

# Avaliação da Eficácia do Uso Pedagógico de Projetos OSS

Erinaldo Santos Oliveira<sup>1</sup>, Christina von Flach<sup>2</sup>

<sup>1</sup>Instituto Federal Baiano

<sup>2</sup>Instituto de Computação, Universidade Federal da Bahia

erinaldo.oliveira@ifbaiano.edu.br, flach@ufba.br

**Abstract.** *One of the challenges of using Open Source Software (OSS) projects in Education is evaluating their effectiveness in achieving the pedagogical objectives of a discipline. This paper presents the E3S conceptual model for evaluating the effectiveness of OSS projects in Teaching of Engenharia de Software (ES). We built the E3S model through document analysis of curricular references from the SBC and international societies and thirty papers from the literature on using OSS projects in Education. We applied the model to evaluate using the JabRef project in a Software Testing course.*

**Resumo.** *Um dos desafios do uso de projetos Software de Código Aberto (OSS) na Educação é a avaliação de sua eficácia para alcançar os objetivos pedagógicos de uma disciplina. Este artigo apresenta o modelo conceitual E3S para avaliação da eficácia de projetos OSS no Ensino de ES. Construímos o modelo por meio de análise documental sobre referenciais curriculares da SBC e de sociedades internacionais, e de trinta artigos da literatura sobre uso de projetos OSS na Educação. Aplicamos o modelo para avaliar o uso do projeto JabRef em uma disciplina de Teste de Software.*

## 1. Introdução

O uso de projetos de Código Aberto (OSS)<sup>1</sup> permite que professores trabalhem conteúdos da disciplina de Engenharia de Software (ES) em projetos realistas, trazendo uma experiência de mundo real próxima das exigências do mundo do trabalho. Vários trabalhos reportam que os benefícios trazidos pelo uso de OSS na Educação em Engenharia de Software (EES) superam os desafios [Holmes et al. 2018, Pinto et al. 2017], e apontam para a necessidade de apoiar a seleção e avaliação de projetos OSS [Ellis et al. 2012, Brito et al. 2018] para facilitar sua adoção por um conjunto amplo de educadores.

Alguns trabalhos apontam caminhos para a construção de um processo avaliativo que seja contínuo e coloque os alunos no centro da aprendizagem. Um método ou técnica de avaliação deve apoiar o professor na análise da adequação de um projeto OSS com respeito a critérios organizados na forma de um modelo de avaliação. Entretanto, *não* encontramos na literatura um modelo de avaliação com esse propósito. Em nossa pesquisa buscamos responder à seguinte questão: *Como o professor pode avaliar a eficácia de um projeto OSS, escolhido como recurso de ensino e aprendizagem, em relação ao objetivo pedagógico proposto pela atividade ou disciplina de engenharia de software?*

<sup>1</sup>Ao longo do texto, usaremos a sigla OSS para designar *Free/Libre Open Source Software*.

Este artigo apresenta o modelo E3S para avaliação da *Eficácia* de projetos OSS no *Ensino de ES*, construído por meio de análise documental sobre diversas fontes de informação e da técnica de Análise de Conteúdo [Câmara 2013]. O desenho da pesquisa inclui duas atividades para embasar a construção do modelo: o *mapeamento de aspectos de influência* nos processos de ensino e aprendizagem relacionados ao uso de projetos OSS no ensino de ES (Seção 2) e o *mapeamento de competências técnicas ou sociotécnicas* esperadas de estudantes de cursos superiores em Computação (Seção 3). A Seção 4 apresenta o modelo E3S, um modelo conceitual para a avaliação da *Eficácia* de projetos OSS no *Ensino de ES*. A Seção 5 apresenta o modelo E3S em uso para avaliar o uso do projeto *JabRef* em uma disciplina de Teste de Software e discute alguns resultados. A Seção 6 apresenta as conclusões.

## 2. Aspectos de Influência no ensino de ES com OSS

Projetos OSS possuem características interessantes para uso na EES. Seu processo de desenvolvimento distribuído, a complexidade das soluções em desenvolvimento, a dinâmica das comunidades e a acessibilidade dos projetos para um amplo público interessado em contribuir com a comunidade, sugerem que projetos OSS podem ser considerados um laboratório dinâmico que traz para o meio acadêmico a rotina de atividades complexas do mundo profissional de desenvolvimento de software [Stamelos 2011, Nascimento 2017].

A necessidade de formação superior em Computação condizente com a realidade da indústria e do mundo do trabalho destaca o potencial dos projetos OSS na EES. Docentes podem considerar projetos OSS como uma ferramenta para alcançar objetivos educacionais. Porém, como em qualquer prática educativa, os objetivos podem não ser alcançados por diferentes razões. Questões discentes, acadêmicas e institucionais podem interferir no processo. Além disso, recursos pedagógicos ou objetos de ensino podem não ser condizentes com a proposta educacional delineada [Nascimento 2017]. Neste artigo, estas questões são tratadas como *Aspectos de Influência (AI)* (ou de interferência) nos processos de ensino e aprendizagem mediados pelo uso de projetos OSS.

A estratégia escolhida para identificar e compreender os AIs foi a análise de artigos ou relatos de experiência de professores publicados, buscando compreender seus elementos temáticos, benefícios, problemas e limitações. Utilizamos um mapeamento sistemático da literatura (SMS) sobre o uso de projetos OSS na EES [Brito et al. 2018]: seus autores disponibilizaram os artefatos de pesquisa, incluindo o protocolo do SMS, planilhas e os estudos primários mapeados.

### 2.1. Análise de Conteúdo dos Estudos Primários

Após a análise do protocolo do SMS, confirmamos que o propósito de nosso mapeamento estava alinhado com critérios temáticos do SMS. Processos de busca e seleção, palavras-chave, bases bibliográficas, critérios de inclusão e exclusão, estavam coerentes com o escopo de nossa pesquisa, representando uma amostra pertinente [BARDIN 2011].

O SMS selecionou trinta e três (33) estudos primários que formaram o *corpus* inicial de nossa análise. Utilizamos fichas de **Unidades de Contexto (UC)** e **Unidades de Registro (UR)** com a mesma identificação do fichamento do SMS de origem: *Código, Autores e Título*. A análise de conteúdo foi realizada a partir da leitura flutuante dos estudos primários (e eventual consulta às planilhas do SMS), considerando resumos,

fundamentação, desenvolvimento e resultados dos trabalhos. A definição de critérios para seleção de AIs deu-se na identificação das temáticas próximas aos elementos (*índices*) citados pelos autores dos estudos primários. As **URs** extraídas a partir das **UCs** dos textos, referem-se a percepções diretas dos próprios autores do estudo primário, e indiretas (citações ou resultados de outras pesquisas mencionadas no estudo primário), sobre interferência em resultados de aprendizagem ou influência em questões pedagógicas, didáticas, técnicas ou comportamentais. Dentre os 33 estudos analisados, 30 artigos apresentaram **URs** referentes a AIs do uso de projetos OSS no ensino de ES. Observamos **URs** frequentes sobre temáticas relacionadas aos aspectos citados nos 30 artigos e, com base da frequência de cada tema recorrente, identificamos os *eixos temáticos*. A análise e organização das **URs** do artigos resultaram na extração de *quatrocentos e vinte e sete (427) URs*. Cada **UR** refere-se a *um tema sobre aspecto de influência técnica ou pedagógica no ensino de ES usando OSS*.

## 2.2. Aspectos de Influência

Dezenove aspectos de influência na aprendizagem foram mapeados a partir das **URs**: (A1) Tamanho e complexidade do projeto, (A2) Interesse, motivação e atração, (A3) Adaptação ao currículo ou curso, (A4) Nível de formação e experiência do estudante, (A5) Tempo, Calendário ou Cronograma, (A6) Uso ou conhecimento de ferramentas e ambientes de desenvolvimento, (A7) Suporte, acompanhamento e avaliação do estudante, (A8) Capacidade e Habilidade de Programação, (A9) Suporte ou viabilidade acadêmica, (A10) Percepção ou Autopercepção das Habilidades de Computação, (A11) Dinâmica, rotina e cultura de Projetos OSS, (A12) Relação com a comunidade, Receptividade do Projeto, (A13) Restrições, Viabilidade e Adequação do Projeto, (A14) Estado atual do projeto, (A15) Assunto ou Domínio do Projeto, (A16) Estratégia para atividades e participação de estudantes, (A17) Curva de Aprendizagem, (A18) Questões sociais, pessoais e legais, (A19) Desgaste, Entusiasmo e Experiência do Docente.

A relação entre os 30 estudos primários analisados e os aspectos de influência identificados (A1–A19) está disponível no Material Suplementar deste artigo<sup>2</sup>.

Ao analisarmos apenas o quantitativo de **URs** comuns a cada tema, observamos que o AI **A2** (*Interesse, motivação e atração*) foi o mais citado, com 11,9% de citações, seguido por **A16** (*Estratégia para atividades e participações dos estudantes*) com 9,84% de citações e por **A6** (*Uso e conhecimento de ferramentas e ambientes de desenvolvimento*), com 9,37% das citações. Os aspectos menos frequentes foram **A9** (*Suporte ou Viabilidade acadêmica*), **A18** (Questões sociais, pessoais e legais), e **A13** (Restrições, viabilidade acadêmica e adequação do projeto) – todos com menos de 10 citações. A análise quantitativa dos 19 AIs, tanto em relação ao número de **URs** (427) quanto em relação aos 30 estudos primários, mostrou a sua relevância no ensino de ES mediado por projetos OSS. Dada a sua relevância, consideramos que *cada AI mapeado é um eixo temático retirado das URs mapeadas*. Todos os eixos temáticos estão alinhados às características temáticas levantadas anteriormente [Nascimento 2017, Stamelos 2011], ou seja, são *aspectos que podem exercer influência nos processos de ensino e aprendizagem na Educação em Engenharia de Software mediada pelo uso de projetos OSS*. A categoria de análise que resume e define toda a perspectiva temática e teórica deste trabalho é a categoria **Aspecto de Influência**.

<sup>2</sup>O material suplementar está disponível em <https://zenodo.org/record/8003776>.

### 3. Competências na Engenharia de Software

O mapeamento de competências e habilidades técnicas ou sócio-técnicas considerou as versões públicas disponíveis dos Referenciais da SBC e Guias da série ACM Curricula que seguem o Currículo de Computação 2005 [Shackelford et al. 2006]<sup>3</sup>.

**Mapeamento das Competências da ES segundo os Referenciais da SBC.** Este mapeamento de competências da ES teve como objetivo principal identificar competências técnicas e sociotécnicas associadas a ES segundo os principais Referenciais de Formação Superior em Computação no Brasil [Zorzo et al. 2017] e incluiu: Análise de cada Referencial de Formação (RF), buscando *conteúdos de ES* e identificação de competências derivadas às quais estes conteúdos estão relacionados; Mapeamento de Competências Gerais (CG) segundo as DCNs estimuladas por tais conteúdos; Mapeamento de Competências Específicas (CE) nas DCNs estimuladas por tais conteúdos nos principais Referenciais de Formação da SBC [Zorzo et al. 2017]; e Mapeamento de Competências específicas ao Graduado em ES (CE-ES).

Ao mapear as Competências relacionadas a ES nos principais Referenciais da SBC, percebeu-se a relação entre *conteúdos específicos de ES e conteúdos sociotécnicos* e as *competências esperadas de egressos dos cursos superiores em Computação* segundo as DCNs. As doze Competências Gerais (CG) das DCNs foram incluídas no modelo E3S [CNE-MEC 2016]. A análise dos Referenciais [Zorzo et al. 2017] mostrou como cada RF relaciona conteúdos de Engenharia de Software e CGs: o RF de Ciência da Computação relaciona conteúdos de ES a cinco CGs e o RF de Sistemas de Informação a três CGs. No caso dos RFs de Engenharia da Computação, Licenciatura e Engenharia de Software, cada RF relaciona conteúdos de ES a seis CGs. O mapeamento de competências identificou a categoria de análise *Competências*, que resume e define as perspectivas temática e teórica dos eixos temáticos extraídos dos Referenciais de Formação que compõem o modelo E3S.

**Mapeamento das Competências da ES segundo Guias internacionais.** Este mapeamento analisou os principais Currículos da ACM desenvolvidos sobre a base do Currículo de Computação de 2005 [Shackelford et al. 2006] (sobre o qual estão baseados os RFs brasileiros), dentre eles, o Currículo de Engenharia de Software de 2014 [Ardis et al. 2015], o Currículo de Sistemas de Informação de 2010 [Topi et al. 2010] e o Currículo de Computação de 2013, para identificar a relevância dos conteúdos de ES e conteúdos sociotécnicos na formação dos estudantes dos principais Cursos Superiores em Computação internacionais. Neste artigo, não apresentaremos os resultados desse mapeamento.

### 4. O Modelo de Avaliação E3S

Através de um diálogo teórico e interpretação, o *Modelo Conceitual E3S* para avaliar a eficácia do uso de Projetos OSS no ensino de ES foi alicerçado. Nesta seção, relacionamos às questões sobre de *Aspectos de Influência e Competências em ES*, os *Objetivos de Aprendizagem* e apresentamos o modelo E3S completo.

---

<sup>3</sup>Em 2020, a ACM publicou um novo Currículo de Computação.

#### 4.1. Objetivos de aprendizagem

Na perspectiva de Bloom [Ferraz et al. 2010], conhecimentos, habilidades e atitudes podem ser concebidas como competências reveladas em níveis cognitivos ou afetivos. No contexto de formação superior em Computação e na formação técnica ou sociotécnica em ES, as competências podem ser relacionadas com objetivos de aprendizagem que devem ser almejados pelas práticas educativas.

*Objetivos de Aprendizagem (OA)* determinam as competências a serem desenvolvidas através das práticas educativas. Sobre isso [Nascimento 2017, p.33] versa que os Objetivos de Aprendizagem “[...] devem descrever o conhecimento ou as habilidades que os estudantes devem apresentar como resultado da atividade”. Em um mundo globalizado e em constante evolução tecnológica, o ensino baseado em competências demanda esforço do docente para usar técnicas e métodos que aproximem o aluno da realidade do mundo do trabalho. Nesse contexto, *ferramentas de ensino e objetos de aprendizagem* são recursos valiosos e têm o poder de levar o estudante de uma realidade de prática educacional a um contexto próximo ao que o professor desejaria.

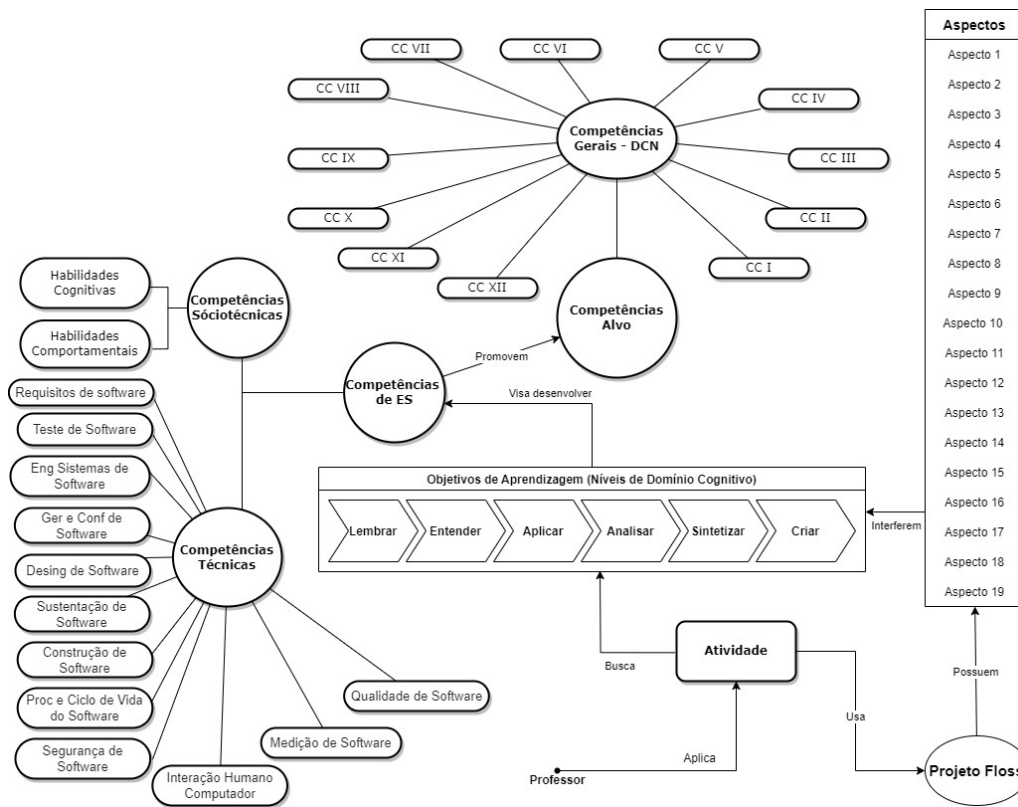
*Objetos de Aprendizagem* são ferramentas de aprendizagem e instrução, usados no ensino de diversos conteúdos e revisão de conceitos [Aguiar and Flôres 2014]. Alguns aspectos devem ser observados em um Objeto, dentre eles, qualidade do conteúdo, aprofundamento na temática, se é contextualizado e se facilita a assimilação de conteúdos [Almeida et al. 2015]. Quando o professor usa um projeto OSS para apoiar o ensino de ES, esse projeto assume características de um objeto de aprendizagem, evoluindo de um recurso tecnológico para um instrumento pedagógico na perspectiva docente. Dessa forma, o professor deve se atentar aos limites do que é técnico e do que é pedagógico na forma de avaliar um projeto OSS a ser usado para ensinar. Considerando que o processo de desenvolvimento de competências dá-se nos objetivos de aprendizagem [Zorzo et al. 2017], pode-se dizer que um recurso educacional usado para atingir tais objetivos interfere no alcance de competências planejadas pelo docente [Aguiar and Flôres 2014, Mussoi et al. 2010]. Sendo o projeto OSS um recurso educacional no contexto da ES, este passa a ter *influência nas competências de ES planejadas pelo professor*.

#### 4.2. O Modelo Conceitual E3S

A Figura 1 ilustra o *Modelo Conceitual E3S* completo. Objetivos de Aprendizagem (organizados segundo a Taxonomia de Bloom) visam desenvolver Competências de ES (técnicas e sóciotécnicas) que promovem *Competências Alvo*, que por sua vez, relacionam Competências Gerais e Específicas das DCNs. Atividades que usam Projetos OSS buscam alcançar Objetivos de Aprendizagem, e Projetos OSS possuem Aspectos que interferem no alcance de Objetivos de Aprendizagem.

### 5. Avaliação do Modelo E3S

O primeiro autor do artigo utilizou o modelo E3S para guiar uma avaliação da eficácia de um projeto OSS escolhido para uso em atividades de uma disciplina de Teste de Software [Nascimento 2017]. Devido ao espaço limitado, os resultados completos da validação do modelo estão disponíveis no material suplementar desta pesquisa e são discutidos parcialmente neste artigo.



**Figura 1. Representação Visual do Modelo Conceitual E3S.**  
 Fonte: Elaboração Própria

### 5.1. Visão Geral do Estudo de Caso

**Objetivo.** O estudo de caso teve como objetivo explorar e investigar consequências e objetivos da aprendizagem mediadas por uso de projetos OSS, conexões entre teoria e prática, a percepção da experiência de mundo real e tratamento de lacunas de formação identificadas na indústria [Nascimento 2017].

**Contexto.** Aplicação de um projeto OSS no ensino de Teste de Software em uma disciplina de um Curso de Graduação em Ciência da Computação.

**Cenário.** O estudo foi realizado em atividades da disciplina Teste de Software, do Bacharelado em Ciência da Computação da Universidade Federal de Sergipe. Aulas expositivas e exercícios obrigatórios, foram complementados por atividades práticas **opcionais** no projeto OSS (notas das provas do estudante poderiam ser substituídas por notas obtidas nas atividades com o projeto OSS). Na primeira etapa, estudantes **em equipe** implementaram **individualmente** testes de unidade e integração para grupos de funções atribuídos para a equipe. Na segunda etapa, os estudantes implementaram e analisaram testes e prepararam planos de testes.

**População.** A população do estudo foi composta pelo docente e 15 alunos que participaram das duas etapas do projeto. A disciplina é de sétimo período, ou seja, *os alunos estão se formando e, portanto, possuem razoável maturidade acadêmica*. Dos 15 estudantes que participaram efetivamente *sete deles não tinham nenhuma experiência com projetos reais, dois participaram de até dois projetos e seis alunos tinham participado de mais projetos*. Todos os estudantes mais experientes *aceitaram o desafio e participaram*

*do projeto*. A professora, mestre na área de Computação, tem experiência no ensino de programação e ES, e prática como desenvolvedora de software. A professora admitiu usar o método tradicional de ensino. Um monitor (já havia cursado a disciplina) deu suporte aos estudantes [Nascimento 2017].

**Contexto de aplicação da atividade com projeto OSS.** A atividade prática com o OSS permitiu a aplicação de testes automatizados em um projeto real. O projeto usado foi JabRef, um gerenciador de referências bibliográficas, de médio porte, desenvolvido em Java para ambiente *desktop* [Nascimento 2017]. A coleta e análise dos dados provenientes da abordagem foi feita por questionários, entrevistas e análise das entregas dos estudantes.

**Objetivos de Aprendizagem.** Objetivos relacionados a teste de software e a práticas profissionais beneficiadas pela aplicação da abordagem foram selecionados. Para os aspectos técnicos de teste de software, foram selecionados OAs relacionados a capacidade de descrever conceitos e práticas de teste, seus tipos, níveis e técnicas; Entender diferenças de porte de testes, ferramentas de integração e testes; Aplicar testes e ferramentas de testes; Avaliar suíte de testes e métricas de testes; e Criar Testes de Unidades não redundantes, planejar e gerar casos de testes e criar e documentar conjunto de testes. Para os sociotécnicos, foram analisados OAs sobre: Respostas a crítica de terceiros; Avaliar trabalhos de terceiros; Ser proativo e criativo; Solucionar problemas e gerar soluções alternativas; Ter visão holística; e Compreender, sumarizar e escrever materiais técnicos.

**Estrutura para Atividade.** Em geral, as atividades propostas visavam criar e executar testes e plano de testes em um projeto real “elaborar e executar ao menos uma vez um plano de testes de regressão, incluindo os testes implementados pela equipe” [Nascimento 2017, p.144].

**Objetivo de Aprendizagem da Atividade.** O OA analisado na aplicação do modelo E3S é *LO104 - Capacidade de planejar e gerar casos de testes para softwares de médio porte*. Este objetivo demanda o alcance de outros objetivos ou requisitos na área de Testes (conceitos e ferramentas e técnicas necessárias para a execução e documentação dos testes a serem realizados). Como os estudantes têm que saber e saber fazer para o alcance deste objetivo, unindo elementos e gerando algo original, este objetivo foi avaliado baseado na dimensão *Criar* da Taxonomia de Bloom.

**Competências de ES.** A Competência Técnica em Teste de Software, segundo [Ardis et al. 2014], resume competências em planejar testes, infraestrutura de testes, técnicas de teste, medição de testes e rastreamento de defeitos. As competências sociotécnicas associadas foram abordadas como competências da prática profissional [Nascimento 2017].

## **5.2. Aspectos de influência relevantes no uso do projeto JabRef**

Fizemos a avaliação da eficácia do projeto JabRef para a atividade “Elaborar e executar ao menos uma vez um plano de testes de regressão, incluindo os testes implementados pela equipe”. Por restrição de espaço, apresentamos apenas alguns AIs; os demais estão no material suplementar deste artigo.

**A16 Estratégia para atividades e participações dos estudantes.** A professora da disciplina fez uma abordagem teórica e expositiva dos conceitos abordados, oferecendo a participação **opcional** do estudante nas atividades com o JabRef e, “Caso o estudante participasse, as notas de suas provas poderiam ser substituídas pelas notas

obtidas com a realização das atividades”. A participação voluntária pode influenciar alunos mais interessados, atraídos e motivados nas atividades com o projeto, o que pode indicar uma influência no aspecto **(A2) Interesse, motivação e atração**.

**A4 Nível de formação e experiência do estudante.** Na seção 5.1, apresentamos o perfil dos estudantes que realizaram as atividades com o OSS. Sobre isso “Cabe ressaltar que todos os estudantes mais experientes da turma aceitaram o desafio e participaram do projeto.” [Nascimento 2017, p.145]. A escolha do projeto OSS e de atividades com base em uma caracterização prévia do perfil da turma poderá ser mais inclusiva.

**A5 Tempo, Calendário ou Cronograma.** A principal dificuldade encontrada foi adequar atividades com o projeto OSS ao cronograma da disciplina, devido ao seu programa extenso e com muitas atividades de avaliação. Sobre isso [Nascimento 2017, p.147] descreve que “[...] precisamos adiar diversas vezes as entregas periódicas dos produtos das atividades anteriormente planejadas. [...], o tempo para execução das demais atividades tornou-se exíguo, fazendo com que, não exigíssemos nenhuma entrega individual na segunda parte do trabalho e afetando a discussão final.”

**A7 Suporte, acompanhamento e avaliação do estudante.** Um estudante que já havia cursado a disciplina e com bastante experiência profissional foi escolhido para ser assistente nas atividades para dar suporte no uso de ferramentas e na manipulação do código. Isso pode ser positivo por manter os alunos próximos e engajados nas atividades, além de que os alunos podem se sentir mais confortáveis em tirar as dúvidas ou demais comunicação com um colega de curso.

**A8 Capacidade e Habilidade de Programação.** O software *JabRef* foi escolhido por ser “implementado em Java, linguagem que faz parte do currículo do curso e é empregada em outras disciplinas” [Nascimento 2017, p.145]. Os alunos tinham familiaridade com JAVA.

**A15 Assunto ou Domínio do Projeto.** O *JabRef* “representa um sistema de informação, com um domínio fácil de ser compreendido” [Nascimento 2017, p.145], sendo também útil a interesses acadêmicos do estudante.

**A19 Desgaste, Entusiasmo e Experiência do Docente.** Segundo [Nascimento 2017, p.147]: “[...] relatamos a dificuldade específica de utilização de um software (livre) sobre o qual o professor não tem o domínio [...] torna-se incerto que as funcionalidades distribuídas entre as equipes sejam de complexidade semelhante [...]”.

### 5.3. Discussão

É importante destacar que os 15 estudantes *realizaram as atividades propostas* e implementaram testes automatizados para testar o *JabRef*, um software de médio porte, envolvendo complexidade similar a encontrada em software do mundo do trabalho. A avaliação da eficácia do projeto *JabRef* para a atividade “Elaborar e executar ao menos uma vez um plano de testes de regressão, incluindo os testes implementados pela equipe” mostrou que seu uso trouxe benefícios mas ainda esbarrou em diversos fatores limitantes descritos neste trabalho. Observamos que alguns AIs exercem influência em outros aspectos, nas atividades e no alcance dos objetivos de aprendizagem, sendo necessário um esforço do docente para entender e planejar ações lidar com tais influências antes do início das atividades com o OSS.



**A16 Estratégia para atividades e participação de estudantes.** Foi fortemente influenciada pelo pouco tempo disponível: o aspecto Tempo, Calendário ou Cronograma (**A5**) limitou o desenvolvimento da abordagem e obrigou mudanças em outras temáticas, como cobertura dos objetivos, entregas e prazos. A cobertura de ferramentas necessárias para a participação nas atividades definidas pela estratégia teve influência positiva na Curva de Aprendizagem (**A17**) dos alunos nesse tema, mas inibiu a participação de outros estudantes que se assustaram com a carga de trabalho. A escolha do projeto OSS e de atividades com base em caracterização prévia do Nível de Formação e Experiência do Estudante (**A4**) poderá estimular a participação e estender o aspecto Interesse, Motivação e Atração (**A2**) a um número maior de estudantes.

**A5 Tempo, Calendário ou Cronograma.** Foi o AI que mais limitou a abordagem. O tempo curto demandou alterações em **A16**, com mudanças no escopo da abordagem, objetivos cobertos e entrega das atividades pelos estudantes. Isto interferiu nas entregas seguintes, alterando prazos e sobrecarregando os organizadores da atividade.

**A17 Curva de Aprendizagem.** Observou-se uma curva de aprendizagem acentuada, principalmente sobre o Uso de Ferramentas e ambientes de controle de versões (**A6**), que demandou tempo adicional para a entrega das atividades. Porém, mesmo com a curva de aprendizagem acentuada, atividades foram fundamentais para a percepção positiva sobre o desenvolvimento de suas habilidades em computação (**A10**), tanto nas competências técnicas planejadas e no domínio das ferramentas necessárias quanto nas competências sociotécnicas desenvolvidas.

## 6. Conclusões

Apresentamos o modelo conceitual E3S, proposto para avaliação da eficácia do uso de projetos OSS no Ensino de ES. Além das contribuições da pesquisa de [Ellis et al. 2012], não encontramos na literatura trabalhos que apresentem um modelo de avaliação com propósito relacionado ao do E3S. Neste contexto, o modelo apresentado traz contribuições originais que podem estimular o uso de projetos OSS no ensino de ES.

O modelo E3S passou por uma validação inicial com base no relato de uma professora que utilizou o projeto `JabRef` em uma disciplina de Teste de Software. O primeiro autor deste artigo, um professor de ES, pôde utilizar o modelo E3S para avaliar a eficácia do `JabRef`. Resultados preliminares mostram que o uso do `JabRef` em uma atividade da disciplina trouxe benefícios mas ainda esbarrou em diversos fatores limitantes descritos neste trabalho, em especial, relacionados ao aspecto de influência **A5**.

Certamente, os resultados da validação não podem ser generalizados: outros estudos são necessários para avaliação do modelo E3S em diferentes contextos e com outros professores. Finalmente, há várias questões do modelo E3S a serem refinadas em trabalhos futuros.

## Referências

- Aguiar, E. V. B. and Flôres, M. L. P. (2014). Objetos de aprendizagem: conceitos básicos. *Objetos de aprendizagem: teoria e prática. Porto Alegre: Evangraf*, pages 12–28.
- Almeida, R. R., Chaves, A. C. L., and de Araújo Jr, C. F. (2015). Avaliação de objetos de aprendizagem: aspectos a serem considerados neste processo. *Revista Educação & Tecnologia*, 1(13).

- Ardis, M. et al. (2014). The Software Engineering Competency Model (SWECOM). *IEEE Computer Society, Los Alamitos, CA, USA*.
- Ardis, M. et al. (2015). SE 2014: Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering. *Computer*, 48(11):106–109.
- BARDIN, L. (2011). Análise de conteúdo/laurence bardin; tradução luís antero reto, augusto pinheiro. *São Paulo: Edições*, 70.
- Brito, M. et al. (2018). Floss in software engineering education: an update of a systematic mapping study. In *Proc. of the XXXII Brazilian Symposium on Software Engineering*, pages 250–259.
- Câmara, R. (2013). Análise de conteúdo: da teoria à prática em pesquisas sociais aplicadas às organizações. *Gerais: Revista Interinstitucional de Psicologia*, 6(2):179–191.
- CNE-MEC (2016). Resolução CNE/CES nº 5, de 16 de novembro de 2016. *Diário Oficial, Brasília, DF. 17 de Nov.*
- Ellis, H. et al. (2012). An approach for evaluating foss projects for student participation. In *Proceedings of the 43rd ACM technical symposium on Computer Science Education*, pages 415–420. ACM.
- Ferraz, A., Belhot, R. V., et al. (2010). Taxonomia de bloom: revisão teórica e apresentação das adequações do instrumento para definição de objetivos instrucionais. *Gest. Prod., São Carlos*, 17(2):421–431.
- Holmes, R. et al. (2018). Dimensions of Experientialism for Software Engineering Education. In *ICSE-Software Engineering Educ. and Training*, page 31–39. ACM.
- Mussoi, E. M., Flores, M. L. P., and Behar, P. A. (2010). Avaliação de objetos de aprendizagem. In *Congresso Iberoamericano de Informática Educativa, Santiago, Chile. Anais.[Google Scholar]*.
- Nascimento, D. M. C. (2017). *Educação em engenharia de software com a adoção de projetos de código aberto: uma análise detalhada*. PhD thesis, Universidade Federal da Bahia.
- Pinto, G. et al. (2017). Training software engineers using open-source software: the professors’ perspective. In *2017 IEEE 30th Conf. on Software Engineering Education and Training (CSEET)*, pages 117–121.
- Shackelford, R. et al. (2006). Computing curricula 2005: The overview report. *ACM SIGCSE Bulletin*, 38(1):456–457.
- Stamelos, I. G. (2011). Teaching software engineering with free/libre open source projects. *Multi-disciplinary advancement in open source software and processes*, pages 67–84.
- Topi, H. et al. (2010). Curriculum Guidelines for Undergraduate Degree Programs in Information Systems. Technical report, ACM, doi 10.1145/2593310.
- Zorzo, A. et al. (2017). *Referenciais de Formação para os Cursos de Graduação em Computação*. Sociedade Brasileira de Computação (SBC). 153p.