

Avaliação da atuação colaborativa de membros de um projeto acadêmico com foco nas responsabilidades definidas

Bernardo Veloso de Resende¹, Gláucia Braga e Silva¹

¹Instituto de Ciências Exatas e Tecnológicas
Universidade Federal de Viçosa (UFV) - campus Florestal
Florestal – MG – Brasil

{bernardo.resende, glaucia}@ufv.br

Resumo. *Este trabalho utiliza os dados de um projeto acadêmico para o ensino de Engenharia de Software como estudo de caso para avaliar a participação colaborativa dos envolvidos. Para isso, os dados dos repositórios das ferramentas computacionais de apoio ao projeto são representados por meio de grafos e submetidos à aplicação de métricas de centralidade. A intenção é buscar uma relação entre os nós mais centrais, segundo duas das dimensões do Modelo 3C de Colaboração (Coordenação e Cooperação), em função dos papéis assumidos pelos alunos. Os resultados obtidos mostraram que há uma grande participação dos papéis gerenciais nas tarefas de Coordenação, assim como um maior número de atividades de Cooperação realizadas por alguns dos Programadores e Testadores. Além disso, foi possível perceber que alguns alunos tiveram baixo envolvimento, não condizente com o papel assumido.*

1. Introdução

No universo acadêmico, uma tarefa muitas vezes é realizada não só por um, mas por grupos de alunos colaborando em prol de uma atividade. A realização de atividades acadêmicas com vários envolvidos traz consigo muitas vantagens, visto que, as diferentes habilidades e formas de pensar favorecem a troca e o complemento de conhecimento entre os participantes [Damiani 2008].

Esse universo de colaboração no âmbito acadêmico pode ser encontrado em disciplinas dos cursos na área de Computação, quando os docentes aplicam atividades práticas desenvolvidas por vários alunos de forma conjunta. Trabalhos práticos que envolvem projetos de desenvolvimento de software são exemplos muito utilizados para o ensino integrado de várias disciplinas, visando aproximar a academia do mercado de trabalho [Braga e Silva et al. 2018]. Nesse contexto, os alunos assumem responsabilidades dentro do projeto, o que gera diferentes tipos de participação com diferentes níveis de envolvimento por parte de cada um. Essas questões são de grande importância para a avaliação final do professor, além de ter influência direta nas relações e nos resultados dos projetos.

Visto que os alunos estão atuando como desenvolvedores em um projeto de software, as mesmas relações de colaboração podem ser observadas, já que eles se comunicam e interagem para produzir diversos artefatos [Wiese et al. 2015]. Como as questões relacionadas à comunicação e à coordenação são cruciais em um processo de software [Curtis et al. 1988], elas devem ser tratadas com atenção também neste universo acadêmico, para que as atividades sejam devidamente cumpridas e os resultados pretendidos sejam alcançados.

Assim como em ambientes de desenvolvimento tradicionais, para apoiar essas atividades colaborativas no contexto dos projetos acadêmicos são usadas ferramentas computacionais, seja para apoiar a comunicação entre os envolvidos, para coordenar as tarefas ou para controlar a produção cooperativa de artefatos. Dessa forma, os repositórios dessas ferramentas representam uma rica fonte de informação acerca da memória do trabalho da equipe, que se devidamente analisada, pode revelar importantes características sobre o comportamento dos envolvidos enquanto colaboradores no processo de desenvolvimento. Em busca de soluções para analisar repositórios de software, alguns trabalhos, como [Dabbish et al. 2012] e [Balieiro et al. 2007], analisam as relações entre os desenvolvedores utilizando o modelo de redes sociais. Esse tipo de abordagem permite modelar a rede no formato de um grafo, e então, aplicar métricas de análise que consigam quantificar o grau de engajamento dos participantes.

Este trabalho objetiva analisar a participação de alunos em um projeto de desenvolvimento de software no contexto acadêmico. O intuito é verificar o comportamento colaborativo dos alunos segundo as responsabilidades atribuídas a eles. Para isso, com base nos dados das ferramentas utilizadas, serão criadas duas redes, uma de coordenação e outra de cooperação, em que os alunos serão dispostos como nós e suas relações serão estabelecidas com base atividades realizadas em conjunto e nos artefatos produzidos de forma cooperativa. Assim, métricas da teoria de grafos serão aplicadas com o intuito de quantificar a influência de cada nó na rede.

Essa análise poderá fornecer, aos professores e coordenadores dos projetos, indicadores que apoiem a atribuição de notas nestes trabalhos de forma mais objetiva, uma vez que atribuir notas individuais para trabalhos em grupo é sempre um desafio [Gatti 2003]. Além disso, essa análise pode revelar alguns pontos de melhoria na metodologia de desenvolvimento adotada nos projetos, seja quanto à divisão de papéis e responsabilidades ou ainda quanto à adoção das ferramentas de apoio.

O texto deste trabalho está organizado de forma que, na Seção 2, são apresentados conceitos relacionados à colaboração no contexto acadêmico. Na Seção 3, são discutidas as análises e as métricas relacionadas às redes sociais. A Seção 4 trata dos trabalhos relacionados. Na Seção 5, descreve-se toda a análise realizada, desde a extração dos dados até a aplicação das métricas sobre as redes de colaboração, destacando-se os resultados obtidos. Por fim, a Seção 6 traz a conclusão do trabalho.

2. Colaboração no contexto acadêmico

As atividades em grupo se aplicam a diversos contextos e trazem muitos benefícios. No contexto acadêmico, esse tipo de trabalho favorece a interação entre os alunos. Segundo Fulford e Zhang [Fulford and Zhang 1993], alunos com altos níveis de interação possuem atitudes mais positivas e altos níveis de realização. Além disso, possibilita-se a troca de conhecimento, e assim, aumenta-se a qualidade do aprendizado, resultando em uma melhor formação.

Neste trabalho, a colaboração vai ser abordada segundo os princípios do Computer Supported Cooperative Work - CSCW (Trabalho Cooperativo Auxiliado por Computador). Ellis et al. [Ellis et al. 1991] apresentam uma discussão mais ampla, definindo o CSCW como “uma área científica interdisciplinar que estuda a forma como o trabalho em grupo pode ser auxiliado por tecnologias da informação e comunicação”. Esse apoio

computacional à colaboração pode ser classificado em três dimensões: comunicação, coordenação e colaboração, formando o chamado Modelo 3C, proposto originalmente por Ellis et al. [Ellis et al. 1991] (Figura 1).

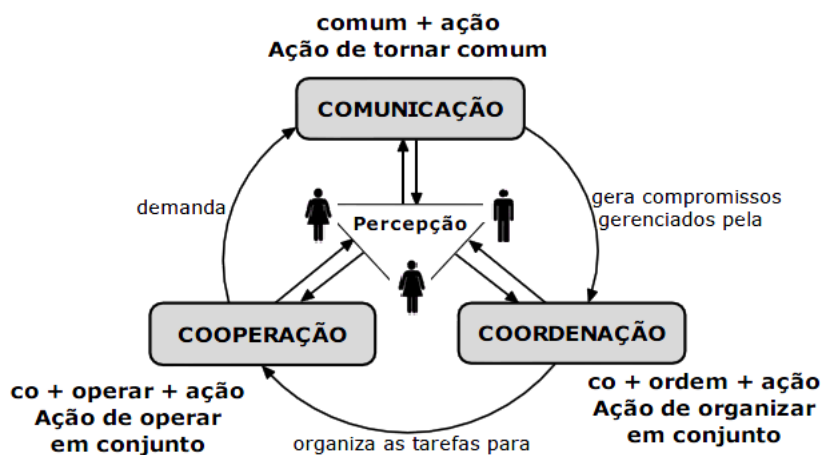


Figura 1. Modelo 3C adaptado por Gerosa [Gerosa 2006]

Fucks et al. [Fuks et al. 2003] utilizam o Modelo 3C para abordar cada uma das três dimensões separadamente. Para eles, a comunicação é caracterizada pela troca de mensagens e pela argumentação entre as pessoas; a coordenação é caracterizada pelo gerenciamento de pessoas, atividades e recursos; e a cooperação é a atuação conjunta visando a produção de objetos ou informações.

O Modelo 3C será usado para avaliar a colaboração dos alunos, considerando o papel assumido por eles enquanto participantes de um projeto acadêmico. Em virtude dos dados disponíveis para análise, neste trabalho, serão avaliadas apenas as dimensões de Coordenação e Cooperação.

3. Análise de redes sociais

No desenvolvimento de um projeto de software, os integrantes trabalham de forma conjunta, realizando diversas atividades em busca da produção de artefatos. Esses participantes e suas ações realizadas durante o processo podem ser visualizados como uma rede social de desenvolvedores, representada por um grafo [Jiang et al. 2013].

Um grafo é um par ordenado de conjuntos $G = (V, E)$, onde $E = \{ij = \{i, j\}\}$, sendo que i e j devem pertencer a V . Os elementos de $v \in V$ são chamados de nós de G e os elementos de $e \in E$ são chamados de arestas [Tutte 1963]. No contexto de redes sociais, os nós representam as pessoas e as arestas, as ações e interações entre elas.

A partir dessa representação, métricas da teoria de grafos podem ser aplicadas. A utilização das métricas tem o intuito de investigar a estrutura de uma rede, obtendo informações importantes sobre os envolvidos e suas interações [Borba 2013]. Um aspecto relevante, por exemplo, é identificar quais são os elementos mais importantes de uma rede.

3.1. Métricas de análise

Métricas de análise aplicáveis à rede de colaboração visam verificar quais são os elementos principais ou centrais de uma rede. Medidas de Centralidade são uma forma de

quantificar essa relevância. Neste trabalho, serão usadas três métricas de centralidade definidas por Freeman e Linton [Freeman 1978]:

- Centralidade de grau: definida como o número de ligações incidentes de um vértice. O grau associado pode ser entendido como a probabilidade que o vértice tem de receber alguma informação da rede. Para um grafo não-orientado, a centralidade de grau de um vértice V é dada por seu grau, ou seja:

$$C_D(v_i) = k_i$$

- Centralidade de intermediação: quantifica quantas vezes um nó age como ponte ao longo do caminho mais curto entre dois outros nós. Esses nós são pontos de articulação na rede.

$$C_B(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

Onde σ_{st} é o número total de caminhos mais curtos desde o nó s ao nó t e $\sigma_{st}(v)$ é o número desses caminhos que passam por v .

- Centralidade de proximidade: é o número de arestas que devem ser percorridas para que um vértice possa ter acesso a todos os outros vértices. Representa a velocidade com que o nó pode disseminar informações pela rede. A centralidade de proximidade é dada pelo recíproco da soma das suas distâncias aos demais nós, ou seja:

$$C_c(v) = \frac{1}{\sum_{a \in V} d(v,a)}$$

4. Trabalhos relacionados

O trabalho apresentado por Thung et al. [Thung et al. 2013] usa os dados do Git para modelar uma rede social colaborativa. Nele, os autores utilizam os dados gerados por interações entre os desenvolvedores para avaliar a importância de cada um dentro do projeto de software. Neste trabalho, a análise será feita sobre os dados de um projeto de desenvolvimento no contexto acadêmico. Outro ponto é que, além do Git, os repositórios de duas outras ferramentas serão considerados. Ainda no contexto de análise de redes no Git, Rocha et al. [Rocha et al. 2016] definiram que a influência da contribuição dos desenvolvedores em linguagens de programação nos repositórios do Github, por meio da centralidade de grau. Entretanto, o trabalho não avaliou a colaboração entre os membros no repositório. Neste trabalho, a análise da colaboração será realizada por meio da aplicação das métricas de centralidades de proximidade, de grau e de intermediação com o intuito de avaliar a influência de cada aluno na rede como um todo nas dimensões de Coordenação e Cooperação.

Fontoura et al. [da Fontoura et al. 2013] constroem redes de Comunicação e de Cooperação, a partir de repositórios gerados pela interação de desenvolvedores, com o intuito de identificar possíveis falhas no processo em cada uma das dimensões. Aqui, serão criadas redes de Coordenação e de Cooperação, onde métricas de análise de redes são aplicadas visando relacionar a participação dos alunos com os papéis anteriormente assumidos por eles.

Wiese e Kuroda [Wiese et al. 2015] analisam dados de um projeto de desenvolvimento de software com o intuito de identificar os nós mais importantes no quesito

comunicação. Para isso, eles constroem uma rede e aplicam métricas da Teoria de Grafos. Aqui, uma rede será gerada para cada dimensão do Modelo 3C, onde a intenção é, não somente identificar os nós mais importantes em cada dimensão, mas relacioná-los com as funções a eles atribuídas dentro de um projeto.

5. Análise de colaborações no contexto de um projeto acadêmico

Esta Seção descreve os passos realizados na avaliação colaborativa de alunos no contexto de um projeto de software acadêmico, detalhando a obtenção dos dados, a partir dos repositórios das ferramentas utilizadas, a criação das redes de coordenação e cooperação representadas por grafos, a aplicação das métricas de análise e os resultados obtidos.

5.1. Contexto do estudo de caso

A abordagem utilizada neste trabalho será aplicada no contexto de um projeto acadêmico para o ensino de Engenharia de Software, que vem sendo realizado anualmente, desde 2016, na UFV *campus* Florestal. O projeto abrange o desenvolvimento de um software, de forma integrada, envolvendo alunos de quatro disciplinas do curso de Ciência da Computação: Banco de Dados (BD), Programação Orientada a Objetos (POO), Arquitetura de Software (AS) e Engenharia de Software II (ES) [Braga e Silva et al. 2018]. Este trabalho utilizará apenas os dados da edição de 2017, quando o projeto envolveu o desenvolvimento de um sistema de busca de trabalhos de conclusão de curso, chamado “GetPoc”.

No início do projeto, os alunos foram distribuídos em papéis, onde passaram a ter responsabilidades dentro do processo de desenvolvimento e limitados pelo escopo da disciplina em que estavam matriculados. A relação pode ser vista na Tabela 1:

Tabela 1. Relação de papéis e responsabilidades no Projeto Interdisciplinar

Papel	Responsabilidade	Disciplina
Gerente de Projeto	Controlar o andamento do projeto	ES
Gerente de Processo	Ajustar e refinar o processo adotado	ES
Gerente de Conhecimento	Disponibilizar informações do projeto e do produto	ES
Especialista em ferramentas	Configurar todo o ambiente de ferramentas	ES
Designer	Desenvolver a interface gráfica	ES
Analista de Qualidade	Garantir a qualidade (modelos, código, ...)	ES
Testador	Realizar testes no software	ES
Scrum Team Master	Dar suporte aos estudantes e aos professores	ES
Arquiteto de Software	Especificação de Software	AS
	Revisão dos modelos de banco de dados	AS
Programador	Revisão do código	AS
	Codificação da camada de apresentação	AS
	Código de integração entre as camadas (MVC)	AS
Programador O.O	Codificação da camada de modelo	POO
	Codificação da camada de controle	POO
	Especificação das consultas ao banco de dados	POO
Desenvolvedor de BD	Modelagem lógica do BD	BD
	Modelagem física do BD	BD
Analista de BD	Criação do banco e inserção de dados	BD
	Desenvolvimento das consultas SQL	BD

A partir da tabela, é possível observar que os estudantes de ES foram responsáveis por papéis de grande responsabilidade, incluindo papéis gerenciais. Os Scrum Team Master tinham o papel de ser uma ponte de comunicação, esclarecendo dúvidas e garantindo o trabalho das equipes de POO e BD. Os Gerentes de Projeto tinham a responsabilidade de elaborar o cronograma do projeto, monitorar prazos, atribuir atividades, dando o suporte necessário para que tudo ocorresse da maneira planejada. O papel de Gerente de Processos era responsável por divulgar o processo de desenvolvimento, orientar a equipe, realizar medições e realizar os refinamentos necessários. Já o Gerente de Conhecimento tinha a função de disponibilizar uma coleção de documentos e informações relevantes ao projeto, a partir de uma *Wiki*.

Os Arquitetos são responsáveis pelo design da arquitetura do sistema, seleção de frameworks e padrões de projeto aplicáveis. Eles também apoiam os programadores nas tarefas de integração entre as camadas. Os Analistas foram responsáveis por elaborar as especificações de requisitos e modelos do sistema. Já os Programadores ficavam com a tarefa de revisar os códigos da camada de modelo e controle vindos de POO, além de codificar a camada de apresentação e ter que realizar a integração entre as três partes.

Na edição de 2017, para apoiar o trabalho dos alunos envolvendo o desenvolvimento colaborativo do sistema "GetPoc", foram utilizadas as seguintes ferramentas:

- Trello: software de apoio à coordenação de atividades, que permite a distribuição de tarefas, a definição de deadlines e o controle de status das tarefas.
- Git: sistema de apoio à gestão de configuração do projeto, usado para apoiar as tarefas de *issue tracking* e o controle de versões dos artefatos.

Percebe-se que os papéis gerenciais pelas responsabilidades atribuídas a eles, tem um caráter forte de Coordenação. A participação constante em definições de cronogramas e atribuição de tarefas faz com que muitas ações desse tipo sejam esperadas de alunos com esse tipo de responsabilidade. Isso acaba ocasionando um uso constante da ferramenta Trello, que tem o papel de apoiar esse tipo de atividade. Da mesma forma se espera grande uso do Git por parte de papéis que tenham a Cooperação como dimensão fundamental, estando diretamente envolvidos com o desenvolvimento e a revisão de código, assim como a produção de especificações do sistema, como Programadores, Testadores e Analistas de Sistemas.

5.2. Obtenção e refinamento dos dados colaborativos

A primeira etapa foi lidar com o modo de acesso aos dados contidos nas ferramentas. Ambas foram utilizadas em sua versão gratuita, o que restringiu a escolha de como seria o acesso aos repositórios.

No Trello, o backup dos dados foi através de arquivos *JSON*. A própria ferramenta permitiu que um arquivo desse tipo fosse criado para cada um dos quadros analisados. Nesses arquivos, cada cartão dentro de um quadro recebe uma estrutura própria, contendo todos os dados a seu respeito. Com isso, quando um aluno realiza uma ação, seu nome e o que foi feito fica registrado na estrutura desse cartão.

Uma aplicação Java foi criada para interpretar o repositório de cada ferramenta. No caso do Trello, foi utilizada uma biblioteca própria para trabalhar com a leitura de

dados desse tipo, a *orgJSON*. A versão utilizada foi a 20170516. A biblioteca *orgJSON* foi utilizada para acessar as ações realizadas em cada cartão. Com isso, uma tabela no formato CSV foi construída contendo o nome dos cartões nas colunas e o nome dos alunos nas linhas. O número de vezes que um aluno havia realizado alguma ação em um dos cartões era preenchido no espaço de cruzamento.

Por fim, o acesso aos dados do Git foi realizado através da API(*Application Programming Interface*) oficial da ferramenta. A biblioteca foi utilizada na sua versão 4.0 para Java e oferece uma interface de acesso ampla aos *commits* realizados. Os dados do Git também deram origem a uma tabela CSV. A aplicação usava a API para importar uma lista com o nome de todas as pessoas que realizaram um *commit*. Após isso, eram construídas as relações de cooperação, considerando os alunos que fizeram um *commit* em sequência no mesmo artefato. Por fim, essas relações eram dispostas na tabela.

5.3. Métricas de análise aplicáveis

Esta seção apresenta as métricas definidas na Seção 3.1, aplicadas no contexto da análise de colaborações de um projeto acadêmico, segundo as dimensões Coordenação e Cooperação do Modelo 3C.

- **Centralidade de Grau:** No contexto de colaboração em projetos de software, essa métrica está relacionada à quantidade de vezes que um membro colaborou em relação aos demais. Assim, um aluno que seja influente na rede certamente estará conectado com muitos outros.
- **Centralidade de Intermediação:** A centralidade de intermediação vai mostrar quantas vezes um aluno serviu como ponte de ligação para que dois outros alunos pudessem colaborar. Isso mostra a influência de um nó, não só no seu ato de colaborar, mas também no ato de fazer com que outros alunos possam criar uma relação de colaboração.
- **Centralidade de Proximidade:** Aqui, a proximidade é importante para analisar quão próximo um aluno é em relação a todos os outros nós na rede. Dessa forma, um aluno que se relaciona com vários outros estará numa posição central.

5.4. Criação e análise das redes de colaboração

As redes foram criadas com base no Modelo 3C, de forma que, cada uma das duas dimensões abordadas foi analisada separadamente. As dimensões de Coordenação e Cooperação foram modeladas como grafos ponderados e não-direcionados, tendo em vista que a colaboração não possui direção, tanto o nó que envia quanto o nó que recebe estão colaborando no desenvolvimento do software de maneira igual. A característica de ponderar as relações se deve ao fato de que a colaboração entre um par de nós pode ser quantificada conforme eles interagem no decorrer do projeto. Aos nós da estrutura, que representam os estudantes envolvidos, foram atribuídos os papéis e não os nomes, preservando a identidade de cada um.

O software escolhido para gerar os grafos e aplicar as métricas foi o NodeXL¹. Trata-se de um software gratuito, de código aberto e que foi utilizado na sua versão básica.

¹<https://nodexl.com/>

Essa ferramenta trabalha acoplada ao Microsoft Excel, o que justifica a escolha do formato CSV para as tabelas.

Em cada uma das redes foram aplicadas as três métricas de análise utilizadas (centralidade de grau, de proximidade e de intermediação). A intenção aqui era avaliar quais eram os nós mais representativos segundo os resultados das métricas.

5.4.1. A dimensão da Coordenação

Na criação da rede de Coordenação, foram utilizados os dados provenientes do Trello. A ferramenta dá suporte à criação de cartões, que nada mais são do que listas de atividades a serem executadas. Cada um dos cartões era atribuído aos alunos que tinham em seu papel a responsabilidade de cumprir as tarefas ali contidas. Somente quem participa de um cartão pode visualizar qualquer ação relacionada a ele. Algumas das ações que poderiam ser realizadas por um aluno dentro de um cartão são: atualizar o status de um cartão, adicionar um membro ou atualizar uma checklist.

Cada ação realizada por um aluno foi contabilizada como uma relação de coordenação com os demais alunos presentes naquele cartão. Assim, a rede foi formada de forma que, cada aluno corresponde a um nó e a aresta de ligação entre os alunos representa o número de vezes que eles interagiram em um mesmo cartão.

Ao todo, foram obtidos dados de 116 cartões, onde 37 alunos realizaram juntos 3777 ações. A rede gerada por essa dimensão pode ser vista na Figura 2. O nó com os melhores resultados, na média, considerando as três métricas, é referente ao papel de Gerente de Projeto 1 e está destacado, assim como suas relações com os outros integrantes da rede.

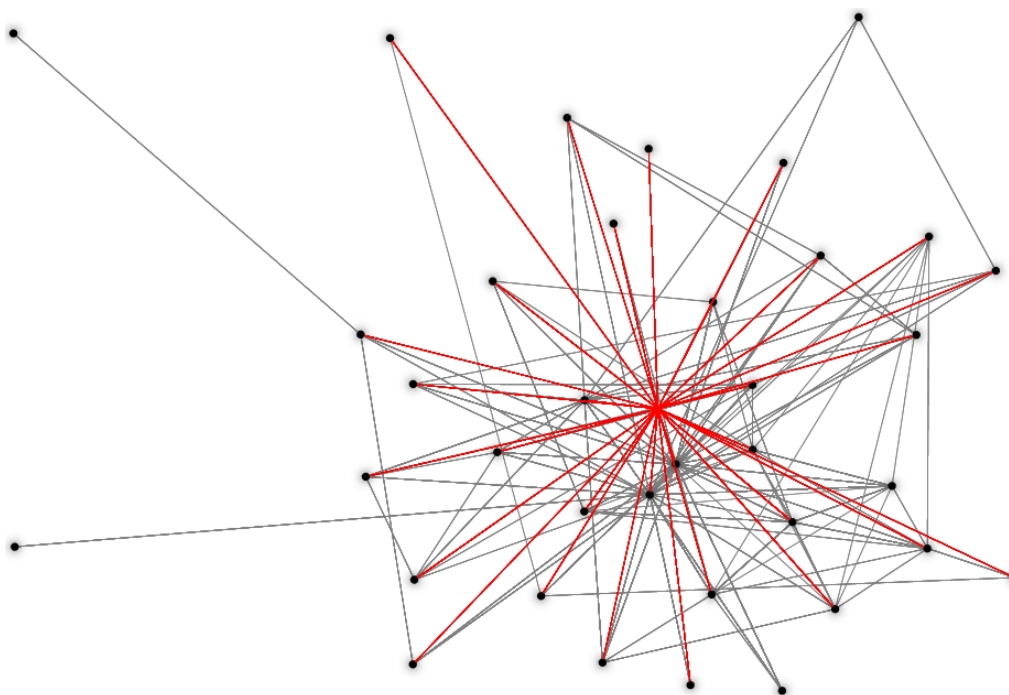


Figura 2. Rede de Coordenação

A Figura 3 apresenta os resultados gerados pela aplicação das métricas. Cada métrica teve seus resultados dispostos em uma tabela diferente, exibindo os 10 melhores colocados, com seus papéis e seus respectivos scores.

Centralidade de Intermediação		Centralidade de Proximidade		Centralidade de Grau	
Papel	Score	Papel	Score	Papel	Score
Gerente de Projeto 1	236.77	Gerente de Projeto 1	0.024	Gerente de Projeto 1	31
Analista de Qualidade	135.99	Gerente de Projeto 2	0.022	Gerente de Projeto 2	26
Gerente de Projeto 2	121.05	Analista de Qualidade	0.021	Analista de Qualidade	25
Testador 1	35.00	Testador 1	0.017	Testador 1	14
Especialista em Ferramentas	18.71	Especialista em Ferramentas	0.016	Especialista em Ferramentas	12
Gerente de Processo	8.64	Testador 2	0.016	Testador 2	10
Testador 2	4.50	Programador	0.016	Programador 1	10
Testador 3	3.80	Gerente de Processo	0.016	Gerente de Processo	9
Arquiteto de Software 1	3.00	Gerente de Conhecimento	0.015	Gerente de Conhecimento	9
Programador	2.32	Testador 4	0.015	Arquiteto de Software 1	9

Figura 3. Resultado das métricas para a Coordenação

Analisando os resultados, é possível observar que as primeiras posições, em ambas as métricas, contém os alunos que assumiram papéis de gerência. Isso é condizente com a função exercida por eles, já que era esperado que tivessem um número significativo de ações de coordenação. Outro papel que esteve entre as três primeiras posições foi o de Analista de Qualidade. Sua presença na segunda posição da tabela de centralidade de intermediação é justificada pela responsabilidade em definir padrões durante todas as etapas do projeto, tendo relação constante com a maioria dos outros papéis.

Os Testadores e o Especialista em Ferramentas também tiveram destaque. Mesmo não existindo código em etapas iniciais do processo, os Testadores são responsáveis por fazer inspeções de qualidade nos modelos produzidos pelos Analistas e nos incrementos produzidos pelas equipes de BD e POO. Isso faz com que sempre existam cartões atribuídos a eles no Trello. Já o Especialista de Ferramentas ganha destaque por, além de ser responsável por configurar as ferramentas, ter também a função de criar os quadros e cartões no Trello.

É possível perceber também como alunos com o mesmo papel tiveram diferentes níveis de participação. Esse exemplo pode ser ilustrado pela comparação entre os papéis de Gerente de Projeto 1 e 2. Por mais que o Gerente de Projeto 2 tenha participado ativamente, o Gerente 1 também teve destaque em todas as métricas aplicadas. Outro exemplo é o do papel de Testador, onde uma pequena variação de alunos aparece em destaque nas tabelas, deixando evidente a fraca participação por parte dos outros alunos com as mesmas funções.

5.4.2. Dimensão da Cooperação

Essa rede foi construída utilizando os *commits* realizados na ferramenta GitLab. Um *commit* é uma ação de cooperação, já que o aluno está contribuindo ativamente com o desenvolvimento do código junto aos outros alunos.

Na construção das arestas de ligação, foi considerado que, quando dois alunos realizam *commits* sequenciais em um mesmo artefato, eles automaticamente estão tendo

uma relação de cooperação, já que estão contribuindo para a realização de uma tarefa relacionada. Assim, cada aluno que realizou um *commit* é um nó na rede e tem uma relação contabilizada com quem realizou um *commit* relacionado. O total de interações entre dois alunos representa o peso da aresta entre eles.

No total, 26 alunos realizaram 394 *commits*, gerando 113 relações diferentes ao longo do projeto. A rede gerada para a Cooperação está ilustrada na Figura 4. O nó que obteve a melhor colocação, considerando as três dimensões, foi o referente ao papel de Testador 1, que está destacado, assim como suas relações com os outros nós da rede.

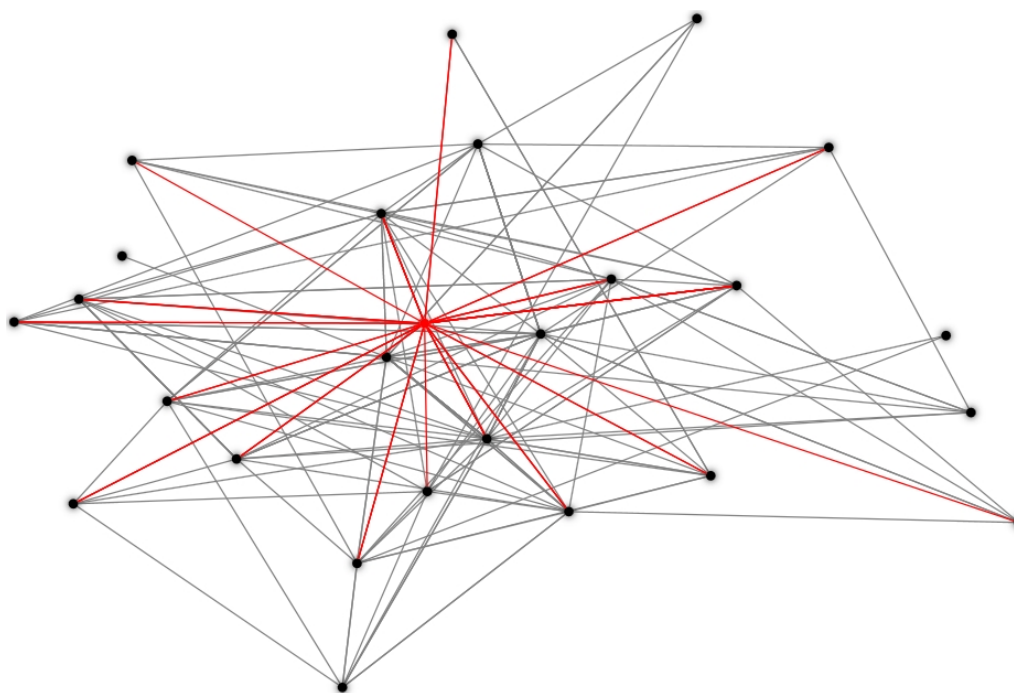


Figura 4. Rede gerada para os dados da Cooperação

Semelhante à análise da coordenação, os dados também estão separados por métrica aplicada, conforme ilustra a Figura 5.

Centralidade de Intermediação		Centralidade de Proximidade		Centralidade de Grau	
Papel	Score	Papel	Score	Papel	Score
Programador 2	39.18	Testador 1	0.030	Testador 1	17
Analista de Qualidade	38.90	Analista de Qualidade	0.029	Analista de Qualidade	17
Testador 1	32.46	Programador 3	0.029	Programador 3	15
Programador 3	24.17	Testador 6	0.028	Testador 6	14
Designer	14.70	Programador 2	0.027	Programador 2	13
Testador 6	12.13	Arquiteto de Software 2	0.027	Arquiteto de Software 2	13
Testador 5	11.54	Gerente de Projeto 2	0.026	Gerente de Projeto 2	12
Arquiteto de Software 2	10.17	Designer	0.026	Designer	11
Gerente de Projeto 2	10.10	Testador 2	0.025	Testador 2	10
Arquiteto de Software 3	8.16	Testador 5	0.025	Testador 5	10

Figura 5. Resultados das métricas para a Cooperação

Os papéis de destaque aqui ficaram para aqueles que diretamente estiveram envolvidos com o código. Os Programadores e os Testadores ocuparam uma boa parte das

posições na tabela. Isso se deve às contribuições e revisões de código inerentes aos seus papéis. Outro papel de destaque foi o de Analista de Qualidade, que com a função de garantir a qualidade do código, teve participação ativa na distribuição de artefatos, como padrões de projeto e relatórios de inspeção de qualidade.

Aqui, também é possível notar que alguns alunos tiveram um engajamento maior do que outros mesmo sendo do mesmo papel. Isso pode ser ilustrado pelos Testadores e pelos Programadores, que mantiveram uma certa ordem entre os alunos que estiveram à frente dos outros que partilhavam das mesmas funções. Esses resultados podem expressar um diferente nível de engajamento entre os indivíduos de mesmo papel, mas também podem revelar outros aspectos, como uma possível divisão de tarefas entre alunos com a mesma responsabilidade.

6. Conclusão

Este trabalho apresentou os resultados das análises feitas sobre um projeto de desenvolvimento de software no contexto acadêmico. Os resultados obtidos foram satisfatórios, visto que o objetivo de quantificar a participação individual dos alunos, relacionando com os papéis assumidos, foi alcançada. A aplicação de métricas de centralidade sobre as redes mostrou o panorama de atuação dos alunos, deixando clara a participação maior de Gerentes de Projeto e Analistas de Qualidade em tarefas de Coordenação, assim como o maior envolvimento de Programadores e Testadores nas atividades de Cooperação. Além disso, foi possível observar a diferença de engajamento entre alunos que possuíam as mesmas responsabilidades.

Destaca-se a importância destes resultados como forma de apoiar a atribuição de notas em trabalhos de grupo por parte dos professores e coordenadores. Além disso, a análise permite que a metodologia do projeto seja aprimorada, podendo influenciar na divisão de papéis ou na escolha das ferramentas de apoio.

Como trabalhos futuros, aponta-se a análise dos dados de outras edições do projeto como uma forma de comparar, ao longo do tempo, a evolução da metodologia usada. Outro ponto é interpretar os resultados com base na divisão de grupos, como mulheres e homens ou alunos de diferentes períodos, em busca de características, além dos papéis, que possam influenciar na participação dos alunos.

Referências

- Balieiro, M. A., Souza Jr, S., Pereira, L. P., and de Souza, C. (2007). Ossnetwork: Um ambiente para estudo de comunidades de software livre usando redes sociais. In *Experimental Software Engineering Latin America Workshop*, pages 33–424.
- Borba, E. M. (2013). Medidas de centralidade em grafos e aplicações em redes de dados.
- Braga e Silva, G., Barbosa, D. M., and Silva, F. A. (2018). An interdisciplinary approach to software engineering teaching: An experience report. In *Anais do 26º Workshop sobre Educação em Computação - WEI 2018*, pages 201–210. SBC.
- Curtis, B., Krasner, H., and Iscoe, N. (1988). A field study of the software design process for large systems. *Communications of the ACM*, 31(11):1268–1287.
- da Fontoura, V. D., Wiese, I., Steinmacher, I., Ré, R., da Silva, J. T., and Gerosa, M. A. (2013). Uma abordagem para classificação de construção de software utilizando re-

- des de comunicação e cooperação. In *WMSWM 2013-X Workshop de Manutenção de Software Moderna*, pages 1–8.
- Dabbish, L., Stuart, C., Tsay, J., and Herbsleb, J. (2012). Social coding in github: transparency and collaboration in an open software repository. In *Proceedings of the ACM 2012 conference on computer supported cooperative work*, pages 1277–1286. ACM.
- Damiani, M. F. (2008). Entendendo o trabalho colaborativo em educação e revelando seus benefícios. *Educar em revista*, 24(31):213–230.
- Ellis, C. A., Gibbs, S. J., and Rein, G. (1991). Groupware: some issues and experiences. *Communications of the ACM*, 34(1):39–58.
- Freeman, L. C. (1978). Centrality in social networks conceptual clarification. *Social networks*, 1(3):215–239.
- Fuks, H., Raposo, A. B., Gerosa, M. A., and Lucena, C. J. P. (2003). Do modelo de colaboração 3c à engenharia de groupware. *Simpósio Brasileiro de Sistemas Multimídia e Web-Webmidia*, pages 0–8.
- Fulford, C. P. and Zhang, S. (1993). Predicting student satisfaction from perceptions of interaction in distance learning. In *Proceedings of the IFIP TC3 Third Teleteaching Conference*, pages 259–268. North-Holland Publishing Co.
- Gatti, B. A. (2003). O professor e a avaliação em sala de aula. *Estudos em avaliação educacional*, (27):97–114.
- Gerosa, M. A. (2006). Desenvolvimento de groupware componentizado com base no modelo 3c de colaboração. *Rio de Janeiro*.
- Jiang, J., Zhang, L., and Li, L. (2013). Understanding project dissemination on a social coding site. In *Reverse Engineering (WCRE), 2013 20th Working Conference on*, pages 132–141. IEEE.
- Rocha, L. M., Silva, T. H. P., and Moro, M. M. (2016). Análise da contribuição ao código entre repositórios do github.
- Thung, F., Bissyande, T. F., Lo, D., and Jiang, L. (2013). Network structure of social coding in github. In *Software maintenance and reengineering (csmr), 2013 17th european conference on*, pages 323–326. IEEE.
- Tutte, W. T. (1963). How to draw a graph. *Proceedings of the London Mathematical Society*, 3(1):743–767.
- Wiese, I. S., Kuroda, R. T., Ré, R., Bulhões, R., Oliva, G., and Gerosa, M. (2015). Do historical metrics and developers communication aid to predict change couplings? *IEEE Latin America Transactions*, 13(6):1979–1988.