**Introduction:**

The main aim of this project is to create a Graphical User Interface which generates Network Graphs.

The goal is to determine the best React library which can be used for the graph generation and which can implement the below properties

**Nodes:** Create, Rename, Delete, Highlight

**Relations (Links):** Create, Rename, Delete, Add Text Label

For this purpose, we are doing research on different React libraries which can generate the graphs and evaluate their performance based on the ability to achieve the desired properties.

In this document, we share our experience of using React D3 Library.

**What is a VM image?**

A Virtual Machine (VM) is a software program or operating system that not only exhibits the behavior of a separate computer, but is also capable of performing tasks such as running applications and programs like a separate computer. A virtual machine, usually known as a guest, is created within another computing environment referred to as a "host." Multiple virtual machines can exist within a single host at one time.

For more information, please refer to below links

<https://cio-wiki.org/wiki/Virtual_Machine_(VM)>

<https://en.wikipedia.org/wiki/Virtual_machine>

**Prerequisites : VM image Installation**

**Instructions:**

Please use the below URL to download VM Image

<https://drive.google.com/file/d/1NltjVmoIvjTOh6qWmQnaCs-YoRQURnmw/view>

To import the VM image in Oracle Virtualbox, please follow the below YouTube tutorial

[How to import a .vdi file to Virtualbox?](https://www.youtube.com/watch?v=VDh9Re_s6c8)

Now, we can run the Virtual Machine with node.js, npm, and drools installed.

**Environment Setup:**

**Creating an App**

* Create React App is an officially supported way to create single-page React applications. It offers a modern build setup with no configuration.
* To create a new app, please follow the steps in the below link <https://create-react-app.dev/docs/getting-started/#creating-an-app>

**→ npx create-react-app my-app**

**→ cd my-app**

**Installing a Dependency**

* To install the dependencies, please follow the steps in the link

**→ npm install d3@^5.5.0 # if you don't have d3 already**

**→ npm install react@^16.4.1 # if you don't have react already**

**→ npm install react-d3-graph**

* For more information, please go through below links

<https://create-react-app.dev/docs/installing-a-dependency>

<https://www.geeksforgeeks.org/how-to-include-an-external-javascript-library-to-reactjs/>

* Please refer to the below documentation for finding sample code and configuring graph generation, nodes and links

<https://danielcaldas.github.io/react-d3-graph/docs/index.html>

<https://github.com/danielcaldas/react-d3-graph>

* Here a live playground page where you can interactively config your own graph, and generate a ready to use configuration

<https://codesandbox.io/s/ieuhn> (The default code in this page has JSON data which is manually entered)

**Running the project(“jay”):**

* Download the React Project named “jay” cloning below Bitbucket repository

<https://bitbucket.org/BeulahWorks/fall2022softwaredesignreactteam/src/master/>

* Now open terminal and change current directory to the directory containing project folder using below command

**→ cd jay**

* Using the below command start running the React project

**npm start jay**

**Creating Dynamic JSON: (Please refer to index.ts file in the Bitbucket repository)**

* JSON data is created dynamically using two for loops using TypeScript
* We enter the number of nodes and links in the program to create the desired graph

**Create, read, update and delete (CRUD) Operations:**

* Once the D3 graph is created, we can create any number of nodes.
* We can create new link between any nodes
* We can update the existing node names and symbol of nodes
* We can delete the existing nodes and links
* All the nodes and links are draggable
* Below are the links for D3 react graph live play ground

<https://github.com/danielcaldas/react-d3-graph>

<https://danielcaldas.github.io/react-d3-graph/sandbox/index.html>

**Performance:**

* 100 Nodes, 100 Edges Generation Time -7 seconds
* 500 Nodes, 500 Edges Generation Time- 15 seconds
* 1000 Nodes, 1000 Edges Generation Time- 30 seconds