PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS
Bacharelado em Engenharia de Software

REPOSITÓRIOS POPULARES

Arthur Bicalho Lana Corrêa Fernandes

Belo Horizonte, 2022

1. INTRODUÇÃO

Github é uma plataforma de hospedagem de código-fonte e arquivos com controle de versão usando o Git. Ele permite que programadores, utilitários ou qualquer usuário cadastrado na plataforma contribuam em projetos privados ou Open Source de qualquer lugar do mundo. GitHub é amplamente utilizado por programadores para divulgação de seus trabalhos ou para que outros programadores contribuam com o projeto, além de promover fácil comunicação através de recursos que relatam problemas ou mesclam repositórios remotos.

A plataforma acaba disponibilizando uma API para que seja possível extrair dados e métricas dos repositórios públicos. Facilitando assim para que seus usuários possam estudar as principais características dos repositórios mais populares. Utilizando essa API e as ferramentas disponibilizadas pelo GitHub, neste trabalho será analizado 1000 repositórios Open-Source com maior popularidade no site.

1.1 MÉTRICAS

Para facilitar o desenvolvimento deste trabalho foi se utilizado métricas para que fique claro e também ajudar a entender as características dos repositórios mais populares, sendo assim as métricas utilizadas foram:

RQ 01. Sistemas populares são maduros/antigos?

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

RQ 02. Sistemas populares recebem muita contribuição externa?

***Métrica**: total de pull requests aceitas*

RQ 03. Sistemas populares lançam releases com frequência?

***Métrica**: total de releases*

RQ 04. Sistemas populares são atualizados com frequência?

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

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

***Métrica**: linguagem primária de cada um desses repositórios*

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

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

2. METODOLOGIA

Para esta pesquisa foi se utilizado a linguagem de programação TypeScript, consumindo uma API do próprio GitHub, que acaba possibilitando a extração dos repositórios públicos mais populares.

Foi realizado a extração de menor ou igual a 100 repositórios e feito a sua automatização de dados, sendo esses repositórios os mais populares. Esses dados foram coletados e armazenados em um arquivo .csv, sendo possível identificar os dados detalhadamente.

Após a coleta dos resultados foi gerado tabelas feitas pelo próprio Google Docs, para que fosse possível ser feito a comparação dos dados e também responder cada uma das perguntas das métricas utilizadas.

3. RESULTADOS

Os resultados apresentados correspondem a cada uma das métricas utilizadas para esta pesquisa. Foram gerados tabelas para que facilitasse a visualização dos resultados.

3.1. RQ 01 - IDADE DOS REPOSITÓRIOS

Idade	Quantidade repositórios
0	4
1	25
2	45
3	90
4	100
5	115
6	140
7	130
8	150
9	80
10	75
11	65
12	30
13	11

Para essa tabela foram pegos repositórios dos anos 2009, até o ano 2022, dessa forma o repositório mais novo possui idade 0 (correspondendo a repositórios que ainda não completaram 1 ano completo) e o maior idade 13.

3.2. RQ 02 - TOTAL DE PULL REQUESTS ACEITAS

Métrica	Menor Valor	Maior Valor
Total de Pull Request	0	102679

Para alcançar estes resultados foi-se utilizado uma função mínima e máxima, para identificar a quantidade de pull requests realizadas até o dia da entrega desse trabalho.

3.3. RQ 03 - TOTAL DE RELEASES

Métrica	Menor Valor	Maior Valor
Total de releases	0	2352

Para alcançar estes resultados foi-se utilizado uma função mínima e máxima, para identificar a quantidade de releases realizadas até o dia da entrega desse trabalho.

3.4. RQ 04 - ÚLTIMA ATUALIZAÇÃO

Métrica	Menor Valor	Maior Valor
Tempo última atualização	120,200	214,700

Para alcançar estes resultados foi-se utilizado uma função mínima e máxima, para identificar a última atualização dos repositórios, desde o dia da entrega desse trabalho.

3.5 RQ 05 - LINGUAGEM PRIMÁRIA

Linguagem	Quantidade de Repositórios
JavaScript	227
Python	112
JAVA	69
TypeScript	102
C#	12
PHP	13
C++	54
Shell	22
C	29
Ruby	18

Para alcançar estes resultados foi extraído da API do GitHub as linguagens primárias e a quantidade de cada uma e os repositórios que elas estão envolvidas até o dia de entrega deste trabalho.

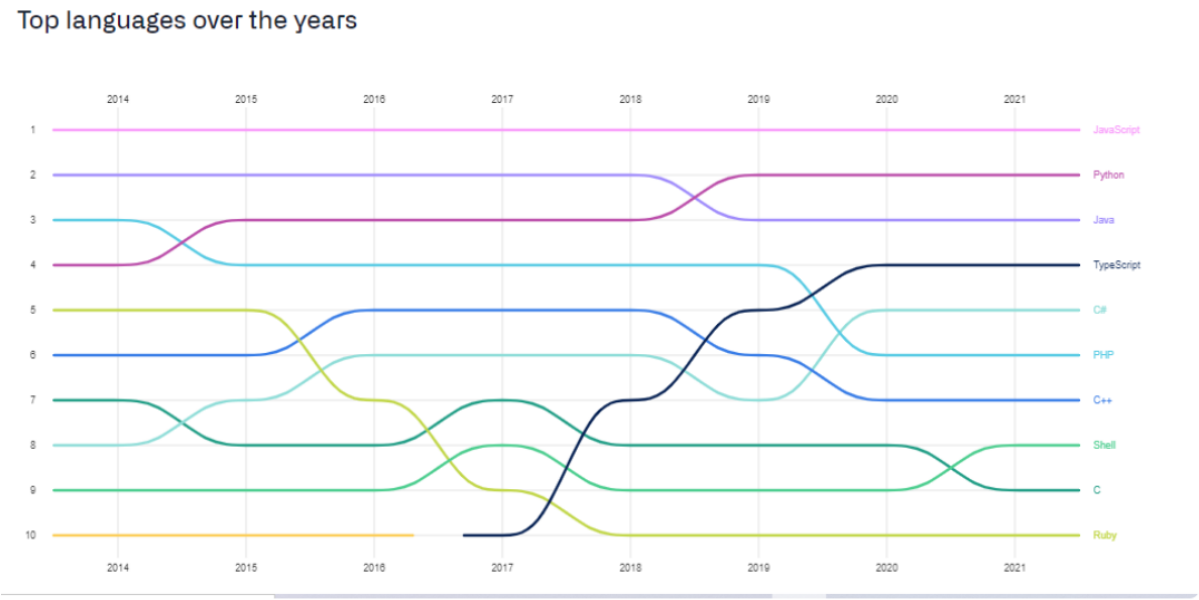

Imagem retirada do site oficial do GitHub para mostrar a evolução das principais linguagens de programação do site e mostrando que os dados estão parecidos com o da tabela apresentada.

3.6. RQ 06 - ISSUES FECHADAS E TOTAIS DE REPOSITÓRIOS

Métrica	Menor Valor	Maior Valor
Total de Issues Fechadas	**0**	**133178**
Total de Issues	**0**	**140350**

Para alcançar estes resultados foi-se utilizado uma função mínima e máxima, para identificar as issues dos repositórios, desde o dia da entrega desse trabalho.

4. CONCLUSÃO

Esta pesquisa tem como intenção utilizar seis métricas e verificar se as suas perguntas podem ser validadas ou refutadas. Dessa maneira foram coletados os dados e a resposta para cada uma dessas métricas foram:

RQ 01 - Sistemas populares são maduros/antigos?

Conforme visto na primeira tabela não necessariamente os repositórios mais antigos são os mais maduros, visto que, se tem repositórios novos com grandes contribuições.

RQ 02. Sistemas populares recebem muita contribuição externa?

Os repositórios novos por se tratarem muitas vezes de algo novo e também por não serem tão conhecidos acabam não estando entre os com maiores contribuições externas.

RQ 03. Sistemas populares lançam releases com frequência?

Visualizando os dados finais os sistemas mais populares acabam não tendo releases com frequência.

RQ 04. Sistemas populares são atualizados com frequência?

Os sistemas populares são sim atualizados com mais frequência, isso provavelmente se deve a sua grande aceitação pela comunidade.

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

Os sistemas mais populares estão nas linguagens mais populares.

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

Os repositórios mais populares possuem um alto fechamento de issues.