Análise dos *Bug Fixes* do *Framework* Hadoop

Disciplina: Visualização Científica
Profª.: Emanuele Marques dos Santos
Alunos: Armando Soares e Juarez Meneses

Agenda

- Introdução
- Descrição dos Dados
- Tecnologias Utilizadas
- Divisão do Trabalho
- Dificuldades Enfrentadas/Lições Aprendidas
- Screenshots das Visualizações

Introdução

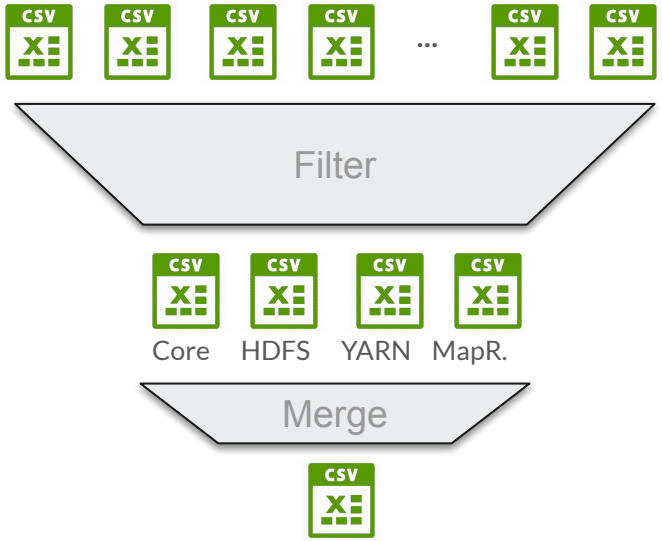
- Este projeto tem como objetivo explorar e analisar um subconjunto de datasets extraídos e propostos no paper [1]:

"From Reports to Bug-Fix Commits:
A 10 Years Dataset of Bug-Fixing Activity from 55 Apache's Open Source Projects"

- Motivado pela a importância da análise de repositórios de softwares livres através da mineração de dados para identificar padrões de comportamento dos bugs ao longo da evolução do software.

Descrição dos Dados

- O dataset publicado pelo artigo contém 156 arquivos .csv
- Foi realizado um filtro em um subconjunto do dataset do framework do Hadoop [2]
- Composto pelos módulos Core [3], Yarn [4], HDFS [5] e Mapreduce [6]
- Representa 30% do conjunto total de bugs registrados



Framework Hadoop

Descrição dos Dados

- Para a limpeza e transformação dos dados foi usada a biblioteca Pandas [8] do Python [9].
- Nessa etapa, cada arquivo .cvs foi analisado de forma a identificar os tipos de dados de cada coluna, identificar dados faltantes, dados nulos, transformações de dados do tipo string para numéricos ou tipos data
- Foram criados alguns campos calculados para facilitar a análise dos dados. Ex: Tempo de Vida do Bug

From	Type	Field
Jira (30)	General (10)	Project
		Owner
		Manager
		Category
		Key
		Priority
		Status
		Reporter
		Assignee
		Components
	Link (2)	InwardIssueLinks
		OutwardIssueLinks
	Summation (4)	NoComments
		NoWatchers
		NoAttachments
		NoAttachedPatches
	Text (3)	SummaryTopWords
		DescriptionTopWords
		CommentsTopWords
	Time (8)	CreationDate
		ResolutionDate
		FirstCommentDate
		LastCommentDate
		FirstAttachmentDate
		LastAttachmentDate
		FirstAttachedPatchDate
		LastAttachedPatchDate
	Versioning (2)	AffectsVersions
		FixVersions

Disponível em: https://colab.research.google.com/drive/1tuxv1PHU-ORo9HmrgvUV3kKRQVi5lHAJ

Tecnologias Utilizadas

- Python,
- Pandas,
- Seaborn,
- Jupyter Notebook,
- Observable Notebook
- JavaScript
- D3, DC e Crossfilter
- CodeFlower

Divisão do Trabalho

Armando Soares

1. Processo de Data Mining e Data Cleaning no Google Colaboratory
2. Implementação dos Gráficos Especiais usando o Code Flower
3. Exportação e Merge entre os projetos

Disponível em:
https://colab.research.google.com/drive/1tuxv1PHU-QRo9HmrgvUV3kKRQVi5lHAJ
https://github.com/scientific-visualization/project-final-datavis

Juarez Meneses

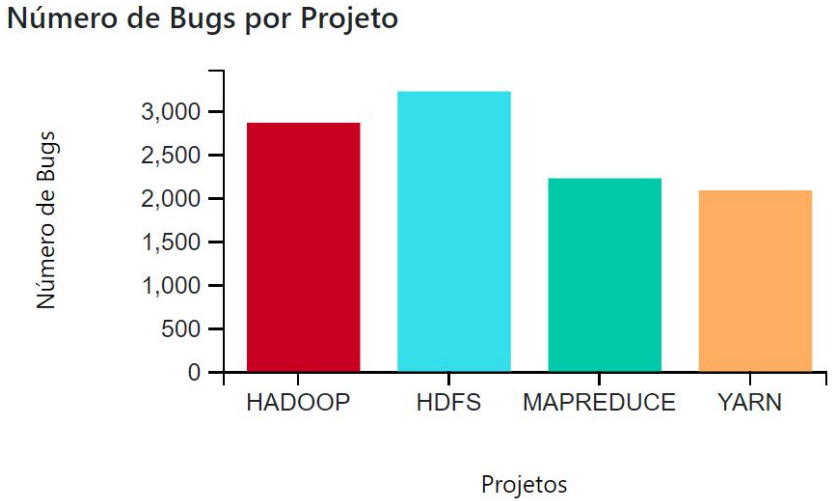
1. Prototipação dos Gráficos no Tableau
2. Implementação dos Gráficos Simples no Observable
3. Exportação do Projeto para o GitHub Pages

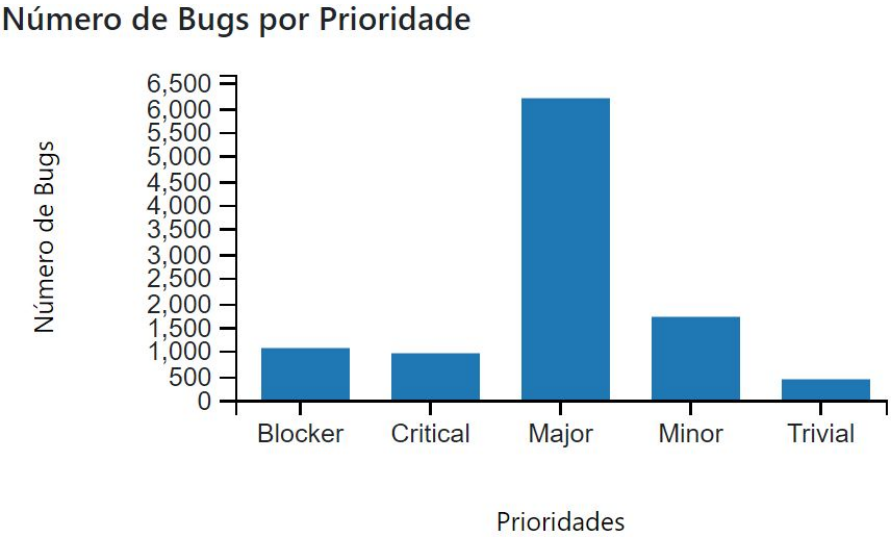
Disponível em:
https://observablehq.com/@juarezmeneses/analise-dos-bug-fixes-do-framework-hadoop/2
https://juarezmeneses.github.io/project-final-datavis/

Dificuldades Enfrentadas/Lições Aprendidas

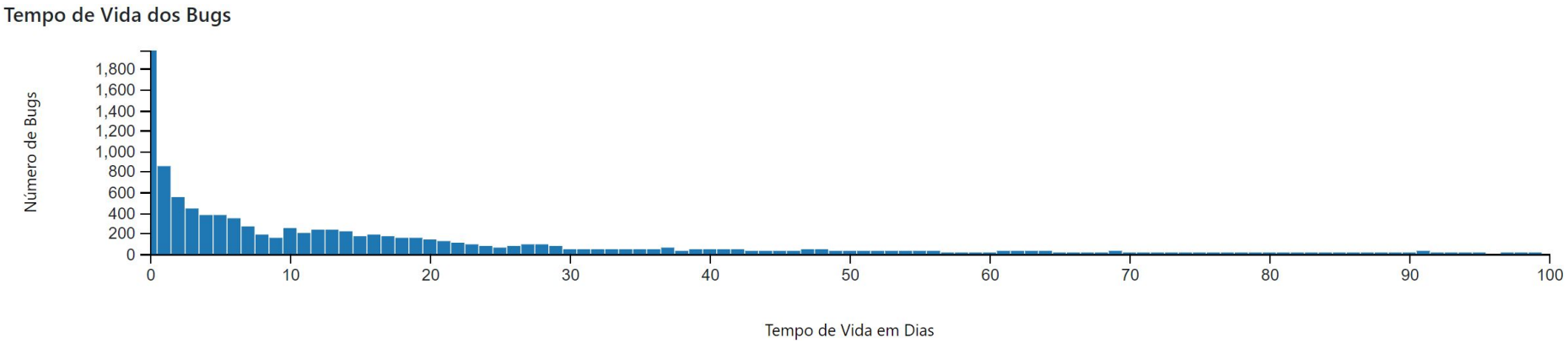
- Lidar com Grande Volume de Dados
- Tratar Ruído nos Dados
- Navegação da estrutura de diretórios e arquivos em cada uma das versões analisadas do Hadoop
- Grandes mudanças arquiteturais entre as versões base 1.x, 2.x e 3.x do Hadoop
- Otimização dos Gráficos
- Foi necessário fazer customizações do Code Flower (alterações no D3, JavaScript e CSS) para suportar os arquivos dos repositórios do Hadoop
- Integrações dos projetos entre site dinâmico e site estático usando o git.io e as diversas ferramentas usadas.

Screenshots das Visualizações



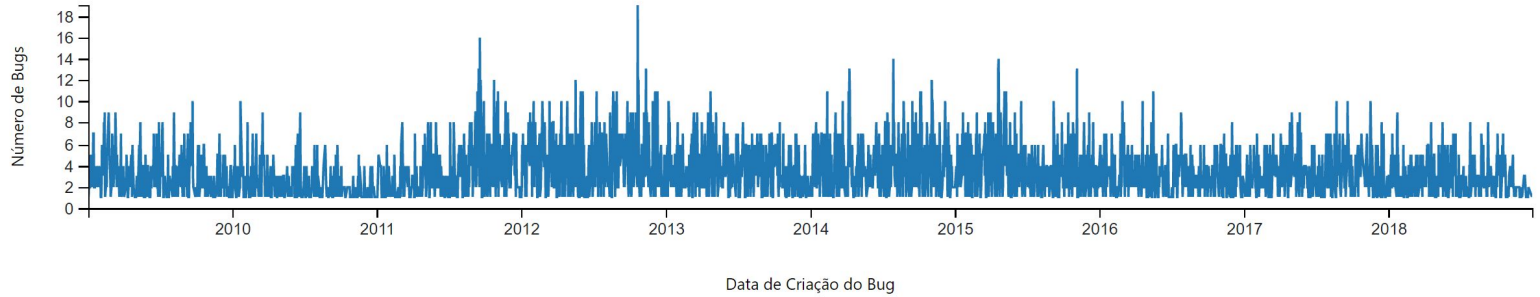


Screenshots das Visualizações

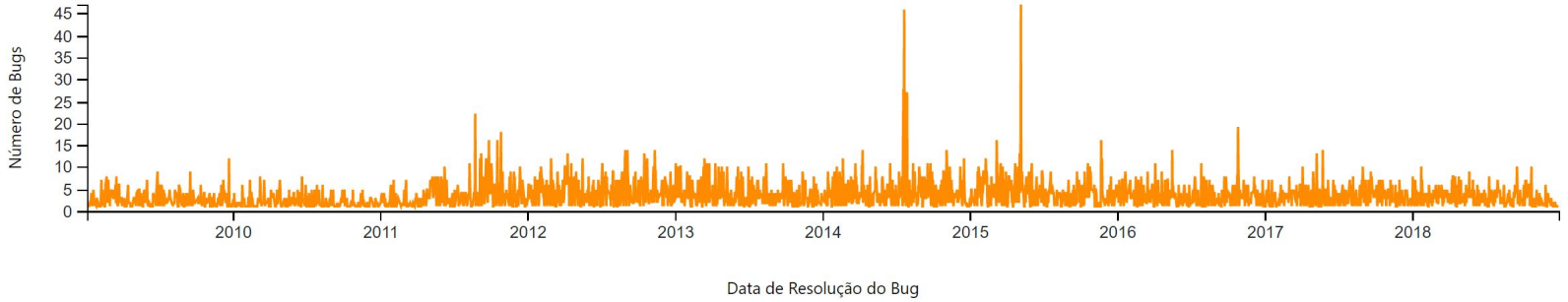


Screenshots das Visualizações

Quantidade de Bugs Criados no Decorrer dos Anos

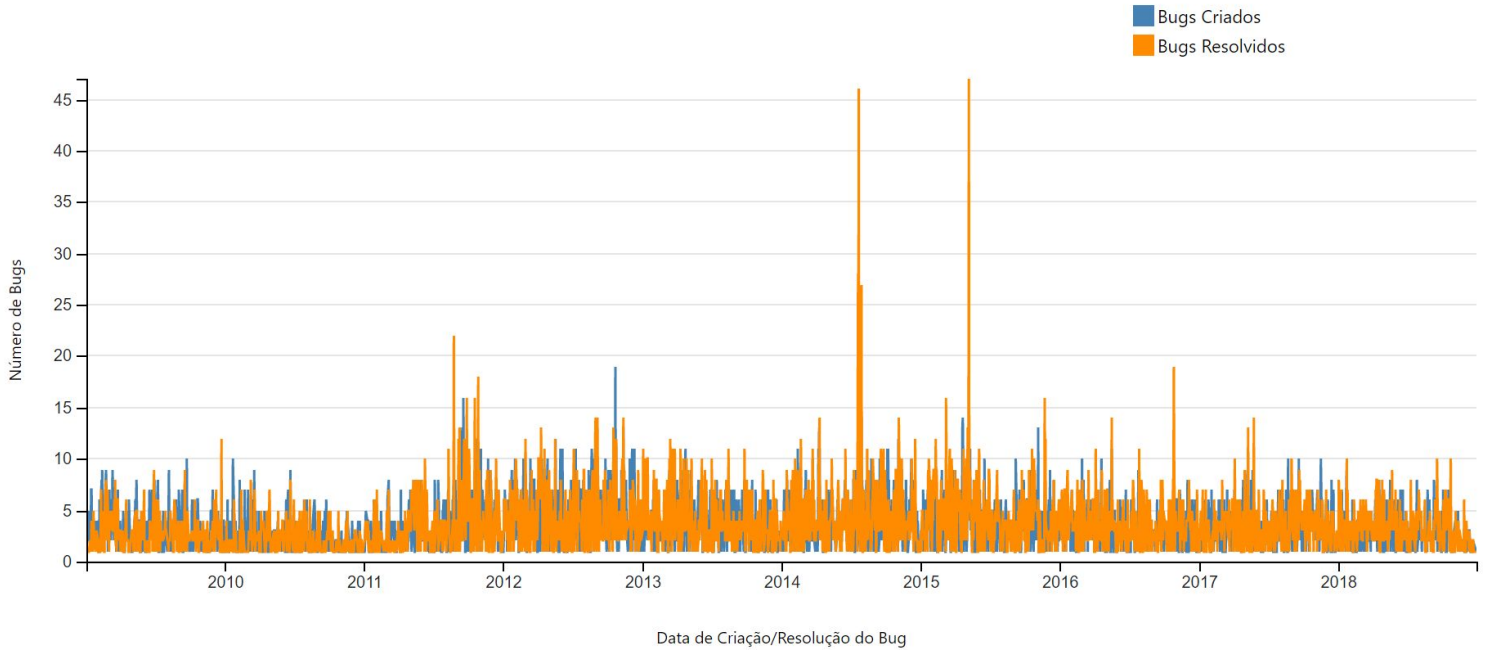


Quantidade de Bugs Resolvidos no Decorrer dos Anos

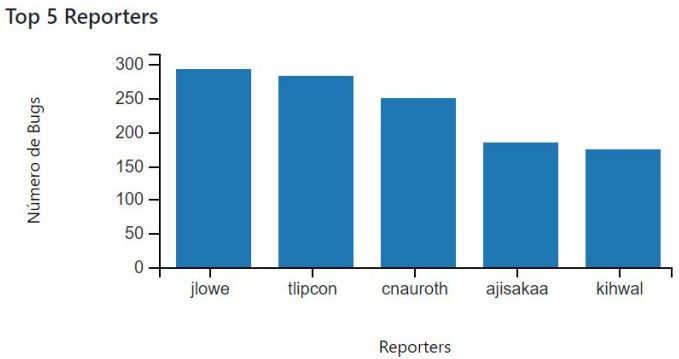


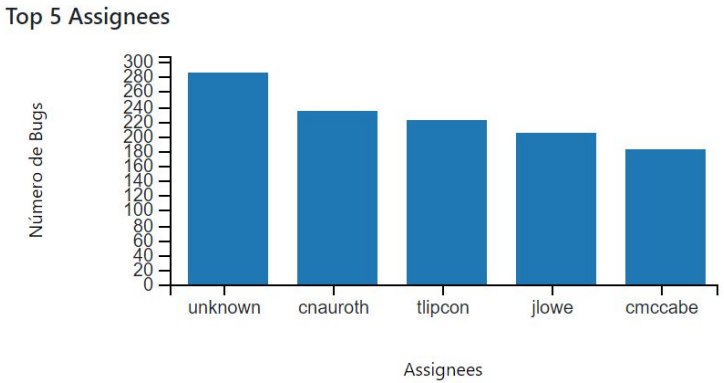
Screenshots das Visualizações

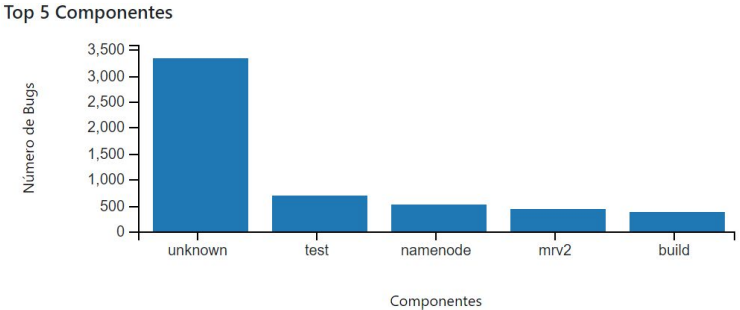
Quantidade de Bugs Criados X Quantidade de Bugs Resolvidos



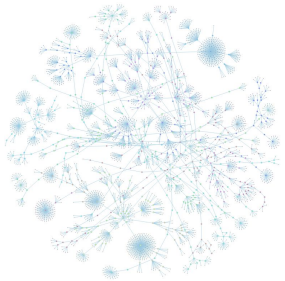
Screenshots das Visualizações



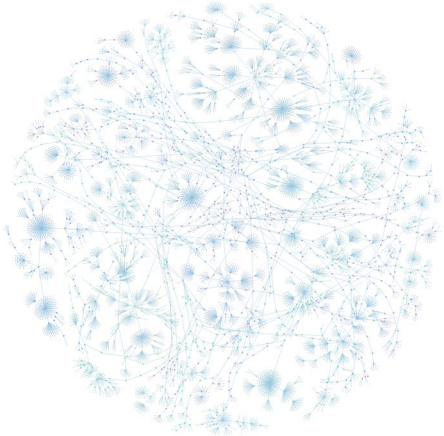




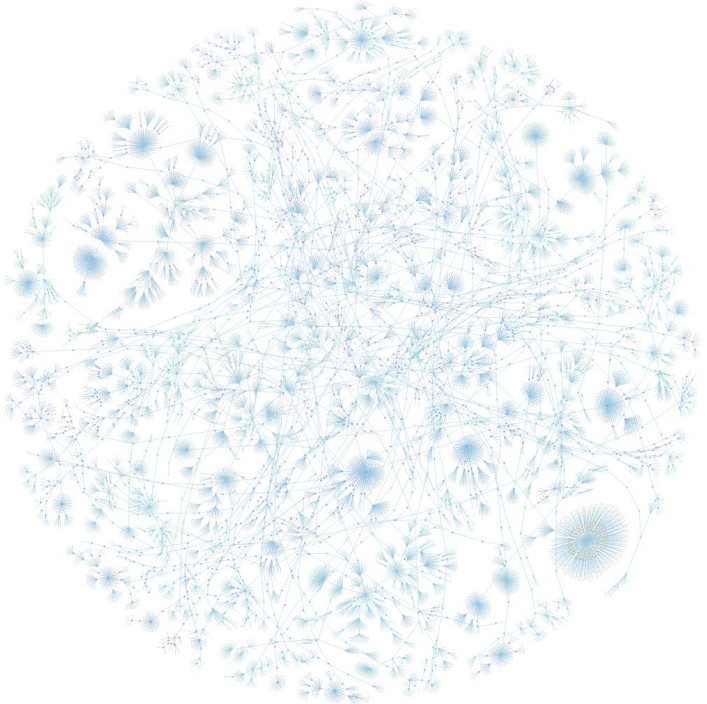
Screenshots das Visualizações (Especial - Pacotes e Arquivos)



Release 1.0 - 2741 files

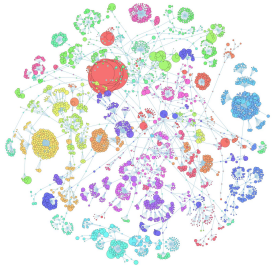


Release 2.3.0 - 5636 files

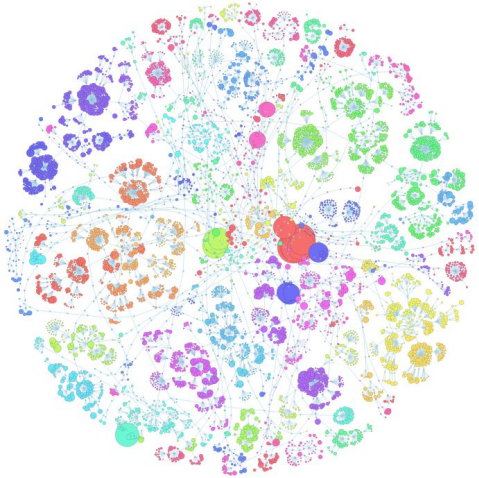


Release 3.2.1 - 12522 files

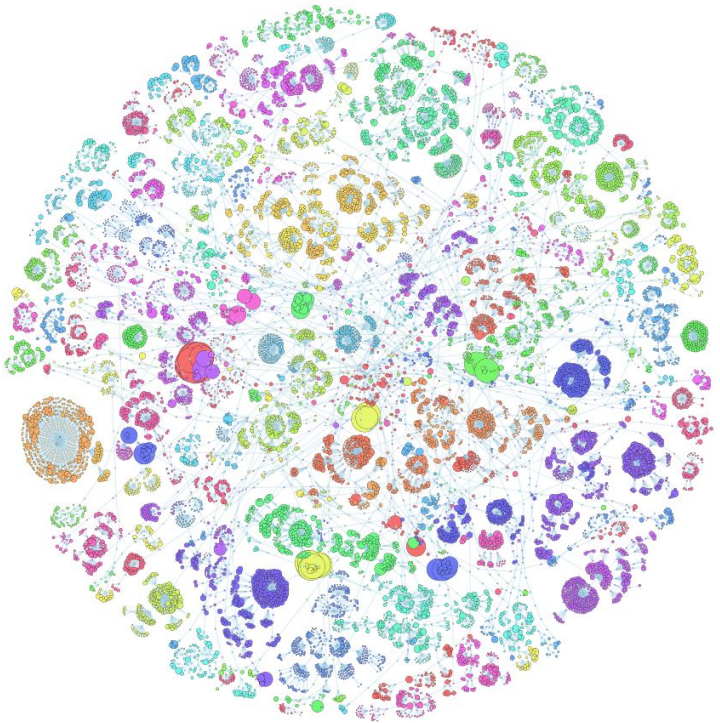
Screenshots das Visualizações (Especial - LOC)



Release 1.0



Release 2.3.0

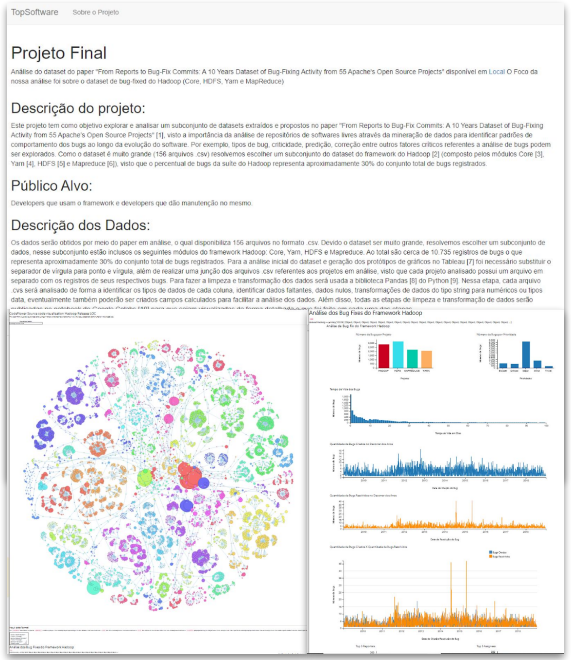


Release 3.2.1

Link do site e do repositório do projeto

- Site dinâmico disponível em: https://scientific-visualization.github.io/project-final-datavis/

- Repositório de código do projeto disponível em: https://github.com/scientific-visualization/project-final-datavis



Referências

[1] Renan Vieira, Antônio da Silva, Lincoln Rocha, and João Paulo Gomes. 2019. From Reports to Bug-Fix Commits: A 10 Years Dataset of Bug-Fixing Activity from 55 Apache's Open Source Projects. In Proceedings of the Fifteenth International Conference on Predictive Models and Data Analytics in Software Engineering (PROMISE'19). ACM, New York, NY, USA, 80-89. DOI: https://doi.org/10.1145/3345629.3345639

[2] Hadoop - Framework open-source para computação massiva paralela e distribuída de dados. Disponível em https://hadoop.apache.org

[3] Hadoop Core - Integração do Sistema de Arquivos distribuídos HDFS e o modelo de programação Mapreduce. Disponível em https://hadoop.apache.org/releases.html

[4] Hadoop Yarn - Gerenciamento dos recursos dentro dos clusters criados. Disponível em https://hadoop.apache.org/docs/current/hadoop-yarn/hadoop-yarn-site/YARN.html

[5] Hadoop HDFS - Sistema de Arquivos Distribuídos do Hadoop. Disponível em https://hadoop.apache.org/docs/r1.2.1/hdfs_design.html

[6] Hadoop Mapreduce - Modelo de Programação Paralela do Hadoop. Disponível em https://hadoop.apache.org/docs/r1.2.1/mapred_tutorial.html

[7] Tableau - Software de Visualização de Dados. Disponível em https://www.tableau.com

[8] Pandas - Biblioteca de Software escrita em Python para manipulação e análise de dados. Disponível em https://pandas.pydata.org

[9] Python - Linguagem de Programação interpretada de propósito geral. Disponível em https://www.python.org

[10] Google Colabs - Plataforma on-line do Google implementada sobre o Jupyter Notebook para publicar na Web programas feitos em Python. Disponível em https://colab.research.google.com/

[11] Seaborn - Biblioteca escrita em Python para construção e exibição de gráficos estatísticos. Disponível em https://seaborn.pydata.org

[12] Jupyter Notebook - Plataforma on-line para publicar dados interativos. Disponível em https://jupyter.org

[13] JavaScript - Linguagem de programação interpretada para Browsers Web e Servidores Web. Disponível em https://developer.mozilla.org/en-US/docs/Web/JavaScript

[14] D3 (Data-Driven Documents) - Biblioteca JavaScript para produzir visualização de dados em Browsers Web disponível em https://d3js.org

Contatos

Armando Soares
armando@ufpi.edu.br

Juarez Meneses
juarezmeneses@great.ufc.br

Dúvidas?