

Ensino de Processo de Software em Colaboração com Empresas: Uma Experiência Potencialmente Extensionista

Andréa Sabedra Bordin
Universidade Federal de Santa Catarina
Araranguá, Santa Catarina, Brazil
andrea.bordin@ufsc.br

RESUMO

O ensino de Engenharia de Software se torna mais efetivo quando realizado por meio de experiências práticas reais. Propiciar esse tipo de experiência demanda articular colaborações com empresas e pode se caracterizar como uma prática extensionista a ser utilizada para fins de creditação curricular. Este artigo relata uma experiência de ensino de processo de software onde os estudantes, por meio de uma metodologia ativa de aprendizagem, modelaram processos de desenvolvimento de empresas desenvolvedoras de software e identificaram oportunidades de melhoria. O trabalho de modelagem e análise de processos de desenvolvimento de software desenvolvido pelos grupos de estudantes produziu resultados práticos que são compartilhados neste artigo. A experiência foi avaliada em relação aos aspectos positivos e desafiadores pelos alunos e pela professora. Na percepção dos estudantes a experiência foi positiva porque propiciou a oportunidade de entrar em contato com processos de desenvolvimentos em cenários reais. Na visão da professora, a experiência auxiliou na formação dos estudantes, assim como estreitou as relações com as empresas de desenvolvimento da região, aumentando as possibilidades de ações extensionistas curricularizadas envolvendo essa comunidade.

CCS CONCEPTS

• **Social and professional topics** → Computing education.

PALAVRAS-CHAVE

Educação em Engenharia de Software, Colaboração com Empresas, Extensão Universitária

1 INTRODUÇÃO

Referenciais curriculares nacionais e internacionais de cursos da área de Computação fazem menção explícita a necessidades de formação que preparem os estudantes para o mercado de trabalho [16, 24]. Especificamente em *Computing Curricula 2020* [10] é mencionado que a cooperação com a indústria e o governo pode servir como um veículo para desenvolver as competências dos alunos em seus cursos.

Particularmente, no Ensino de Engenharia de Software (EES), há um consenso que propiciar experiências práticas de aprendizagem

aos estudantes resulta em profissionais mais preparados para o mercado de trabalho [7, 15, 17]. Abordagens de aprendizagem ativa como a Aprendizagem baseada em Projetos (*Project-based Learning - PjBL*) e outras são comuns no EES, porque permitem aos alunos vivenciarem experiências mais próximas aos desafios encontrados na indústria, possibilitando a aquisição de competências técnicas e comportamentais (*hard and soft skills*) valorizadas no mercado de trabalho [8].

Planejar experiências de EES práticas e reais frequentemente demanda colaboração dos professores com empresas da indústria de software. Foram encontrados alguns relatos de experiências de colaboração [19, 20, 22]. Ståhl et al. [22] apresentam um modelo da relação aluno-universidade-*stakeholder* no EES baseado em projetos como um ecossistema, e apresenta um conjunto de fatores que contribuem para manter esse ecossistema produtivo e sustentável. Contudo, a configuração de experiências de ensino que abordem os conhecimentos e habilidades requeridos por essa indústria de software é complexa e envolve tempo e esforço consideráveis, sendo, portanto, um desafio para professores [15, 19]. Além disso, observa-se que a participação efetiva dos *stakeholders* da indústria de software em experiências de EES ainda é bastante limitada [8, 11].

A despeito do cenário de dificuldades reportadas na literatura, entende-se que a colaboração com empresas deva ser estimulada, pois além de propiciar experiências ricas de aprendizagem, podem ser utilizadas para fins de creditação curricular de horas de extensão, em atendimento à Resolução CNE Nº 7 de 18 de dezembro de 2018 (RN 7). A RN 7 institui as diretrizes da Extensão Universitária no país, determinando que 10% da carga horária de cursos de graduação devam ser realizadas por meio de ações de extensão [5]. Nesta perspectiva, a colaboração com empresas pode acontecer por meio de programas e projetos de extensão, cursos e eventos, assim como prestação de serviços, todos esses tipos de ações de extensão previstos na RN 7 e na Política Nacional de Extensão [4].

A curricularização da extensão é uma necessidade de todos os cursos de graduação no país. Um mapeamento recente das iniciativas de curricularização da extensão em cursos de Computação, revelou a existência de poucos relatos na literatura, assim como poucos documentos de Projeto Pedagógico de Curso (PPC) indexados por ferramentas de busca [2].

Este artigo relata uma experiência prática de ensino de processo de software em uma disciplina de Engenharia de Software II, desenvolvida em colaboração com empresas da indústria de software. A experiência foi conduzida por meio de uma metodologia de aprendizagem ativa, com os estudantes divididos em grupos, interagindo com representantes das empresas para entender, modelar e analisar o processo de desenvolvimento atual de empresas, assim como

Fica permitido ao(s) autor(es) ou a terceiros a reprodução ou distribuição, em parte ou no todo, do material extraído dessa obra, de forma verbatim, adaptada ou remixada, bem como a criação ou produção a partir do conteúdo dessa obra, para fins não comerciais, desde que sejam atribuídos os devidos créditos à criação original, sob os termos da licença CC BY-NC 4.0.

EduComp'24, Abril 22-27, 2024, São Paulo, São Paulo, Brasil (On-line)

© 2024 Copyright mantido pelo(s) autor(es). Direitos de publicação licenciados à Sociedade Brasileira de Computação (SBC).

identificar oportunidades de melhoria. São apresentados o planejamento, os principais resultados obtidos pelos alunos, a avaliação da experiência na visão dos alunos e da professora e, por fim, as discussões sobre os aspectos extensionistas da experiência, que demonstram seu potencial para ser creditada como atividade extensionista curricularizada.

As contribuições principais deste trabalho são: a) a descrição detalhada do planejamento da experiência, que pode ser reproduzida em contextos onde o tópico de processo de software é ensinado; b) os principais benefícios e desafios encontrados na configuração e execução de uma experiência de ensino envolvendo a colaboração de atores externos da indústria de software, assim como sugestões para dirimi-los; c) uma discussão acerca do potencial que esse tipo de experiência de ensino tem para fins de creditação de horas de extensão em atendimento à resolução que institui a curricularização da extensão.

O restante deste artigo está organizado da seguinte forma: na Seção 2, é apresentada uma breve fundamentação teórica sobre processo de software e curricularização da extensão; Na Seção 3 são apresentadas experiências de ensino de processo de software encontradas na literatura; Na Seção 4, o planejamento e a execução da experiência, assim como os principais resultados obtidos pelos grupos são descritos; na Seção 5 os grupos de alunos e a professora da disciplina avaliam a experiência de ensino; e, por fim, as considerações finais são expostas na Seção 6.

2 FUNDAMENTAÇÃO TEÓRICA

Nesta seção são apresentados conceitos relacionados à experiência relatada neste artigo, como processo de software e curricularização da extensão.

2.1 Processo de Software

É comum se encontrar na literatura o uso, algumas vezes indiscriminado, dos termos processo de software, modelo de processo e modelo de ciclo de vida. Em *Software Engineering Body of Knowledge (SWEBOK)* [3], processo de software é definido como um conjunto de atividades e tarefas inter-relacionadas que transformam entradas de produto de trabalho em saídas de produto de trabalho.

Processos de software são especificados por vários motivos: para facilitar a compreensão, comunicação e coordenação humana; auxiliar no gerenciamento de projetos de software; para medir e melhorar a qualidade dos produtos de software de uma maneira eficiente; para apoiar a melhoria dos processos; e fornecer uma base para o suporte automatizado da execução do processo. Não existe um processo de software melhor, ou seja, processos devem ser selecionados, adaptados e aplicados conforme cada projeto e cada contexto organizacional [3].

Um ciclo de vida de desenvolvimento de software (*Software Development Life Cycle - SDLC*) inclui os processos de software usados para especificar e transformar os requisitos de software em um produto de software entregável. Segundo o *SWEBOK*, processos de software individuais não possuem ordenação temporal entre eles. As relações temporais entre os processos de software são fornecidas por um modelo de ciclo de vida de software [3].

Modelos de ciclo de vida ou modelos de processo de software podem ser lineares, com as fases/processos de software sendo executados sequencialmente, como o modelo Cascata (*Waterfall*); iterativos, nos quais o software é desenvolvido em incrementos de ciclo iterativo, como o Modelo Espiral; e ágeis, que envolvem entregas frequentes de software funcionando em ciclos iterativos curtos.

O sucesso de um projeto de software é altamente influenciado pela seleção de um modelo de processo de software [23]. De acordo com Kuhlmann et al [12] modelos de processo de software são a “cola” que mantém organizações, projetos e pessoas juntos. Portanto, o desenvolvimento, a manutenção e a melhoria de um modelo de processo de software constituem tarefas desafiadoras que requerem engenheiros de processo bem treinados e experientes.

2.2 Curricularização da Extensão

A Extensão Universitária é definida na Política Nacional de Extensão do Fórum de Pró-Reitores de Extensão das Universidades Públicas Brasileiras (FORPROEX)[4] como “um processo interdisciplinar, educativo, cultural, científico e político que promove a interação transformadora entre Universidade e outros setores da sociedade”.

São diretrizes da extensão: a) a interação dialógica, que orienta o desenvolvimento de relações de mão-dupla entre universidade e setores sociais; b) a interdisciplinariedade e interprofissionalidade, que direciona para o emprego de várias áreas do conhecimento e pela construção de alianças interprofissionais nos objetivos das ações; c) a indissociabilidade Ensino-Pesquisa-Extensão, que indica que as ações de extensão adquirem maior efetividade se estiverem vinculadas ao processo de formação de pessoas (Ensino) e de geração de conhecimento (Pesquisa); d) o impacto na formação do estudante, que permite o enriquecimento da experiência discente em termos teóricos e metodológicos, ao mesmo tempo em que abrem espaços para a materialização dos compromissos éticos e solidários da Universidade Pública brasileira; e) o impacto e transformação social, que prevê uma atuação voltada para os interesses e necessidades da maioria da população e propiciadora do desenvolvimento social e regional [4].

Mais recentemente, como uma estratégia que visa o cumprimento da Meta 12.7 do Plano Nacional de Educação (PNE) instituído pela Lei nº 13.005/2014, a extensão foi inserida como parte da estrutura curricular dos cursos de graduação do país. A Resolução nº 7, de 18 de Dezembro de 2018, do Conselho Nacional de Educação (CNE) institui as diretrizes para a Extensão na Educação Superior Brasileira, as quais compreendem os princípios, os fundamentos e os procedimentos que devem ser observados no planejamento, nas políticas, na gestão e na avaliação das Instituições de Educação Superior (IES) do país [5]. O Art. 4º da referida Resolução estabelece que: “As atividades de extensão devem compor, no mínimo, 10% do total da carga horária curricular dos cursos de graduação, as quais deverão fazer parte da matriz curricular dos cursos”. Dessa forma, todos os cursos de graduação devem adequar seus Projetos Pedagógico de Curso (PPC) para contemplar atividades de extensão agora curricularizadas.

Cursos de graduação que ofereçam disciplinas de Engenharia de Software (ES) tem um grande potencial para realizar atividades extensionistas em diálogo (colaboração) com a comunidade. Na

literatura foram encontrados alguns relatos de experiência de Ensino de Engenharia de Software associados à ações extensionistas, assim como Projetos Pedagógico de Curso (PPC) com estratégias de curricularização de extensão com foco na ES.

Lopes et al. [14] relatam o desenvolvimento de um app para controle de protocolo de uma prefeitura na disciplina de Resolução de Problemas V de um curso de ES. Nesta disciplina o objetivo de aprendizagem era desenvolver conhecimentos e habilidades práticas no uso de um processo ágil de desenvolvimento. A ideia do desenvolvimento do app surgiu de um mapeamento de necessidades da comunidade realizado no escopo de um programa de extensão já em andamento na instituição de ensino. O desenvolvimento contou a colaboração de servidores da prefeitura, que informaram os requisitos e participaram da avaliação do app. Lima et al. [9] descrevem um trabalho de levantamento e especificação de requisitos de um sistema de gerenciamento de biblioteca escolar para escolas públicas do município, na disciplina de Resolução de Problemas I de um curso de ES. Cada grupo de alunos escolheu uma escola para levantar os requisitos e desenvolver o documento de especificação de requisitos de acordo com as necessidades da escola. No caso deste último relato, consta que a carga horária da disciplina RP I é creditada como horas de extensão.

O mapeamento de estratégias de curricularização da extensão em cursos da área de Computação de Bordin [2] identificou apenas três (3) PPC onde ações de extensão com foco em ES são curricularizadas em disciplinas: dois (2) deles são de Bacharelados em Engenharia de Software e o outro é de um Bacharelado em Sistemas de Informação.

3 ENSINO DE PROCESSO DE SOFTWARE

O ensino de processo de software é determinado pelos principais referenciais curriculares de cursos de graduação da área de Engenharia de Software [16, 24]. De acordo com Bezerra et al. [1] muitos currículos não dedicam esforço para ensinar processos de software, focando mais nos princípios básicos da Engenharia de Software, como requisitos, arquitetura e linguagens de programação. Para os autores o ensino de processo de software torna-se desafiador, principalmente devido à grande ênfase na teoria e poucas práticas.

Nesta seção, são apresentadas algumas experiências de ensino nesta temática encontradas na literatura [1, 6, 12, 13, 18, 21, 23].

Bezerra e Coutinho [1] afirmam que um aspecto importante da aprendizagem de processo de software é a modelagem, pois esta fornece uma base para gerenciar, automatizar e apoiar a melhoria do processo de software. Os autores utilizaram uma prática de modelagem de processos de desenvolvimento em diversos domínios de aplicação, tais como: IoT, nuvem, mobile, sistemas críticos, sistemas auto-adaptáveis e jogos, argumentando que há necessidade de soluções específicas para processos de desenvolvimento específicos de domínio. Os processos foram modelados na ferramenta Eclipse Process Framework (EPF) Composer. A prática de modelagem de processos foi avaliada e chegou-se a conclusão de que ferramentas de modelagem e a maturidade no domínio são essenciais para o bom desempenho do processo.

Tiwari e Rathore [23] destacam que o conhecimento básico de ES deve ser suplementado com aulas focadas em modelos de processo de software. Os autores propuseram o uso de um recurso lúdico (peça teatral) para representar cenários de aplicação de modelos de

processo no mundo real. Cada grupo de estudantes escolheu um modelo de processo para desenvolver seu projeto e teve que preparar uma peça/esquete para explicar como o modelo de processo escolhido os ajudou a compreender, obter requisitos e desenvolver sua aplicação. A peça teatral objetivou criar um ambiente para demonstrar como os modelos de processos funcionam em cenários do mundo real. Os feedbacks dos alunos indicaram que o exercício os ajudou a compreender os processos pelos quais devem passar durante o desenvolvimento de software.

Moura e Santos [18] argumentam que normas e modelos de maturidade, como a ISO/IEC 29110, CMMI e outras podem ser de grande auxílio para definição de processos, mas que seu conteúdo é extenso e de difícil absorção. Dessa forma, apresentaram um jogo de tabuleiro educativo (ProcSoft) com o objetivo de ensinar conceitos, definição, estrutura e conteúdo de um processo de software de forma descontraída. Os alunos aprendem a compor um processo por meio de um jogo sobre as fases do ciclo de vida de um software. Os resultados positivos mostraram o envolvimento dos participantes em sala de aula e a contribuição para a aprendizagem.

Kuhrmann et al. [12] afirmam que tópicos de processo de software estão situados numa camada organizacional e não são ensinados na universidade de forma que os alunos possam experimentar os efeitos de processos de software ausentes ou mal implementados. Eles propuseram uma estrutura de disciplina com objetivos, conteúdos e práticas, cujo objetivo é ensinar os conceitos básicos de modelagem de processos de software, análise de processos, a implementação com suporte de ferramentas e avaliação. A proposta em questão enfoca as habilidades que um engenheiro de processo deve ter na indústria de software, indo além de um formato clássico de ensino com palestras e exercícios, com a adoção de *workshops* práticos.

Landes et al. [13] relatam que devido à falta de experiência dos alunos com grandes projetos, a compreensão dos benefícios e deficiências de determinados modelos de processo é muito limitada, o que torna difícil ensinar as especificidades desses modelos por meio de uma abordagem de ensino tradicional. Os autores relatam a utilização de duas estratégias de ensino adotadas em instituições diferentes: uma baseada na resolução de problemas (*Problem-based Learning - Pbl*) com o desenvolvimento de um jogo de tabuleiro para aprendizagem do V-Model; e outra com enfoque prático, onde os alunos trabalharam em grupo, em um projeto de desenvolvimento de software. Nesta última estratégia cada grupo adotou um modelo de processo diferente. A experiência mostrou que essa estratégia permite que os alunos obtenham uma compreensão muito melhor das características dos modelos de processos.

Srinivasan e Lundqvist [21] projetaram um jogo de simulação de processos de software cujo objetivo é simular um projeto de desenvolvimento de software aeroespacial. O projeto conta com sessenta requisitos gerados para refletir a arquitetura real do sistema. O jogo é jogado por uma equipe de projeto, composta por cinco subequipes cobrindo os requisitos, design, implementação, integração e aspectos de financiamento do ciclo de vida de desenvolvimento do software. Como o jogo não envolve atividades reais de programação ou design, ele pode ser usado de forma eficaz para ensinar engenheiros de software iniciantes e experientes.

Carrington et al. [6] descrevem a utilização do jogo de cartas *Problems and programmers (PnP)* na disciplina de Processo de Software para simular processos de engenharia de software desde a especificação de requisitos até a entrega do produto.

Percebe-se, nesta revisão da literatura sobre ensino de processo de software, que a maioria das experiências adotam abordagens baseadas em jogos e simulação de situações reais, não se aproximando portanto, da vivência real em modelagem e análise de processo de software relatada neste artigo.

4 PLANEJAMENTO E EXECUÇÃO DA EXPERIÊNCIA

A disciplina de Engenharia de Software II do Curso de Graduação em Tecnologias da Informação e Comunicação da Universidade Federal de Santa Catarina - Campus Araranguá conta com três unidades de ensino em seu programa: Unidade Processo de Software, Unidade Evolução de Software e Unidade Qualidade de Software. A disciplina possui quatro (4) créditos e é ministrada em dois (2) encontros semanais de duas (2) horas-aula. A experiência de ensino relatada neste artigo abordou a Unidade Processo de Software (UPS) e ocorreu no semestre 2022/02.

A experiência foi planejada de forma a contemplar um número de aulas expositivas e um número de aulas dedicadas ao desenvolvimento, acompanhamento e avaliação do projeto proposto. A Figura 1 ilustra o processo de configuração desta experiência de ensino.

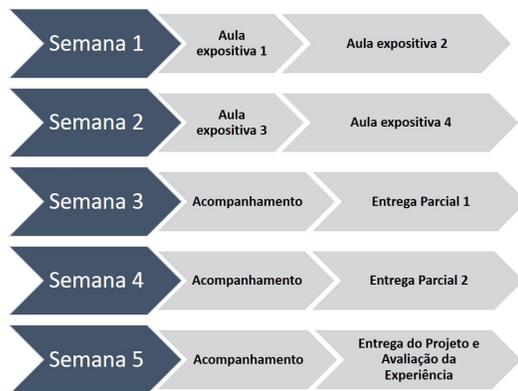


Figura 1: Processo de Planejamento e Execução

Nas duas primeiras semanas foram ministradas aulas expositivas dialogadas, totalizando 8 horas-aula (4 encontros de 2 h-a), com o objetivo de introduzir os principais tópicos acerca da unidade estudada. Nas aulas foram abordados os seguintes tópicos: **fundamentos de processo de software, modelos de processos, principais modelos prescritivos e ágeis (com ênfase em Extreme Programming - XP e Scrum).**

As três últimas semanas foram dedicadas ao desenvolvimento, acompanhamento e avaliação do projeto proposto. Foram estabelecidas duas (2) entregas parciais e, ao final da quinta semana, a entrega e apresentação final. Os requisitos do projeto foram compartilhados com os alunos ao final do último encontro de aula expositiva (Semana 2). Para o desenvolvimento do projeto, os alunos

1. Identificar o modelo de processo de desenvolvimento de uma empresa do segmento de desenvolvimento de produto de software.
2. Realizar a modelagem do processo atual (AS-IS).
3. Identificar junto a equipe oportunidades de melhoria do processo atual;
4. Modelar o processo proposto (TO-BE).

Tabela 1: Requisitos do Projeto da Unidade Processo de Software

Perfil da Empresa: nome da empresa, segmento de desenvolvimento, número de produtos de software em desenvolvimento e em manutenção.
Perfil da Equipe de Desenvolvimento: número de pessoas, funções de cada membro da equipe.
Características do processo de desenvolvimento atual: 1) Existe um processo de desenvolvimento bem estabelecido? Se sim, qual é o modelo de processo de desenvolvimento (tradicional, ágil)? 2) Qual é o fluxo do processo de desenvolvimento (quais são as atividades do processo de desenvolvimento, quem são as pessoas envolvidas, quais os artefatos gerados?).
Modelagem atual (AS-IS) do processo de desenvolvimento em <i>Business Process Modeling Notation (BPMN)</i> ;
Análise preliminar do processo atual de desenvolvimento.

Tabela 2: Requisitos da Entrega Parcial 1

organizaram-se por afinidade em cinco grupos de cinco integrantes (25 alunos).

Os requisitos foram planejados com o intuito de propiciar uma experiência de contato real com um processo de desenvolvimento de uma empresa de software, de forma que os alunos pudessem identificar e analisar de forma crítica, a partir dos conhecimentos teóricos e conversas com os *stakeholders* da empresa, o modelo de processo vigente e possíveis pontos de melhoria do processo de software. A Tabela 1 mostra os requisitos do projeto solicitado.

As Semanas 3, 4 e 5 foram dedicadas ao desenvolvimento do trabalho. No primeiro encontro de cada semana ocorreu o acompanhamento e orientação dos grupos. No segundo encontro de cada semana, os grupos apresentaram as entregas parciais e foram avaliados individualmente. Ao final da quinta semana apresentaram o resultado final do trabalho e realizaram a avaliação da experiência de ensino.

Foram estabelecidos os requisitos de cada entrega parcial. Para a **Entrega Parcial 1** foi requerido que os grupos entregassem um documento contendo os itens expostos na Tabela 2. Já para a **Entrega Parcial 2** com os itens constantes na Tabela 3.

Sobre o **método de avaliação da aprendizagem**, cada aluno foi avaliado individualmente pela professora durante as entregas parciais e apresentação do projeto final, assim como se auto-avaliou e foi avaliado também pelos seus pares no grupo, correspondendo a 50% da nota. O restante da nota foi dada pela avaliação do entregável final (projeto final) realizado pelo grupo. A auto-avaliação e

Identificação das oportunidades de melhoria do processo de desenvolvimento, justificando as práticas e fluxo propostos.
Modelagem preliminar da primeira versão do processo de desenvolvimento com as oportunidades de melhoria (TO-BE), também em BPMN.

Tabela 3: Requisitos da Entregar Parcial 2

Tabela 4: Perfil das Empresas

	Número de produtos	Perfil da equipe	
		Quant.	Função
Empresa A	3	2 3	Des. Front end Elicitação e Programação
Empresa B	4	5 2 3	Requisitos Qualidade Desenvolvimento
Empresa C	7	3 1 1 1 1 1	Desenvolvimento Scrum Master Team Manager PO interno PO externo e Arquiteto de Soluções DevOps
Empresa D	2	2 5 4	Gestor Des. Front end Des. Back end
Empresa E	3	13 3 4 3 3 1	Desenvolvimento Teste PO Tech Lead Analista Agile Master

a avaliação dos pares, conhecida como Avaliação 360°, foi realizada por meio de um formulário eletrônico e teve o objetivo de engajar o aluno, tornando-o co-responsável pelo processo de avaliação do seu grupo.

4.1 Resultados dos Projetos

Nesta seção são apresentados de forma sintetizada e discutidos brevemente os principais resultados do trabalho de análise e modelagem de processo de desenvolvimento de software em uma empresa do segmento de desenvolvimento de software, que cada grupo desenvolveu com a colaboração de um ou mais *stakeholders* da empresa.

Na Tabela 4 as empresas escolhidas por cada grupo estão nomeadas com A, B, C, D e E, para preservar a sua anonimidade. Pode-se observar também o número de produtos de software em manutenção ou desenvolvimento, assim como a quantidade de funcionários e as principais funções ou área onde trabalham.

Na Tabela 5 estão sintetizadas informações como o modelo de processo atual das empresas e as oportunidades de melhorias identificadas por cada grupo, conforme solicitado na Entrega Parcial 1. Essas oportunidades derivaram-se da identificação de problemas relacionados ao fluxo, tarefas, pessoas e ferramentas envolvidas nos processos analisados. Esses problemas foram ora apontados pelos próprios *stakeholders*, ora identificados pelo grupo.

Entende-se que está fora do escopo deste relato de experiência de ensino analisar em profundidade os resultados obtidos com o trabalho de modelagem. No entanto, de forma breve, dois pontos cabem ser mencionados: 1) A Empresa C, com mais produtos em desenvolvimento ou manutenção é aquela que possui a equipe desenvolvimento com funções mais estruturadas e condizentes ao modelo ágil de processo informado. Nesta empresa o fluxo de processo atual modelado já indica fases e atividades bem definidas de acordo com o modelo ágil. Dessa forma, não é surpresa que o grupo não tenha identificado muitas oportunidades de melhoria; 2) A proposição de uma atividade de especificação/documentação de requisitos, na proposição de melhorias de processo da maioria das empresas analisadas, confirmando, portanto, a realidade de falta de documentação já bem conhecida da indústria de software.

5 AVALIAÇÃO DA EXPERIÊNCIA

Nesta seção são apresentadas as percepções dos estudantes e da professora em relação à experiência de ensino.

5.1 Percepção dos Estudantes

A cada grupo foi solicitada uma única avaliação consolidada dos aspectos positivos e desafiadores da experiência de ensino. Foi aplicada a técnica de codificação aberta para análise das respostas qualitativas dos cinco (5) grupos. Os resultados conduziram a duas categorias de percepção: a relacionada com os requisitos de projeto em si e a relacionada com aspectos da metodologia de aprendizagem ativa, como o trabalho em grupo e a autonomia no processo de aprendizagem.

5.1.1 Aspectos Positivos e Desafiadores do Projeto Proposto. A Tabela 6 apresenta os aspectos positivos e desafiadores identificados nas respostas dos grupos, conjuntamente com a frequência com que apareceram. Percebe-se que os aspectos relacionados ao conhecimento da realidade da empresa se sobressaem em relação ao demais aspectos, evidenciando o impacto que experiências de colaboração reais com empresas tem nos alunos.

Em relação aos aspectos desafiadores, a maioria dos grupos apontou o desconhecimento da notação BPMN como um desafio. Cabe mencionar que o tópico de modelagem de processo (notações e ferramentas) não foi abordado nas aulas expositivas, justamente para que os alunos fossem desafiados a adquirir um novo conhecimento, uma vez que essa é uma premissa das abordagens de aprendizagem ativas.

5.1.2 Aspectos Positivos e Desafiadores da Metodologia de Aprendizagem Ativa. O trabalho em grupo, a autonomia nos processos de busca do conhecimento são característicos da Aprendizagem Baseada em Projetos, assim como em muitas metodologias ativas de aprendizagem. Pode-se observar na Tabela 7 que o trabalho em grupo é apontado como um aspecto desafiador, aparecendo com

Tabela 5: Oportunidades de Melhoria no Processo das Empresas

	Modelo de processo	Oportunidades de Melhoria
Empresa A	Ágil (Scrum e Kanban)	-Elicitar e documentar requisitos -Agrupar o conjunto de tarefas em um backlog da sprint -Utilizar ferramentas de gerenciamento
Empresa B	Cascata	-Analisar o impacto do pedido de manutenção -Documentar requisitos -Realizar reuniões diárias para melhorar a comunicação
Empresa C	Ágil (Lean Kanban)	-Incluir Desenvolvedor Líder na fase de Discovery
Empresa D	Ágil (Scrum) Cascata	-Transformar a função de Gerente de Projetos em Product Owner (requisitos) -Criar a função do Scrum Master e Tech Lead -Documentar requisitos -Substituir o método de distribuição de tarefas pela escolha das tarefas pela equipe -Implantar um ambiente de homologação mais semelhante ao de produção -Unificar os documentos de produção
Empresa E	Ágil (Scrum)	-Adicionar um Agile Masters para cada time Dev -Promover melhor comunicação entre os setores de Dev e Produto -Criar critérios de análise de viabilidade de demandas fora da sprint -Promover a programação em pares -Permitir a escolha da tarefa da Sprint por nível de conhecimento

Tabela 6: Aspectos Positivos e Desafiadores do Projeto Proposto

Aspectos Positivos	Freq.
Contato próximo com fluxo ágil de desenvolvimento	2
Contato com ferramentas e notação de modelagem de processos utilizadas no mercado	2
Sugerir melhorias no processo de desenvolvimento de software.	2
Identificar os principais aspectos das metodologias ágeis em ambientes corporativos.	1
Conhecer o dia a dia e vida real das empresas da comunidade	2
Aumentar a importância social da universidade em <Araranguá - SC.>	1
Aspectos Desafiadores	Freq.
Disponibilidade da empresa para conversar com a equipe	1
Somente um membro do grupo conhecer o processo da empresa	1
Não conhecer a notação BPMN	4

mais frequência, por meio de descrições diferentes. Dentre os aspectos positivos, o protagonismo do aluno foi mencionado pela maioria dos grupos.

5.2 Percepção da Professora

A percepção da professora sobre os aspectos positivos (+) e desafiadores (-) da experiência de ensino também foi construída à luz das categorias encontradas nas percepções dos estudantes, assim como dialoga com os resultados dessas percepções.

Tabela 7: Aspectos Positivos e Desafiadores da Metodologia de Aprendizagem

Aspectos Positivos	Freq.
Trabalho envolvente	1
Comunicação fluida entre membros do grupo	2
Proatividade	1
Organização e divisão de tarefas	1
O aluno participa ativamente do processo tornando o conhecimento mais consolidado	4
O acompanhamento da professora durante as aulas foi muito bom	1
O conhecimento foi adquirido através das interações	1
Fortalece as habilidades pessoais.	1
Aspectos Desafiadores	Freq.
Dificuldade de se reunir devido aos diferentes horários dos membros	1
Balanceamento da distribuição das tarefas.	1
Fazer as atividades com antecedência para evitar o Go Horse.	1
Falta de preparo para a apresentação do trabalho	1

5.2.1 Aspectos Positivos e Desafiadores do Projeto Proposto. +Sobre os objetivos de aprendizagem: Os requisitos do projeto permitiram aos alunos conhecerem como é um fluxo de processo de desenvolvimento real de uma empresa, propiciaram o desenvolvimento de análise crítica a partir dos conhecimentos teóricos obtidos nas primeiras semanas de aula e a aquisição de conhecimentos novos relacionados à técnicas e ferramentas de modelagem de processos. Os objetivos de aprendizagem foram atingidos de forma bastante

satisfatória no tempo planejado (2 semanas para aulas expositivas e 3 semanas para o desenvolvimento do projeto).

-Sobre o fator tempo: Neste tipo de experiência de EES, envolvendo atores externos, há desafios para todos os atores envolvidos. Os representantes das empresas não costumam dispor de tempo de qualidade para atender os alunos, devido a alta demanda dessa indústria e a falta de percepção dos benefícios desse tipo de colaboração; os alunos, especialmente aqueles do turno noturno e que trabalham (a grande maioria), não possuem muitas opções de horário nos turnos diurnos para interagir com os atores externos; o professor da disciplina, especialmente aquelas com uma ementa mais abrangente de tópicos de ES, como as nomeadas Engenharia de Software II e afins, precisam ficar atentos ao número de semanas que vão dedicar à experiência, uma vez que outros tópicos de ensino precisam ser abordados. Nesta experiência em específico, percebe-se que um prazo adicional de uma a duas semanas permitiria uma análise mais aprofundada das oportunidades de melhoria do processo modelado, assim como o compartilhamento dos resultados com os *stakeholders* das empresas e o consequente recebimento de *feedback* desses *stakeholders*. A configuração de um projeto único, envolva diversos unidades/tópicos de ensino e em colaboração com empresas é um desafio ainda maior.

-Sobre o estabelecimento da colaboração com empresas: Nesta experiência a escolha da empresa foi feita pelos grupos, que por uma questão de conveniência, consideraram melhor escolher uma empresa onde um dos membros já trabalhava. No entanto, em um cenário onde os membros de um grupo não conseguissem a confirmação de colaboração de alguma empresa, o contato da empresa deveria ser fornecido pela professora da disciplina. Entende-se que a responsabilidade e o desafio de contatar e esperar o aceite das empresas, em última instância, é do(a) docente da disciplina.

Sobre os desafios percebidos pelos estudantes: A maioria dos grupos mencionou o desconhecimento da técnica de modelagem de processo sugerida (BPMN), uma vez que ela não foi apresentada nas aulas teóricas. Entende-se que a aquisição de conhecimentos novos para o desenvolvimento do trabalho faz parte de desafios que os alunos devem enfrentar, especialmente em contexto onde alguma metodologia ativa é empregada.

5.2.2 Aspectos Positivos e Desafiadores da Metodologia de Aprendizagem Ativa. A metodologia ativa baseada em projeto utilizada nesta experiência foi bastante adequada aos propósitos de aprendizagem de processo de software. Por meio dela, percebeu-se que os alunos ficaram mais motivados, comparativamente às aulas expositivas.

O acompanhamento semanal dos grupos, em períodos iterativos ocorridos em cada aula da semana, propiciou momentos imersivos de discussão com os grupos. Nestes momentos foi possível perceber que a metodologia despertou as habilidades de autonomia, cooperação e comunicação dos alunos.

Um desafio comum, compartilhado por docentes que adotam metodologias ativas com alunos trabalhando em grupo é a avaliação do desempenho individual. A técnica de avaliação 360° empregada, onde todos os membros do grupo se avaliam, se mostrou efetiva, na medida em que mostrou percepções semelhantes à da professora. Dessa forma, o conceito/nota final individual foi atribuído levando

em consideração a percepção de trabalho empregado dos demais membros do grupo.

5.2.3 Potencial Extensionista da Experiência de Ensino. Entende-se que a experiência atendeu a várias diretrizes estabelecidas na RN7, mais especificamente no seu Art. 5: promoveu uma interação dialógica da comunidade acadêmica com um setor específico da sociedade por meio da troca de conhecimentos, da participação e do contato com demandas desse setor; propiciou vivências que complementaram a formação cidadã dos estudantes (como apontado nas percepções positivas de um grupo); possibilitou a articulação entre ensino-extensão-pesquisa. Dessa forma, possui todos os elementos para ser creditada como atividade de extensão curricularizada.

5.3 Sugestões para os Desafios Percebidos

Sobre os desafios da colaboração com empresas, entende-se que primeiramente é preciso que haja uma espécie de movimento no sentido da universidade para as empresas divulgando os benefícios dessa colaboração. A indústria de software necessita perceber de forma mais concreta o valor do investimento de tempo na colaboração com o ensino (e com a pesquisa também) da universidade. É sabido o quanto essa indústria investe de tempo e recursos para treinar profissionais na execução das atividades que necessita. Ao enxergar valor na colaboração com professores e estudantes, parte desse investimento pode ser compartilhado com a universidade, gerando benefícios para ambos.

Para os alunos e professores de cursos de graduação da universidade, um grande estímulo para que essa colaboração aconteça pode ser a curricularização da extensão, que demanda que seja ofertada uma carga horária expressiva em ações de extensão. Especialmente para cursos noturnos, a estratégia de curricularização mais adequada é a que insere as atividades de extensão nas disciplinas do currículo. Dessa forma, podem ser desenvolvidos projetos que abordem conjuntos de conhecimentos específicos da ES, como por exemplo, qualidade de software, teste de software, desenvolvimento de software, na solução de problemas/demandas tanto da própria indústria de software como de qualquer entidade civil ou pública da sociedade.

Para os alunos, além dos conhecimentos e habilidades que podem ser adquiridos em experiências práticas oriundas da colaboração com empresas, existe também a rede de contatos que é estabelecida e pode proporcionar outras oportunidades para os alunos, como estágios e empregos. A ampliação da rede de contatos é uma vantagem que deve ser percebida pelas empresas também, uma vez que existe uma grande demanda por profissionais nessa indústria.

6 CONSIDERAÇÕES FINAIS

A experiência relatada neste artigo aborda o ensino de processo de software por meio de uma metodologia de aprendizagem ativa, onde os alunos em colaboração com empresas da indústria de software, analisaram modelos e fluxos de processo de desenvolvimento vigentes e propuseram melhorias. Na percepção dos alunos e da professora, tanto os requisitos do projeto, que envolveu a colaboração de *stakeholders* de empresas, quanto a metodologia de aprendizagem utilizada no estudo deste tópico de engenharia de software, se mostraram adequados e propiciaram a aquisição de conhecimentos

e habilidades importantes. Os desafios e as sugestões para dirimi-los também foram reportados e discutidos.

Entende-se que a experiência possui elementos que a caracterizam com uma potencial ação de extensão curricularizada. Ela dialoga com a sociedade, neste caso um setor específico - o de software; ela auxilia na formação dos estudantes; e ela promove a indissociabilidade entre ensino e extensão, estando, portando em consonância com as principais diretrizes da extensão universitária. Em um contexto onde os cursos devem se adequar à Resolução Normativa nº 7, experiências como esta podem ser o caminho para a creditação de horas de extensão por meio de disciplinas do currículo, uma das opções previstas na referida Resolução.

A obrigatoriedade da creditação curricular de atividades de extensão pode ser estímulo para uma colaboração com empresas a longo prazo. No entanto, é preciso haver um esforço coordenado de vários atores na universidade (pró-reitores, coordenadores de curso, professores, etc), para o estabelecimento de estratégias de institucionalização das colaborações e, mais importante, de sustentabilidade a longo prazo.

Cabe destacar que esta experiência de ensino e extensão na disciplina de Engenharia de Software II ainda não está curricularizada no curso de graduação onde ela é ofertada. Contudo, como já mencionado, a experiência possui características e evidências que a habilitam para uma creditação de horas de extensão.

6.1 Trabalhos Futuros

Como trabalhos futuros pretende-se aplicar e divulgar os resultados dessa experiência ao longo de vários semestres. Como uma forma de fomentar experiências de ensino reais em colaboração com a indústria de software, com potencial extensionista, pretende-se realizar um mapeamento sistemático das iniciativas de cooperação entre a universidade (especificamente no que tange ao ensino de ES) e a indústria de software, a fim de verificar possibilidades e desafios já encontrados. Outra proposta consiste em realizar um survey com representantes da indústria de software regional para levantar necessidades e possibilidades de colaboração.

REFERÊNCIAS

- [1] Carla Bezerra and Emanuel Coutinho. 2022. Teaching software processes from different application domains. In *Anais do XXI Simpósio Brasileiro de Qualidade de Software* (Curitiba/PR). SBC, Porto Alegre, RS, Brasil, 277–286. <https://sol.sbc.org.br/index.php/sbqs/article/view/23314>
- [2] Andrea Bordin. 2023. Uma Análise da Curricularização da Extensão na Graduação em Computação: Possibilidades e Desafios. In *Anais do III Simpósio Brasileiro de Educação em Computação* (Evento Online). SBC, Porto Alegre, RS, Brasil, 262–269.
- [3] Pierre Bourque, Richard E. Fairley, and IEEE Computer Society. 2014. *Guide to the Software Engineering Body of Knowledge (SWEBOOK(R)): Version 3.0* (3rd ed.). IEEE Computer Society Press, Washington, DC, USA.
- [4] Brasil. 2012. Política Nacional de Extensão Universitária. <https://www.ufmg.br/proex/renex/index.php/documentos/documentos>
- [5] Brasil. 2018. Resolução CNE/CES Nº 7, de 18 de dezembro de 2018. Estabelece as Diretrizes para a Extensão na Educação Superior Brasileira. https://normativasconselhos.mec.gov.br/normativa/view/CNE_RES_CNECESN72018.pdf
- [6] D. Carrington, A. Baker, and A. van der Hoek. 2005. It's All in the Game: Teaching Software Process Concepts. In *Proceedings Frontiers in Education 35th Annual Conference*. F4G–F4G.
- [7] Jianguo Chen, Huijuan Lu, Lixin An, and Yongxia Zhou. 2009. Exploring teaching methods in software engineering education. In *2009 4th International Conference on Computer Science Education*. 1733–1738.
- [8] Orges Cico, Letizia Jaccheri, Anh Nguyen-Duc, and He Zhang. 2021. Exploring the intersection between software industry and Software Engineering education - A systematic mapping of Software Engineering Trends. *Journal of Systems and Software* 172, 110736.
- [9] Henrique Hoff de Lima, Guilherme Samuel S. da Silva, Matheus Kildere C. de Vasconcelos, Maicon Bernardino, and Andrea Sabedra Bordin. 2019. Engenharia de Requisitos na Resolução de Problemas da Comunidade: Lições Aprendidas. In *Anais da III Escola Regional de Engenharia de Software* (Rio do Sul). SBC, Porto Alegre, RS, Brasil, 105–112. <https://sol.sbc.org.br/index.php/eres/article/view/8502>
- [10] CC2020 Task Force. 2020. *Computing Curricula 2020: Paradigms for Global Computing Education*. Association for Computing Machinery, New York, NY, USA.
- [11] Vahid Garousi, Kai Petersen, and Baris Ozkan. 2016. Challenges and best practices in industry-academia collaborations in software engineering: A systematic literature review. *Information and Software Technology* 79, 106–127.
- [12] Marco Kuhrmann, Daniel Méndez Fernandez, and Jürgen Münch. 2013. Teaching software process modeling. In *2013 35th International Conference on Software Engineering (ICSE)*. 1138–1147.
- [13] Dieter Landes, Yvonne Sedelmaier, Volkhard Pfeiffer, Jürgen Mottok, and Georg Hagel. 2012. Learning and teaching software process models. In *Proceedings of the 2012 IEEE Global Engineering Education Conference (EDUCON)*. 1–8.
- [14] Jonnathan Lopes, Gabriela Medeiros, Dienefer Fialho, and Andréa Bordin. 2017. Resolução de Problemas no Curso de Engenharia de Software: Uma Experiência Envolvendo Extensão e Ensino. In *Anais da I Escola Regional de Engenharia de Software* (Alegrete). SBC, Porto Alegre, RS, Brasil, 97–104. <https://sol.sbc.org.br/index.php/eres/article/view/10086>
- [15] Maira R. Marques, Alcides Quispe, and Sergio F. Ochoa. 2014. A systematic mapping study on practical approaches to teaching software engineering. In *2014 IEEE Frontiers in Education Conference (FIE) Proceedings*. 1–8.
- [16] Ministério da Educação. 2016. Resolução Nº 5, de 16 de novembro de 2016. Institui as Diretrizes Curriculares Nacionais para os cursos de graduação na área da Computação. Retrieved May 11, 2018 from <http://portal.mec.gov.br/>
- [17] Melody Moore and Colin Potts. 1994. Learning by doing: Goals and experiences of two software engineering project courses. In *Software Engineering Education*, Jorge L. Díaz-Herrera (Ed.). Springer Berlin Heidelberg, Berlin, Heidelberg, 151–164.
- [18] Victor Moura and Gleison Santos. 2018. ProcSoft: A Board Game to Teach Software Processes Based on ISO/IEC 29110 Standard. In *Proceedings of the XVII Brazilian Symposium on Software Quality* (Curitiba, Brazil) (SBQS '18). Association for Computing Machinery, New York, NY, USA, 363–372.
- [19] Maria Paasivaara, Jari Vanhanen, and Casper Lassenius. 2019. Collaborating with Industrial Customers in a Capstone Project Course: The Customers' Perspective. In *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET)*. 12–22.
- [20] Maria Spichkova. 2019. Industry-Oriented Project-Based Learning of Software Engineering. In *2019 24th International Conference on Engineering of Complex Computer Systems (ICECCS)*. 51–60.
- [21] Jayakanth Srinivasan and Kristina Lundqvist. 2007. A Constructivist Approach to Teaching Software Processes. In *Proceedings of the 29th International Conference on Software Engineering (ICSE '07)*. IEEE Computer Society, USA, 664–672.
- [22] Daniel Ståhl, Kristian Sandahl, and Lena Buffoni. 2022. An Eco-System Approach to Project-Based Learning in Software Engineering Education. *IEEE Transactions on Education* 65, 4, 514–523.
- [23] Saurabh Tiwari and Santosh Singh Rathore. 2019. Teaching Software Process Models to Software Engineering Students: An Exploratory Study. In *2019 26th Asia-Pacific Software Engineering Conference (APSEC)*. 308–315.
- [24] A. F. Zorzo, D. Nunes, E. Matos, J. Steinmacher, Land Leite, R. Araujo, R. M. and Correia, and S. Martins. 2017. Referenciais de Formação para os Cursos de Graduação em Computação. 153.