

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS
Bacharelado em Engenharia de Software
Laboratório de Experimentação de Software

Breno Rosa Almeida
Lucas Lage e Silva
Matheus Brandão Freire
Paloma Dias de Carvalho

RELATÓRIO: CARACTERÍSTICAS DE REPOSITÓRIOS POPULARES

Belo Horizonte
2023

Breno Rosa Almeida
Lucas Lage e Silva
Matheus Brandão Freire
Paloma Dias de Carvalho

RELATÓRIO: CARACTERÍSTICAS DE REPOSITÓRIOS POPULARES

Relatório apresentado no Laboratório 1 da
disciplina de Medição e Experimentação de
Software

Professor: Dr. José Laertes Pires Xavier

Belo Horizonte

1 INTRODUÇÃO

O Relatório tem como objetivo apresentar as características dos repositórios populares disponíveis no GitHub. Através das análises realizadas é possível identificar os padrões dos códigos desenvolvidos e os parâmetros de avaliação dos repositórios de código aberto disponibilizados na plataforma.

A análise foi realizada contemplando os 1.000 repositórios mais bem avaliados segundo a quantidade de estrelas, disponibilizados na plataforma até Agosto de 2023. A linguagem utilizada foi Python para mineração e API GraphQL para requisições.

Os dados analisados e os gráficos estão disponíveis no seguinte repositório: <https://github.com/LucasLS02/lab-experimentacao-de-software>

1.1 Hipóteses

RQ 01. Sistemas populares são maduros/antigos?

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

Hipótese: Sim, sistemas populares tendem a ser mais maduros e antigos. Isso porque a popularidade muitas vezes é construída ao longo do tempo, à medida que o projeto acumula contribuições, usuários e melhorias. Projetos mais antigos também têm a oportunidade de atrair uma base de usuários sólida ao longo dos anos.

RQ 02. Sistemas populares recebem muita contribuição externa?

Métrica: Total de pull requests aceitas

Hipótese: Sim, se o sistema é muito popular, significa que as pessoas se interessam em realizar melhorias, fazendo com que o número de pull requests no projeto seja alto, e com um número muito alto de pull requests, a quantidade que é aceita tende a ser alta também.

RQ 03. Sistemas populares lançam releases com frequência?

Métrica: Total de releases

Hipótese: Sim, sistemas populares geralmente lançam releases com frequência. Isso ocorre porque projetos populares são tipicamente ativos e estão em constante desenvolvimento. Lançar releases regularmente permite que eles entreguem novos recursos, correções de bugs e melhorias aos usuários de forma consistente.

RQ 04. Sistemas populares são atualizados com frequência?

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

Hipótese: Sim, pois um sistema popular possui um número grande de surgimento de erros, que podem acontecer pelo uso em massa de usuários simultâneos, além de possuir muitas prováveis melhorias/adição de funcionalidades, sendo assim a frequência de atualizações deve ser alta.

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

Métrica: Linguagem primária de cada um desses repositórios

Hipótese: Sim, sistemas populares tendem a ser escritos nas linguagens mais populares do momento. Isso ocorre porque as linguagens populares muitas vezes possuem uma ampla base devs e uma vasta coleção de bibliotecas de suporte. No entanto, existem exceções, pois a escolha da linguagem pode depender da finalidade e das preferências dos desenvolvedores.

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

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

Hipótese: Sim, pois, para um sistema ser popular, entende-se que o número de erros dentro do software deve ser minimizado ao máximo, assim o percentual de issues fechadas deve ser alto.