

apter]quadroloqQuadro



**FEDERAL UNIVERSITY OF CEARÁ**  
**CENTER OF XXXXXXXXX**  
**DEPARTMENT OF XXXXXXXXXX**  
**POST-GRADUATION PROGRAM IN XXXXXX**  
**DOCTORAL DEGREE IN XXXXXXXX**

**FULL NAME**

**TITLE**

**FORTALEZA**

**2019**

FULL NAME

TITLE

Thesis submitted to the Post-Graduation Program in Xxxxxx of the Center of Xxxxxxxx of the Federal University of Ceará, as a partial requirement for obtaining the title of Doctor in xxxxxx Engineering. Concentration Area: xxxxxx Engineering

Advisor: Prof. Dr. Xxxxxxxx Xxxxxx Xxxxxxx

FORTALEZA

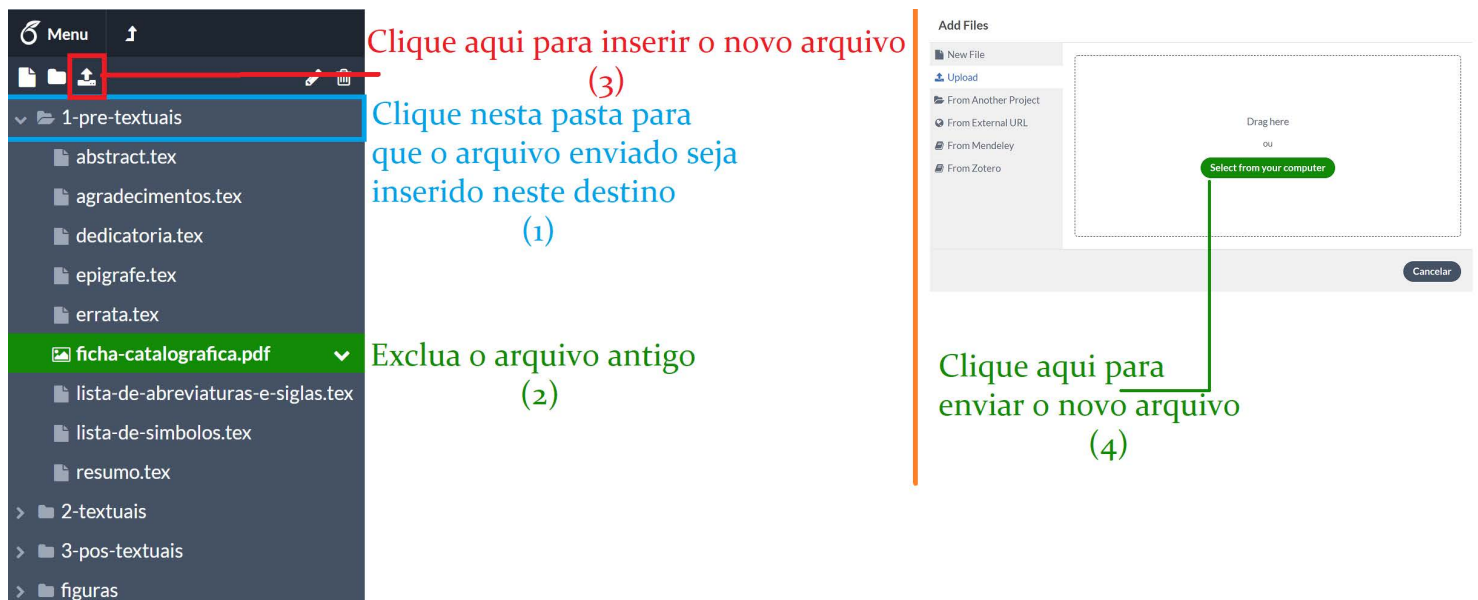
2019

Para criar sua ficha catalográfica, preencha corretamente o Módulo de Elaboração de Fichas Catalográficas (CATALOG!) disponibilizado no link:

<http://fichacatalografica.ufc.br/>

Em seguida, deve-se renomear o arquivo gerado como “ficha-catalografica” e adicioná-lo ao *template* na pasta “1-pre-textuais”. É necessário, contudo, excluir o antigo arquivo “ficha-catalografica” antes de adicionar o novo.

A figura a seguir mostra os passos enumerados para a inclusão da ficha catalográfica no *Overleaf*.



FULL NAME

TITLE

Thesis submitted to the Post-Graduation  
Program in XXXXXX of the Center of  
XXXXXXXX of the Federal University of  
Ceará, as a partial requirement for obtaining  
the title of Doctor in xxxxxx Engineering.  
Concentration Area: xxxxxx Engineering

Approved on:

EXAMINATION BOARD

---

Prof. Dr. XXXXXXXX XXXXXX  
XXXXXXXX (Advisor)  
Federal University of Ceará (UFC)

---

Prof. Dr. XXXXXXXX XXXXXX XXXXXXXX  
Universidade do Membro da Banca Dois  
(SIGLA)

---

Prof. Dr. XXXXXXXX XXXXXX XXXXXXXX  
Universidade do Membro da Banca Três  
(SIGLA)

---

Prof. Dr. XXXXXXXX XXXXXX XXXXXXXX  
Universidade do Membro da Banca Quatro  
(SIGLA)

Agradeço à minha família, por sua capacidade de acreditar em mim e investir em mim, sem eles não seria nada do que sou hoje. Também dedico aos amigos que conquistei durante a graduação e levarei pra vida!

## ACKNOWLEDGEMENTS

À Prof. Emanuele Santos por me orientar no meu TCC e ter sido uma ótima orientadora, sempre tirando minhas dúvidas e auxiliando no meu percurso.

Ao Prof. Lincoln Rocha que forneceu os dados para a realização desse trabalho.

Ao Doutorando em Engenharia Elétrica, Ednardo Moreira Rodrigues, e seu assistente, Alan Batista de Oliveira, aluno de graduação em Engenharia Elétrica, pela adequação do *template* utilizado neste trabalho para que o mesmo ficasse de acordo com as normas da biblioteca da Universidade Federal do Ceará (UFC).

Ao Vinícius Almeida de Castro que sempre me apoiou nos meus desafios pessoais/profissionais e me ajudou com algumas dificuldades que tive na construção desse projeto.

Aos meus amigos do "Segue a Call" (apelido do nosso grupo) que sempre apoiaram minhas iniciativas e projetos, sem esse estímulo com certeza não teria caminhado tão rápido como caminhei. Além disso foram um forte braço de apoio na graduação e neste trabalho. Em especial: Messias Tayllan, Aluísio Alves, Laura Santiago, Pedro Henrique, Tomás Coelho, Matheus Pereira, Helena Oliveira, Michel Vasconcelos e José Vanderley.

“É preciso coragem para ser diferente e muita  
competência para fazer a diferença”

(Autor Desconhecido)

## **ABSTRACT**

In this on the radio waves: popular culture, peasants and the Basic Education Movement we analyze the participation of peasants of the Brazilian northeastern region in the Basic Education Movement. The focus of this thesis is to demonstrate how the labors involved with broadcast schools have elaborated actions for maintaining and spreading the schools in their communities, in order to achieve the necessary means to improve their way of life. Peasants of the Basic Education Movement have been coadjuvant of the modernizing catholic proposition of the early 1960s, by means of quite peculiar political and cultural representations. Some of these representations were: a meaning for the school, a role for the union and for the political participation, precepts of the land use rights and labor rights, and the multiple meanings of the radio as a mass communication, information and leisure medium. This study intends to stress that the actions – and the political enrollment – of the northeastern peasant could not ever be separated from the modernizing process. The connection can be observed in different social movements of the period, such as the Basic Education Movement, rural unions, the Catholic Agrarian Youth and the MCP. In this sense, we consider that, if the Brazilian modernization was a guideline for the institutions, political organisms and parties for the social movement, such a modernization was a guideline of demands based on elements of material life. Those elements included, by that time, the agrarian reform, the educational issue and labor urgencies.

**Keywords:** Adult education. Community schools. Peasants. Popular culture.



## RESUMO

Em *Pelas Ondas do Rádio: Cultura Popular, Camponeses e o MEB* analisa a participação de camponeses do nordeste brasileiro no Movimento de Educação de Base. A perspectiva da tese é a de demonstrar como os trabalhadores envolvidos com as escolas radiofônicas elaboraram ações para manutenção e reprodução da escola em sua comunidade, visando obter os benefícios necessários à reprodução e melhoria de seu modo de vida. A partir de representações políticas e culturais singulares, dentre as quais vigoraram: um sentido para escola, um papel para o sindicato e para participação política, preceitos do direito de uso da terra e dos direitos do trabalho, assim como, sentidos múltiplos para o uso do rádio como meio de comunicação, informação e lazer, os camponeses do MEB, foram coadjuvantes da proposição católica modernizadora de inícios de 1960. Isto posto, demarca que a ação do camponês nordestino e seu engajamento político, seja no MEB, nos sindicatos rurais, nas Juventudes Agrárias Católicas (JAC's), no MCP, e nas mais diversas instâncias dos movimentos sociais do período, não se apartaram do processo modernizador. Neste sentido, considera-se que a modernização brasileira foi pauta das instituições, organismos políticos e partidos, assim como, do movimento social, instância em que ela foi ressignificada a partir de elementos da vida material, que envolviam diretamente, no momento em questão, a problemática do direito a terra, do direito a educação e cultura e dos direitos do trabalho.

**Palavras-chave:** Camponeses. Cultura popular. Educação de adultos. Escola rural.

## LIST OF FIGURES

Figure 1 – Distribuição do tempo de correção dos bugs (TimeFixBug), em dias, em função do engajamento do desenvolvedor . . . . .	21
Figure 2 – Distribuição de commits por grupo de engajamento . . . . .	22
Figure 3 – Distribuição de desenvolvedores por grupo de engajamento . . . . .	23
Figure 4 – Gráfico de dispersão dos bugs em relação ao tempo de resolução e número de comentários. Pontos de dispersão coloridos em função do Projeto em que o bug está inserido. . . . .	24
Figure 5 – Gráfico de dispersão dos bugs em relação ao tempo de resolução e número de commits. Pontos de dispersão coloridos em função do Projeto em que o bug está inserido. . . . .	24
Figure 6 – Gráfico de dispersão dos bugs em relação ao tempo de resolução e número de committers. Pontos de dispersão coloridos em função do Projeto em que o bug está inserido. . . . .	25
Figure 7 – Gráfico de Violino dos bugs em relação ao tempo de resolução e número de autores por projeto. . . . .	25
Figure 8 – Gráfico de Violino dos bugs em relação ao tempo de resolução e número de autores por engajamento. . . . .	26
Figure 9 – Distribuição do tempo de correção dos bugs (TimeFixBug), em dias, em função de prioridade do bug e número de comentários . . . . .	27
Figure 10 – Distribuição do tempo de correção dos bugs (TimeFixBug), em dias, em função de prioridade do bug e número de comentários . . . . .	27
Figure 11 – Distribuição do tempo de correção dos bugs (TimeFixBug), em dias, em função de prioridade do bug e número de comentários . . . . .	28
Figure 12 – Distribuição do tempo de correção dos bugs (TimeFixBug), em dias, em função de prioridade do bug . . . . .	29
Figure 13 – Gráfico de dispersão dos bugs em relação ao tempo de resolução e número de comentários. Pontos de dispersão coloridos em função da Prioridade . . .	30
Figure 14 – Gráfico de dispersão dos bugs em relação ao tempo de resolução e número de commits. Pontos de dispersão coloridos em função da Prioridade. . . . .	30
Figure 15 – Gráfico de dispersão dos bugs em relação ao tempo de resolução e número de committers. Pontos de dispersão coloridos em função da Prioridade. . . . .	31

Figure 16 – Distribuição do numero de comentários da issue (NoComments) em função de prioridade do bug atrelado a issue . . . . .	31
Figure 17 – Gráfico de dispersão dos bugs em relação ao tempo de resolução e número de métodos presentes em arquivo de código-fonte envolvidos. Pontos de dispersão coloridos em função de Projetos. . . . .	32
Figure 18 – Gráfico de dispersão dos bugs em relação ao tempo de resolução e número de linhas de código modificadas em arquivos de código-fonte envolvidos. Pontos de dispersão coloridos em função de Projetos. . . . .	33
Figure 19 – Gráfico de dispersão dos bugs em relação ao tempo de resolução e complexidade ciclomática dos arquivos de código-fonte envolvidos. Pontos de dispersão coloridos em função de Projetos. . . . .	33
Figure 20 – Gráfico de dispersão dos bugs em relação ao tempo de resolução e número de tokens nos arquivos de código-fonte envolvidos. Pontos de dispersão coloridos em função de Projetos. . . . .	34
Figure 21 – Gráfico de distribuição dos bugs em relação ao grupo de engajamento e prioridade . . . . .	35
Figure 22 – Gráfico de distribuição do tempo de resolução dos bugs em relação à Prioridade para o grupo de desenvolvedores Pouco Engajados. . . . .	36
Figure 23 – Gráfico de distribuição do tempo de resolução dos bugs em relação à Prioridade para o grupo de desenvolvedores Engajados. . . . .	36
Figure 24 – Gráfico de distribuição do tempo de resolução dos bugs em relação à Prioridade para o grupo de desenvolvedores Muito Engajados. . . . .	37

## **LIST OF TABLES**

## **LIST OF ABBREVIATIONS AND ACRONYMS**

PDF   *Portable Document Format*

## LIST OF SYMBOLS

$A_e$	Área efetiva da antena
$B$	Largura de faixa em que o ruído é medido em Hertz
$d$	Distância em metros
$E$	Campo elétrico
$FA$	Fator da antena
$Gr$	Ganho de recepção
$h$	Altura efetiva ou comprimento efetivo de uma antena
$I$	Corrente elétrica
$k$	Constante de Boltzmann's
$K$	Eficiência de irradiação
$M$	Variação do patamar de ruído em função da RBW
$N$	Condutor de neutro
$NF$	Figura de ruído
$N_i$	Potência do ruído na entrada
$N_o$	Potência do ruído na saída
$P$	Potência
$R$	Resistência
$S_i$	Potência do sinal na entrada
$S_o$	Potência do sinal na saída
$t$	Tempo
$V$	Tensão
$Z_L$	Impedância da antena
$Z_o$	Impedância de referência ( $50\Omega$ )
$\lambda$	Comprimento de onda
$\Gamma$	Coefficiente de reflexão

## CONTENTS

<b>1</b>	<b>INTRODUÇÃO . . . . .</b>	<b>16</b>
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA . . . . .</b>	<b>17</b>
<b>2.1</b>	<b>Open Source . . . . .</b>	<b>17</b>
<b>2.2</b>	<b>Os desafios que envolvem open source . . . . .</b>	<b>17</b>
<b>2.3</b>	<b>Métricas em projetos de software . . . . .</b>	<b>17</b>
<b>3</b>	<b>TRABALHOS RELACIONADOS . . . . .</b>	<b>18</b>
<b>3.1</b>	<b>Mineração de Repositórios . . . . .</b>	<b>18</b>
<b>3.2</b>	<b>Técnicas de Visualização Utilizadas . . . . .</b>	<b>18</b>
<b>4</b>	<b>METODOLOGIA . . . . .</b>	<b>19</b>
<b>4.1</b>	<b>Base de Dados . . . . .</b>	<b>19</b>
<b>4.2</b>	<b>Hipóteses e Questionamentos . . . . .</b>	<b>19</b>
<b>4.3</b>	<b>Metodologia da Análise . . . . .</b>	<b>19</b>
<b>4.4</b>	<b>Métricas criadas . . . . .</b>	<b>19</b>
<b>4.5</b>	<b>Implementação . . . . .</b>	<b>20</b>
<b>5</b>	<b>RESULTADOS . . . . .</b>	<b>21</b>
<b>5.1</b>	<b>Hipótese 1: Os bugs corrigidos por desenvolvedores mais ativos possuem um tempo de correção de bug menor? . . . . .</b>	<b>21</b>
<b>5.2</b>	<b>Hipótese 2: Os bugs que possuem um maior envolvimento da comunidade (número de comentários, número de autores (desenvolvedores), por exemplo) possuem um tempo de correção menor? . . . . .</b>	<b>23</b>
<b>5.3</b>	<b>Hipótese 3: A prioridade definida na atividade de correção de bug se correlaciona com o tempo de correção? . . . . .</b>	<b>28</b>
<b>5.4</b>	<b>Hipótese 4: Existe uma correlação/relação entre a quantidade de modificações (sejam em linhas de código-fonte ou testes) e o tempo de resolução dos bugs? . . . . .</b>	<b>32</b>
<b>5.5</b>	<b>Desenvolvedores mais engajados realizam atividades de maior complexidade em menor tempo comparado aos desenvolvedores menos engajados? . . . . .</b>	<b>34</b>
<b>6</b>	<b>CONCLUSÕES E TRABALHOS FUTUROS . . . . .</b>	<b>38</b>
	<b>REFERENCES . . . . .</b>	<b>39</b>
	<b>APPENDICES . . . . .</b>	<b>40</b>
	<b>APPENDIX A – Exemplo de apêndice . . . . .</b>	<b>40</b>

<b>APPENDIX B</b> – Questionário utilizado para...	41
<b>APPENDIX C</b> – Códigos-fontes utilizados para...	42
<b>APPENDIX D</b> – <i>IEEE CEFC 2016</i>	43
<b>ANNEXES</b>	45
<b>ANNEX A</b> – Exemplo de um anexo	45
<b>ANNEX B</b> – Exemplo de um anexo em PDF	46



## 1 INTRODUÇÃO

Para começar a utilizar este *template*, siga o tutorial clicando no seguinte *link*:  
<<https://biblioteca.ufc.br/wp-content/uploads/2015/09/tutorial-sharelatex.pdf>>

Neste *template*, o autor irá encontrar diversas instruções e exemplos dos recursos do uso do  $\text{\LaTeX}$  na plataforma *ShareLatex*. O  $\text{\LaTeX}$  foi desenvolvido, inicialmente, na década de 80, por Leslie Lamport e é utilizado amplamente na produção de textos matemáticos e científicos, devido a sua alta qualidade tipográfica (GOOSSENS *et al.*, 1994).

O *ShareLatex* é uma plataforma *online* que pode ser acessado por meio de qualquer navegador de internet até mesmo de um *smartphone*. Essa plataforma dispensa a instalação de aplicativos no computador para desenvolver trabalhos em  $\text{\LaTeX}$ . Também, não é necessário instalar *packages*, ou seja, pacotes que permitem diferentes efeitos na formatação e no visual do trabalho. Todos os *packages* que este *template* utiliza são encontrados *online*.

Apresentam-se, também, neste modelo, algumas orientações de como desenvolver um trabalho acadêmico. Entretanto, este arquivo deve ser editado pelo autor de acordo com o seu trabalho sendo que a formatação já está de acordo com o aceito pela Universidade Federal do Ceará.

A introdução, tem como finalidade, dar ao leitor uma visão concisa do tema investigado, ressaltando-se o assunto de forma delimitada, ou seja, enquadrando-o sob a perspectiva de uma área do conhecimento, de forma que fique evidente sobre o que se está investigando; a justificativa da escolha do tema; os objetivos do trabalho; o objeto de pesquisa que será investigado. Observe que não se divide a introdução em seções, mas a mesma informa como o trabalho ao todo está organizado.

## **2 FUNDAMENTAÇÃO TEÓRICA**

### **2.1 Open Source**

### **2.2 Os desafios que envolvem open source**

### **2.3 Métricas em projetos de software**

### **3 TRABALHOS RELACIONADOS**

#### **3.1 Mineração de Repositórios**

#### **3.2 Técnicas de Visualização Utilizadas**

## 4 METODOLOGIA

### 4.1 Base de Dados

O conjunto de dados, que foi utilizado para a realização da análise, traz informações envolvendo atividades de correções de bugs em 4 projetos open-source voltados para ambiente big data da Apache Software Foundation: Hadoop Core, Yarn, MapReduce e HDFS. As atividades ocorreram entre o período do dia 01 de Janeiro de 2001 ao dia 02 de Janeiro de 2019. Esses dados foram coletados dos repositórios desses projetos no GitHub e do Jira, softwares voltados para hospedagem com versionamento de código e monitoramento de atividades, respectivamente.

### 4.2 Hipóteses e Questionamentos

Neste estudo, visamos estudar relações entre o tempo para resolução de bugs em comparação a outros fatores envolvidos em um projeto de desenvolvimento de software. Também buscamos melhores formas de construir visualizações para complementar essa análise. Para guiar essa análise, definimos hipóteses que buscaremos validar neste processo:

**Hipótese 1** Os bugs corrigidos por desenvolvedores mais ativos possuem um tempo de correção de bug menor?

**Hipótese 2** Os bugs que possuem um maior envolvimento da comunidade (número de comentários, por exemplo) possuem um tempo de correção menor?

**Hipótese 3** A prioridade definida na atividade de correção de bug se correlaciona com o tempo de correção?

**Hipótese 4** Existe uma correlação entre a quantidade de modificações (sejam em linhas de código-fonte ou testes) e o tempo de resolução dos bugs?

**Hipótese 5** Desenvolvedores mais engajados realizam atividades de maior complexidade em menor tempo comparado aos desenvolvedores menos engajados?

### 4.3 Metodologia da Análise

### 4.4 Métricas criadas

**TimeFixBug** Foi obtido a partir da diferença entre as variáveis de CreationDate e ResolutionDate, colocando em dimensão de dias.

**Categorização de Desenvolvedores** É baseado na média de frequência de commits feitos por ano em relação a todo o período mapeado. O desenvolvedor pouco engajado possui uma frequência média de igual ou inferior de 10 commits por ano. O desenvolvedor engajado possui uma frequência média entre 10 e 72 commits por ano. E, por fim, o desenvolvedor muito engajado possui uma frequência média de 72 commits por ano. Os valores que definem essa separação levam em conta os quartis da distribuição da frequência média de commits feitos por ano.

## 4.5 Implementação

O Python foi utilizado para realizar a extração de dados das bases, a manipulação dos dados e construção das visualizações. Para melhorar a acessibilidade as análises foi gerada uma visualização web utilizando o framework Streamlit, que permitiu uma melhor visibilidade das análises, aplicações de filtros e realização de algumas iteratividades. Esse framework foi combinado com bibliotecas de manipulação de dados, como Pandas e Numpy, juntamente com pacotes de visualização de dados: Matplotlib, Seaborn e Plotly.

## 5 RESULTADOS

### 5.1 Hipotese 1: Os bugs corrigidos por desenvolvedores mais ativos possuem um tempo de correção de bug menor?

Com a categorização feita dos desenvolvedores em relação ao seu engajamento no projeto como um todo, realizamos a análise da distribuição de tempo de correção dos bugs. Com o auxílio do gráfico de violino (ViolinPlot), podemos verificar como a variável de TimeFixBug está distribuída ao longo das categorias de engajamento dos desenvolvedores.

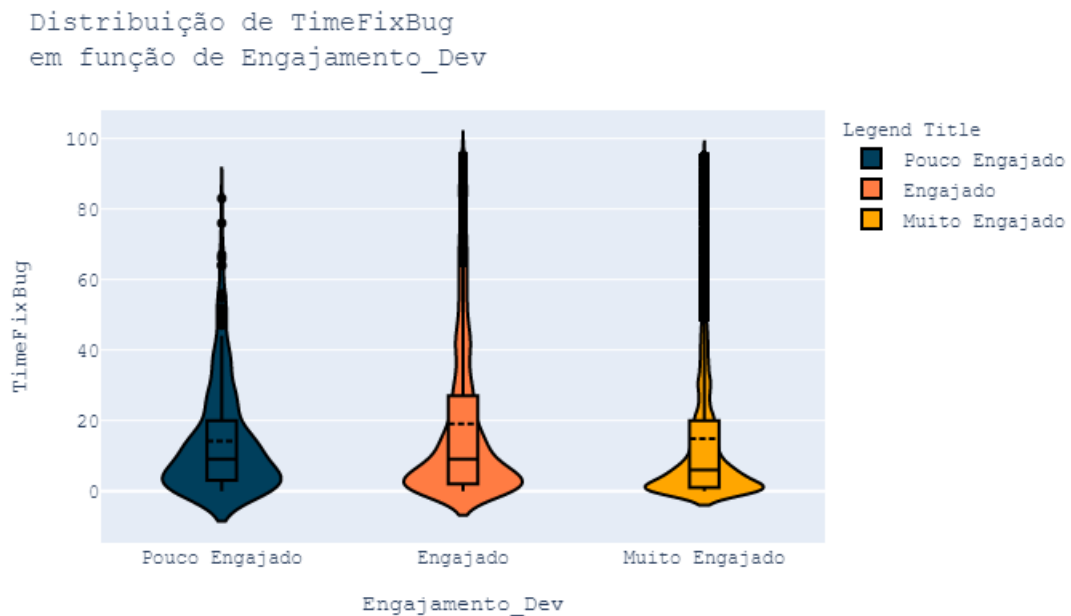


Figure 1 – Distribuição do tempo de correção dos bugs (TimeFixBug), em dias, em função do engajamento do desenvolvedor

No grupo de desenvolvedores muito engajados, 50% dos bugs resolvidos por esse grupo tiveram até 20 dias de tempo de resolução. Ao direcionarmos a análise para a distribuição do grupo engajado notamos que 75% dos bugs foram alocados até 27 dias de tempo de resolução. Por fim, no grupo de pouco engajado 75% dos bugs solucionados tiveram um tempo de resolução de até 20 dias.

Fazendo uma outra comparação, verificamos que entre os 3 grupos de desenvolvedores, o grupo de muito engajados possuem 50% dos bugs com até 6 dias de resolução e 25% dos bugs com até 1 dia de resolução, sendo os menores tempos comparando com os outros grupos

para essa mesma porção.

Vale ressaltar que o grupo de desenvolvedores muito engajados são responsáveis por mais de 20 000 commits, enquanto os engajados representam, aproximadamente, 6 000 commits e pouco engajados por, aproximadamente, 400 commits. Essa quantidade sendo a total olhando para o período total mapeado.

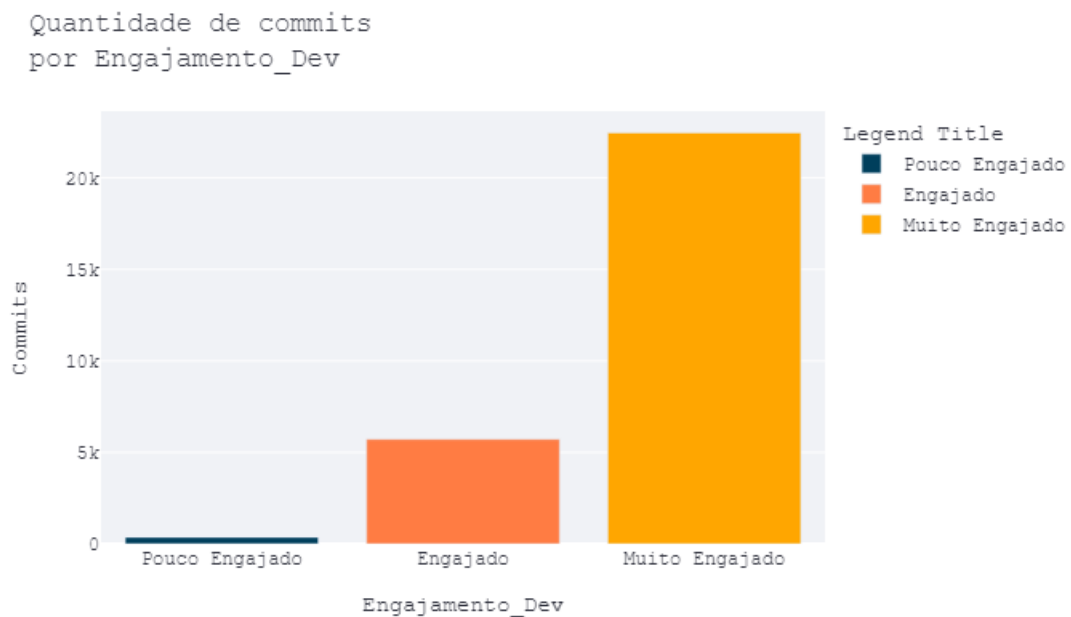


Figure 2 – Distribuição de commits por grupo de engajamento

Ao analisarmos a quantidade de desenvolvedores para cada grupo, percebemos que o grupo de muito engajados é composto por 36 desenvolvedores, engajados possuem 71 desenvolvedores e pouco engajados é composto por 57 desenvolvedores.

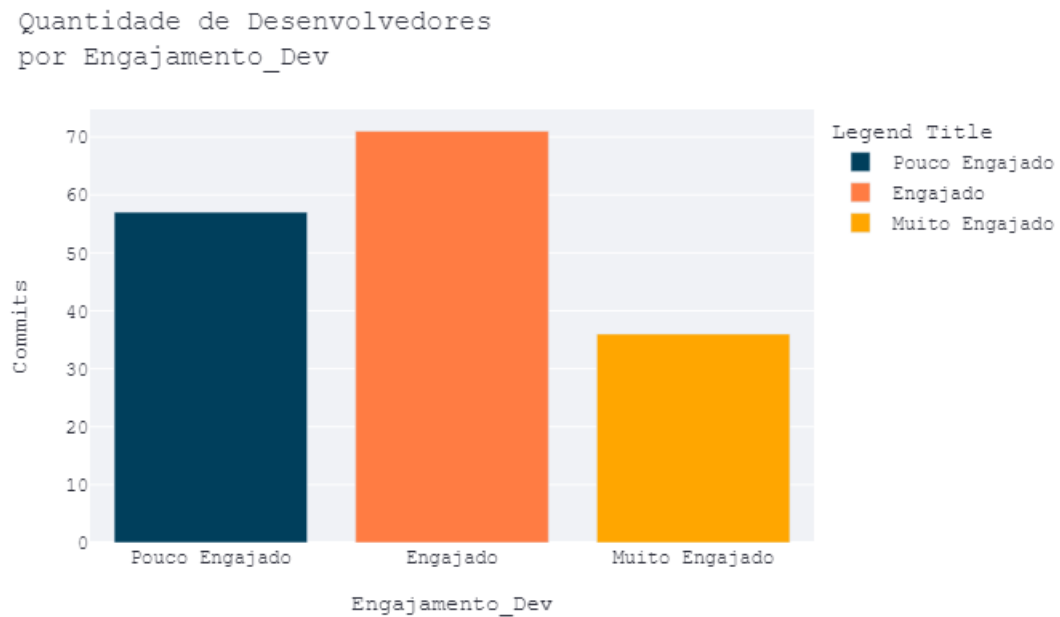


Figure 3 – Distribuição de desenvolvedores por grupo de engajamento

## 5.2 Hipótese 2: Os bugs que possuem um maior envolvimento da comunidade (número de comentários, número de autores (desenvolvedores), por exemplo) possuem um tempo de correção menor?

Construímos gráficos de dispersão dos bugs em relação ao tempo de solução e algumas variáveis numéricas (número de comentários, número de commits e número de committers), colorindo os marcadores em função do projeto em que o bug pertence. É perceptível que não existe uma tendência que permita afirmarmos uma relação linear direta entre as duas variáveis numéricas para todos os casos.



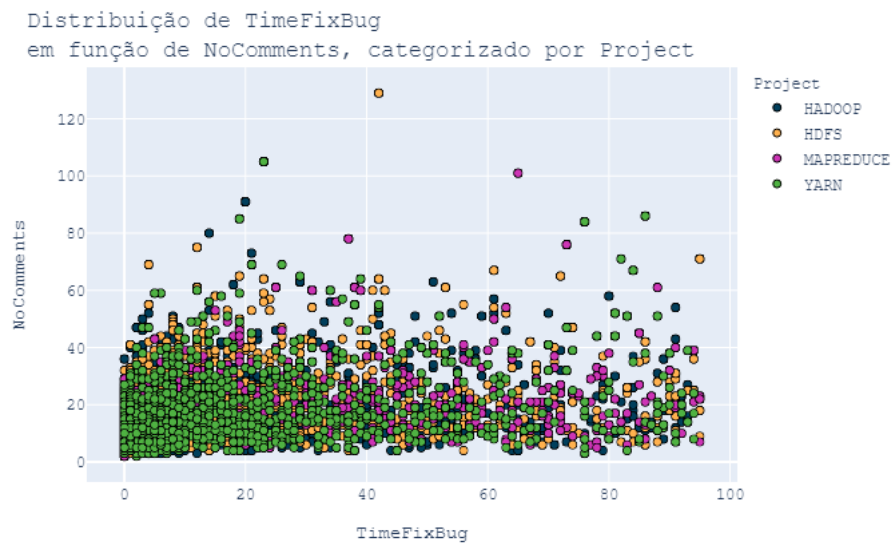


Figure 4 – Gráfico de dispersão dos bugs em relação ao tempo de resolução e número de comentários. Pontos de dispersão coloridos em função do Projeto em que o bug está inserido.

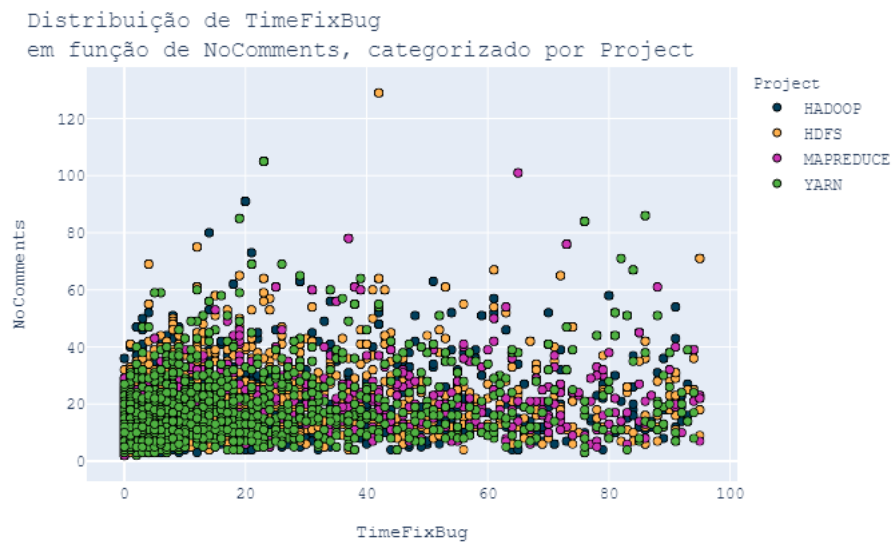


Figure 5 – Gráfico de dispersão dos bugs em relação ao tempo de resolução e número de commits. Pontos de dispersão coloridos em função do Projeto em que o bug está inserido.

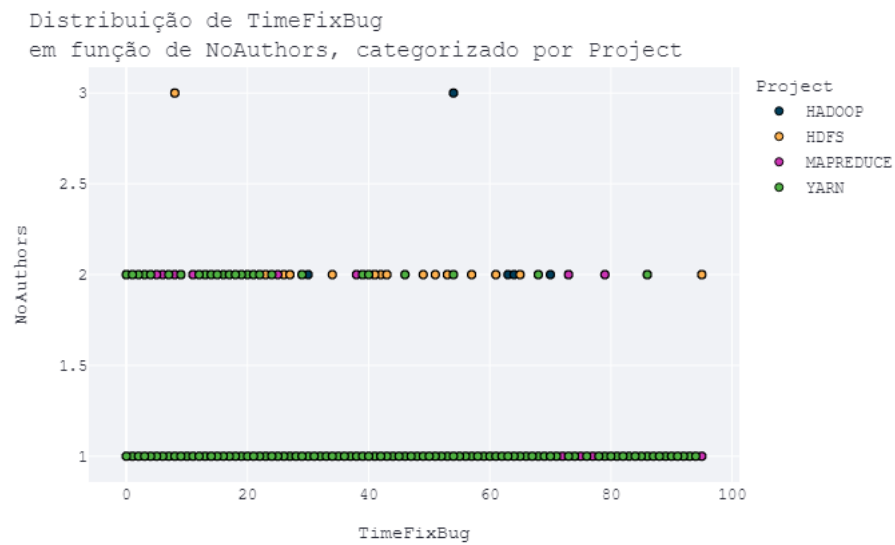


Figure 6 – Gráfico de dispersão dos bugs em relação ao tempo de resolução e número de comiters. Pontos de dispersão coloridos em função do Projeto em que o bug está inserido.

No entanto, quando analisamos a distribuição de tempo de solução de bug fazendo uma separação entre os casos com apenas um autor (desenvolvedor) e com dois ou mais autores (desenvolvedores), notamos que para os todos os projetos, 50% dos casos para mais de um autor tiveram um teto de TimeFixBug maior comparado aos 50% dos casos para um autor.

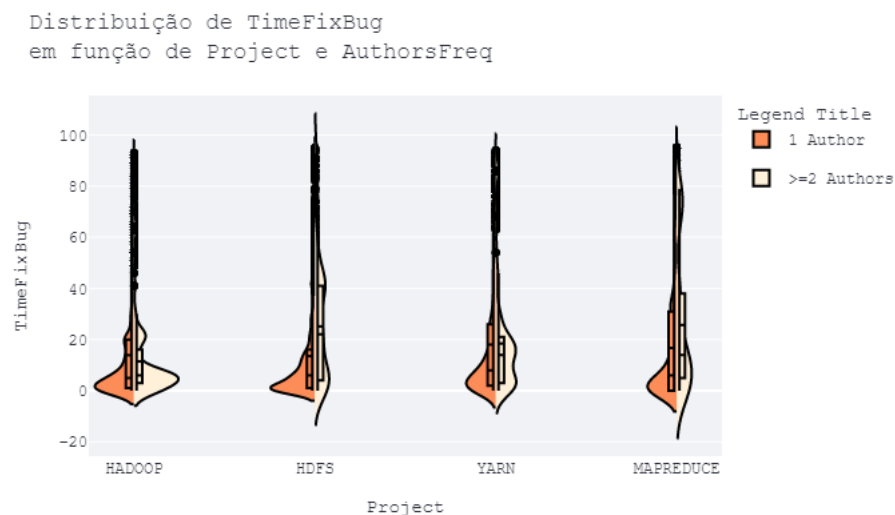


Figure 7 – Gráfico de Violino dos bugs em relação ao tempo de resolução e número de autores por projeto.

Gerando uma visualização similar, só que, em vez de ser por projeto, iremos visu-

alizer em função do engajamento.

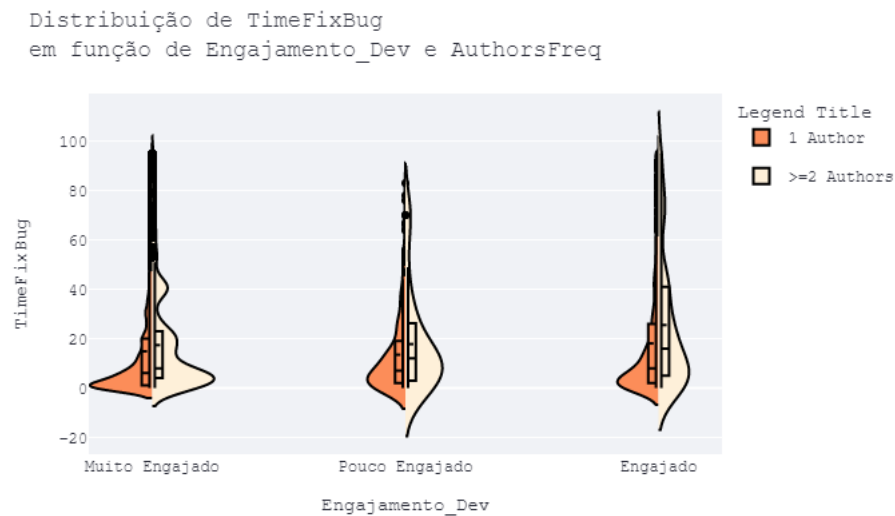


Figure 8 – Gráfico de Violino dos bugs em relação ao tempo de resolução e número de autores por engajamento.

Neste caso, ocorre uma situação similar ao do gráfico anterior (em relação aos projetos). Os 50% dos casos com mais de um autor tiveram um teto de TimeFixBug maior comparado aos 50% dos casos para um autor.

Apesar desses aspectos, não podemos afirmar que, obrigatoriamente, tendo mais de um desenvolvedor envolvido irá provocar um tempo de solução de bug mais longo. Apenas é notório a presença de casos de TimeFixBug mais longo em casos com mais de um autor. Vale reforçar que a quantidade de bugs com mais de um autor é muito pequena, cerca de 75% dos bugs possuem só um desenvolvedor envolvido.

Provocando uma análise similar à anterior, mas, em vez de envolver o número de desenvolvedores, envolvemos o número de comentários. Percebemos que para situações acima de vinte comentários tivemos uma maior ocorrência de maiores tempos de solução em comparativo aos bugs com menos de vinte comentários.

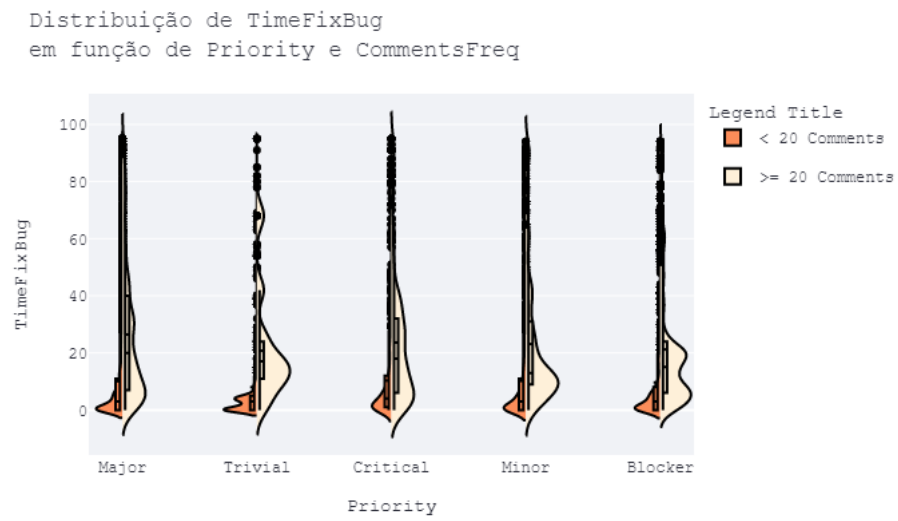


Figure 9 – Distribuição do tempo de correção dos bugs (TimeFixBug), em dias, em função de prioridade do bug e número de comentários

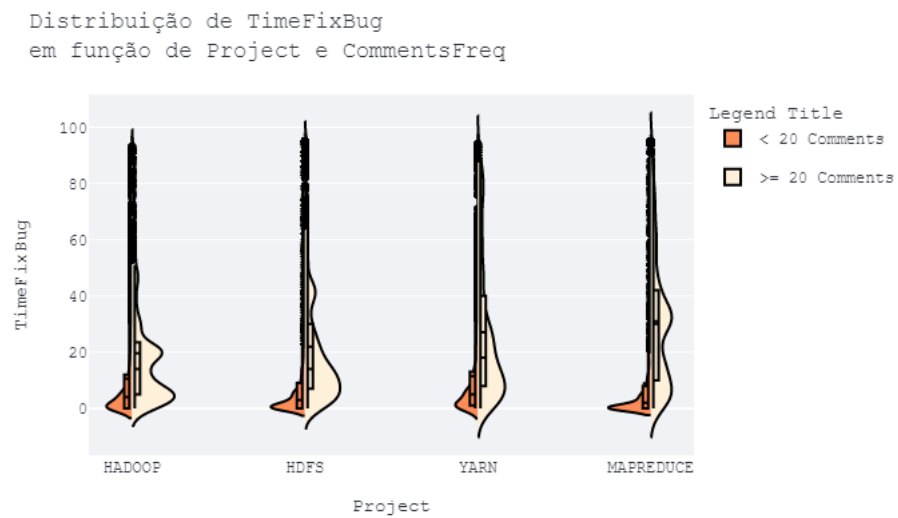


Figure 10 – Distribuição do tempo de correção dos bugs (TimeFixBug), em dias, em função de prioridade do bug e número de comentários

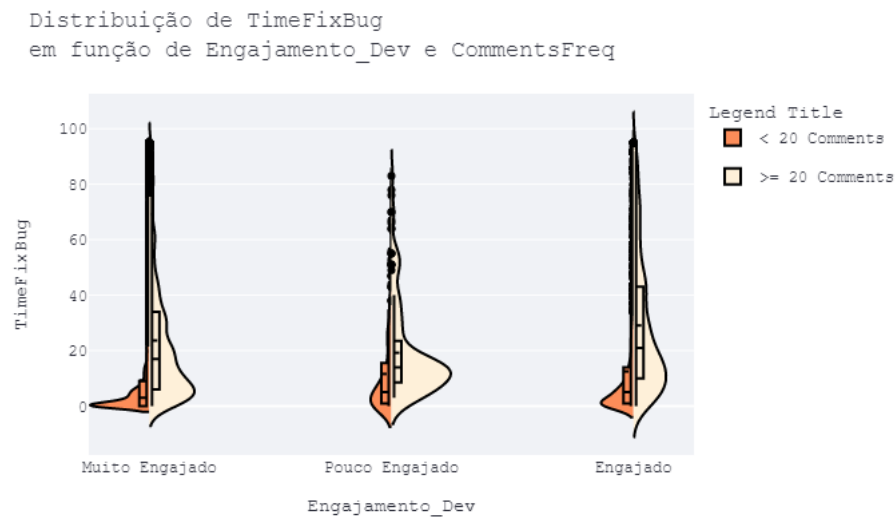


Figure 11 – Distribuição do tempo de correção dos bugs (TimeFixBug), em dias, em função de prioridade do bug e número de comentários

Por exemplo, olhando em relação ao engajamento, para os muito engajados (que é o grupo com maior presença na base), 75% dos bugs com mais de vinte comentários possuem até 34 dias de solução, enquanto 75% dos bugs com menos de vinte comentários possuem até 9 dias de solução.

Vale lembrar que até 50% dos bugs como um todo possuem até 16 comentários e 75% possuem até 25 comentários. O bug com maior número de comentários possui 129 comentários.

Apesar de não podermos afirmar que o fato de ter muitos comentários levará, obrigatoriamente, um tempo de solução maior, não podemos descartar que, por meio dessa análise, os bugs que possuem mais comentários estão menos concentrados em intervalos de tempos de solução menor.

### 5.3 Hipótese 3: A prioridade definida na atividade de correção de bug se correlaciona com o tempo de correção?

Para iniciarmos a análise, realizamos a construção do gráfico de Violino (Violin-Plot), tendo a variável de tempo de resolução como parâmetro da distribuição em função das prioridades.

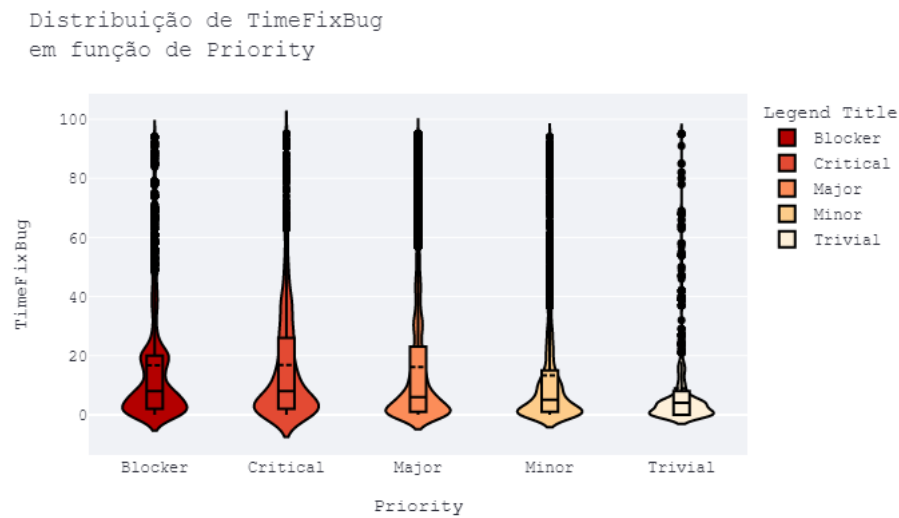


Figure 12 – Distribuição do tempo de correção dos bugs (TimeFixBug), em dias, em função de prioridade do bug

Pelo gráfico de distribuição podemos visualizar que quanto menor o nível de prioridade, menor é o tempo de correção do bug. A título comparativo 50% dos bugs com prioridade Blocker demandaram até 16.7 dias para serem solucionados, enquanto 50% dos bugs com prioridade Trivial demandaram até 8.9 dias para serem solucionados. Esse resultado faz sentido, pois bugs com prioridade Blocker/Critical costumam ter um nível de complexidade maior comparado aos de nível Trivial.

Ao colocarmos gráficos de dispersão dos bugs em relação ao tempo de solução e algumas variáveis numéricas (número de comentários, número de commits e número de committers), colorindo os marcadores em função da prioridade, percebemos que não existe uma tendência que permita afirmarmos uma relação direta entre as duas variáveis numéricas, assim como ocorreu nos gráficos de dispersão envolvendo projetos.

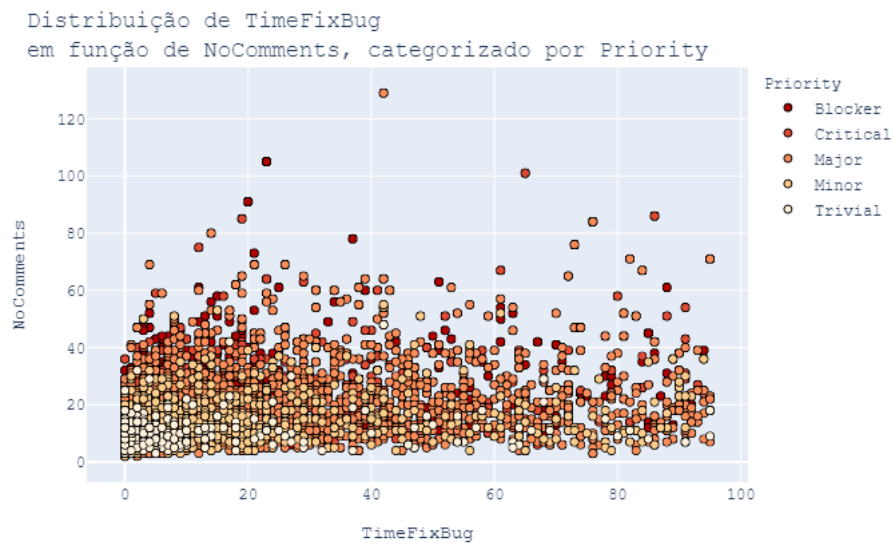


Figure 13 – Gráfico de dispersão dos bugs em relação ao tempo de resolução e número de comentários. Pontos de dispersão coloridos em função da Prioridade

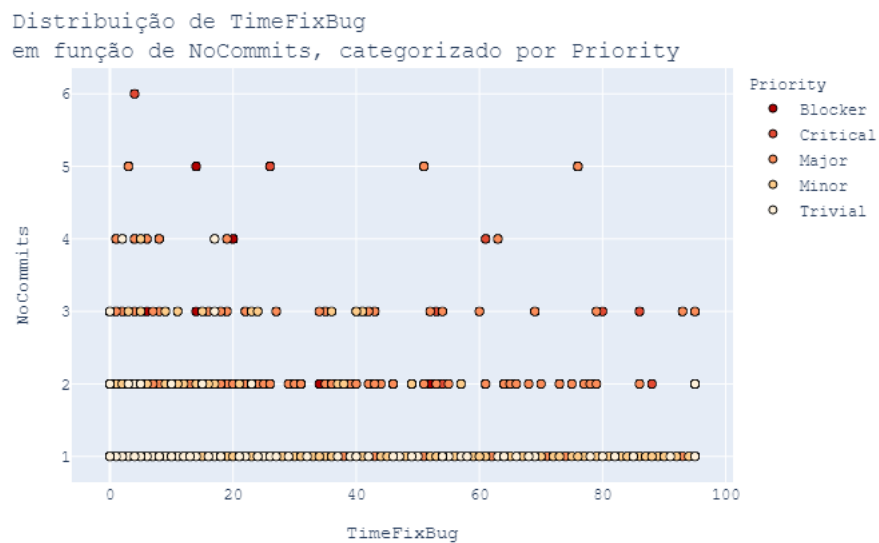


Figure 14 – Gráfico de dispersão dos bugs em relação ao tempo de resolução e número de commits. Pontos de dispersão coloridos em função da Prioridade.

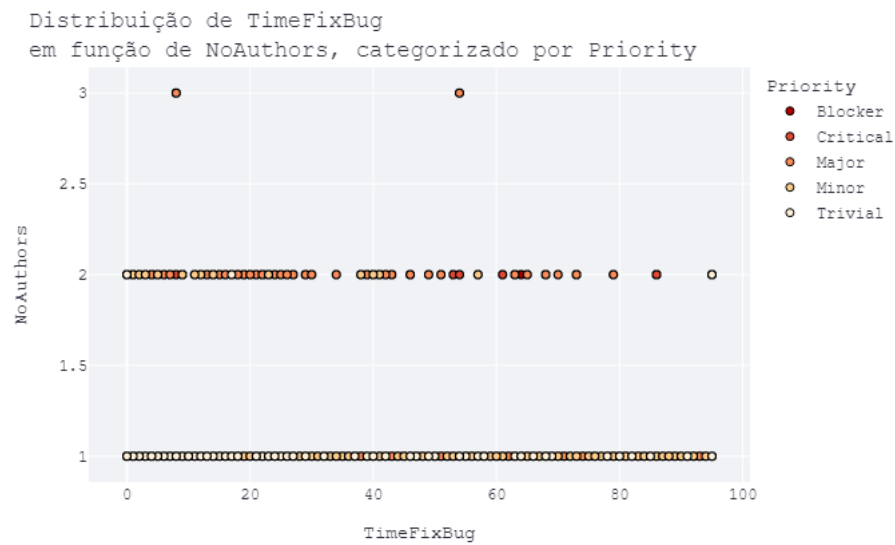


Figure 15 – Gráfico de dispersão dos bugs em relação ao tempo de resolução e número de committers. Pontos de dispersão coloridos em função da Prioridade.

No entanto, ao observar em relação a variável de prioridade, podemos notar que, em boa parte dos bugs de prioridade igual ou acima de major, existe um maior envolvimento da comunidade, isto é, um maior número de comentários. Isso pode ser reforçado observando a distribuição do número de comentários para cada prioridade por meio de gráficos de violino.

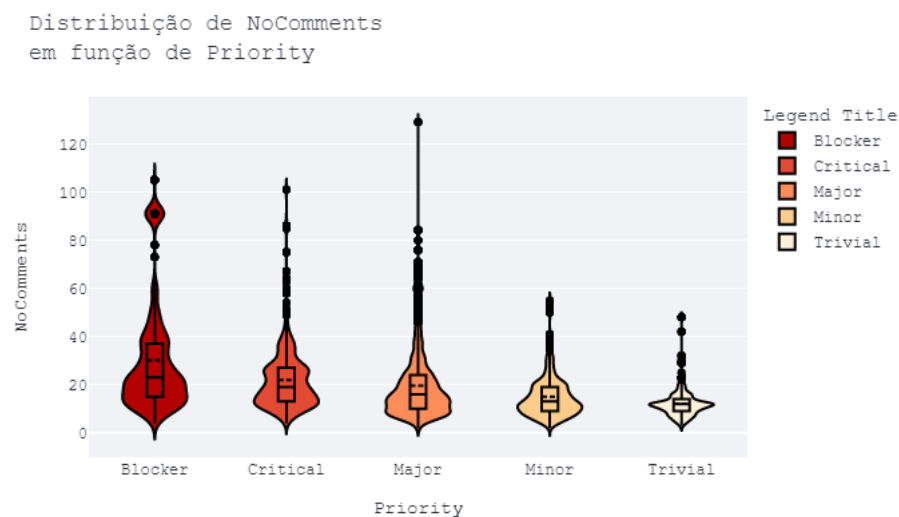


Figure 16 – Distribuição do numero de comentários da issue (NoComments) em função de prioridade do bug atrelado a issue

Enquanto 75% dos bugs de prioridade Trivial possuem até 14 comentários e 75% dos bugs de prioridade Minor possuem até 19 comentários, 75% dos bugs de prioridade Major



tiveram até 24 comentários, 75% dos bugs de prioridade Critical tiveram até 27 comentários e 75% dos bugs de prioridade Blocker possuem até 37 comentários. Além disso, é perceptível a existência de casos com mais de 60 comentários nos bugs das prioridades Major, Critical e Blocker.

#### 5.4 Hipótese 4: Existe uma correlação/relação entre a quantidade de modificações (sejam em linhas de código-fonte ou testes) e o tempo de resolução dos bugs?

Para validar essa hipótese colocamos gráficos de dispersão de TimeFixBug em relação variáveis que indicam características das modificações realizadas na correção dos bugs.

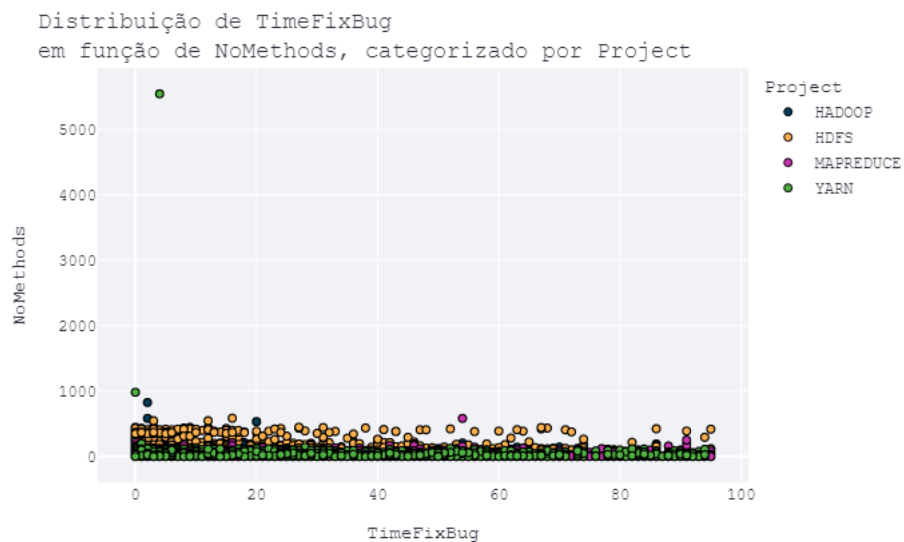


Figure 17 – Gráfico de dispersão dos bugs em relação ao tempo de resolução e número de métodos presentes em arquivo de código-fonte envolvidos. Pontos de dispersão coloridos em função de Projetos.

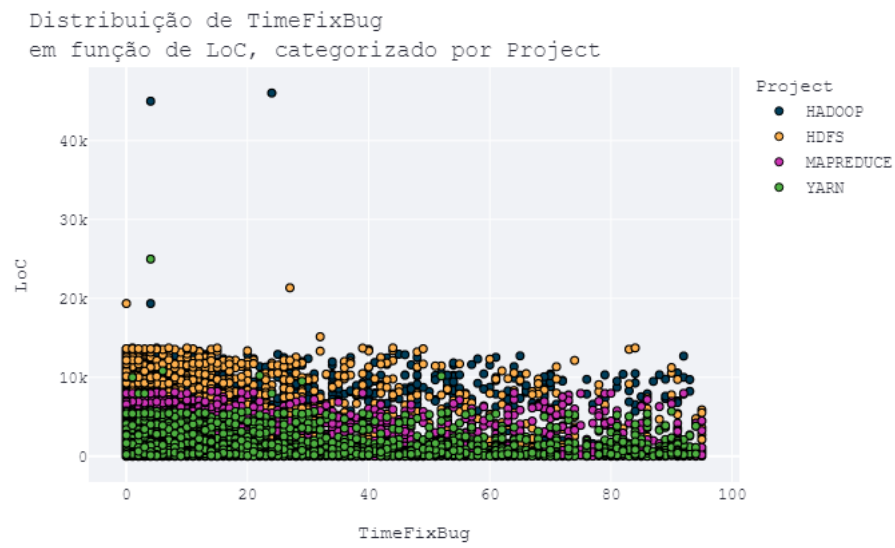


Figure 18 – Gráfico de dispersão dos bugs em relação ao tempo de resolução e número de linhas de código modificadas em arquivos de código-fonte envolvidos. Pontos de dispersão coloridos em função de Projetos.

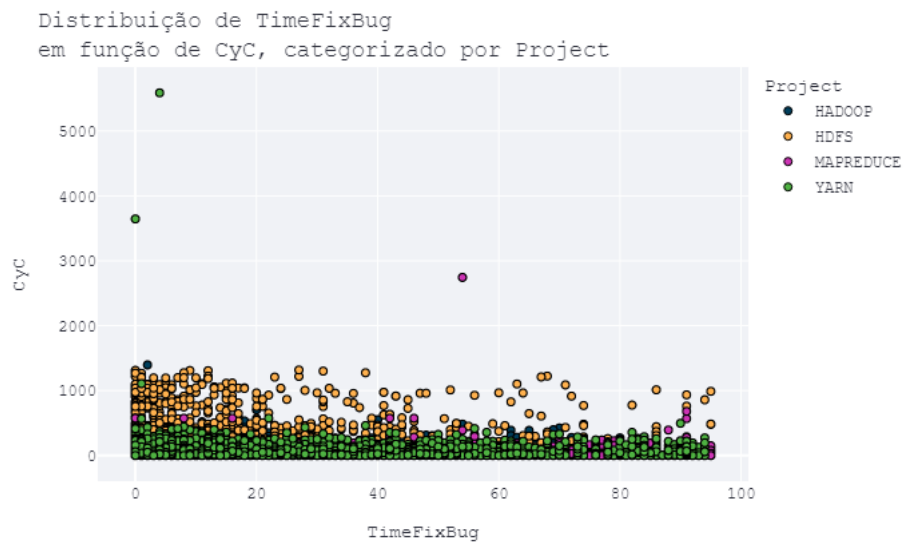


Figure 19 – Gráfico de dispersão dos bugs em relação ao tempo de resolução e complexidade ciclomática dos arquivos de código-fonte envolvidos. Pontos de dispersão coloridos em função de Projetos.

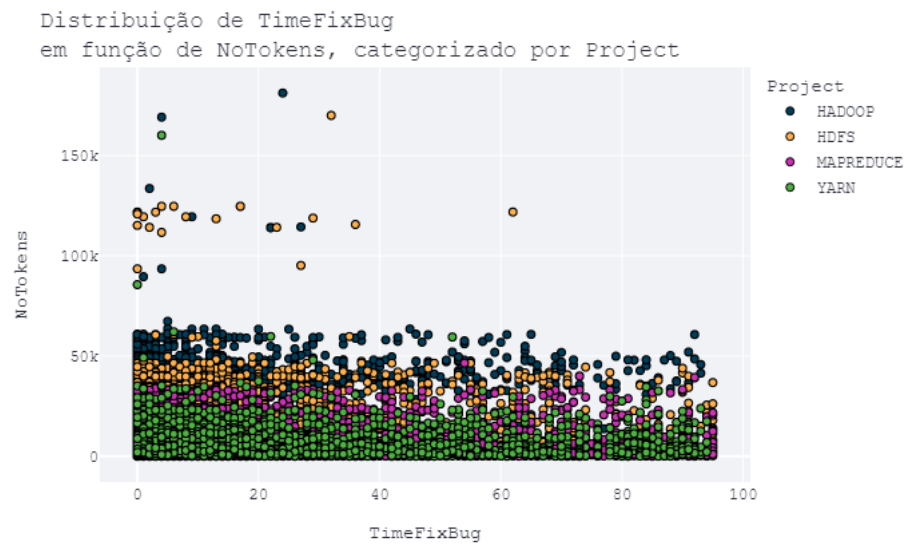


Figure 20 – Gráfico de dispersão dos bugs em relação ao tempo de resolução e número de tokens nos arquivos de código-fonte envolvidos. Pontos de dispersão coloridos em função de Projetos.

Pelos gráficos de dispersão exibidos não notamos nenhuma relação linear entre as variáveis numéricas.

### 5.5 Desenvolvedores mais engajados realizam atividades de maior complexidade em menor tempo comparado aos desenvolvedores menos engajados?

Ao observarmos a distribuição de atividades levando em conta a complexidade pela categoria de engajamento, notamos que para os três níveis de engajamentos tivemos uma maior presença dos desenvolvedores em issues de nível Major.

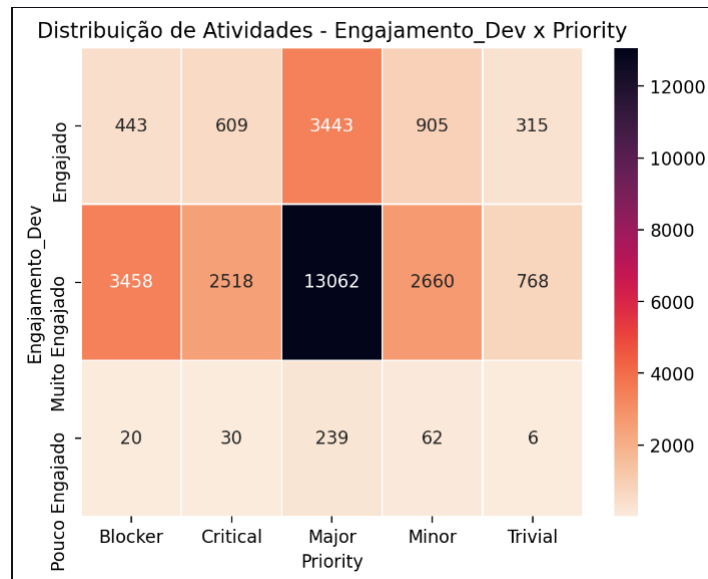


Figure 21 – Gráfico de distribuição dos bugs em relação ao grupo de engajamento e prioridade

Temos 239 tarefas de prioridade Major feitas pelo grupo de desenvolvedores pouco engajados, 3443 tarefas de prioridade Major desenvolvidas pelo grupo engajado e 13062 de atividades de nível Major desenvolvidas pelo grupo muito engajado. Apesar de termos uma discrepância entre a quantidade de atividades realizadas pelos grupos, como para os três grupos temos uma maior presença de desenvolvedores em issues da prioridade Major comparando com as outras prioridades, utilizaremos o nível Major para comparar o tempo de resolução entre os grupos de desenvolvedores.

A seguir colocamos a distribuição de tempo de solução do bug para cada um dos grupos em relação a prioridade.

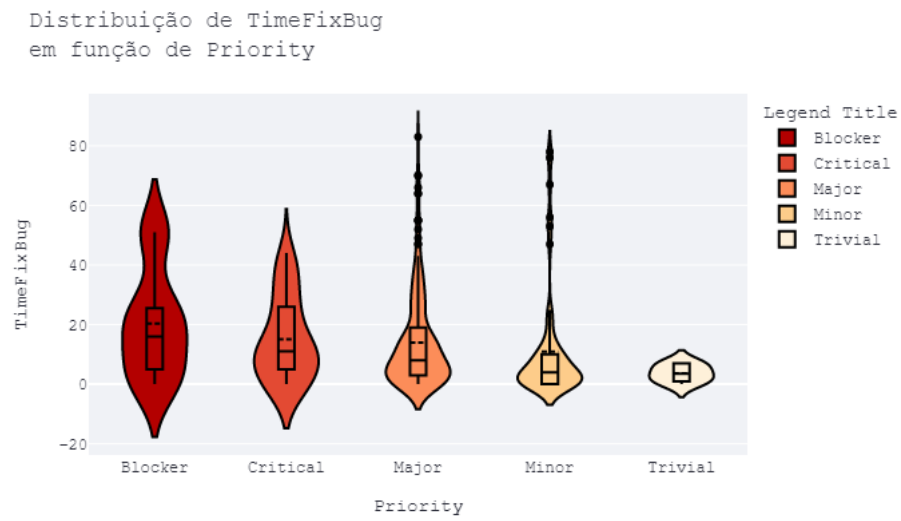


Figure 22 – Gráfico de distribuição do tempo de resolução dos bugs em relação à Prioridade para o grupo de desenvolvedores Pouco Engajados.

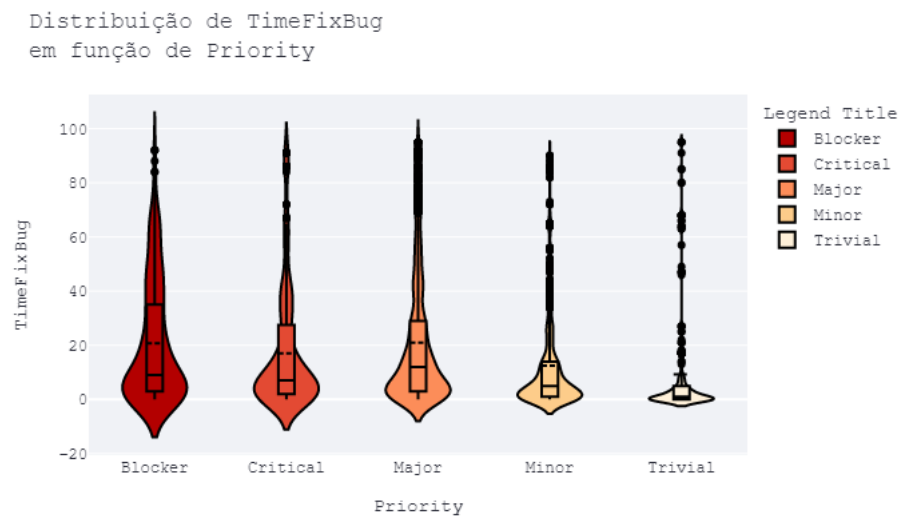


Figure 23 – Gráfico de distribuição do tempo de resolução dos bugs em relação à Prioridade para o grupo de desenvolvedores Engajados.

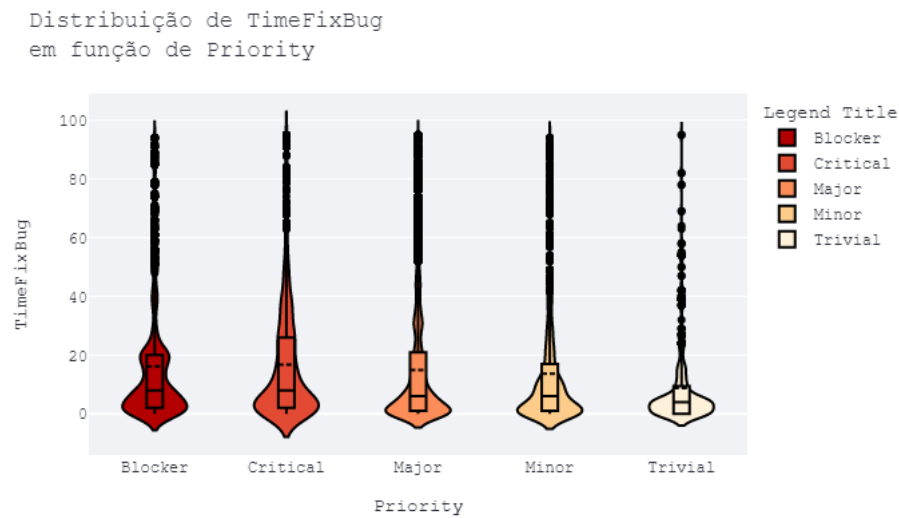


Figure 24 – Gráfico de distribuição do tempo de resolução dos bugs em relação à Prioridade para o grupo de desenvolvedores Muito Engajados.

Focalizando nas atividades de prioridade Major, notamos que até 75% das atividades realizadas por desenvolvedores muito engajados demandaram até 21 dias e 50% levaram até 6 dias para serem resolvidos. Já para o grupo de engajados, 75% das atividades foram resolvidas em até 29 dias e 50% demandaram até 12 dias. Por fim, os desenvolvedores pouco engajados demandaram até 19 dias em 75% das atividades e até 8 dias em 50% das tarefas para serem resolvidas.

A título de curiosidade, para as atividades de nível Blocker, os desenvolvedores muito engajados demandaram até 8 dias para resolver as demandas em 50% dos casos, enquanto os desenvolvedores engajados e pouco engajado levaram, respectivamente, até 9 dias e 16 dias para resolverem 50% das issues em que foram alocados.

## 6 CONCLUSÕES E TRABALHOS FUTUROS

Nesta análise vimos que desenvolvedores mais engajados nos projetos possuem, normalmente, um tempo de resolução de bug menor, em comparação aos desenvolvedores menos engajados. Além disso, apesar do grupo de desenvolvedores muito engajados ser o que possui menor quantidade de profissionais, estes são responsáveis pela maior parte das resoluções de bugs.

Não foi notório nenhuma relação direta entre Número de Comentários, Número de Desenvolvedores e Número de Commits com o tempo de solução do bug. No entanto, observamos que nos bugs que possuíam mais de 20 comentários possuíam uma maior concentração de ocorrências de tempo de resolução maior do que os com menos comentários.

Notamos que bugs com nível Major, Critical ou Blocker possuem um maior envolvimento, com uma maior quantidade de comentários em comparação ao Minor e Trivial. No entanto, de modo geral, a maioria dos bugs tiveram apenas 1 autor envolvido.

No geral, desenvolvedores muito engajados possuem um tempo de solução menor para atividades de prioridades e complexidade maior em comparação aos outros grupos de desenvolvedores.

Com esse estudo, notamos que os desenvolvedores mais engajados possuem um alto poder de contribuição ao projeto de software livre, mesmo sendo o grupo com menor número de pessoas. É fundamental a comunidade envolver cada vez mais esses desenvolvedores e estes desenvolvedores buscarem envolver desenvolvedores menos engajados nas atividades. As atividades com maior complexidade normalmente já puxam um maior envolvimento da comunidade, então seria interessante trazer desenvolvedores menos engajados para trabalharem juntos com desenvolvedores mais engajados. Isso pode fortificar a comunidade e auxiliar na solução de bugs de maior complexidade, assim como no tempo de solução de bugs com uma menor prioridade.

## REFERENCES

GOOSSENS, M.; MITTELBAACH, F.; SAMARIN, A.; SOUIDI, E. M. **The LATEX companion.**  
[S. l.]: Addison-Wesley Reading, Massachusetts, 1994.



## **APPENDIX A – EXEMPLO DE APÊNDICE**

Um apêndice é um documento elaborado pelo autor, diferentemente do anexo. Geralmente, se coloca como apêndice, questionários, códigos de programação, tabelas que tomariam muito espaço no meio do trabalho. Artigos, resumos ou qualquer publicação relacionada ao trabalho podem ser utilizados como apêndice.

## APPENDIX B – QUESTIONÁRIO UTILIZADO PARA...

**Questão 1.** Esta é a primeira questão com alguns itens:

- (a) Este é o primeiro item
- (b) Segundo item

**Questão 2.** Esta é a segunda questão:

- (a) Este é o primeiro item
- (b) Segundo item

**Questão 3.** Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc dictum sed tortor nec viverra. consectetur adipiscing elit. Nunc dictum sed tortor nec viverra.

- (a) consectetur
- (b) adipiscing
- (c) Nunc
- (d) dictum

## APPENDIX C – CÓDIGOS-FONTES UTILIZADOS PARA...

### Código-fonte 1 – Hello World em C++

```
1 #include <iostream>
2 using namespace std;
3 int main() {
4     cout<<"Hello World!"<<endl;
5     system("pause");
6 }
```

### Código-fonte 2 – Hello World em Java

```
1 public class HelloWorld {
2     public static void main(String[] args) {
3         System.out.println("Hello World!");
4     }
5 }
```

**APPENDIX D – IEEE CEFC 2016**

*Digest submetido ao The 17th Biennial Conference on Eletromagnetic Field Computation, Miami FL - NOV 13-16, 2016, USA.*

# Lightning Incidence Model Based on the Electric Field Gradient: 3D Electrostatic Analyses

Ednardo M. Rodrigues, Ricardo S. T. Pontes and Tobias R. Fernandes Neto

Federal University of Ceará, Department of Electrical Engineering, Fortaleza CE, BRAZIL  
ednardorodrigues@dee.ufc.br

**Abstract**— The paper deals with the 3D electrostatic analysis of a lightning strike in a hangar and a power transmission line. The lightning incidence model is based on the electric field gradient. Finally, the simulation results are described and discussed.

**Index Terms**—Lightning, Electrostatic, Finite element.

## I. INTRODUCTION

In [1], a 2D electrostatic analysis of a new lightning incidence model based on the electric field gradient (EFG) was presented. Moreover, the simulations results were carried out for a building and a power transmission line and they were compared with the classical electrogeometrical model (EGM), the rolling-sphere technique (RST) and the leader progression model (LPM) [2]. The present paper estimates the trajectory of lightning strikes from the thundercloud to a grounded metal roof of a hangar. Furthermore, the same procedure will be carried out for 500kV power transmission lines.

## II. ELECTROSTATIC ANALYSES

A lightning occurs when the electric field is higher than the breakeven field (400kV/m —3MV/m) [2]. This model is based on the electric field gradient described by

$$\vec{E}_L(\vec{r}) \approx \vec{E}_b(\vec{r}) + \lambda_t \nabla E_b(\vec{r}), \quad (1)$$

$\vec{E}_b(\vec{r})$  is the background electric field, which is function of the position  $\vec{r}$  and it is generated by the electric potential difference (EPD) between the cloud and the ground.  $\lambda_t$  is the lightning step length (~50m) [3], and  $\vec{E}_L(\vec{r})$  is the lightning electric field. More details about Eq. (1) can be found in [1].

A 3D finite element method (FEM) model of a hangar and a power transmission line (TL) were designed by using the electrostatic module. All simulations were carried out within a cubic domain of 250m x 250m x 250m. The upper level of each domain is defined with -12.5MV, while the lower level is the ground. This is equivalent to a real thundercloud with a potential of (-100MV) at 2km of altitude [4].

The dimensions of the hangar are: 8.60m height, 77.37m width and 229.00m length. The aluminum metal roof has 0.7mm thickness and it is grounded. The second simulation is for a TL composed by three phase conductors, equally spaced by 11.5m and positioned at 40.5m above the reference plane. The TLs are protected by two earth wires spaced by 19m over 54.47m of the reference plane.

## III. RESULTS

In order to evaluate the 3D model, the software COMSOL Multiphysics® was used in a computer with quad-core processor of 2.6GHz. For the hangar, the simulation time was around 4s. The necessary physical memory for the simulation

was 1.34GB and 5.6GB of virtual memory. The electric field is very intense at the roof (about 80MV/m) and the lightning (cyan lines) strikes the building roof, as shown in Fig. 1a. In summary, it is not necessary to add air terminals, as long the roof is grounded. The simulation time for the TL was around 6 min, using 15GB of physical memory and 32GB of virtual memory. As shown in Fig 1b, the cyan lines strike the earth wires.

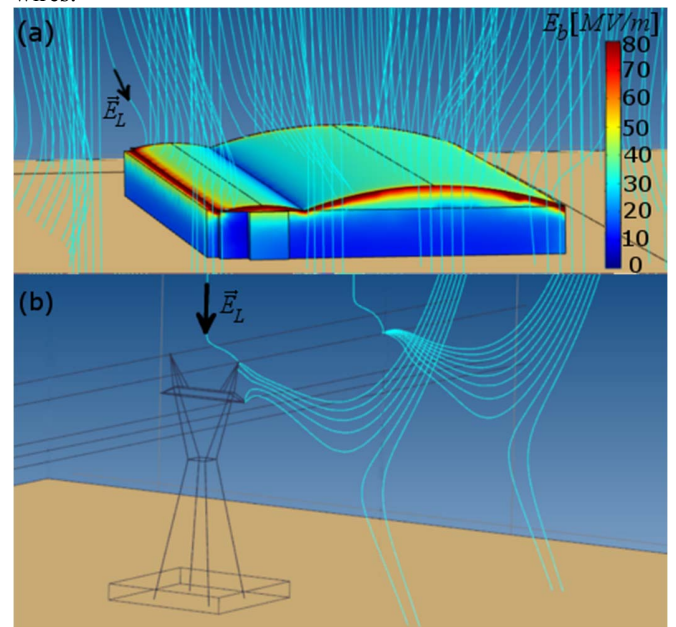


Fig. 1. Case of studies: (a) hangar and (b) power transmission line.

## IV. CONCLUSIONS

The EFG simulations predicted that the aluminum metal roof is able to protect the hangar against lightning strikes. In the TL simulation, the earth wires have fulfilled the protection for the phase conductors. Finally, the protection zone and the design of lightning protection system can be evaluated by 3D electrostatic analyses, which are closer to the reality than the 2D analyses. However, 3D models are often more complex and require more simulation time.

## REFERENCES

- [1] E. M. Rodrigues, *Novel Lightning Incidence Model Based on the Electric Field Gradient: 2D Electrostatic Analyses*. GROUND'2016 & 7<sup>th</sup> LPE, 2016.
- [2] V. Cooray, *Lightning protection*, The Institution of Engineering and Technology, 2009.
- [3] V.A. Rakov and M. A. Uman, *Lightning: physics and effects*. Cambridge University Press, 2007.
- [4] S. Visacro, *Descargas atmosféricas: Uma Abordagem de Engenharia (Lightning strike: An Engineering approach)*, Artliber, 2005.

## ANNEX A – EXEMPLO DE UM ANEXO

Um anexo é um documento que não foi elaborado pelo autor, ou seja, o autor apenas anexa. Anexos podem ser tabelas, mapas, diagramas, *datasheets*, manuais e etc.

## **ANNEX B – EXEMPLO DE UM ANEXO EM PDF**

O autor pode anexar um *Portable Document Format* (PDF), traduzido como formato portátil de documento. Veja o código fonte utilizado para anexar o arquivo “Sikasil.pdf” que foi colocado dentro da pasta “anexos” que por sua vez está dentro da pasta “elementos-pos-textuais”. Tenha muita atenção na hora de especificar o local do arquivo. Recomenda-se não utilizar caracteres especiais para nomear pastas e, principalmente, arquivos.

Pode-se fazer uma descrição sucinta do arquivo anexado.

**Sikasil® GS-630**

Glazing sealant for structural &amp; non-structural use

## Technical Product Data

Chemical base	1-C silicone
Color (CQP <sup>1</sup> 001-1)	See Product Overview
Cure mechanism	Moisture-curing
Cure type	Neutral
Density (uncured) (CQP 006-4)	1.4 kg/l approx.
Non-sag properties (CQP 061-4 / ISO 7390)	< 2 mm approx.
Application temperature	5 - 40°C (41 - 104°F)
Skin time <sup>2</sup> (CQP 019-2)	10 min approx.
Tack-free time <sup>2</sup> (CQP 019-1)	60 min approx.
Curing speed (CQP 049-1)	See diagram 1
Shore A-hardness (CQP 023-1 / ISO 868)	32 approx.
Tensile strength (CQP 036-1 / ISO 37)	1.2 N/mm <sup>2</sup> approx.
Elongation at break (CQP 036-1 / ISO 37)	480% approx.
Tear propagation resistance (CQP 045-1 / ISO 34)	6 N/mm approx.
100% modulus (CQP 036-1 / ISO 37)	0.6 N/mm <sup>2</sup> approx.
Movement accommodation capability (ASTM C 719)	±50%
Thermal resistance (CQP 513-1)	long term
Short term	4 h
	1 h
Service temperature	-40 - 150°C approx. (-40 - 302°F)
Shelf life (storage below 25°C) (CQP 016-1)	15 months

<sup>1)</sup> CQP = Corporate Quality Procedure<sup>2)</sup> 23°C (73°F) / 50% r.h.**Description**

Sikasil® GS-630 is a durable, neutral-curing silicone sealant and adhesive which combines mechanical strength with high elongation. It adheres excellent to a wide range of substrates.

Sikasil® GS-630 is manufactured in accordance with ISO 9001 quality assurance system and the responsible care program.

**Product Benefits**

- Outstanding UV and weathering resistance
- Excellent adhesion to glass, coated glass, metals and plastics
- Fast curing
- Long-term durability
- High movement capability

**Areas of Application**

Sikasil® GS-630 is a silicone sealant and adhesive designed for sealing, bonding and mending tasks in a wide variety of industrial applications, e. g. structural and nonstructural applications in facades.

This product is suitable for professional experienced users only. Tests with actual substrates and conditions have to be performed to ensure adhesion and material compatibility.





## Cure Mechanism

Sikasil® GS-630 cures by reaction with atmospheric moisture. The reaction thus starts at the surface and proceeds to the core of the joint. The curing speed depends on the relative humidity and the temperature (see diagram 1 below). Heating above 50°C to speed-up the vulcanization is not advisable as it may lead to bubble formation. At low temperatures the water content of the air is lower and the curing reaction proceeds more slowly.

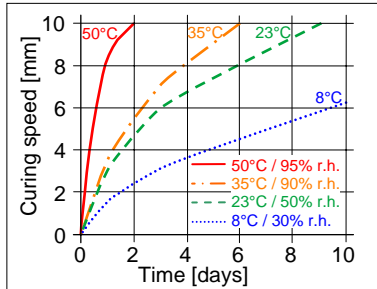


Diagram 1: Curing speed 1C-Sikasil®

## Application Limits

All Sikasil® WS, FS, SG, IG, WT and other engineering silicone sealants and adhesives are compatible with each other. Sikasil® WS and FS sealants as well as other Sika engineering silicone sealants are compatible with SikaGlaze® IG sealants. All other sealants have to be approved by Sika before using them in combination with Sikasil® GS-630. Where two or more different reactive sealants are used, allow the first to cure completely before applying the next.

Do not use Sikasil® GS-630 on pre-stressed polyacrylate and polycarbonate elements as it may cause environmental stress cracking (crazing).

The compatibility of gaskets, backer rods and other accessory materials with Sikasil® GS-630 must be tested in advance. Joints deeper than 15 mm should be avoided.

The above information is offered for general guidance only. Advice on specific applications will be given on request.

## Method of Application

### Surface preparation

Surfaces must be clean, dry and free from oil, grease and dust.

Advice on specific applications and surface pretreatment methods is available from the Technical Service Department of Sika Industry.

### Application

After suitable joint and substrate preparation, Sikasil® GS-630 is gunned into place. Joints must be properly dimensioned as changes are no longer possible after construction. For optimum performance the joint width should be designed according to the movement capability of the sealant based on the actual expected movement. The minimum joint depth is 6 mm and a width / depth ratio of 2:1 must be respected if used for weatherproofing. For backfilling it is recommended to use closed cell, sealant compatible foam backer rods e.g. high resilience polyethylene foam rod. If joints are too shallow for backing material to be employed, we recommend using a polyethylene tape. This acts as a release film (bond breaker), allowing the joint to move and the silicone to stretch freely.

For more information please contact the Technical Service Department of Sika Industry.

### Tooling and finishing

Tooling and finishing must be carried out within the skin time of the adhesive.

When tooling freshly applied Sikasil® GS-630 press the adhesive to the joint flanks to get a good wetting of the bonding surface.

### Removal

Uncured Sikasil® GS-630 may be removed from tools and equipment with Sika® Remover-208 or another suitable solvent. Once cured, the material can only be removed mechanically.

Hands and exposed skin should be washed immediately using Sika® Handclean Towel or a suitable industrial hand cleaner and water. Do not use solvents!

### Overpainting

Sikasil® GS-630 cannot be over-painted.

## Further Information

Copies of the following publications are available on request:

- Material Safety Data Sheet

## Packaging Information

Unipack	600 ml
Cartridge	300 ml
Pail	26 kg
Drum	280 kg

## Value Bases

All technical data stated in this Product Data Sheet are based on laboratory tests. Actual measured data may vary due to circumstances beyond our control.

## Health and Safety Information

For information and advice regarding transportation, handling, storage and disposal of chemical products, users should refer to the actual Material Safety Data Sheets containing physical, ecological, toxicological and other safety-related data.

## Legal Notes

The information, and, in particular, the recommendations relating to the application and end-use of Sika products, are given in good faith based on Sika's current knowledge and experience of the products when properly stored, handled and applied under normal conditions in accordance with Sika's recommendations. In practice, the differences in materials, substrates and actual site conditions are such that no warranty in respect of merchantability or of fitness for a particular purpose, nor any liability arising out of any legal relationship whatsoever, can be inferred either from this information, or from any written recommendations, or from any other advice offered. The user of the product must test the product's suitability for the intended application and purpose. Sika reserves the right to change the properties of its products. The proprietary rights of third parties must be observed. All orders are accepted subject to our current terms of sale and delivery. Users must always refer to the most recent issue of the local Product Data Sheet for the product concerned, copies of which will be supplied on request.

Further information available at:

[www.sika.ch](http://www.sika.ch)

[www.sika.com](http://www.sika.com)

Sika Schweiz AG

Industry

Tüffenwies 16

CH-8048 Zurich

Switzerland

Tel. +41 44 436 40 40

Fax +41 44 436 45 30

