Use Case Template

Use Case ID	UC-01
Use Case	Identify and Display the Topology
Use case Description	The goal of this use case is to identify the topology the user has been working on and Display that topology on the UI/CLI/Terminal
Trigger	To display the identified topology:- The end user will write a command on the CLI/Terminal: "show topo"
Primary Actor	The primary actor here is the End User.
Secondary Actor	The secondary actors are: Application, Terminal, network, nodes, interfaces, etc.
Pre-Conditions	The user must've built a network topology using our TCP/IP stack implementation application. .
Normal Scenario	1. The User builds a network topology 2. The application assigns the IP and MAC addresses 3. The API's create the desired topology, with all the info 4. The user types: "show topo" 5. The application dumps the currently made Topology by the user, with all the information

Extension points	5a. In step 5, if there has been no topology built by the user: 　1.　Then the system will prompt an error of "No Topology exists" 　2.　The user returns to the basic step and builds a topology from scratch 　3.　Use case resumes on step 5 again
Alternate Flow	4a. At step 4, if the user wants to edit the topology he/she built: 　1.　They're sent to step 1 and let them make changes 　2.　Then all the steps are followed from the starting 　3.　All the changes are incorporated 　4.　New edited Topology is displayed 1a. The user can exit the application.
Post Conditions	<u>Success end condition</u> The build Topology is identified and displayed. <u>Failure end condition:</u> End-user can't build the desired topology, so no topology is displayed And the further steps couldn't follow.

Special Requirements

1. The supported Operating system to run the application is UNIX/LINUX-based systems.

2. The System should have GTK-3 installed

3. To understand the making of topology, the user should know Graph Modelling

4. The user must know the explicit commands of our exclusive CLI to operate the application.