PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS E INFORMÁTICA
UNIDADE EDUCACIONAL PRAÇA DA LIBERDADE
Bacharelado em Engenharia de Software

Belle Nerissa Aguiar Elizeu
Felipe Caldas Liduario
Letícia Amanda Franco Gonçalves

RELATÓRIO: CARACTERÍSTICAS DE REPOSITÓRIOS POPULARES

Belo Horizonte
2023

Belle Nerissa Aguiar Elizeu
Felipe Caldas Liduario
Letícia Amanda Franco Gonçalves

RELATÓRIO: CARACTERÍSTICAS DE REPOSITÓRIOS POPULARES

Relatório apresentado na Sprint 1 da disciplina Laboratório de Experimentação de Software.

Professor: José Laerte Pires Xavier

Belo Horizonte

2023

1 INTRODUÇÃO

O relatório apresentado visa analisar as características de relatórios populares do GitHub. Essa atividade foi proposta na disciplina de Laboratório de Experimentação de Software.

1.1 Hipóteses

RQ01. Sistemas populares são maduros/antigos?

Métrica: idade do repositório (calculado a partir da data de sua criação).

Hipótese: Sim, sistemas populares são mais antigos, ao terem mais tempo no GitHub para receber contribuições e para serem vistos por mais pessoas.

RQ02. Sistemas populares recebem muita contribuição externa?

Métrica: total de pull requests aceitas.

Hipótese: Sim, considerando que esses sistemas são normalmente escritos em linguagens muito conhecidas, existem inúmeras pessoas que entendem e que contribuem para a evolução desse sistema. A hipótese içada é de que repositórios populares tenham a quantidade de pull requests superior a 350.

RQ03. Sistemas populares lançam releases com frequência?

Métrica: total de releases.

Hipótese: Sim, devido ao grande número de pessoas contribuindo com os sistemas mais populares, estes lançam releases com uma frequência maior se comparado a alguns sistemas mais desconhecidos. A hipótese é de que tenha pelo menos 4 releases por ano em um repositório popular.

RQ04. Sistemas populares são atualizados com frequência?

Métrica: tempo até a última atualização (calculado a partir da data de última atualização).

Hipótese: Sim, devido à abundância de contribuições que esses sistemas recebem, as atualizações são feitas com uma frequência maior se compararmos com sistemas menos populares. Portanto, a hipótese é de que um repositório popular seja atualizado pelo menos 2 vezes por mês.

RQ05. Sistemas populares são escritos nas linguagens mais populares?

Métrica: linguagem primária de cada um desses repositórios

Hipótese: Sim, quanto mais popular e conhecida a linguagem for, mais pessoas conheceram o sistema e conseguirão interagir com pull requests e avaliações. Segundo o Tecnoblog, temos como hipóteses que as linguagens mais populares utilizadas são: JavaScript, Python, Java, TypeScript e C#.

RQ06. Sistemas populares possuem um alto percentual de issues fechadas?

Métrica: razão entre número de issues fechadas pelo total de issues

Hipótese: Sim, por se tratar de um sistema popular, existem mais pessoas trabalhando nele, portanto a chance de uma pessoa fechar um issue aberta nesse sistema é grande quando comparamos a um sistema não popular onde poucas pessoas realizam contribuições. A hipótese é de que tenha um percentual de pelo menos 80% das issues fechadas e resolvidas.

RQ 07: Sistemas escritos em linguagens mais populares recebem mais contribuição externa, lançam mais releases e são atualizados com mais frequência?

Métrica: Divisão dos resultados obtidos nas RQs 02, 03 e 04 por linguagem e análise de como esses valores se comportam conforme as linguagens de cada repositório.

Hipótese: Linguagens de programação populares possuem alto engajamento da sua comunidade, ou seja, repositórios escritos em linguagens populares podem obter mais contribuições dessas pessoas. Como referência para esta hipótese, assumimos que para cada linguagem a quantidade de pull requests aceita deve ser pelo menos metade dos pull requests totais, a quantidade de releases por ano deve ser pelo menos 10 e a atualização deve ser pelo menos mensal.

2 METODOLOGIA

Para realizar a análise das hipóteses, foi desenvolvido um script em Python com uma consulta em GraphQL que realiza consultas dos repositórios mais populares do Github e faz a exportação dos dados em um arquivo csv. O código-fonte do script de cálculo da geração do Csv, dos cálculos das métricas das RQs, o arquivo csv e demais itens estão contidos no repositório: https://github.com/BelleNerissa/lab6-T1.

Foram analisados 1.000 repositórios na data 04/03/2023 e calculadas as medianas por outro script em python que utiliza das bibliotecas pandas, para o cálculo das respostas das RQs conforme as métricas solicitadas, e matplotlib para a plotagem dos gráficos.

3 RESULTADOS OBTIDOS

Esta seção tem o objetivo de apresentar a apuração dos resultados executados no projeto, de modo a responder às perguntas levantadas.

RQ01: A mediana de idade dos repositórios é de 2733,5 dias, que correspondem a cerca de 7 anos e 5 meses e meio. A figura 1, apresenta um gráfico do tipo boxplot com a idade dos repositórios em dias.

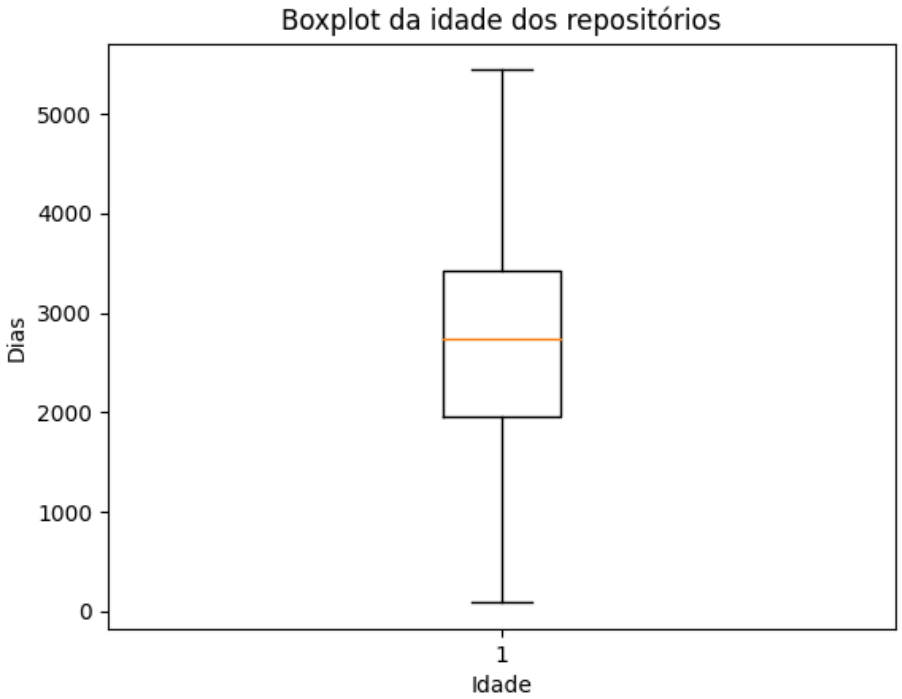

Figura 1 - Boxplot das idades dos repositórios

RQ02: Conforme apresentado na figura 2, a mediana do total de pull requests aceitas dos repositórios é de 466,5.

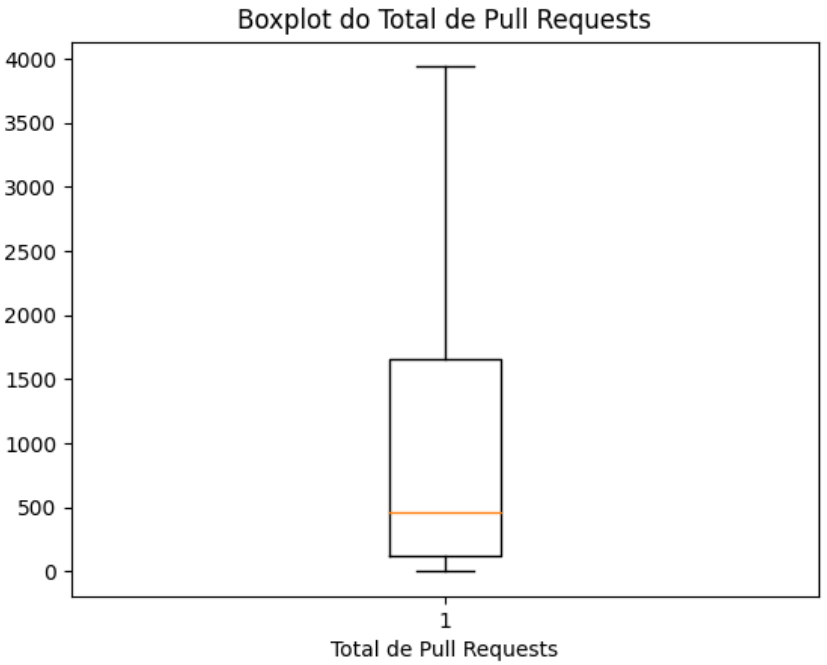

Figura 2 - Boxplot do total de pull requests aceitas dos repositórios

RQ03: A mediana de releases é 24,0. A mediana de quantidade de releases por ano segundo a mediana da idade dos repositórios é 24,0/ 7,41 ≈ 3,23886, representado na figura 3.

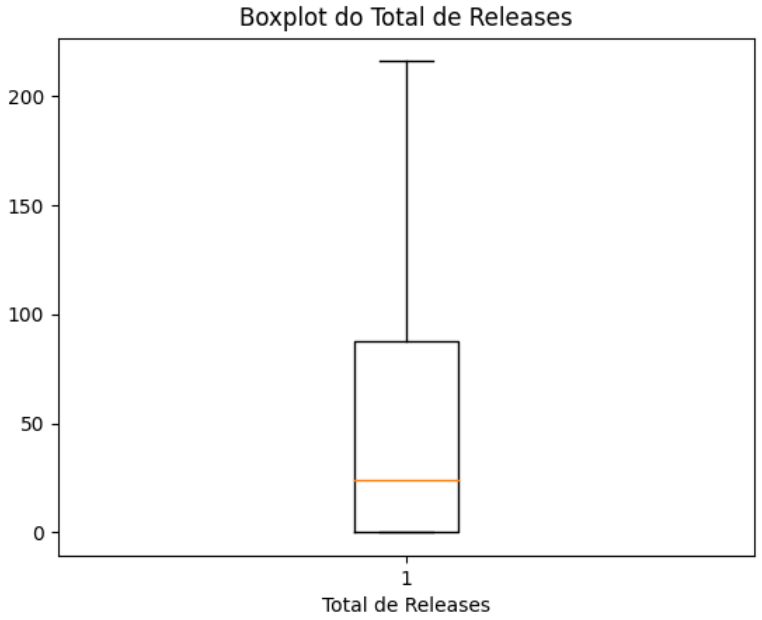

Figura 3 - Boxplot do Total de Releases dos repositórios

RQ04: A figura 4, apresentada abaixo, exibe a mediana do tempo desde a última atualização é de 8 dias. Portanto, os repositórios são atualizados frequentemente.

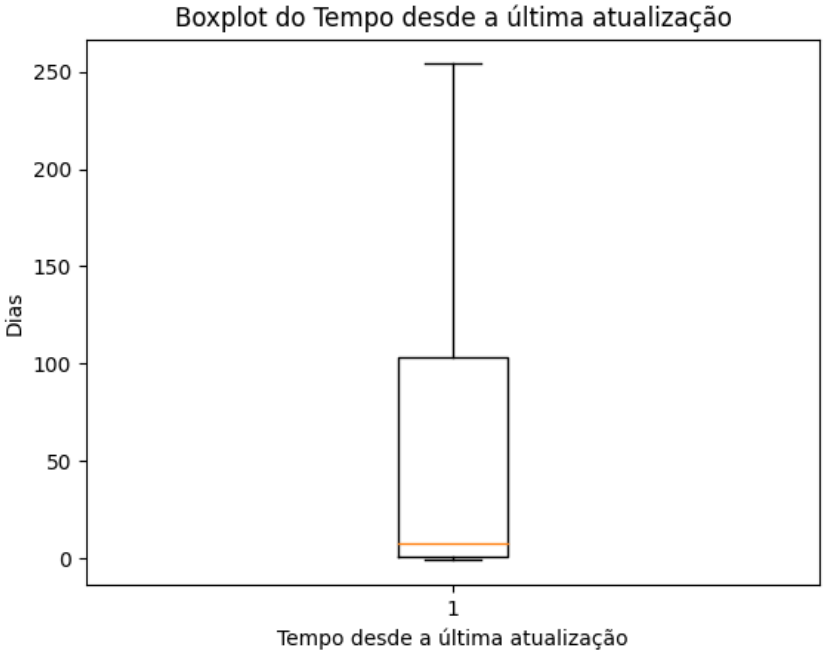

Figura 4 - Boxplot do Total de Tempo desde a última atualização dos repositórios

RQ05: As 5 linguagens que mais apareceram nos repositórios populares foram: Python, Typescript, Go, Java e C++. Na figura 5, é possível visualizar as linguagens mais utilizadas.

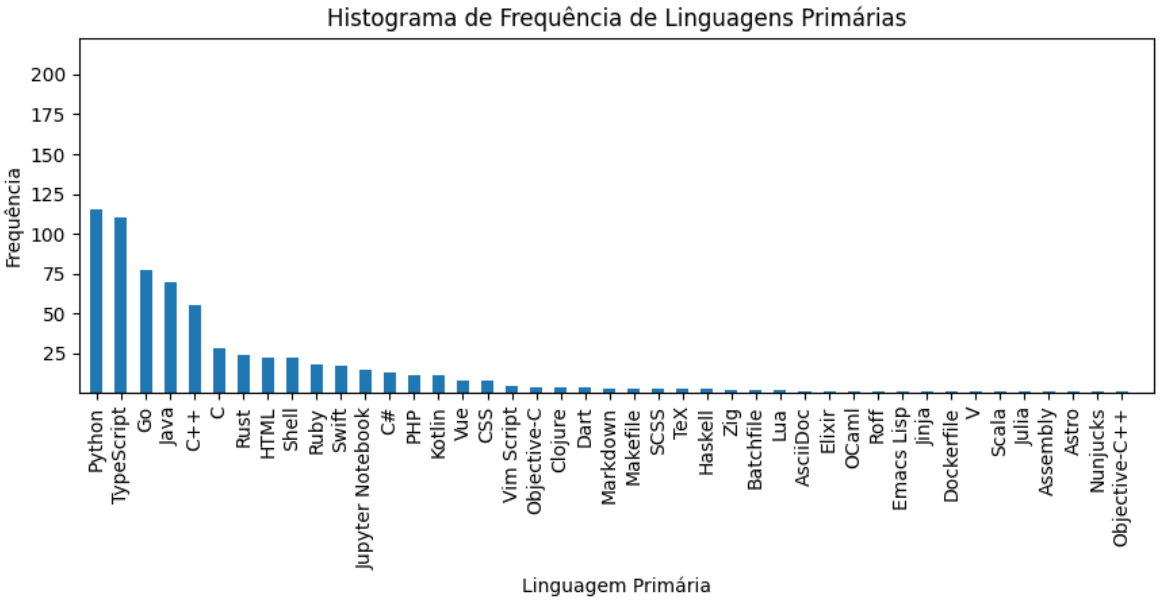

Figura 5 - Histograma com as linguagens mais populares

RQ06: A mediana da razão entre issues fechadas e o total é de ≈ 87%.

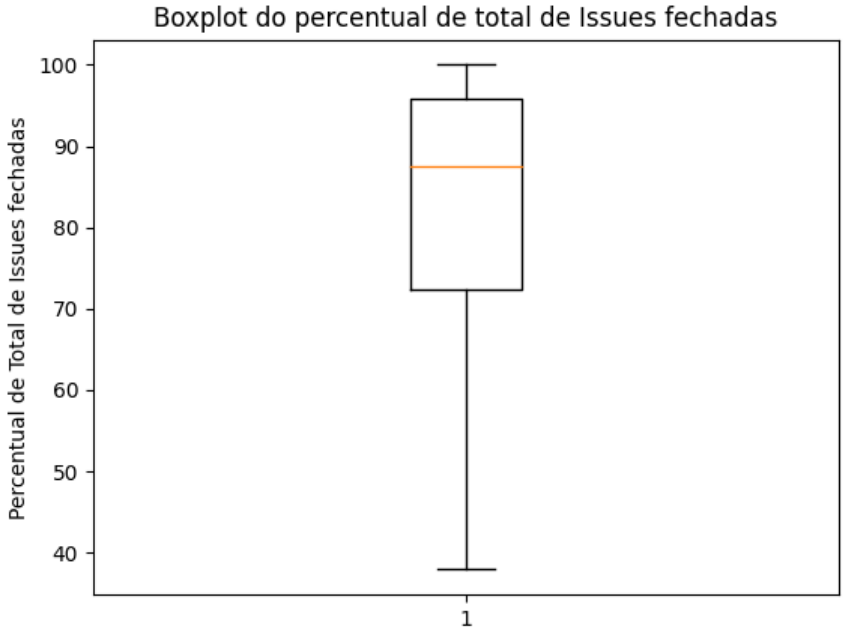

Figura 6 - Boxplot do percentual do total de issues Fechadas dos repositórios mais populares

RQ07: Resultados obtidos para cada uma das cinco principais linguagens dos repositórios analisados:

TypeScript
Mediana de **pull requests totais**: 2309,5
Mediana de **pull requests aceitas**: 1631,5
Mediana de **releases**: 106,5
Mediana de **atualizações**: 2 dias

Python
Mediana de **pull requests totais**: 779,0
Mediana de **pull requests aceitas**: 530,0
Mediana de **releases**: 8,0
Mediana de **atualizações**: 6 dias

Go
Mediana de **pull requests totais**: 1272,0
Mediana de **pull requests aceitas**: 905,0
Mediana de **releases**: 78,0
Mediana de **atualizações**: 3 dias

Java
Mediana de **pull requests totais**: 764,0
Mediana de **pull requests aceitas**: 398,5
Mediana de **releases**: 26,0
Mediana de **atualizações**: 5 dias

C++
Mediana de **pull requests totais**: 1008,0
Mediana de **pull requests aceitas**: 1087,0
Mediana de **releases**: 40,0
Mediana de **atualizações**: 2 dias

4 ANÁLISE DOS RESULTADOS

Nesta seção serão apresentadas as análises dos resultados adquiridos para cada RQ durante a pesquisa em relação às hipóteses inicialmente levantadas.

RQ01: A hipótese inicial foi confirmada, isso porque normalmente um repositório possui no máximo 14 anos, e um sistema considerado maduro tem aproximadamente a metade da idade no github, ou seja, 7 anos. O resultado apontou que a mediana de idade dos sistemas populares no Github é de **7 anos e 5 meses e meio**.

RQ02: A hipótese inicial foi confirmada, uma vez que o valor definido era de 350 pull requests e o resultado indica que a mediana de pull requests aceitas foram de **466,5**. Logo, podemos afirmar que sistemas populares recebem muitas contribuições externas.

RQ03: A hipótese inicial foi refutada, já que o resultado apresentado foi de **≈ 3,2386** e a hipótese era de pelo menos 4 releases por ano. Logo, podemos afirmar sistemas populares não necessariamente lançam releases com frequência.

RQ04: A hipótese inicial foi confirmada, pois a mediana de atualização no repositório é de pelo menos **8 dias**. Logo, podemos afirmar que sistemas populares são atualizados com frequência.

RQ05: A hipótese inicial foi parcialmente confirmada, isso porque três das cinco linguagens citadas foram apresentadas como as mais populares, sendo elas: Python, Typescript, e Java. Além disso, as linguagens Go e C++ que não foram levantadas na hipótese estão presente entre as mais populares. Entretanto, as linguagens JavaScript e C# que foram citadas como hipótese não estão entre as cinco mais populares. Logo, podemos afirmar que em sua maioria, sistemas populares são escritos nas linguagens populares.

RQ06: A hipótese foi confirmada, pois o resultado obtido é de aproximadamente **87%** e ultrapassa o valor de 80%, definido na hipótese. Logo, podemos afirmar que sistemas populares possuem alto percentual de issues fechadas.

RQ07: JavaScript: A linguagem não aparece entre as cinco populares mais utilizadas nos repositórios. Então a hipótese inicial foi refutada.

Python: A hipótese inicial foi refutada, apesar de ter a mediana dos pull requests aceitos acima da metade (530) em relação ao total (779,0), e a atualização ser durante o prazo de seis dias, não atingiu a meta de (10) releases anual, ficando apenas com (8).

Java: A hipótese inicial foi confirmada, pois o resultado obtido ultrapassa a metade (398,5) dos pull requests aceitos em relação ao total (764,0). Possui a mediana de (26) em releases, situando-se acima da hipótese definida, (10). A mediana alcançada em atualizações foi de (5) dias, enquanto a hipótese indicava ao menos uma vez no mês.

TypeScript: A hipótese inicial foi confirmada, pois o resultado obtido ultrapassa a metade (1631,5) dos pull requests aceitos em relação ao total (2309,5). Possui a mediana de (106,5) em releases, situando-se acima da hipótese definida, (10). A mediana alcançada em atualizações foi de (2) dias, enquanto a hipótese indicava ao menos uma vez no mês.

C#: A linguagem também não aparece entre as cinco populares mais utilizadas nos repositórios. Então a hipótese inicial foi refutada.

5 REFERÊNCIAS

Tecnoblog. **Aplicativos e Software**. Disponível em: www.tecnoblog.net/
Acesso em: 2 Mar. 2023.