Análise de repositórios populares do Github

Alfredo Luis Vieira, Bruno Evangelista Gomes, Vinicius Salles

27 de agosto de 2025

1 Contextualização

Neste trabalho realizaremos a análise de repositórios populares do github, mais precisamente, os 1000 repositórios mais populares, esta análise visa identificar características destes repositórios, como número de issues fechadas, pull requests, releases e etc, e a partir destas métricas obtidas, levantar hipóteses acerca destes repositórios, identificar padrões, e analisar correlações entre os mesmos.

Este estudo é realizado a partir da realidade diversa dos repositórios situados no github, onde características como linguagem principal, contribuição, tamanho, total de estrelas e etc são muito variadas.

2 Hipóteses

A partir do conhecimento existente antes da execução da análise, podemos inferir algumas hipóteses/perguntas informais, que serão testadas, e reformuladas baseando-se na realidade identificada no estudo, sendo elas:

- Sistemas populares são maduros/antigos? (Métrica: idade do repositório, calculado a partir da data de sua criação).
- Sistemas populares recebem muita contribuição externa? (Métrica: total de pull requests aceitas).
- Sistemas populares lançam releases com frequência? (Métrica: total de releases).
- Sistemas populares são atualizados com frequência? (Métrica: tempo até a última atualização, calculado a partir da data de última atualização).
- Sistemas populares são escritos nas linguagens mais populares?(Métrica: linguagem primária de cada um desses repositórios).
- Sistemas populares possuem um alto percentual de issues fechadas?(Métrica: razão entre número de issues fechadas pelo total de issues).

3 Metodologia

Neste estudo foi aplicada a metodologia baseada na mineração de dados de repositórios do github, mais precisamente, os 1000 repositórios mais populares da plataforma, essa mineração foi realizada a partir de requisições junto à API GraphQL do próprio Github. Nessa API, buscamos as infomações relevantes para o estudo proposto. Em questões mais técnicas, utilizamos a Python como linguagem principal, utilizando bibliotecas da linguagem como o pandas. Vale citar tentativas de obtenção dos dados em massa junto a API, o que resultou em falha, pois a API foi implementada para retornar um erro após certo tempo de processamento de requisição, o que fez com que a consulta a mesma seja paginada, realizando um processamento em lote dos dados obtidos.

Para obter os resultados, foi realizada uma análise quantitativa dos dados, se baseando em dados estatísticos como média, mediana, razão entre valores etc, estes resultados serão analisados mais pra frente.

4 Resultados obtidos

Nesta seção exibiremos os resultados obtidos, e uma breve discussão sobre eles, após a conclusão destas reflexões, uma reflexão geral será tratada na próxima seção.

4.1 Sistemas populares são maduros/antigos?

Nesta etapa da análise, observamos uma grande variedade de idades nos repositorios, abaixo, listaremos os repositórios mais jovens e mais velhos da análise:

4.1.1 Repositórios mais velhos

- rails/rails 208.4
- git/git 205.06
- jekyll/jekyll 202.12
- redis/redis 197.12
- jquery/jquery 196.70

4.1.2 Repositórios mais novos

- OpenCut-app/OpenCut 2
- google-gemini/gemini-cli 4.10
- openai/codex 4.27
- FoundationAgents/OpenManus 5.49
- x1xhlol/system-prompts-and-models-of-ai-tools 5.52

Nesta analise, podemos identificar uma média de 96.9 meses nestes repositórios, o que pode se considerar uma idade avançada, podemos assim, afirmar que sistemas populares são SIM maduros/antigos. Podemos tambem observar uma grande variedade de valores em questão de idade do repositório, o que nos leva a crer que repositórios mais famosos não possuem um intervalo muito constante.

Abaixo tambem possuímos um gráfico que lista a relação entre os repositorios considerados pelo nosso estudo Maduro ou Jovem:

Figura 1: Relação entre niveis de maturidade dos repositórios

4.2 Sistemas populares recebem muita contribuição externa?

Agora, devemos analisar se os sistemas analisados possuem um nível alto de contribuição externa, ou seja, se os membros da comunidade realmente tendem a contribuir mais com sistemas populares, para fazer isso, calcularemos a média dos top 50 repositorios mais populares, e compararemos, coma média dos 50 repositórios observados com menos popularidade.

Podemos observar uma média de pull requests de 5371.90 nos 50 repositórios mais populares da lista, e uma média de 1741.23 pull requests nos 50 repositórios menos populares, ou seja, podemos dizer que a média aumentou em torno de 208%, ou seja, podemos identificar um grande aumento e afirmar que sim, repositórios populares tendem a ter contribuições externas.

Também elaboramos dois gráficos para melhor visualização destes dados, podemos também encontrar uma grande massa de repositórios com até 3 mil pull requests aceitas, e no segundo gráfico, podemos encontrar repositórios com altos números de contribuições, ultrapassando 50 mil.

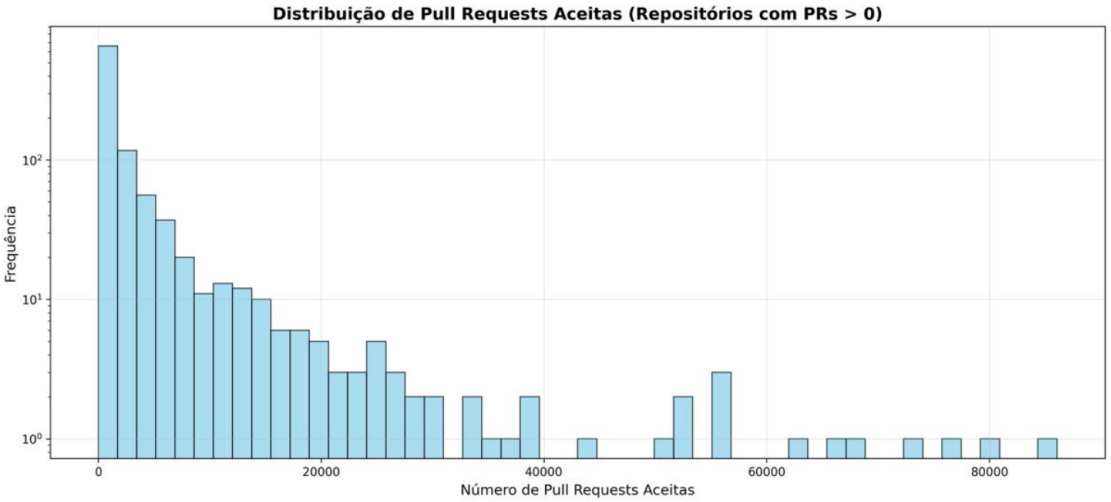

Figura 2: Distribuição de pull requests aceitas

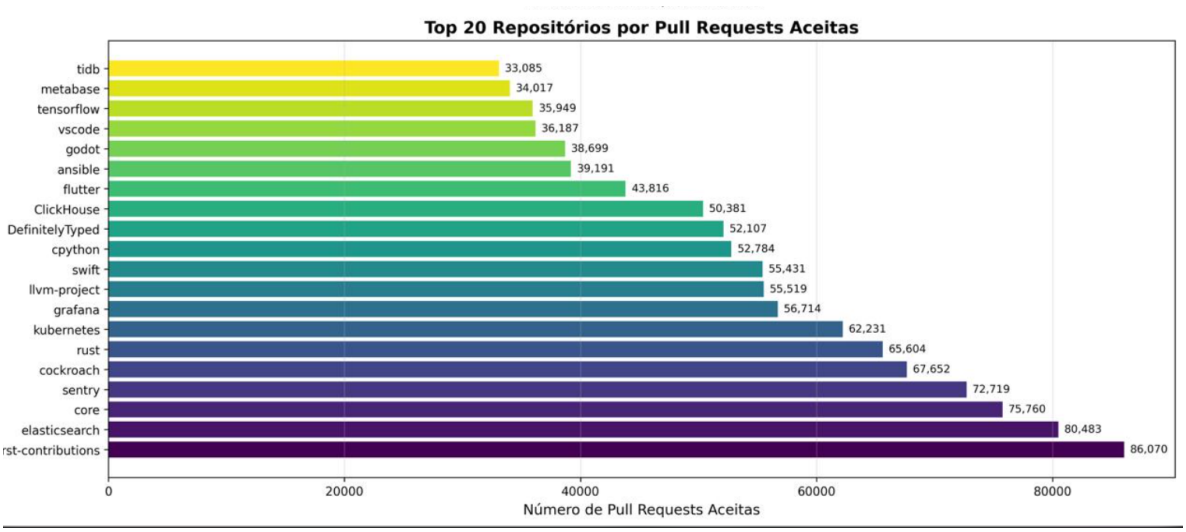

Figura 3: Top 20 repositórios com pull requests aceitas

4.3 Sistemas populares lançam releases com frequência?

Para resolver essa questão, tentaremos entender o total de releases de cada repositório, e o que esses dados em conjunto tem a dizer, inicialmente, devemos entender que 692 dos 1000 repositórios analisados possuiam releases, o que equivale a uma taxa de 69,2%, o que é uma taxa consideravelmente ok.

Para entendermos melhor o contexto, cruzando com os dados relativos a linguagem principal, podemos observar uma coisa curiosa, que repositorios baseados em typeScript, Go e Rust possuem uma média maior de releases, linguagem estas que estão no top 15 de linguagens mais populares, de acordo com a pesquisa do stackoverflow de 2025, isso nos mostra que projetos nestas linguagens possuem uma alta taxa de liberação de releases, e um grande desenvolvimento da comunidade, como mostra o gráfico abaixo.

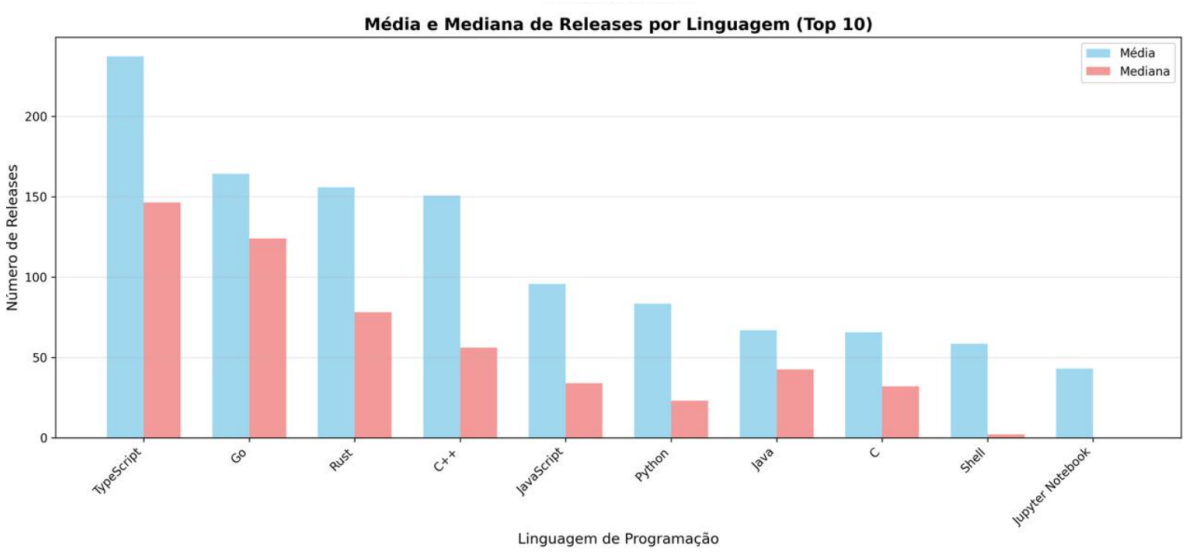

Figura 4: Média/mediana de releases por linguagem principal

4.4 Sistemas populares são atualizados com frequência?

Com essa questão, queremos entender se os repositórios possuem atualizações recentes, o que pode nos dar indícios de que sua atualização é constante. Após realizar cálculos, a partir de um boxplot, encontramos dados interessantes; podemos visualizar que a média de atualização em dias, de repositórios jovens e maduros é bem parecida, girando em torno de 5 a 7 dias, o que nos mostra que a maturidade de um repositório não influencia em sua frequência de atualizações. Visto isso, podemos entender que repositórios populares como um todo tendem a ser atualizados rapidamente, devido ao grande nível de engajamento de usuários; como visto anteriormente neste estudo, a imagem abaixo nos mostra visualmente.

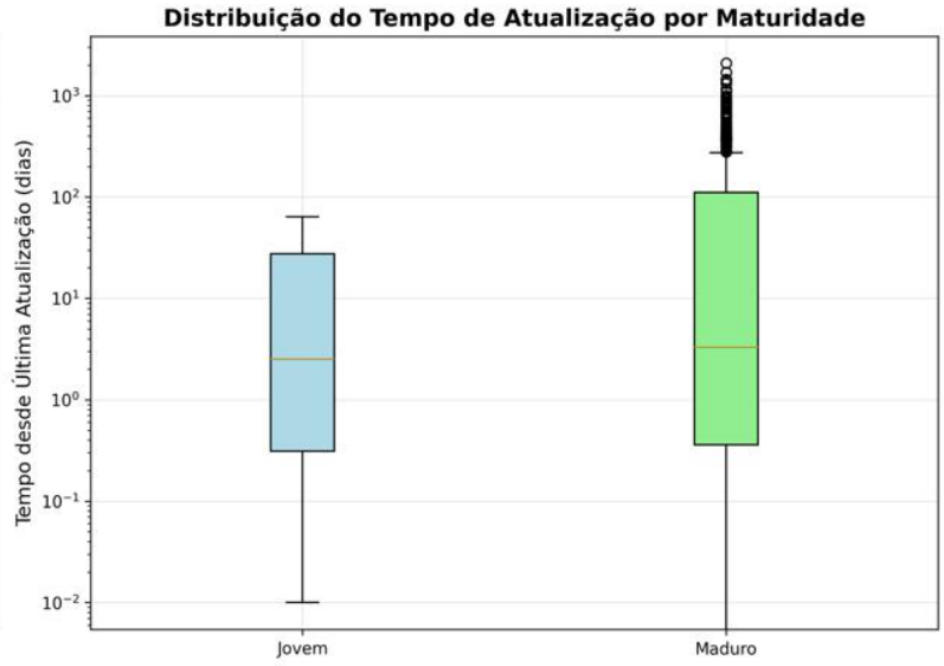

Figura 5: BoxPlot de taxa de atualização em dias

4.5 Sistemas populares são escritos nas linguagens mais populares?

Neste estudo, encontramos repositórios com diversas linguagens principais; segue abaixo um gráfico que torna mais amigável esta visualização:

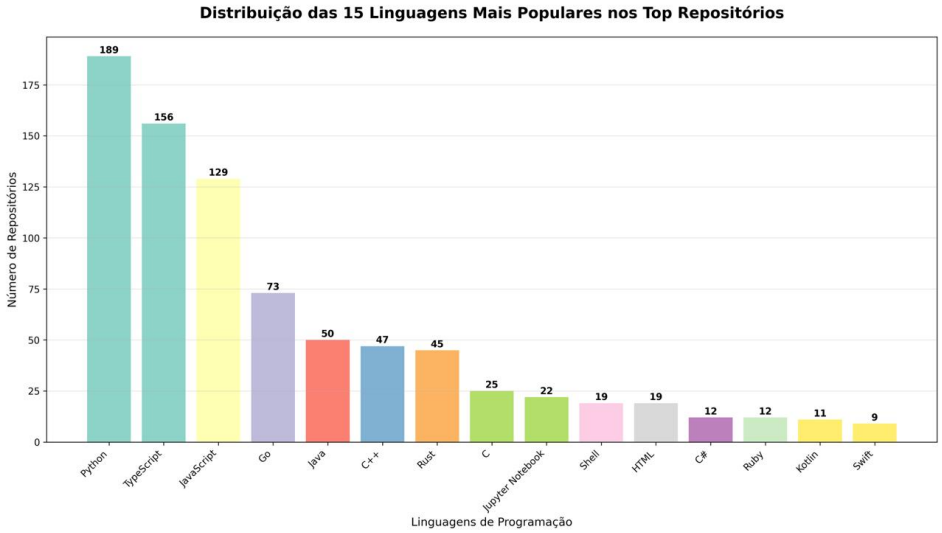

Figura 6: Linguagens mais presentes nos repositórios minerados

Seguindo a pesquisa do stackoverflow do ano de 2025 (cujo os resultados estão inseridos no gráfico abaixo), correlacionando com os dados encontrados nos repositórios, podemos identificar uma certa semelhança entre os lideres dos ranks em ambas as medições, como por exemplo, o Javascript, que ficou em terceiro lugar nos repositórios que listamos, e que aparece em primeiro na pesquisa realizada em 2025, outro exemplo é o python, que está em primeiro nos levantamentos deste estudo, e em quarto na pesquisa realizada em 2025.

Podemos entender um certo padrão quando falamos de linguagens de programação utilizadas, principalmente quando falamos de tecnologias já consolidadas (como Javascript, Python e Java), podemos afirmar que existe sim uma tendência de que as pesquisas relacionadas às linguagens favoritas sigam caminhos semelhantes às linguagens principais dos repositórios mais populares.

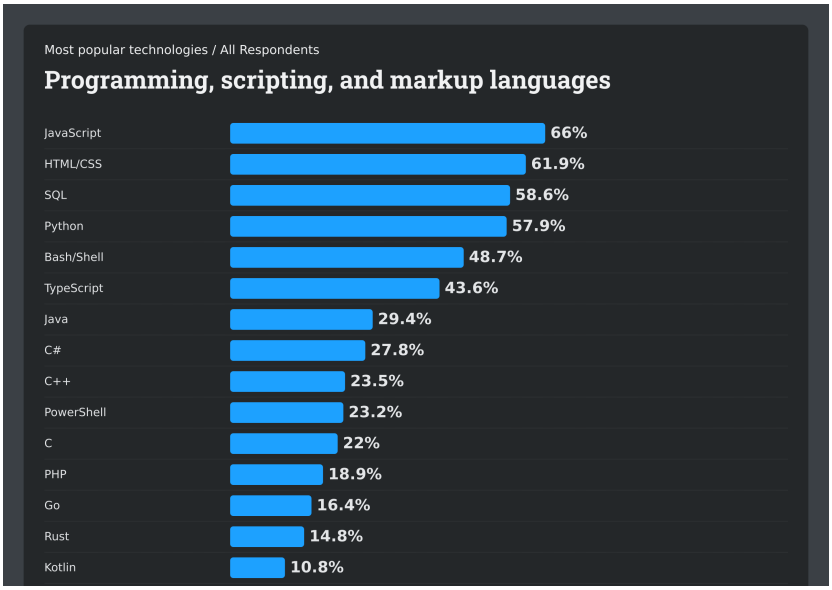

Figura 7: Top 15 linguagens da pesquisa do stackoverflow de 2025

4.6 Sistemas populares possuem um alto percentual de issues fechadas?

Nestes estudos, encontramos que os sistemas possuem uma média de 79% de suas issues fechadas, o que pode indicar que seus bugs, problemas relacionados à segurança ou qualquer outra coisa desta natureza, tendem a ser resolvidos. O gráfico abaixo apresenta uma visualização destes dados.

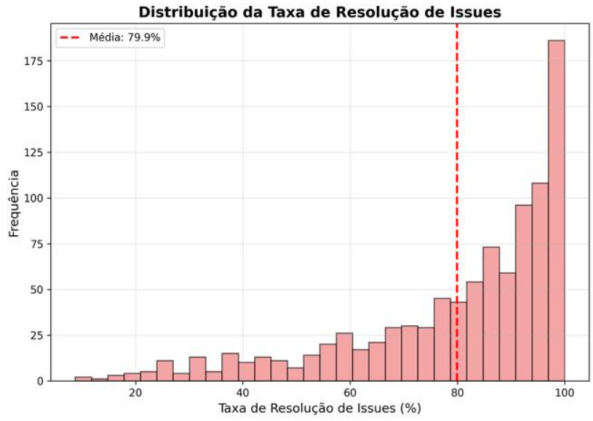

Figura 8: Taxa de resolução de issues

Abaixo temos um gráfico que mostra a relação do total de issues fechadas e sua taxa de resolução com a idade do repositório. Podemos entender que a maioria dos repositórios com idade mais avançada possui uma porcentagem maior de issues fechadas, mas que em repositórios mais novos a idade não é um fator influente, pois possuem valores distribuídos ao longo do espectro do gráfico. Isso pode mostrar um crescimento na contribuição de resolução de issues ao longo dos anos.

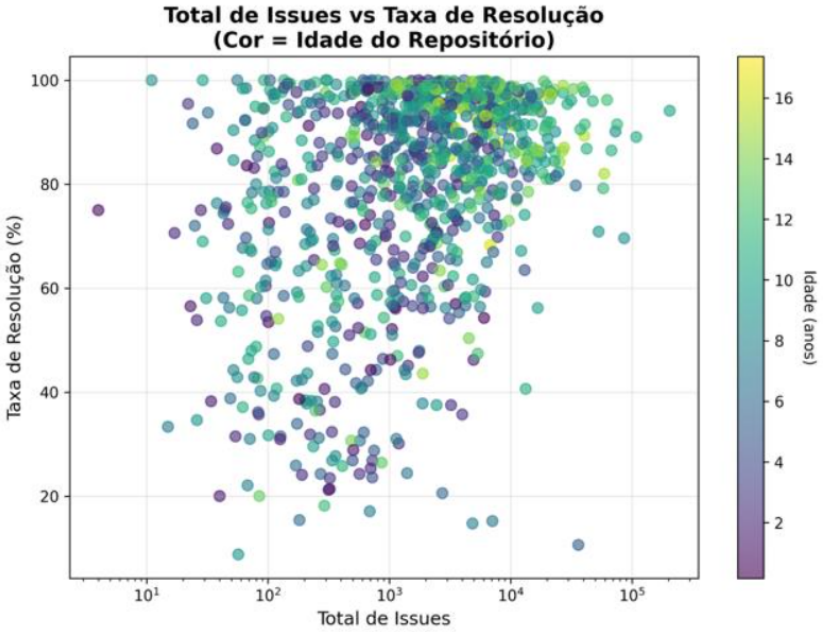

Figura 9: Numero de issues e suas taxas resolução de x idade do repositório

5 Considerações finais

Com base na análise quantitativa dos 1000 repositórios mais populares do GitHub, é possível traçar um perfil claro do que caracteriza um projeto de sucesso na plataforma. A investigação confirmou a

maioria das hipóteses iniciais, revelando que a popularidade está intrinsecamente ligada à maturidade e à atividade contínua. Os resultados demonstram que, em média, os repositórios populares são projetos estabelecidos, com uma idade média de quase 97 meses, indicando que a confiança e a base de usuários são construídas ao longo do tempo. Essa maturidade, no entanto, não é sinônimo de estagnação. Pelo contrário, o estudo revelou uma frequência de atualização notavelmente alta, com uma média de apenas 5 a 7 dias desde a última modificação, um padrão consistente tanto em projetos jovens quanto nos mais antigos.

Essa vitalidade é impulsionada por uma forte contribuição da comunidade, como evidenciado pela diferença expressiva no número de pull requests aceitos entre os repositórios mais e menos populares do nosso conjunto de dados, onde os mais populares recebem, em média, 208% mais contribuições. Além do engajamento externo, a manutenção interna também se mostra um pilar fundamental, com uma taxa média de 79% de issues resolvidas, sugerindo que os problemas são ativamente gerenciados e corrigidos. O alinhamento tecnológico com as tendências do mercado também se provou um fator relevante, com as linguagens de programação predominantes nos repositórios analisados, como Python e Javascript, espelhando as mais populares em pesquisas da indústria, como a do Stack Overflow. Portanto, conclui-se que um repositório popular no GitHub é tipicamente um ecossistema dinâmico: um projeto maduro, construído sobre uma tecnologia relevante, e que é constantemente aprimorado por uma comunidade engajada e uma equipe de manutenção responsiva.