

An Exploratory Study on Developers Opinions about Influence in Open Source Software Ecosystems

Vinicius Condina, Paulo Malcher,
Victor Farias, Rodrigo Santos
UNIRIO

Rio de Janeiro - Brasil
vinicius.lima@uniriotec.br, malcher@edu.unirio.br,
victor.farias@uniriotec.br, rps@uniriotec.br

Awdren Fontão
UFMS

Campo Grande - Brasil
awdren.fontao@ufms.br

Igor Wiese
UTFPR

Campo Mourão - Brasil
igor@utfpr.edu.br

Davi Viana
UFMA

São Luís - Brasil
davi.viana@lsdi.ufma.br

ABSTRACT

Software Engineering researchers and practitioners have sought a systematic way of characterizing technical and social behaviors in software development, from the perspective of either an artifact (e.g. design pattern) or an actor (e.g. developer). In this context, influencers are those who guide the development methods and disseminate patterns due to their popularity or status in the community. Especially in software ecosystems, the dynamics of interactions with interventions of external developers over a common technological platform leverage the effects of influence and still remains as a challenge. This paper presents an exploratory study on the sense of influence in open source software ecosystems, more specifically from the opinions of 95 developers who contribute to npm projects based on GitHub. To do so, qualitative data from a conducted survey research were analyzed based on Grounded Theory (GT) procedures. Based on two main categories (Technical and Social), we recognized aspects that reinforce some key characteristics of an influencer, e.g., 'status in the project', 'participation with code', 'participation with comments', and 'content value'. However, some diverging aspects were observed, e.g., the difference between qualitative and quantitative responses related to 'long-time interaction with the project'. This study contributes with the identification of influencers' characteristics that can aid both researchers in further studies on open source software ecosystems and practitioners in strategies to manage project-based ecosystems.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
SBES'20, October 21-23, 2020, Natal, RN, Brazil

© 2020 Copyright held by the owner/author(s). 978-1-4503-0000-0/18/06...\$15.00
<https://doi.org/...>

CCS CONCEPTS

- Software and its engineering → Software creation and management → Collaboration in software development

KEYWORDS

Software Ecosystems, Open Source, Influence, Social Developer, GitHub, npm

ACM Reference Format:

Vinicius Condina, Paulo Malcher, Victor Farias, Rodrigo Santos, Awdren Fontão, Igor Wiese, and Davi Viana. 2020. An Exploratory Study on Developers Opinions about Influence in Open Source Software Ecosystems. In XXXIV Brazilian Symposium on Software Engineering (SBES 2020), October 21–23, 2020, Natal, RN, Brazil. ACM, New York, NY, USA, 10 pages. DOI: <https://doi.org/...>

1 Introdução

O desenvolvimento de software tem se transformado e passou a ser caracterizado não só pelo aspecto técnico, mas também pelo caráter social das interações entre desenvolvedores [1]. Além disso, as mudanças ocorridas nos modelos de negócio de software, como novos domínios de aplicação e diferentes tipos de nichos de usuários, fizeram com que as organizações diminuíssem suas fronteiras. Essas interações entre os aspectos técnicos, sociais e de negócios e as mudanças decorrentes têm favorecido o surgimento de ecossistemas de software (ECOS). Manikas [2] define ECOS como conjunto de atores e artefatos, internos e externos a uma organização ou comunidade, que trocam recursos e informações centrados em uma plataforma tecnológica comum. Dentre os aspectos sociais em um ECOS, pode se destacar o poder da influência entre os atores. O termo influência pode ser entendido como “*um tipo de poder, onde ter poder sobre alguém é fazer com que alguém faça algo que de outra forma não faria*” [3]. Nesse sentido, os múltiplos produtos

desenvolvidos em um ECOS podem ser afetados pela influência exercida pelos atores.

De acordo com um mapeamento sistemático no assunto [4], influenciadores são atores que podem começar a liderar o desenvolvimento e ditar como o projeto de software progredirá em um ECOS. Por exemplo, um influenciador pode guiar o ecossistema rumo ao uso de boas práticas de desenvolvimento, ou atuar no gerenciamento da comunidade de software a fim de atrair novos contribuidores, fazendo com que o ECOS prospere. Por outro lado, um influenciador pode impactar negativamente o ecossistema, caso seu comportamento social e/ou influência técnica não sejam uma referência adequada para a comunidade. Em outras palavras, a influência positiva pode levar todo o ecossistema ao estado de sucesso, ao passo que a influência negativa pode levar o ECOS ao fracasso [5, 6]. Por esta razão, compreender melhor os influenciadores se torna relevante para entender a evolução de um ECOS [7], além de possibilitar entender a importância deste papel para o caráter dinâmico oriundo da participação de atores externos no ecossistema [4].

Em ecossistemas mais abertos – como os ECOS *open source* –, a diversidade de projetos relacionados, o número de atores e a dinâmica de migrações podem dificultar que desenvolvedores tenham noção do nível de influência que exercem nos projetos [1, 3, 5, 7]. Por exemplo, é o caso de ECOS envolvendo diversos projetos com objetivos comuns construídos na plataforma GitHub¹, como acontece no ecossistema *npm*² (*Node Package Manager*). Por haver um controle menos rígido, esses ECOS disponibilizam conjuntos de dados das interações entre os desenvolvedores, possibilitando a análise de vários conteúdos como *commits*, solicitações de mudanças (*pull_requests*) e tarefas (*issues*) [8, 9, 10]. Segundo estudos prévios da literatura [4, 7, 11, 12], esses conteúdos podem auxiliar na identificação e caracterização de um influenciador em um ecossistema.

Este trabalho apresenta um estudo exploratório sobre o senso de influência em ECOS *open source*, mais especificamente a partir das opiniões de 95 desenvolvedores que contribuem para projetos no ECOS *npm*. Para isso, dados qualitativos de uma pesquisa de opinião conduzida foram analisados a partir de procedimentos de *Grounded Theory* (GT). Esta pesquisa envolveu 242 desenvolvedores do ECOS *npm* avaliando quantitativamente oito características de influenciadores em ECOS identificadas em um mapeamento sistemático da literatura [7]. Para o presente trabalho, pretende-se entender com maior profundidade as questões quantitativas do estudo e responder à seguinte questão de pesquisa: *Qual a opinião de desenvolvedores sobre a sua influência em um ECOS open source?*

Este trabalho contribui para entender como aspectos técnicos e sociais estão intrinsecamente relacionados à noção de influência em ECOS *open source*, considerando o cenário selecionado. Além disso, permitiu analisar detalhadamente as categorias e características de um influenciador a partir das opiniões dos desenvolvedores. A análise qualitativa mostrou

que as opiniões dos desenvolvedores corroboram fortemente características como *status* no projeto, participação com código, valor do conteúdo e participação com comentários. Por outro lado, há alguns aspectos divergentes, tais como a percepção dos integradores (uma das duas classes de participantes) sobre o 'valor do conteúdo' e a percepção dos desenvolvedores em geral sobre o 'longo tempo de interação com o projeto'. A partir deste trabalho, projetos de software e ecossistemas podem ter insumos para mecanismos mais específicos para identificar os influenciadores que guiam evolução do ECOS, ao passo que os desenvolvedores podem compreender melhor a respeito das características que envolvem quem influencia o projeto para o qual estão desenvolvendo bem como o ECOS onde atuam.

Este artigo está organizado da seguinte forma: a Seção 2 apresenta a fundamentação teórica. Na Seção 3, é discutida a metodologia. Na Seção 4, os resultados da análise qualitativa são mostrados, cuja discussão é feita na Seção 5. A Seção 6 traz as implicações para a prática e a Seção 7, as limitações. Por fim, a Seção 8 conclui o artigo e indica trabalhos futuros.

2 Ecossistemas de Software e Influência

Um ECOS pode ser analisado na perspectiva de projetos como: *"grupos de projetos que são desenvolvidos e coevoluem no mesmo ambiente"* [12]. Tal ambiente pode ser uma empresa ou um repositório de código aberto. É possível ainda estabelecer uma visão em três dimensões de ECOS, inicialmente distinguida por Campbell e Ahmed [13] e desenvolvida por Santos e Werner [14], dividida em: (i) dimensão técnica (i.e., base de tecnologia, infraestrutura ou organização), (ii) dimensão de negócios (i.e., modelos para gestão de artefatos, recursos e informações) e (iii) dimensão social (i.e., redes de influência e conhecimento). Neste contexto, segundo Hoving *et al.* [15], os atores de um ECOS podem ser definidos por certas características, como tipos de atores, papéis e conexões estabelecidas nestes ambientes. Por sua vez, para Lima *et al.* [16], um ator pode ter papéis específicos em um ECOS, caracterizando-se como uma empresa ou outro tipo de organização, um desenvolvedor, um usuário final, um fornecedor ou um cliente e, de maneira geral, pode abranger quaisquer outros envolvidos ou interessados.

A partir dessa visão, pode-se tentar entender a evolução de um ECOS a partir dos efeitos de ações de atores com papel de influenciador, analisando principalmente a dimensão social, bem como relações com a dimensão técnica. Segundo Kirkhart [17], influência pode ser definida como *"a capacidade ou poder de pessoas ou coisas de produzir efeitos sobre os outros por meios intangíveis ou indiretos"*. Por sua vez, conforme Dictionary.com [18], influência é *"a capacidade ou poder de pessoas ou coisas de serem força convincente ou produzirem efeitos sobre as ações, comportamentos ou opiniões"*. Desta forma, em ECOS, a influência pode ser compreendida como um tipo e nível de poder de um ator sobre outro que sofre influência em suas

¹ www.github.com

² <https://github.com/npm/cli>

interações [3]. Nesse sentido, é importante investigar não só a interação ator-ator, mas também a interação ator-artefato [10].

De acordo com Hagel *et al.* [19], um membro da comunidade de um ECOS pode ser também um influenciador, o que não é reservado somente à organização que gere a plataforma tecnológica comum. Os autores afirmam que um influenciador interfere na qualidade dos artefatos, constrói capacidades e ganha forte posição de mercado ao direcionar desenvolvedores. Por sua vez, Silva *et al.* [20] resumiram os papéis de atores de ECOS e destacam a importância do influenciador: desenvolve produtos para o ECOS e contribui para sua evolução, mantendo-o ativo e inspirando relacionamentos de atores e artefatos ao se comprometer com a estratégia de gestão da organização ou da comunidade. Um influenciador não só participa ativamente do ECOS, mas também exerce sua influência sobre a organização central [21], ou pode servir como líder em comunidades locais, expandindo a estrutura organizacional [10].

Como trabalhos relacionados, Blincoe *et al.* [22] estudaram a motivação para seguir (ou não) outras pessoas e a influência de usuários populares em seus seguidores no GitHub. Como resultado, os autores descobriram que usuários populares atraem seus seguidores para novos projetos e que, à medida que a popularidade de um usuário aumenta, a sua taxa de influência também aumenta. Porém, Blincoe *et al.* [22] não identificaram características de um influenciador. Por sua vez, Valença e Alves [3] apresentam uma teoria substantiva para explicar como o poder e a dependência se manifestam em um ECOS e descrevem elementos estruturais. A teoria proposta destaca as interações entre as diferentes formas de poder e fornece uma melhor compreensão de como o poder e a dependência influenciam o comportamento de um ecossistema. Todavia, essa influência não é apresentada especificamente na perspectiva de desenvolvedores que a exercem, mas considera pequenas e médias empresas. Além disso, o trabalho dos autores não realizou análise no domínio de ECOS *open source*.

No contexto de ECOS *open source*, a estrutura da equipe de desenvolvimento *Free/Libre Open Source Software* (FLOSS) foi discutida por Crowston *et al.* [23]. Nesta visão, os integradores (*committers*) são desenvolvedores principais (i.e., aqueles que contribuem com frequência e gerenciam a evolução do projeto) e os colaboradores (*collaborators*) são co-desenvolvedores (i.e., aqueles que contribuem esporadicamente com a revisão de código e correções de *bugs*). Dado que identificar influenciadores em um ECOS pode ser uma atividade complexa, torna-se relevante avançar na descoberta e análise de influenciadores em ECOS para que gestores de ecossistemas e/ou de projetos possam entender melhor quais (e como) os atores estão afetando a evolução do ecossistema, o que depende de aspectos técnicos, mas também de aspectos sociais [1].

3 Pesquisa de Opinião

Pesquisas de opinião são executadas com o objetivo de obter conhecimentos das pessoas, buscando entender determinado

aspecto de uma população [24]. Neste trabalho, o objetivo é entender melhor o senso de influência em ECOS *open source* a partir das opiniões de 95 desenvolvedores que contribuem para projetos no ECOS *npm*. Este trabalho se baseia em uma pesquisa de opinião conduzida e visa analisar, de maneira qualitativa, as justificativas dadas por desenvolvedores quanto à sua opinião sobre o nível de influência que exercem nos projetos que estão envolvidos neste ecossistema.

Inicialmente, a pesquisa envolveu 242 desenvolvedores do ECOS *npm* avaliando quantitativamente oito características de influenciadores em ECOS identificadas em um mapeamento sistemático da literatura [7]. Essas características consideradas relevantes para definir influenciadores em ecossistemas são: (i) proximidade com o dono do projeto do GitHub; (ii) longo tempo de interação com o projeto; (iii) *status* no projeto; (iv) *status* (popularidade no GitHub); (v) participação com comentários; (vi) participação com código; (vii) fonte de conhecimento; e (viii) valor do conteúdo.

As cinco primeiras são características classificadas como sociais, enquanto que as três últimas são características técnicas. A análise qualitativa, nesse sentido, pode ajudar a entender os motivos que levaram os desenvolvedores a atribuírem um determinado nível de influência e relacioná-lo às características dos influenciadores levantadas anteriormente.

3.1 Planejamento

A pesquisa coletou informações sobre a opinião de desenvolvedores acerca do seu senso de influência no ECOS *npm*. As razões pelas quais o ECOS *npm* foi escolhido estão associadas ao número de projetos e desenvolvedores que contribuem transversalmente entre eles, que fazem parte deste ecossistema, além do fato do *npm* e seus projetos já terem sido definidos como um ecossistema na literatura [25]. A pesquisa de opinião foi realizada por meio de um questionário *online* enviado para duas classes de participantes em projetos *open source*, explicadas na Seção 2: integradores e colaboradores.

No escopo deste trabalho, buscou-se realizar a análise qualitativa dos resultados a partir da seguinte questão: “Qual o seu nível de influência nos projetos que você está envolvido? Justifique sua resposta”. Nesta questão, os participantes informaram o seu nível de influência nos projetos (valores inteiros entre 1 e 10, onde 1 significa nível de influência muito baixo e 10 significa nível muito alto) e obrigatoriamente justificar sua resposta (estilo de questão aberta). O conjunto de respostas fornece apoio para responder à questão de pesquisa principal deste estudo que foi apresentada na Seção 1.

Os desenvolvedores foram divididos em grupos: um com desenvolvedores 'integradores' e outro com desenvolvedores 'colaboradores'. Como colaboradores, foram considerados os desenvolvedores que contribuíram com pelo menos um *pull request* para um projeto do ECOS *npm*. Por sua vez, como integradores, foram considerados aqueles que realizavam o processo de revisão do código e conseqüentemente aceitavam ou rejeitavam *pull request* em um dado projeto no ECOS *npm*.

3.2 Execução da Pesquisa de Opinião

A pesquisa foi enviada por e-mail para 3.419 desenvolvedores, sendo 3.260 colaboradores, com uma taxa de resposta de 6,8% (224), e 159 integradores, com taxa de resposta de 11,3% (18). Para a questão analisada neste trabalho, apenas 95 respostas foram consideradas válidas (85 de colaboradores e 10 de integradores). Isso se deve ao fato de que muitas respostas possuíam apenas um espaço em branco ou um outro caractere no campo obrigatório de justificativa.

3.3 Procedimentos de Análise

Segundo Angrosino [26], a pesquisa qualitativa tem o objetivo de entender, descrever e explicar fenômenos sociais de diferentes formas. Para a análise qualitativa apresentada neste trabalho, foram utilizados procedimentos de *Grounded Theory* (GT) [27]. GT objetiva desenvolver uma teoria fundamentada em dados. Este método pode ser dividido em três codificações: (1) aberta, (2) axial e (3) seletiva.

A codificação aberta é a primeira fase do processo de análise de dados, quando os dados são codificados, comparados com outros dados e classificados em categorias. Inicialmente, é feita uma leitura detalhada das respostas e cada trecho relevante do texto recebe um código ou categoria, que pode ser formado por uma palavra, frase ou expressão. A codificação axial consiste no aprimoramento e diferenciação das categorias resultantes da codificação aberta, buscando-se identificar os relacionamentos existentes entre elas. Por fim, na codificação seletiva, a categoria ou ideia central do estudo é originada.

O processo de codificação realizado neste trabalho utilizou apenas as codificações aberta e axial para analisar os dados, visando à identificação da percepção do nível de influência que os desenvolvedores dos projetos do ECOS *npm* acreditam possuir (e a justificativa). Strauss e Corbin [27] explicam que o pesquisador pode utilizar apenas alguns passos para atingir seu objetivo de pesquisa como, por exemplo, quando precisam compreender determinado fenômeno ou situação. Além disso, o processo foi verificado por dois pesquisadores de Engenharia de Software com experiência de 8 e 10 anos em ECOS. Por fim, o processo foi validado por outros dois pesquisadores com experiência de 5 e 7 anos em estudos com dados qualitativos.

3.4 Condução da Análise dos Dados

A análise qualitativa foi realizada com base nas justificativas das respostas obtidas para a questão da pesquisa de opinião. Nesta análise, foram aplicadas as codificações abertas e axial, conforme mencionado na Seção 3.3. Na codificação aberta, os dados foram analisados detalhadamente para a associação de trechos a códigos que remetem às características identificadas em um trabalho anterior [7], como contextualizado no início da Seção 3. Este procedimento é exemplificado na Figura 1. O número ao lado do nome do código representa a quantidade de trechos que foram relacionados a uma dada característica.

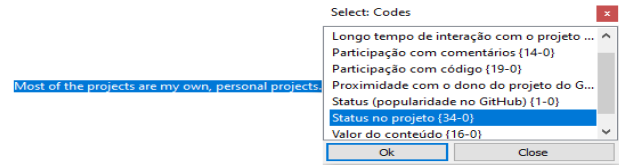


Figura 1: Exemplo de código na ferramenta Atlas.TI

Após a codificação aberta, os códigos foram relacionados às categorias. As categorias são agrupamentos de conceitos em um grau de abstração mais alto. As representações gráficas foram criadas com apoio da ferramenta Atlas.TI³. Códigos diretamente relacionados às categorias (ou famílias, de acordo com Atlas.TI) são mostrados por ligação vermelha (Figura 3).

4 Resultados

4.1 Demografia dos Participantes e Resultados Quantitativos

A seguir, são apresentadas as características demográficas de todos os participantes (242 desenvolvedores) e o resultado quantitativo sobre as opiniões dos colaboradores e integradores a respeito das suas percepções sobre influência em um ECOS *open source* baseado no GitHub (Figura 2).

4.1.1 Colaboradores. Neste grupo, 94,6% se identificaram como sexo masculino, 91,1% profissionais, 69,6% possuíam formação em Ciência da Computação e 79% contribuíam em 5 ou mais projetos. De acordo com os colaboradores, as características consideradas relevantes para identificar um influenciador em ECOS estavam mais relacionadas ao projeto e à colaboração real (códigos/comentários). “Longo tempo de interação com o projeto”, “participação com código” e “status no projeto” tiveram os níveis mais altos de concordância entre os participantes (concordo e concordo totalmente). Conforme explicado na Seção 3.1, em um intervalo de 1 a 10 (onde 1 significa nível de influência muito baixo e 10 significa nível muito alto), 52,3% dos participantes se consideraram com nível de influência de 1 até 5; 10,7% indicaram 6; 11,6% indicaram 7; 15,6% indicaram 8; 4% indicaram 9; e 5,8% indicaram 10.

4.1.2 Integradores. Neste grupo, 94,4% se identificaram como sexo masculino, 94,4% profissionais, 66,7% possuíam formação em Ciência da Computação e 77,7% contribuíam para 5 ou mais projetos. As opiniões dos integradores divergiram das opiniões dos colaboradores a respeito das características mais relevantes para se identificar um influenciador em ECOS. “Longo tempo de interação com o projeto”, “participação com código” e “valor do conteúdo” obtiveram os níveis mais altos de concordância entre os participantes (concordo e concordo totalmente). Nem todos os integradores se consideraram com um alto nível de influência no ECOS *npm*: 50% se declararam possuindo nível de influência de 1 até 5; 11,1% indicaram 6; 22,2% indicaram 7; 11,1% indicaram 8; e 5,6% indicaram 10.

³ <https://atlasti.com>

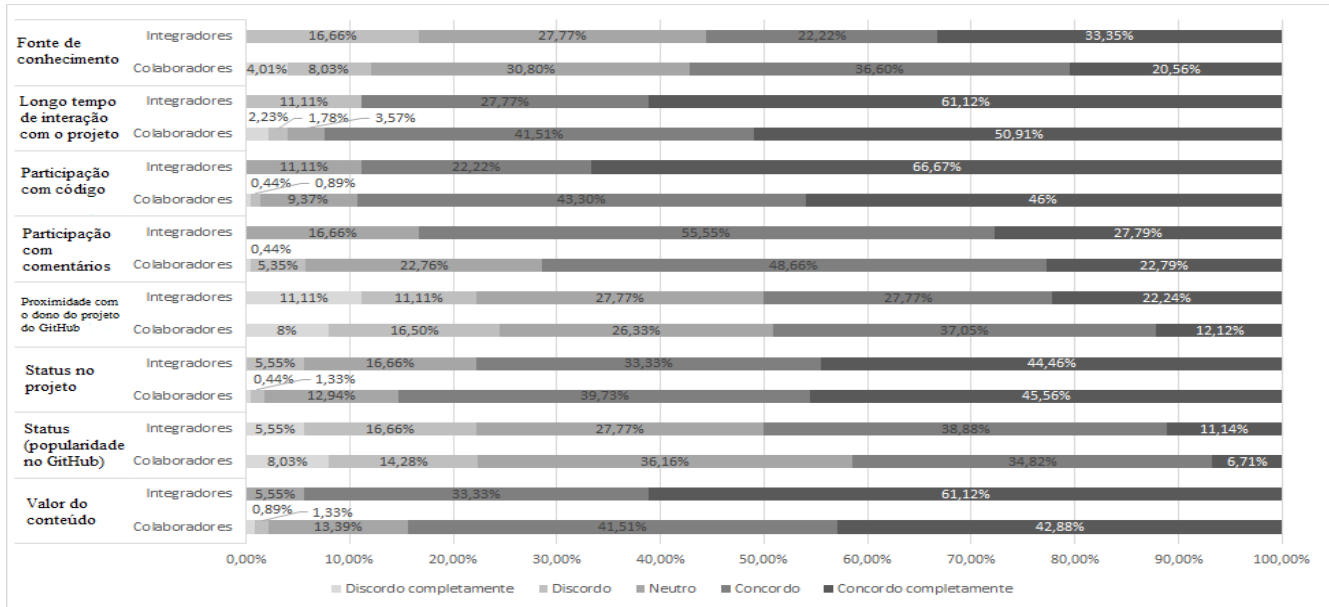


Figura 2: Resultado quantitativo sobre o que caracteriza um influenciador em um ECOS open source baseado no GitHub [4]

A relevância das oito características apresentadas no início da Seção 3 foram avaliadas na escala de Likert de 5 níveis (i.e., discordo completamente, discordo, neutro, concordo e concordo completamente). Estes resultados quantitativos foram apresentados em um trabalho anterior [4].

4.2 Resultados Qualitativos

Na análise qualitativa, foram identificadas duas categorias relativas à natureza de cada característica de um influenciador em ECOS: (i) técnica e (ii) social. A estas categorias, foram relacionados códigos que representam as características de influenciadores e suas definições, conforme explicado no início da Seção 3. Na Tabela 1, apresenta-se este relacionamento, bem como o número de citações realizadas pelos integradores (INT) e colaboradores (COL) para cada um deles.

Tabela 1: Citações por característica de influenciadores

| Categoria | Característica | INT | COL |
|-----------|--|-----|-----|
| Técnica | Fonte de conhecimento | 0 | 2 |
| Técnica | Participação com código | 3 | 16 |
| Técnica | Valor do Conteúdo | 1 | 15 |
| Social | Longo tempo de interação com o projeto no GitHub | 1 | 5 |
| Social | Participação com comentários | 2 | 12 |
| Social | Proximidade com o dono do projeto no GitHub | 0 | 2 |
| Social | Status no projeto | 3 | 32 |
| Social | Status (popularidade no GitHub) | 0 | 1 |

A seguir, são apresentados os códigos gerados e a discussão das características no contexto das duas categorias.

4.2.1 Categoria Técnica. A Figura 3 apresenta os códigos relacionados ao aspecto técnico acerca do senso de influência. Com relação à característica **fonte de conhecimento**, entendida como a distinção de alguém cujo conhecimento em algum aspecto técnico seja suficiente para que outras pessoas adotem técnicas e ideias de apoio de sua referência, foram observadas duas citações. Em sua essência, essas opiniões reforçam o senso de um desenvolvedor ser referência e/ou ser ainda uma fonte de conhecimento para a comunidade de desenvolvedores no ecossistema:

“Usualmente, é para mim que as pessoas pedem a opinião.”
[Colaborador 166]

“Estou envolvido em muitos projetos. Mais proeminentes são minhas contribuições ao projeto Angular. Acho que ajudei muitas pessoas a superar os obstáculos que as pessoas enfrentam quando começam a usar um novo framework. Não tive uma influência notável no desenvolvimento, mas espero ter alguma influência de que mais pessoas tenham tido um início mais agradável e que ajudei o Angular a obter uma reputação de fácil aprendizado e produtiva para trabalhar.”
[Colaborador 150]

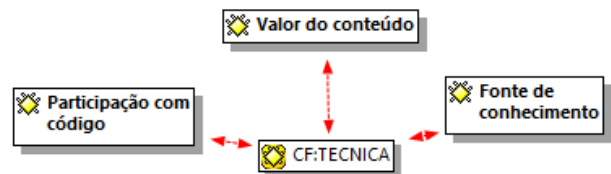


Figura 3: Características relacionadas à categoria técnica

Por sua vez, **participação com código** pode ser definida como a quantidade de *pull requests* que o desenvolvedor criou e foi aceita, mais a quantidade de *pull requests* integrada dentro do repositório. Entre 19 comentários identificados, dois deles destacam a relevância de contribuir com código em ECOS *open source* e a quantidade de projetos que se beneficiam delas:

“Sou o desenvolvedor líder de vários projetos e também estou ativamente envolvido na revisão de código de outros, por isso eu exerço um forte grau de minha influência pessoal nos projetos em que contribuo.”
[Colaborador 7]

“Contribuo para resolver problemas com os quais trabalho ou outros projetos. Eu não tendo a contribuir sem uma necessidade concreta em meus próprios projetos. Sempre que faço, tento elevar o nível técnico.”
[Colaborador 86]

A última característica técnica é **valor do conteúdo**, que pode significar a distinção de alguém como fonte de contribuição de alto valor que ajuda um desenvolvedor a ter as suas contribuições aceitas e a sua influência espalhada. Dos 16 trechos relacionados ao código da análise qualitativa, dois deles destacam o entendimento deste valor como impacto do código:

“No projeto em que eu mais participo, 98% dos meus pull requests são integrados.”
[Colaborador 224]

“Estou respondendo com base em quando participei, já que não participo mais. Eu contribuí com grandes trechos de códigos importantes.”
[Colaborador 121]

4.2.2 *Categoria Social*. Os participantes mencionaram ainda características relacionadas ao aspecto social. A Figura 4 apresenta os códigos deste tipo acerca do senso de influência.

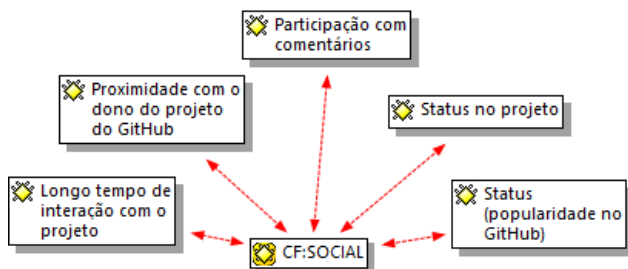


Figura 4: Características relacionadas à categoria social

A característica **status no projeto** pode estar relacionada ao nível de privilégio que um desenvolvedor possui em um projeto do qual faz parte no ECOS. Dos 35 trechos relacionados

a este código, dois deles destacam o papel da autoria dos projetos de um ECOS *open source*:

“Sou autor da maioria dos projetos nos quais contribuo.”
[Colaborador 21]

“Depende do projeto, se o iniciei, sinto que sou responsável por ele.”
[Colaborador 49]

Outra característica é **participação com comentários**, definida como o senso de que comentários podem levar os desenvolvedores do ECOS a contribuírem para os interesses do influenciador, e.g., demanda por uma nova funcionalidade. De 14 passagens associadas à esta característica, duas delas exemplificam o poder de interferir na evolução de projetos e reforçar a influência a partir de ações bem-sucedidas:

“Minhas palavras podem influenciar a direção do projeto.”
[Integrador 14]

“Meus comentários ajudam a tomar decisões e muitos levam a melhorias ou recursos de código.”
[Colaborador 78]

A característica relacionada a **longo tempo de interação com o projeto** ocorre quando um desenvolvedor interage com um projeto por longo tempo, de maneira que as suas ações aumentem a sua chance de influenciar um ECOS, seja com comentários ou contribuição de código. Foram observados seis trechos relacionados, dos quais dois sintetizam, dentre os demais, a questão de haver vínculo de trabalho formal ou mesmo da quantidade de tempo dedicada ao projeto:

“Um projeto é o meu trabalho principal no meu empregador em tempo integral, por isso recebo a maior parte do meu tempo e atenção.”
[Colaborador 222]

“Eu não sou colaborador em tempo integral hoje em dia.”
[Colaborador 54]

Em relação à característica **proximidade com o dono do projeto do GitHub**, entende-se que, quanto mais próximo o desenvolvedor estiver do mantenedor dos projetos do GitHub, mais fácil ele/ela influencia o ECOS, pois suas contribuições e mudanças seriam mais provavelmente aceitas. Foi possível encontrar duas citações correspondentes à esta característica, reforçando a importância dos relacionamentos entre atores em um ecossistema, sobretudo com donos de projetos:

“A maioria dos projetos com os quais estou envolvida são aqueles em que meu parceiro de negócios ou nós criamos.”
[Colaborador 4]

“Relacionamento com os mantenedores do projeto.”
[Colaborador 29]

A última característica relacionada à categoria social é **status (popularidade no GitHub)**. Quanto mais seguidores um desenvolvedor tiver, mais fácil ele/ela influenciará um ECOS, da mesma forma que o seu *status* poderia levar outros desenvolvedores a seguirem a sua visão e tornarem a sua contribuição mais provavelmente aceita. Apesar do desenvolvedor a seguir não ter muitos seguidores, a opinião dele indica que esta é uma característica de um influenciador:

“Eu tenho apenas três seguidores e meus projetos são copiados de outros repositórios que eu gosto.”
[Colaborador 143]

5 Discussão dos Resultados

Os resultados obtidos neste trabalho permitiram identificar opiniões de desenvolvedores acerca de oito características que definem influenciadores em ECOS, que foram agrupadas em duas perspectivas (técnica e social). Após a análise qualitativa, para cada categoria, foi feita uma comparação com o corpo de conhecimento da literatura existente. Os trabalhos utilizados para a discussão desta seção foram extraídos de dois estudos secundários: um mapeamento sistemático da literatura sobre influência em ECOS [7] e um mapeamento sistemático da literatura sobre desenvolvimento de software no GitHub [28].

Na demografia dos participantes e resultados quantitativos [4], observou-se que **participação com código, longo tempo de interação com o projeto e valor do conteúdo** obtiveram alto nível de concordância por parte dos colaboradores e integradores. Porém, para outras características como **status no projeto e participação com comentários**, mesmo com alto nível de concordância dos participantes, houve diferenças entre colaboradores e integradores. Na visão dos colaboradores, *status* tem mais relação com influência do que participação com comentários, ao passo que para integradores é o contrário. Uma possível razão para esta ocorrência pode ser explicada pelo fato dos integradores já terem um *status* no projeto. Portanto, na perspectiva deles, esta característica não poderia ser tão importante para um influenciador como seria para alguém que possui uma participação ativa com comentários relevantes.

Nas respostas qualitativas dos integradores, os participantes citaram **status no projeto, participação com comentários e participação com código**. Por sua vez, para os colaboradores, **status no projeto** tem muito mais relação com influência, seguido de **participação com código e valor do conteúdo**. É interessante notar que, apesar dos integradores terem em si a característica referente ao *status* no projeto, tal como pode ser observado na demografia dos participantes, *status* não foi apontado por eles como sendo uma característica com grande destaque em relação às demais.

Em algumas respostas, também foi possível identificar dificuldades dos respondentes em caracterizar o seu nível de influência. Por exemplo, para o Colaborador 160, o seu nível de influência varia de projeto para projeto, mas não mencionou qualquer tipo de característica que tivesse ou não relação com influência. Com este tipo de comentário, similar ao que outros fizeram, pode-se inferir que, para alguns desenvolvedores, um influenciador pode exercer influência em um projeto dentro de um ECOS. Entretanto, em outros projetos dentro do mesmo ecossistema, ele pode não ter influência alguma. Isso pode ser investigado em pesquisas futuras. Assim, chegamos à conclusão de que a influência (positiva ou negativa), por não ser a função principal de um ator do ECOS, é um mecanismo dependente das interações e do papel do ator no dinamismo do ecossistema. Um direcionamento é que as interações em ECOS precisam ser exploradas em níveis detalhados, i.e., o que é influência para um indivíduo, para uma população e para uma comunidade.

Participação com código foi apontada por integradores e por colaboradores como uma das características com maior relação com um influenciador. Nos resultados quantitativos, a grande maioria dos participantes indicaram concordar em algum nível com o fato de que esta característica é relevante. Além disso, é a segunda característica mais identificada (19 citações) nas opiniões dos participantes deste estudo. Na literatura, Dabbish *et al.* [29] estudaram a percepção de desenvolvedores no GitHub a respeito de ações sobre código. Para alguns deles, o histórico de modificações de código se mostrou influente para as futuras contribuições de um projeto. Outros também mencionaram “recrutar” desenvolvedores que tivessem boas colaborações (em termos de código) para aumentarem fortemente a sua influência no ECOS. Por exemplo, *pull requests* bem documentados [30], incluindo casos de testes [31, 32] e implementados de maneira eficaz [29], possuem maior probabilidade de aceitação.

Na contramão, *pull requests* que não aderem ao estilo do projeto [33, 34] são menos propensos a serem aceitos. A convergência dos estudos indica o forte papel que esta característica tem sobre os usuários do GitHub: um desenvolvedor que contribui com código de qualidade é, normalmente, considerado um influenciador em potencial. A contribuição com outros tipos de artefato, como documentação, também pode ser vista como um elemento básico de influência, uma vez que também são recursos importantes para o ECOS. No entanto, pesquisas futuras precisam endereçar esse assunto mais diretamente. Dessa forma, chega-se à conclusão de que participação com contribuição para um artefato técnico como código é um elemento básico de influência, uma vez que código é um recurso “essencial” para o ECOS e seus projetos. Participar com código permite ainda a avaliação da contribuição do ponto de vista da qualidade de software, o que pode apoiar na análise do trabalho do influenciador e de suas contribuições.

Status no projeto foi a característica mais apontada por colaboradores e integradores como indicativo de influenciador. Foi possível relacionar a ela 35 citações. Esta característica

também foi considerada muito relevante para a grande maioria dos participantes do estudo quantitativo. Na literatura, Tsay *et al.* [31] abordam a hipótese de que desenvolvedores com alto *status* em um projeto específico possuem mais possibilidades de terem as suas contribuições aceitas.

Soares *et al.* [35] também apontam que membros de um time principal de um projeto têm 35% a mais de probabilidade de terem o seu código integrado. Diversas citações relacionadas seguiram o raciocínio de que ser o mantenedor de um projeto significa ter grande influência sobre dele. Este fato corrobora o estudo prévio [4], que reporta que será mais fácil para um desenvolvedor influenciar um ECOS desde que ele tenha a responsabilidade de integrar contribuições no projeto. Chegou-se à conclusão de que, quando relacionada a *status* dentro de um projeto, a influência pode direcionar a mudança de papéis e a atribuição de níveis mais elevados de responsabilidade, considerando um modelo de participação em ECOS.

Por outro lado, na análise qualitativa, *status (popularidade no GitHub)* não foi apontada pelos desenvolvedores como uma característica relevante. Apenas um colaborador relacionou a quantidade de seguidores que possui. Este comportamento também pode ser observado no resultado quantitativo, onde esta característica foi a menos apontada pelos participantes. Tsay *et al.* [31] relatam que um número significativo de seguidores pode aumentar em média 18,1% a probabilidade de aceitação de contribuições. Porém, nas respostas qualitativas e quantitativas dos desenvolvedores, esta característica não possui impacto relevante em comparação às demais. Todavia, isto não significa que essa característica não seja importante para um influenciador, uma vez que mais de 40% dos que responderam à pesquisa de opinião concordaram sobre a sua relevância, sendo necessárias novas investigações.

Conforme também indica Blincoe *et al.* [22], a maior parte dos desenvolvedores do GitHub segue um pequeno número de usuários e que, para 18,2% deles, não haveria benefícios em seguir outro usuário. Assim, chegamos à conclusão de que um ponto para futura análise pode ser o nível de envolvimento dos seguidores com o “influenciador” e com projetos no ECOS: como se cada um dos seguidores pudesse ser avaliado segundo algum nível de influência, ou como os seguidores utilizam a contribuição gerada pelo “influenciador” para expandir o ECOS.

Por sua vez, Tsay *et al.* [31] apontam que desenvolvedores com forte conexão social com o projeto têm maior probabilidade de terem as suas contribuições aprovadas. Esta probabilidade pode aumentar em média 35,6% de acordo com o estudo desses autores, nos casos de prévia interação. Para alguns desenvolvedores que responderam à pesquisa de opinião, o *longo tempo de interação com o projeto* interfere nesse sentido, pois poderia indicar um conhecimento mais amplo sobre o projeto, que novos contribuidores ainda não possuem. É possível que para um desenvolvedor esta característica seja, na verdade, um componente de uma característica em um nível “hierárquico” acima. Por exemplo, se um desenvolvedor menciona que participações com código e

com comentários são características de um influenciador, isto significaria dizer que o desenvolvedor possuiria uma interação intensa com o projeto, de modo que mencionar isto seria redundante em seu ponto de vista.

A conexão social pode ser observada ainda nos casos em que existe uma *proximidade com o dono do projeto do GitHub*. De acordo com Tsay *et al.* [31], esta proximidade poderia aumentar em média 187% a aceitação de uma contribuição, quando o colaborador segue o mantenedor. Na opinião dos participantes da pesquisa de opinião, pouco menos da metade concordou em algum nível que esta característica possa estar relacionada com um influenciador em ECOS. Na análise qualitativa, identificou-se que dois colaboradores teceram comentários referentes ao seu relacionamento com o dono do projeto. Tal relacionamento é visto por esses desenvolvedores como ser influenciado a participar de um projeto por ter um relacionamento direto com o mantenedor.

Outra característica descrita neste trabalho e que também foi relatada na literatura é *fonte de conhecimento*. Blincoe *et al.* [22] constataram que 11,2% dos respondentes seguiam outros usuários do GitHub a fim de aprender com eles. Na análise qualitativa realizada neste estudo, um desenvolvedor citou que auxilia usuários novatos de um *framework* a superarem obstáculos comuns àqueles que estão começando, na esperança de ter influência sobre a percepção deles a respeito da facilidade de aprendizado da plataforma. Desta forma, conclui-se que é possível notar que um usuário novo pode ter um certo grau de dependência ao entrar em um projeto do qual ainda não está familiarizado, sendo que o influenciador pode ter um papel de guia no começo da colaboração.

Blincoe *et al.* [22] também discorrem sobre a *participação com comentários* em projetos do GitHub. Esta foi a única medida de atividade que obteve impacto sobre influência no estudo, mesmo que o efeito tenha sido considerado pequeno. Para alguns participantes que responderam à questão aberta analisada neste trabalho, a opinião é que os seus comentários têm o poder de influenciar a direção do projeto e que também levam à tomada de decisão e melhorias do código. Pode-se perceber que, apesar de Blincoe *et al.* [22] não terem notado uma grande influência dos comentários em seu estudo, os desenvolvedores que participaram da pesquisa têm percepção de que comentários influenciam decisões em torno do projeto.

Valor do conteúdo teve nível de concordância muito alto por parte dos integradores na análise quantitativa. No entanto, foi possível relacionar a resposta de apenas um integrador a esta característica. Uma possível explicação, a partir da análise qualitativa, pode estar na naturalidade com que os integradores lidam com esta característica. Em outras palavras, para eles, este ponto seria tão vital para definir um influenciador que não faria sentido ser comentado, conforme mencionado na Seção 4.2.1 no que se refere ao impacto da contribuição.

Ceccagnoli *et al.* [36] abordam a cocriação de valor em ecossistemas, concluindo que os fornecedores de software independentes (no caso, desenvolvedores no GitHub) alcançam

benefícios significativos para o ecossistema como, por exemplo, aumento de vendas. Por isto, quando abordados a respeito desta característica, os desenvolvedores podem entendê-la como algo vital para a evolução de um ECOS, já que contribui diretamente no produto final. Tendo em vista que a literatura corrobora com a alta concordância em algum nível por parte dos participantes no resultado quantitativo, a diferença encontrada entre as respostas qualitativas e quantitativas dos integradores pode ser mais explorada em um trabalho futuro.

6 Implicações para a Prática

Os resultados deste trabalho podem ser relevantes para que, no futuro, organizações que atuam em ECOS *open source* possam ter conhecimento sobre os influenciadores de seu ecossistema. Conforme citado por um colaborador, um exemplo é o poder de influência que um desenvolvedor pode ter sobre a percepção de facilidade de uso do seu software, assim como a propensão para se tornar um colaborador ou um integrador em um projeto do ECOS. Chega-se à conclusão de que um projeto reconhecido por ser de fácil aprendizagem por parte de novatos tem mais chance de atrair futuros colaboradores para o desenvolvimento.

Além disso, observa-se dois tipos de influência exercida: (1) *intrínseca*: um influenciador exerce função de representação da gestão do ECOS (e.g., Microsoft MVP, Google Experts), com papel crítico na expansão e liderança da comunidade técnica; e (2) *extrínseca*: um influenciador pode ajudar a mudar fluxos do uso e contribuição de recursos dentro do ECOS, movendo a comunidade para um conjunto de objetivos esperados que mantenham o ecossistema sustentável.

Os resultados podem ser usados para construir uma função cujos pesos das características poderiam vir dos resultados obtidos pelas respostas dos participantes, e.g., *status* no GitHub teria nenhuma ou pouca influência enquanto que participação com código/comentários e valor do conteúdo poderiam ter pesos maiores na definição deste "fator" de influência de um indivíduo. Gestores do ECOS podem apoiar a disseminação de boas práticas, treinamento, acompanhamento de dúvidas, migração de produtos, evolução da plataforma etc., dando suporte à orquestração do ECOS por meio de influenciadores.

Adicionalmente, é possível incentivar comunidades de desenvolvedores a terem uma interação de forma mais prolongada com o projeto, considerando a existência de um ECOS. Dessa forma, dado que este é um ponto muito observado pelos participantes da pesquisa de opinião, compreende-se que deve acentuar o poder de influência de sua comunidade para que um desenvolvedor possa atrair mais colaboradores.

7 Limitações do Estudo

É importante destacar que esta pesquisa levou em consideração a opinião de diversos desenvolvedores que têm papel de integradores e colaboradores que atuam em projetos do ECOS *npm* suportados pelo controle de versão Git e hospedados pela

plataforma GitHub. Algumas limitações inerentes a pesquisas desta natureza envolvem efeitos da participação de terceiros.

Com relação à quantidade de participantes, notou-se um número menor de integradores. No entanto, foram obtidas respostas de 95 desenvolvedores para a análise qualitativa. Além disso, a participação de colaboradores em maior escala contribui para entender o senso de influência pela sua posição em um ECOS *open source*. Destaca-se a dificuldade de obter alto número participantes em pesquisas de opinião e que a força de um estudo qualitativo não reside necessariamente neste número, mas na qualidade dos dados e das análises [37, 38, 39].

Vale esclarecer que GT, assim como qualquer outro método, possui limitações, tais como o viés do pesquisador e a falta de diferentes visões a respeito dos dados analisados. Isso se deve ao fato de que um conjunto de dados está sujeito a inúmeras possibilidades de interpretações durante a análise. Pinto e Santos [40] afirmam que o viés pessoal dos envolvidos na elaboração do trabalho pode limitar a pesquisa. Para mitigar este cenário, todo o processo realizado ao aplicar GT (codificações aberta e axial) foi avaliado e validado por dois pesquisadores com experiência em ECOS e por dois pesquisadores com experiência em métodos qualitativos.

Além disso, algumas dificuldades na utilização da ferramenta Atlas.TI foram encontradas ao aplicar GT. Para diminuir esta limitação, foram consultadas algumas pesquisas (como de Costa e Itelvino [41]), que serviram como suporte para elaboração deste trabalho, por possuir um tutorial detalhado de como usar os conceitos de GT na ferramenta.

8 Considerações Finais

Este artigo apresentou um estudo exploratório sobre o senso de influência em ECOS *open source*, mais especificamente a partir das opiniões de 95 desenvolvedores que contribuem para projetos no ECOS *npm*. Dados qualitativos de uma pesquisa de opinião conduzida foram analisados a partir de procedimentos de GT. Nesta análise, foi possível identificar que as características mais apontadas pelos desenvolvedores em seus comentários foram *status* no projeto, participação com código, valor do conteúdo e participação com comentários. Há aspectos divergentes, tais como a percepção dos integradores sobre o 'valor do conteúdo' e a percepção dos desenvolvedores sobre o 'longo tempo de interação com o projeto'.

Como contribuições deste trabalho, projetos de software e ecossistemas têm insumos para mecanismos mais específicos para identificação dos influenciadores que guiam evolução do ECOS, ao passo que os desenvolvedores podem compreender melhor as características de quem influencia o projeto para o qual estão desenvolvendo e o ECOS onde atuam. A partir dos resultados, alguns trabalhos futuros foram identificados para avançar o estado da arte e da prática sobre influência em ECOS *open source*, tais como: (i) investigar a diferença entre as respostas qualitativas e quantitativas dos integradores acerca de 'valor do conteúdo'; (ii) verificar se os resultados obtidos no

ecossistema *npm* são equivalentes aos de outros ECOS *open source* como RubyGems; (iii) expandir o estudo de influência considerando outras plataformas além do GitHub; e (iv) investigar a diferença entre as respostas quantitativas e qualitativas acerca de 'longo tempo de interação com o projeto'.

AGRADECIMENTOS

O presente trabalho foi realizado com apoio da UNIRIO (bolsa) e da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

REFERÊNCIAS

- [1] T. Mens, M. Cataldo, D. Damian, 2019. The Social Developer: The Future of Software Development. *IEEE Software* 36(1):11-14. DOI: <https://doi.org/10.1109/MS.2018.2874316>.
- [2] K. Manikas. 2016. Revisiting software ecosystems Research: A longitudinal literature study. *Journal of Systems and Software* 117(C):84-103. DOI: <https://doi.org/10.1016/j.jss.2016.02.003>.
- [3] G. Valença, C. Alves. 2016. Understanding How Power Influences Business and Requirements Decisions in Software Ecosystems. In: Proceedings of the 31st Annual ACM Symposium on Applied Computing (SAC'16), Pisa, Italy, pp. 1258-1263. DOI: <https://doi.org/10.1145/2851613.2851756>.
- [4] V. Farias, I. Wiese, R. P. Santos. 2019. What Characterizes an Influencer in Software Ecosystems?. *IEEE Software*. 36(1):42-47. DOI: <https://doi.org/10.1109/MS.2018.2874325>.
- [5] J. C. Carver, H. Muccini, A. Yamashita. 2017. Distributed Teams, Developer Participation, and More. *IEEE Software* 34(3):114-116. DOI: <https://doi.org/10.1109/MS.2017.85>.
- [6] K. Crowston, H. Annabi, J. Howison, C. Masango. 2004. Effective work practices for software engineering: free/libre open source software development. In: Proceedings of the 2004 ACM Workshop on Interdisciplinary Software Engineering Research (WISER'04). ACM, New York, USA, pp. 18-26. DOI: <https://doi.org/10.1145/1029997.1030003>.
- [7] V. G. Farias. 2017. Investigating the Role of an Influencer in Software Ecosystems. B.Sc. Thesis, UNIRIO.
- [8] E. Guzman, D. Azócar, Y. Li. 2014. Sentiment analysis of commit comments in GitHub: an empirical study. In: Proceedings of the 11th Working Conference on Mining Software Repositories (MSR 2014). ACM, New York, USA, pp. 352-355. DOI: <https://doi.org/10.1145/2597073.2597118>.
- [9] F. Jurado, P. Rodriguez. 2015. Sentiment Analysis in monitoring software development processes. *Journal of Systems and Software* 104(C):82-89. DOI: <https://doi.org/10.1016/j.jss.2015.02.055>.
- [10] V. Sinha, A. Lazar, B. Sharif. 2016. Analyzing developer sentiment in commit logs. In: Proceedings of the 13th International Conference on Mining Software Repositories (MSR'16). ACM, New York, USA, pp. 520-523. DOI: <https://doi.org/10.1145/2901739.2903501>.
- [11] A. Fontão, S. Cleger-Tamayo, I. Wiese, R. P. Santos, A. Dias-Neto. 2020. On Value Creation in Developer Relations (DevRel): a practitioners' perspective. In: Proceedings of the 15th ACM/IEEE International Conference on Global Software Engineering (ICGSE '20).
- [12] M. Lungu, M. Lanza, T. Girba, R. Robbes. 2010. The Small Project Observatory: Visualizing software ecosystems. *Sci. of Comp. Programming* 75(4):264-275. DOI: <https://doi.org/10.1016/j.scico.2009.09.004>.
- [13] P. R. J. Campbell, F. Ahmed. 2010. A three-dimensional view of software ecosystems. In Proceedings of the Fourth European Conference on Software Architecture: Companion Volume (ECSA'10). ACM, New York, USA, pp. 81-84. DOI: <https://doi.org/10.1145/1842752.1842774>.
- [14] R. P. Santos, C. M. L. Werner. 2011. A Proposal for Software Ecosystems Engineering. In: Proceedings of the 3rd International Workshop on Software Ecosystems, Brussels, Belgium. CEUR-WS, vol. 746, pp. 40-51.
- [15] R. Hoving, G. Slot, S. Jansen. 2013. Python: Characteristics identification of a free open source software ecosystem. In: IEEE International Conference on Digital Ecosystems and Technologies, Menlo Park, USA, pp. 13-18. DOI: <https://doi.org/10.1109/DEST.2013.6611322>.
- [16] T. M. P. Lima, R. P. Santos, J. Oliveira, C. Werner. 2016. The importance of socio-technical resources for software ecosystems management. *Journal of Innovation in Digital Ecosystems* 3(2):98-113. DOI: <https://doi.org/10.1016/j.jides.2016.10.006>.
- [17] K.E. Kirkhart. 2000. Reconceptualizing evaluation use: An integrated theory of influence. *New Directions for Evaluation* 2000(88):5-23. DOI: <https://doi.org/10.1002/ev.1188>.
- [18] Dictionary.com. 2020. Disponível em: <https://www.dictionary.com/>.
- [19] J. Hagel, J. S. Brown, L. Davison. 2008. Shaping strategy in a world of constant disruption. *Harvard Business Review* 86(10):80-89.
- [20] R. T. da Silva, L. G. F. Aguiar, R. P. Santos, E. C. Genvigir. 2016. Levantamento de Papéis e Atores em um Ecossistema de Software no Domínio Público. In: Anais do I Workshop sobre Aspectos Sociais, Humanos e Econômicos de Software (WASHES'16), Maceió, Brasil, pp. 76-80.
- [21] S. Jansen, S. Brinkkemper, A. Finkelstein. 2009. Business network management as a survival strategy: a tale of two software ecosystems. In: Proceedings of the First International Workshop on Software Ecosystems (IWSECO-2009), New York, USA, pp. 34-48.
- [22] K. Blincoe, J. Sheoran, S. Goggins, E. Petakovic, D. Damian. 2016. Understanding the popular users: Following, affiliation influence and leadership on GitHub. *Information and Software Technology* 70(2016):30-39. DOI: <https://doi.org/10.1016/j.infsof.2015.10.002>.
- [23] K. Crowston, H. Annabi, J. Howison, C. Masango. 2004. Effective work practices for software engineering: free/libre open source software development. In: Proceedings of the 2004 ACM Workshop on Interdisciplinary Software Engineering Research (WISER'04). ACM, New York, USA, pp. 18-26. DOI: <https://doi.org/10.1145/1029997.1030003>.
- [24] C. Wohlin, P. Runeson, M. Hst, M. C. Ohlsson, B. Regnell, A. Wessln. 2012. Experimentation in Software Engineering. Springer Publishing Company.
- [25] T. Mens, M. Goeminne. 2011. Analysing the evolution of social aspects of open source software ecosystems. In: Proceedings of the International Workshop on Software Ecosystems, Brussels, Belgium. CEUR, vol. 746, pp. 1-14.
- [26] M. Angrosino. 2009. Etnografia e observação participante. Coleção Pesquisa Qualitativa coordenada por Uwe Flick (eds.). Porto Alegre: Artmed.
- [27] A. Strauss, J. Corbin. 2007. Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory. SAGE Publications, 3rd ed. Thousand Oaks, CA: Sage.
- [28] V. Cosentino, J. L. Cánovas Izquierdo, J. Cabot. 2017. A Systematic Mapping Study of Software Development with GitHub. *IEEE Access* 5(2017):7173-7192. DOI: <https://doi.org/10.1109/ACCESS.2017.2682323>.
- [29] L. Dabbish, C. Stuart, J. Tsay, J. Herbsleb. 2012. Social coding in GitHub: transparency and collaboration in an open software repository. In: Proc. of the ACM 2012 Conference on Computer Supported Cooperative Work (CSCW'12). ACM, New York, USA, pp. 1277-1286.
- [30] G. Gousios, M.-A. Storey, A. Bacchelli. 2016. Work practices and challenges in pull-based development: the contributor's perspective. In: Proceedings of the 38th International Conference on Software Engineering (ICSE'16). ACM, New York, USA, 285-296. DOI: <https://doi.org/10.1145/2884781.2884826>.
- [31] J. Tsay, L. Dabbish, J. Herbsleb. 2014. Influence of social and technical factors for evaluating contribution in GitHub. In: Proceedings of the 36th International Conference on Software Engineering (ICSE 2014). ACM, New York, USA, pp. 356-366. DOI: <https://doi.org/10.1145/2568225.2568315>.
- [32] J. Tsay, L. Dabbish, J. Herbsleb. 2014. Let's talk about it: evaluating contributions through discussion in GitHub. In: Proc. of the 22nd ACM SIGSOFT Intern. Symposium on Foundations of Software Engineering. ACM, New York, USA, 144-154. DOI: <https://doi.org/10.1145/2635868.2635882>.
- [33] G. Gousios, M. Pinzger, A. Deursen. 2014. An exploratory study of the pull-based software development model. In: Proceedings of the 36th International Conference on Software Engineering (ICSE 2014). ACM, New York, NY, USA, 345-355. DOI: <https://doi.org/10.1145/2568225.2568260>.
- [34] V. J. Hellendoorn, P. T. Devanbu, A. Bacchelli. 2015. Will they like this? evaluating code contributions with language models. In: Proc. of the 12th Working Conf. on Mining Soft. Repos. (MSR'15), Florence, Italy, pp. 157-167.
- [35] D. M. Soares, M. L. de Lima Júnior, L. Murta, A. Plastino. 2015. Acceptance factors of pull requests in open-source projects. In: Proceedings of the 30th Annual ACM Symposium on Applied Computing (SAC'15). ACM, New York, USA, pp. 1541-1546. DOI: <https://doi.org/10.1145/2695664.2695856>.
- [36] M. Ceccagnoli, C. Forman, P. Huang, D. J. Wu. 2012. Cocreation of Value in a Platform Ecosystem: The Case of Enterprise Software. *MIS Quarterly* 36.
- [37] S. Brinkmann, S. Kvale. 2015. *InterViews: Learning the craft of qualitative research interviewing* (3rd ed.). SAGE Publications, Inc.
- [38] M. O'Reilly, N. Parker. 2013. Unsatisfactory saturation: A critical exploration of the notion of saturated sample sizes in qualitative research. *Qualitative Research* 13(2):190-197. <https://doi.org/10.1177/1468794112446106>.
- [39] D. E. Polkinghorne. 2005. Language and meaning: Data collection in qualitative research. *Journal of Counseling Psychology* 52(2):137-145. DOI: <https://doi.org/10.1037/0022-0167.52.2.137>.
- [40] M. R. Pinto, L. L. S. Santos. 2012. A Grounded Theory como abordagem metodológica: relatos de uma experiência de campo. *Organ. Soc.* [online] 19(62):417-436. <https://doi.org/10.1590/S1984-92302012000300003>.
- [41] P. R. Costa, L. S. Itelvino. 2018. Grounded theory com utilização do software Atlas.TI: um exemplo empírico de estudo sobre estratégia de ascensão do empreendedorismo inovador em negócios sociais. *Revista Ibero-Americana de Estratégia - RIAE* 17(3):17-40.