

Graphnet - JavaScript Graph Library with WebSocket Connections

Project Description

The Graphnet project is a JavaScript graph library that is intended to be used as a module within the Semantic UI framework. It provides developers with the ability to dynamically render nodes and edges using WebSocket connections.

The project involves working on a graph library that offers Canvas layouts and Node/Edge positioning algorithms. The library provides built-in layout algorithms such as Circular and Force that can be used to automatically position the nodes and edges of a graph.

In addition, the library offers various options for rendering nodes and edges on the canvas, such as size, shape, and color properties. The library also allows manual positioning, which enables developers to override or adjust the positions of individual nodes and edges as needed.

Project Goals

The following are the primary goals of the Graphnet project:

- Develop a JavaScript graph library that can be used as a module within the Semantic UI framework.
- Create WebSocket connections to enable dynamic rendering of nodes and edges.
- Implement various node and edge positioning algorithms as built-in modules within the library.
- Enhance the Canvas layouts by adding more built-in layout algorithms that can be used to position the nodes and edges of a graph.
- Refine the existing Node/Edge positioning algorithms by implementing more advanced techniques that provide better control over the position of the nodes and edges.

- Provide clear documentation and examples of how to use the library for developers.

Implementation Details

The following are the details on how I plan to implement the Graphnet project:

- Research existing graph libraries to gain insights into best practices and determine potential areas of improvement.
- Design the architecture of the library, including the WebSocket connections and built-in modules for node and edge positioning algorithms.
- Implement the library, focusing on the WebSocket connections and built-in modules.
- Test the library to ensure that it functions as expected and meets the project goals.
- Create documentation and examples of how to use the library for developers.

Timeline

The following is an estimated timeline for the Graphnet project:

- Week 1-2: Research existing graph libraries and design the architecture of the library.
- Week 3-6: Implement the library, focusing on the WebSocket connections and built-in modules.
- Week 7-8: Test the library and make any necessary adjustments.
- Week 9-10: Create documentation and examples of how to use the library for developers.

Deliverables

The following are the deliverables for the Graphnet project:

- A JavaScript graph library that can be used as a module within the Semantic UI framework.
- WebSocket connections to enable dynamic rendering of nodes and edges.
- Built-in modules for various node and edge positioning algorithms, including force, random, circular, and others.

- Clear documentation and examples of how to use the library for developers.

Future Work

The following are potential future work areas for the Graphnet project:

- Add more positioning algorithms as built-in modules within the library.
- Optimize the library's performance to handle larger graphs efficiently.
- Explore the possibility of integrating with other popular web development frameworks.

Personal Information

Email: olalerebabatunde2000@gmail.com

My name is Olalere Babatunde and I am a final-year student studying mathematics with experience in JavaScript and web development. I have also worked on several projects that involve data visualization and user interface design.

I believe that this project will provide me with an opportunity to further develop my skills while contributing to an open-source project that will benefit the web development community.

Conclusion

The Graphnet project is a JavaScript graph library that provides developers with the ability to dynamically render nodes and edges using WebSocket connections. By implementing various node and edge positioning algorithms as built-in modules within the library, the project aims to provide developers with a flexible and powerful tool for creating dynamic visualizations.

I am confident that I have the necessary skills and experience to successfully complete this project. I am excited to contribute to the open-source community and look forward to working with the mentors and other contributors.