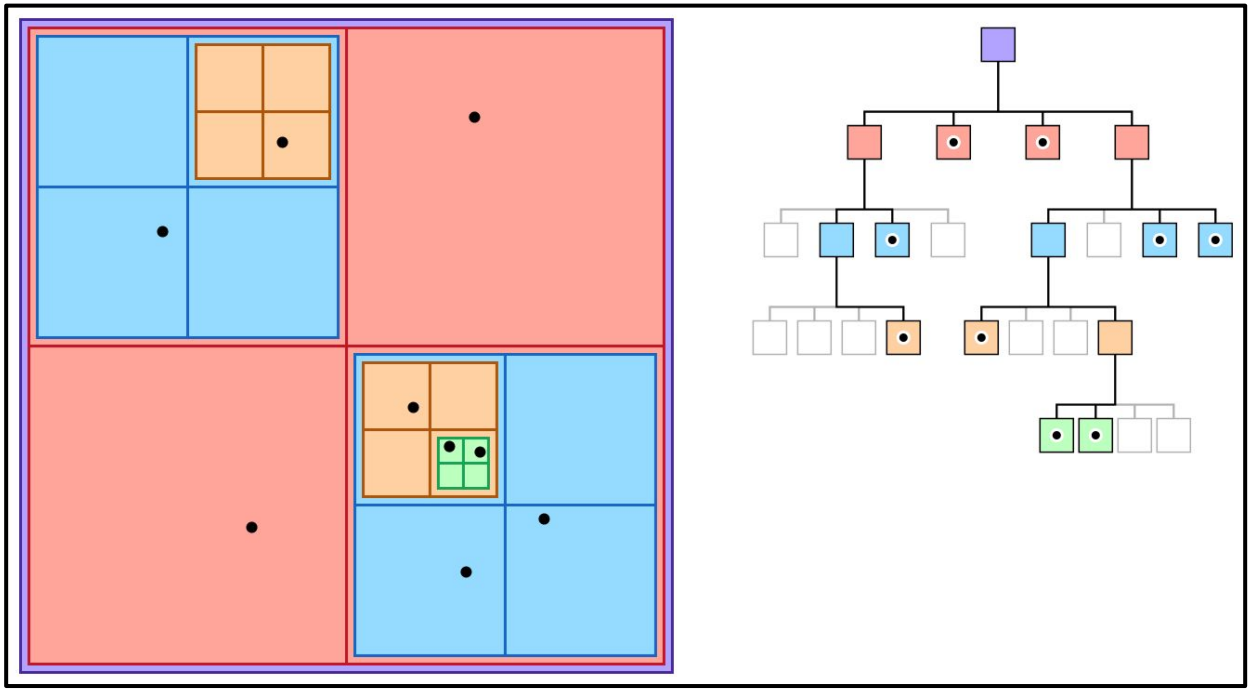
ABSTRACT

- Many digital map applications have the need to present large quantities of precise point data on the map. Such data can be weather information, the population in towns, etc. With the advancements in data science/ML, we expect such data will grow at a rapid pace. How to visualize such magnitude of data becomes a problem. QuadTree is a data structure in which each internal node has exactly four children. Quadtrees are trees implemented to efficiently store data of points on a two-dimensional space. This project aims to build an efficient visualizer for interactively visualizing the data. In this project, we will be using collision detection to illustrate the functionalities of QuadTree.

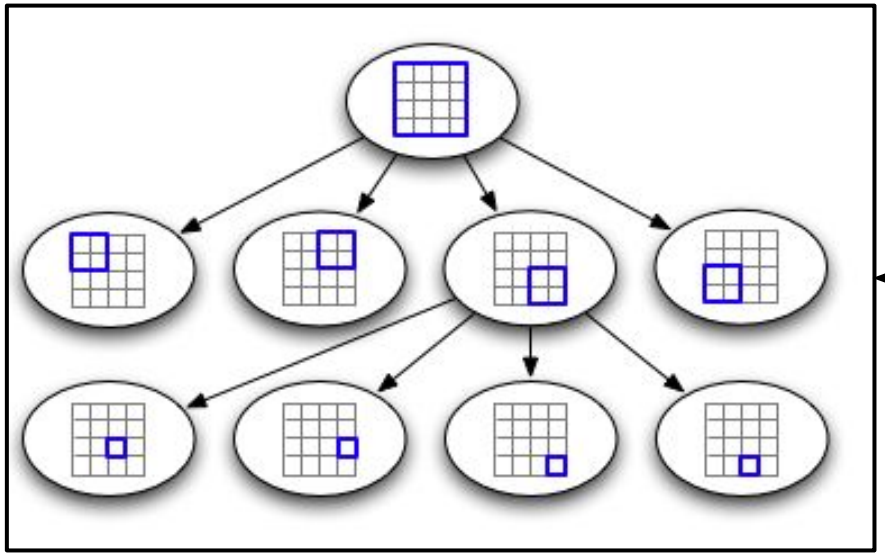
INTRODUCTION

What is QuadTree?

A data structure for organizing objects based on their locations in a two-dimensional space.



INTRODUCTION

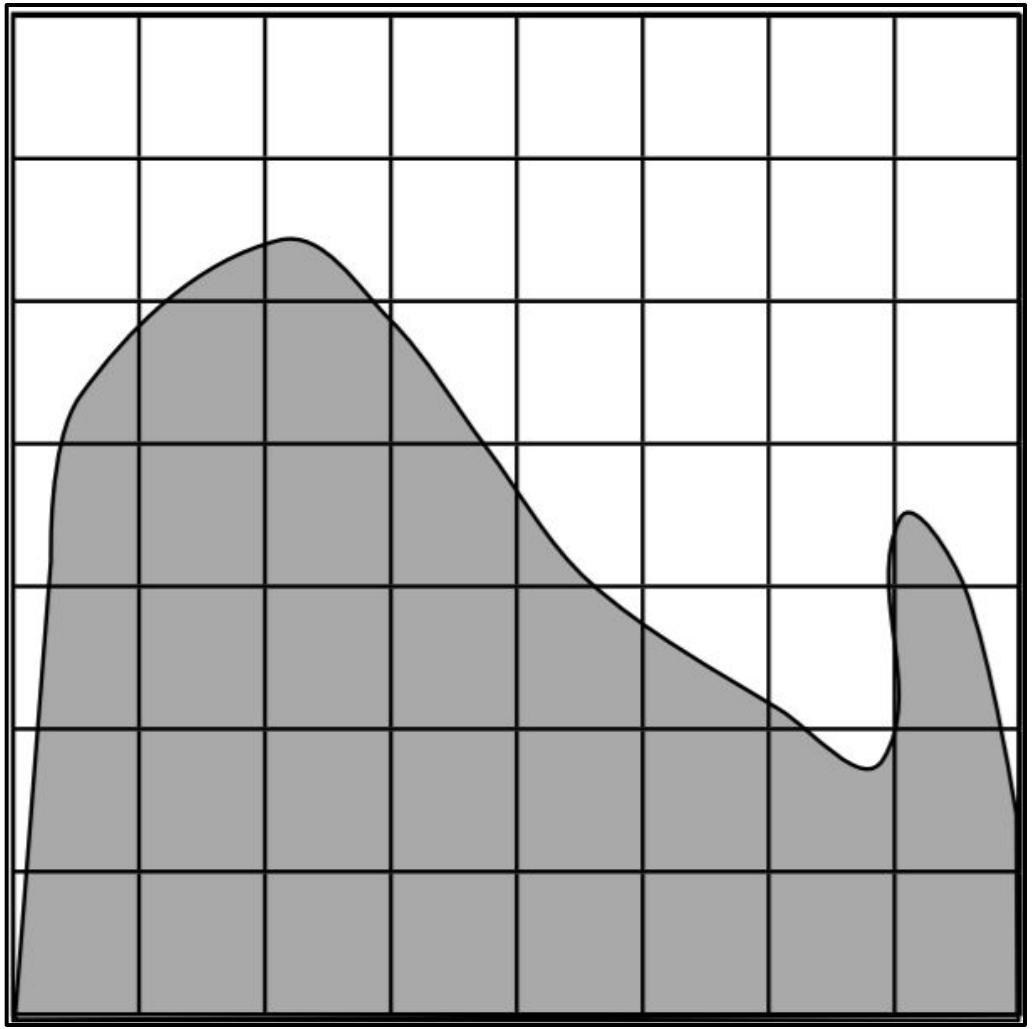


The QuadTree partitioning strategy divides space into four quadrants at each level. When a quadrant contains more than one object, the tree subdivides that region into four smaller quadrants, adding a level to the tree.

- So to speak in layman's term, a *quadtree* is a tree whose nodes either are leaves or have 4 children. The children are ordered 1, 2, 3, 4.

How does QuadTree Works?

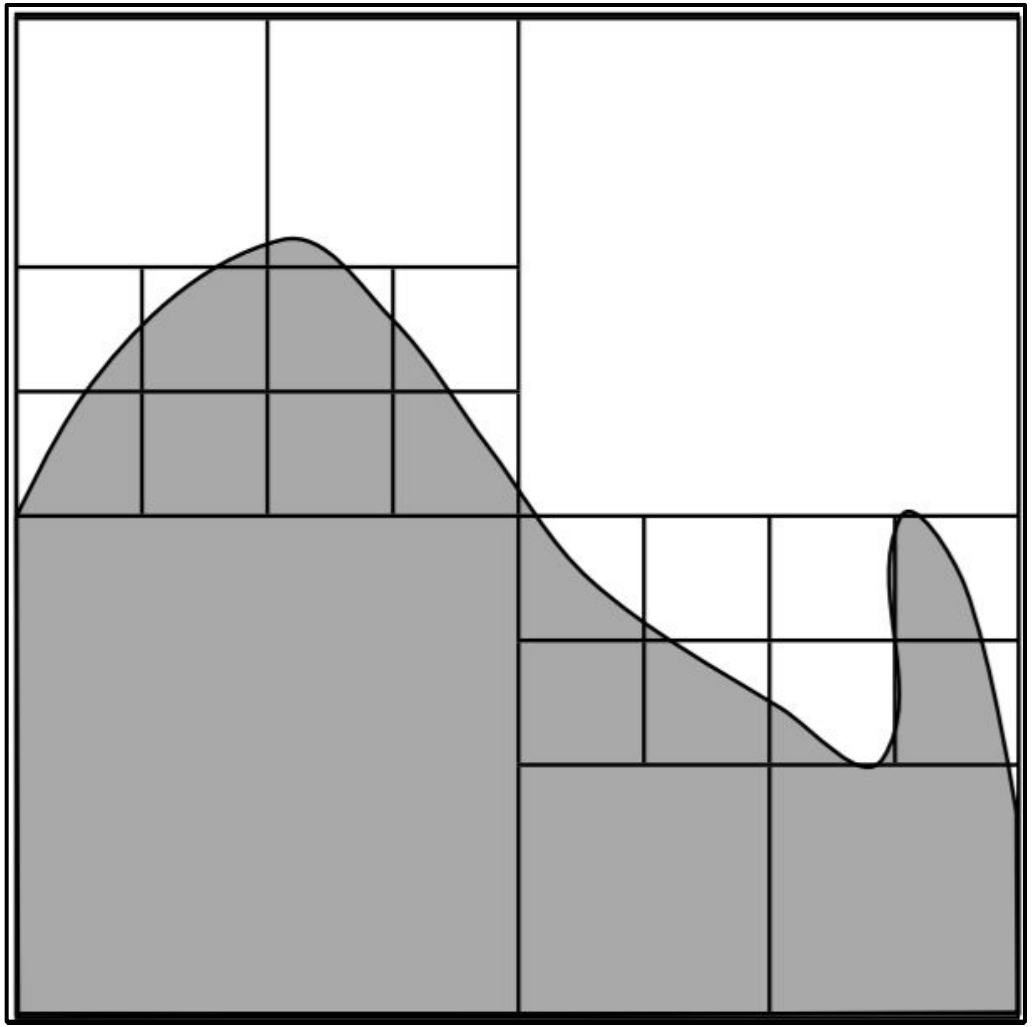
Subdivide into uniform blocks



How does QuadTree Works?

Subdivide into uniform blocks

Merge Similar Brothers

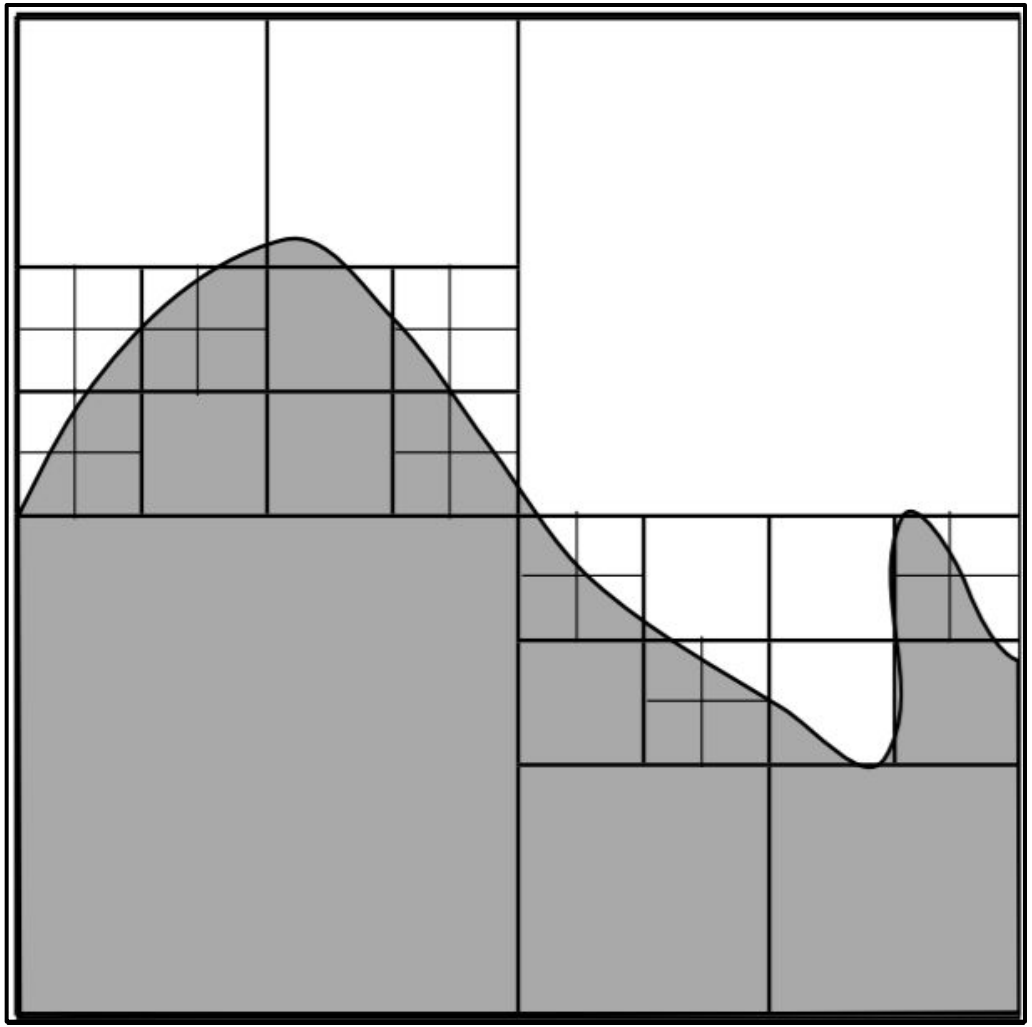


How does QuadTree Works?

Subdivide into uniform blocks

Merge Similar Brothers

Subdivide Non-homogenous Cells



How does QuadTree Works?

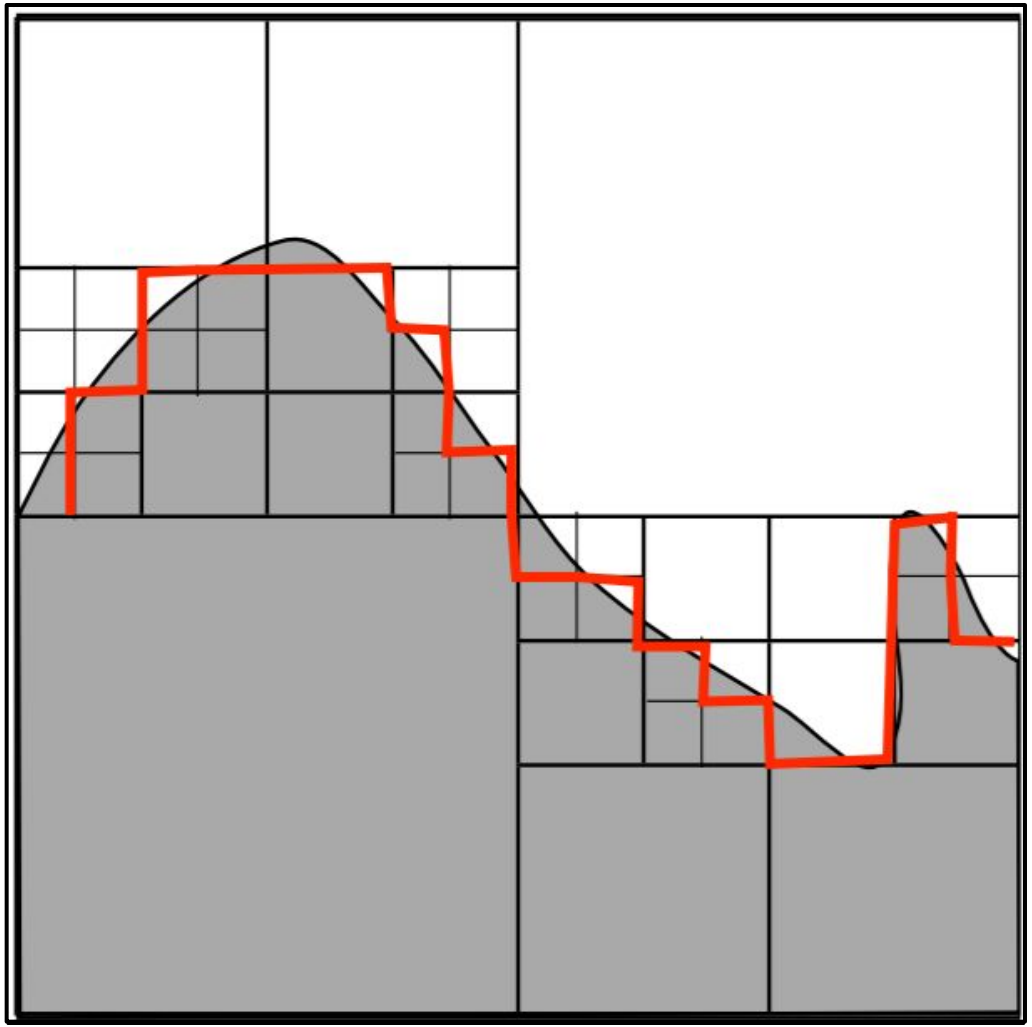
Subdivide into uniform blocks

Merge Similar Brothers

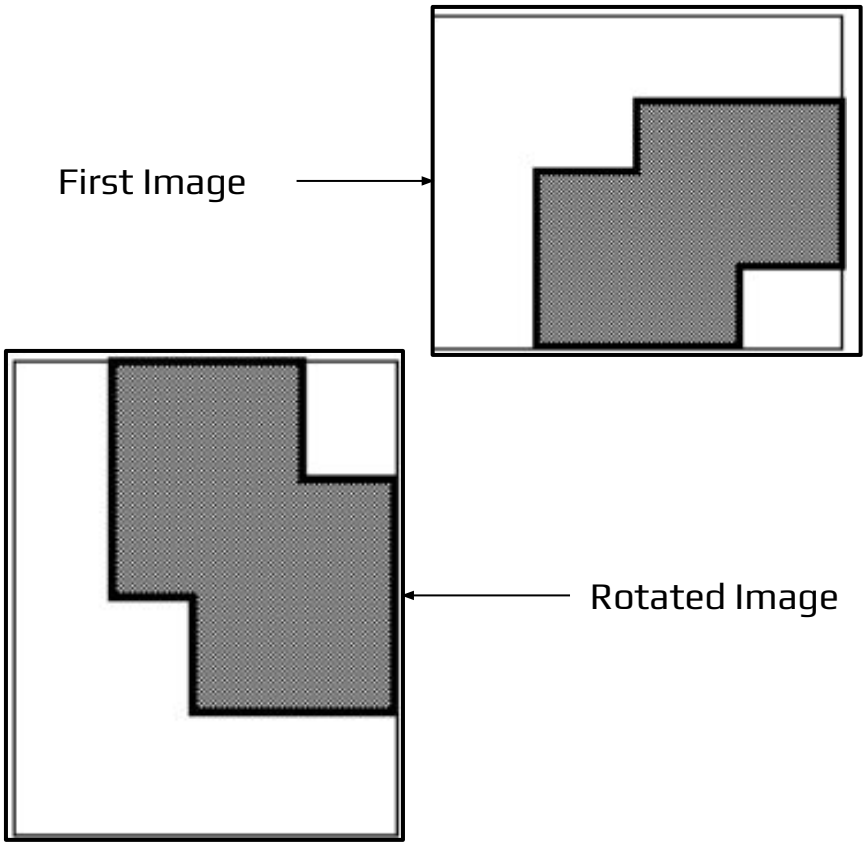
Subdivide Non-homogenous Cells

Group Identical Blocks

to get regions



LIMITATIONS OF QUADTREE



- ➔ The main disadvantage of quadtrees is that it is almost impossible to compare two images that differ only in rotation or translation. This is because the quadtree representation of such images will be so totally different.
- ➔ The algorithms available for rotation of an image are restricted to rotations of 90 degrees (or multiples thereof). No other rotation is available, nor is there a facility for translation.

OBJECTIVES

- This project's objective is to implement a quadtree visualizer that can be helpful in understanding working of QuadTree.

- QuadTree aims to be:
  - ➔ Versatile (can be used in dynamic and static contexts)
  - ➔ Simple
  - ➔ Lightweight
  - ➔ Easy to use
  - ➔ Fast

SCOPE

- We propose to develop a QuadTree library and a visualization tool for the same.
- This project aims to build an efficient visualizer for interactively visualizing the data.
- In this project, we will be using collision detection to illustrate the functionalities of QuadTree.

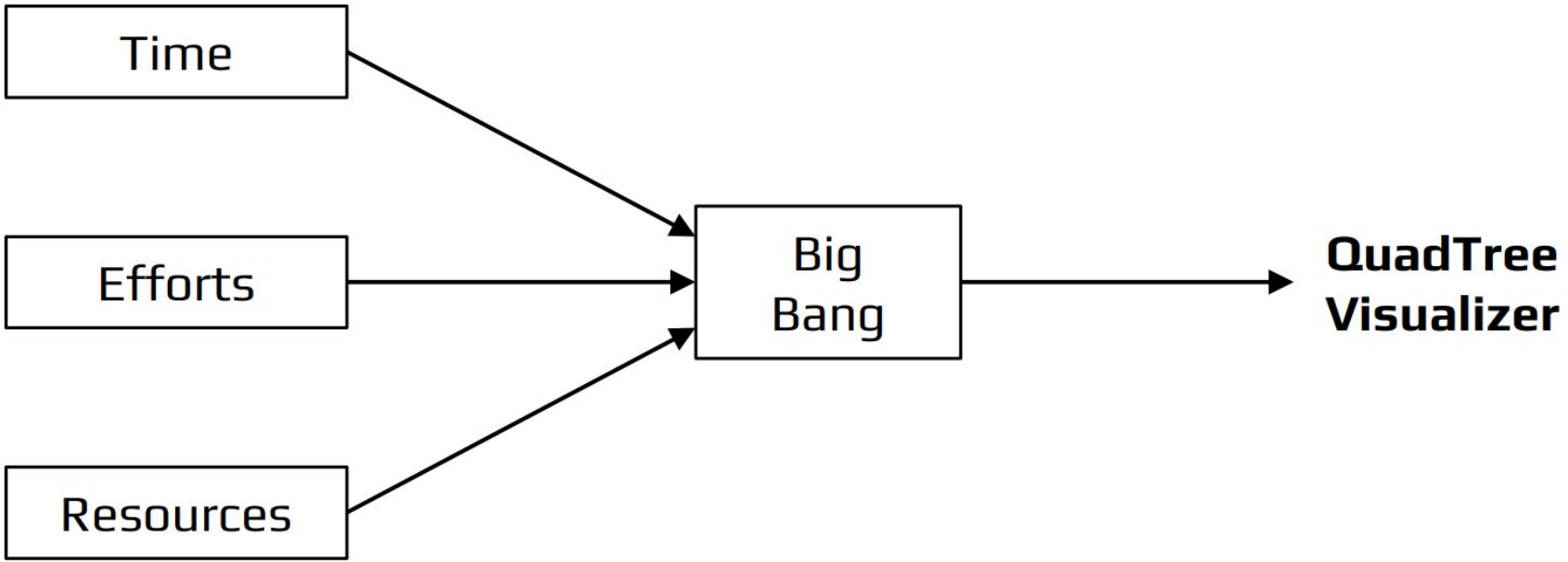
LITERATURE SURVEY

Author's Name	Title and Year of Publication	Findings
Irene Gargantini, Qing Cai, Yimin Zhou	An effective way to represent quadtrees, 1982	This paper proposes a new structure very similar to quadtree, called as "linear quadtree" and different algorithms used to represent that structure. The linear quadtree saves 66% of the computer storage required by regular quadtrees.
Clifford A.Shaffer, Hanan Samet	Optimal quadtree construction algorithms, 1987	In this paper, an algorithm for constructing a quadtree in time proportionate to the number of blocks in a given picture is described.
Qing Cai, Yimin Zhou	A quadtree-based hierarchical clustering method for visualizing large point dataset, 2016	This paper introduces a new clustering method with quadtree spatial indexing. It explains a grid- based, partitioning, hierarchical clustering method on quadtree file system storage.

PROBLEM STATEMENT

- The importance of data nowadays has increased significantly, as we are living in a data driven society. Many digital map applications have the need to present large quantities of precise point data on the map. With the development of the , we expect such data will grow at a rapid pace. However, visualizing and looking for a data point in such a magnitude of data becomes a problem. We are proposing the implementation of a quadtree visualizer to visualize data more easily.

THE BIG BANG MODEL



THE BIG BANG MODEL

- There is no planning required for this.
- Suitable for small projects.
- Very few resources are required.
- As there is no proper planning hence it does not require managerial staffs.
- Easy to implement.
- It develops the skills of the newcomers.
- Very much flexible for the developers working on it.

REQUIREMENTS, TOOLS & TECHNOLOGIES

Software Requirements

- ➔ GitHub
- ➔ VSCode
- ➔ Web Browser

Hardware Requirements

- ➔ 4 GB RAM
- ➔ Any Operating System

Tools Used

- ➔ NPM Dependencies
- ➔ CMD/Terminal

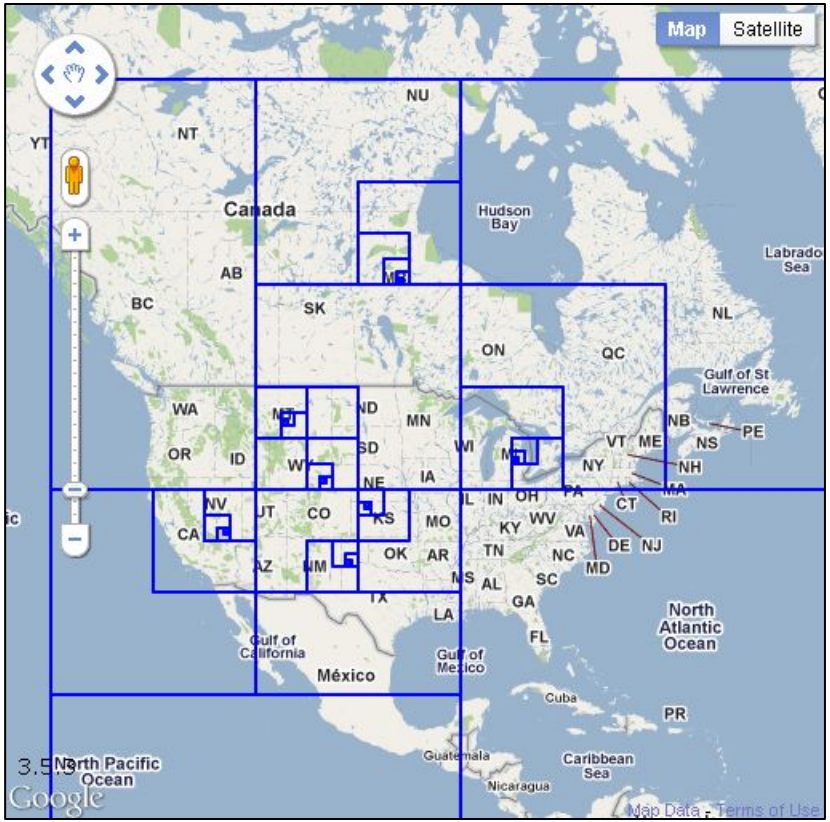
Technologies Used

- ➔ HTML 5.0
- ➔ CSS 3.0
- ➔ JavaScript, v. ES13
- ➔ TypeScript, v.4.6.3
- ➔ Node.js, v17.9.0
- ➔ Next.js, v10.0.5
- ➔ React, v17.0.1

USE CASES

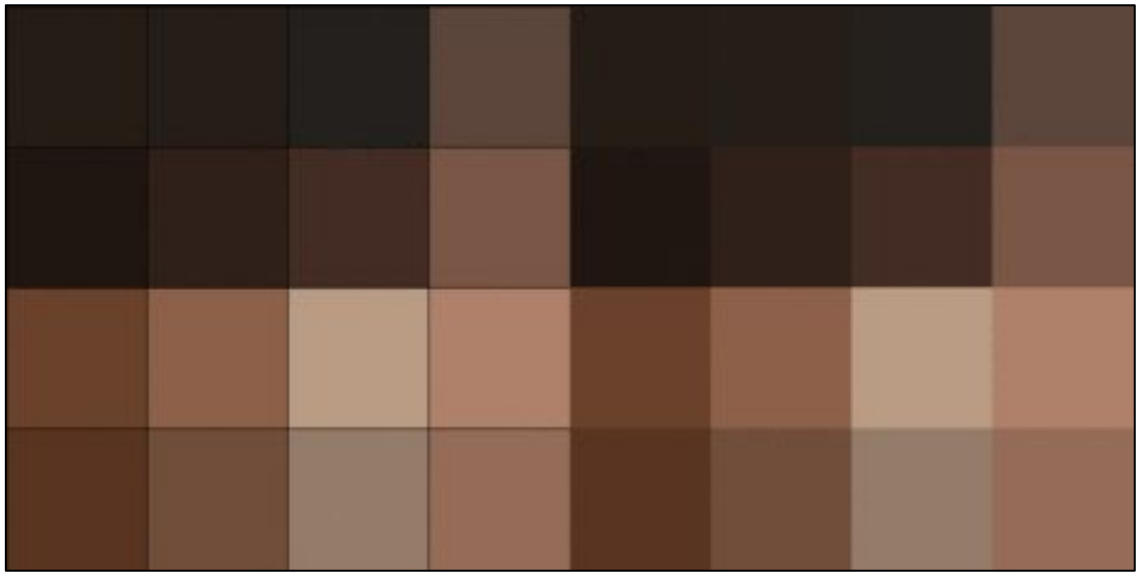
- Use Cases of QuadTree includes:
  - ➔ Image Processing
  - ➔ Sparse Data Storage
  - ➔ Spatial Indexing

- Examples:
  - ➔ Computer Graphics, Games, Movies
  - ➔ Computer Vision, CAD, Street Maps
  - ➔ (Google Maps/Google Earth)
  - ➔ Visualization (Graphing Complex Functions)



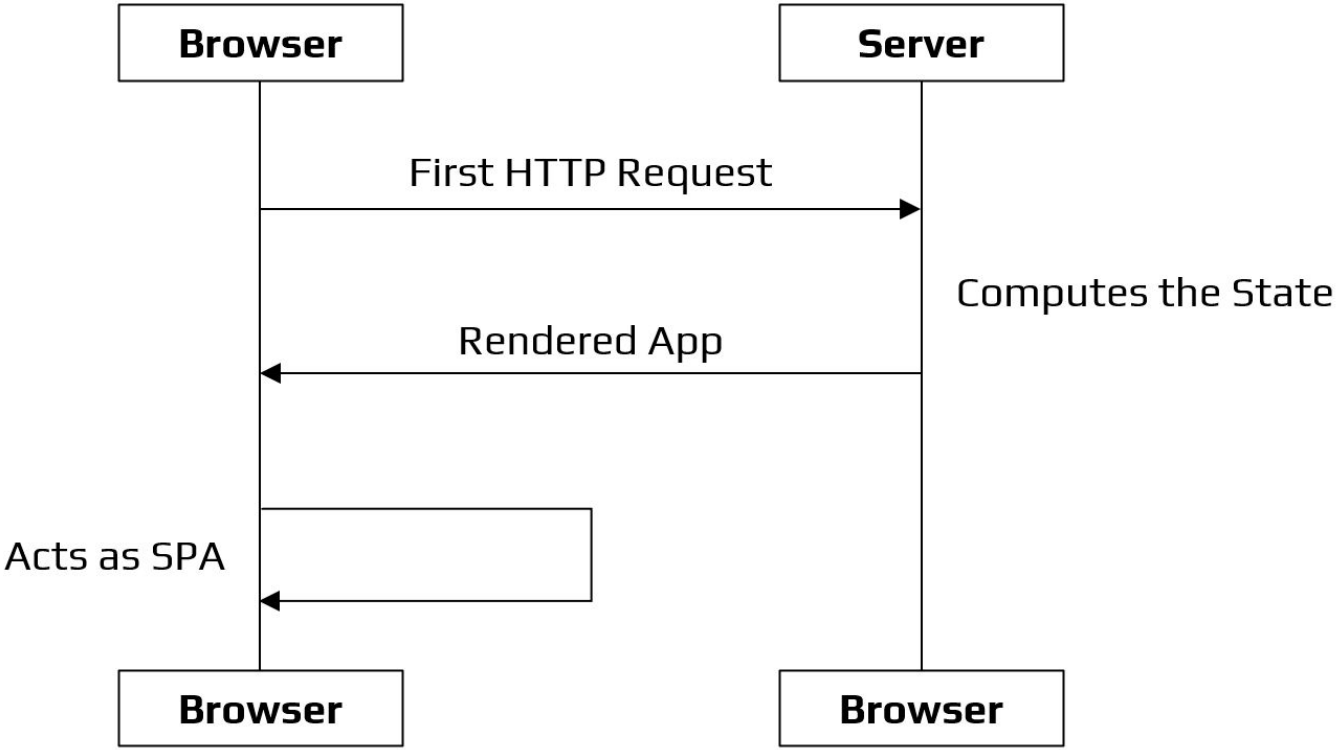
USE CASE - QuadTree compression of an image

Step-by-Step

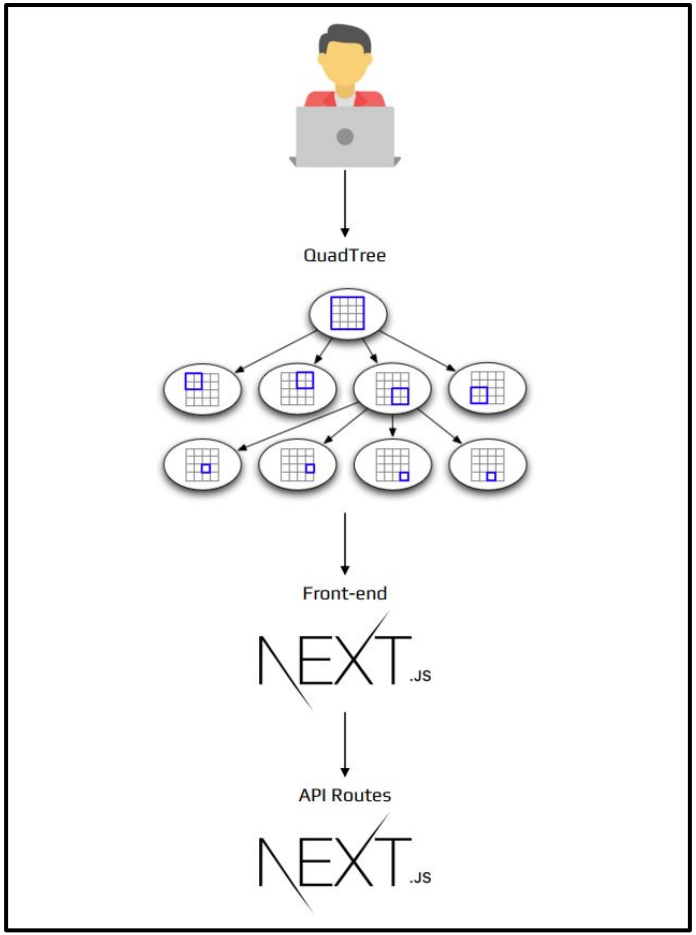


Left shows the compressed image with the tree bounding boxes while the right shows just the compressed image.

MODEL ARCHITECTURE

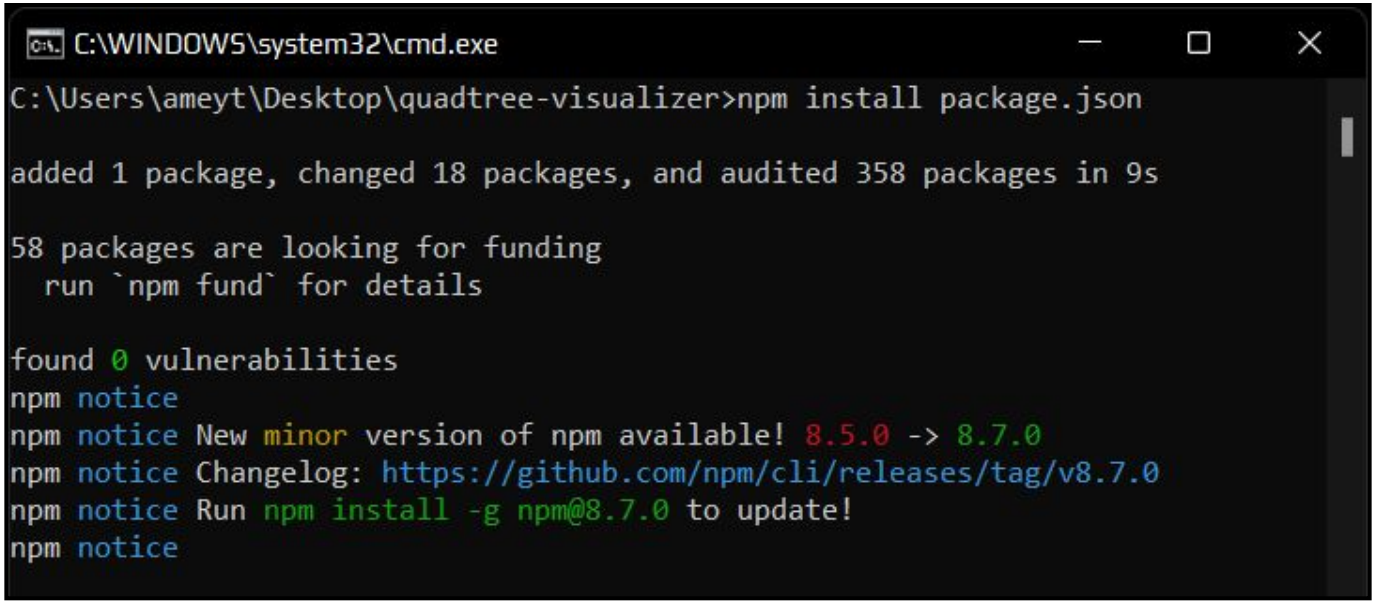


WORKFLOW OF QUADTREE VISUALIZER



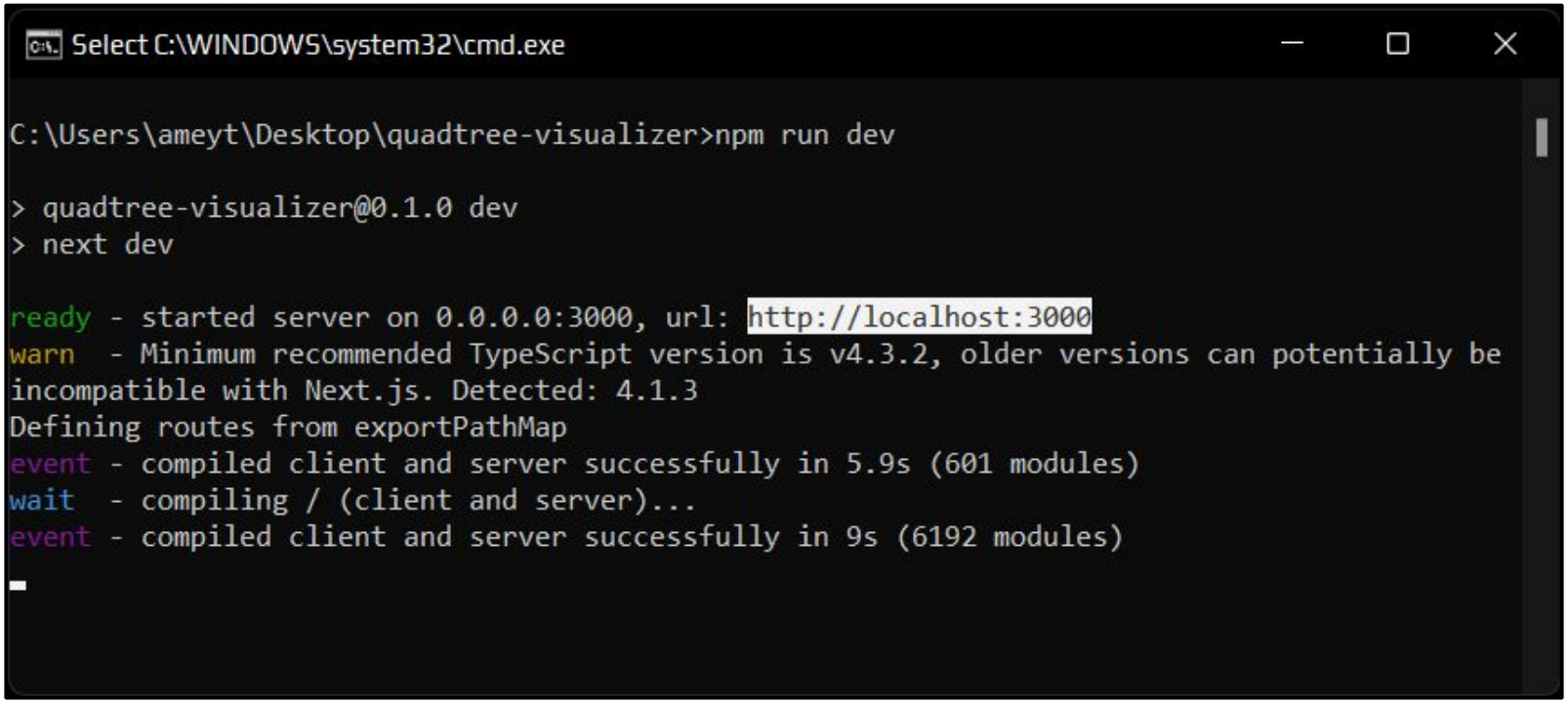
EXPERIMENTAL SETUP

- Since we are using Next.js in our project, we first need to have Node.js. The web application is currently working on http://localhost:3000.
- To run the application locally, we need to install the packages required using the npm command: ***npm install package.json***

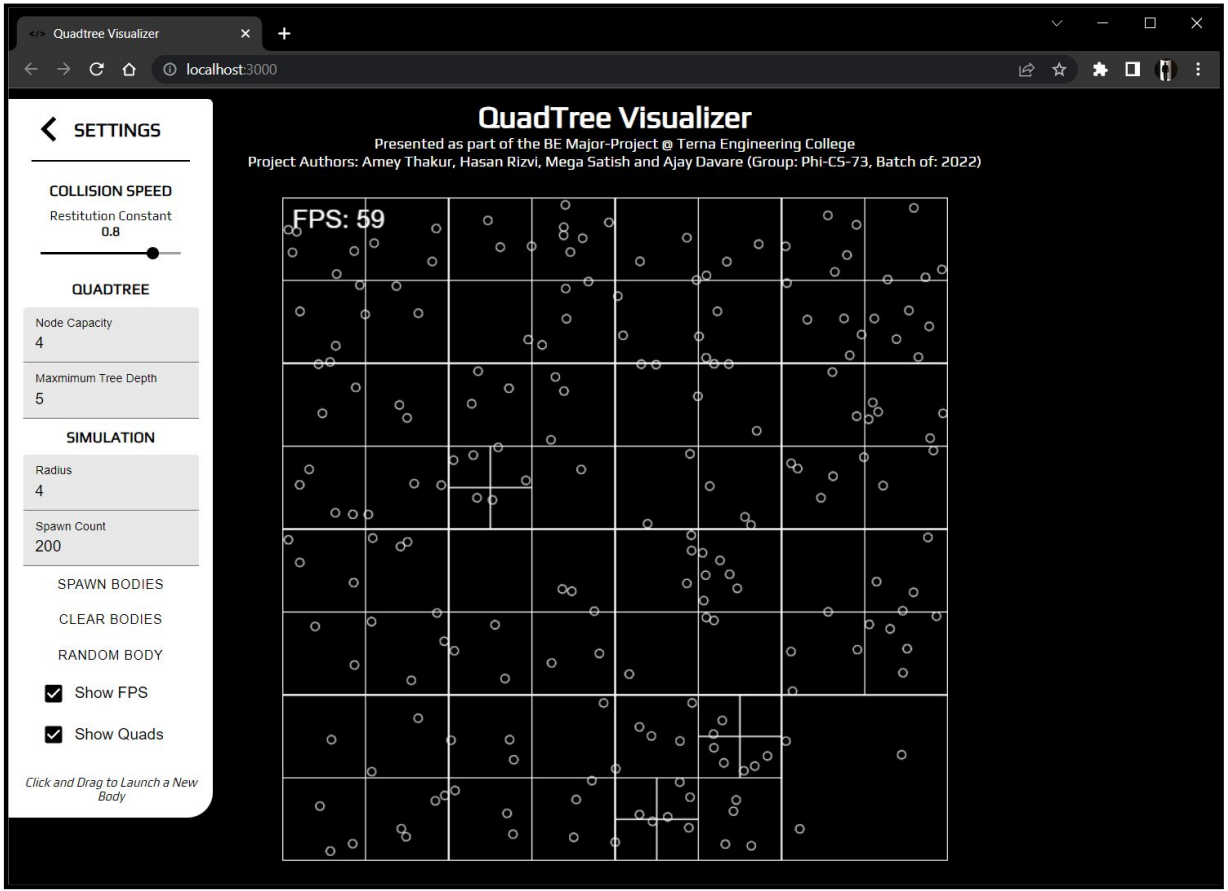


EXPERIMENTAL SETUP

- After installing all the dependencies, we then run the command: ***npm run dev***.
- This command will run the developer server.

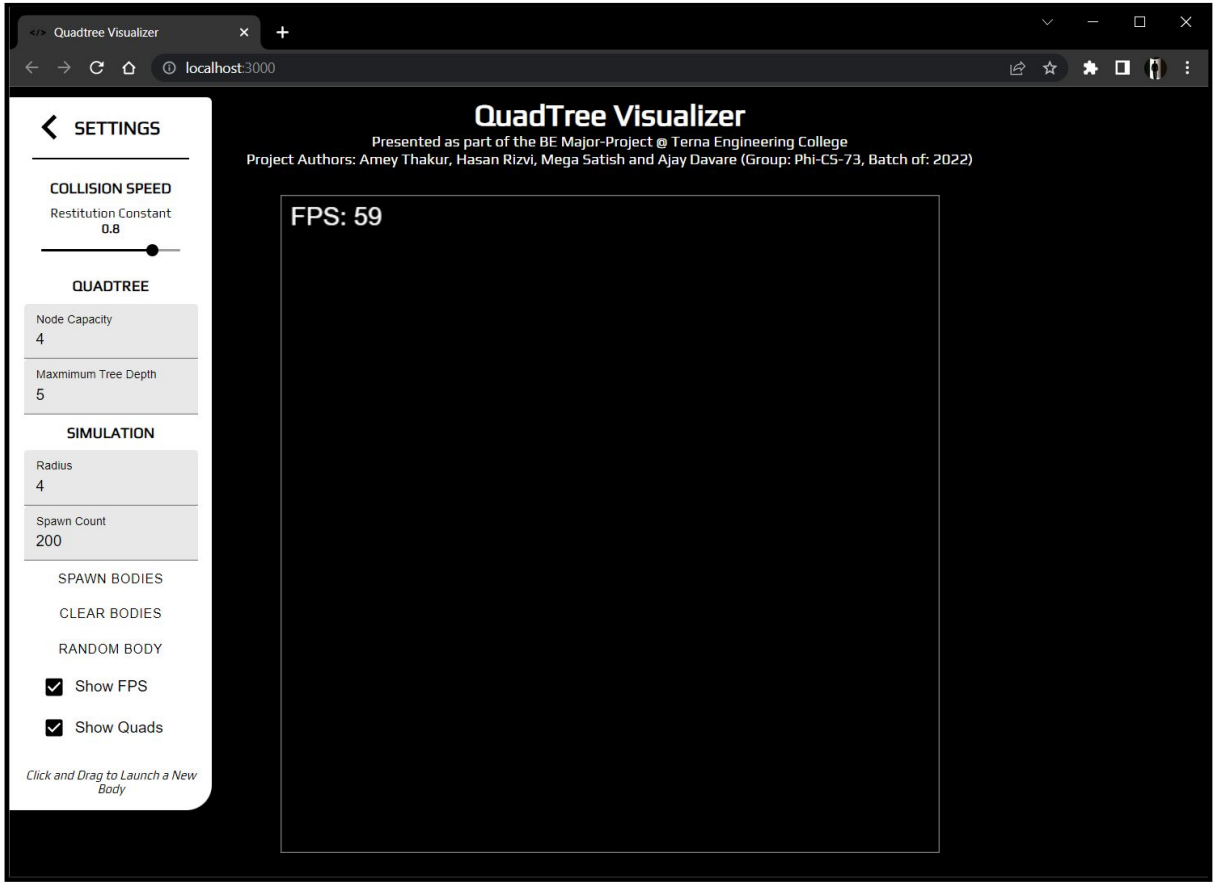


SNAPSHOTS



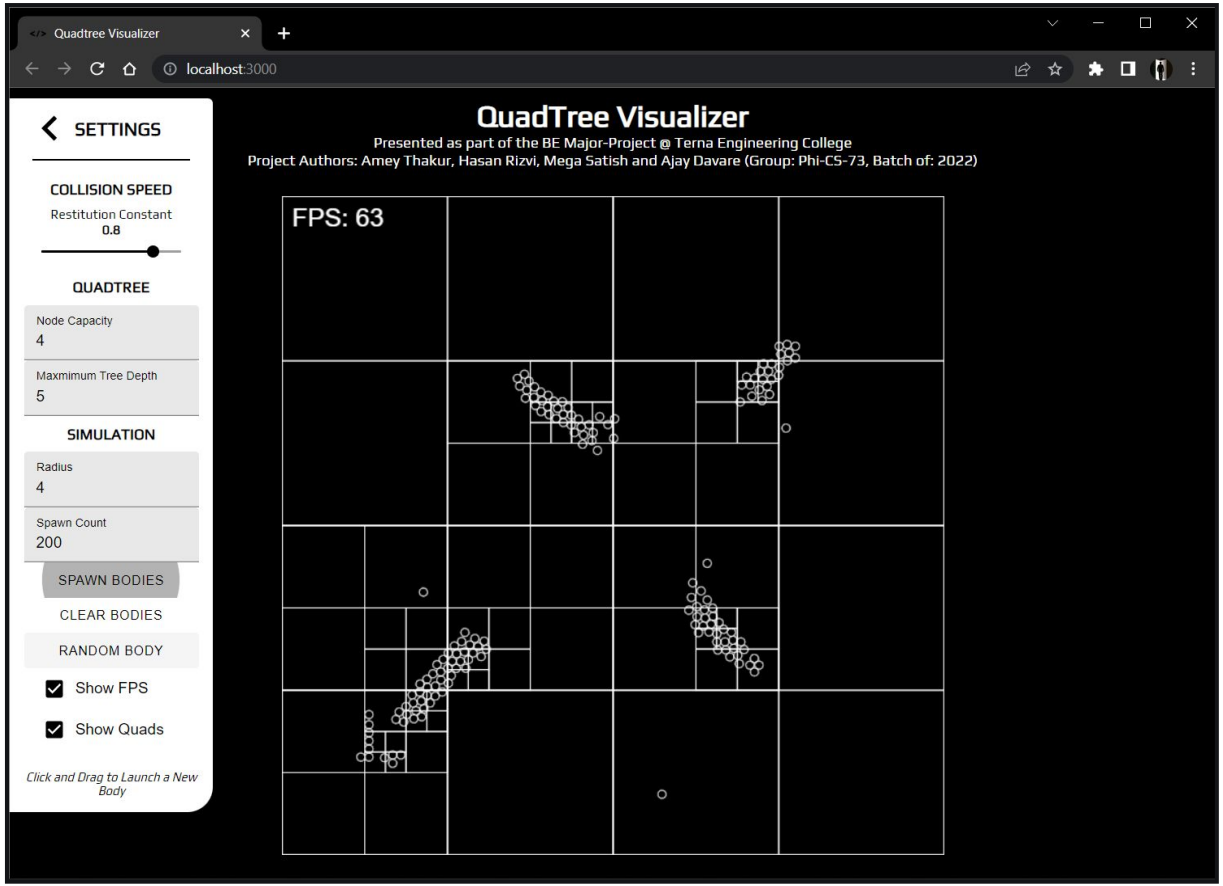
Homepage

SNAPSHOTS



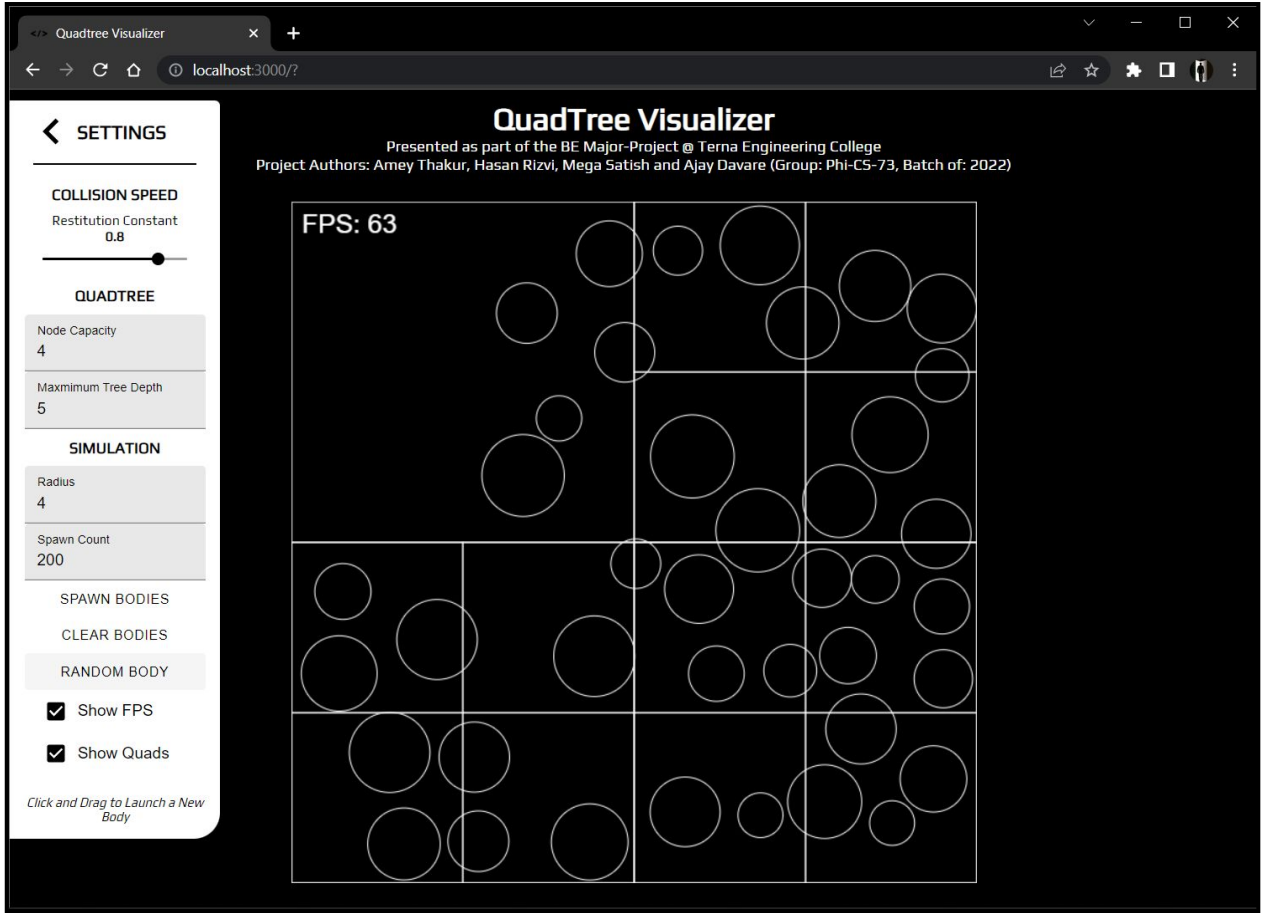
Clear QuadTree

SNAPSHOTS



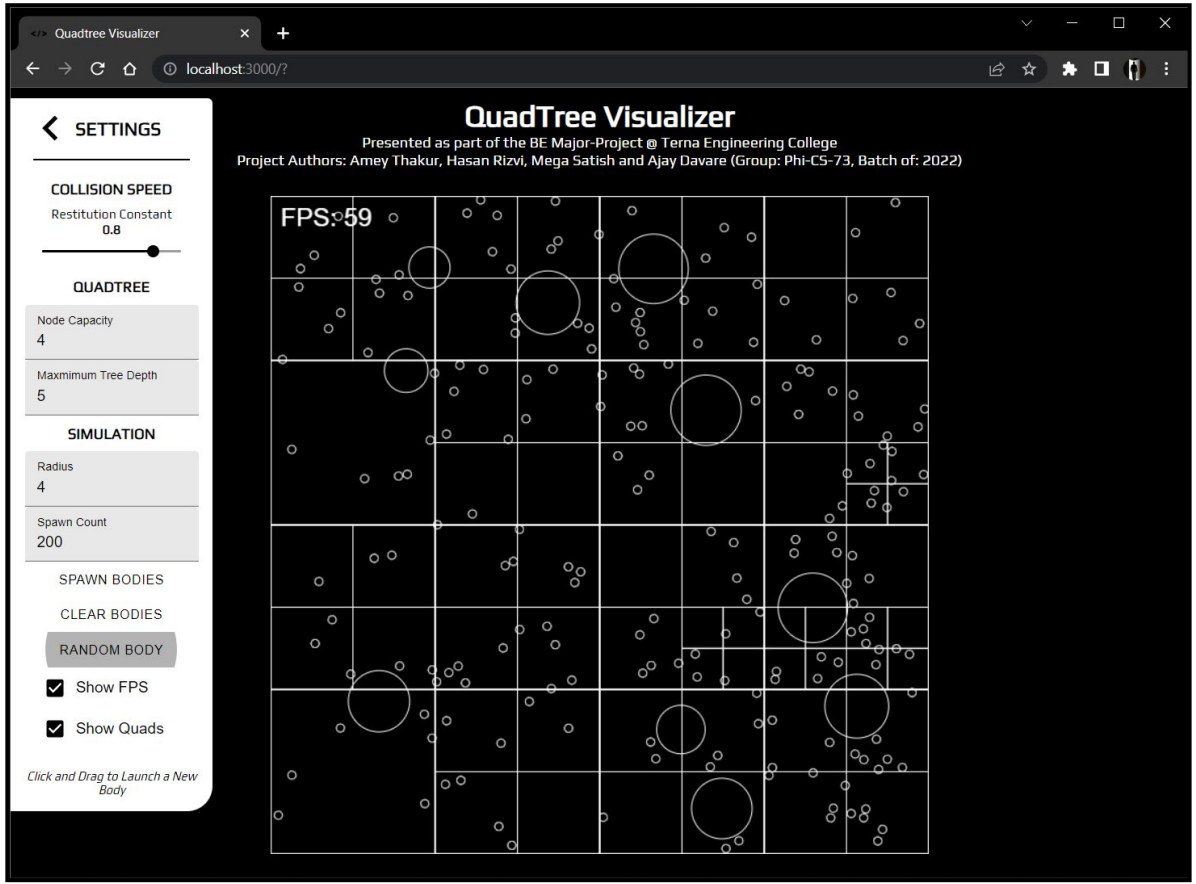
Spawn Bodies

SNAPSHOTS



Random Bodies

SNAPSHOTS



Spawn & Random Bodies

PROJECT PLAN

QuadTree Visualizer

By Amey Thakur, Hasan Rizvi, Mega Satish & Ajay Davare

Project Start: Wed, 9-15-2021

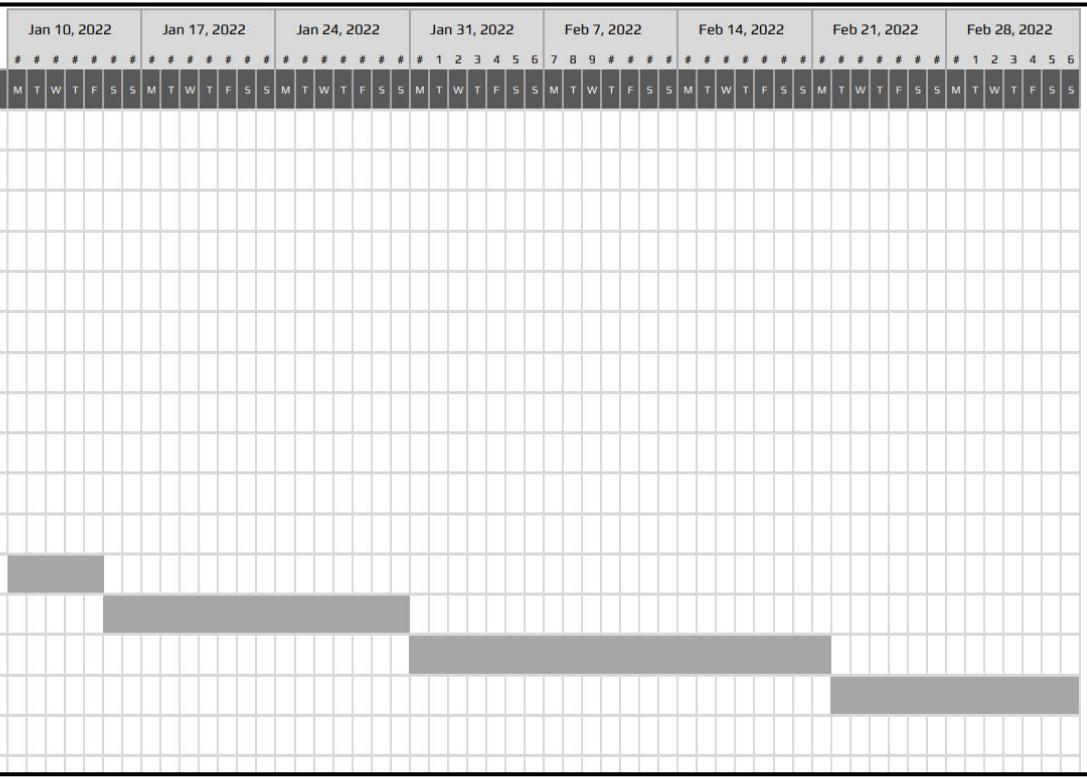
Display Week: 22

TASK	PROGRESS	START	END
Research Paper			
Conceptual understanding	100%	9-15-21	9-30-21
Literature Survey	100%	10-1-21	10-22-21
High-level analysis of algorithm	100%	10-23-21	11-12-21
Problem Statement			
Comprehensive Study of Core Concepts and Terminologies	100%	11-13-21	11-28-21
Design			
Define Wokflow	100%	11-29-21	12-13-21
Design Architecture	100%	12-14-21	12-28-21
Selection of Necessary Tools and Technologies	100%	12-15-21	12-29-21
Implementation			
Selection of Server Hosting	100%	12-30-21	1-14-22
Server Configuration	100%	1-15-22	1-30-22
Deployment of Core Module	100%	1-31-22	2-21-22
Deployment of Supporting Modules	100%	2-22-22	3-9-22
Final Deployment	100%	3-10-22	3-25-22

GANTT CHART

Display Week: 18

TASK	PROGRESS	START	END
Research Paper			
Conceptual understanding	100%	9-15-21	9-30-21
Literature Survey	100%	10-1-21	10-22-21
High-level analysis of algorithm	100%	10-23-21	11-12-21
Problem Statement			
Comprehensive Study of Core Concepts and Terminologies	100%	11-13-21	11-28-21
Design			
Define Wokflow	100%	11-29-21	12-13-21
Design Architecture	100%	12-14-21	12-28-21
Selection of Necessary Tools and Technologies	100%	12-15-21	12-29-21
Implementation			
Selection of Server Hosting	100%	12-30-21	1-14-22
Server Configuration	100%	1-15-22	1-30-22
Deployment of Core Module	100%	1-31-22	2-21-22
Deployment of Supporting Modules	100%	2-22-22	3-9-22
Final Deployment	100%	3-10-22	3-25-22



EXPECTED OUTCOMES

- A generic Quadtree Visualizer Web App, which can be used by anyone to visualize a quadtree.

- The application should be easy to use and the visualisation must be simple yet functional.

- Users should be able to manipulate the quadtree by adding, deleting the data points.

CONCLUSION

- It can be concluded quadtrees are extremely powerful data structures that are still heavily under-utilised in both the industry and community applications. By the time of completion of this project we've learned to develop scalable and reusable codebases for large projects, understood the fundamentals of API build and interaction, developed a visualization tool and understood how to function in a time-bound manner and collaborate at scale across various tasks and disciplines.

WHAT HAVE WE LEARNT SO FAR...

- To adopt the practice of pair-programming and co-ordinate in a group to develop the project.
- To develop scalable and reusable codebases for large projects.
- To understand the fundamentals of API build and interaction.
- To be acclimatised with a unique data structure like Quadtree
- To understand the scope of research and public work still needed to fully utilise the power of this data structure.
- To know how to function in a time-bound manner and collaborate at scale across various tasks and disciplines.

RESEARCH PAPER

Paper Citation

- Amey Thakur , Mega Satish , Randeep Kaur Kahlon , Hasan Rizvi, Ajay Davare, 2022, *QuadTree Visualizer*, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 11, Issue 04 (April 2022), https://www.ijert.org/quadtree-visualizer

REFERENCES

[1] Q. Cai and Y. Zhou, "A quadtree-based hierarchical clustering method for visualizing large point dataset," 2016 Sixth International Conference on Information Science and Technology (ICIST), 2016, pp. 372-375, doi: 10.1109/ICIST.2016.7483441.

[2] “An effective way to represent quadtrees” Communications of the ACM, Volume 25, Issue 12, Dec 1982 pp 905–910, doi:10.1145/358728.358741.

[3] “Optimal quadtree construction algorithms” Computer Vision, Graphics, and Image Processing, Volume 37, Issue 3, March 1987, pp 402–419, doi:10.1016/0734-189X(87)90045-4.

[4] Sullivan, Gary J., and Richard L. Baker. "Efficient quadtree coding of images and video." IEEE Transactions on image processing 3, no. 3 (1994): 327-331

[5] Mathew, Reji, and David S. Taubman. "Quad-tree motion modeling with leaf merging." IEEE Transactions on Circuits and Systems for Video Technology 20, no. 10 (2010): 1331-1345.

[6] Tilkov, Stefan, and Steve Vinoski. "Node. js: Using JavaScript to build high-performance network programs." IEEE Internet Computing 14, no. 6 (2010): 80-83.

[7] Fenton, Steve, Fenton, and Spearing. “Pro TypeScript.” Apress, 2014

[8] Cantelon, Mike, Marc Harter, T. J. Holowaychuk, and Nathan Rajlich. “Node. js in action.” Greenwich: Manning, 2014.

View publication stats