

- **Nome do Serviço 1:** GET /tasklists
- **Tipo de operações:** leitura
- **Arquivos envolvidos:** server/src/routes/tasklists.ts
<https://github.com/CiroJrGit/taskr/blob/main/server/src/routes/tasklists.ts>
- **Arquivos com o código fonte de medição do SLA:**
load-testing/scripts/get-taskr-results.json
[https://github.com/CiroJrGit/taskr/blob/main/load-testing/scripts/get-taskr-results.j
son](https://github.com/CiroJrGit/taskr/blob/main/load-testing/scripts/get-taskr-results.json)
- **Data da medição:** 18/06/2023
- **Descrição das configurações:** Máquina pessoal: CPU: Ryzen 5 5600, RAM: 16gb / MongoDB Atlas: Storage: 512MB, RAM: shared, vCPU: shared
- **Testes de carga (SLA):**

1 - Métricas Gerais:

Total de requisições HTTP: 700

Taxa de requisições HTTP por segundo: 351.767

Taxa de envio de dados: 31,307.263 bytes por segundo

Taxa de recebimento de dados: 119,249.012 bytes por segundo

2 - Duração das Requisições:

Duração média das requisições HTTP: 2.804 segundos

Duração mínima das requisições HTTP: 0 segundos

Duração mediana das requisições HTTP: 2.609 segundos

Duração máxima das requisições HTTP: 7.016 segundos

Percentil 90 da duração das requisições HTTP: 5.004 segundos

Percentil 95 da duração das requisições HTTP: 5.130 segundos

3 - Tempo de Espera das Requisições:

Tempo de espera médio: 2.773 segundos

Tempo de espera mínimo: 0 segundos

Tempo de espera mediano: 2.565 segundos

Tempo de espera máximo: 7.016 segundos

Percentil 90 do tempo de espera: 5.004 segundos

Percentil 95 do tempo de espera: 5.123 segundos

4 - Dados Enviados e Recebidos:

Total de dados enviados: 62,300 bytes

Taxa de dados enviados: 31,307.263 bytes por segundo

Total de dados recebidos: 237,300 bytes

Taxa de dados recebidos: 119,249.012 bytes por segundo

5 - Requisições Bloqueadas:

Tempo de bloqueio médio: 0.006 segundos

Tempo de bloqueio mínimo: 0 segundos

Tempo de bloqueio mediano: 0 segundos

Tempo de bloqueio máximo: 3.518 segundos

Percentil 90 do tempo de bloqueio: 0 segundos

Percentil 95 do tempo de bloqueio: 0 segundos

6 - Outras Métricas:

Número total de iterações: 700

Taxa de iterações por segundo: 351.767

Número de falhas nas requisições HTTP: 0



Legenda: Gráficos do teste de carga do Serviço 1. Gerado via Grafana Dashboard.

- **Potenciais gargalos do sistema**

O tempo médio das requisições HTTP é de 2.804 segundos, com duração máxima de 7.016 segundos. Isso sugere que o sistema pode estar enfrentando atrasos no processamento das requisições ou na comunicação com recursos necessários.

A taxa de envio e recebimento de dados mostra que o sistema está enviando dados a uma taxa de 31.307.263 bytes por segundo e recebendo dados a uma taxa de 119.249.012 bytes por segundo. Essas taxas podem indicar possíveis gargalos na largura de banda ou na capacidade de processamento para lidar com a quantidade de dados transferidos.

O tempo médio de bloqueio pode ocorrer devido a recursos compartilhados, concorrência ou problemas de sincronização. Esses bloqueios podem afetar negativamente o desempenho do sistema.

Conclusões:

O desempenho das requisições HTTP obtidas indica que as requisições têm uma duração considerável e podem impactar o desempenho geral do sistema. Em relação ao tempo de espera, em média, as requisições levam algum tempo para serem atendidas, o que pode afetar a experiência do usuário. Otimizar o tempo de espera para garantir uma melhor performance é importante para sistemas.

Para os dados enviados e recebidos, as informações geradas são relevantes para entender a carga de dados do sistema e podem ser usadas para otimizar a transmissão de dados, se necessário.

Já para as requisições bloqueadas, o relatório indica que algumas requisições podem ter sido bloqueadas por um curto período de tempo. É importante investigar as razões por trás desses bloqueios e tomar medidas para minimizá-los, garantindo um fluxo suave das requisições.

O relatório indica que não houve falhas nas requisições HTTP, com um total de 700 requisições bem-sucedidas. Isso é uma boa notícia, pois mostra que o sistema está lidando bem com as requisições sem apresentar erros.

- **Nome do Serviço 2:** POST /tasklists, { title: 'title-test' });
- **Tipo de operações:** inserção
- **Arquivos envolvidos:** server/src/routes/tasklists.ts
<https://github.com/CiroJrGit/taskr/blob/main/server/src/routes/tasklists.ts>
- **Arquivos com o código fonte de medição do SLA:**
load-testing/scripts/post-taskr-results.json
<https://github.com/CiroJrGit/taskr/blob/main/load-testing/scripts/post-taskr-results.json>
- **Data da medição:** 18/06/2023
- **Descrição das configurações:** Máquina pessoal: CPU: Ryzen 5 5600, RAM: 16gb / MongoDB Atlas: Storage: 512MB, RAM: shared, vCPU: shared
- **Testes de carga (SLA):**
 - 1 - Métricas Gerais:
Total de requisições HTTP: 700
Taxa de requisições HTTP por segundo: 5584.07
Taxa de dados enviados: 977211.43 bytes por segundo
Taxa de dados recebidos: 2049351.97 bytes por segundo
Total de requisições com falha: 0
 - 2 - Duração das Requisições:
Duração mínima da requisição: 0 segundos
Duração média da requisição: 0.120 segundos
Duração mediana da requisição: 0 segundos
Duração máxima da requisição: 1.5037 segundos
Percentil 90 da duração das requisições HTTP: 0.51411 segundos
Percentil 95 da duração das requisições HTTP: 0.52372 segundos
 - 3 - Tempo de Espera das Requisições:
Tempo de espera mínimo: 0 segundos
Tempo de espera médio: 0.09053 segundos
Tempo de espera mediano: 0 segundos
Tempo de espera máximo: 1.5037 segundos
Percentil 90 do tempo de espera: 0.51221 segundos
Percentil 95 do tempo de espera: 0.51511 segundos
 - 4 - Dados Enviados e Recebidos:
Total de dados enviados: 122500 bytes
Taxa de dados enviados: 977211.43 bytes por segundo
Total de dados recebidos: 256900 bytes
Taxa de dados recebidos: 2049351.97 bytes por segundo

5 - Requisições Bloqueadas:

Tempo de bloqueio mínimo: 0 segundos

Tempo de bloqueio médio: 0.00359 segundos

Tempo de bloqueio mediano: 0 segundos

Tempo de bloqueio máximo: 1.9999 segundos

Percentil 90 do tempo de bloqueio: 0 segundos

Percentil 95 do tempo de bloqueio: 0 segundos

6 - Outras Métricas:

Tempo médio de uma iteração: 0.17023 segundos

Duração mínima de uma iteração: 0 segundos

Duração mediana de uma iteração: 0 segundos

Duração máxima de uma iteração: 4.1448 segundos

90º percentil da duração de uma iteração: 0.52514 segundos

95º percentil da duração de uma iteração: 0.58131 segundos



Legenda: Gráficos do teste de carga do Serviço 2. Gerado via Grafana Dashboard.

- **Potenciais gargalos do sistema**

Para o tempo médio de espera, embora relativamente curto, existem requisições com tempos de espera mais longos. Isso pode indicar problemas no processamento das requisições ou problemas de escalabilidade do sistema.

Já em relação a taxa de envio e recebimento de dados, se as taxas estiverem próximas ou excederem a capacidade do sistema, isso pode resultar em gargalos de rede ou problemas de largura de banda. É importante garantir que a infraestrutura de rede seja dimensionada adequadamente para lidar com o volume de dados esperado, e embora o tempo médio de bloqueio seja baixo, é importante monitorar qualquer aumento significativo nessa métrica. Tempos de bloqueio mais

longos podem indicar problemas de concorrência de recursos ou bloqueios prolongados em determinadas operações.

Conclusões:

Os testes sugerem que o sistema analisado está lidando com um volume considerável de requisições HTTP, com bom desempenho geral. O tempo de resposta das requisições é curto, e o sistema não apresenta falhas ou bloqueios significativos. No entanto, é importante observar a variação no desempenho de iterações, com algumas delas sendo mais lentas que outras. Isso pode indicar possíveis áreas de otimização ou investigação adicional para melhorar a consistência e a velocidade do sistema, sendo algumas delas listadas abaixo:

Para o tempo de espera das requisições, é importante avaliar a capacidade do sistema de processar um maior volume de requisições, identificando possíveis gargalos e otimizando o código ou a infraestrutura para lidar com a carga.

Implementar estratégias de escalonamento horizontal, distribuindo as requisições em vários servidores pode equilibrar a carga e reduzir o tempo de espera.

Utilizar caches para armazenar dados frequentemente acessados, reduzindo a necessidade de realizar requisições repetidas e também reduzir o tempo de espera.

Otimizar as partes do código identificadas como sendo mais lentas, utilizando técnicas como refatoração, otimização de algoritmos ou paralelismo, dependendo da natureza do problema.

Implementar um monitoramento contínuo do desempenho do sistema para identificar padrões ou tendências e atuar proativamente na otimização do sistema.

Implementar estratégias de concorrência adequadas, como o uso de semáforos, threads ou processos separados para evitar bloqueios desnecessários.

Otimizar o acesso a recursos compartilhados, minimizando a necessidade de bloqueio, por exemplo, através da utilização de acesso concorrente.