Task: Network Topology Visualization

Objective:

Create a web application using Vue.js 3 that allows users to visualize and interact with a network topology.

Requirements:

1. **Setup**:
    - Use Vue.js 3 for the application.
    - Use Vite for project setup.
    - Use TypeScript for type safety.
2. **Network Topology Visualization**:
    - Implement a component to display a network topology graph. Use a library like v-network-graph, Cytoscape.js, or any other suitable graph visualization library.
    - Nodes should represent devices in the network (e.g., computers, routers, switches).
    - Links should represent connections between devices.

1. **Interactivity**:
    - Users should be able to add, edit, and remove nodes and links.
    - Users should be able to drag nodes to reposition them.
    - Display additional information about a node or link when it is clicked (e.g., node name, type, connection details) in a Model.
2. **Component Design**:
    - Create reusable components for nodes, links, and the network canvas.
    - Use Vuex or Pinia for state management to handle the network data (nodes and links).
3. **Form Validation**:
    - Implement form validation for adding and editing nodes and links (e.g., ensure that node names are unique, links do not duplicate existing connections).
4. **Styling**:
    - Use anything you want for styling.
    - Ensure the application is responsive and looks good on different screen sizes.
5. **Documentation**:
    - Provide clear instructions on how to run the project.
    - Write a brief explanation of the design decisions and architecture of the application.

Evaluation Criteria:

- Code quality and organization.
- Proper use of Vue.js features and ecosystem (Vue Router, Pinia, etc.).
- Effective use of TypeScript.
- Implementation of interactive features.
- User experience and responsiveness.

- Clarity and maintainability of the code.
- Quality of documentation.

Instructions for the Developer:

1. **Project Setup**: Create a new Vue.js 3 project using Vue CLI or Vite.
2. **Development**: Implement the network topology visualization with the specified requirements.
3. **Submission**: Provide the complete source code along with instructions on how to run the project locally.

P.S : you can can use any mock data to create this project , no need for API’s