



**UNIVERSIDADE DE SÃO PAULO**  
**FACULDADE DE FILOSOFIA, CIÊNCIAS E LETRAS DE RIBEIRÃO PRETO**  
**GRADUAÇÃO EM INFORMÁTICA BIOMÉDICA**

Desenvolvimento de um protótipo de rede de um hospital  
inteligente

**Nome:** Manoela Barreto de Oliveira Reis **Nro.USP:** 8928764

Ribeirão Preto, Janeiro de 2023

## **Sumário**

<b>1. Introdução</b>	<b>3</b>
1.1. Internet of Medical Things	3
1.2. Hospital inteligente	4
<b>2. Objetivo</b>	<b>4</b>
<b>3. Material e Métodos</b>	<b>5</b>
3.1. Cisco Packet Tracer	5
3.2. Cursos sobre o Cisco Packet Tracer	6
3.3. Imagens ilustrativas	7
3.4. Aulas e livros	8
3.5. Git e Github	8
<b>4. Resultados</b>	<b>9</b>
4.1. Construindo a topologia de rede da Aula 14, parte 1	9
4.2. Criando o projeto do Hospital Inteligente	13
<b>5. Conclusão</b>	<b>31</b>
<b>6. Referências</b>	<b>32</b>
<b>ANEXO 1 - Código do Monitor de Heart Rate</b>	<b>33</b>
<b>ANEXO 2 - Código do Painel de Chamada de Pacientes</b>	<b>34</b>

## 1. Introdução

Sem dúvidas redes de computadores revolucionaram o mundo permitindo que diversos aparelhos heterogêneos e em localizações diferentes estejam conectados através de redes de diversos tamanhos, dois exemplos disso seriam uma rede de um escritório que conecta os computadores e outros aparelhos localmente e a própria internet, que interliga inúmeros dispositivos por todo o mundo.

Esse trabalho foi inspirado no artigo de Lorenzo Tomé (2022), que trata da aplicação de IoT para a área médica, o que é chamado de *Internet of Medical Things* (IoMT), uma importante aplicação dos conhecimentos de redes de computadores e de informática biomédica como um todo. A IoMT será detalhada na sequência.

### 1.1. Internet of Medical Things

A IoT, ou Internet das coisas, se refere aos objetos no mundo físico que podem se conectar à internet e possuem capacidade de computação. Objetos do dia-a-dia que se conectam à rede são cada vez mais comuns, exemplos que podem ser citados são: Lâmpadas inteligentes, televisores inteligentes, entre outros. O assunto está cada vez mais em alta nas pesquisas científicas pois ainda existem muitos desafios a serem enfrentados. Um deles é o advento da tecnologia 5G e como adaptar os objetos para esse novo paradigma (ROSE, ELDRIDGE e CHAPIN, 2015).

Seguindo essa tendência, o conceito de IoT começou a ser utilizado para auxílio na medicina, principalmente, interconectando objetos com o objetivo de aprimorar o monitoramento de pacientes promovendo mais qualidade no cuidado. Esse conceito novo é chamado de IoMT (*Internet of Medical Things*), traduzindo para o português, Internet das Coisas Médicas. Plataformas de IoMT são principalmente compostas por sensores e circuitos eletrônicos que obtêm sinais biomédicos do paciente, um aparelho de rede para transmitir os dados pela rede, uma unidade de armazenamento e uma plataforma de visualização com esquemas de inteligência artificial para tomar decisões (VISHNU, RAMSON e JEGAN, 2020).

Essas tecnologias são utilizadas para capturar os dados de pacientes e um exemplo de uso delas é para ler os eletrocardiogramas de pacientes utilizando sensores, como um pulsômetro e um pedômetro. Um segundo uso possível é para obter sinais neurológicos, onde são utilizados sensores de EEG (eletroencefalograma), sensores galvânicos de resposta da pele, relógios de pulso com multi-sensores integrados etc. Um terceiro caso é o acompanhamento de pacientes diabéticos, um exemplo é o uso de um sistema de monitoramento de glicemia baseado em Bluetooth, que está conectado a um smartphone com um aplicativo capaz de mostrar os valores medidos. Um último exemplo seria o monitoramento de quedas de pacientes idosos utilizando giroscópios, sensores de aceleração, de vibração e câmeras (VISHNU, RAMSON, e JEGAN, 2020).

Diante disso pode se afirmar que esta temática é relevante para estudo, e o impacto positivo na saúde é notório.

## 1.2. Hospital inteligente

Hospitais inteligentes, ou smart hospitals, são hospitais onde o tratamento aos pacientes é auxiliado por tecnologias de IoMT, softwares de gestão avançados e equipamentos que utilizam inteligência artificial. Várias aplicações foram criadas para hospitais inteligentes, como a telemedicina, que proporciona tratamento clínico remoto, aparelhos de cirurgia robótica, que realizam cirurgias com orientações de um médico remotamente ou presencialmente, entre outras (RAZDAN e SHARMA, 2022). O termo “Hospital Inteligente” ainda é novo na literatura, mas autores já afirmam que o foco desses novos ambientes estarão na resolução das dores individuais dos pacientes de maneira personalizada, aumentando assim a qualidade e a eficiência do serviço, impactando diretamente na saúde dos usuários desses hospitais (ACETO et al. 2020).

## 2. Objetivo

Desenvolver e realizar simulações diversas em uma rede de computadores criada para um hospital inteligente no software Cisco Packet Tracer.

### 3. Material e Métodos

#### 3.1. Cisco Packet Tracer

Segundo o site oficial da empresa Cisco, o Cisco Packet Tracer, comumente conhecido como Packet Tracer apenas, consiste em:

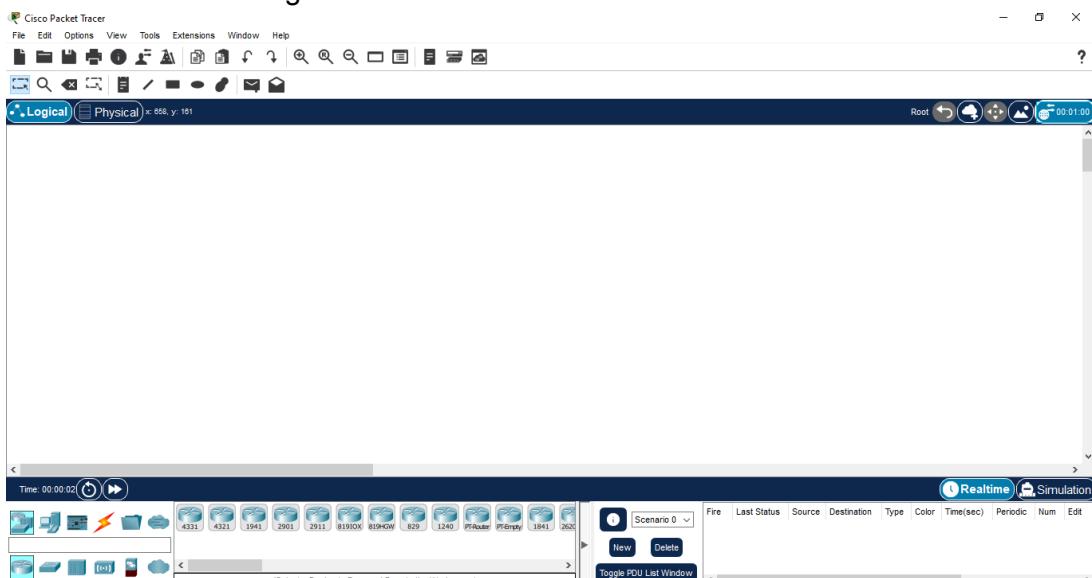
Uma ferramenta abrangente de ensino e aprendizado de tecnologia de rede, que oferece uma combinação única de experiências realistas de simulação e visualização, avaliação, recursos de criação de atividades e colaboração com vários usuários e oportunidades para competições. Os recursos inovadores do Packet Tracer ajudarão os alunos e professores a colaborar, resolver problemas e aprender conceitos em um ambiente social dinâmico e envolvente. (CISCO NETWORK ACADEMY, 2023).

Escolhi essa ferramenta por dois motivos principais:

- Ser um software que funciona também em sistemas operacionais Windows;
- Riqueza de materiais e cursos disponíveis on-line sobre a ferramenta.

A Figura 1 apresenta a tela inicial do programa em questão. A partir dessa imagem já é possível notar a grande quantidade de recursos disponíveis para realizar as mais diversas simulações de rede.

Figura 1. Tela Inicial do Cisco Packet Tracer



Fonte: Própria autoria

### 3.2. Cursos sobre o Cisco Packet Tracer

O site oficial da Cisco oferece três cursos totalmente gratuitos para aprender a utilizar a ferramenta de simulação de rede aqui empregada. O primeiro deles é intitulado “Começando com o Cisco Packet Tracer”, e ele tem por objetivo apresentar conceitos gerais sobre o software. Entre os conhecimentos aprendidos, pode-se citar os tipos de cabeamento disponíveis, assim como os diferentes modos de simulação possíveis (Realtime e Simulation).

Já o segundo curso intitula-se “Exploring Networking with Cisco Packet Tracer”, que em uma tradução livre significa “Explorando redes com o Cisco Packet Tracer”. O mesmo ensinou como simular redes sem fio dentro de um ambiente um pouco mais complexo, assim como aprendemos a simular a transmissão de pacotes entre dispositivos da rede. Por fim, conheci o controlador de rede, um dispositivo que permite monitorar, gerenciar e configurar uma rede.

O último da série ensina a montar redes de casas inteligentes utilizando tecnologias de IoT. Criou-se um novo aparelho de IoT e foram realizadas diversas simulações com sensores, objetos e dispositivos para gerenciar toda a rede criada. Ao concluir o terceiro curso, a Cisco disponibiliza uma avaliação sobre todo o aprendizado até o momento, e obtive nota suficiente nessa avaliação para receber uma certificação oficial do software. A Figura 2 apresenta o certificado obtido.

Figura 2. Certificação obtida pela Cisco Networking Academy.

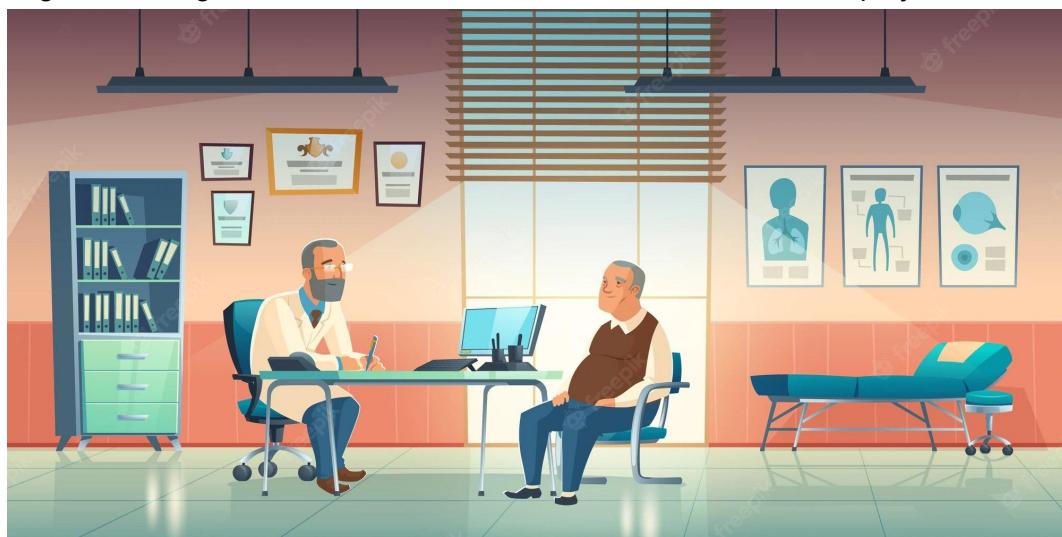


Fonte: [https://www.credly.com/badges/9c2f1875-5e68-450b-80c9-25dea29c9b18/public\\_url](https://www.credly.com/badges/9c2f1875-5e68-450b-80c9-25dea29c9b18/public_url)

### 3.3. Imagens ilustrativas

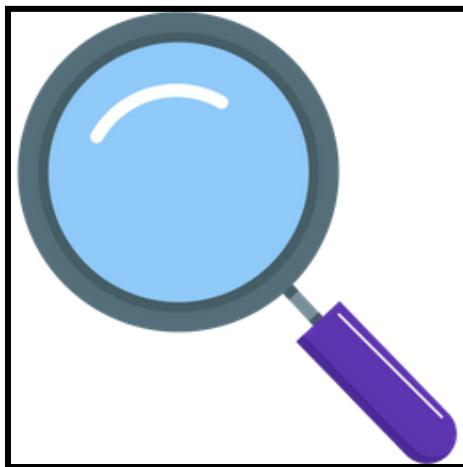
No projeto também foram utilizadas diversas imagens e ícones que ajudam a entender o propósito de determinada clusterização ou de determinado dispositivo dentro da rede. Para encontrar tais imagens foram utilizados dois sites especializados, o Freepik.com (2023), que é um site onde as imagens podem ser usadas sem que haja preocupação com os direitos das mesmas, e o Iconscout.com que é um site que o Github for Students recomenda para a procura e download de ícones, igualmente sem preocupação com direitos. As imagens as quais não encontrei disponíveis nesses sites, usei no projeto mesmo que possuam marca d'água do site a que pertencem. As Figuras 3 e 4 representam duas dessas imagens utilizadas no projeto.

Figura 3. Imagem ilustrativa utilizada no cluster “Consultório” do projeto.



Fonte: freepik.com, 2023

Figura 3. Ícone ilustrativo utilizado no projeto.



Fonte: <https://iconscout.com/icons/search>, 2023

### 3.4. Aulas e livros

Para esse trabalho foi utilizado o conteúdo das aulas da disciplina 5954019 - Redes de computadores lecionada pelo Prof. Cléver Ricardo Guareis de Farias. Também foram consultados os livros Computer Networking: A Top-Down Approach, 7th Edition (KUROSE e ROSS, 2017) e Redes de Computadores, 5a Edição (TANENBAUM e WETHERALL, 2011) para apoiar a autora na revisão de conceitos de redes.

### 3.5. Git e Github

Git é um sistema de controle de versão de código aberto que permite que os desenvolvedores rastreiem as alterações feitas em um projeto e colaborem uns com os outros. Ele permite que você volte facilmente para uma versão anterior do código, faça ramificações para trabalhar em recursos separadamente e mescle essas ramificações de volta para o código principal (GIT, 2023).

Github é uma plataforma baseada na web que usa o Git e fornece recursos adicionais para gerenciar projetos de código aberto e colaboração. Ele permite que os desenvolvedores armazenem e gerenciem seus projetos em repositórios públicos ou privados, rastreiem problemas, gerenciem as atribuições de tarefas e colaborem uns com os outros através de pull requests. Ele também fornece ferramentas para que os desenvolvedores possam visualizar e analisar o código, incluindo históricos de commits, gráficos de ramificações e estatísticas de cobertura de código.

(GITHUB, 2023).

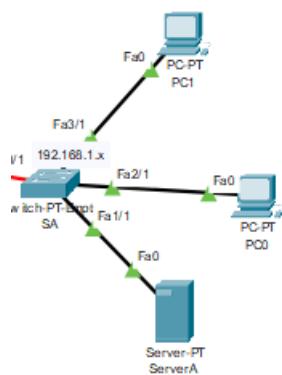
Com o objetivo de versionar o projeto para que caso ocorra algum erro e seja necessário voltar para a versão anterior, foi utilizado o Github e comandos Git para que esse versionamento fosse possível. O uso da versão desktop do Github foi escolhido para o projeto.

## 4. Resultados

### 4.1. Construindo a topologia de rede da Aula 14, parte 1

Após a realização dos cursos da Cisco e de todas as suas atividades, como sugestão do professor, o time iniciou o projeto arquitetando a rede proposta na aula de redes de número 14, na parte 1. Começamos fazendo as ligações cabeadas básicas entre os switches e os computadores e servidores. Esses últimos encontram-se na parte de “End Devices” dentro da UI do Packet Tracer. Escolhemos os cabos de cobre mais simples para conectar esses primeiros componentes da rede (chamados de Copper Straight Through). Conectamos nas portas do tipo FastEthernet. E então a primeira sub-rede ficou da forma como está sendo apresentada na Figura 5.

Figura 5. Primeira parte da rede da aula 14 parte 1.

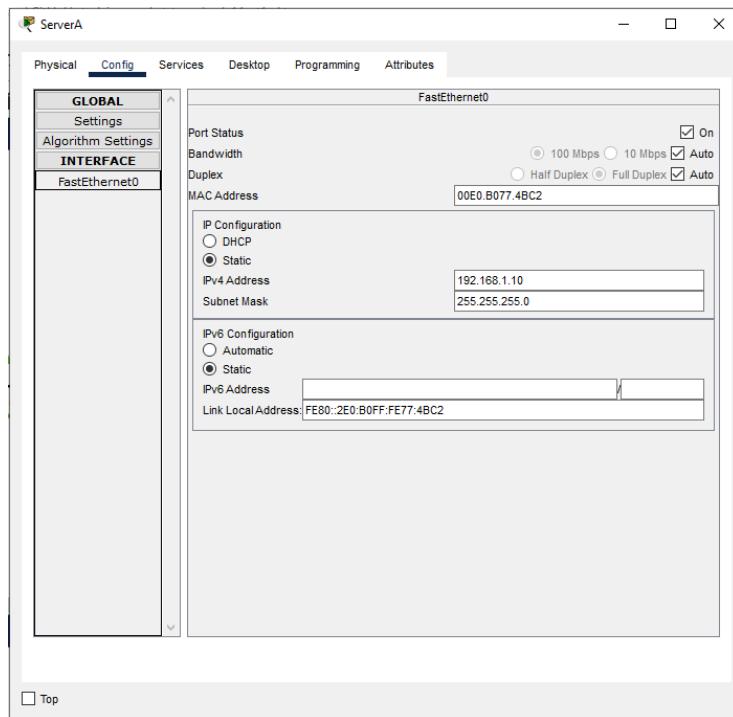


Fonte: Própria autoria

Como é possível notar na Figura 5, utilizou-se a faixa de endereços com o prefixo 192.168.1, sendo a última parte dos endereços, ou seja, a parte dos hosts sendo preenchida com qualquer valor dentro do range permitido. A máscara de

sub-rede é preenchida automaticamente e não a alteramos, permaneceu como 255.255.255.0. A princípio configuramos cada um dos hosts manualmente, de maneira estática. Para isso, acessamos cada dispositivo e navegamos até as configurações referentes à porta a qual está sendo feita a conexão até o switch. A Figura 6 mostra a tela que se apresenta quando acessamos o menu de configurações do host chamado ServerA.

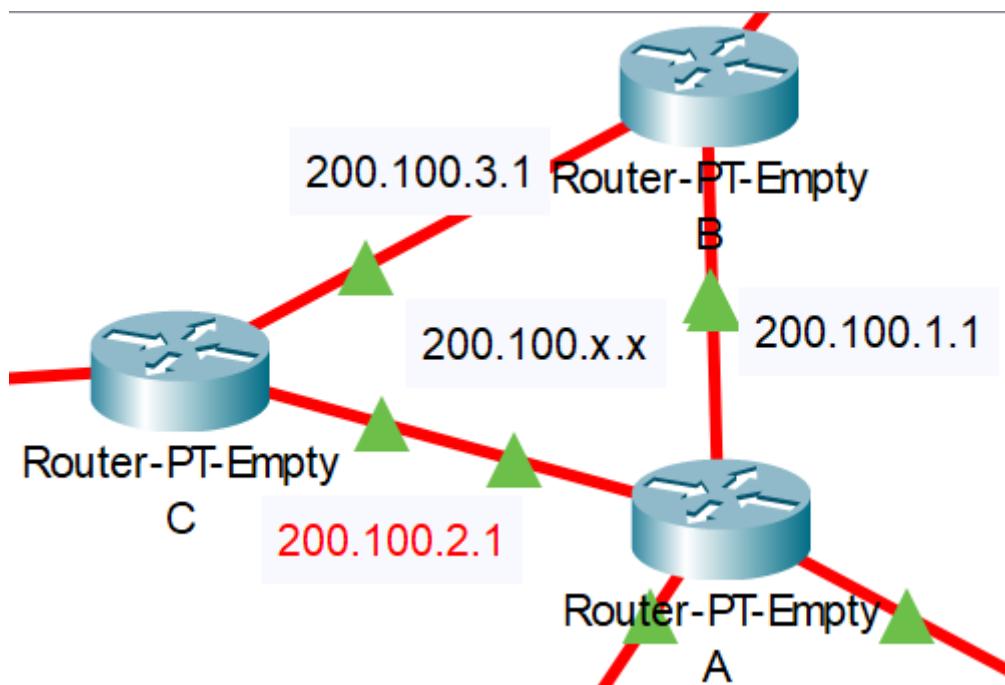
Figura 6. Tela de configurações do ServerA.



Fonte: Própria autoria

Após isso, foram adicionados roteadores e o restante da rede foi tomando forma. Para os roteadores foram instaladas 3 entradas para cada um deles, exatamente como é na aula 14, sendo que para aumentar a dificuldade preferi adicionar cabos diferentes, portanto, utilizei cabos de fibra do tipo Gigabit Ethernet entre os roteadores e para os switches, esses cabos são representados pela cor laranja. Entre os roteadores resolvi mudar o endereçamento IP pois as interfaces deles com os switches deveriam ser as mesmas das sub-redes onde estariam os PCs e Servidores, então caso eu permanecesse com o mesmo prefixo (192.168.x.x), haveria uma sobreposição. Então entre os roteadores adotei o prefixo 200.100.x.x. A Figura 7 mostra as interligações realizadas.

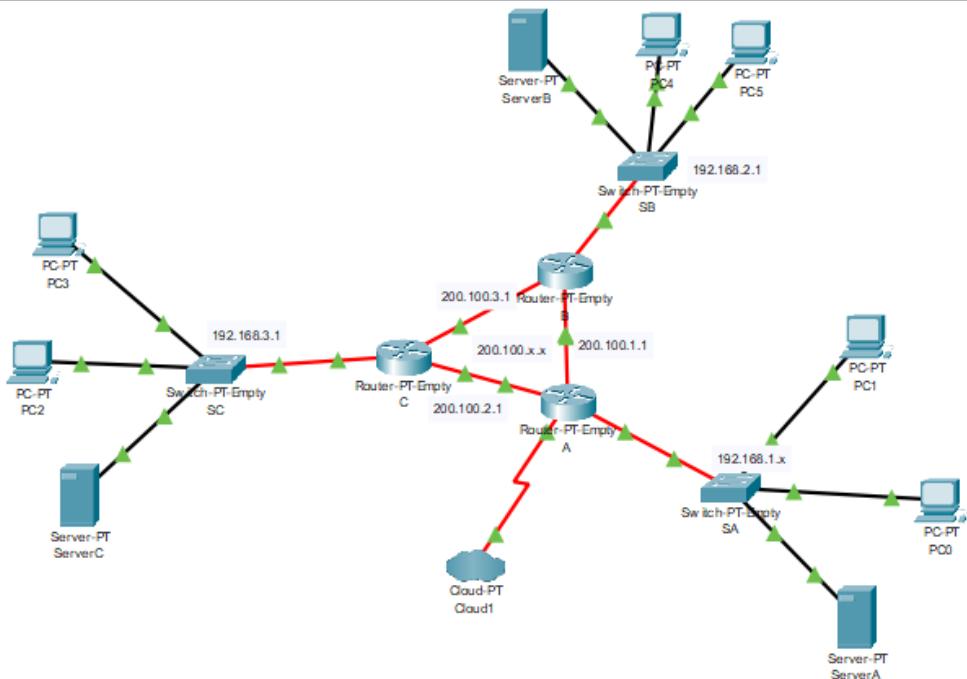
Figura 7. Interligações dos roteadores centrais.



Fonte: Própria autoria

Assim, após todas as configurações de todas as interfaces dos roteadores nomeados A, B e C, finalizei a rede como se mostra na Figura 8.

Figura 8. Reprodução da topologia da rede da aula 14 parte 1.

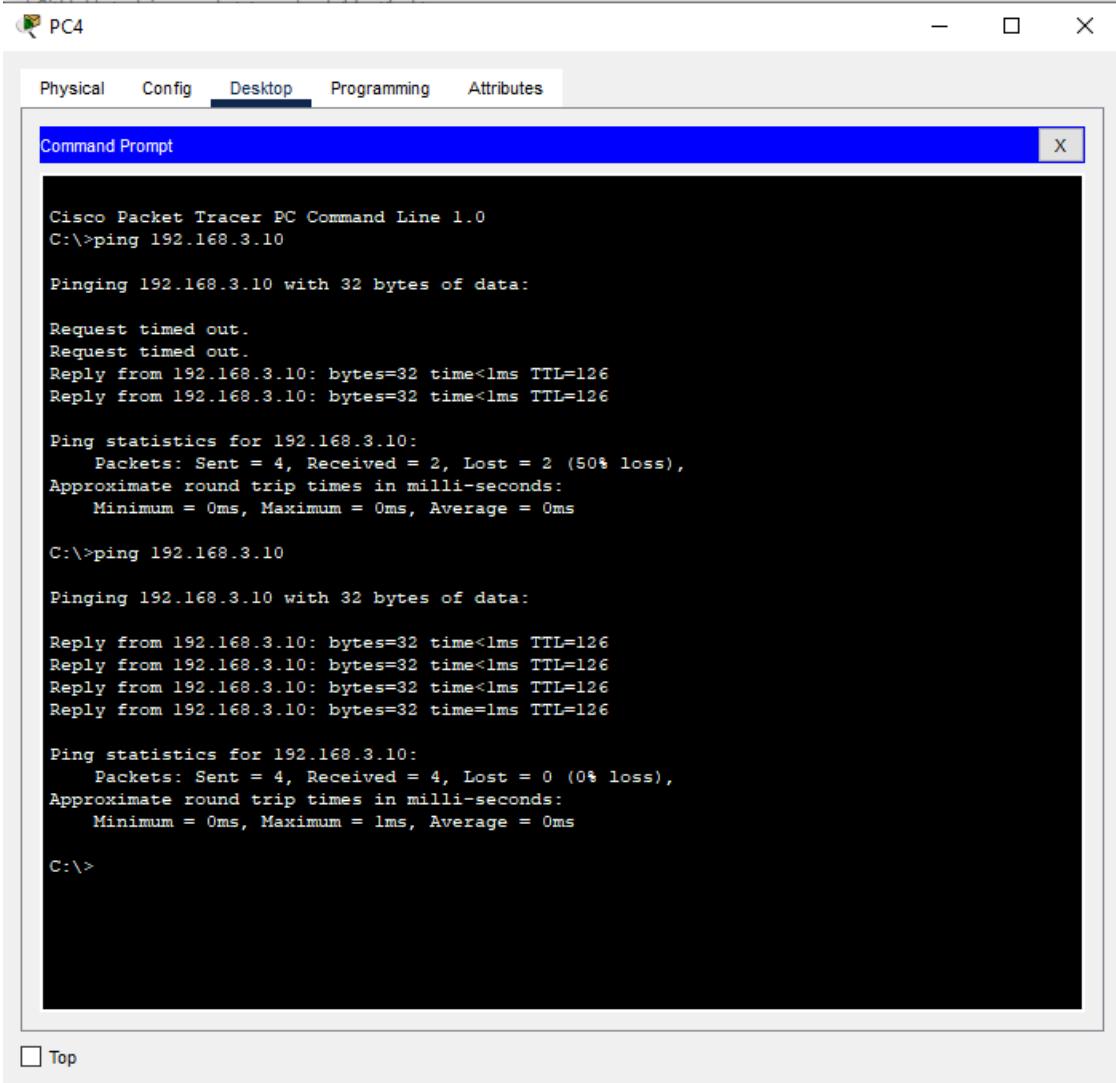


Fonte: Própria autoria

Usei o protocolo RIP (Routing Information Protocol) em cada roteador para adicionar os endereços das redes que suas respectivas interfaces estavam conectadas nas tabelas RIP de todos.

Posteriormente iniciei os testes de envio de pacotes entre dispositivos usando o comando ping (Packet Internet Groper) que se utiliza do Internet Control Message Protocol (ICMP) para obter uma resposta do ICMP de um host ou gateway. Segundo a IBM (2021), esse comando é importante para determinar o status da rede e de diversos hosts estrangeiros; rastrear e isolar problemas de hardware e software e testar, medir e gerenciar redes. A figura 9 mostra 2 tentativas de uso do ping do PC4 (192.168.2.2) para o ServerC (192.168.3.10) em duas redes diferentes.

Figura 9. Ping da origem 192.168.2.2 até o destino 192.168.3.10.



The screenshot shows a Windows-style window titled "Command Prompt" running on a virtual machine named "PC4". The window contains the following command-line session:

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.3.10

Pinging 192.168.3.10 with 32 bytes of data:

Request timed out.
Request timed out.
Reply from 192.168.3.10: bytes=32 time<1ms TTL=126
Reply from 192.168.3.10: bytes=32 time<1ms TTL=126

Ping statistics for 192.168.3.10:
    Packets: Sent = 4, Received = 2, Lost = 2 (50% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 192.168.3.10

Pinging 192.168.3.10 with 32 bytes of data:

Reply from 192.168.3.10: bytes=32 time<1ms TTL=126
Reply from 192.168.3.10: bytes=32 time<1ms TTL=126
Reply from 192.168.3.10: bytes=32 time<1ms TTL=126
Reply from 192.168.3.10: bytes=32 time=1ms TTL=126

Ping statistics for 192.168.3.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\>
```

Fonte: Própria autoria

Por fim, aprendi como fazer uma emulação de WAN, cujo ícone é uma pequena nuvem, ligada à rede com um cabo do tipo Serial.

Depois de mais alguns testes considerei como concluída essa primeira etapa.

## 4.2. Criando o projeto do Hospital Inteligente

A partir da inspiração em um artigo voltado para médicos, que falava sobre o advento das tecnologias chamadas de IoMT, ou Internet das Coisas Médicas (TOMÉ, 2022), imaginei um hospital inteligente que contivesse vários setores que se interligam através da rede construída. Primeiramente criei o desenho de uma cidade e a chamei de cidade de Saquarema, o desenho foi feito a partir da junção de vários outros desenhos em perspectiva isométrica. A cidade construída pode ser vista na Figura 10.

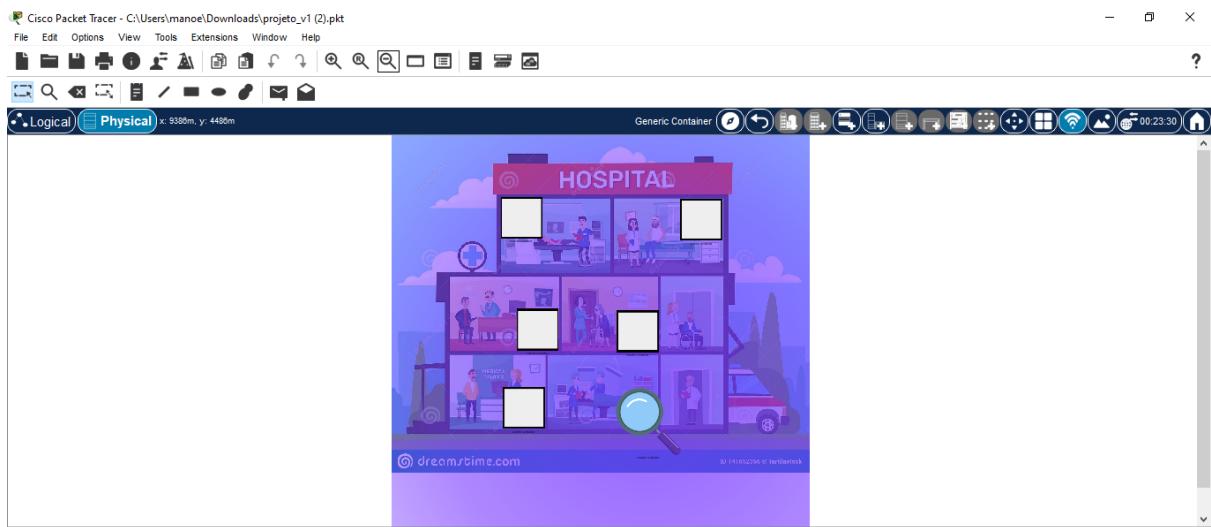
Figura 10. Cidade de Saquarema.



Fonte: Própria autoria

Dentro do software Packet Tracer é possível visualizar o projeto sob um olhar físico e sob um olhar lógico. Os desenhos me auxiliaram a entender onde cada elemento estaria. E então a organização em containers foi utilizada para separar cada parte do hospital que estaria interligada. A Figura 11 mostra o resultado desse primeiro teste e o primeiro commit do projeto foi feito.

Figura 11. Containers criados dentro do hospital da cidade criada (visão física).



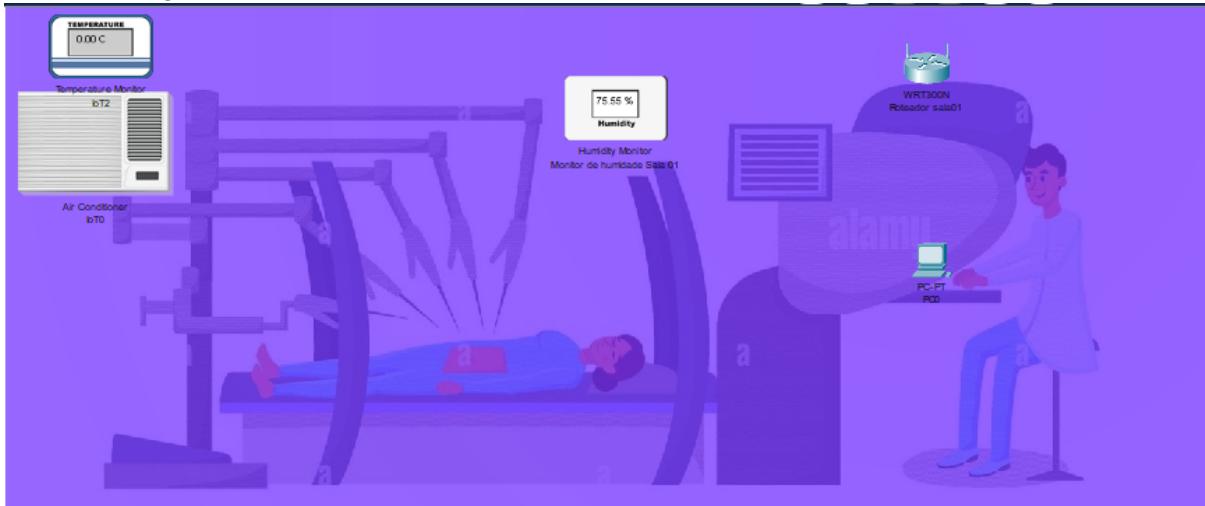
Fonte: Containers: Próprios do Packet Tracer; Imagem de background: <https://pt.dreamstime.com/salas-m%C3%A9dicas-dicas-escrit%C3%B3rio-interior-da-constru%C3%A7%C3%A3o-hospital-sala-de-espera-doutor-cl%C3%ADnica-emerg%C3%A3ncia-e-desenhos-anim-image141652356>, Acesso em: 2 jan. 2023

A primeira sala a ser construída foi a sala de cirurgia, que inicialmente eu gostaria que contivesse os seguintes elementos:

- Ar condicionado inteligente
- Sensor de temperatura
- Sensor de umidade
- Um computador para o médico
- Um roteador wireless

A abstração que gostaria que fosse feita é uma sala de cirurgia robótica com sensores e equipamentos que se conectam à internet e o médico pudesse controlar com o seu computador. A Figura 12 mostra a primeira versão da sala de cirurgia:

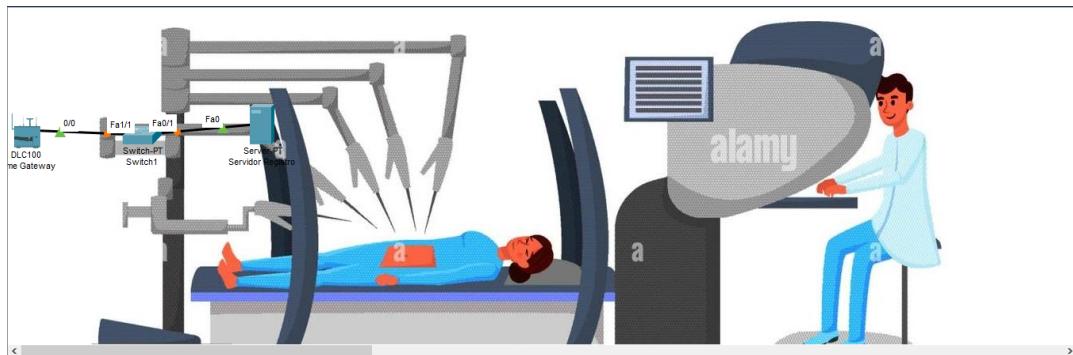
Figura 11. Containers criados dentro do hospital da cidade criada (visão física).



Fonte: Equipamentos: Próprios do Packet Tracer; Imagem de background: <https://c8.alamy.com/comp/W0T4CC/robot-assisted-surgery-flat-vector-illustration-surgeon-and-patient-under-anesthesia-cartoon-characters-young-doctor-performing-remote-surgical-procedure-with-high-tech-equipment-modern-medicine-W0T4CC.jpg>, Acesso em: 2 jan. 2023

Retornando ao curso da Cisco de “Exploring Internet of Things with Cisco Packet Tracer”, fui seguindo o passo-a-passo para que um equipamento pudesse ser o controlador dos equipamentos específicos de IoT. Os primeiros testes foram feitos com o roteador sem fio mostrado na Figura 11, mas depois de alguns testes e referenciais disponíveis online sobre o assunto, percebi que seria mais adequado utilizar um equipamento chamado Home Gateway. Esse equipamento provê acesso à internet e conectividade sem fio. Para conectar os aparelhos de IoT apenas é preciso configurar o IP dos mesmos e indicar que eles precisam se conectar ao Home Gateway. É possível também conectar os aparelhos a um Servidor Registro (Register Server), que é um servidor comum, porém, com o serviço de registro de aparelhos de IoT ativado. A Figura 12 mostra uma versão atualizada da sala de cirurgia na qual excluí todos os aparelhos de IoT e mantive o foco apenas no Home Gateway e no Servidor Registro.

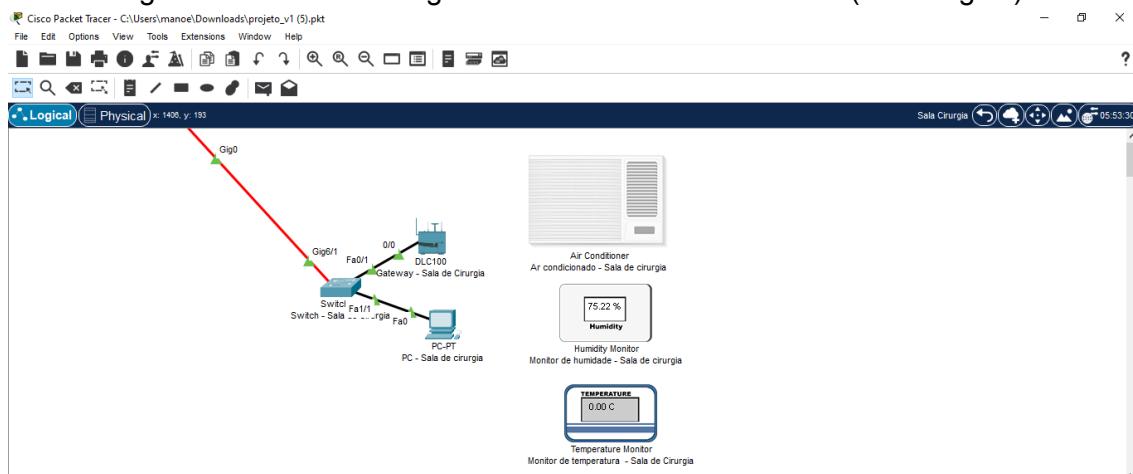
Figura 12. Home Gateway e Servidor Registro na sala de cirurgia (visão lógica).



Fonte: Equipamentos: Próprios do Packet Tracer; Imagem de background: <https://c8.alamy.com/comp/W0T4CC/robot-assisted-surgery-flat-vector-illustration-surgeon-and-patient-under-anesthesia-cartoon-characters-young-doctor-performing-remote-surgical-procedure-with-high-tech-equipment-modern-medicine-W0T4CC.jpg>, Acesso em: 2 jan. 2023

Assim que consegui entender o funcionamento de ambos equipamentos, prossegui na inserção de mais objetos na sala. Vale ressaltar que não faria sentido ter um servidor dentro de uma sala de cirurgia, então criei outro container no prédio ao lado do hospital e o chamei de administração do hospital. Nesse prédio se concentrariam todos os servidores e equipamentos estratégicos que impactam o hospital como um todo. E em específico para interligar os equipamentos da sala de cirurgia com o servidor registro que estava no prédio da administração do hospital, utilizei um cabo de fibra óptica do tipo GigaBit Ethernet, pois gostaria que fosse um enlace com uma boa velocidade de troca de dados e uma boa qualidade, além de ser possível trafegar dados em longas distâncias. Então a versão até aquele momento da sala de cirurgia foi a seguinte mostrada na Figura 13:

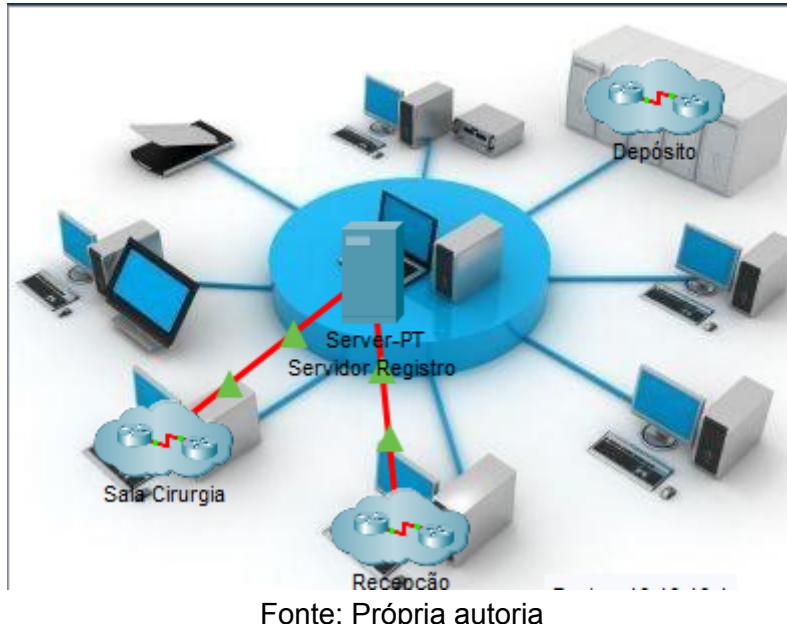
Figura 13. Sala de Cirurgia em uma das versões iniciais (visão lógica).



Fonte: Própria autoria

Depois mais um cluster foi criado. O cluster da sala de Recepção. Com a interligação entre o prédio da administração e esses dois clusters, a topologia da rede ficou da forma como a Figura 14 apresenta.

Figura 14. Servidor conectado à sala de cirurgia e à recepção (visão lógica).



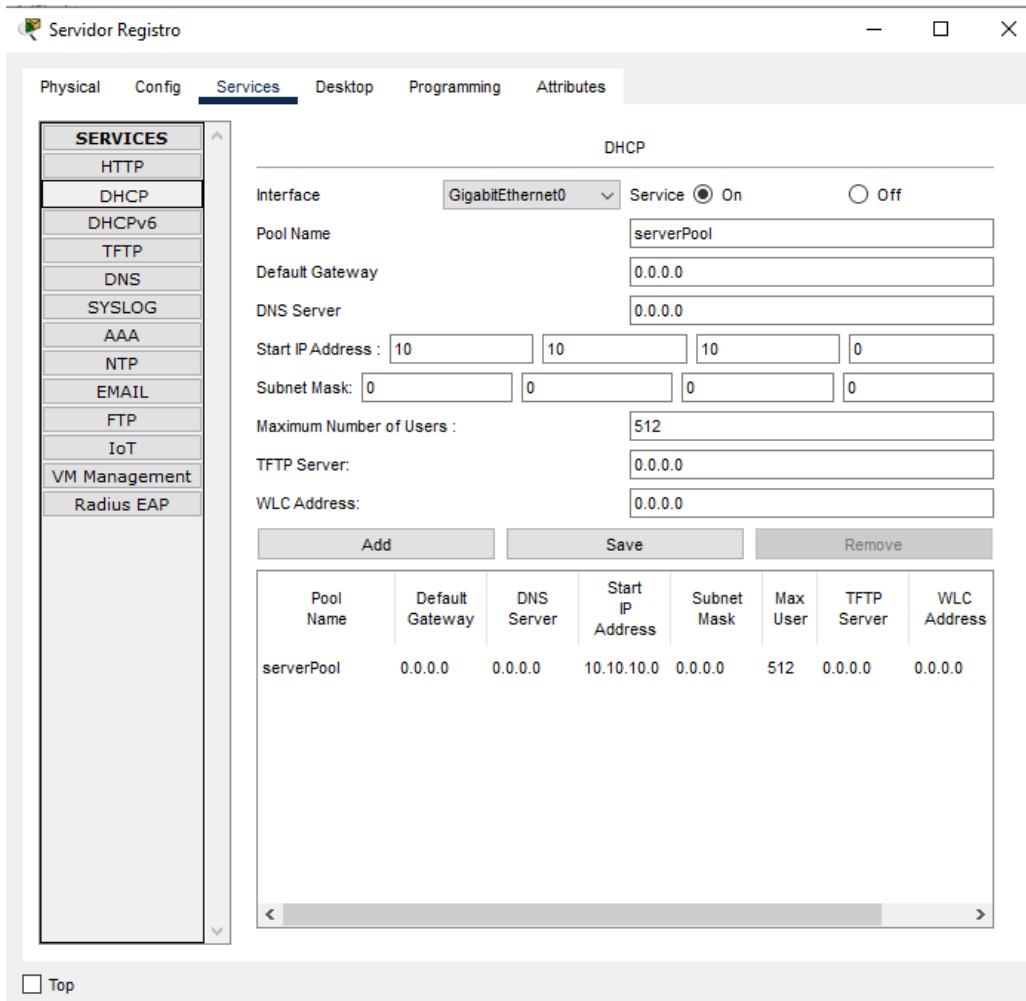
Fonte: Própria autoria

Para aumentar a complexidade, imaginei que o mesmo servidor que provê o serviço de registro para os objetos de IoT também poderia ser o mesmo que aplicasse o serviço de configuração dinâmica de IPs para equipamentos que se conectarem à rede. Para tal, foi configurado esse serviço da forma com que aparece na Figura 15. Até este ponto eu não sabia como fazer com que o servidor se comunicasse com os outros equipamentos como aprendemos em sala, com as mensagens de descoberta, oferta, requisição e reconhecimento. Então com a configuração mostrada na Figura 15, o DHCP não funcionou da maneira que eu gostaria. Os erros que identifiquei foram os seguintes:

- A máscara de sub-rede está incorreta, dando a possibilidade de configuração de todos os endereços de IP na versão 4 que existem acima da faixa de 10.10.10.0, sendo que o objetivo é apenas que o DHCP atribua endereços de IP que alterem apenas a parte do host, então a máscara deveria ser 255.255.255.0;

- O número máximo de usuários está muito grande para apenas a rede desse pequeno hospital.

Figura 15. Aba de configurações do Servidor DHCP até o momento.

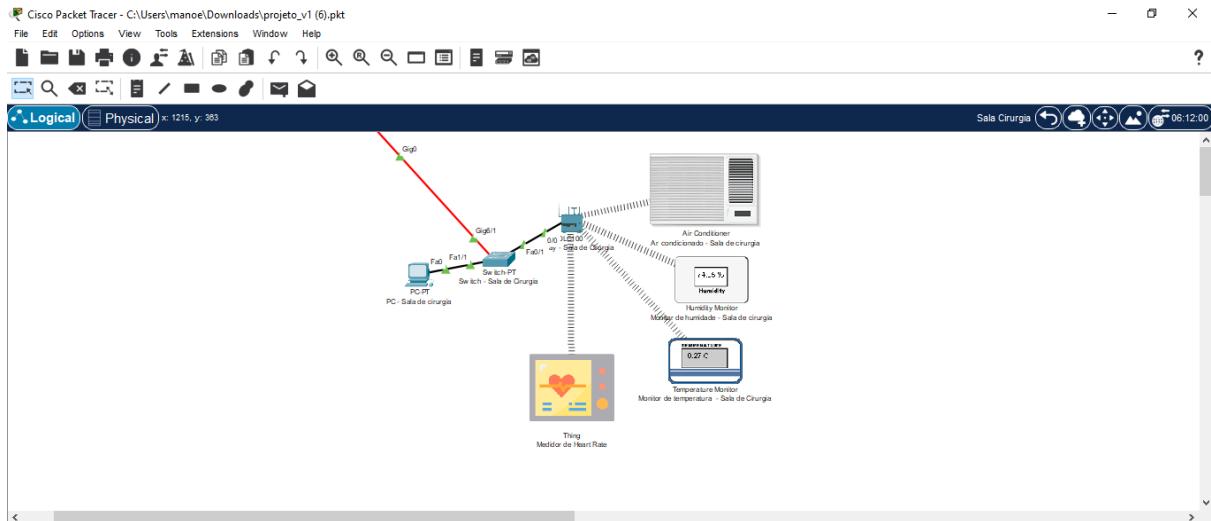


Fonte: Própria autoria

Almejando aumentar a complexidade do projeto, imaginei que seria interessante adicionar um equipamento para simular o monitor de batimentos cardíacos do paciente na sala de cirurgia. Com o Packet Tracer é possível adicionar ao projeto um equipamento chamado “Thing”, que nada mais é do que um aparelho em branco o qual o desenvolvedor pode adicionar sua própria imagem e também programar seu comportamento em Python ou em Javascript. Então criei o objeto “Medidor de Heart Rate” e o programei para piscar entre duas imagens simulando que está captando um batimento cardíaco constante do paciente. Ele também é um aparelho com uma entrada wireless e se conecta ao Home Gateway assim como os

outros equipamentos IoT da sala. Assim, foi criada mais uma versão da sala de cirurgia, a qual é possível notar na Figura 16.

Figura 16. Sala de cirurgia com o Monitor de Heart Rate criado.



Fonte: Própria autoria

Posteriormente adicionei um som para aumentar a semelhança entre a realidade e o equipamento construído. O programa simples que está rodando no Monitor de Heart Rate foi feito em Javascript e se encontra no Anexo 1 do presente trabalho.

Todos os endereços de IP de cada equipamento presente na sala de cirurgia teve seu IP configurado manualmente de maneira estática.

Iniciei os testes com o comando PING entre os objetos de IoT e o Home Gateway da sala de cirurgia e após isso, testei também se os equipamentos conseguiam enviar pacotes para o PC da mesma sala. Os resultados obtidos foram os seguintes mostrados na Figura 17:

Figura 16. Sala de cirurgia com o Monitor de Heart Rate criado.

PDU List Window	Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num
	Successful		Monitor de humidade - Sala de cirurgia	Gateway - Sala de Cirurgia	ICMP	■	0.000	N	0
	Successful		Monitor de temperatura - Sala de Cirurgia	Gateway - Sala de Cirurgia	ICMP	■	0.000	N	1
	Successful		Ar condicionado - Sala de cirurgia	Gateway - Sala de Cirurgia	ICMP	■	0.000	N	2
	Successful		Medidor de Heart Rate	Gateway - Sala de Cirurgia	ICMP	■	0.000	N	3
	Failed		Monitor de humidade - Sala de cirurgia	PC - Sala de cirurgia	ICMP	■	0.000	N	4
	Successful		Ar condicionado - Sala de cirurgia	PC - Sala de cirurgia	ICMP	■	0.000	N	5
	Successful		Monitor de temperatura - Sala de Cirurgia	PC - Sala de cirurgia	ICMP	■	0.000	N	6
	Successful		Medidor de Heart Rate	PC - Sala de cirurgia	ICMP	■	0.000	N	7

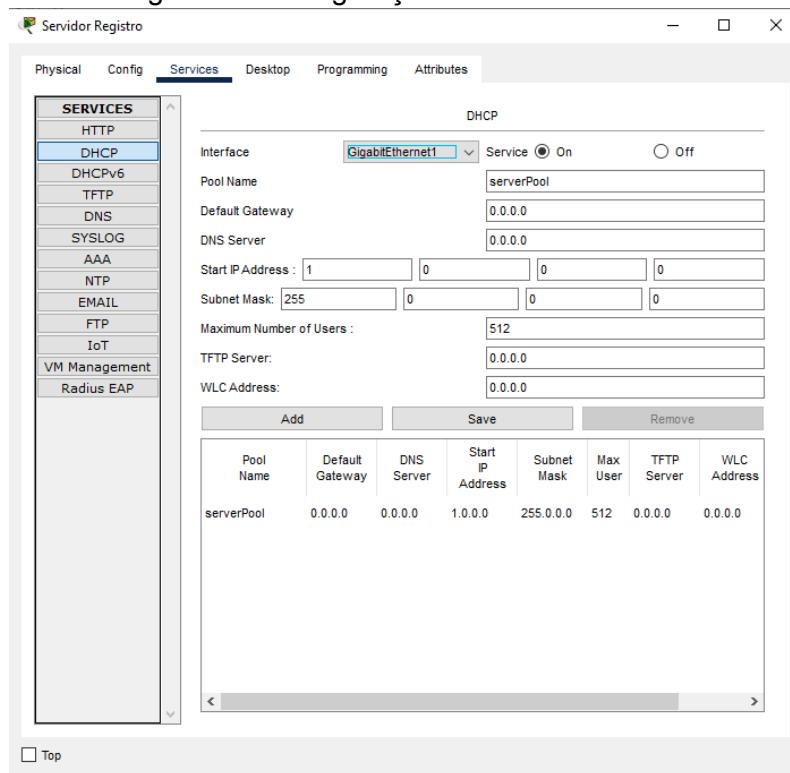
Fonte: Própria autoria

Como é possível notar, todos os pacotes tiveram sucesso, com exceção de um deles, que depois foi testado novamente e obteve sucesso.

A partir disso, concluí que essa parte da rede já possuía uma base funcional, então passei a voltar meu foco para o servidor DHCP, uma vez que planejei adicionar diversos dispositivos na rede e seria inviável no mundo real configurar cada dispositivo de maneira manual.

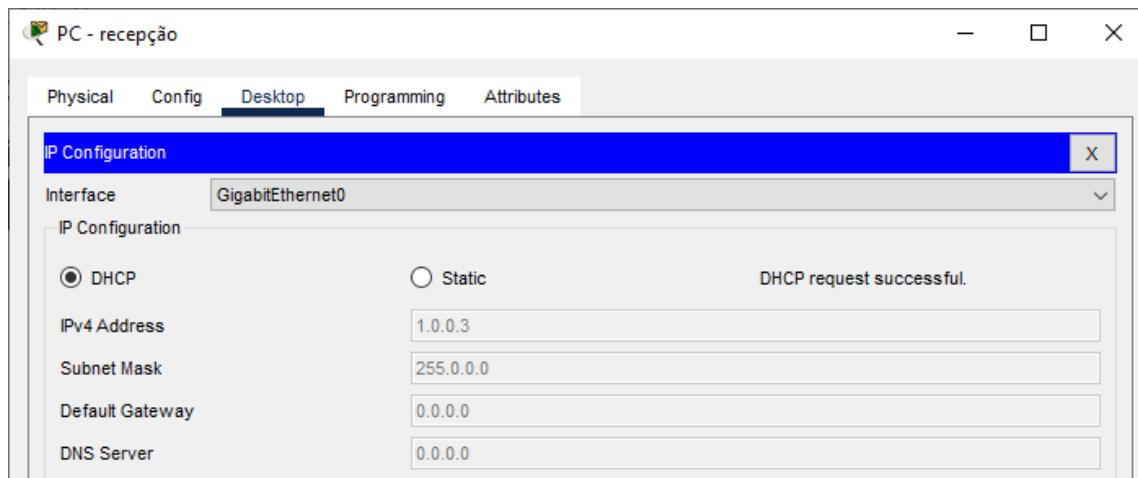
Na recepção, consegui fazer com que o PC recebesse o endereço IP dinamicamente. Utilizei uma classe de IPs que não era adequada em minha opinião. Uma vez que o endereço padrão da LAN dos Home Gateways que conectam os dispositivos de IoTs é 192.168.25.1, então os endereços estariam muito destoantes. A configuração do servidor encontra-se na Figura 17 e a Figura 18 mostra a aba de IP configuration do PC da recepção, mostrando que a requisição DHCP foi bem sucedida. Já a Figura 19 mostra a topologia inicial desenvolvida na recepção.

Figura 17. Configuração do servidor DHCP.



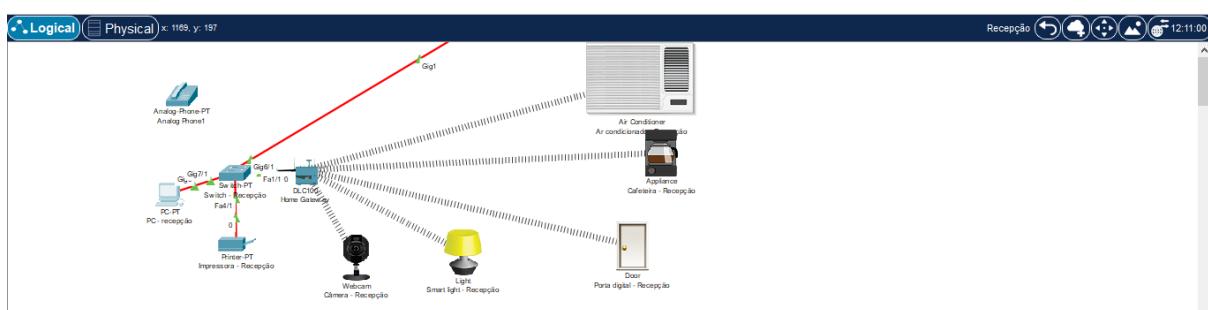
Fonte: Própria autoria

Figura 18. Aba IP Configuration do PC da recepção.



Fonte: Própria autoria

Figura 19. Topologia inicial da recepção (visão lógica).



Fonte: Própria autoria

Após finalizada essa etapa, fiz testes com o comando ping e com o comando tracert, que nada mais é do que a técnica traceroute que monitora o pacote desde a origem, mostrando por onde o pacote passa até chegar em seu destino. Um exemplo desses testes encontra-se na Figura 20. Sabendo-se que o servidor possui endereço IP 1.0.0.100 na interface conectada ao cluster da recepção e que o PC da sala de cirurgia (em outro cluster) recebeu o seguinte endereço via DHCP: 10.0.0.1.

Figura 20. Comandos PING e TRACERT partindo do PC da recepção.

The screenshot shows a Windows-style window titled "PC - recepção" with a tab bar containing "Physical", "Config", "Desktop", "Programming", and "Attributes". The "Desktop" tab is selected. Inside the window, there is a "Command Prompt" window with the following text:

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 1.0.0.100

Pinging 1.0.0.100 with 32 bytes of data:

Reply from 1.0.0.100: bytes=32 time<1ms TTL=128

Ping statistics for 1.0.0.100:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>tracert 1.0.0.100

Tracing route to 1.0.0.100 over a maximum of 30 hops:
  1  0 ms      0 ms      0 ms      1.0.0.100

Trace complete.

C:\>tracert 10.0.0.1

Tracing route to 10.0.0.1 over a maximum of 30 hops:
  1  *          *          *          Request timed out.
  2  *          *          *          Request timed out.
  3  *          *          *          Request timed out.
  4  *          *          *          Request timed out.
  5  *          *          *          Request timed out.
  6  *          *          *          Request timed out.
  7  *          *          *          Request timed out.
  8  *          *          *          Request timed out.
  9  *          *          *          Request timed out.|
```

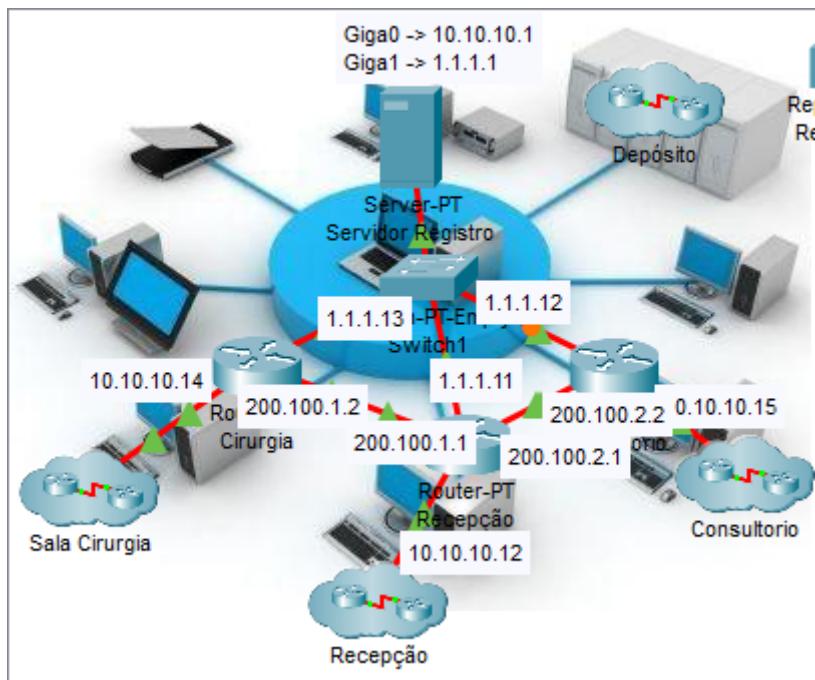
At the bottom left of the Command Prompt window, there is a checkbox labeled "Top".

Fonte: Própria autoria

Com esses resultados pude perceber que a rede estava inadequada, e que um cluster não se comunicava com outro. E também percebi que o servidor estava fazendo um papel incorreto de switch, então recomecei praticamente o projeto do zero e adicionei mais um cluster, o consultório.

Adicionei três roteadores e um switch no projeto, todos interligados com cabos de fibra óptica do tipo Gigabit Ethernet. Os endereços de IP e a topologia estão presentes na Figura 21.

Figura 21. Topologia da rede até o momento (visão lógica).

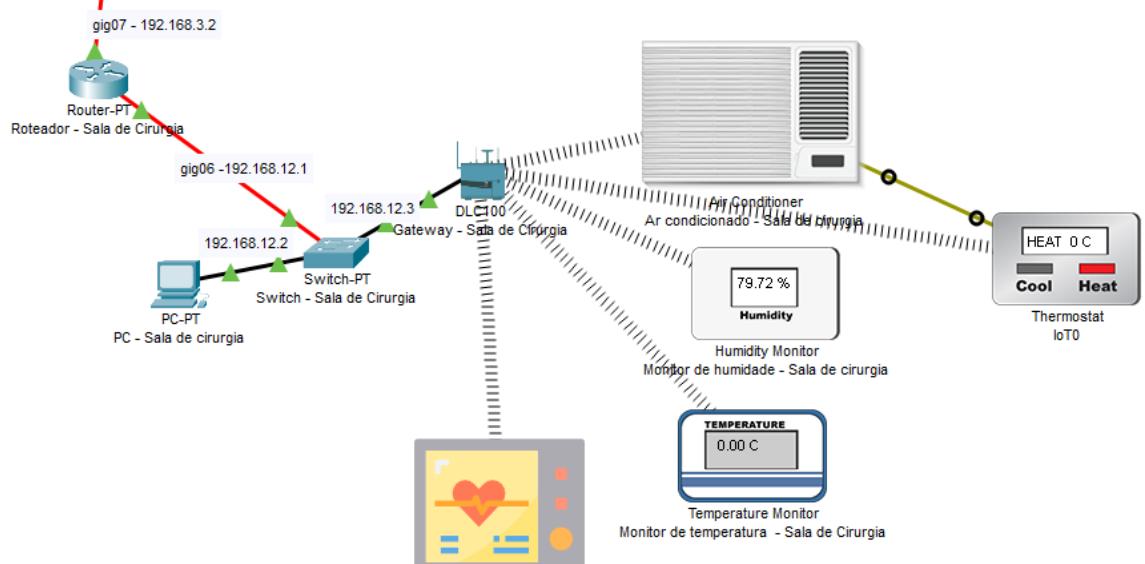


Fonte: Própria autoria

Assim como fiz na simulação da aula 14 parte 1, utilizei o protocolo RIP para que os roteadores pudessem conhecer os endereços das diferentes sub-redes que os outros roteadores estão conectados.

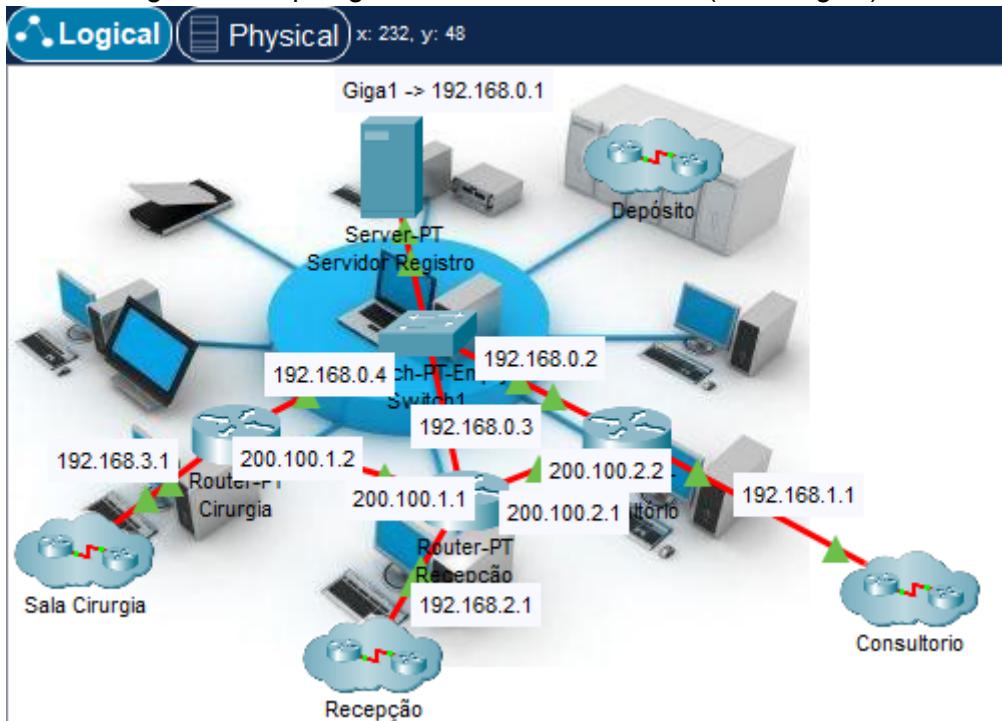
Quis aumentar ainda mais a complexidade do trabalho, e então adicionei mais um roteador a cada uma das instalações. Totalizando 6 roteadores em todo o projeto. Um exemplo da topologia até este momento pode ser visto na Figura 22, que mostra a sala de cirurgia em sua visualização lógica. Ainda sim os pacotes não estavam sendo recebidos de um cluster para outro, revisei então todos os endereços das redes que os roteadores se conectavam, troquei os endereços de IP para a classe dos 192.168.x.x., com exceção das conexões entre os roteadores principais, cujos ips mantive na casa dos 200.100.x.x, por uma questão de organização. Essas mudanças podem ser vistas na Figura 23.

Figura 22. Topologia da Sala de Cirurgia até o momento (visão lógica).



Fonte: Própria autoria

Figura 23. Topologia da rede até o momento (visão lógica).

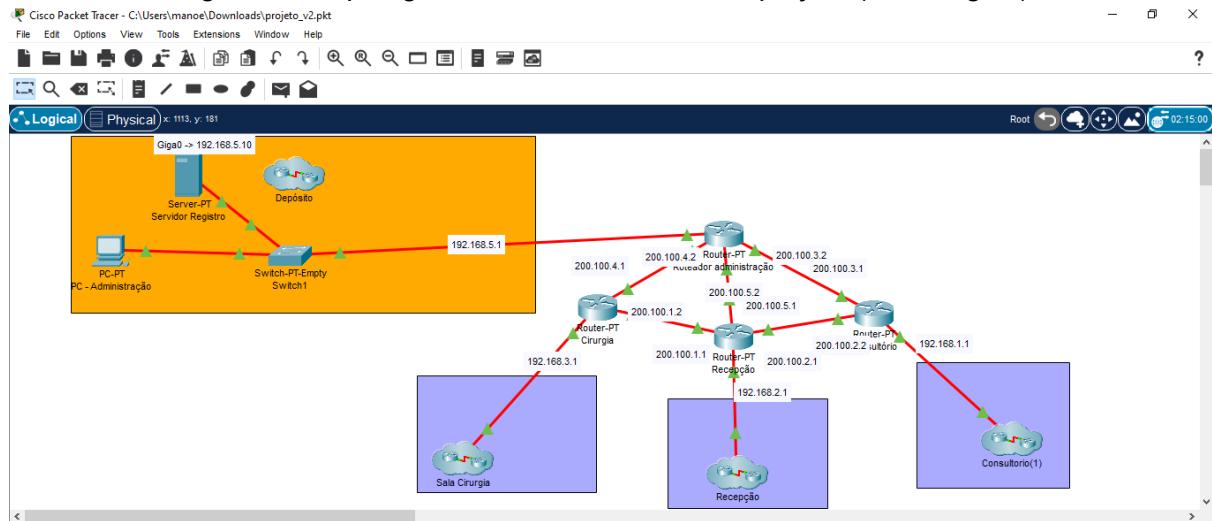


Fonte: Própria autoria

Posteriormente, realizei diversos testes de diferentes pacotes sendo enviados de um cluster para outro, e também de vários equipamentos dentro dos clusters para o servidor registro. Aconteceram diversas falhas, que me fizeram perceber que eu necessitava revisar cada elemento novamente. Por isso criei uma nova versão do projeto, revisitando cada elemento. Organizei melhor a topologia, de maneira

com que ficasse mais claro onde cada equipamento está e quais são os endereços de cada interface. O resultado dessa melhoria pode ser visto na Figura 24.

Figura 24. Topologia da rede na versão 2 do projeto (visão lógica).



Fonte: Própria autoria

Separei por cores o que fazia parte da administração (laranja) e o que fazia parte do hospital em si (lilás). E então iniciei os testes. Vale destacar que o PC da administração do projeto deveria ter acesso a todos os objetos de IoT do hospital, assim como todos os dispositivos de controle de IoTs.

Alguns dos testes realizados são mostrados na Figura 25. E na Figura 26 é possível ver a lista de eventos disparados no modo simulação, e na Figura 27 mostra-se a tabela de roteamento do roteador nomeado “Cirurgia” que aparece na topologia da Figura 24.

Figura 25. Topologia da rede na versão 2 do projeto (visão lógica).

Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit	Delete
	Successful	Medidor de Heart Rate	Servidor Registro	ICMP		0.000	N	0	(edit)	(delete)
	Successful	Medidor de Heart Rate	PC - Administração	ICMP		0.000	N	1	(edit)	(delete)
	Successful	Medidor de Heart Rate	PC - Administração	ICMP		0.000	N	2	(edit)	(delete)
	Successful	Gateway Recepção	Cirurgia	ICMP		0.000	N	3	(edit)	(delete)
	Successful	Câmera - Recepção	Gateway - Sala d...	ICMP		0.000	N	4	(edit)	(delete)
	Successful	PC - Sala de cirurgia	PC - recepção	ICMP		0.000	N	5	(edit)	(delete)
	Successful	PC - Sala de cirurgia	PC - recepção	ICMP		0.000	N	6	(edit)	(delete)

Fonte: Própria autoria

Figura 26. Parte da lista de eventos disparados no modo simulação.

Event List				
Vis.	Time(sec)	Last Device	At Device	Type
0.002	Switch - Recepção		Recepção	ICMP
0.002	Switch - Sala de Cirurgia		Cirurgia	ICMP
0.003	Switch - Sala de Cirurgia		Cirurgia	ICMP
0.003	Recepção		Cirurgia	ICMP
0.003	Cirurgia		Recepção	ICMP
0.003	--		Switch - Sala de Cirurgia	ICMP
0.004	Switch - Sala de Cirurgia		Cirurgia	ICMP
0.004	Cirurgia		Roteador administração	ICMP
0.004	Roteador administração		Recepção	ICMP
0.004	Recepção		Switch - Recepção	ICMP
0.005	Cirurgia		Recepção	ICMP
0.005	Roteador administração		Switch1	ICMP
0.005	Recepção		Switch - Recepção	ICMP
0.006	Switch - Sala de Cirurgia		Cirurgia	ICMP
0.006	Recepção		Switch - Recepção	ICMP
0.006	Switch1		Servidor Registro	ICMP
0.007	Cirurgia		Roteador administração	ICMP
0.007	Servidor Registro		Switch1	ICMP
0.007	Switch - Recepção		Recepção	ICMP
0.008	Roteador administração		Switch1	ICMP
0.008	Switch1		Roteador administração	ICMP
0.008	Recepção		Cirurgia	ICMP
0.009	Switch1		PC - Administração	ICMP
0.009	Switch - Recepção		Recepção	ICMP
0.009	Roteador administração		Cirurgia	ICMP
0.009	Cirurgia		Switch - Sala de Cirurgia	ICMP
0.010	PC - Administração		Switch1	ICMP
0.010	Recepção		Cirurgia	ICMP
0.010	Cirurgia		Switch - Sala de Cirurgia	ICMP
0.011	Switch1		Roteador administração	ICMP
0.011	Cirurgia		Switch - Sala de Cirurgia	ICMP
0.012	Roteador administração		Cirurgia	ICMP
0.013	Cirurgia		Switch - Sala de Cirurgia	ICMP

Fonte: Própria autoria

Figura 27. Tabela de roteamento com sua legenda no Roteador com nome “Cirurgia”.

The screenshot shows a Cisco IOS Command Line Interface window titled "Cirurgia". The tabs at the top are "Physical", "Config", "CLI" (which is selected), and "Attributes". The main area displays the output of the "sh ip rou" command. A red box highlights the legend and the first few entries of the routing table. The legend provides the following route types:

- C - connected
- S - static
- I - IGRP
- R - RIP
- M - mobile
- B - BGP
- D - EIGRP
- EX - EIGRP external
- O - OSPF
- IA - OSPF inter area
- N1 - OSPF NSSA external type 1
- N2 - OSPF NSSA external type 2
- E1 - OSPF external type 1
- E2 - OSPF external type 2
- E - EGP
- i - IS-IS
- L1 - IS-IS level-1
- L2 - IS-IS level-2
- ia - IS-IS inter area
- \* - candidate default
- U - per-user static route
- o - ODR
- P - periodic downloaded static route

Below the legend, it says "Gateway of last resort is not set". The routing table entries are:

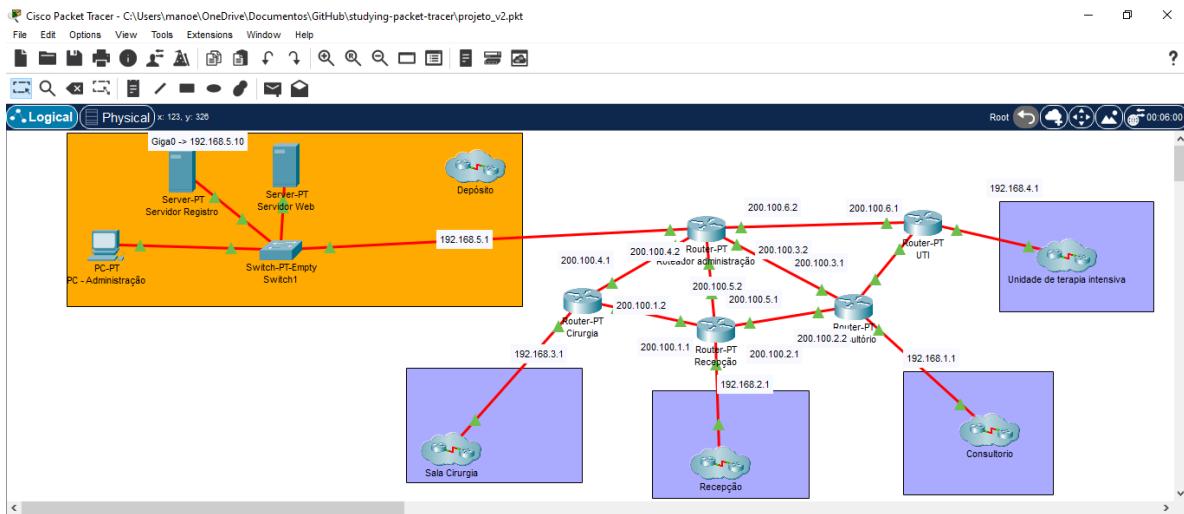
Type	Network	Mask	Next Hop	Cost	Interface
R	192.168.1.0	/24	[120/2]	via 200.100.1.1, 00:00:10, GigabitEthernet8/0	
R	192.168.2.0	/24	[120/1]	via 200.100.4.2, 00:00:07, GigabitEthernet6/0	
C	192.168.3.0	/24	is directly connected, GigabitEthernet7/0		
R	192.168.5.0	/24	[120/1]	via 200.100.4.2, 00:00:07, GigabitEthernet6/0	
	200.100.1.0	/30	is subnetted, 1 subnets		
C	200.100.1.0		is directly connected, GigabitEthernet8/0		
R	200.100.2.0	/24	[120/1]	via 200.100.1.1, 00:00:10, GigabitEthernet8/0	
R	200.100.3.0	/24	[120/1]	via 200.100.4.2, 00:00:07, GigabitEthernet6/0	
	200.100.4.0	/30	is subnetted, 1 subnets		
C	200.100.4.0		is directly connected, GigabitEthernet6/0		
R	200.100.5.0	/24	[120/1]	via 200.100.4.2, 00:00:07, GigabitEthernet6/0	
			[120/1]	via 200.100.1.1, 00:00:10, GigabitEthernet8/0	

At the bottom, the prompt shows "Router>" followed by a blank line.

Fonte: Própria autoria

Meu planejamento inicial era que o projeto contivesse 5 instalações, abstraindo o escritório da administração do hospital, a sala de cirurgia, a sala de recepção, o consultório e a unidade de terapia intensiva. Assim, criei mais um cluster, o cluster da UTI, e com ele adicionei mais um nível de complexidade configurando o serviço de DNS no servidor e adicionando um servidor web ao projeto. A Figura 28 a seguir apresenta a topologia final da rede projetada.

Figura 28. Topologia final da rede do Hospital Lagos (visão lógica).



Fonte: Própria autoria

Para controlar os objetos de IoT, todos aqueles dispositivos que estiverem registrados no servidor registro aparecem em uma lista que pode ser acessada de duas formas. A primeira forma é através da navegação até a aba de “web browser” nos equipamentos que possuem essa possibilidade e que estão na mesma rede local que os objetos de IoT, nessa página, deve-se digitar no navegador o endereço IP do servidor registro e ao logar com o usuário e senha determinado, é possível visualizar todos os dispositivos IoT conectados. Outra forma de gerenciar esses dispositivos é através da aba “IoT Monitor”. Por exemplo, ao acessar a página IoT Monitor no PC da administração obtemos a seguinte lista de objetos gerenciáveis na Figura 28.

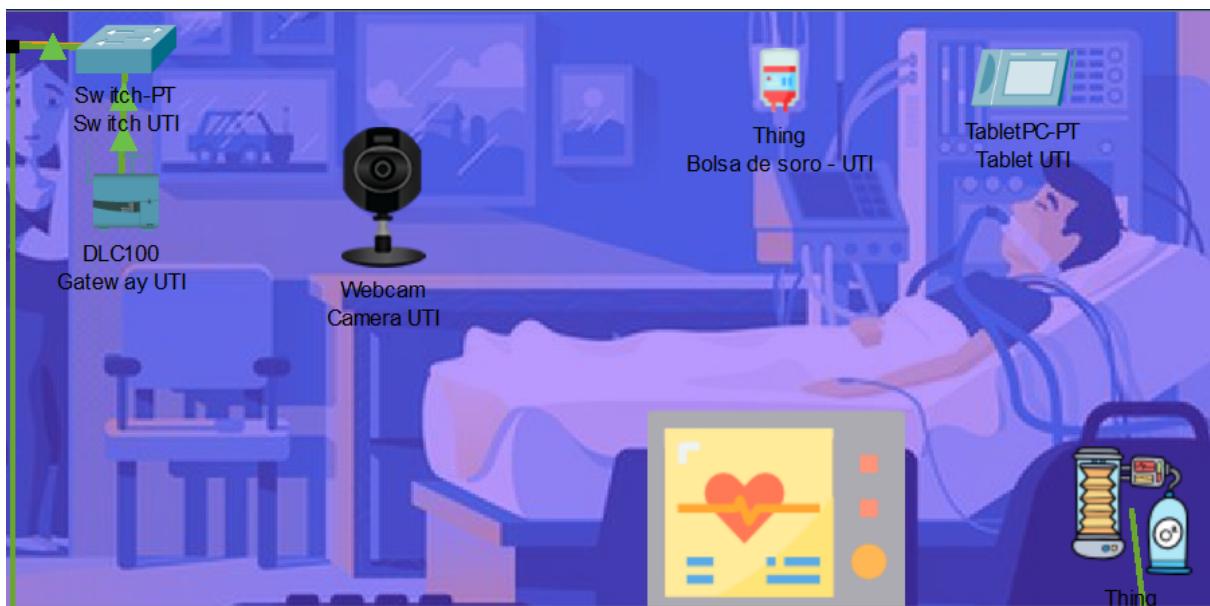
Figura 28. Parte da lista de dispositivos IoT gerenciáveis pelo PC da administração.



Fonte: Autoria própria

Programei mais dispositivos que não existiam no Packet Tracer, mas que são importantes no Hospital e, principalmente, um hospital que se utiliza de IoMT (Internet das Coisas Médicas). Desenvolvi um dispensador de soro para um paciente, e um respirador. Ambos foram inseridos na UTI do Hospital e, em específico o dispensador de soro foi conectado via wireless ao Gateway UTI e o respirador está conectado via cabo de cobre simples, uma vez que imaginei que esse dispositivo no mundo real não poderia ser ligado e desligado via wireless que é uma tecnologia com possibilidade de perdas de pacotes mais alta do que uma conexão cabeada. A Figura 29 mostra a visão física da UTI mostrando os aparelhos mencionados anteriormente assim como uma câmera de monitoramento do paciente, um medidor de batimentos cardíacos e também um tablet para o médico que adentrar a sala possa verificar o status de todos os equipamentos conectados e assim auxiliar em seu diagnóstico.

Figura 29. Unidade de Terapia Intensiva do Hospital (visão física).

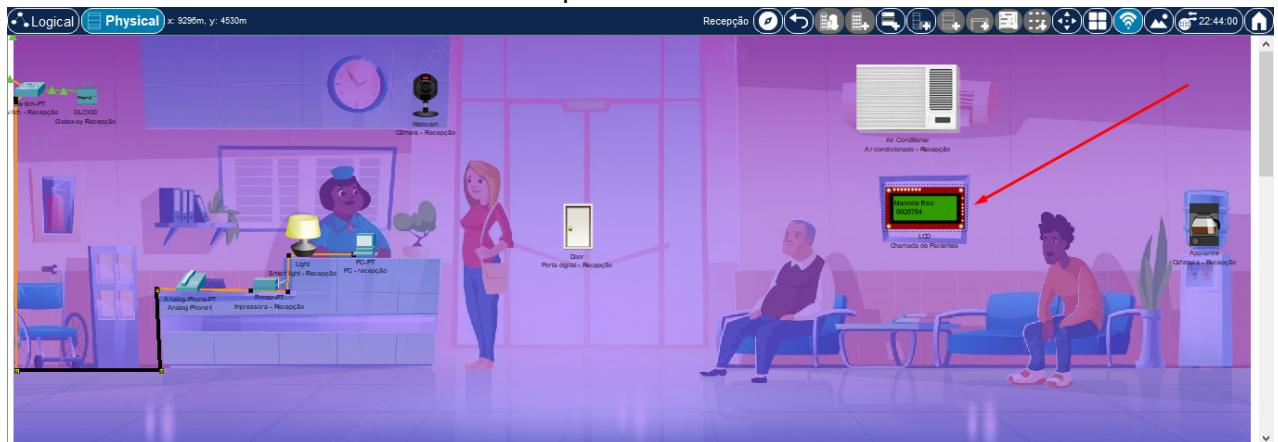


Fonte: Equipamentos: Autoria própria; Imagem de Fundo:  
<https://dribbble.com/shots/3607019-Hospital-scene>

Programei usando Javascript inclusive um painel de LCD para chamar nomes de pacientes na sala de recepção. A recepcionista do hospital apenas digita o nome do paciente no computador, aciona um botão e o nome do paciente aparece no dispositivo que foi posicionado na parede da sala de recepção. A visão física da sala

de recepção e do equipamento citado pode ser conferida na Figura 30 a seguir, e o código desenvolvido para este dispositivo funcionar está presente no Anexo 2.

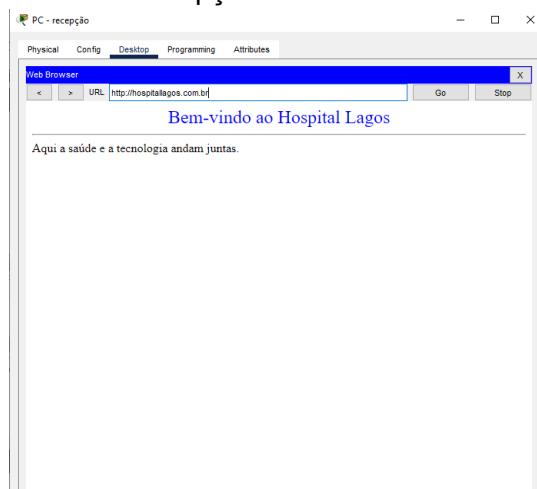
Figura 30. Recepção do Hospital (visão física). Seta vermelha indica o painel de chamada de pacientes



Fonte: Equipamentos: Autoria própria; Imagem de Fundo:  
<https://www.istockphoto.com/pt/vetorial/people-in-hospital-hallway-clinic-hall-interior-gm1257707382-368678024>

Para testar os serviços de DNS e HTTP/HTTPS pode-se utilizar qualquer dispositivo conectado à rede. Inclusive o PC presente na recepção (mostrado na Figura 30). Para testar o DNS fiz uma configuração no servidor para que ele conhecesse a string que os usuários da rede digitariam em seus navegadores, assim como criei uma página HTML básica para simular o site do hospital. O site então seria acessado via “<http://hospitallagos.com.br>” e está associado ao servidor web cujo endereço IP é 192.168.5.3. A Figura 31 apresenta o teste descrito sendo bem sucedido.

Figura 31. PC da recepção acessando o site do Hospital.



Fonte: Própria Autoria

## **5. Conclusão**

Diante dos resultados apresentados, conclui-se que é possível simular uma rede com objetos de IoMT em um hospital inteligente. O uso dos protocolos RIP, DHCP, ICMP, IP, entre outros presentes no trabalho em questão, pôde fornecer à autora um reforço nos conceitos aprendidos nas aulas de Redes de Computadores. Além disso, destaca-se o fato que o software Cisco Packet Tracer é um excelente aliado ao aprendizado de redes e até de conceitos mais atuais como o de Internet das Coisas. Apesar de desafiador, realizar esse projeto foi bem divertido e todo o processo de construção da rede me fez inclusive querer ir além com esse software e até montar um treinamento do mesmo para meu grupo de estudos da universidade.

## 6. Referências

ACETO, G., et. al. (2020). Industry 4.0 and health: Internet of things, big data, and cloud computing for healthcare 4.0. **Journal of Industrial Information Integration**.

CISCO NETWORK ACADEMY. Perguntas frequentes sobre o Cisco Packet Tracer. Disponível em: <<https://www.netacad.com/pt-br/courses/packet-tracer/faq#1>>, Acesso em: 10 de janeiro de 2023.

GIT. About. Disponível em: <<https://git-scm.com/about>>, Acesso em: Acesso em: 17 jan. 2023.

GITHUB. Let's build from here. (2023), Disponível em: <<https://github.com/about>>, Acesso em: Acesso em: 17 jan. 2023.

IBM. Comando ping (2021). Disponível em: <<https://www.ibm.com/docs/pt-br/power8?topic=commands-ping-command>>, Acesso em: 17 jan. 2023.

KUROSE, J. F.; ROSS, K. W. Computer Networking: A Top-Down Approach, 7th Edition. **Pearson**, 2017.

RAZDAN, Sahshanu; SHARMA, Sachin. Internet of medical things (IoMT): overview, emerging technologies, and case studies. **IETE Technical Review**, v. 39, n. 4, p. 775-788, 2022.

ROSE, Karen; ELDRIDGE, Scott; CHAPIN, Lyman. The internet of things: An overview. **The internet society (ISOC)**, v. 80, p. 1-50, 2015.

TANENBAUM, A.S; WETHERALL, D. Redes de Computadores, 5a Edição. **Pearson**, 2011.

TOMÉ, Lorenzo. Internet das Coisas Médicas: entenda a Revolução da IoMT!. Saúde Digital. 2022. Disponível em: <<https://saudedigital.tech/internet-das-coisas-medicas-entenda-a-revolucao-iomt/>>, Acesso em 2 de jan. 2023.

VISHNU, S; RAMSON, S.R. Jino; JEGAN, R. Internet of medical things (IoMT)-An overview. In: **2020 5th international conference on devices, circuits and systems (ICDCS)**. IEEE, 2020. p. 101-104.

## **ANEXO 1 - Código do Monitor de Heart Rate**

```
function setup() {  
    pinMode(1, OUTPUT);  
    Serial.println("Blinking");  
}  
  
function loop() {  
    addSound('sound1', 'Sounds/buzzLow.wav');  
    digitalWrite(1, HIGH);  
    playSound('sound1', 1);  
    delay(1000);  
    digitalWrite(1, LOW);  
    delay(1000);  
}
```

## ANEXO 2 - Código do Painel de Chamada de Pacientes

```
// Purpose:  
// Display text.  
  
// How many characters are allowed per line and the maximum number of lines allowed.  
var charLimitPerLine = 14;  
var lineLimit = 2;  
  
// Top left position and the clip area size for the text.  
var textPos = {x: 16, y: 20};  
var textAreaSize = {w: 230, h: 52 };  
  
var state = ";  
  
IoEClient.setup({  
    type: "Chamada de Pacientes",  
    states: [{  
        name: "Nome do próximo paciente:",  
        type: "string",  
        controllable: true  
    }]  
});  
  
// Purpose:  
// Set initial text to display.  
function setup() {  
  
    IoEClient.onInputReceive = function(input) {  
        processData(input, true);  
    };  
    attachInterrupt(0, function() {  
        setText(customRead(0));  
    });  
  
    attachInterrupt(A0, function() {  
        setText(" + ((255*analogRead(A0))/1023));  
    });  
  
    setText('Manoela Reis \n 8928764');  
}  
  
// Purpose:  
// Set the displayed text to the given text. Apply restrictions.  
function setText(text) {  
    if((null !== text) && (undefined !== text)) {  
        var textToDisplay = ";
```

```
var lines = text.split('\n');

var lineCount = lineLimit;
if(lines.length < lineCount)
    lineCount = lines.length;

for(var ind = 0; ind < lineCount; ++ind)
    textToDisplay += (lines[ind].substring(0, charLimitPerLine) + '\n');

setCustomText(textPos.x, textPos.y, textAreaSize.w, textAreaSize.h,
textToDisplay);
setDeviceProperty(getName(), "text", textToDisplay);
}

}

function processData(data, blsRemote) {
    if ( data.length <= 0 )
        return;
    setState(setText(data));
}

function setState(newState) {
    if (newState.length > charLimitPerLine)
        newState = "";
    state = newState;

    analogWrite(A1, state);
    customWrite(0, state);
    IoEClient.reportStates(state);
    setDeviceProperty(getName(), "state", state);
}
```