**Project Report**

**Group members:**

**20k-1743**

**20k-0225**

**Abstract:**

Today we live in the age of science and technology where ever we go we need a knowledge of technology. Computer technology serves as a milestone and sorting algorithms play a vital role. We have a lot of real-life examples like the sports scores are quickly organized by a quick sort algorithm in real time. The contact list in our mobile which you can directly access to find your desired contact since data is arranged in this manner Moreover while shopping you can sort items according to your choice. and Nowadays sorting algorithms are widely used in computer software. The goal of this project is to show implementation of various sorting algorithms how much time they take to sort an array and how they sort it by visualization.

**Introduction:**

The aim of the project was to create a website application in which there is a list of sorting algorithms and a data file that is used to sort the numbers or the data in the file. Our project is about the implementation of sorting algorithms which includes :

* Insertion sort
* Bubble sort
* Merge Sort
* Heap Sort
* Quick sort
* Radix Sort
* Bucket Sort
* Courting sort
* 7.4.5. from book
* 8.2.4. from book

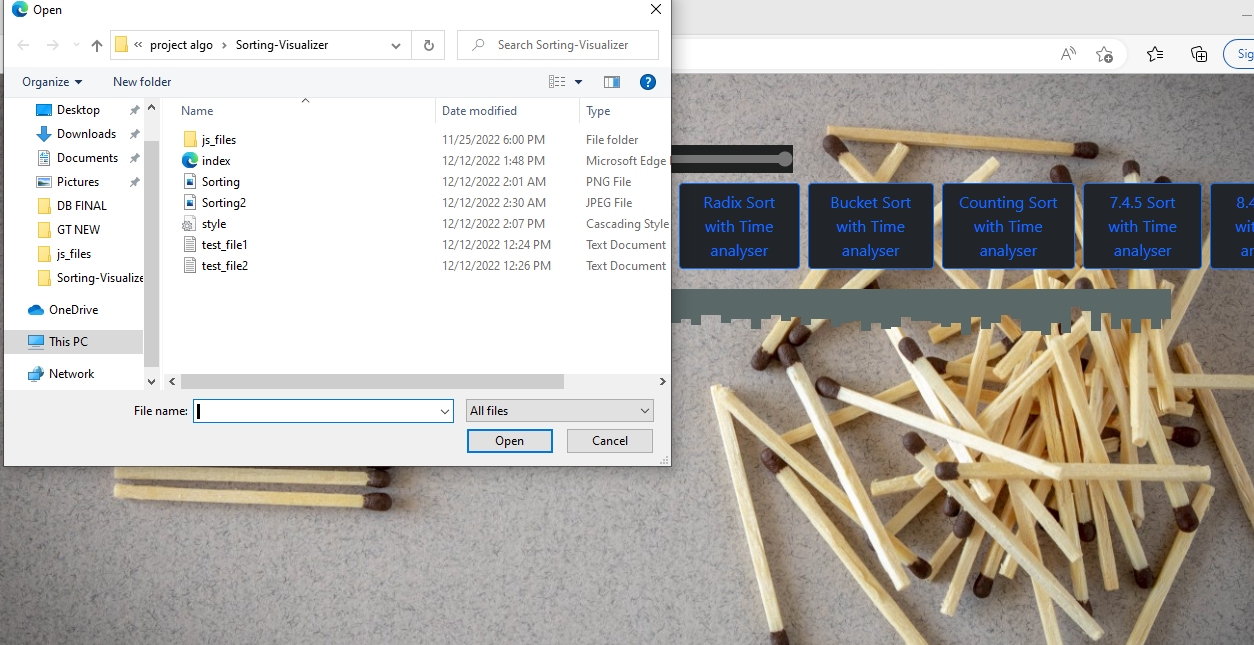
The visualization shows how each algorithm sorts an array and the bars of the graph act as elements of the file and the position of elements are being sorted and how much time it takes to sort by each algorithm.

**Programming Design:**

In our projects, we saw bars as our numbers for visualization of array elements. As the algorithm continues working, the bar’s position changes showing how the array elements are changing. We used HTML and CSS services to make an aesthetic view of the front end of the websites it includes a speed bar through which speed increase in a millisecond. We have a file function through which the file of data has been inserted and the sorting algorithms which I code using JavaScript will sort the data the bar graph act as an element to be visualized which is sorted in a user-friendly way.

**Results:**

When a txt file with generated numbers is uploaded then is stored in our array. A menu of different sorting algorithms appears asking the user which algorithm he wants to see. Once he chooses and clicks on the algorithm, the algorithm starts working and arrays start getting sorted, and the bars start changing their position to show visualization and how the array elements are getting sorted.



A picture containing text

Description automatically generated

**After sorting**:

