

# Relatório Final

Alunos:

- Gabriel Victor
- João Victor Guerra
- Luiz Gustavo Santos

## Metodologia

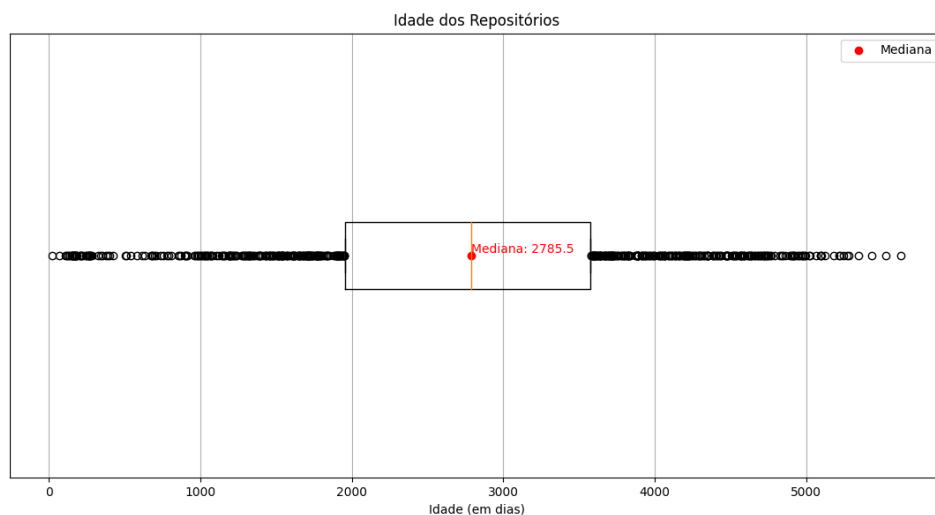
Com o objetivo de analisar as principais características dos repositórios públicos mais populares do GitHub, o grupo conduziu o estudo utilizando a API GraphQL disponibilizada pela plataforma, assim como a linguagem Python para obter, processar e mostrar os dados obtidos no formato de gráficos, com o intuito de tornar a visualização das respostas para cada pergunta mais intuitiva e simples.

De todo modo, os gráficos foram montados desconsiderando a ocorrência de valores vazios, isto é, na linguagem Python, rotulados como “None”. Tal abordagem foi adotada pensando no resultado final, na simplicidade de estruturação do código e na baixa ocorrência de valores desse tipo obtidos.

## Resultados Obtidos

**RQ 01.** Sistemas populares são maduros/antigos?

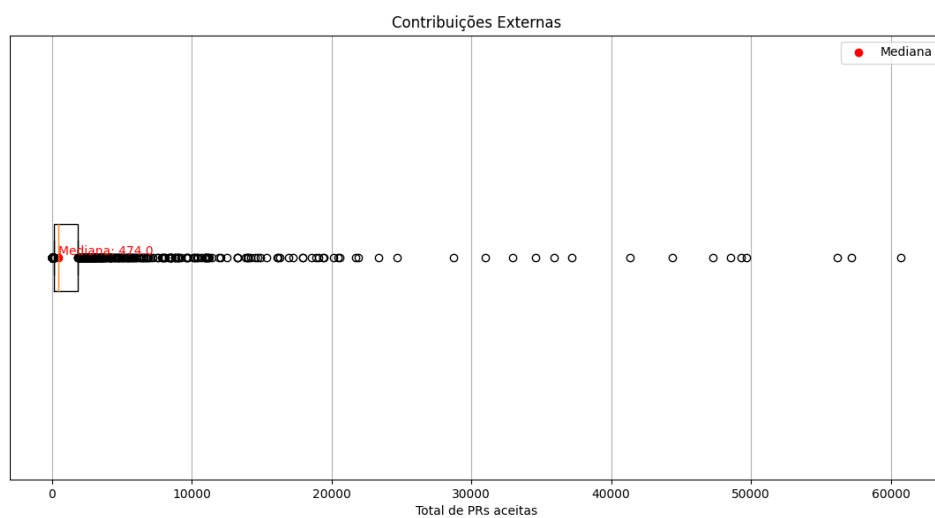
R:



Existe uma variedade grande de sistemas populares cuja maioria existe em uma janela de tempo de 1000 dias e 5000 dias, cuja tendência é ter a maioria dos repositórios populares próximos à mediana de 2785,5 dias (quase 7 anos). Isto nos leva a poder considerar que sistemas populares tendem a ser, de certo modo, antigos.

**RQ 02.** Sistemas populares recebem muita contribuição externa?

R:



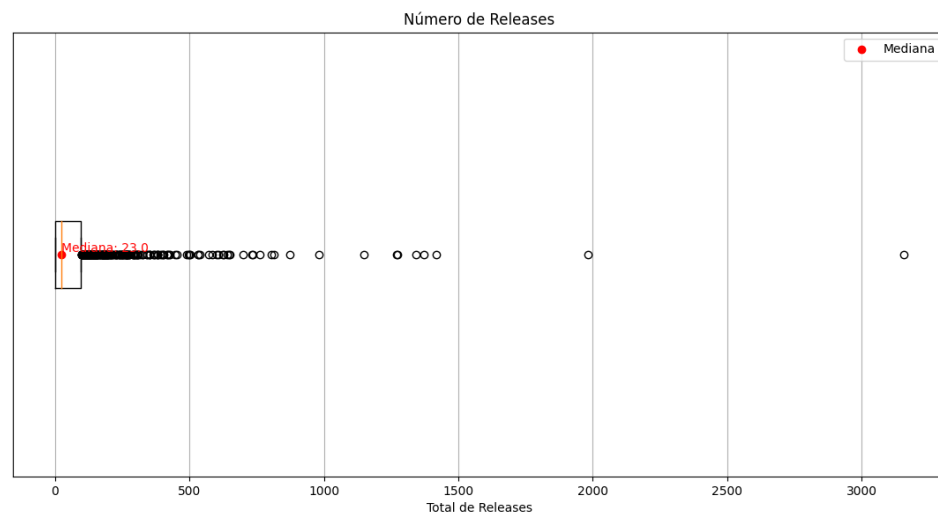
A partir do gráfico acima, é possível constatar que sistemas populares recebem um grande número de contribuições externas. A mediana se encontra em 474 contribuições aceitas, e é necessário salientar que não são compreendidas todas as contribuições que um projeto recebe – apenas as aprovadas.

Vários projetos excederam 1000 pull requests aceitas, e um superou 60000.

Se essa métrica for combinada às idades dos repositórios e sua correspondente mediana, nota-se o quão elevado é o número de contribuições e o incentivo à contribuição de usuários externos ao projeto.

**RQ 03.** Sistemas populares lançam releases com frequência?

R:

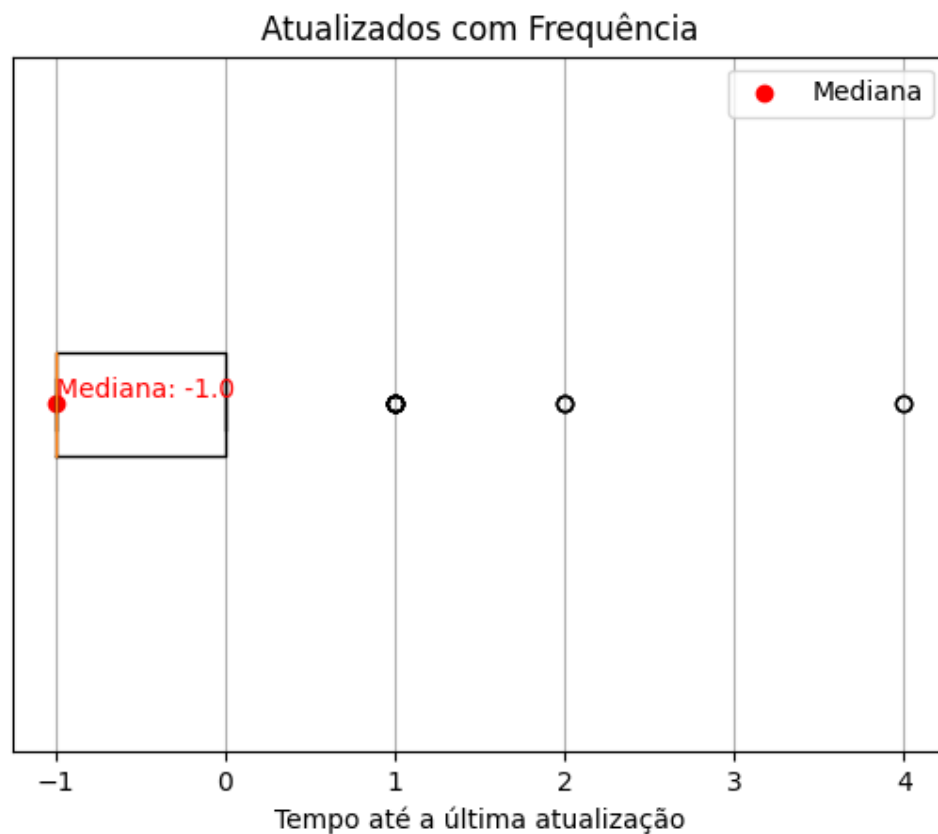


Adotando-se a métrica de total de releases, é possível perceber que os sistemas mais populares tendem a lançar releases, mas não tão frequentemente. Com a mediana de 23 releases, e considerando também as idades dos repositórios, é seguro dizer que é um tempo comum e satisfatório para um projeto popular.

Há ainda extremos que devem ser destacados, como projetos que não apresentaram nenhuma release e um projeto que possui mais de 3000 releases – até o momento da coleta de dados.

**RQ 04.** Sistemas populares são atualizados com frequência?

R:

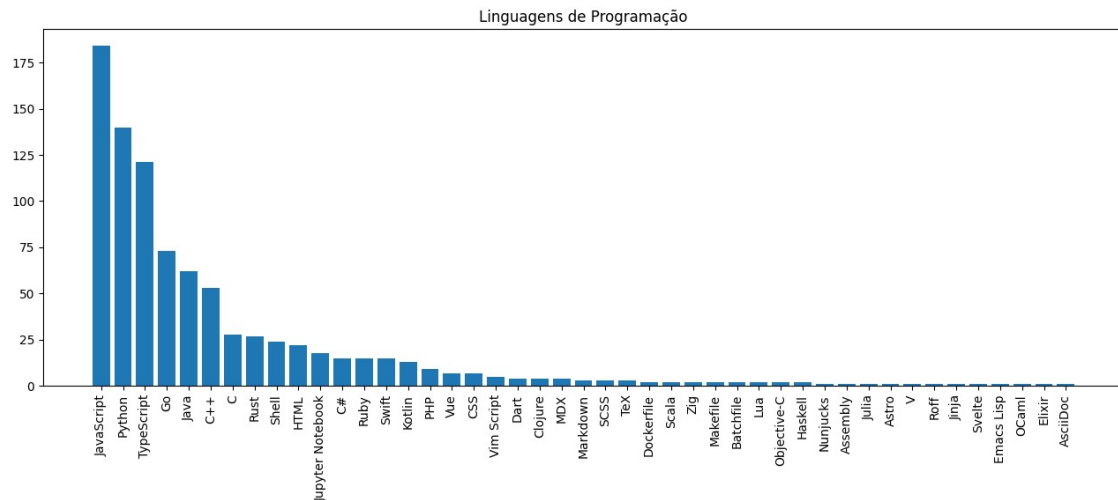


No gráfico e na consequente coleta de dados, valores menores que 0 indicam que o repositório foi atualizado no mesmo dia da coleta.

Dessa forma, através dos dados coletados, pode-se concluir que os repositórios atualizam com frequência, visto que a maioria deles apresentou atualizações no mesmo dia, com mediana de -1.

**RQ 05.** Sistemas populares são escritos nas linguagens mais populares?

R:



O gráfico mostra a incidência de linguagens de programação nos repositórios mais populares, considerando apenas a linguagem mais popular, e não um amálgama delas. JavaScript, Python e TypeScript figuram nas mais populares, consequentemente com mais adesão.

Para classificar a popularidade das linguagens de programação, utilizou-se o índice TIOBE (<https://www.tiobe.com/tiobe-index/>) e seu relatório das linguagens mais populares referente ao mês de agosto de 2023.

1	1			Python
2	2			C
3	4	▲		C++
4	3	▼		Java
5	5			C#
6	8	▲		JavaScript
7	6	▼		Visual Basic
8	9	▲		SQL
9	7	▼		Assembly language
10	10			PHP

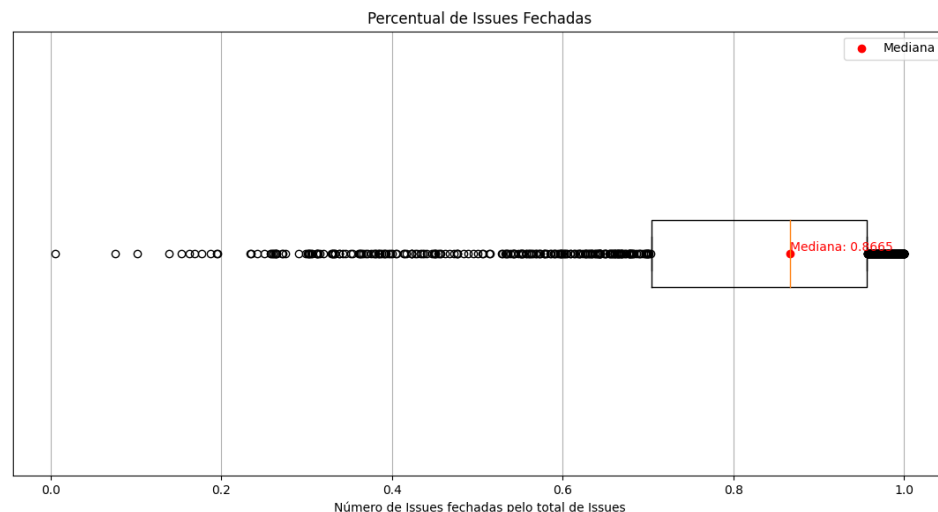
Correlacionando os dados obtidos no gráfico e na tabela, é possível afirmar que sim, sistemas populares tendem a ser escritos nas linguagens mais populares.

Vale ressaltar que a linguagem Go, que aparece como a quarta linguagem com mais incidência no gráfico, embora não esteja entre as dez primeiras do índice, se encontra na

posição 12, com perspectivas de ascensão – deve se considerar também o fato de ser uma linguagem nascida recentemente e com uma comunidade ainda a se estabelecer. O mesmo vale para a linguagem Rust, que figura na posição 19.

**RQ 06.** Sistemas populares possuem um alto percentual de issues fechadas?

R:



Nesse gráfico, pode-se perceber claramente que sistemas populares, em geral, possuem um alto percentil de issues fechadas, visto que a mediana se encontra em 86%. Isso pode indicar que projetos populares são alvos de maior engajamento e contribuição de usuários, tanto no caráter de interessados no projeto quanto de facilitadores e contribuidores. Ainda pode ser levado em conta a organização do repositório.

## Hipóteses

**RQ 01.** Sistemas populares são maduros/antigos?

Hipótese: Talvez, os repositórios mais antigos sejam relativamente mais populares.

R: Sim, nos dados obtidos, a mediana da idade dos sistemas populares se encontra em 7 anos, um tempo considerável para softwares e projetos. Pela adesão e por tempo de experiência, esses projetos conseguem permanecer relevantes por mais tempo.

**RQ 02.** Sistemas populares recebem muita contribuição externa?

Hipótese: A contribuição externa de sistemas populares pode ser maior do que em sistemas recentes, porém diluída ao longo de anos.

R: Embora não haja dados sobre o número de contribuições em sistemas recentes, é seguro afirmar que sistemas populares são os maiores alvos de contribuição externa, e que

o volume dessas contribuições podem colaborar para um sistema se tornar popular. Dessa forma, sistemas populares também recebem muita contribuição externa.

### **RQ 03. Sistemas populares lançam releases com frequência?**

Hipótese: Sistemas populares devem sim lançar releases com frequência, caso sejam recentes.

Hipótese: Repositórios mais antigos talvez sejam conservadores, lançando versões mais estáveis entre longos períodos.

R: É possível inferir que sistemas populares mais recentes lançam frequentemente mais releases, com o fim de alcançar versões estáveis e evoluir um projeto em início e ascensão. Repositórios mais antigos tendem a lançar releases de tempo em tempo, focando na estabilidade.

### **RQ 04. Sistemas populares são atualizados com frequência?**

Hipótese: Sistemas populares provavelmente recebem contribuições semanais ou diárias.

R: Sim, a grande parte dos sistemas populares receberam contribuições diárias, demonstrando seu engajamento com usuários e comunidade desenvolvedora. Por serem sistemas populares, pode-se inferir que recebem contribuições pois são aderidos pela comunidade que contribui com o projeto.

### **RQ 05. Sistemas populares são escritos nas linguagens mais populares?**

Hipótese: Esperamos que seja um ciclo retroativo onde sistemas populares criam demandas para uso das linguagens, que por sua vez passam a ser a próxima escolha para a criação de novos sistemas.

R: É possível perceber que tal hipótese se confirma, principalmente com relação a linguagens recentes como Go e Rust, adotadas por sistemas grandes e populares, que consequentemente influenciaram outros sistemas em seu desenvolvimento ou criação. De qualquer forma, é uma tendência que sistemas populares sejam escritos nas linguagens mais populares.

### **RQ 06. Sistemas populares possuem um alto percentual de issues fechadas?**

Hipótese: Sistemas populares muito provavelmente têm muitas issues fechadas, por receberem bastante atenção de suas comunidades.

R: Sim, percebe-se que a grande maioria dos sistemas populares possuem um alto percentil de issues fechadas, pois são alvos de atenção na comunidade, que procura solucionar dúvidas sobre o projeto ou contribuir com o sistema. Para suscitar uma comunidade engajada e organizar o projeto, é necessário que issues sejam resolvidas.