

UNIVERSIDADE FEDERAL DE JUIZ DE FORA
INSTITUTO DE CIÊNCIAS EXATA
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

Análise de Colaboração em Plataformas de Desenvolvimento de Software

André Dias Nunes

JUIZ DE FORA
AGOSTO, 2022

Análise de Colaboração em Plataformas de Desenvolvimento de Software

ANDRÉ DIAS NUNES

Universidade Federal de Juiz de Fora

Instituto de Ciências Exatas

Departamento de Ciência da Computação

Bacharelado em Ciência da Computação

Orientador: Victor Stroele De Andrade Menezes

Coorientador: José Maria Nazar David

JUIZ DE FORA

AGOSTO, 2022

ANÁLISE DE COLABORAÇÃO EM PLATAFORMAS DE DESENVOLVIMENTO DE SOFTWARE

André Dias Nunes

MONOGRAFIA SUBMETIDA AO CORPO DOCENTE DO INSTITUTO DE CIÊNCIAS
EXATA DA UNIVERSIDADE FEDERAL DE JUIZ DE FORA, COMO PARTE INTE-
GRANTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE
BACHAREL EM CIÊNCIA DA COMPUTAÇÃO.

Aprovada por:

Victor Stroele De Andrade Menezes
<<Título do Orientador>>

José Maria Nazar David
<<Título do Co-orientador>>

<<Nome do Examinador 1>>
<<Título do Examinador 1>>

<<Nome do Examinador 2>>
<<Título do Examinador 2>>

JUIZ DE FORA
<<DIA>> DE AGOSTO, 2022

Aos meus amigos e irmãos.

Aos pais, pelo apoio e sustento.

Resumo

A interação entre os desenvolvedores sempre foi essencial para o aprendizado, manutenção e evolução da engenharia de software. Seja compartilhando conhecimento, incorporando novas ferramentas, buscando suporte e até o desenvolvimento colaborativo, a adesão às plataformas que impulsionam essa interação está aumentando a cada dia devido ao constante surgimento de novas tecnologias, demandas do mercado e até mesmo contexto de isolamento global, pós-pandemia de COVID19. Comunidades, centradas em torno de uma única plataforma, constroem o que é chamado de Ecossistema de Software (ECOS).

Para adaptar a essa tendência e manter competitividade no mercado, muitas empresas também adotaram essas plataformas e/ou até construíram suas próprias, formando ECOS internos. Este trabalho tem como objetivo estudar *frameworks* de análise de dados relacionados a Sistema de Reputação, propostos para ECOS de plataformas abertas, para então propor um novo *framework* e aplicá-lo em ambientes reais e privados de empresas. Por meio dos resultados obtidos, validam-se e identificam-se as semelhanças e diferenças entre estes ECOS, para a obtenção de colaboradores chave e suas especialidades com intuito de auxiliar na gestão interna de pessoas.

Palavras-chave: Ecossistema de software, redes complexas, sistema de reputação, gestão de equipes, coleta e análise de dados.

Abstract

The interaction between developers has always been essential for the learning, maintenance and evolution of software engineering. Whether sharing knowledge, incorporating new tools, seeking support and collaborative development, adherence to the platforms that drive this interaction is increasing every day due to the constant emergence of new technologies, market demands and even the context of post-pandemic global isolation of COVID19. Communities centered around a specific platform build what is called a Software Ecosystem (SECO).

To adapt to this trend and remain competitive in the market, many companies have also adopted these platforms and even built their own, forming internal SECO. This work aims to study data analysis *frameworks* related to Reputation System, proposed for SECO of open platforms, to then propose a *framework* and apply it in real and private environments of companies. Through the results obtained, the similarities and differences between these SECO are validated and identified, in order to obtain key collaborators and their specialties in order to assist in the internal management of people.

Keywords: Software ecosystem, complex networks, reputation system, team management, data collection and analysis.

Agradecimentos

A todos os meus parentes, pelo encorajamento e apoio. Ao professor Victor Stroele pela orientação, amizade e principalmente, pela paciência, sem a qual este trabalho não se realizaria. Aos professores do Departamento de Ciência da Computação pelos seus ensinamentos e aos funcionários do curso, que durante esses anos, contribuíram de algum modo para o nosso enriquecimento pessoal e profissional.

Conteúdo

Lista de Figuras	7
Lista de Tabelas	8
Lista de Abreviações	9
1 Introdução	10
1.1 Apresentação do Tema	10
1.2 Contextualização	10
1.3 Problema	11
1.4 Justificativa	12
1.5 Objetivos	12
1.5.1 Objetivos Gerais	12
1.5.2 Objetivos Específicos	13
1.6 Metodologia	13
1.6.1 Caracterização da pesquisa	13
1.6.2 Framework	13
1.6.3 Coleta de Dados	14
1.6.4 Análise dos Dados e Resultados	14
2 Fundamentação Teórica	15
2.1 Introdução	15
2.2 Ecossistema de Software	15
2.2.1 Introdução	15
2.2.2 Definição	16
2.2.3 Dimensões	18
2.3 Sistemas De Reputação	20
2.3.1 Introdução	20
2.3.2 Desenvolvimento Distribuído de Software	20
2.3.3 Reputação	21
2.3.4 Colaborador Chave	22
2.4 Considerações Finais	22
3 Trabalhos Relacionados	23
3.1 Introdução	23
3.2 Managing and Monitoring Software Ecosystem to Support Demand and Solution Analysis	23
3.3 Um Modelo Dinâmico de Reputação para Apoiar a Manutenção Colaborativa de Software	25
3.4 Topological Analysis in Scientific Social Networks to Identify Influential Researchers	26
3.5 Ecosystems in GitHub and a Method for Ecosystem Identification Using Reference Coupling	27
3.6 Complex Network Analysis in a Software Ecosystem: Studying the Eclipse Community	28

3.7	sSECO-Process: Avaliando a Dimensão Social em Ecossistemas de Software	29
3.8	Análise comparativa	29
3.9	Considerações Finais	30
4	Conclusões	31
4.1	Introdução	31
4.2	Conclusão e Cronograma de Execução	31
	Bibliografia	33

Lista de Figuras

2.1	Modelo de domínio das relações entre os termos atualmente existentes em ECOS (BERK; JANSEN; LUINENBURG, 2010)	17
2.2	Visão Tridimensional do ECOS (CAMPBELL; AHMED, 2010)	18

Lista de Tabelas

3.1	Tabela comparativa dos trabalhos relacionados.	30
4.1	Cronograma de execução.	32

Lista de Abreviações

API	Application Programming Interface
BPM	Business Process Management
CDB	Component-Based Development
CRM	Customer Relationship Management
DCC	Departamento de Ciência da Computação
DDS	Desenvolvimento Distribuído de Software
ECM	Enterprise Content Management
ECOS	Ecosystema de Software
GUI	Graphical User Interface
RPP	Research Platforms and Portals
SE	Software Engineering
SPL	Software Product Line
TCC	Trabalho de Conclusão de Curso
TI	Tecnologia da informação
UFJF	Universidade Federal de Juiz de Fora

1 Introdução

1.1 Apresentação do Tema

Seja na área profissional ou acadêmica, os desenvolvedores interagem principalmente em Ecossistemas de Software (ECOS) para adquirir novos conhecimentos, buscar suporte, fornecer seus conhecimentos para a resolução de problemas, manutenção e evolução de seus projetos (GUERCIO et al., 2018). Os benefícios desses ecossistemas são vastos e vão além das características técnicas. Os dados obtidos a partir da análise desses ambientes são valiosos, mas ainda não totalmente explorados.

A definição de ECOS adotada neste trabalho é a de que são comunidades, dentro e/ou fora de uma organização, que interagem sobre uma plataforma tecnológica comum, visando contribuição e troca de informações em busca de soluções e serviços (SANTOS; VIANA, 2016). Os participantes dessas comunidades podem consistir em desenvolvedores, *stakeholders*, *software* e muito mais.

Um exemplo comum de ECOS é a plataforma *GitHub*, onde desenvolvedores publicam e versionam seus projetos, possibilitando um trabalho colaborativo e público. Outro exemplo é a plataforma *Stack Overflow*, onde a comunidade de desenvolvedores, usuários e as organizações podem interagir e promover suporte para diversas linguagens, ferramentas e softwares.

Projetos, em sua grande maioria, não são desenvolvidos isoladamente, eles utilizam de tecnologias pré-existentes que, quando aplicadas, podem evoluir e, posteriormente, ser aplicadas em novos projetos (BLINCOE; HARRISON; DAMIAN, 2015). Esta é uma rede que fomenta o desenvolvimento do ecossistema e dos desenvolvedores que ele contém.

1.2 Contextualização

O Ecossistema de Software pode ser mapeado sistematicamente em uma perspectiva tri-dimensional, denominadas em *Architectural*, *Business* e *Social* (BARBOSA et al., 2013).

A dimensão Técnica (*Architectural*) concentra-se na plataforma, diretamente relacionada à evolução e arquitetura de software. A dimensão de negócio (*Business*) concentra-se no gerenciamento de conhecimento, ou seja, artefatos, recursos, informações, inovação e planejamento estratégico (compreendendo como, quando, onde e quem realizar determinadas tarefas). A dimensão social se concentra em como os stackholders interagem, entendem e modificam o conhecimento em um ecossistema de software (CAMPBELL; AHMED, 2010).

Devido à crescente adoção de plataformas globais de desenvolvimento de *software*, como os exemplos já citados, *GitHub* e *Stack Overflow*, estudos têm se concentrado nesses ecossistemas para extrair informações de seus bancos de dados, como detecção de especialistas, relacionamentos entre projetos, melhores soluções de busca, evolução da arquitetura, tecnologias aplicadas, sistema de reputação e estatísticas das linguagens de programação (BADASHIAN et al., 2014; BLINCOE; HARRISON; DAMIAN, 2015).

A maior dificuldade que estes estudos apresentam, está relacionada à complexidade de coleta, tratamento e processamento de dados, seja pela enorme amostra de dados que plataformas abertas oferecem, ou pela dificuldade de obter amostras de ambientes profissionais e privados das empresas. Os ativos de conhecimentos que podem ser derivados destes dados e sua validação, têm se mostrado uma área de pesquisa com grande potencial de valor.

1.3 Problema

As empresas de TI e desenvolvimento de software necessitam de formas para gerenciar suas equipes, por isso sempre buscam aplicar métodos para mensurar o desenvolvimento de seus projetos e colaboradores. Atualmente, metodologias ágeis têm sido adotadas visando aprimorar e determinar a cadência do desenvolvimento. No entanto, ainda existe uma carência para aprimorar o planejamento estratégico do gerenciamento de pessoal.

À medida que as diferentes estratégias de aquisição e processamento de dados do ecossistema de software continuam a ser abordadas em plataformas abertas (BLINCOE; HARRISON; DAMIAN, 2015), surgem questões quanto a validação e efetividade dos processos e *frameworks*, propostos por trabalhos relacionados, quando aplicados em

plataformas e ambientes internos das empresas, visando a perspectiva da dimensão de negócio do ecossistema.

Encontrar um ecossistema adequado para implementar testes é uma tarefa complexa, pois as informações referenciadas pelo banco de dados podem levar a perdas financeiras se não forem cuidadas e protegidas adequadamente, tornando difícil que as empresas forneçam uma amostra adequada de dados. Além disso, cada ambiente possui fatores de adaptação únicos, aumentando a complexidade do processamento e análise dos dados.

1.4 Justificativa

Desenvolver um método para identificar os colaboradores chaves e sua reputação dentro de um ECOS é fundamental para o gerenciamento interno das empresas de desenvolvimento de *software* (BOEHM, 2006). Uma substituição mal planejada pode prejudicar o desenvolvimento de um projeto, aumentando a curva de aprendizado da equipe e prolongando o prazo de entrega do projeto.

Com os dados obtidos da análise deste ambiente, busca-se estimar o desenvolvimento, eficiência e perícia de cada colaborador e/ou áreas carentes de aprendizado, auxiliar no processo de remuneração, planejamento estratégico para formação e/ou manutenção de equipes, estimar o ritmo de uma equipe para determinar a necessidade de reforçá-la e muito mais.

A gama de benefícios que essa abordagem pode trazer para as empresas é ampla e ainda inexplorada. Portanto, seu desenvolvimento pode fornecer uma base para a solução de problemas desconhecidos. Para isso, é importante realizar validação e pesquisa no ecossistema interno das empresas.

1.5 Objetivos

1.5.1 Objetivos Gerais

Estudos atuais sobre a análise de dados de Ecossistemas de Software em plataformas abertas, tem produzido avanços promissores. O objetivo deste trabalho é validar as me-

metodologias e *frameworks*, propostos em estudos relacionados, em um ambiente interno de desenvolvimento profissional. A partir da análise das informações obtidas neste ambiente, promover a discussão da relação dos resultados obtidos com os resultados dos estudos relacionados. Dessa forma, semelhanças, diferenças, falhas e melhorias podem ser identificadas e validadas.

1.5.2 Objetivos Específicos

A amostra de dados para teste será obtido de um ambiente real de desenvolvimento profissional. Dados de equipes de desenvolvedores de projetos na plataforma SYDLE ONE, serão coletados. Esta é uma plataforma completa de desenvolvimento de software, onde as equipes atuam diretamente em automação de processos (BPM), gerenciamento de documentos (ECM), gestão e relacionamento (CRM), *e-commerce*, ciência de dados e *billing*.

Este trabalho visa propor um *framework* analítico para as amostras coletadas. Os resultados devem permitir estimar as necessidades de cada equipe, os principais funcionários de cada área e o impacto de sua ausência. Informações que podem ser utilizadas para a gestão interna da empresa.

1.6 Metodologia

1.6.1 Caracterização da pesquisa

Este trabalho adota uma vertente metodológica de natureza quantitativa, visando coletar dados e estatísticas reais como base para validar modelos relacionados à gestão de colaboradores em Ecossistemas de Software. Para obter uma resposta objetiva, é proposto um framework que pode ser replicado e testado em diferentes projetos centralizados em uma plataforma.

1.6.2 Framework

Na fase inicial, serão estudados os fundamentos de ECOS e sua literatura. Posteriormente, alguns dos modelos identificados em trabalhos relacionados serão utilizados para extrair

questões relevantes à gestão de pessoas. Uma especificação de um *framework*, baseado nesses trabalhos, será então proposta.

Os principais módulos deste *framework* incluem: coleta de dados, limpeza e transformação de dados, uso de modelos semânticos e análise de resultados.

1.6.3 Coleta de Dados

A principal etapa compreende o aspecto da obtenção dos dados. O ambiente de estudo pertence a uma empresa real de desenvolvimento de *software*, que será chamada de empresa parceira para manter seu anonimato. A amostra é extensa e inclui *logs* de desenvolvimento de diferentes projetos desenvolvidos na mesma plataforma. *Logs*, semelhante aos *commits* da plataforma *Github*, contém informações sobre o conteúdo alterado, seu autor, data e área de atuação (*billing*, CRM, BPM, etc.).

Para a coleta dos dados, uma busca extensiva será realizada utilizando o *Elastic-Search* no banco de dados da plataforma. A amostra será então disponibilizada via um *endpoint* de uma API a ser desenvolvida. A amostra será tratada de modo a transformar ou retirar informações sigilosas e pessoais da empresa e seus colaboradores.

1.6.4 Análise dos Dados e Resultados

Após a coleta de dados, será realizada a análise dos dados, aplicando métodos de identificação para Sistema de Reputação e métricas de Redes Complexas, propostas por trabalhos precursores. Nesta fase, o objetivo é identificar colaboradores chave de cada projeto, suas especialidades e impacto da ausência.

Com os resultados, busca-se uma análise em comparação aos resultados obtidos de plataformas abertas, iniciando uma discussão quanto a validação do *framework* e metodologias para ambientes abertos e restritos.

2 Fundamentação Teórica

2.1 Introdução

Neste capítulo, são apresentados os tópicos que formam a base teórica deste trabalho. Iniciando pelo conceito de ecossistema de software, sua definição, conceitos básicos, origens e dimensões. Seguido pela apresentação do sistema de reputação e definição do Desenvolvimento Distribuído de Software, colaboradores chave, suas especialidade e reputação dentro do ecossistema para gerenciamento da empresa. Por fim, as considerações finais deste capítulo.

2.2 Ecossistema de Software

2.2.1 Introdução

Atualmente, empresas e fornecedores de software estão cada vez mais abrindo seus negócios para outras empresas de software e, como resultado, encontram-se inseridas em um ECOS com seus desenvolvedores e parceiros. Esses atores, especialmente os fornecedores de *software* que estão no centro do ecossistema, têm dificuldade em entender o ecossistema e como suas ações afetam seu desempenho e evolução (BERK; JANSEN; LUINENBURG, 2010).

Para modelar o ECOS, suas características formais precisam ser determinadas, como tamanho, meios de subsistência, a presença de padrões ou organizações de padrões e os diferentes papéis que os fornecedores de software podem desempenhar. O desafio de modelagem reside no fato de que atualmente não existem formalismos de modelagem para ECOS (BRINKKEMPER; SOEST; JANSEN, 2009). As empresas de software agora precisam considerar seu papel estratégico no ecossistema para sobreviver, mas o ECOS apresenta muitos novos desafios de pesquisa nos níveis técnico e de negócios (JANSEN; FINKELSTEIN; BRINKKEMPER, 2009).

Esta seção visa introduzir conceitos básicos sobre o ECOS, suas origens, propostas de modelagem e suas dimensões.

2.2.2 Definição

O estudo do ECOS na comunidade SE foi inicialmente motivado pela evolução de uma linha de *Software Product Line* (SPL) e Reuso de Software para permitir que desenvolvedores externos contribuam para plataformas até então fechadas em uma indústria de *software* global impulsionada pelo paradigma de *Component-Based Development* (CBD) (BOSCH, 2009; SANTOS, 2016).

Um ecossistema de software (ECOS) é um conjunto de atores e artefatos dentro e fora de uma organização ou comunidade que trocam recursos e informações em torno de uma plataforma de tecnologia comum (JANSEN; FINKELSTEIN; BRINKKEMPER, 2009). Esse contexto influencia as decisões de gerenciamento e desenvolvimento dessas plataformas, principalmente os modelos de arquitetura, governança e colaboração nos mais diversos domínios de aplicação. É necessário integrar mecanismos e ferramentas para apoiar a troca de informações, recursos e artefatos, bem como garantir a comunicação e interação entre desenvolvedores e usuários (SANTOS; VIANA, 2016).

O ECOS possui uma plataforma onde os produtos e serviços que fornece podem ser incluídos, modificados ou estendidos como artefatos de software (SANTOS, 2016). De acordo com (IANSITI; LEVIEN, 2004), em um Ecossistema de Software, há diferentes papéis que existem, sendo estes: *keystones*, *dominators* e *niche players*. Tanto o *keystone* quanto o *dominator* são *hubs* em um ecossistema. Suas definições são:

- *Keystone*: é a organização que lidera uma plataforma ECOS cria e compartilha valor com o resto do ecossistema.
- *Dominator*: extrai o máximo valor do ecossistema ao fornecer uma plataforma alternativa.
- *Niche Players*: são os *stackholders* que agem para desenvolver ou aprimorar capacidades especializadas que os diferenciam de outros *stackholders*, alavancando os recursos da rede enquanto ocupam apenas uma pequena parte da própria rede. Indi-

vidualmente, podem influenciar, comprometer-se, contribuir, promover ou estender a plataforma e juntos compõem a maioria do ecossistema em massa total e variedade. Em (HAGEL; BROWN; DAVISON, 2008), os *niche players* são subdivididos entre:

Influencer: comprometem-se cedo e proeminentemente com uma estratégia de modelagem.

Hedger: desenvolvem seus produtos ou serviços para suportar múltiplas plataformas de modelagem.

Disciple: se comprometem exclusivamente com uma única plataforma de modelagem.

Para o ECOS, analogamente ao ecossistema animal, *health* são todos os meios que um *hub* tem para melhorar a saúde de seu ecossistema, incluindo níveis de produtividade, robustez e criação de nichos (RAPPORT; COSTANZA; MCMICHAEL, 1998). Essas características do ecossistema de negócios e seus relacionamentos podem ser observados no modelo da Figura 2.1.

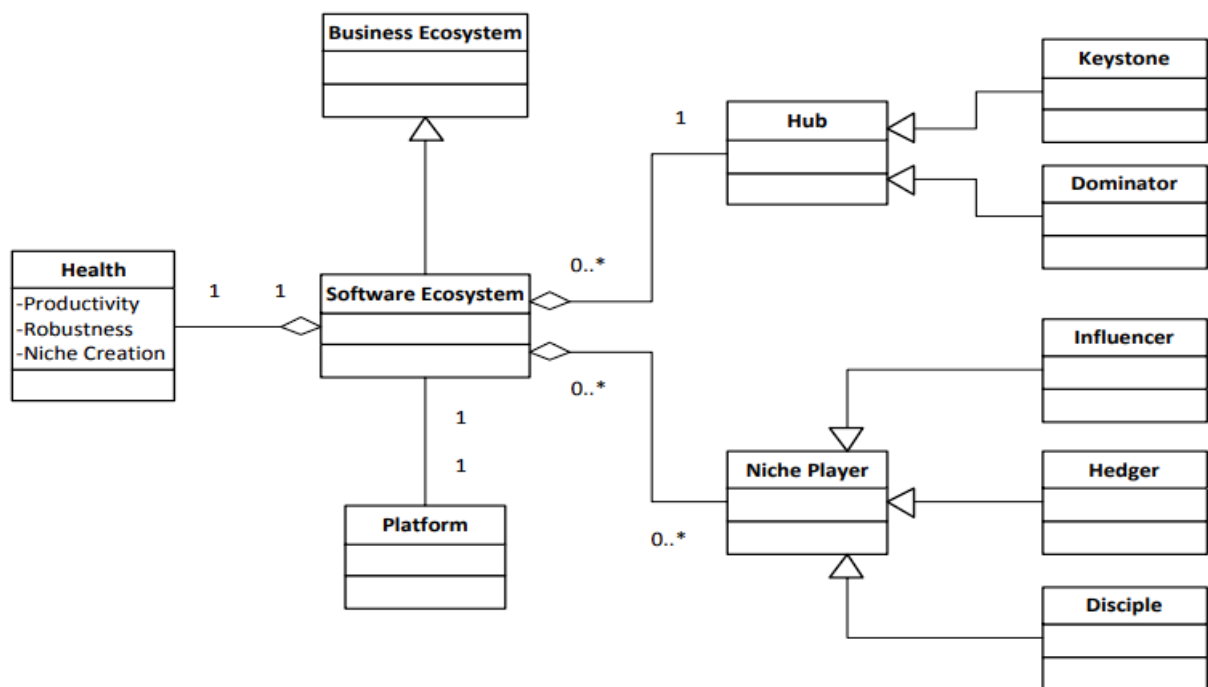


Figura 2.1: Modelo de domínio das relações entre os termos atualmente existentes em ECOS (BERK; JANSEN; LUINENBURG, 2010)

Nessas definições um termo essencial é mencionado, sendo a plataforma do ecos-

sistema. Uma plataforma de ecossistema é um conjunto de soluções para problemas disponibilizado aos membros do ecossistema através de um conjunto de pontos de acesso ou *interfaces* (IANSITI; LEVIEN, 2004). Em ECOS, a plataforma geralmente é um produto de software, mas também pode consistir em bibliotecas de software que *niche players* podem usar para extrair e criar valor (BERK; JANSEN; LUINENBURG, 2010).

2.2.3 Dimensões



Figura 2.2: Visão Tridimensional do ECOS (CAMPBELL; AHMED, 2010)

As empresas de software estão melhorando as operações de negócios em áreas como tecnologia, gerenciamento e processos de desenvolvimento de produtos para capturar a maior parte da participação de mercado e aumentar a lucratividade. No ambiente competitivo de hoje, a chave para o sucesso empresarial é a inovação, e as organizações estão continuamente adotando a inovação em áreas-chave das operações de negócios.

O conceito ECOS é derivado da arquitetura geral e da teoria do desenvolvimento de redes sociais, por isso é mais uma abordagem transitória, evolucionária e inovadora usando o conceito de linha de produtos de software. Para gerenciar a diversidade de organizações e relacionamentos criados em torno do ECOS por programadores, fornecedores,

parceiros e clientes/usuários, foi desenvolvido uma visão tridimensional (BARBOSA et al., 2013; CAMPBELL; AHMED, 2010) do ECOS representado pela Figura 2.2.

Essencialmente, todas as três dimensões são fortemente integradas por meio do processo de engenharia de software, amplamente aceitas pela literatura e adotadas em trabalhos subsequentes (GUERCIO, 2018).

Architectural Dimension

Também definida como dimensão técnica, possui foco na plataforma ECOS, ou seja, no mercado, tecnologia, infraestrutura ou organização, através do processo de engenharia de domínio da plataforma (estabelecendo seu ciclo de vida), gerenciando similaridade e variabilidade (definindo recursos da plataforma) e arquitetura SPL desenvolvida (tratando a plataforma como uma SPL) (SANTOS; WERNER, 2011).

Esta dimensão está mais relacionada aos níveis de escopo organizacional e Software Supply Networks (SSN) (visão interna) do que para o nível de escopo do ECOS (visão externa), visto que se concentra no elemento plataforma. Esta dimensão visa entender como o SE é aplicado na concepção, desenvolvimento e manutenção da plataforma (BERK; JANSEN; LUINENBURG, 2010; SANTOS; WERNER, 2011).

Social Dimension

Esta dimensão foca nos stackholders do ECOS, por meio do equilíbrio da proposta e a realização da utilidade à medida que as partes interessadas integram, ampliam e modificam o conhecimento e interagem entre si no ECOS. Também busca avaliar a promoção do ECOS, à medida que as capacidades e o engajamento das partes interessadas são implícitos e explicitamente reconhecidos e, por fim, visa o conhecimento sobre quais colaborações, desenvolvimento de código e outras oportunidades de redes sociais contribuem para as partes interessadas (SANTOS; WERNER, 2011).

Business Dimension

Esta dimensão foca no fluxo de conhecimento (ou seja, nos artefatos, recursos e informações da empresa, estabelecendo metas e planos de ação através de programas e

projetos), inovações ligando o ECOS ao mercado, e entendendo como, quando, onde e quem cumprirá as metas para o planejamento estratégico (SANTOS; WERNER, 2011).

2.3 Sistemas De Reputação

2.3.1 Introdução

As organizações investem pesadamente em seus sistemas de software. Eles constituem importantes ativos de negócios. No entanto, eles devem ser constantemente atualizados para manter seu valor. Diante disso e da necessidade de manter a produtividade e a qualidade, as empresas têm distribuído suas atividades geograficamente (LÉLIS, 2017).

Nesta seção será definido o Desenvolvimento Distribuído de Software empregado atualmente nas empresas e introduzido o conceito de colaborador chave, obtido através de um sistema de reputação.

2.3.2 Desenvolvimento Distribuído de Software

O novo ambiente, enfatizando o processo de globalização dos negócios, criou enormes desafios ao processo de desenvolvimento de software, que se tornou cada vez mais fragmentado e globalizado, aumentando a complexidade das empresas construindo sistemas de informação tornando o campo de desenvolvimento de software cada vez mais interdisciplinar, exigindo conhecimento e uso de conceitos e práticas de outras áreas. Daí a abordagem de Desenvolvimento Distribuído de Software (DDS), onde recursos humanos geograficamente distribuídos (programadores, *designers*, clientes, etc.) colaboram para realizar suas atividades durante o processo de desenvolvimento (AUDY, 2007).

Em geral, podem ser identificados três fatores que criam um ambiente propício ao desenvolvimento distribuído de *software* (todos relacionados à evolução do ambiente de negócios): globalização, crescente importância dos sistemas de informação nas empresas, terceirização de processos (*outsourcing*). O desenvolvimento de software é uma atividade complexa, por isso enfrenta vários desafios que podem ser descritos de muitas perspectivas diferentes (AUDY, 2007). Os principais desafios do desenvolvimento de software são baseados em três categorias.

- Pessoas: envolvem características que afetam diretamente os recursos humanos envolvidos no desenvolvimento de software, como capacitação de pessoal, coordenação de pessoas, motivação, produtividade e trabalho em equipe (AUDY, 2007).
- Processo: estão relacionados à forma como o projeto é desenvolvido, como análise de custo/benefício, especificação de requisitos, DDS, gerência de projeto, gerência de risco, gestão de conhecimento, manutenção de software, planejamento organizacional, processo de estimativa, reutilização, qualidade e teste de software (AUDY, 2007).
- Tecnologia: envolvem diferentes ferramentas e equipamentos técnicos que podem ser usados como suporte ou base para o desenvolvimento de software, como ferramentas de apoio e infra-estrutura de comunicação (AUDY, 2007).

A colaboração é essencial para o bom funcionamento das atividades distribuídas das equipes de manutenção que desejam colaborar e devem se organizar e focar em elementos como cooperação, comunicação e coordenação (FUKS; RAPOSO; GEROSA; LUCENA, 2003; LÉLIS, 2017). Para colaborar, os indivíduos devem trocar informações (comunicação), organizar (coordenação) e operar juntos em um espaço compartilhado (cooperação) (FUKS; RAPOSO; GEROSA; LUCENA, 2003). A reputação se torna um elemento chave, afetando estes elementos de colaboração (LÉLIS, 2017).

2.3.3 Reputação

No contexto do ECOS, é importante investigar a disponibilidade de informações sobre a reputação dos desenvolvedores, pois esse é um aspecto importante para melhorar a colaboração no contexto de equipes de desenvolvimento distribuídas e colaborativas. Fornecer recursos e mecanismos de visualização para apoiar a análise dessas informações tem o potencial de melhorar a descoberta e identificação de informações relevantes para os gestores tomarem decisões de alocação de desenvolvedores geograficamente distribuídos (LÉLIS, 2017).

A reputação é uma medida coletiva de confiabilidade baseada nas referências ou classificações dos membros de uma comunidade. Dados anteriores fornecem a reputação

atual do desenvolvedor. Por meio desse conceito, se percebe a possibilidade de mudança de reputação ao longo do tempo (LÉLIS, 2017). Os sistemas de reputação são empregados para facilitar a coleta, agregação e distribuição de informações de reputação sobre uma entidade (HENDRIKX; BUBENDORFER; CHARD, 2015).

Considerando o contexto de um desenvolvedor desempenhando tarefas de manutenção, sua reputação seria a percepção criada mediante as tarefas anteriormente tratadas. Os dados passados fornecem a reputação atual do desenvolvedor. Por meio desse conceito, se percebe a possibilidade da reputação mudar com o tempo.

2.3.4 Colaborador Chave

No presente trabalho, o termo Colaborador Chave será utilizado para definir aqueles colaboradores que possuem especialidade ou influência em determinada área de atuação. Os dados obtidos a partir da identificação destes colaboradores podem ser utilizados, por exemplo, em metodologias para o gerenciamento e manutenção de equipes.

2.4 Considerações Finais

O objetivo deste capítulo foi definir os conceitos-base presentes neste trabalho. Foi apresentado o conceito de Ecossistema de Software e sua importância no contexto atual de desenvolvimento de software, principalmente no meio empresarial.

A análise da relação entre as três dimensões do ECOS será o objeto principal de pesquisa para a geração de ativos de conhecimento para auxiliar no gerenciamento interno empresarial. Para isso, foi apresentado o conceito de Sistema de Reputação, presente no Desenvolvimento Distribuído de Software, e a importância na identificação de colaboradores chave. Estes conceitos já foram testados em trabalhos relacionados a ECOS em ambientes abertos, mas ainda não foram completamente validados em ambientes internos e privados de empresas de desenvolvimento de software.

3 Trabalhos Relacionados

3.1 Introdução

Neste capítulo são apresentados 6 artigos relacionados a este presente trabalho. Estes artigos elaboram a literatura de ECOS, métodos, de identificação de reputação para auxiliar a gestão de TI, métricas de redes complexas e outros aspectos do estudo ECOS. O presente trabalho, com base na literatura definida, aplica e valida essas metodologias em uma nova perspectiva. Os resultados obtidos pelos artigos relacionados foram utilizados para gerar um padrão de comparação entre eles e com este trabalho. Uma tabela é então apresentada, correlacionando aspectos dos testes realizados.

3.2 Managing and Monitoring Software Ecosystem to Support Demand and Solution Analysis

Em SANTOS (2016), a perspectiva de ECOS é examinada para propor e avaliar uma abordagem de gerenciamento e monitoramento para apoiar as atividades de gestão de TI, mais especificamente a análise de demanda e solução. A pesquisa iniciou uma investigação de um *framework* para modelagem e análise de ECOS para os seguintes objetivos:

1. Com base em uma revisão da análise da literatura, desenvolver um *framework* para ajudar os pesquisadores a entenderem melhor as dimensões e principais conceitos de ECOS e analisar as plataformas das organizações.
2. Identificar mecanismos de gestão de TI em governança e socialização no contexto de ECOS com base na opinião de especialistas.
3. Identificar os principais indicadores de monitoramento para a gestão de TI quanto à sustentabilidade de uma plataforma em um ambiente de ECOS, com base em estudos observacionais em cenários reais.

4. Definir métodos de gestão e monitoramento de ECOS para suporte às atividades de gestão de TI, mais especificamente análise de demanda e solução.
5. Avaliação de alguns módulos da abordagem com praticantes (gerentes e arquitetos de TI) realizando análise de demanda e solução em um cenário real.

Um estudo de viabilidade foi realizado com profissionais em um cenário real para avaliar a abordagem e contribuir para a pesquisa e a prática na comunidade ECOS. O piloto foi realizado inicialmente com três participantes. Após o refinamento, o estudo foi realizado com oito participantes. No entanto, o uso da abordagem não foi tão eficiente, pois leva algum tempo para que os praticantes aprendam a usá-lo em seu benefício.

Após analisar as respostas dos participantes às tarefas propostas e avaliações do estudo, há indicações de que a abordagem é aplicável para o gerenciamento e monitoramento do ECOS para apoiar as atividades de gestão de TI, especialmente a análise de demanda e soluções. Diversas oportunidades de melhoria foram identificadas, principalmente relacionadas à GUI. Dada a utilidade, os participantes se conscientizaram do impacto da perspectiva ECOS em suas atividades diárias (SANTOS, 2016). Também destaca a importância de manter uma plataforma sustentável.

A principal dificuldade da análise é quanto ao tamanho da amostra, com o pequeno número de participantes, o que não é o ideal do ponto de vista estatístico (SANTOS, 2016). Infelizmente, essa é uma dificuldade recorrente para estudos empíricos na área de SE, principalmente para métodos que requerem avaliação industrial, como neste caso. Portanto, o estudo apresenta uma limitação nos resultados, considerados como indicações (e não evidências). O presente trabalho contribui nesse sentido, fornecendo uma significativa amostra de dados para poder gerar evidências e prover validação ao *framework* modelado.

3.3 Um Modelo Dinâmico de Reputação para Apoiar a Manutenção Colaborativa de Software

LÉLIS (2017) apresenta um modelo para cálculo da reputação dos desenvolvedores de software, apoiado por técnicas de Dinâmica de Sistemas, o qual permite simular a evolução da reputação ao longo do tempo. Esse modelo serve como base para a construção de uma infraestrutura para informações de reputação dinâmica, cujo objetivo é poder gerenciar e monitorar informações de reputação de desenvolvedores distribuídos geograficamente para dar suporte à atribuição desses desenvolvedores a tarefas de manutenção. Além disso, fornece elementos de visualização e colaboração em um ambiente integrado às atividades de manutenção de software.

Demonstrações de provas de conceito e experimentos usando dados reais de uma empresa são mostrados para determinar a viabilidade e conformidade do modelo proposto, bem como outros recursos fornecidos pela infraestrutura. O experimento envolveu avaliar a viabilidade do modelo de reputação e, por meio da coleta de dados simulados, comparar as escolhas que os gerentes fazem ao atribuir desenvolvedores àqueles com o maior índice de reputação. A prova de conceito mostra a viabilidade de utilização e adesão à visualização, bem como os aspectos colaborativos proporcionados pela infraestrutura. Nos experimentos, os dados foram usados para construir as equações necessárias para simular o processo e observar o comportamento da reputação do desenvolvedor. Além disso, as informações simuladas podem ser associadas às escolhas dos gerentes ao atribuir os desenvolvedores às solicitações (LÉLIS, 2017).

A teoria da dinâmica de sistemas pode ser aplicada e, a partir dos dados obtidos, é possível entender o passado, estabelecer o que acontece no presente e projetar o comportamento futuro do objeto em análise. Modelos desse tipo podem ser considerados modelos preditivos desde que as informações geradas sejam estáveis e utilizem um banco de dados confiável. A infraestrutura desenvolvida por LÉLIS (2017), serve como base para o módulo de sistema de reputação do framework proposto neste trabalho, devido às características similares do cenário de teste e amostra de dados.

3.4 Topological Analysis in Scientific Social Networks to Identify Influential Researchers

GUÉRCIO et al. (2017) apresentam uma análise da colaboração, modelada a partir da estrutura de redes complexas, onde os nós da rede representam agentes que colaboram para atingir objetivos comuns. A partir daí, busca-se encontrar pesquisadores da rede que possuem maior importância na produção científica da rede modelada. Os nós mais importantes têm maior potencial de disseminação de informações e podem atingir diversos agentes, reduzindo o esforço e a perda de comunicação. Dessa forma, as informações produzidas e suportadas pelos usuários mais importantes da rede têm maior visibilidade. Além disso, agentes com maior influência na rede também possuem maior potencial de colaboração, pois estão fortemente conectados a um grande número de nós.

O objetivo é descrever como a topologia de rede pode ser usada para identificar pesquisadores que possam ter o maior potencial de colaboração na rede modelada. Após identificar os agentes com maior potencial de colaboração, foi realizada a remoção de nós para verificar o impacto dessa perda de pesquisador. O impacto é avaliado pela distância média de colaboração na rede, que indica a colaboração entre os nós que a compõem (GUÉRCIO et al., 2017).

Foi utilizada uma medida de centralidade para detectar a força da colaboração na rede social científica brasileira. Os resultados permitem a análise de redes complexas, observando o nível de colaboração entre pesquisadores, seus relacionamentos fortes e fracos e tenta identificar pesquisadores que desempenham um papel central na rede de coautoria. Além disso, a rede complexa foi classificada como uma rede livre de escala.

Os fundamentos da análise são aplicados no framework deste trabalho para auxiliar na identificação dos relacionamentos dos colaboradores entre as equipes, bem como sua influência e especialidade.

3.5 Ecosystems in GitHub and a Method for Ecosystem Identification Using Reference Coupling

Neste artigo, BLINCOE; HARRISON; DAMIAN (2015) propõem um novo método chamado acoplamento de referência é para detectar dependências técnicas entre projetos. Esse método estabelece dependências por meio de referências cruzadas entre projetos especificados pelo usuário.

O método foi utilizado para analisar o ECOS em projetos hospedados no GitHub e identificou diversas características. Constatou-se que a maioria dos ECOS está centrada em torno de um projeto e está interconectada com outros ECOS. O tipo predominante de ecossistemas são aqueles que desenvolvem ferramentas de apoio ao desenvolvimento de software (BLINCOE; HARRISON; DAMIAN, 2015).

Verificou-se também que o comportamento social dos proprietários do projeto se alinha bem com as dependências técnicas dentro do ecossistema, mas o comportamento social dos colaboradores do projeto era inconsistente com essas dependências (BLINCOE; HARRISON; DAMIAN, 2015).

O estudo demonstra a importância de ECOS para a manutenção e desenvolvimento colaborativo de projetos, da mesma forma que empresas com várias equipes trabalhando em projetos diferentes podem colaborar para melhorar a produtividade, qualidade e velocidade de seus respectivos projetos. Cenário que se enquadra à amostra de dados coletados neste trabalho.

3.6 Complex Network Analysis in a Software Ecosystem: Studying the Eclipse Community

O objetivo de GUERCIO et al. (2018) neste artigo, é modelar a dimensão social dos ECOS por meio de dados de contribuição. A avaliação das informações da dimensão social do SECO visa esclarecer como os desenvolvedores colaboram nos projetos e entre si. Este artigo visa identificar aspectos que destacam as interações entre os diferentes grupos de pesquisa. Essa interação reflete as ações de alguns colaboradores para ajudar outros em dificuldade em determinados aspectos dos projetos em que estão envolvidos.

Diferentes redes complexas foram modeladas usando dados extraídos do sistema de controle de versão do ecossistema Eclipse. As redes representam interações entre usuários e projetos. As análises de rede realizadas indicam a viabilidade de usar essa abordagem para identificar contribuintes significativos no ecossistema, melhorar a saúde e a eficiência de ECOS. Ao aumentar a força e o número de relacionamentos entre os membros ativos do ecossistema e os recém-chegados, é possível fortalecer o ecossistema e ajudar os novos membros a desenvolver novas capacidades que contribuirão para o ecossistema (GUERCIO et al., 2018).

Três redes diferentes foram modeladas. A primeira envolve as contribuições dos usuários do ecossistema para o projeto. A segunda rede avalia os colaboradores que fizeram comentários. Na terceira rede, foi usado uma medida R que usa informações das contribuições de cada usuário para traçar um relacionamento entre os usuários com base em suas contribuições (GUERCIO et al., 2018).

Após a modelagem dos dados, foi possível verificar se as redes estavam fortemente conectadas e como seus membros estavam distribuídos. Dentre as dificuldades encontradas, pode-se destacar o longo tempo gasto para extrair e limpar os dados, e a dificuldade em atribuir corretamente um valor de importância às contribuições. Neste trabalho serão implementados e aperfeiçoados novos métodos para atribuir valor no sistema de reputação e validar os modelos de redes na nova amostra de dados.

3.7 sSECO-Process: Avaliando a Dimensão Social em Ecossistemas de Software

GUERCIO (2018) apresenta o sSECO-Process, um processo de análise da dimensão social de ECOS, apoiado em técnicas de Redes Complexas, que permite apresentar as relações existentes em ECOS através da visualização e utilização de métricas relevantes da rede. Esse processo serve como base para analisar diferentes ECOS, e deu origem a tecnologias que dão suporte à coleta, tratamento, armazenamento e processamento de dados, que auxiliarão cientistas a realizar novas pesquisas relacionadas ao desenvolvimento de software. O trabalho também propõe medidas cujo objetivo é medir o nível de colaboração entre os usuários em um projeto, e como os usuários podem ser associados sem um registro das interações entre eles.

Foi realizada uma avaliação inicial com dados reais e suporte especializado para avaliar o processo desenvolvido, e após a avaliação inicial o processo foi ampliado para aprimorar e complementar as atividades realizadas durante a análise da dimensão social de ECOS (GUERCIO, 2018).

A fonte de dados escolhida para a extração foi o GitHub, por possibilitar que estudos com outros escopos sejam possíveis. A forma de extração, assim como o do presente trabalho, é via API, disponibilizada pela própria plataforma. Outra semelhança está nos próprios dados, que seguem padrões de informações similares, facilitando a replicação do processo de coleta, tratamento e análise desses dados.

3.8 Análise comparativa

Os trabalhos relacionados são comparados a um conjunto de características relacionadas aos seus dados, metodologia e contribuição na Tabela 3.1, organizada nas seguintes colunas:

- Trabalho: Referência ao trabalho relacionado.
- Dados: Representa a origem dos dados coletados.
- ECOS: Representa o ecossistema e/ou a plataforma de análise.

- Contribuição: Principal contribuição para o framework proposto neste trabalho.

Tabela 3.1: Tabela comparativa dos trabalhos relacionados.

Trabalho	Dados	ECOS	Contribuição
(SANTOS, 2016)	Planilhas e documentos	Organização Bancária	Framework para análise de ECOS
(LÉLIS, 2017)	Solicitações de mudanças e histórico de alterações	MantisBT	Infraestrutura de reputação dinâmica
(GUÉRCIO et al., 2017)	Relação entre autores de artigos científicos	Lattes Platform	Identificação de influência na rede
(BLINCOE; HARRISON; DAMIAN, 2015)	Código fonte	GitHub	Identificação de dependência entre projetos e comportamento social
(GUERCIO et al., 2018)	Código fonte e comunicação entre colaboradores	Eclipse	Identificação da interação entre diferentes grupos
(GUERCIO, 2018)	Código fonte	GitHub	Processo para análise de ECOS e métricas de redes complexas

3.9 Considerações Finais

O objetivo deste capítulo é apresentar trabalhos relacionados e precursores, seus objetivos, resultados, contribuições e deficiências. Cada trabalho elabora e detalha processos e/ou *frameworks* que contribuíram para a literatura e desenvolvimento deste TCC. Além disso, os testes e resultados apresentados promovem uma discussão comparativa com os resultados deste trabalho, avaliando as semelhanças e diferenças para auxiliar a análise de viabilidade desses modelos em ambientes de desenvolvimento de software profissional e privado.

Por fim, é apresentada uma tabela de análise comparativa entre os trabalhos com relação às suas características, ambientes, testes aplicados e área de contribuição.

4 Conclusões

4.1 Introdução

Neste capítulo é apresentado as etapas e o cronograma de execução proposto para o desenvolvimento desse trabalho de conclusão de curso, trata-se de uma estimativa incluindo a previsão de defesa do TCC. A conclusão do projeto também é apresentada.

4.2 Conclusão e Cronograma de Execução

Este trabalho fornece uma nova perspectiva para o estudo de ECOS em ambientes empresariais para auxiliar a gestão interna. Contribui principalmente ao fornecer uma amostra significativa de dados de cenários reais, que se mostraram difíceis de obter em trabalhos relacionados.

Com os dados obtidos da Empresa Parceira, diferentes projetos e suas equipes serão analisados com base no histórico de desenvolvimento de cerca de 100 desenvolvedores e no histórico de gestão das equipes. Com base nesses dados, pode ser conduzida uma análise detalhada da experiência dos desenvolvedores, reputação, impacto de sua ausência e recomendações para ajudar a gerenciar a equipe.

Essa análise fomenta a discussão da comparação dos resultados obtidos de ECOS abertos e privados, abrindo novas portas para aplicações comerciais desenvolvidas neste mercado em crescimento através do framework proposto.

O cronograma de execução estimado para o desenvolvimento da pesquisa é apresentado na Figura 4.1, organizado por atividades e em intervalos de duas semanas. A escrita da monografia será realizada durante os 7 primeiros intervalos e a apresentação está prevista para Janeiro de 2023.

A execução será dividida em 6 principais atividades, sendo:

- **Atividade 1:** Escrita da monografia e revisão da literatura.
- **Atividade 2:** Coleta e tratamento dos dados.
- **Atividade 3:** Construção do framework e aplicação das metodologias.
- **Atividade 4:** Realização dos testes, análise e comparação dos resultados.
- **Atividade 5:** Revisão do texto pelo orientador.
- **Atividade 6:** Defesa do TCC.

Tabela 4.1: Cronograma de execução.

Semanas	Atividade					
	1	2	3	4	5	6
12/09 à 25/09						
26/09 à 09/10						
10/10 à 23/10						
24/10 à 06/11						
07/11 à 20/11						
21/11 à 04/12						
05/12 à 18/12						
19/12 à 01/01						
02/01 à 15/01						

Bibliografia

- AUDY, J. L. N. *Desenvolvimento distribuído de software*. [S.l.]: Elsevier, 2007.
- BADASHIAN, A. S.; ESTEKI, A.; GHOLIPOUR, A.; HINDLE, A.; STROULIA, E. Involvement, contribution and influence in github and stack overflow. In: *CASCON*. [S.l.: s.n.], 2014. v. 14, p. 19–33.
- BARBOSA, O.; SANTOS, R. P. dos; ALVES, C.; WERNER, C.; JANSEN, S. A systematic mapping study on software ecosystems from a three-dimensional perspective. *Software ecosystems*, Edward Elgar Publishing, 2013.
- BERK, I. V. D.; JANSEN, S.; LUINENBURG, L. Software ecosystems: a software ecosystem strategy assessment model. In: *Proceedings of the fourth european conference on software architecture: Companion volume*. [S.l.: s.n.], 2010. p. 127–134.
- BLINCOE, K.; HARRISON, F.; DAMIAN, D. Ecosystems in github and a method for ecosystem identification using reference coupling. In: IEEE. *2015 IEEE/ACM 12th Working Conference on Mining Software Repositories*. [S.l.], 2015. p. 202–211.
- BOEHM, B. A view of 20th and 21st century software engineering. In: *Proceedings of the 28th international conference on Software engineering*. [S.l.: s.n.], 2006. p. 12–29.
- BOSCH, J. From software product lines to software ecosystems. In: *SPLC*. [S.l.: s.n.], 2009. v. 9, p. 111–119.
- BRINKKEMPER, S.; SOEST, I. v.; JANSEN, S. Modeling of product software businesses: Investigation into industry product and channel typologies. In: *Information Systems Development*. [S.l.]: Springer, 2009. p. 307–325.
- CAMPBELL, P. R.; AHMED, F. A three-dimensional view of software ecosystems. In: *Proceedings of the Fourth European Conference on Software Architecture: Companion Volume*. [S.l.: s.n.], 2010. p. 81–84.
- FUKS, H.; RAPOSO, A. B.; GEROSA, M. A.; LUCENA, C. J. P. Do modelo de colaboração 3c à engenharia de groupware. *Simpósio Brasileiro de Sistemas Multimídia e Web-Webmidia*, p. 0–8, 2003.
- GUERCIO, H. *sSECO-Process: Avaliando a Dimensão Social em Ecossistemas de Software*. Tese (Doutorado) — Ph. D. Dissertation. Universidade Federal de Juiz de Fora, 2018.
- GUÉRCIO, H.; STRÖELE, V.; DAVID, J. M. N.; BRAGA, R.; CAMPOS, F. Topological analysis in scientific social networks to identify influential researchers. In: IEEE. *2017 IEEE 21st International Conference on Computer Supported Cooperative Work in Design (CSCWD)*. [S.l.], 2017. p. 287–292.
- GUERCIO, H.; STROELE, V.; DAVID, J. M. N.; BRAGA, R.; CAMPOS, F. Complex network analysis in a software ecosystem: Studying the eclipse community. In: IEEE. *2018 IEEE 22nd International Conference on Computer Supported Cooperative Work in Design ((CSCWD))*. [S.l.], 2018. p. 618–623.

- HAGEL, J.; BROWN, J. S.; DAVISON, L. Shaping strategy in a world of constant disruption. *Harvard Business Review*, v. 86, n. 10, p. 80–89, 2008.
- HENDRIKX, F.; BUBENDORFER, K.; CHARD, R. Reputation systems: A survey and taxonomy. *Journal of Parallel and Distributed Computing*, Elsevier, v. 75, p. 184–197, 2015.
- IANSTITI, M.; LEVIEN, R. *The keystone advantage: what the new dynamics of business ecosystems mean for strategy, innovation, and sustainability*. [S.l.]: Harvard Business Press, 2004.
- JANSEN, S.; FINKELSTEIN, A.; BRINKKEMPER, S. A sense of community: A research agenda for software ecosystems. In: IEEE. *2009 31st International Conference on Software Engineering-Companion Volume*. [S.l.], 2009. p. 187–190.
- LÉLIS, C. A. S. *Um modelo dinâmico de reputação para apoiar a manutenção colaborativa de software*. Tese (Doutorado) — Ph. D. Dissertation. Universidade Federal de Juiz de Fora, 2017.
- RAPPORT, D. J.; COSTANZA, R.; MCMICHAEL, A. J. Assessing ecosystem health. *Trends in ecology & evolution*, Elsevier, v. 13, n. 10, p. 397–402, 1998.
- SANTOS, R.; VIANA, D. Ecossistemas de software no desenvolvimento de plataformas para web, redes sociais e multimídia. *Sociedade Brasileira de Computação*, 2016.
- SANTOS, R. P. dos. *Managing and monitoring software ecosystem to support demand and solution analysis*. Tese (Doutorado) — Ph. D. Dissertation. Universidade Federal do Rio de Janeiro, 2016.
- SANTOS, R. P. dos; WERNER, C. M. L. A proposal for software ecosystems engineering. In: *IWSECO@ ICSOB*. [S.l.: s.n.], 2011. p. 40–51.