

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

PEDRO AFONSO CARRARO

**UM ESTUDO COMPARATIVO DE DESEMPENHO ENTRE AS DIFERENTES
VERSÕES DO PROTOCOLO HTTP**

GUARAPUAVA

2025

PEDRO AFONSO CARRARO

**UM ESTUDO COMPARATIVO DE DESEMPENHO ENTRE AS DIFERENTES
VERSÕES DO PROTOCOLO HTTP**

**A comparative study of performance between the different HTTP protocol
versions**

Projeto de Trabalho de Conclusão de Curso de Graduação apresentado como requisito para obtenção do título de Tecnólogo em Tecnologia em Sistemas para Internet do Curso Superior de Tecnologia em Sistemas para Internet da Universidade Tecnológica Federal do Paraná.

Orientador: Prof. Dra. Sediane Carmem Lunardi
Hernandes

Coorientador: Prof. Dr. Hermano Pereira

GUARAPUAVA

2025



[4.0 Internacional](https://creativecommons.org/licenses/by/4.0/)

Esta licença permite compartilhamento, remixe, adaptação e criação a partir do trabalho, mesmo para fins comerciais, desde que sejam atribuídos créditos ao(s) autor(es). Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.

LISTA DE ABREVIATURAS E SIGLAS

Abreviaturas

art.	Artigo
cap.	Capítulo
sec.	Seção

Siglas

UTFPR	Universidade Tecnológica Federal do Paraná
HTTP	Protocolo de Transferência de Hipertexto, do inglês <i>Hypertext Transfer Protocol</i>
UDP	Protocolo de datagrama do usuário, do inglês <i>User Datagram Protocol</i>
TCP	Protocolo de Controle de Transmissão, do inglês <i>Transmission Control Protocol</i>
IP	Protocolo de Internet, do inglês <i>Internet Protocol</i>
QUIC	Conexão Rápida UDP de Internet, do inglês <i>Quick UDP Internet Connections</i>

SUMÁRIO

1	INTRODUÇÃO	3
1.1	Objetivo geral	4
1.2	Objetivos específicos	4
1.3	Justificativa	4
1.4	Estrutura do trabalho	5
2	REFERENCIAL TEÓRICO	7
2.1	O protocolo HTTP	7
2.2	HTTP/1.1	8
2.3	HTTP/2	8
2.4	HTTP/3	8
2.5	Comparação entre as diferentes versões do HTTP	9
2.6	Trabalhos relacionados	11
3	PROPOSTA	12
4	MATERIAIS E MÉTODOS	14
4.1	Documentação teórica	14
4.2	Servidor	14
4.3	Ferramentas de análise	14
4.4	Métodos	15
4.5	Cronograma	16
5	CONSIDERAÇÕES FINAIS	18
	REFERÊNCIAS	19

1 INTRODUÇÃO

Embora muitas vezes tratada como uma inovação recente e disruptiva, a Internet é, em certo sentido, a materialização técnica de ideias muito mais antigas. A noção de uma rede interligando inteligências humanas remonta a séculos anteriores à invenção dos computadores. No século XVII, por exemplo, Gottfried Wilhelm Leibniz idealizou uma máquina capaz de organizar todo o conhecimento humano por meio de uma linguagem universal. Athanasius Kircher, por sua vez, concebia sistemas que conectassem mentes por meio de tubos e sons — visões que, embora rudimentares, antecipavam a ideia de uma comunicação global automatizada. A Internet é fruto de uma longa tradição de sonhos e especulações sobre como expandir a capacidade da mente humana por meio de redes artificiais (SMITH, 2022).

Com o avanço tecnológico e a digitalização da sociedade, essas ideias escalaram e tomaram forma. Atualmente, a Internet é uma infraestrutura essencial para o funcionamento da vida contemporânea. Ela está presente em praticamente todos os aspectos do cotidiano, sendo alguns deles, a comunicação pessoal, os serviços bancários, a educação, o trabalho remoto, o comércio eletrônico, o lazer e o acesso a informações. Segundo o Pew Research Center (2024), mais de 90% dos adultos em países desenvolvidos utilizam a Internet diariamente. No Brasil, a pesquisa TIC Domicílios do Cetic.br (2024) aponta que 84% dos lares têm acesso à rede, sendo o celular o principal meio de conexão. Essa presença definitiva e constante revela uma dependência crescente da sociedade em relação às tecnologias digitais e à infraestrutura que as sustenta.

O crescimento da Internet veio acompanhado de diversos desafios relacionados ao seu desempenho e à confiabilidade. Problemas como lentidão no carregamento de páginas, interrupções em serviços online, congestionamento de tráfego e alto consumo de dados impactam diretamente a experiência do usuário e a eficiência de sistemas que dependem da rede. Tais falhas técnicas não são triviais; elas afetam desde a produtividade no ambiente de trabalho até a qualidade do ensino remoto, o funcionamento de serviços públicos digitais e a competitividade de negócios online. Isso evidencia a importância de compreender não apenas as estruturas subjacentes que sustentam o funcionamento da Internet, mas também os protocolos de alto nível.

Dentre os diversos protocolos existentes, destaca-se o *Hyper Transfer Protocol* (HTTP), responsável por definir as regras para a comunicação entre servidores e clientes (i.e., navegadores) na *World Wide Web* (Web) (FOROUZAN; MOSHARRAF, 2013).¹ Mesmo sendo invisível para a maior parte dos usuários, o HTTP é um dos pilares da Web, sendo responsável por 82% do tráfego da Internet (CLOUDFLARE, 2024a). Nesse contexto, é compreensível que a sua eficiência esteja diretamente ligada aos problemas citados anteriormente. Com isso, foram

¹ A Web é um serviço cliente-servidor distribuído que permite a recuperação de documentos por meio de páginas Web Forouzan e Fegan (2009).

desenvolvidas diferentes versões do protocolo, buscando superar limitações técnicas e oferecer mais velocidade, segurança e confiabilidade na comunicação.

Por fim, este trabalho tem como objetivo confrontar o desempenho das diferentes versões do HTTP, com ênfase nas versões HTTP/1.1, HTTP/2 e HTTP/3. Busca-se, através de testes, identificar as vantagens e limitações de cada versão em diferentes contextos de uso, quanto ao carregamento de páginas em redes com estabilidade distinta, verificando o tempo de resposta e o consumo de dados. Com isso, pretende-se fornecer subsídios para profissionais e para o meio acadêmico da área na tomada de decisões mais coerentes sobre a adoção de cada versão conforme as necessidades específicas de seus projetos, sistemas ou aplicações.

1.1 Objetivo geral

Analisar comparativamente o desempenho das versões do protocolo HTTP, com foco nas versões HTTP/1.1, HTTP/2 e HTTP/3.

1.2 Objetivos específicos

Os objetivos específicos deste trabalho são:

- Analisar as especificações técnicas das versões HTTP/1.1, HTTP/2 e HTTP/3, com base nas respectivas *Request for Comments* (RFCs), destacando suas principais características, inovações introduzidas e limitações observadas em cada uma.
- Examinar a complexidade de configuração das diferentes versões do protocolo HTTP em um ambiente real, utilizando o servidor web NGINX como base para a comparação, a fim de compreender os requisitos práticos para sua adoção.
- Avaliar o comportamento das versões HTTP/1.1, HTTP/2 e HTTP/3 quanto ao desempenho, considerando métricas como o tempo de resposta, por exemplo, eficiência na utilização da largura de banda e estabilidade em cenários variados.

1.3 Justificativa

O protocolo HTTP, utilizado na Web, é responsável por grande parte do tráfego global da Internet. Essa grande dominância acaba subdividida entre suas versões, sendo que em 2024 apenas 21% do tráfego foi representado pelo HTTP/3, o tráfego HTTP/2 correspondeu a aproximadamente 50% e as versões HTTP/1.0 e HTTP/1.1 equivaleram a cerca de 30% do tráfego (CLOUDFLARE, 2024b).

Essa diferença na distribuição do tráfego do protocolo pode ser explicada pelo fato do HTTP/3 ser relativamente recente e em processo de adoção. Isso resulta em menos documen-

tação e experiência consolidada sobre seu uso em diferentes contextos, causando certa incerteza de sua eficiência. Paralelamente, versões anteriores continuam sendo usadas porque, em muitos casos, ainda atendem de forma satisfatória às necessidades, além de não exigirem modificações nas aplicações e servidores já existentes.

Nesse contexto, torna-se relevante analisar e comparar o desempenho destas diferentes versões, levando em conta que cada uma pode apresentar vantagens ou desvantagens a depender do ambiente em que se encontra. A velocidade de carregamento de um site, por exemplo, é um dos principais fatores que impactam a usabilidade e a percepção de qualidade por parte do usuário. Segundo Nielsen (2010), atrasos acima de 1 segundo já interrompem o fluxo de pensamento do usuário, e acima de 10 segundos geram frustração e abandono. Dados mais recentes também mostram que essa tolerância tem diminuído, uma pesquisa citada pela Dycoders (2023) aponta que 47% dos usuários esperam que a página carregue em até 2 segundos, e 40% abandonam o site se levar mais de 3 segundos. Esses fatores contribuem diretamente para altas taxas de rejeição, frustração do usuário e perda de oportunidades de negócio. Em contrapartida, um site rápido e com bom desempenho pode melhorar a satisfação do usuário. Essa comparação pode auxiliar profissionais e instituições a tomarem decisões mais embasadas quanto à adoção ou manutenção de determinadas versões do protocolo, conforme suas demandas específicas.

Assim, a finalidade deste trabalho provém da necessidade de uma melhor compreensão do comportamento do protocolo e suas versões na prática, contribuindo para um uso mais consciente e adequado de acordo com cada situação. Os resultados deste estudo podem beneficiar tanto a comunidade acadêmica, ao ampliar o conhecimento técnico sobre o tema, quanto o setor de tecnologia, ao apresentar um estudo prático do protocolo e seu desempenho. O estudo busca contribuir para um campo ainda em consolidação na literatura técnica e acadêmica, especialmente no que se refere à comparação prática entre as versões do protocolo HTTP e sua configuração em ambientes reais. Embora existam documentos normativos, como as RFCs, e relatórios técnicos de desempenho, há uma escassez de trabalhos que integrem essas abordagens de forma aplicada, o que reforça o potencial deste estudo como base para futuras pesquisas e aplicações.

Por fim, a escolha do tema também reflete um interesse pessoal do autor na área de redes e protocolos, reforçado por experiências acadêmicas relacionadas à infraestrutura da Web. A viabilidade do estudo é garantida pelo acesso a ambientes de teste e ferramentas adequadas, os quais permitem a realização de análises práticas e objetivas do desempenho das diferentes versões do protocolo HTTP.

1.4 Estrutura do trabalho

O trabalho é estruturado como segue. O Capítulo 1 apresenta a introdução do trabalho, objetivos e justificativa. No Capítulo 2, apresenta-se o referencial teórico do trabalho e no Capí-

tulo 3, a proposta. Materiais e métodos são apresentados no Capítulo 4, as considerações finais são descritas no Capítulo 5 e, posteriormente, as referências utilizadas.

2 REFERENCIAL TEÓRICO

A Web teve origem como uma plataforma voltada à disseminação de informações científicas, mas rapidamente evoluiu para uma das principais estruturas de comunicação, comércio e entretenimento do mundo contemporâneo. Essa transformação foi possível graças à arquitetura em camadas da Internet, que permite a interoperabilidade entre dispositivos e aplicações distribuídas globalmente (FOROUZAN; FEGAN, 2009).

No centro dessa arquitetura estão os protocolos fundamentais Internet Protocol (IP) e Transmission Control Protocol (TCP). O IP é encarregado do endereçamento e do roteamento dos pacotes de dados, garantindo que as informações cheguem ao seu destino correto. Já o TCP provê um serviço de transporte confiável, orientado à conexão, com controle de fluxo e mecanismos de retransmissão para lidar com perdas de pacotes (FOROUZAN; FEGAN, 2009).

Esses protocolos formam a base da chamada pilha TCP/IP, composta por camadas como a de enlace, de rede, de transporte e de aplicação. Na camada de transporte, além do TCP, destaca-se o User Datagram Protocol (UDP), um protocolo mais simples, sem verificação de estado e que não exige o estabelecimento de conexão entre emissor e receptor. O UDP é amplamente utilizado em aplicações sensíveis à latência, como chamadas de voz e transmissões ao vivo, justamente por evitar os atrasos inerentes ao controle de confiabilidade do TCP (FOROUZAN; FEGAN, 2009).

A camada de aplicação, por sua vez, é onde atua o protocolo *Hypertext Transfer Protocol* - HTTP, responsável por estruturar a comunicação entre navegadores e servidores. Ao longo do tempo, o HTTP passou por diferentes versões, cada uma buscando resolver limitações da anterior no que tange à performance, paralelismo e segurança. A seção a seguir detalha o protocolo HTTP e as principais características das versões HTTP/1.1, HTTP/2 e HTTP/3, com base nas respectivas especificações publicadas pela Internet Engineering Task Force (IETF) (FIELDING; NOTTINGHAM; RESCHKE, 2022; BELSHE; PEON; THOMSON, 2015; BISHOP, 2022).

2.1 O protocolo HTTP

O protocolo HTTP é um protocolo da camada de aplicação do tipo requisição/resposta que opera sobre um protocolo de transporte confiável, o TCP, sendo utilizado para definir como os programas cliente e servidor podem ser implementados para recuperar páginas web Fielding, Nottingham e Reschke (2022). Logo, um cliente HTTP é um programa que estabelece uma conexão com um servidor com o objetivo de enviar uma ou mais requisições HTTP, enquanto um servidor HTTP é um programa que aceita conexões com o propósito de aceitar requisições HTTP e enviar respostas HTTP Fielding, Nottingham e Reschke (2022).

2.2 HTTP/1.1

A versão HTTP/1.1, padronizada em 1997 e ainda amplamente utilizada, introduziu melhorias em relação ao HTTP/1.0, como o suporte a conexões persistentes e mecanismos de controle de cache (FIELDING; NOTTINGHAM; RESCHKE, 2022). No entanto, essa versão apresenta limitações significativas quanto à multiplexação: por padrão, apenas uma requisição pode ser processada por conexão TCP. Isso leva ao fenômeno conhecido como *head-of-line blocking*, no qual uma requisição lenta ou bloqueada impede o envio das próximas pela mesma conexão.

Para contornar essa limitação, os navegadores passaram a abrir múltiplas conexões paralelas para o mesmo servidor, frequentemente de quatro a seis por domínio, a fim de possibilitar o carregamento simultâneo de recursos como *scripts*, imagens e folhas de estilo. No entanto, esse modelo aumenta o consumo de recursos tanto no cliente quanto no servidor, além de gerar maior sobrecarga de gerenciamento de conexões. Essas restrições motivaram o desenvolvimento de versões mais eficientes do protocolo, como o HTTP/2 e o HTTP/3.

2.3 HTTP/2

O HTTP/2, definido em 2015 pela IETF, representa uma evolução significativa do HTTP/1.1. Ele introduz um modelo binário de comunicação e permite a multiplexação de múltiplos *streams* em uma única conexão TCP, eliminando o *head-of-line blocking* a nível de aplicação. Além disso, traz compressão de cabeçalhos e permite ao servidor realizar *server push*, antecipando respostas antes mesmo que o cliente as solicite ¹ (BELSHE; PEON; THOMSON, 2015).

Apesar dos avanços, o HTTP/2 ainda herda algumas limitações do TCP. Como o protocolo é orientado à conexão, qualquer perda de pacote em uma conexão HTTP/2 pode impactar todos os *streams* multiplexados, pois o TCP exige a entrega ordenada dos dados. Esse problema é especialmente visível em redes instáveis, como as móveis.

2.4 HTTP/3

O HTTP/3 surge como uma resposta às limitações estruturais do HTTP/2 ao transferir sua camada de transporte do TCP para o protocolo Quick UDP Internet Connections (QUIC), desenvolvido pelo Google e posteriormente padronizado pela IETF. O QUIC é implementado sobre o UDP, o que permite maior controle da lógica de transporte na camada de aplicação, eliminando as restrições do modelo tradicional do TCP.

¹ O *server push* é um mecanismo pelo qual o servidor pode enviar proativamente recursos ao cliente, como scripts ou folhas de estilo, assim que detecta que eles provavelmente serão necessários, sem esperar por requisições explícitas. Isso visa reduzir o tempo de carregamento de páginas web.

Uma das principais inovações do QUIC é a integração nativa com o Transport Layer Security (TLS)1.3, permitindo que a negociação criptográfica e o estabelecimento de conexão ocorram simultaneamente, em um único *handshake*. Essa abordagem reduz significativamente a latência, especialmente em conexões seguras, ao agilizar o tempo até a primeira resposta do servidor (*Time to First Byte – TTFB*).

Além disso, o QUIC oferece suporte à multiplexação de múltiplos *streams* independentes dentro de uma mesma conexão. Diferentemente do TCP, que impõe um único fluxo ordenado de *bytes*, o QUIC permite que cada *stream* opere de forma isolada. Isso elimina o *head-of-line blocking* (em português, o bloqueio de cabeçote de linha) na camada de transporte, isto é, um pacote na frente de uma fila impede que outros pacotes, que podem estar prontos para serem processados, sejam transmitidos. Logo, a perda de pacotes em um *stream* específico não impacta o processamento dos demais, promovendo maior robustez e desempenho em redes com perdas ou alta variação de latência.

Entre suas principais vantagens, o QUIC oferece:

- **Redução na latência de conexão:** combina o *handshake* criptográfico do TLS com o estabelecimento da conexão de transporte, eliminando rodadas adicionais de comunicação;
- **Multiplexação nativa e eficiente:** os *streams* são independentes, de forma que a perda de pacotes em um *stream* não afeta os demais;
- **Migração de conexão:** permite que uma sessão seja retomada após mudanças de IP, característica útil em dispositivos móveis;
- **Segurança integrada:** ao contrário do TCP, que depende de camadas adicionais para criptografia (como o TLS), o QUIC incorpora segurança por padrão.

Com essas características, o HTTP/3 busca atender às exigências de desempenho e confiabilidade da Web moderna, especialmente em dispositivos móveis e redes com maior variabilidade. Estudos recentes apontam ganhos relevantes de desempenho em diversas implementações e contextos (TREVISAN *et al.*, 2021; LIU *et al.*, 2024; BALEJ; SOCHOR, 2023).

2.5 Comparação entre as diferentes versões do HTTP

A versão HTTP/1.1 foi amplamente adotada, permanecendo como o padrão dominante por mais de duas décadas. Ela introduziu melhorias significativas em relação ao HTTP/1.0, como conexões persistentes (*keep-alive*) e mecanismos mais eficientes de controle de cache (FIELDING; NOTTINGHAM; RESCHKE, 2022). No entanto, com o crescimento da Web e o aumento da complexidade das páginas e aplicações, surgiram desafios relacionados à latência

e à eficiência do protocolo, sobretudo devido ao seu modelo de comunicação sequencial sobre o TCP (BELSHE; PEON; THOMSON, 2015).

Para superar essas limitações, foi desenvolvido o HTTP/2, que trouxe avanços expressivos na forma como os dados são transmitidos. Seu principal diferencial é a multiplexação de *streams* em uma única conexão TCP, eliminando o bloqueio por ordem de requisição e permitindo maior paralelismo. Adicionalmente, o protocolo incorporou a compressão de cabeçalhos por meio do algoritmo HPACK, reduzindo a sobrecarga causada pela repetição de informações como *cookies* e campos de autenticação (BELSHE; PEON; THOMSON, 2015).

Apesar dessas melhorias, o HTTP/2 ainda herdava as limitações do TCP, em especial os efeitos do (*head-of-line blocking*). Como o TCP garante a entrega ordenada dos pacotes, a perda de um único segmento pode atrasar a entrega de todos os demais streams multiplexados naquela conexão, comprometendo a performance em ambientes de rede com alta latência ou perda de pacotes.

Com o HTTP/3, o protocolo passa por uma mudança mais profunda ao adotar o QUIC como novo transporte subjacente, substituindo o TCP. O QUIC, por sua vez, é implementado sobre o UDP, e foi projetado para oferecer conexões mais rápidas, seguras e resilientes. Uma das inovações centrais do QUIC está em seu processo de *handshake*, que combina o estabelecimento de conexão e a negociação criptográfica em uma única etapa, reduzindo o tempo até a primeira resposta (*Time to First Byte – TTFB*) (BISHOP, 2022).

Além disso, o QUIC permite a multiplexação de streams independentes, o que significa que perdas de pacotes em um fluxo não afetam os demais, eliminando o problema do *head-of-line blocking*. Essa característica resulta em melhorias substanciais de desempenho, especialmente em redes móveis ou instáveis, onde a latência e a perda de pacotes são mais frequentes (BISHOP, 2022). A Figura 1 mostra a pilha de protocolos TCP/IP considerando as diferentes versões do protocolo HTTP.

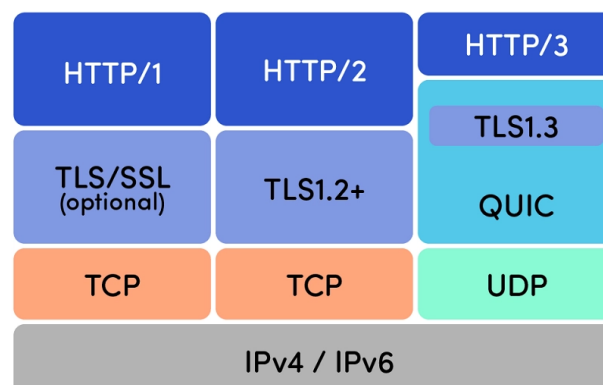


Figura 1 – Pilha de Protocolos considerando as diferentes versões do protocolo HTTP. Fonte: (COMMUNICATIONS, 2023)

2.6 Trabalhos relacionados

Diversos trabalhos recentes têm se dedicado à análise do protocolo HTTP, com foco em aspectos específicos como desempenho, adoção e impacto em diferentes ambientes. Trevisan *et al.* (2021), por exemplo, realizaram uma análise em larga escala da adoção e performance do HTTP/3 em domínios populares, revelando melhorias significativas na latência e no tempo de resposta, especialmente em cenários com alta perda de pacotes. No entanto, o estudo limita-se à versão HTTP/3, sem uma comparação prática com versões anteriores, nem abordagem sobre a configuração de servidores web em ambientes locais.

Já Liu *et al.* (2024) exploram o comportamento do HTTP/2 e HTTP/3 em ambientes com e sem proxy, investigando o impacto de *middleboxes* e CDNs (*Content Delivery Networks*) sobre a eficiência do protocolo. Embora o estudo aprofunde-se nas interações entre protocolos modernos e infraestrutura de rede, ele não contempla o HTTP/1.1, e sua análise se concentra no ambiente de produção com componentes adicionais de rede.

Outro estudo relevante é o de Balej e Sochor (2023), que compara diferentes implementações de servidores HTTP/3, como Caddy, OpenLiteSpeed e NGINX. O foco está na forma como cada servidor lida com a nova versão do protocolo, mas não há comparação direta entre as versões HTTP/1.1, HTTP/2 e HTTP/3 dentro de um mesmo ambiente.

Em contraste, este projeto de Trabalho de Conclusão de Curso propõe uma análise comparativa entre as versões HTTP/1.1 Fielding, Nottingham e Reschke (2022), HTTP/2 Belshe, Peon e Thomson (2015) e HTTP/3 Bishop (2022), com foco tanto nas especificações técnicas apresentadas pelas RFCs quanto no comportamento prático das versões em um mesmo servidor (NGINX). Além disso, busca-se examinar a complexidade de configuração, o impacto da performance no uso real da Web e identificar quais versões oferecem maior eficiência em diferentes cenários.

Assim, compreender essas três versões, seus mecanismos internos, vantagens e limitações, é essencial para avaliar o impacto do HTTP na experiência do usuário, no desempenho da Web e nas decisões técnicas adotadas por organizações. Essa análise se torna ainda mais relevante diante de um cenário em que cada milissegundo de atraso pode significar perda de receita, engajamento ou confiabilidade.

3 PROPOSTA

Este trabalho propõe uma análise comparativa entre as versões HTTP/1.1, HTTP/2 e HTTP/3, com foco em suas especificações técnicas, complexidade de configuração e desempenho em diferentes contextos. O objetivo é fornecer uma compreensão aprofundada sobre como essas versões evoluíram em termos de eficiência, latência, consumo de recursos e usabilidade prática. O problema central abordado é a carência de estudos atualizados que integrem análise teórica, testes práticos com servidor Web e interpretação de dados reais de tráfego, de forma consolidada e acessível.

A abordagem adotada será estruturada em três eixos principais:

- **Estudo técnico** das RFCs que definem cada versão do protocolo, destacando as mudanças mais relevantes, como multiplexação, compressão de cabeçalho, eliminação do *handshake* TCP e uso do QUIC.
- **Configuração prática** de cada versão utilizando o servidor NGINX, identificando as diferenças, dificuldades e exigências específicas de cada uma.
- **Avaliação de desempenho**, por meio de testes controlados e análise de relatórios existentes, que permitirão identificar os contextos em que cada versão apresenta melhor rendimento.

Adicionalmente, os resultados obtidos ao longo da pesquisa serão comparados com os dados apresentados em outros estudos relacionados, como os de Trevisan et al. (TREVISAN *et al.*, 2021), Liu et al. (LIU *et al.*, 2024) e Balej e Sochor (BALEJ; SOCHOR, 2023). Essa comparação servirá para verificar a consistência dos achados e identificar eventuais discrepâncias, contribuindo assim para a validação dos testes realizados e para o entendimento das variações de desempenho em diferentes cenários.

Entre os resultados esperados, destacam-se:

- Um mapeamento das vantagens e limitações de cada versão do protocolo HTTP;
- Um conjunto de observações práticas que podem auxiliar na escolha da versão mais adequada a diferentes tipos de serviços;
- A consolidação de informações dispersas em documentos técnicos e estudos, tornando-as acessíveis e aplicáveis tanto na academia quanto no mercado;
- A verificação empírica da compatibilidade entre os dados obtidos nesta pesquisa e os apresentados por trabalhos anteriores.

Por fim, o público-alvo da pesquisa inclui desenvolvedores web, administradores de sistemas, pesquisadores e profissionais da área de redes e infraestrutura. Além disso, a comunidade

acadêmica pode se beneficiar dos resultados como base para estudos futuros que envolvam protocolos de transporte e desempenho em aplicações web.

4 MATERIAIS E MÉTODOS

Esta seção descreve os materiais que serão utilizados para o desenvolvimento do trabalho e a metodologia adotada para a realização dos experimentos propostos neste estudo. O objetivo é conduzir uma análise comparativa entre as versões HTTP/1.1, HTTP/2 e HTTP/3, observando aspectos de desempenho, comportamento de rede e impacto na infraestrutura.

4.1 Documentação teórica

As especificações técnicas de cada versão do protocolo serão obtidas diretamente dos documentos normativos publicados pela IETF:

- RFC 9110 – HTTP/1.1 Semantics (FIELDING; NOTTINGHAM; RESCHKE, 2022)
- RFC 7540 – Hypertext Transfer Protocol Version 2 (HTTP/2) (BELSHE; PEON; THOMSON, 2015)
- RFC 9114 – HTTP/3 (BISHOP, 2022)

Esses documentos servirão não somente como base conceitual e técnica para a implementação e análise dos testes, mas também para o estudo técnico empregado sobre cada versão.

4.2 Servidor

O servidor web a ser utilizado nos testes será o NGINX, na sua versão mais recente, com suporte habilitado para HTTP/1.1, HTTP/2 e HTTP/3. O NGINX foi escolhido por sua ampla adoção na indústria, flexibilidade de configuração e suporte ativo às últimas versões do protocolo. Além disso, segundo Balej e Sochor (2023), entre diferentes servidores com suporte a HTTP/3, o NGINX apresentou o melhor desempenho geral nos testes realizados, destacando-se em métricas como estabilidade, tempo de resposta e uso de recursos.

4.3 Ferramentas de análise

Para os testes serão necessárias algumas ferramentas, dentre elas:

1. **H2Load**: Ferramenta de *benchmark* desenvolvida como parte do projeto nghttp2, utilizada para testar o desempenho de servidores HTTP/2 e HTTP/3. Permite simular múltiplas conexões e requisições concorrentes, fornecendo métricas como tempo de resposta, taxa de transferência e número de requisições por segundo. É útil para avaliar a eficiência e escalabilidade de implementações modernas do protocolo HTTP.

2. **htop**: Utilitário interativo de monitoramento de processos em sistemas Unix. Exibe em tempo real o uso de processamento, memória e outras métricas do sistema, sendo especialmente útil para observar o impacto de testes de carga no consumo de recursos do servidor durante a execução de experimentos com diferentes versões do HTTP.
3. **tc (Traffic Control)**: Ferramenta do Linux que permite manipular o tráfego de rede, simulando condições como latência, perda de pacotes e limitação de banda. A ferramenta é frequentemente utilizada para criar cenários de rede adversos em ambientes de teste, permitindo verificar o comportamento de protocolos como TCP, UDP e QUIC em situações realistas de instabilidade ou congestionamento.

4.4 Métodos

A metodologia adotada neste trabalho combina abordagem experimental com revisão bibliográfica, de modo a proporcionar uma compreensão abrangente sobre a evolução do protocolo HTTP. A revisão teórica fundamenta-se na análise das principais RFCs de cada versão do protocolo, bem como em artigos científicos recentes que investigam aspectos técnicos e práticos relacionados ao desempenho, adoção e implementação do HTTP/1.1, HTTP/2 e HTTP/3. Essa etapa servirá de base para definir os critérios de comparação e estruturar os testes experimentais.

Complementando essa análise teórica, serão realizados experimentos controlados e replicáveis em diferentes ambientes de rede e sob múltiplas condições de operação. Os ambientes definidos para os experimentos foram:

- **Local (Docker)**: Cliente e servidor simulados via contêineres Docker na mesma máquina. Ambiente de latência mínima, ideal para observar o comportamento puro dos protocolos.
- **Rede LAN**: Cliente e servidor em máquinas distintas conectadas via switch ou roteador. Permite simular uma rede interna com condições mais realistas e controle sobre os dispositivos de rede.
- **Nuvem (provedor externo)**: O servidor será hospedado em um ambiente de nuvem pública. O provedor ainda está em processo de escolha, podendo incluir alternativas como AWS, Oracle Cloud, entre outros. Esse cenário simula um ambiente real de internet, com latência variável e fatores externos de rede.

Para cada versão do protocolo, os testes serão executados nas mesmas condições de hardware, rede e carga, garantindo isonomia nos resultados. Desta forma, serão montados diferentes cenários de comunicação entre cliente(s) e servidor, considerando os três ambientes principais definidos: **local (Docker)**, **rede interna (LAN)** e **nuvem (externo)**. Cabe salientar

que em cada cenário serão testadas as versões HTTP/1.1, HTTP/2 e HTTP/3. A quantidade de clientes será variável entre um único cliente e múltiplos clientes simultâneos. A Tabela 1 apresenta os cenários descritos.

Tabela 1 – Cenários definidos para os testes

Cenário	Ambiente	Topologia	Protocolos	Sessões
1a	Local (Docker)	1 cliente → 1 servidor	HTTP/1.1, HTTP/2, HTTP/3	1 ... n
2a	Local (Docker)	n clientes → 1 servidor	HTTP/1.1, HTTP/2, HTTP/3	1 ... n
1b	Rede LAN	1 cliente → 1 servidor	HTTP/1.1, HTTP/2, HTTP/3	1 ... n
2b	Rede LAN	n clientes → 1 servidor	HTTP/1.1, HTTP/2, HTTP/3	1 ... n
1c	Nuvem (externo)	1 cliente → 1 servidor	HTTP/1.1, HTTP/2, HTTP/3	1 ... n
2c	Nuvem (externo)	n clientes → 1 servidor	HTTP/1.1, HTTP/2, HTTP/3	1 ... n

Os parâmetros a serem coletados para síntese e comparação em cada cenário de teste são apresentados abaixo:

- Quantidade de sessões simultâneas (simulando diferentes níveis de carga).
- Tamanhos variados de payload (arquivos pequenos e grandes).
- Métricas como latência, throughput, uso de CPU e memória.

Pretende-se realizar repetições dos testes em diferentes horários para observar as variações dos dados coletados. Os dados obtidos serão comparados de forma sistemática, com o auxílio de tabelas e gráficos, a fim de evidenciar os contrastes entre as versões analisadas. Essa abordagem possibilita avaliar não apenas os ganhos teóricos propostos pelas especificações formais, mas também observar como essas melhorias se refletem em ambientes práticos e reais de uso.

4.5 Cronograma

O cronograma de execução elaborado para o desenvolvimento do Trabalho de Conclusão de Curso 2 é apresentado na Tabela 2.

Tabela 2 – Cronograma Geral do Projeto

Período	Atividade
Julho (01–31/07)	Leitura aprofundada das RFCs e comparação teórica das versões do HTTP.
Agosto (01–15/08)	Conclusão do estudo comparativo e organização dos dados em tabela.
Agosto (16–31/08)	Configuração das versões do protocolo no servidor NGINX.
Setembro (01–15/09)	Redação das seções de configuração e ambiente de testes.
Setembro (16–30/09)	Escrita dos requisitos de configuração por versão.
Outubro (01–20/10)	Realização dos testes de desempenho nos cenários determinados.
Outubro (21–31/10)	Análise preliminar dos resultados.
Novembro (01–15/11)	Redação da seção de testes e considerações iniciais.
Novembro (16–30/11)	Redação final do TCC e revisão e ajustes de formatação.
Dezembro (01–10/12)	Entrega do TCC e preparação para a apresentação.

De acordo com o cronograma, espera-se concluir o trabalho e, ao final, redigir um artigo para envio em Simpósio Nacional relacionado à área de desenvolvimento do TCC.

5 CONSIDERAÇÕES FINAIS

Este trabalho tem como objetivo analisar e comparar diferentes versões do protocolo HTTP (HTTP/1.1, HTTP/2 e HTTP/3) com foco em desempenho, complexidade de configuração e comportamento prático em ambientes controlados. Espera-se, com isso, entender os avanços trazidos por cada versão e suas implicações no contexto atual da Web.

A relevância deste projeto reside no fato de que o HTTP é o principal protocolo de comunicação na Web, sendo diretamente responsável pela experiência de usuários, pela eficiência de aplicações online e pela otimização de recursos de rede. Diante da crescente adoção de novas versões do protocolo, como o HTTP/3, torna-se essencial compreender os ganhos reais que essas atualizações oferecem, assim como suas limitações.

Nesse contexto, espera-se que os resultados obtidos possam fornecer subsídios técnicos para profissionais e pesquisadores na escolha adequada da versão do protocolo a ser utilizada, conforme suas demandas específicas. Além disso, espera-se que a análise dos dados contribua com a literatura existente, oferecendo um panorama atualizado e prático sobre a adoção e desempenho das versões do HTTP.

Por fim, reconhece-se que alguns desafios podem surgir ao longo da execução, como a necessidade de configurar corretamente ambientes de teste que representem cenários reais de uso, as diferenças de suporte às versões do protocolo entre navegadores e servidores, além da complexidade envolvida na coleta e análise de métricas de desempenho. Também pode haver limitações relacionadas à variabilidade das condições de rede e à própria maturidade de implementações do HTTP/3 em determinados contextos. Apesar disso, espera-se que o trabalho contribua de forma significativa para a compreensão do tema e possa servir como base para estudos futuros sobre protocolos da camada de aplicação e desempenho da Web.

REFERÊNCIAS

- BALEJ, M.; SOCHOR, T. Performance comparison of http/3 server implementations. **Mendel University Repository**, 2023. Disponível em: https://doi.mendelu.cz/artkey/doi-990007-2000_PERFORMANCE-COMPARISON-OF-HTTP-3-SERVER-IMPLEMENTATIONS.php.
- BELSHE, M.; PEON, R.; THOMSON, M. **Hypertext Transfer Protocol Version 2 (HTTP/2)**. RFC Editor, 2015. RFC 7540. (Request for Comments, 7540). Disponível em: <https://www.rfc-editor.org/info/rfc7540>.
- BISHOP, M. **HTTP/3**. RFC Editor, 2022. RFC 9114. (Request for Comments, 9114). Disponível em: <https://www.rfc-editor.org/info/rfc9114>.
- CETIC.BR. **TIC Domicílios 2023: Pesquisa sobre o uso das tecnologias de informação e comunicação nos domicílios brasileiros**. 2024. Site Cetic.br. Disponível em: <https://cetic.br/pt/pesquisa/domicilios/>. Acesso em: 06 abr. 2025.
- CLOUDFLARE, I. **Cloudflare Radar**. 2024. Site Cloudflare Radar. Disponível em: <https://radar.cloudflare.com/pt-br#protocols>. Acesso em: 10 abr. 2025.
- CLOUDFLARE, I. **Cloudflare Radar 2024 Year**. 2024. Site Cloudflare Radar 2024. Disponível em: <https://radar.cloudflare.com/year-in-review/2024>. Acesso em: 06 abr. 2025.
- COMMUNICATIONS, S. **Building the Next-Generation Web with HTTP/3**. 2023. Spirent Blog. Disponível em: <https://www.spirent.com/blogs/building-next-generation-web-with-http-3>. Acesso em: 25 maio 2025.
- DYCODERS. **The Role of Website Speed in User Experience**. 2023. Blog Dycoders. Disponível em: <https://dycoders.com/blog/the-role-of-website-speed-in-user-experience/>. Acesso em: 16 abr. 2025.
- FIELDING, R. T.; NOTTINGHAM, M.; RESCHKE, J. **HTTP Semantics**. RFC Editor, 2022. RFC 9110. (Request for Comments, 9110). Disponível em: <https://www.rfc-editor.org/info/rfc9110>.
- FOROUZAN, B.; FEGAN, S. **Protocolo TCP/IP - 3.ed.** McGraw Hill Brasil, 2009. ISBN 9788563308689. Disponível em: <https://books.google.com.br/books?id=fNvlgp3kkyQC>. Acesso em: 16 abr. 2025.
- FOROUZAN, B.; MOSHARRAF, F. **Redes de Computadores: Uma Abordagem Top-Down**. AMGH, 2013. ISBN 9788580551693. Disponível em: <https://books.google.com.br/books?id=57BIAgAAQBAJ>. Acesso em: 16 abr. 2025.
- LIU, F. *et al.* Performance comparison of http/3 and http/2: Proxy vs. non-proxy environments. **arXiv preprint arXiv:2409.16267**, 2024. Disponível em: <https://arxiv.org/abs/2409.16267>.
- NIELSEN, J. **Website Response Times: The 3 Important Limits**. 2010. Site Nielsen Norman Group. Disponível em: <https://www.nngroup.com/articles/response-times-3-important-limits/>. Acesso em: 16 abr. 2025.
- Pew Research Center. **Internet, Broadband Fact Sheet**. 2024. Site Pew Research Center. Disponível em: <https://www.pewresearch.org/internet/fact-sheet/internet-broadband/>. Acesso em: 06 abr. 2025.

SMITH, J. E. H. **The Internet Is Not What You Think It Is: A History, a Philosophy, a Warning**. Princeton University Press, 2022. ISBN 9780691212326. Disponível em: <https://books.google.com.br/books?id=6Z1EEAAAQBAJ>. Acesso em: 06 abr. 2025.

TREVISAN, M. *et al.* Measuring http/3: Adoption and performance. *In*: **2021 19th Mediterranean Communication and Computer Networking Conference (MedComNet)**. IEEE, 2021. p. 1–8. Disponível em: <https://ieeexplore.ieee.org/document/9501274>.