

Uso de Métodos Ágeis e Aprendizagem Baseada em Problema no Ensino de Engenharia de Software: Um Relato de Experiência

Maria Alcimar Costa Meireles¹, Bruno Araújo Bonifácio¹

¹Instituto de Ciências Exatas e Tecnologia (ICET)
Universidade Federal do Amazonas (UFAM)
Caixa Postal 15.064 – 91.501-970 – Itacoatiara – AM – Brazil
{mariameireles797, brunnoboni}@gmail.com

Abstract. *The emergence of new technologies both in hardware and software has significantly changed the how information and communication technologies can be developed. To support this transformation, the teaching of Software Engineering also has evolved, to create new technical skills related to software development process. For this reason, learning institutions need to change the way teach, creating a motivating environment in which it is possible to prepare professionals for the increasingly demanding and dynamic market. This paper aims to report the experience of students in learning the concepts of software engineering, using as pedagogical tool the combination between problem based learning and agile method scrum through of real projects developing mobile application.*

Resumo. *O aparecimento constante de novas tecnologias de software tem alterado a forma como as tecnologias de software são desenvolvidas. Para acompanhar essa transformação o processo de ensino em Engenharia de Software também tem evoluído, para criar novas habilidades aos estudantes nos aspectos técnicos e não técnicos relacionados ao desenvolvimento de sistemas de software. Por essa razão, os centros de ensino têm buscado criar um ambiente motivador, no qual seja possível preparar profissionais para o mercado cada vez mais exigente. O objetivo desse artigo é relatar a experiência de alunos no aprendizado de engenharia de software, utilizando como proposta pedagógica a aprendizagem baseada em problema e o uso de scrum para construção de um projeto real, aplicado para dispositivos móveis.*

1. Introdução

A evolução tecnológica atual, tanto de hardware quanto de software, trouxe importantes mudanças no modo como seus usuários interagem e acessam as informações [Glasmann *et al.* 2010]. Esse cenário contribuiu para o surgimento de novos paradigmas, tanto relacionados ao acesso à informação quanto na forma de aprendizado dos usuários [Bonifácio *et al.* 2015].

Enquanto a ciência, a tecnologia e as artes progredem e criam novos paradigmas e tendências para atender às exigências da sociedade e do mercado, a educação tenta adaptar-se aos resultados desse progresso para descobrir que rumos tomar e quais os

melhores meios de preparar o homem para o futuro [Martins 2007]. Dessa forma, o uso de novas metodologias de ensino tem despertado grande interesse no ambiente educacional, por fornecer a flexibilidade, melhorar a integração e incentivar práticas de aprendizagem a partir de uma perspectiva pedagógica inovadora [Glasemann *et al.* 2010, Marçal *et al.* 2010]. Na área de Tecnologia da Informação (TI), onde aparecimento constante de novas tecnologias acabam influenciando no processo de desenvolvimento tradicional, o ensino também tem evoluído [Prickladinick *et al.* 2009].

Por essa razão, as metodologias de ensino precisam focar nas mudanças tecnológicas, visando à preparação dos estudantes, criando novas habilidades nos aspectos técnicos e não técnicos relacionados ao desenvolvimento de sistemas de software. Assim, os centros de ensino da área de Computação têm buscado criar um ambiente motivador, no qual seja possível preparar os estudantes e profissionais da área de TI para o mercado cada vez mais competitivo. Por isso, várias abordagens de ensino têm focado em problemas demandados pelo mercado, como forma de estimular o pensamento crítico, a habilidade de solução e a aprendizagem de conceitos fundamentais, especificamente da área de Engenharia de Software, onde a produção de novas tecnologias acrescenta novos desafios no processo de desenvolvimento, cada vez mais globalizado [Costa *et al.*, 2010]. A combinação de metodologia baseada em problema, no processo de produção de software pode criar novas formas de aprendizado e com isso atender a demanda do mercado por profissionais mais qualificados.

Este artigo apresenta uma experiência prática no processo de ensino de Engenharia de Software, utilizando como ferramenta metodológica a combinação de metodologia de ensino baseada em problema e o método ágil SCRUM, no processo de produção de aplicativos móveis. O objetivo desse trabalho é relatar a experiência no aprendizado de alunos de graduação, no contexto da disciplina de Engenharia de Software, ministrada na Universidade Federal do Amazonas. Ao descrever os resultados desse estudo, espera-se contribuir para a criação de um ambiente motivador, tornando o processo de aprendizagem mais dinâmico, colaborativo e prazeroso. E com isso, incentivar os centros de ensino a adotar combinações de metodologias de ensino com as aplicadas no mercado de trabalho.

O restante deste trabalho está estruturado da seguinte forma: a Seção 2 apresenta os conceitos e as estratégias utilizadas no processo de ensino em Engenharia de Software. A Seção 3 apresenta o contexto da fábrica de software criada para o ensino disciplina de Engenharia de Software, assim como os procedimentos metodológicos aplicada na disciplina e os aplicativos produzidos. Na Seção 4 são apresentados e discutidos os resultados da análise qualitativa. E por fim, a Seção 5 apresenta as lições aprendidas e considerações finais.

2. Metodologias Ágeis e Abordagem Baseada em Problema no Processo de Ensino de Engenharia de Software

Para acompanhar o avanço tecnológico atual, o processo de produção de tecnologias de software, cada vez mais globalizado, mudou a forma como as soluções são desenvolvidas. Esse cenário acrescentou novos desafios no processo de ensino da disciplina de Engenharia de Software (ES), onde essa nova forma de produzir software precisa se torna elemento fundamental. Isto porque a academia usualmente ensina

conceitos e fundamentos de ES através de disciplinas teóricas, com aulas expositivas e leituras complementares, que contemplam a competência prática mediante o desenvolvimento de um projeto em um curto espaço de tempo [Huang e Distante, 2006].

No entanto, essa metodologia apresenta substancial insatisfação quanto ao nível de preparação dos universitários recém-formados que entram no mercado de trabalho, pois quando buscam oportunidades na indústria, estes profissionais encontram um cenário no qual técnicas e métodos aprendidos são pouco aplicados [Nauman e Uzair, 2007]. Em muitos casos as empresas de software têm que complementar os conhecimentos dos recém-formados com treinamentos e prover habilidades nos aspectos técnicos e não técnicos relacionados ao desenvolvimento de sistemas de software [Thompson, 2007; Hilburn, 2007]. Por essa razão, o ensino de ES, deve ser transformado para refletir a demanda por software mais complexo, no qual seja possível preparar profissionais para o mercado cada vez mais exigente. [Baker *et al.*, 2005].

A partir desse cenário, grande parte das instituições tem adotado estratégias diferentes, por meio de abordagem de Aprendizagem Baseada em Problemas (PBL - do inglês *Problem-Based Learning*). Essa proposta metodológica de ensino e aprendizagem é caracterizada pelo uso de problemas reais para estimular o desenvolvimento de pensamento crítico, a habilidade de solução de tais problemas e a aprendizagem de conceitos fundamentais da área de conhecimento em questão [Bezerra *et al.* 2014]. Por meio da combinação de PBL e métodos ágeis é possível criar um ambiente mais dinâmico, despertando maior interesse e motivação por parte dos alunos. Isto tem motivado várias pesquisas. Andrade *et al.* (2010) realizaram um estudo de caso baseado no uso do PBL para ensino de ES no curso de Engenharia da Computação na Universidade Estadual de Feira de Santana (UEFS). O estudo apresenta uma visão focada baseada em responsabilidades com papéis definidos pelos grupos formados na disciplina. Os resultados mostram que a motivação e a criação de novas habilidades podem ser potencializadas por meio do uso de PBL. No entanto, a utilização do processo de desenvolvimento tradicional pode trazer maior dificuldade de solução de problemas em cada etapa do processo, principalmente na fase de análise das necessidades do cliente.

Outra contribuição para o uso de PBL está na compreensão de conteúdos e temas da engenharia de *software* e no impacto direto que a abordagem tem na melhoria do processo de desenvolvimento de *software*. Por essa razão, estudos podem aumentar a experiência de ensino usando PBL. Ronnie *et al.* (2014) apresentam um trabalho com intuito de identificar, analisar e discutir as ferramentas, métodos e experiências de ensino desta disciplina, realizando um mapeamento sistemático. Os resultados deste estudo mostram que os métodos de ensino estão ligados a propostas didáticas e métodos de trabalho sem necessariamente uso de *software* educativo. Por isso, é de grande importância o trabalho em equipe, onde haja um contato real entre os alunos, propício à cooperação e discussões a respeito do desenvolvimento de atividades.

Nesse contexto, o uso de metodologias ágeis de desenvolvimento pode contribuir para facilitar a comunicação e colaboração dos alunos, melhorando o gerenciamento e a adaptação do processo de criação do software. As metodologias ágeis surgiram como alternativa aos modelos de desenvolvimento tradicionais, onde o cliente tinha pouca participação, e consiste da utilização de ciclos iterativos e incrementais,

para atender aos critérios de qualidade na produção do software [Prickladinick *et al.*, 2009]. O uso de ciclos mais curtos de desenvolvimento alteraram a forma como os produtos são concebidos, construídos, testados e entregues aos clientes finais. Além disso, metodologias ágeis, no processo de ensino, pode permitir um aprendizado mais consistente do modelo de processo de software, já que cada ciclo de entrega foca em todas as etapas do desenvolvimento dos projetos, visando tornar o software cada vez mais confiável, correto e seguro, por meio de ciclos mais curtos com validação do cliente para minimizar os riscos na entrega do produto final [Santos *et al.*, 2010].

Por essa razão, empresas de desenvolvimento de software tem adotado métodos ágeis como forma de maximizar a entrega de produtos de software, o que popularizou o uso de tais métodos [Prickladinick, *et al.* 2009]. Dentre os métodos mais utilizados temos: SCRUM, eXtreme Programming (XP), Lean Development. Devido sua característica, com ênfase no gerenciamento do desenvolvimento, o SCRUM é atualmente o método mais utilizado nas empresas de desenvolvimento de software [Prickladinick, *et al.* 2009].

O funcionamento do SCRUM baseia-se em princípios como: equipes pequenas de, no máximo, sete pessoas; requisitos que são pouco estáveis ou desconhecidos; e iterações curtas. E divide o desenvolvimento em três papéis: *Product Owner (P.O)*: representa os interesses de todos no projeto; *Time*: desenvolve as funcionalidades do produto; *ScrumMaster*: garante que todos sigam as regras e práticas do Scrum, além de ser o responsável por remover os impedimentos do projeto.

Para melhor gerenciamento, o desenvolvimento é dividido em intervalos de tempos de no máximo, trinta dias, chamados de *Sprints*. Durante a execução das *Sprints*, diariamente o time faz uma reunião de quinze minutos, chamadas de *Daily Meeting*, para acompanhar o progresso do trabalho e agendar outras reuniões necessárias. No final da *Sprint* é realizada a reunião de revisão (*Sprint Review Meeting*) para que o Time apresente o resultado alcançado na iteração ao P.O. Neste momento, as/funcionalidades são inspecionadas e adaptações do projeto podem ser realizadas. Em seguida, o *ScrumMaster* conduz a reunião de retrospectiva (*Sprint Retrospective Meeting*), com o objetivo de melhorar o processo/time e/ou produto para a próxima *Sprint*.

Devido ao funcionamento do SCRUM, que permite maior comunicação e colaboração entre os envolvidos no processo de desenvolvimento, a combinação desse método com PBL pode facilitar o processo de ensino em Engenharia de Software, permitindo criar novas práticas pedagógicas de ensino e habilidades que possam ser aplicadas no mercado de trabalho. A próxima Seção apresenta um relato da experiência no uso das duas tecnologias.

3. Combinando o Método Ágil SCRUM com PBL no Ensino de ES: um Estudo de Caso

Com o intuito de melhorar o processo de ensino-aprendizagem em Engenharia de Software, foi adotada como abordagem pedagógica o uso de PBL e o método ágil SCRUM. O estudo de caso foi realizado na disciplina de Engenharia de Software II, do curso de Sistemas de Informação, no Instituto de Ciências Exatas e Tecnologia (ICET) da Universidade Federal do Amazonas (UFAM). A turma formada por 18 alunos foi

organizada em quatro grupos, onde cada grupo determinava um aplicativo a ser desenvolvido ao longo da disciplina.

O processo adotado para o ensino consistiu em estabelecer papéis, por meio de aplicação de conceitos teórico-práticos, onde após as aulas conceituais sobre Engenharia de Software, os alunos aplicavam os conceitos práticos no desenvolvimento dos seus respectivos aplicativos. O propósito com essa estratégia foi incentivar a colaboração e comunicação e estimular os alunos a desenvolver novas habilidades, simulando o ambiente real. A Figura 1 apresenta o processo adotado e os principais atores envolvidos.



Figura 1: Processo de ensino usado na disciplina.

O contexto foi simular uma fábrica de software, onde cada equipe ficaria responsável por desenvolver um projeto, passando por cada fase do processo de desenvolvimento. Para isso, o gerenciamento foi feito usando o SCRUM, por conter marcos e papéis bem definidos, com entregas contínuas, o que permite melhor gestão do desenvolvimento.

O processo consiste em: 1) aulas e elaboração de materiais didáticos, que foram desenvolvidos por meio de dinâmicas de grupos, treinamentos em laboratórios e aulas expositivas em sala de aula; 2) desafios que foram elaborados para simular as entregas do aplicativo, onde a cada entrega os membros de cada grupo deveriam apresentar um ambiente no qual se almeja vender um produto, estimulando os grupos a desempenharem habilidades de comunicação e oratória; 3) processo de condução dos desafios foram gerenciados utilizando as fases do SCRUM.

Para melhor compreensão dos conhecimentos preliminares de cada aluno, foi elaborado um formulário de caracterização com informações que permitissem aos alunos descrever o grau de conhecimento nos conceitos de Engenharia de Software conforme apresentado na Tabela 1. A resposta do Formulário foi útil para melhor compreensão e elaboração do material de ensino.

Tabela 1 - Caracterização do Perfil dos Alunos.

Conceitos	Descrição	Participantes
<i>Conhecimento em Engenharia de Software</i>	Não possuo nenhum conhecimento.	3
	Possuo conhecimento adquirido em sala de aula, mas nunca aplicado de forma prática.	15
	Possuo conhecimento adquirido em sala de aula, aplicado em outra(s) disciplina(s) de forma prática.	0
<i>Conhecimento em Processos de Software</i>	Não possuo nenhum conhecimento em processos de software	1
	Possuo conhecimento, adquirido por meio de livros, blogs e fontes externas (Internet).	1
	Possuo conhecimento adquirido em sala de aula, mas nunca aplicado de forma prática.	15
	Possuo conhecimento adquirido em sala de aula e aplicado em outra(s) disciplina(s).	1
	Possuo conhecimento sólido e aplicado de forma prática em outros projetos (profissionais).	0
<i>Conhecimento em Métodos Ágeis</i>	Não possuo nenhum conhecimento em processos de software.	1
	Possuo conhecimento, adquirido por meio de livros, blogs e fontes externas (Internet).	1
	Possuo conhecimento adquirido em sala de aula, mas nunca aplicado de forma prática.	13
	Possuo conhecimento adquirido em sala de aula e aplicado em outra(s) disciplina(s).	0
	Possuo conhecimento sólido e aplicado de forma prática em outros projetos (profissionais).	0
<i>Tipos de Metodologias</i>	SCRUM.	10
	eXtreme Programming.	9
	Lean Development.	3
	Feature Driven-Development.	4
	Agile Modelling.	2
	Outros.	0

3.1 Condução do Processo de Ensino

O processo foi aplicado no período de agosto a novembro de 2014, e foi estruturado da seguinte forma: o professor da disciplina inicialmente apresentava os conceitos de ES, processos e metodologias aplicados no desenvolvimento de software. E a cada duas aulas teóricas, uma aula prática era realizada em laboratório. O objetivo consistiu em misturar os aspectos práticos aplicados no desenvolvimento de aplicativos móveis.

Após a etapa de ensino, foi solicitado um desafio utilizando os conceitos abordados nas aulas anteriores, que continham o incremento do aplicativo a ser desenvolvido. Os desafios continham atividades relativas ao desenvolvimento específico de cada projeto, tais como: elicitação de requisitos, projeto, protótipo e entregas de incrementos do aplicativo. A cada desafio os membros se organizavam, estabelecendo os papéis do scrum, como forma de enriquecer a experiência dos alunos em cada papel do processo. Ao final de cada desafio era coletados a percepção de aprendizado, o grau de contribuição e a motivação dos alunos, feitas em apresentação onde os alunos mencionavam as dificuldades, as lições aprendidas, o que foi bom durante a execução do trabalho. A partir da coleta dos dados foi possível avaliar a contribuição da combinação da metodologia.

3.2 Resultados Obtidos

Ao final da disciplina, dos 18 alunos apenas 15 foram aprovados. Como as atividades foram desenvolvidas em grupos os aplicativos desenvolvidos não sofreram prejuízo no desenvolvimento e foram apresentados, considerando a apresentação formal do software. Os aplicativos desenvolvidos são mostrados na Figura 2.

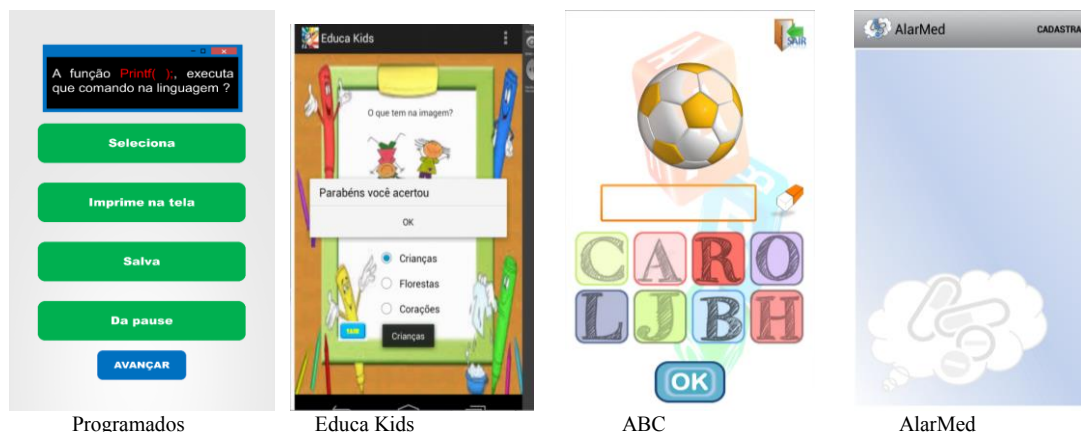


Figura 2: Telas capturadas dos aplicativos desenvolvidos.

Como o desempenho na disciplina pode não prover dados satisfatório, decidiu-se realizar uma avaliação qualitativa. Para avaliar a experiência dos alunos e o grau de satisfação com a metodologia aplicada, foi utilizado um questionário com informações organizadas em três categorias de aprendizagem: 1) **Percepção sobre o aprendizado**: consiste em estabelecer o nível de aprendizado dos estudantes; 2) **Percepção sobre motivação**: grau de motivação que a abordagem pedagógica trouxe para o aprendizado dos alunos; 3) **Percepção do valor do aprendizado**: consiste em avaliar a relevância do que foi ensinado na disciplina.

Sobre a percepção de aprendizado é possível observar, na Figura 3, que maior parte dos alunos conseguiu aprender os novos conceitos em ES. No entanto, grande parte dos conceitos abordados não foram assimilados devido à falta de tempo para execução da disciplina, como relatado por alguns alunos: "O tempo para ministrar as aulas da disciplina era curto, principalmente na prática de desenvolvimento de tecnologia mobile" e também "Acredito que o espaço de tempo foi curto para o desenvolvimento da disciplina.". Acredita-se que um mês a mais seria ideal para o termino das atividades de desenvolvimento dos aplicativos. Devido à falta de tempo, focou-se em aspectos das etapas iniciais, o que explica a maior contribuição na fase de análise e projeto, conforme a Figura 3.

Você se sente capaz de desenvolver uma aplicação, no paradigma ensina, com mais habilidade que antes da disciplina? A disciplina agregou novos conhecimentos em qual conceito especificamente

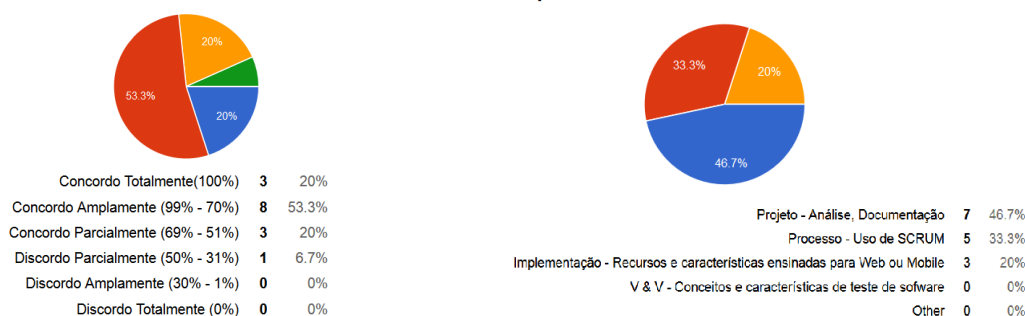
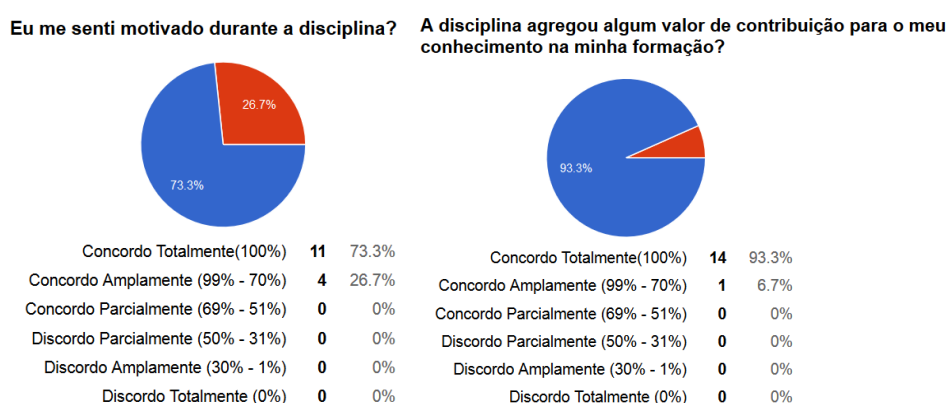
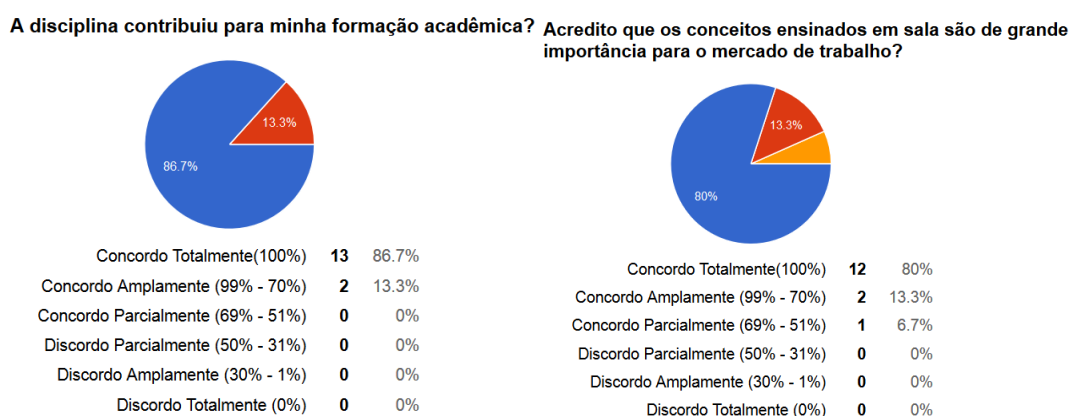


Figura 3: Percepção sobre aprendizagem.

Quanto a percepção sobre motivação a Figura 4 mostra que os alunos se sentiram bastante motivados. O que gera um indício de que **a combinação de PBL e SCRUM pode criar um ambiente mais dinâmico e prazeroso aos alunos**. De acordo com a Figura 4, o grau de concordância no ensino de processos, usando PBL teve significativa contribuição, conforme relatado por um aluno: *"Essa disciplina foi muito importante para minha formação porque pude aplicar tudo que aprendi em sala de aula, ou seja, é mais fácil você aprender fazendo."* Em relação ao grau de contribuição de métodos ágeis a disciplina foi útil para aprimorar os conhecimentos dos alunos como relatado: *"Contribuiu para compreender melhor o que são metodologias ágeis, como trabalhar em equipe e resolver problemas mais rápido."*

**Figura 4: Percepção sobre a motivação: grau de motivação que a abordagem pedagógica trouxe para o aprendizado dos alunos**

Sobre o valor do aprendizado e o grau de importância da sua aplicação, os alunos se mostraram bastantes favoráveis ao uso da abordagem de ensino, conforme a Figura 5. Especialmente sobre **os conceitos aprendidos e como estes podem contribuir para oportunidades no mercado de trabalho**.

**Figura 5: Percepção do valor do aprendizado**

Apesar de alguns resultados desfavoráveis, devido a falta de tempo, esse estudo foi importante para avaliar a combinação de PBL e métodos ágeis no ensino de ES, gerando importante contribuição para o ensino da área.

5. Lições Aprendidas e Considerações Finais

Com a crescente evolução do processo de desenvolvimento das novas tecnologias de software, o ensino de Engenharia de Software se torna cada vez mais importante. A busca por profissionais mais qualificados nas tecnologias usadas no mercado atual faz com que as instituições busquem alternativas inovadoras no processo de ensino, visando melhorar a integração dos envolvidos com atividades pedagógicas mais colaborativas para tornar o processo de aprendizagem mais dinâmico e prazeroso.

Este artigo apresenta um relato de experiência de alunos de graduação, incumbidos da realização de projeto de desenvolvimento para dispositivos móveis na disciplina de Engenharia de Software por meio de uma combinação baseada em problema (PBL) e SCRUM. A experiência prática dos alunos permitiu aos mesmos identificar desafios e soluções durante a realização do projeto, além de ter uma visão de como a experiência prática em um projeto real ajuda a equipe a ter um melhor entendimento das teorias, práticas, métodos, processos e ferramentas que envolvem a Engenharia de Software.

Apesar das dificuldades dos alunos, relatadas em entrevistas, durante o processo de ensino, o uso da abordagem se mostrou bastante favorável, misturando a aplicação dos conceitos teórico-práticos que tornaram possível a criação de quatro aplicativos móveis na área de educação (ABC, Educa kids e Programados) e também para auxiliar as pessoas no controle de medicamentos (AlarMed). Com o relato é possível perceber as dificuldades e adaptar as metodologias de ensino visando melhorar a experiência dos alunos para outras disciplinas da área de Engenharia de Software. Espera-se com este relato, contribuir para disseminar o uso de PBL e métodos ágeis no processo de ensino e aprendizagem da Engenharia de Software, buscando proporcionar melhor qualificação dos profissionais, por meio da teoria aliada a prática.

Como trabalhos futuros, espera-se melhorar a metodologia e aplicar em novas turmas, aplicando para outros paradigmas como: Programação Web e Sistemas Embarcados, realizando uma análise mais rica sobre as percepções de aprendizado, usando grupos de controle como forma de determinar o nível de aprendizado após o estudo sobre os conceitos ministrados em sala.

6. Referências

- Andrade, Antônio Gabriel Pereira de *et al.* (2010) “Aplicação do Método PBL no Ensino de Engenharia de Software: Visão do Estudante”, In: ERBASE.
- Baker, A.; Navarro, E.O.; Van der Hoek, A. (2005), “An Experimental Card Game for Teaching Software Engineering Processes”, In: Journal of Systems and Software, New York, v. 75, n. 1-2, p. 3-16.
- Bezerra, C. A.; Trindade, C. C.; Souza, E.P.R.; Magalhães, C. V. C.; Santos, R. E. S.(2014) “Aplicação de Técnicas de Aprendizagem de Máquina em Objetos de Aprendizagem baseado em Software: um Mapeamento Sistemático a partir das Publicações do SBIE”, RENO. Revista Novas Tecnologias na Educação, v. 12, p. 380-390.
- Bonifácio, B. A.; Grana, C. B.; Coutinho, W.; Couto, E.; Fernandes, P. S. (2015) “Improving an education mobile application through usability evaluations”,

- In: 9th International Technology, Education and Development Conference, Madrid, Espanha. v. 1. p. 30-42.
- Costa, C.; Rocha, R.; Figueirêdo, J.; Duarte, M.; Meira, S.; Prikladnicki, R. (2010) “Ensino da Engenharia de Software por meio de Fábricas de Software no contexto Distribuído: Um Relato de Experiência”, In: III Fórum de Educação em Engenharia de Software.
- Glasemann M., Kanstrup Anne Marie and Ryberg T. (2010) “Design and Exploration of a Mobile Game Scenario in a Diabetic Youth Camp”, In: IADIS International Conference Mobile Learning, Porto, Portugal.
- Hilburn, T.B and Towhidnejad, M. (2007) “A Case For Software Engineering”, In: Conference On Software Engineering Education & Training, Dublin. Proceedings Washington: IEEE Computer Society, p. 107-114.
- Huang, S. and Distant, D. (2006) “On Practice-Oriented Software Engineering Education”, In: Conference on Software Engineering Education & Training Workshops, Turtle Bay. Proceedings. Washington: IEEE Computer Society, p. 15.
- Marçal, E. ; Andrade, R.M. ; Júlio Wilson Ribeiro ; Leite Junior, A. J. M. ; Lima, L. ; Viana, W.(2010) “Da Elicitação de Requisitos ao Desenvolvimento de Aplicações de Mobile Learning em Matemática”, In: Simpósio Brasileiro de Informática na Educação, João Pessoa..
- Martins, J. S. (2007). Projetos de Pesquisa: estratégias de ensino e aprendizagem em sala de aula. Armazém do Ipê (Autores Associados), Campinas, SP, 2 edição.
- Nauman, M. and Uzair, M. (2007) “SE and CS Collaboration: Training Students for Engi-neering Large, Complex Systems”. In: Conference on Software Engineering Education & Training, Dublin. Proceedings. Washington: IEEE Computer Society, p. 167-174.
- Power T. and Shrestha P. (2010) “Exploration in the Context of Bangladesh”, In: IADIS International Conference Mobile Learning, Porto, Portugal.
- Prikladnicki, R. et al. (2009) “Ensino de engenharia de software: desafios, estratégias de ensino e lições aprendidas”, In: II Fórum de Educação em Engenharia de Software.
- Ronnie E.S.; Magalhães, Cleyton V. C.; Correia Neto, Jorge S.; Souza, Ellen P.R.; Vilar, Guilherme (2014) “Ferramentas, métodos e experiências no ensino de Engenharia de *Software*: um mapeamento sistemático”, In: XXV Simpósio Brasileiro de Informática na Educação.
- Santos et al (2010) “Experiência acadêmica de uma fábrica de software utilizando scrum no desenvolvimento de software”, In: Wokshop Brasileiro de Métodos Ágeis. p. 86-96.
- Thopson, J. B. (2007) “Improving Software Practice through Education: Challenges and Future Trends”, In: Future of software engineering, international conference on software engineering, Minneapolis. Proceedings. Washington: IEEE Computer Society, p. 12-28.