Características de Repositórios Populares

Lucca V. P. Bessa[1] , Samuel R. Freitas[1]

[1]Bacharelado em Engenharia de Software
Instituto de Ciências Exatas e Informática - PUC Minas
Ed. Fernanda. Rua Cláudio Manoel, 1.162, Funcionários
Belo Horizonte – MG – Brasil

Introdução

O GitHub é uma plataforma de hospedagem de código-fonte e controle de versão baseada na Web. Ele permite que os desenvolvedores armazenem e gerenciem o código-fonte de seus projetos em repositórios online, onde podem colaborar com outros desenvolvedores e obter feedback da comunidade de desenvolvimento.

Esta ferramenta é especialmente popular entre os desenvolvedores de software de código aberto, que usam a plataforma para compartilhar seus projetos e receber contribuições da comunidade. Ela permite que os desenvolvedores contribuam com o código-fonte para projetos existentes e bifurquem projetos para desenvolver suas próprias versões.

O GitHub também oferece uma variedade de ferramentas e recursos para desenvolvedores, incluindo rastreamento de problemas, integração contínua, hospedagem de sites, gerenciamento de projetos e muito mais. Além disso, o é conhecido por sua comunidade ativa e engajada, fornecendo suporte e feedback para desenvolvedores em todo o mundo.

Será realizado uma análise dos repositórios presentes no GitHub com maior número de estrelas para identificar como eles são desenvolvidos, a frequência de contribuições externas, a frequência de lançamento de novas versões, bem como outras características relevantes. Serão coletados dados de 1.000 repositórios e os valores obtidos serão discutidos.

O desenvolvimento do sistema em geral está sujeito a diversas questões e hipóteses. Em relação à maturidade do sistema, hipotetiza-se que os sistemas mais comuns são maduros e antigos, pois os sistemas tendem a amadurecer quanto mais tempo estiverem em desenvolvimento.

Em relação às contribuições externas, uma hipótese é que quanto mais popular for um sistema, maior a probabilidade de incluir contribuições externas. A popularidade de um sistema pode ser medida por estrelas ou downloads do GitHub.

Outra hipótese é que os sistemas populares são lançados com frequência devido ao número crescente de funcionalidades. Novos recursos e correções de bugs são implementados regularmente e novas versões são lançadas com frequência.

Os sistemas populares são atualizados com frequência devido à sua grande base de usuários. Esses usuários exigem novos recursos e correções de bugs regularmente, portanto, os desenvolvedores precisam atualizar esses repositórios com mais frequência.

Por fim, levanta-se a hipótese de que os sistemas mais populares são escritos nas linguagens mais populares, facilitando a localização de colaboradores e pessoas interessadas em aprender esse idioma.

Metodologia

Para coletar informações sobre repositórios no GitHub, será utilizada a API GraphQL fornecida pela plataforma. Através dessa interface, serão obtidos uma grande quantidade de dados dos 1000 repositórios mais populares da plataforma, incluindo informações sobre o número de estrelas, releases, pull requests aceitos, data da última atualização, total de issue e issues fechadas, além das linguagens de programação utilizadas. Com a utilização da API GraphQL, será possível automatizar a coleta de dados e realizar uma análise mais abrangente de um grande número de repositórios, o que seria inviável de ser feito manualmente.

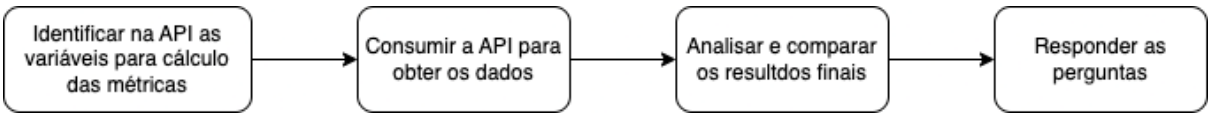

Figura 1. Fluxograma da metodologia

A metodologia utilizada neste estudo consiste em usar a API GraphQL do Github para recuperar dados de repositórios selecionados. Primeiramente, serão identificadas as variáveis necessárias para o cálculo dos indicadores pertinentes à questão de pesquisa proposta. Em seguida, a do GitHub API será usada para recuperar esses dados. Posteriormente, os dados serão analisados e comparados para responder às perguntas da pesquisa. O processo de análise utiliza técnicas estatísticas e ferramentas de visualização de dados para extrair informações relevantes e identificar padrões e tendências nos dados. Ao final do processo, será gerado um resultado final que possibilitará a resposta às questões de pesquisa propostas.

Resultados e análise das amostras

A partir dos questionamentos levantados no enunciado do estudo e após a aplicação da metodologia, foram desenvolvidos seis gráficos representativos para ilustrar os resultados. Os gráficos, inseridos abaixo, contemplam os seguintes dados: idade dos repositórios, total

de pull requests aceitas, total de releases, tempo até a última atualização, linguagem primária dos repositórios e a razão entre issues fechadas e total de issues.

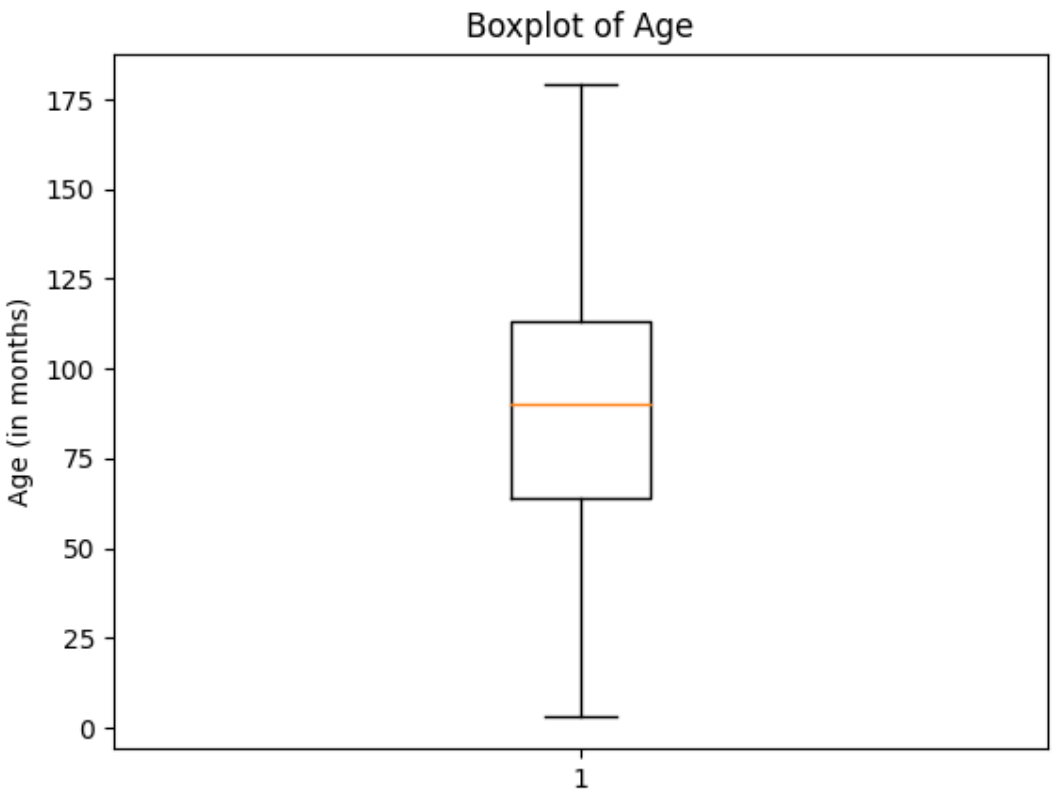

Gráfico 1. Idade

Com base na distribuição indicada no gráfico, os repositórios têm entre 5 e 10 anos, demonstrando que são softwares maduros.

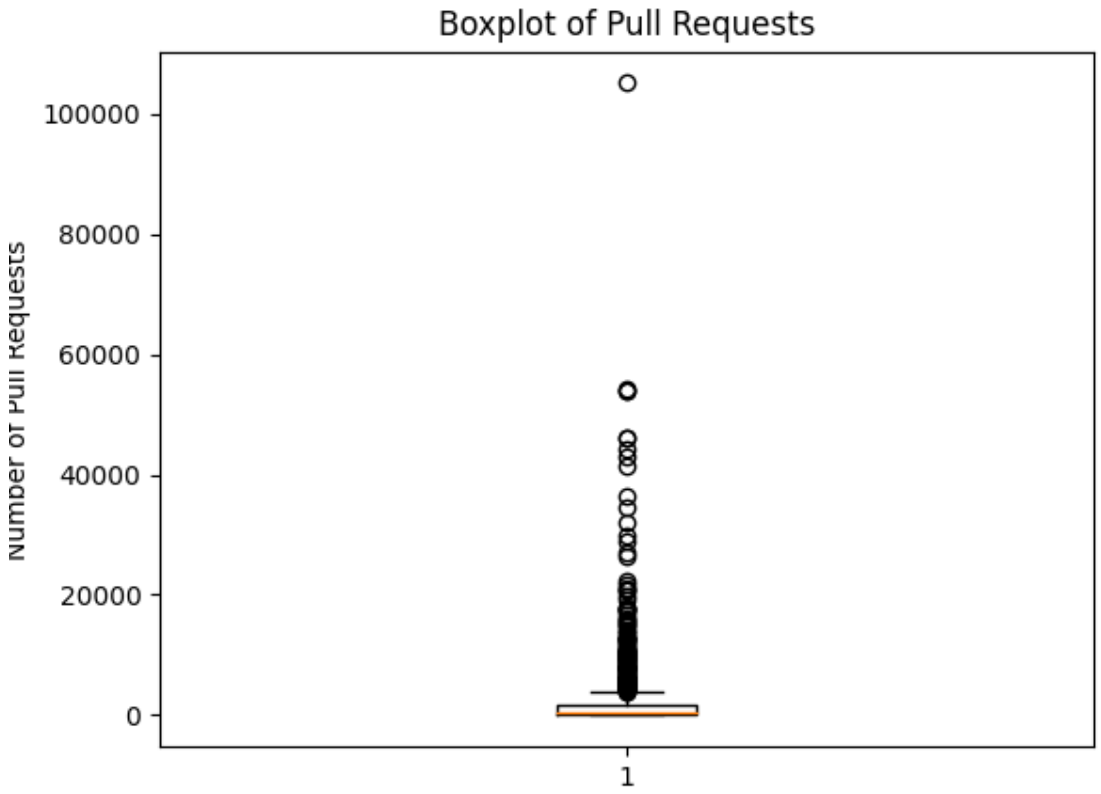

Gráfico 2. Total de Pull requests aceitas

Quanto ao total de pull requests, a distribuição no gráfico indica que a maioria dos repositórios possuem poucas contribuições externas, apresentando um volume considerável de outliers.

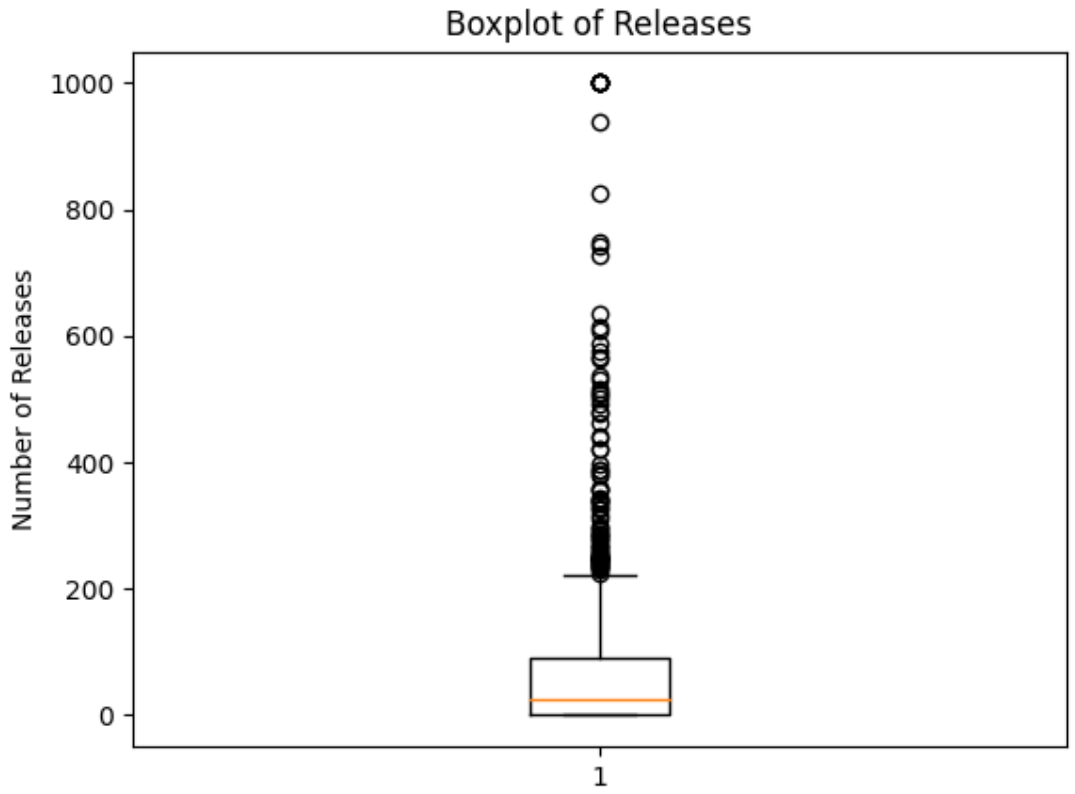

Gráfico 3. Total de Releases

A partir da análise do gráfico, conclui-se que os repositórios mais populares possuem um grande número de releases. Considerando que a idade e o formato de execução dos projetos, o alto número de releases é esperado.

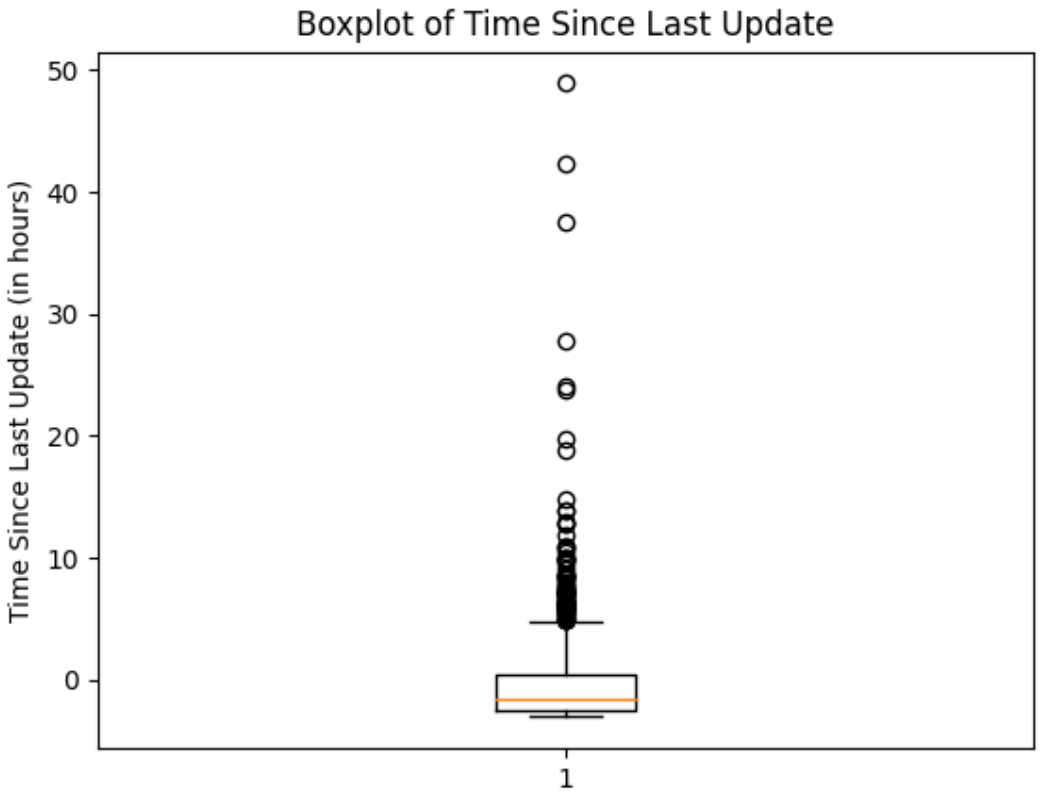

Gráfico 4. Tempo desde a última atualização

Baseado no gráfico apresentado, conclui-se que a maioria da distribuição está próximo de zero, demonstrando que os repositórios populares são atualizados com frequência.

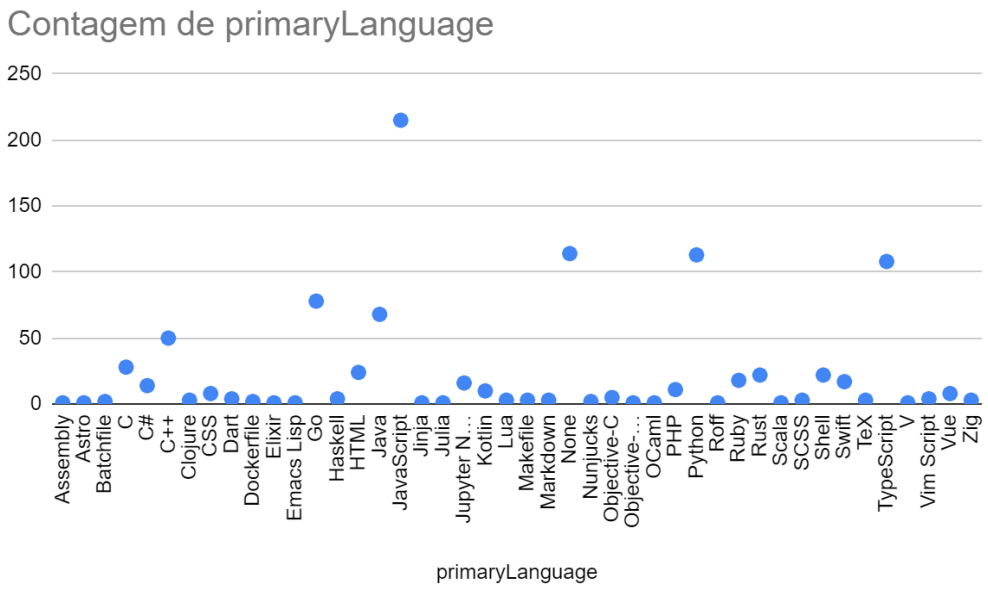

Gráfico 5. Linguagem primária dos repositórios

De acordo com uma pesquisa realizada pelo GitHub em 2022, a linguagem mais utilizada na plataforma é Java Script, seguida de Python. O gráfico aponta que esse mesmo cenário se repete neste estudo, considerando que a linguagem primária mais utilizada nos repositórios em questão é Java Script, em seguida, Python e TypeScript.

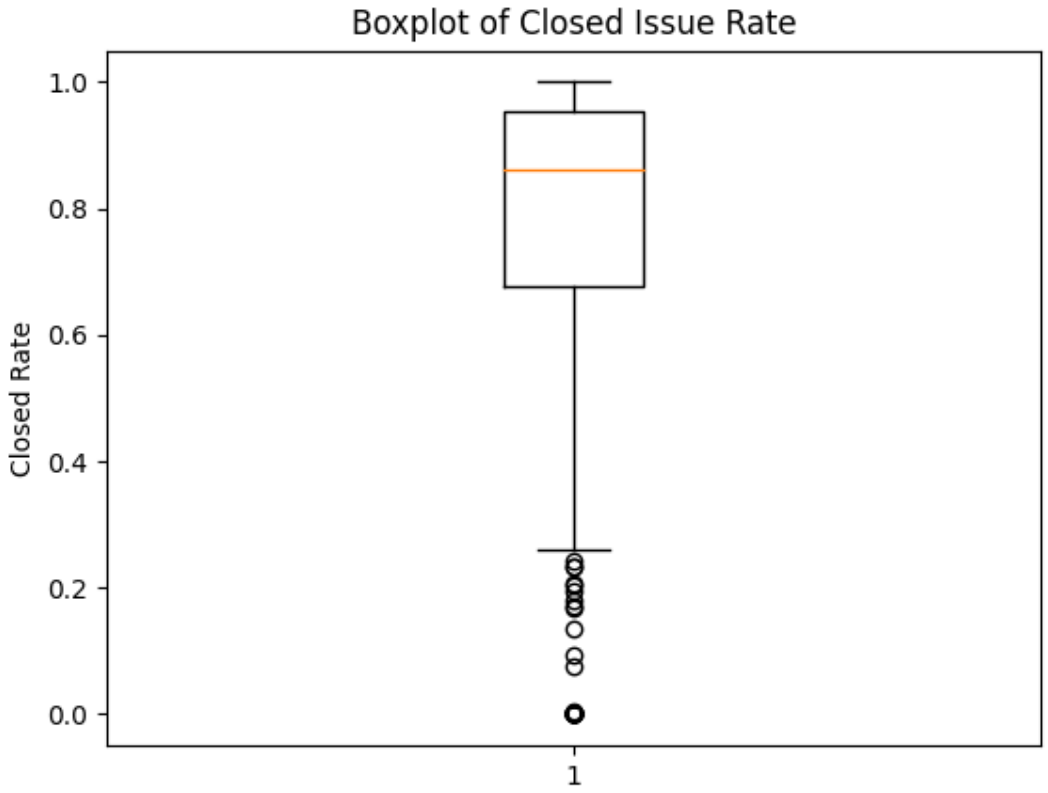

Gráfico 6. Taxa de fechamento de issues

Baseado na análise, é possível afirmar que o percentual de issues fechadas nos repositórios mais populares é alto. O limite inferior do primeiro quartil está acima de 60%, o que indica que a maioria das issues abertas são fechadas.

Discussão

Com base nos dados minerados do GitHub, pode-se concluir que as hipóteses levantadas em relação à maturidade dos sistemas, contribuições externas, frequência de lançamento e linguagens populares foram todas confirmadas como verdadeiras. Os sistemas mais comuns são realmente maduros e antigos, já que tendem a amadurecer com o tempo. Além disso, a popularidade de um sistema, medida por estrelas ou downloads do GitHub, está diretamente relacionada à probabilidade de incluir contribuições externas.

Os sistemas populares são lançados com frequência devido ao número crescente de funcionalidades e à necessidade de correções de bugs e implementação de novos recursos para atender às demandas dos usuários. Esses sistemas são atualizados com frequência devido à sua grande base de usuários, que exigem melhorias constantes. Finalmente, os sistemas mais populares são geralmente escritos nas linguagens mais populares, tornando-os mais acessíveis para colaboradores e pessoas interessadas em aprender a linguagem.

Essas conclusões indicam a importância de se concentrar no desenvolvimento de sistemas populares e, ao mesmo tempo, em manter a qualidade do código e a inovação para garantir a sustentabilidade e relevância no longo prazo. As empresas e desenvolvedores podem utilizar essas informações para tomar decisões estratégicas, como escolher a linguagem de programação mais adequada para o desenvolvimento de um sistema ou para determinar o ciclo de lançamento de novas versões.