

SCRUM PROJECT BASED LEARNING (PBL) NA EDUCAÇÃO DE LICENCIATURA EM ENGENHARIA INFORMÁTICA

SCRUM PROJECT BASED LEARNING (PBL) IN COMPUTER ENGINEERING UNDERGRADUATE EDUCATION

APRENDIZAJE BASADO EN PROYECTOS (ABP) EN LA ENSEÑANZA UNIVERSITARIA DE INGENIERÍA INFORMÁTICA

Helder Rodrigo Pinto¹ [0009-0003-5810-9383]

¹ISLA Gaia, Portugal, helder.pinto@islaguia.pt

Resumo

Este estudo explora a integração da Aprendizagem Baseada em Projetos (PBL) e da framework Scrum na Licenciatura em Engenharia Informática, com o objetivo de preparar os estudantes para os desafios do mercado tecnológico. A metodologia coloca os alunos no centro do processo educativo, promovendo a aplicação prática de conhecimentos teóricos através de projetos semi-reais. O PBL desenvolve competências como trabalho em equipa e pensamento crítico, enquanto o Scrum introduz práticas ágeis de gestão de projetos. A implementação destas metodologias é exemplificada no projeto da Unidade Curricular de Laboratório de Programação, estruturado em sprints. A avaliação baseia-se em critérios técnicos e na eficácia da metodologia de trabalho. Os resultados indicam que os estudantes adquiriram competências técnicas e transversais, melhorando a comunicação e a adaptação a desafios. A conclusão destaca que a combinação PBL-Scrum enriquece o processo educativo, preparando os estudantes para o mercado de trabalho com uma mentalidade ágil e capacidade de aprendizagem contínua. Este modelo híbrido é uma opção viável para atender às necessidades atuais da educação em Engenharia Informática, proporcionando um ambiente de aprendizagem dinâmico e colaborativo.

Palavras-chave: Project-Based Learning (PBL), SCRUM, Software, Engenharia Informática

Abstract

This study explores the integration of Project-Based Learning (PBL) and the Scrum framework in the Computer Engineering degree, aiming to prepare students for the challenges of the technological market. The methodology places students at the center of the educational process, promoting the practical application of theoretical knowledge through semi-real projects. PBL develops skills such as teamwork and critical thinking, while Scrum introduces agile project management practices. The implementation of these methodologies is exemplified in the Programming Laboratory Curricular Unit project, structured in sprints. The assessment is based on technical criteria and the effectiveness of the work methodology. Results indicate that students acquired technical and transversal skills, improving communication and adaptation to challenges. The conclusion highlights that the PBL-Scrum combination enriches the educational process, preparing students for the job market with an agile mindset and continuous learning capability. This hybrid model is a viable option to meet the current needs of education in Computer Engineering, providing a dynamic and collaborative learning environment.

Keywords: Project-Based Learning (PBL), SCRUM, Software, Computer Engineering

Resumen

Este estudio explora la integración del Aprendizaje Basado en Proyectos (ABP) y el marco de trabajo Scrum en el grado de Ingeniería Informática, con el objetivo de preparar a los estudiantes para los retos del mercado tecnológico.

La metodología sitúa a los estudiantes en el centro del proceso educativo, promoviendo la aplicación práctica del conocimiento teórico a través de proyectos semi-reales. El ABP desarrolla habilidades como el trabajo en equipo y el pensamiento crítico, mientras que Scrum introduce prácticas de gestión ágil de proyectos. La implementación de estas metodologías se ejemplifica en el proyecto de la Unidad Curricular de Laboratorio de Programación, estructurado en sprints. La evaluación se basa en criterios técnicos y en la efectividad de la metodología de trabajo. Los resultados indican que los estudiantes adquirieron habilidades técnicas y transversales, mejorando su comunicación y adaptación a los desafíos. La conclusión destaca que la combinación ABP-Scrum enriquece el proceso educativo, preparando a los estudiantes para el mercado laboral con una mentalidad ágil y capacidad de aprendizaje continuo. Este modelo híbrido es una opción viable para satisfacer las necesidades actuales de la educación en Ingeniería Informática, proporcionando un entorno de aprendizaje dinámico y colaborativo.

Palabras clave: Aprendizaje Basado en Proyectos (ABP), SCRUM, Software, Ingeniería Informática

INTRODUÇÃO

A integração de metodologias ativas de aprendizagem, como a Aprendizagem Baseada em Projetos (PBL) e a framework ágil Scrum, no ensino superior, em particular nos cursos de Engenharia Informática, tem demonstrado um potencial significativo para melhorar a aquisição de competências dos estudantes. O PBL coloca os alunos no centro do processo educativo, incentivando-os a aplicar conhecimentos teóricos em projetos práticos e a desenvolver competências essenciais como trabalho em equipa, resolução de problemas e pensamento crítico.

A adoção do Scrum, por sua vez, complementa a metodologia PBL ao introduzir práticas de gestão de projetos ágeis que fomentam a flexibilidade, a colaboração e a entrega contínua de valor. Esta framework ágil permite que os estudantes experimentem o ciclo de vida do desenvolvimento de software num contexto semi-real, desde o planeamento e a análise de requisitos até à implementação e os testes. Através da divisão do trabalho em sprints e da realização de reuniões de planeamento, revisão e análise, os alunos aprendem a gerir o tempo eficazmente, a adaptar-se a mudanças e a trabalhar de forma autónoma e responsável.

No contexto do curso de Licenciatura em Engenharia Informática do Instituto Politécnico de Gestão e Tecnologia, a implementação destas metodologias é exemplificada através do projeto da Unidade Curricular (UC) de Laboratório de Programação. Este projeto, estruturado em sprints conforme os princípios do Scrum, visa a integração e aplicação dos conhecimentos adquiridos em diversas unidades curriculares. A metodologia de trabalho proposta enfatiza a análise de user stories, a aplicação do processo de engenharia de software, o planeamento e distribuição equitativa de tarefas e a colaboração efetiva em equipa. A avaliação do projeto segue critérios específicos, com foco nos aspetos técnicos e na metodologia de trabalho adotada, oferecendo aos estudantes uma oportunidade valiosa de desenvolverem as suas competências técnicas e transversais num ambiente que simula desafios profissionais reais.

A necessidade de inovação no ensino de Engenharia é impulsionada pela evolução constante da necessidade do mercado e pela transformação digital. Segundo Castells (2009), o poder de comunicar conhecimento deslocou-se dos centros de produção para a estrutura dinâmica da sociedade, exigindo uma reconfiguração do ensino para torná-lo mais eficiente. A implementação de metodologias ativas, como a metodologia PBL e o SCRUM, tem mostrado ser eficaz na criação de um ambiente de aprendizagem mais interativo e multidisciplinar (Devincenzi et al., 2022).

1 APRENDIZAGEM BASEADA EM PROJETOS

A metodologia de Aprendizagem Baseada em Projetos (PBL) constitui uma estratégia pedagógica que coloca os estudantes como protagonistas do seu percurso educativo, integrando-os em projetos reais. Nesta abordagem, os estudantes colaboram em grupo para abordar questões práticas, fazendo uso dos seus conhecimentos teóricos. O PBL fomenta a cooperação, a inovação, a capacidade de resolver problemas e o raciocínio crítico, dotando os alunos de competências que os permitam superar desafios complexos no seu futuro contexto profissional. Esta metodologia incentiva o sentido de responsabilidade e o empenho dos alunos, conduzindo a um processo de aprendizagem profundo e persistente. (Markham, T., et al., 2008)

De acordo com Campos (2011), a Aprendizagem Baseada em Projetos destaca-se como um tópico central nas discussões não só por ser uma metodologia de aprendizagem ativa, mas também como uma opção viável para a criação de currículos e a implementação de práticas inovadoras no ensino de engenharia. O PBL representa uma

abordagem pedagógica do século XXI, que requer um maior comprometimento tanto por parte dos estudantes quanto dos docentes.

No PBL espera-se que (Stepien, W., et al., 1998):

- Os alunos se envolvam ativamente nas discussões, contribuindo com conhecimentos e experiências anteriores, e, nas reuniões seguintes, partilhem novas aprendizagens, justificando-as com referências bibliográficas.
- Colaborem com o grupo para ultrapassar desafios que possam surgir durante o desenvolvimento do projeto.

No entanto, Powell (2000) refere que a metodologia PBL pode apresentar algumas desvantagens para os alunos numa perspectiva de que o conhecimento teórico mais avançado possa ser menos explorado, insuficiente e/ou impreciso. Além de que, estas metodologias, também obrigam a uma adaptação, por parte dos alunos, a trabalharem ao ritmo de um grupo, podendo gerar sentimentos de frustração àqueles que têm mais dificuldades por se compararem.

1.1 Planeamento dos Projetos

Para Batista (2004), "*planejar não pode ser considerado uma atividade neutra, mas intencional; conta com a projeção de objetivos e estabelecimento de meios, sendo um exercício de reflexão ideologicamente comprometido.*"

Com o objetivo de concluir com sucesso um projeto, é essencial que o docente orientador incentive e desafie o estudante a integrar-se numa equipa, dado que o PBL constitui uma abordagem eficaz, ainda que exigente, requerendo visão, organização e um entendimento do projeto, definição de cronogramas, estratégias de gestão e uma avaliação dos resultados obtidos. Segundo Markham et al. (2008), os estudantes devem ser guiados através dos seguintes passos:

- Criação ou estudo da ideia do projeto,
- Definição do âmbito e levantamento de requisitos do projeto,
- Escolha dos critérios de avaliação,
- Integração dos resultados obtidos de forma contínua,
- Desenvolvimento baseado na concepção inicial do projeto,
- Estabelecimento de um ambiente de trabalho propício.

A metodologia PBL é utilizada como estratégia educativa em várias disciplinas no ensino superior. Contudo, na prática, os facilitadores frequentemente adaptam o PBL, enriquecendo o conceito original. (Wells, S., et al., 2009)

2 SCRUM

Para Prikladnicki, Willi e Milani (2014) citados por Conceição e Silveira (2015), o Scrum é uma framework ágil que facilita a gestão de projetos complexos e a criação de produtos, sendo amplamente adotado no desenvolvimento de software. Esta framework é aplicável a projetos de diversas áreas que apresentem complexidade, permitindo práticas de monitorização e constante adaptação.

Fadel e Silveira (2010), menciona que o Scrum não prevê uma estratégia fixa para o desenvolvimento de software, mas tem como principal foco a gestão de equipas e o planeamento das atividades, dando autonomia aos colaboradores para que escolham as melhores estratégias de atingir os objetivos.

Segundo Martins et al. (2009), o Scrum possibilita entregas mais eficientes, de maior qualidade e em menos tempo, adaptando-se às necessidades do cliente e seguindo os princípios do manifesto ágil para orientar o processo de desenvolvimento.

2.1 Intervenientes

Segundo Srivastava (2017), no Scrum, os principais intervenientes são:

- Scrum Master, que é responsável por eliminar impedimentos que possam atrapalhar o progresso da equipa; facilita as reuniões Scrum e assegura que as práticas Scrum são seguidas corretamente.

- Product Owner, que representa os interesses dos stakeholders e do cliente; define e prioriza os itens do Product backlog, garantindo que a equipa trabalha nas funcionalidades mais valiosas.
- Equipa de Desenvolvimento, composta por profissionais multifuncionais, incluindo programadores, testadores e outros especialistas necessários para o projeto; e responsável por entregar incrementos de produto potencialmente utilizáveis ao final de cada sprint.

O Scrum promove uma colaboração estreita entre estes intervenientes, assegurando que todos estão alinhados com os objetivos do projeto e que a comunicação é eficaz. A equipa de desenvolvimento trabalha de forma autónoma, mas com o suporte contínuo do Scrum Master e a orientação do Product Owner.

2.2 Product Backlog

Para Srivastava (2017) e Sachdeva (2016) Product Backlog é uma lista ordenada de todos os requisitos desejados para o produto em desenvolvimento, apresentada em formato de user stories. Este backlog é gerido pelo Product Owner e serve como a fonte única de requisitos para qualquer alteração necessária no produto. Cada item no Product Backlog inclui uma descrição, uma prioridade e uma estimativa do esforço necessário para completá-lo. O Product Backlog é dinâmico e evolui à medida que o produto e o ambiente em que ele será usado evoluem. É constantemente atualizado para refletir novas necessidades, mudanças de prioridade e feedback dos stakeholders. A priorização dos itens é feita pelo Product Owner, garantindo que a equipa de desenvolvimento trabalha nas funcionalidades mais valiosas e alinhadas com os objetivos do negócio.

2.3 User Stories

Segundo Cohn (2009), as user stories são descrições em linguagem natural e semi-estruturada, feitas da perspectiva do utilizador, que detalham a funcionalidade necessária em sistemas de software. Estas, são cruciais ao longo do ciclo de vida do projeto para esclarecer objetivos e funcionalidade do sistema, monitorizar o progresso, identificar problemas e aumentar a satisfação do cliente.

2.4 Sprint

De acordo com Schwaber e Sutherland (2013), a sprint é o evento principal do Scrum, sendo o período no qual são criados incrementos "prontos", e que normalmente têm durações de até um mês. Uma nova sprint inicia-se logo após a conclusão da anterior. Cada sprint possui uma definição do que deve ser construído. Ao longo do processo, o âmbito é detalhado e revisto entre a equipa e o Product Owner, podendo ser clarificado e renegociado, desde que não afete os objetivos inicialmente descritos. Além disso, não é aconselhável alterar a composição da equipa ou a sua duração.

2.5 Reuniões

Segundo Srivastava (2017) e Sachdeva (2016):

- O Sprint Planning é uma reunião que marca o início de cada sprint. Durante esta reunião, a equipa de desenvolvimento, o Scrum Master e o Product Owner colaboram para definir quais itens do Product Backlog serão trabalhados na próxima sprint. O Product Owner apresenta os itens prioritários e a equipa de desenvolvimento estima o esforço necessário para completar cada item. O objetivo é selecionar um conjunto de itens que a equipa acredita ser capaz de completar durante a sprint, criando assim o Sprint Backlog.
- A Daily Scrum é uma reunião curta, geralmente de 15 minutos, realizada todos os dias durante a sprint. O objetivo é sincronizar as atividades da equipa e criar um plano para as próximas 24 horas. Cada membro da equipa responde a três perguntas: O que fiz ontem? O que farei hoje? Existem impedimentos no meu caminho? Esta reunião é facilitada pelo Scrum Master e ajuda a identificar rapidamente problemas e a ajustar o plano de trabalho conforme necessário.
- A Sprint Review é realizada no final de cada sprint e tem como objetivo inspecionar o incremento do produto e adaptar o Product Backlog, se necessário. Durante esta reunião, a equipa de desenvolvimento demonstra o trabalho concluído aos stakeholders, incluindo o Product Owner. Os stakeholders fornecem feedback, que é utilizado para ajustar o Product Backlog e planear as próximas sprints. Esta reunião é crucial para garantir

que o produto está a evoluir de acordo com as expectativas dos stakeholders e que qualquer ajuste necessário é feito de forma ágil.

- A Sprint Retrospective é uma reunião realizada após a Sprint Review e antes do início do próximo Planning. O objetivo é refletir sobre o sprint que acabou de terminar e identificar oportunidades de melhoria. A equipa discute o que correu bem, o que poderia ser melhorado e desenvolve um plano para implementar melhorias no próximo sprint. Esta reunião é facilitada pelo Scrum Master e é essencial para a melhoria contínua da equipa e do processo de desenvolvimento.

2.6 Scrum como metodologia educativa

Usar o Scrum como metodologia no ensino, conforme discutido por Devincenzi et al. (2022), tem como objetivo melhorar a gestão e o controlo das equipas em projetos de aprendizagem. Este modelo híbrido combina a metodologia PBL com práticas ágeis do Scrum, facilitando a divisão de tarefas, a responsabilidade individual e a colaboração em equipa.

Estudos, como os de Doerschuk (2004) e Bell e Prabhu (2014), demonstraram melhorias significativas na gestão de tarefas e no desempenho individual dos estudantes, através de reuniões semanais e monitorização contínua. Esta abordagem inovadora mostra-se eficaz na potencialização do processo de ensino-aprendizagem.

A adoção do Scrum no ensino, integrando a metodologia PBL e práticas ágeis, promove melhor gestão de tarefas, responsabilidade individual e colaboração, contribuindo para melhorias significativas no desempenho dos estudantes.

3 PROJETO DA UC LABORATÓRIO DE PROGRAMAÇÃO – UM CASO PRÁTICO

3.1 Enquadramento

O quinto semestre da Licenciatura em Engenharia Informática do ISLA – Instituto Politécnico de Gestão e Tecnologia adota um método de Aprendizagem Baseada no desenvolvimento de um Projeto único. Este método tem como objetivo a integração e aplicação prática dos conhecimentos adquiridos em várias Unidades Curriculares (UC), tais como: Fundamentos de Programação; Algoritmos e Estruturas de Dados; Programação Orientada a Objetos; Programação Avançada; Programação de Bases de Dados; Programação Web – Cliente; Programação Web – Servidor; Análise e Modelação de Sistemas; e Engenharia de Software; Interfaces e Usabilidade; entre outras.

O processo inicia-se com uma fase de Planeamento, onde é criado um checklist inicial que reúne todas as características conhecidas do projeto e possíveis estimativas de solução (SCRUM Guide, 2020). Em seguida, são definidos os Sprints, períodos curtos e iterativos de trabalho focados em tarefas específicas, seguindo um cronograma detalhado, para a realização de diferentes fases, tal como é exemplificado na Tabela 1.

Tabela 1

Exemplo de Calendarização do Projeto

Sprint	Objetivos gerais
A	Proceder ao levantamento de requisitos; Desenhar os modelos de análise de SI e Engenharia de Software; Planear a gestão do projeto.
B	Implementar a estrutura da Base de Dados; Desenvolver o estudo de UI/UX.
C	Criar a estrutura base client-side; Implementar e testar as funcionalidade client-side e server-side de nível 1.
D	Implementar e testar as funcionalidade client-side e server-side de nível 2; Proceder à aplicação de testes funcionais.

Nota. Esta tabela descreve os objetivos a serem desenvolvidos em cada uma das quatro fases (sprint) do projeto.

Cada sprint prevê uma apresentação intermédia, a *sprint review*, que é marcada com um *deadline*.

3.2 Metodologia de Trabalho

3.2.1 Introdução

Para iniciar o semestre de forma eficaz, é divulgado um enunciado que equilibre uma visão geral e clara do projeto, bem como, que seja passível de deixar espaço para a autonomia e a tomada de decisões por parte dos estudantes ao longo dos sprints. Nesta visão geral do projeto, o enunciado destaca os objetivos principais e como estes estão relacionados com as Unidades Curriculares.

A estrutura do enunciado deve responder aos seguintes pontos:

- **Introdução ao Projeto:** descrição do propósito geral do projeto.
- **Objetivos Gerais:** os objetivos principais do projeto; como devem ser desenvolvidas as competências técnicas; e como promover a colaboração em equipa, bem como aplicar metodologias ágeis.
- **Metodologia e Estrutura das Sprints:** descrição da estrutura dos sprints, incluindo as tarefas principais de cada, destacando a importância das *sprint reviews* e das reuniões periódicas.
- **Descrição do Problema:** apresentação inicial que revele a essência do problema a ser desenvolvido. sem entrar em detalhes específicos das sprints, permitindo que os estudantes explorem e definam os detalhes ao longo do tempo.
- **Levantamento de Requisitos:** apresentação de vários requisitos funcionais e não funcionais, bem como alguns cenários ou problemas que o projeto pretende abordar ou dar resposta, sem detalhar as tarefas específicas.
- **Expectativas:** lista de resultados esperados no final do projeto, como um sistema funcional, documentação completa e uma apresentação final.
- **CrITÉrios de Avaliação:** lista de critérios gerais para cada sprint e alternativas a avaliação contínua.

Podem ser disponibilizados alguns componentes previamente implementados, para que os estudantes analisem a melhor forma de integrar esses componentes.

3.2.2 Sprints

No início de cada sprint, é disponibilizado um conjunto de *user stories* indicativas do trabalho a ser realizado, quase como uma *check-list*, sendo da responsabilidade de cada equipa, em cada sprint:

- Analisar a informação disponibilizada e, se necessário, solicitar esclarecimentos relevantes para a conclusão bem-sucedida dessas *user stories*;
- Aplicar corretamente o processo de engenharia, incluindo o processo de desenvolvimento de software e as suas diferentes atividades (i.e.: análise, design, implementação, testes, implantação), deixando evidências da sua aplicação;
- Planejar as tarefas a serem realizadas, tendo em consideração os objetivos do sprint e possíveis dependências entre tarefas. Se necessário, rever e ajustar o planeamento anteriormente realizado;
- Distribuir equitativamente as tarefas entre os membros da equipa, considerando fatores como esforço, conhecimento e complexidade. Isso garante que cada membro esteja apto a descrever e explicar como um requisito específico foi ou está a ser implementado no sistema e possa efetuar alterações quando necessário, bem como defender e argumentar as tomadas de decisão sobre o mesmo, durante uma avaliação.

Ao longo do desenvolvimento do projeto são realizadas reuniões periódicas para monitorizar o progresso, distribuir ou redistribuir tarefas e fomentar a responsabilidade individual e coletiva dos estudantes (Fernandes et al., 2021).

Estas reuniões acontecem durante as aulas práticas. A integração do SCRUM no PBL permite uma gestão mais eficaz das atividades, promovendo a autonomia dos estudantes e a colaboração em equipa.

As equipas são orientadas a definir os seus objetivos e a trabalharem regularmente no projeto utilizando ferramentas, processos e métodos adequados para gerir e executar o projeto, bem como partilhar informações entre todos os membros.

3.3 Avaliação

A avaliação do projeto é de acordo com a respetiva Ficha de Unidade Curricular (FUC).

As avaliações são de carácter individual e/ou em equipa, sobre o trabalho realizado até ao momento, no final de cada sprint. Os critérios de avaliação centram-se nos aspetos técnicos do projeto relacionados com as user stories identificadas, bem como em algumas competências transversais como o trabalho em equipa, responsabilidade, capacidade de resolver problemas, comunicação e capacidade de síntese.

No entanto, a avaliação de um projeto em grupo, especialmente em contexto académico, deve refletir tanto o desempenho coletivo quanto o individual. Além dos requisitos específicos nas User Stories, são propostos alguns dos critérios para avaliação em equipa, como:

- **Qualidade do Produto Final:**
 - **Funcionalidade:** Avaliar se o sistema atende aos requisitos especificados.
 - **Usabilidade:** Avaliar a interface e a experiência do utilizador.
 - **Desempenho:** Verificar a eficiência e a eficácia do sistema.
 - **Documentação:** Avaliar a completude e a clareza da documentação técnica e do utilizador.
- **Processo de Desenvolvimento:**
 - **Adesão ao Scrum:** Avaliar a aplicação correta das práticas ágeis.
 - **Planeamento e Execução dos Sprints:** Verificar a eficácia do planeamento e a execução das tarefas.
 - **Evidências de Aplicação de Engenharia de Software:** Avaliar a qualidade das análises, designs, implementações e testes realizados.

Quanto à avaliação individual, podem ser enumerados a seguinte proposta de critérios:

- **Contribuição Individual:**
 - **Participação nas Reuniões:** Avaliar a presença e a participação ativa nas reuniões de sprint review.
 - **Tarefas Realizadas:** Verificar a quantidade e a qualidade das tarefas realizadas por cada membro.
 - **Responsabilidade e Colaboração:** Avaliar a responsabilidade individual e a colaboração com a equipa através de ferramentas como JIRA, Trello ou GitHub, bem como registos de commits no repositório de código para monitorizar o progresso das tarefas e a contribuição individual.
- **Autoavaliação e Avaliação pelos Pares:**
 - **Autoavaliação:** Cada aluno deve refletir sobre sua própria contribuição e desempenho.
 - **Avaliação pelos Pares:** Os membros da equipa devem avaliar-se uns aos outros, considerando fatores como esforço, qualidade do trabalho e colaboração.

As sprint reviews, são o momento formal de avaliação, pois consistem nas apresentações intermédias, no final de cada sprint, onde a equipa demonstra o trabalho realizado e recebe feedback. A apresentação deve ser abrangente do produto final, mostrando um caso de sucesso principal e destacando as funcionalidades implementadas, os desafios superados e as conclusões.

Além de fazer um ponto de situação, as reuniões têm também como objetivo fornecer feedback construtivo e orientar os alunos sobre como melhorar.

No caso dos estudantes com estatuto especial e que não tenham possibilidade de comparecer às aulas regularmente, podem optar fazer a apresentação do projeto integral no deadline do último sprint, assumindo que será perdida a experiência de trabalhar por sprints de forma ágil. Não havendo a possibilidade de fazer equipas neste regime,

assume-se ainda que será perdida a oportunidade de trabalhar em equipa e que resultará bem mais trabalho apenas para um elemento.

4 RESULTADOS

Este modelo foi testado com estudantes da Licenciatura em Engenharia de Informática do ISLA Gaia.

Os alunos expressaram uma apreciação positiva, destacando como esta abordagem os prepara de forma abrangente para o mercado de trabalho. A integração da metodologia PBL com a framework Scrum permitiu-lhes não só consolidar os conhecimentos teóricos adquiridos nas diversas disciplinas, mas também incentivou a investigação aprofundada e o desenvolvimento de competências transversais cruciais, como trabalho em equipa, responsabilidade, gestão de tempo, resolução de problemas e capacidade de comunicação e apresentação. Esta metodologia pedagógica promoveu uma aprendizagem ativa, colocando os estudantes no centro do processo educativo e encorajando-os a aplicar na prática os conceitos teóricos. Além disso, preparou-os eficazmente para o projeto final do curso promovendo várias estratégias e recursos que podem reutilizar, assegurando que adquirissem as competências necessárias para enfrentar os desafios do setor tecnológico com confiança e competência.

Os estudantes relataram que o método foi eficiente em auxiliar todos os grupos, melhorando a comunicação entre alunos e professores e facilitando a adaptação às dificuldades encontradas durante o processo.

A resposta entusiástica dos alunos a este projeto reflete o seu reconhecimento do valor desta abordagem integrada na sua formação académica e profissional.

CONCLUSÃO

A implementação da Aprendizagem-Baseada em Projetos (PBL) juntamente com a framework Scrum na Licenciatura em Engenharia Informática representa um avanço significativo na forma como o ensino e a aprendizagem são conduzidos no ensino superior, especialmente em áreas de rápido desenvolvimento como as Tecnologias da Informação (IT). Esta abordagem pedagógica não só promove uma maior interação entre teoria e prática, mas também prepara os estudantes de forma mais eficaz para os desafios do mercado de trabalho, ao simular condições reais de desenvolvimento de projetos e fomentar competências como liderança, trabalho em equipa, gestão do tempo e adaptabilidade. Através da realização de projetos que integram conhecimentos de diversas unidades curriculares e da aplicação de metodologias ágeis de gestão de projetos, os estudantes são capazes de desenvolver uma compreensão holística do processo de desenvolvimento de software, desde a concepção até à entrega final, bem como apresentações ao cliente, preparando-os para serem profissionais competentes e inovadores.

Além disso, a experiência prática adquirida com a aplicação destas metodologias no contexto educativo promove uma aprendizagem ativa, que é fundamental para o desenvolvimento de competências técnicas avançadas e para a formação de profissionais qualificados e adaptáveis às mudanças constantes do setor tecnológico. A avaliação contínua do trabalho em equipa e do projeto, baseada em critérios técnicos e metodológicos, assegura que os estudantes não apenas adquiram conhecimentos específicos da área de Engenharia Informática, mas também desenvolvam uma mentalidade ágil e uma capacidade de aprendizagem contínua. Assim, a integração do PBL e do Scrum na educação em Engenharia Informática não só enriquece o processo educativo, mas também contribui significativamente para a formação de futuros profissionais preparados para enfrentar os desafios e as oportunidades do século XXI.

Em suma, o modelo híbrido PBL-SCRUM mostra-se uma opção viável para atender às necessidades atuais da educação em Engenharia Informática, proporcionando um ambiente de aprendizagem mais dinâmico e colaborativo, alinhado com as exigências do mercado e as novas diretrizes curriculares.

REFERÊNCIAS

- Batista, N.A. Planejamento na prática docente em saúde. In: Batista NA, Batista SH, orgs. Docência em saúde: temas e experiências.
- Bell, J. T., Prabhu, A. (2014) "An innovative approach to software engineering term projects coordinating student efforts between multiple teams over multiple semesters",
- Campos, L.C., Aprendizagem Baseada em projetos: uma nova abordagem para a Educação em Engenharia.
- Castells, M. (2009) Communication power. Oxford University Press.
- Cohn, M. (2009) User Stories Applied For Agile Software Development, 13th ed.
- Conceição, J.; Silveira, S. R. (2015) Aplicação de Metodologias Ágeis para Desenvolvimento de Software: um Estudo de Caso na Empresa Alliance Software.
- Devincenzi, S. et al. (2022) SCRUM applied to Problem-Based Learning: a hybrid model for managing the teaching-learning process.
- Doerschuk, P. (2004) "Incorporating team software development and quality assurance in software engineering education".
- Fadel, A. C.; Silveira, H. M. (2010) Metodologias ágeis no contexto de desenvolvimento de software: XP, Scrum e Lean.
- Fernandes, S., Dinis-Carvalho, J., & Ferreira-Oliveira, A. T. (2021) Improving the performance of student teams in project-based learning with scrum.
- Markham, T., Larmer, J., Ravitz, J., (2008) Aprendizagem Baseada em Projetos
- Martins, L.; Rocha, M.; Santos, M.; Savione, M. (2009) Análise de Gerenciamento de Projeto de Software Utilizando Metodologia Ágil XP e Scrum: Um Estudo de Caso Prático.
- Powell, P. From classical to project-led education. In: Pouzada, A. S. (ed) (2000) Project based learning: project-led education and group learning.
- Sachdeva, S. Scrum Methodology. (2016) International Journal Of Engineering And Computer Science, 5(6), 16792-16799.
- Schwaber, J.; Sutherland, K. (2013) Um guia definitivo para o Scrum: As regras do jogo.
- SCRUM Guide.(2020) The Definitive Guide to Scrum: The Rules of the Game.
- Srivastava, A., Bhardwaj, S., & Saraswat, S. (2017) SCRUM Model for Agile Methodology.
- Stepien, W.; Gallagher, S. (1998) Problem-based learning: as authentic as it gets. In: R. Fogarty, R. (Ed.). Problem-based learning: a collection of articles.
- Wells, S., Warelow, P., & Jackson, K. (2009). Problem based learning (PBL): A conundrum. Contemporary Nurse, 33(2), 191-201. <https://doi.org/10.5172/conu.2009.33.2.191>