Client	Server Management System
User	User
Functional requirements	Req 1: Server Management Req RF 1.1: The system shall allow users to add servers to the simulated environment. Req RF 1.2: The system shall allow users to remove servers from the simulated environment. Req 2: Connection Management Req RF 2.1: The system shall allow users to establish connections between servers. Req RF 2.2: The system shall allow users to remove connections between servers. Req 3: Connection Speed Req RF 3.1: The system shall allow user to find the fastest route to send data between to servers.
Problem context	In 2023 there's been a dire need for services that enable companies to quickly know the speeds and routes their servers provide for better use of their immediate resources. Rather than having to manually check every server they would rather set them up in a simulated environment with their respective connection information to quickly learn the fastest route between servers for stability and efficiency. For this problem our company aims to develop a simulated environment that mimics the respective server connections for companies to input their networks to and micromanage their resources. The system needs a javafx interface with an always showing diagram of all servers and server connections. Must have a add

	server button which includes name of server, and IP address. Must have a add connection button which links servers through the IP's. The server diagram must be always adjusting to changes. A server can't connect to more than 8 adjacent servers.
Non-functional requirements	Req NF 1: The system shall have a JavaFX interface. Req NF 2: User Interface Req RF 2.1: The interface shall display a diagram showing all servers and their connections. Req RF 2.1.1: Diagram should be updated with every addition or removal of both servers and connections.

Name or identifier	R1.1		
Abstract	The system shall allow users to add servers to the simulated environment.		
Inputs	Input name	Data type	Condition of select or repetition
	serverName	String	Can’t be null.
General activities needed to obtain the results	1. Click add server button 2. Insert inputs 3. Click Confirm		
Result or postcondition	Creates a new server and is shown in diagram		
Outputs	Output name	Data type	Condition of select or repetition
	MsgConfirmation	String	On successful addition

Name or identifier	R1.2		
Abstract	The system shall allow users to remove servers from the simulated environment.		
Inputs	Input name	Data type	Condition of select or repetition
	serverName	String	Can’t be null.
General activities needed to obtain the results	4. Click remove server button 5. Insert input 6. Click Confirm		
Result or postcondition	Removes server if server exists		
Outputs	Output name	Data type	Condition of select or repetition
	MsgConfirmation	String	On successful removal, or error on server not found.

Name or identifier	R2.1		
Abstract	The system shall allow users to establish connections between servers.		
Inputs	Input name	Data type	Condition of select or repetition
	firstServerName	String	Can't be null.
	secondServerName	String	Can't be null.
General activities needed to obtain the results	7. Click add connection button 8. Insert inputs 9. Click Confirm		
Result or postcondition	Creates new connection and shows in diagram		
Outputs	Output name	Data type	Condition of select or repetition
	MsgConfirmation	String	On successful connection or error on wrong inputs.

Name or identifier	R2.2		
Abstract	The system shall allow users to remove connections between servers.		
Inputs	Input name	Data type	Condition of select or repetition
	firstServerIP	String	Can't be null.
	secondServerIP	String	Can't be null.
General activities needed to obtain the results	10. Click remove connection button 11. Insert inputs 12. Click Confirm		

Result or postcondition	Removes connection and change is shown in diagram		
Outputs	Output name	Data type	Condition of select or repetition
	MsgConfirmation	String	On successful removal or error on wrong inputs.

Name or identifier	R3.1		
Abstract	The system shall allow users to find the fastest route to send data between to servers.		
Inputs	Input name	Data type	Condition of select or repetition
	firstServerName	String	Can't be null.
	secondServerName	String	Can't be null.
	dataAmount	double	Cant be null or negative.
General activities needed to obtain the results	13. Click two servers in diagram 14. Click find route 15. Input Amount of data 16. Click confirm		
Result or postcondition	Shows fastests server routes to send that amount of data and shows amount of time it takes to send data.		
Outputs	Output name	Data type	Condition of select or repetition
	MsgConfirmation	String	Shows fastest route, and route speed of the data.