

# Ensino Remoto Emergencial de Engenharia de Software com PBL: um relato de experiência

Simone R. S. Souza<sup>1</sup>, João Choma Neto<sup>1</sup>, Leo Natan Paschoal<sup>1</sup>, Elis Hernandez<sup>2</sup>

<sup>1</sup>Universidade de São Paulo (ICMC/USP)  
P.O. Box 1212 – São Carlos – SP – Brasil

<sup>2</sup>Monitora Soluções Tecnológicas – São Carlos – SP – Brasil

srocio@icmc.usp.br, {joaochoma,paschoalln}@usp.br

elis.hernandes@monitoretec.com.br

**Resumo.** *A adaptação do ensino de engenharia de software para o modelo remoto é desafiador principalmente quando Project-based learning é adotado. Esta metodologia oferece uma oportunidade de aplicar a teoria em um projeto prático, melhorando o aprendizado e a motivação dos estudantes. Este artigo relata a experiência em ensinar engenharia de software por meio de aulas online no 1o semestre de 2020. A abordagem para adaptação a esse novo modelo é apresentada, incluindo como os projetos práticos foram desenvolvidos em parceria com uma empresa de TI da cidade. Como resultados, são apresentados dados qualitativos obtidos com os participantes, os quais indicam as impressões sobre o aprendizado e a experiência. De uma maneira geral, apesar das várias dificuldades impostas pelo novo modelo, observou-se que a comunicação e o aprendizado foram muito positivos nesse modelo remoto de ensino.*

## 1. Introdução

De acordo com as diretrizes curriculares da SBC [Zorzo et al. 2017] e ACM/IEEE [ACM and IEEE 2013], é esperado que o ensino de computação ofereça, além de conhecimento teórico, oportunidades para que os estudantes desenvolvam habilidades práticas necessárias para a sua formação. Em relação ao ensino de Engenharia de Software, é possível estabelecer estratégias que potencializem o aprimoramento de algumas habilidades essenciais, dentre elas a capacidade de: a) projetar soluções computacionais para diferentes domínios; b) trabalhar em equipes e gerenciá-las, e c) aplicar conceitos, métodos, técnicas e ferramentas para o gerenciamento de projetos de software. Observam-se iniciativas que relatam estratégias para o ensino de Engenharia de Software que buscam contemplar um balanceamento entre teoria e prática [Jianguo Chen et al. 2009, Silva et al. 2017, Bastarrica et al. 2017, Ramos et al. 2018, Moresi et al. 2019]. Estes trabalhos mostram o quão desafiador é ensinar engenharia de software, pois não é trivial equacionar projetos práticos em que os alunos consigam exercitar os conceitos, porém sem serem projetos extensos demais que tornem inviável a aplicação prática dos conceitos aprendidos.

Durante o desenvolvimento de um projeto de software, é esperado enfrentar dificuldades como: entender um novo domínio, trabalhar com requisitos de software inconsistentes ou faltantes, interagir com cliente, trabalhar com tecnologias novas, definir prazos e estabelecer cronogramas. Essas dificuldades só são vivenciadas realmente quando

os estudantes “aprendem fazendo”, ou seja, aprendem desenvolvendo projetos práticos. *Project-based learning* tem se mostrado uma alternativa eficiente para o ensino de engenharia de software [Bastarrica et al. 2017, Fioravanti et al. 2018, Souza et al. 2019]. Nessa abordagem, os estudantes constroem conhecimento aplicando a teoria para resolver problemas e desafios que imitam o “mundo real” [DeFillippi 2001]. Essa abordagem permite engajar os estudantes, os quais de maneira colaborativa desenvolvem uma solução.

Experiências anteriores no ensino de Engenharia de Software em nosso Instituto, utilizaram *Project-based learning* para apoiar a prática da disciplina [Souza et al. 2016, Paschoal and Souza 2020]. Com a parceria de empresas e *startups* da região, projetos reais, de escopo reduzido, foram fornecidos aos estudantes, os quais foram estimulados a gerarem soluções de software. Essas experiências indicaram a importância da parceria com as empresas, que atuam como clientes dos projetos, para que a prática do desenvolvimento de software seja vivenciada de maneira estimulante.

O isolamento social imposto devido à pandemia do Covid-19 exigiu mudanças drásticas na maneira em que realizamos nossas atividades cotidianas. Em particular, o ensino foi transferido imediatamente para o modelo remoto e, no nosso Instituto, adaptações foram realizadas *on the fly*. Buscou-se manter a qualidade dos cursos e atender todas as especificidades causadas pelo isolamento, dentre elas, as dificuldades dos estudantes em manter o mesmo horário e tempo para as aulas. Uma nova maneira de ensinar e interagir com os estudantes foi necessária, pois o ensino que se buscou não foi o Ensino a distância (EaD) e sim um ensino remoto emergencial, síncrono e com o máximo possível de interação com os estudantes. Dado esse desafio, a experiência de ensinar Engenharia de Software usando *Project-based learning* precisou ser adaptada para esse nova realidade.

Este artigo relata a experiência em ensinar Engenharia de Software de maneira remota, no 1o semestre de 2020, com a participação da Monitora<sup>1</sup>, organização de desenvolvimento de software da cidade, que atuou como “*cliente*” dos projetos reais propostos aos estudantes. Durante a experiência, os estudantes aplicaram as práticas do framework Scrum para o desenvolvimento de projetos reais. Diferentemente do que foi realizado em experiências anteriores similares, nesta disciplina os estudantes iniciaram a especificação dos seus projetos no início do semestre e, a medida que o conteúdo ia sendo ensinado, os mesmos eram aplicados no projeto em construção. Adaptações foram realizadas para que houvesse um melhor acompanhamento dos times pelo professor, tutor e clientes. Ao final, estudantes e clientes forneceram um *feedback* a respeito da experiência, indicando as percepções, aprendizado e oportunidades observadas pela abordagem de ensino.

O artigo está organizado da seguinte forma. Na Seção 2 é apresentada a metodologia de ensino adotada, descrevendo como os projetos foram desenvolvidos. Na Seção 3 são apresentados os resultados sobre a percepção dos estudantes em relação à abordagem de ensino adotada. Na Seção 4 discutem-se os resultados obtidos. A Seção 5 apresenta as conclusões sobre essa experiência de ensino.

## 2. Metodologia de Ensino

Esta experiência foi realizada na disciplina de Engenharia de Software do curso de Bacharelado em Sistemas de Informação (noturno) do Instituto de Ciências Matemáticas e

---

<sup>1</sup><https://www.monitoretec.com.br/>

de Computação (ICMC/USP). A disciplina é obrigatória com carga horária de 90 horas (4 horas-aula semanais) e é oferecida no 5º semestre do curso. O objetivo da disciplina é oferecer uma visão geral sobre o gerenciamento de projetos de software, dos processos de desenvolvimento e dos métodos e técnicas que podem ser utilizadas no ciclo de vida do software. A disciplina contou com o apoio de um aluno de doutorado, que exerceu o papel de estagiário de ensino e colaborou em todas as atividades. No total, 41 estudantes participaram da disciplina até a sua finalização.

Com a mudança do ensino presencial para o remoto emergencial, manteve-se, na medida do possível, o planejamento inicial da disciplina. As aulas foram mantidas no mesmo horário, divididas em dois momentos:

- Parte teórica: ensino dos aspectos conceituais do conteúdo, por meio de aulas online síncronas na plataforma *Google Meet*. As aulas foram gravadas e disponibilizadas, juntamente com todo o material. Exercícios foram propostos e discutidas suas resoluções.
- Parte prática: desenvolvimento dos projetos desde a sua especificação até a implementação do protótipo, com atividades previstas para as aulas e extra-aulas. Na Seção 2.1 é descrito como os projetos e protótipos foram desenvolvidos.

Antes do isolamento, foi possível realizar cinco aulas presenciais. Além dos conceitos iniciais, foram realizadas dinâmicas para preparar os estudantes para o desenvolvimento dos projetos. Em uma dessas aulas, os estudantes foram apresentados à equipe da Monitora que realizou um treinamento sobre o framework Scrum. Buscando melhorar a dinâmica da disciplina, a descrição dos projetos foi apresentada no início do semestre e, a medida que o conteúdo teórico foi ensinado, os estudantes foram aplicando nos seus projetos. Essa alteração foi feita com base em experiências anteriores, em que os estudantes alegaram que houve pouco tempo para trabalharem nos projetos e que seria interessante se pudessem começar o projeto a medida que o conteúdo fosse ensinado. Por meio do *Google Meet* foi possível identificar os alunos presentes e o tempo logados nas aulas. O acesso aos materiais no Moodle também foi monitorado.

## 2.1. Desenvolvimento dos Projetos

Os alunos foram divididos em cinco equipes de 8 a 9 integrantes. Cada equipe escolheu um aluno para ser o *PO - Product Owner* e outro para ser o *Scrum Master* da equipe. Para cada projeto, a Monitora indicou um funcionário para exercer o papel de cliente, atuando também como tutores e dando suporte para a aplicação do Scrum. A Tabela 1 apresenta os projetos utilizados.

As equipes definiram as ferramentas de gerenciamento de projetos a serem utilizadas (Trello, Taiga, Jira foram as escolhidas) e o sistemas de versionamento do software (Github e Gitlab) para que professor, aluno estagiário e tutores pudessem acompanhar a evolução do projeto. O desenvolvimento seguiu as práticas do Scrum, definindo-se: *product backlog*, *sprint backlog*, *sprint planning*, *daily Scrum*, *sprint review* e *sprint retrospective*. Os estudantes não tem dedicação integral ao projeto. Por ser um curso noturno, muitos trabalham e, desta forma, foi orientado que eles adaptassem a rotina do Scrum conforme a disponibilidade da equipe. Além disso, houve a questão do desenvolvimento remoto dos projetos, os quais tiveram que interagir a distância para o desenvolvimento.

---

<sup>2</sup><http://gapsi.icmc.usp.br/>

**Tabela 1. Descrição dos projetos**

Projeto	Descrição do projeto
Saúde Mental	Aplicativo de apoio à prevenção do risco de suicídio e orientações focada no trabalho do GAPsi-ICMC <sup>2</sup> .
Master of Notas	Aplicativo tipo <i>Quiz</i> para disciplinas no qual os alunos matriculados podem responder perguntas e acumular pontos na referida disciplina. Os pontos podem ser trocados por notas ou outras vantagens.
Vigilância Comunitária	Aplicativo para as comunidades se organizarem nas ações de "vigilância". Os usuários podem fazer denúncias, alertar fatos, ocorrências, convocar vizinhos e organizar ações de interesse na comunidade.
Trabalho Voluntário	Sistema que reúne eventos que precisam de voluntários para trabalhar nos mesmos. Interessados podem buscar trabalhos de interesse e se voluntariar.
Quarto de Família	Aplicativo para gerenciar o aluguel de quartos de família, similar ao <i>Homestay</i> em que a pessoa participa da família. Esse aplicativo pode ser utilizado também por idosos que buscam companhia para ajudar em tarefas cotidianas.

Um cronograma de atividades a serem desenvolvidas foi apresentado aos times, dando a liberdade para optarem pela melhor forma de produzirem e organizarem suas atividades (Tabela 2). No período de Fevereiro a Maio, ocorreram as aulas teóricas. Acompanhando a teoria, neste período, os times desenvolveram o Planejamento, Análise e Projeto do Software. Os artefatos desta fase foram produzidos a medida que os conteúdos foram transmitidos (Tabela 2). A partir da 2<sup>a</sup> quinzena do mês de maio, os estudantes iniciaram a implementação dos projetos. Neste período, os horários das aulas foram destinados apenas para reuniões com os times, nas quais eram apresentados os resultados parciais, avanços, dificuldades e situação atual dos projetos. Nesses encontros, foi possível avaliar a aplicação prática do *Scrum*, observando os avanços nas ferramentas de gerenciamento dos projetos e as entregas feitas.

Para o desenvolvimento, foram sugeridos *sprints* de uma a duas semanas, entretanto as equipes podiam modificar o tempo e quantidade de *sprints* de acordo com os seus projetos e capacidade das equipes. As ferramentas de gestão do projeto ajudaram as equipes na comunicação, principalmente neste modelo remoto de desenvolvimento. Assim, os membros do time que estavam em um determinado *sprint* estudando as tecnologias, deveriam adicionar o estudo de tecnologias como uma de suas atividades na ferramenta de gestão. Com isso foi possível ter uma visão melhor da participação de todos, mesmo com o distanciamento da equipe.

Uma mudança importante foi definir que os times deveriam marcar reuniões semanais com os clientes, mostrando a evolução dos projetos. Nas reuniões foi solicitado que os clientes dessem direcionamentos sobre as práticas ágeis, incluindo auxiliar na organização do projeto nas ferramentas de gestão. Também, auxiliaram na priorização de requisitos para o desenvolvimento dos protótipos.

Os times realizaram uma apresentação final dos resultados para a turma e para os clientes da Monitora. Um vídeo do protótipo desenvolvido foi apresentado juntamente com algumas discussões sobre: as decisões dos projeto (processo de desenvolvimento, ferramentas e tecnologias escolhidas), os resultados, continuidade e lições aprendidas. O material utilizado na disciplina para acompanhar e avaliar os projetos, bem como as apresentações finais, estão organizados no repositório disponível em<sup>3</sup>.

<sup>3</sup>Repositório com material da disciplina relacionado aos projetos:< <https://cutt.ly/PxzITYc>>.

**Tabela 2. Cronograma e artefatos gerados no desenvolvimento dos projetos**

Atividades	Meses	Artefatos
Planejamento do Projeto	março a maio	Plano do projeto Cronograma inicial do projeto Plano de teste
Análise	março a abril	Planejamento da entrevista Prototipação e validação com o cliente Construção do <i>backlog</i> Priorização do <i>backlog</i> e validação com o cliente
Projeto	abril a maio	Modelagem do Banco de Dados Modelagem Casos de Uso Definição da arquitetura do software
Implementação Total de 5 sprints	maio a julho	Sprint planning Desenvolvimento do sprint Sprint Review e Retrospective Entrega de cada sprint
Apresentação final	julho	Preparar apresentação e vídeos

### 3. Percepção dos estudantes

Para avaliar a percepção dos estudantes sobre a disciplina, um questionário foi construído e aplicado no final da disciplina. As respostas não foram utilizadas na avaliação e os alunos participaram voluntariamente, de maneira anônima (Formulário disponível em <sup>4</sup>).

As questões buscam avaliar 4 principais aspectos: 1) sobre o uso do processo ágil; 2) sobre o desenvolvimento de projetos reais; 3) sobre a aprendizagem percebida e 4) sobre o ensino remoto. Os resultados são sintetizados a seguir.

#### 3.1. Sobre o uso do processo ágil

Neste item foi questionado se os estudantes concordam que o o processo ágil é adequado aos projetos desenvolvidos, se o mesmo foi aplicado de maneira adequada a distância e se o processo ágil colaborou com o desenvolvimento do projeto a distância.

Com base nas respostas, foi identificado que 88% dos alunos consideraram que o processo ágil foi adequado ao desenvolvimento do projeto, 77% dos alunos concordaram que o processo ágil foi aplicado adequadamente para o trabalho remoto e, na mesma proporção, 77% dos alunos afirmaram que o processo ágil ajudou no desenvolvimento do projeto remoto. Neste sentido, a maioria dos estudantes respondeu que o processo ágil colaborou positivamente para o desenvolvimento dos projetos, mesmo de maneira remota.

#### 3.2. Sobre o desenvolvimento de projetos reais

Neste item, foi verificado como que os alunos se sentiram no início do desenvolvimento do projeto e no final, em relação a trabalhar com clientes reais, as tecnologias envolvidas e o fato de desenvolver o projeto de maneira remota, com as equipes distribuídas.

Em relação às expectativas de desenvolver um projeto real, antes e depois do seu desenvolvimento, foi observado que os alunos imaginavam um projeto complexo e com muitas restrições: *“Que fosse algo mais difícil e exigente por ter um cliente para pedir e aprovar todo o nosso trabalho”*, *“[...] as experiências adquiridas iriam além daquelas adquiridas nos livros”*, *“Mais responsabilidade, achei que poderia não conseguir corresponder ao esperado”*, *“Esperava que seria muito burocrático e engessado, sem espaço*

<sup>4</sup><https://cutt.ly/qxb7cCi>

*para tomarmos muitas das decisões do projeto”, “Foi muito mais motivante saber que o projeto tem uma aplicação real e que não seria usado somente para dar uma nota ao aluno. E que após o término da disciplina ele não ficará guardado em uma pasta do professor [...]”.*

Com o decorrer das semanas, os estudantes observaram que as maiores dificuldades foram sobre a necessidade deles se adaptarem as novas tecnologias de gestão e desenvolvimento, ao contrário do que imaginavam no início: “[...] *foi muito confuso, ninguém conhecia as ferramentas direito, todos aprendendo quase do zero*”, “[...] *o contato com a empresa não só nos forneceu um feedback rico, como uma maneira ágil de pensar e executar decisões importantes*”, “[...] *Senti que tivemos muita liberdade na hora de planejar e desenvolver o projeto. Isso foi ótimo*”, “[...] *o cliente se mostrou muito cooperativo e ajudou no desenvolvimento*”.

Em relação à participação da empresa, os alunos relataram que conseguiram obter conhecimento profissional que completou a teoria vista: “[...] *Com eles (clientes), a sensação de estar fazendo algo realmente importante no nosso projeto estava presente o tempo todo*”, “[...] *Acredito que a presença deles agregou ao time [...]*”, “[...] *desde motivar, explicar e instruir como fazer, dando uma visão de mundo real a disciplina*”.

Quando questionados sobre suas percepções iniciais em relação ao desenvolvimento do projeto a distância, foi observado que eles estavam preocupados em conseguir gerenciar a comunicação e o tempo oferecido ao projeto: “[...] *Imaginei que seria um pouco mais difícil em questão de horário [...]*”, “[...] *Que por estarmos totalmente a distância, fosse ser mais fácil a comunicação [...]*”, “[...] *como boa parte da minha equipe trabalha, senti que seria bem complicado encontrar horários em comum pra reuniões*”.

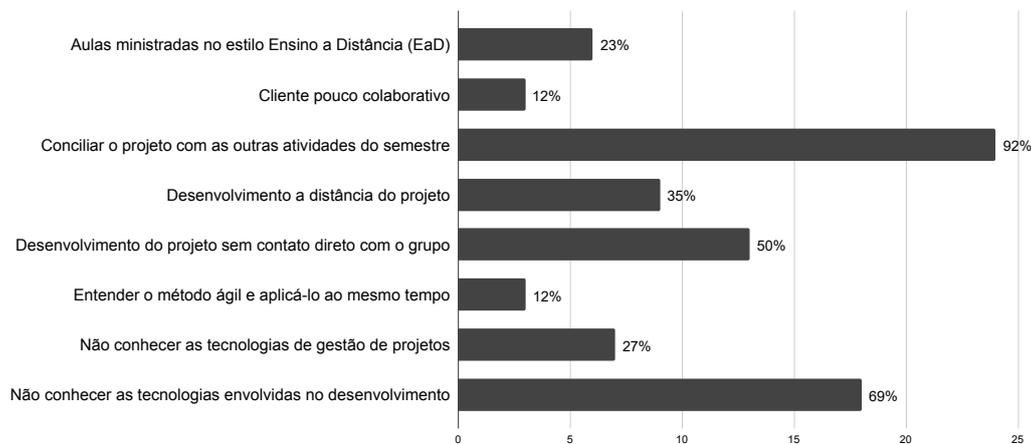
### **3.3. Sobre a aprendizagem percebida**

A Figura 1 apresenta o que os estudantes selecionaram como maiores dificuldades enfrentadas durante a realização dos projetos. A maior dificuldade apontada (92,3%) foi conciliar o projeto com as outras atividades do semestre. Este era um resultado esperado porque a maioria dos alunos não tem dedicação exclusiva ao curso de graduação. Outro resultado alto é a dificuldade por não conhecer as tecnologias envolvidas no desenvolvimento (69,2%) e o desenvolvimento remoto, sem contato presencial com a equipe (50%).

Ainda sobre a aprendizagem percebida, os alunos indicaram a importância do desenvolvimento da aula utilizando a abordagem PBL, a inclusão de projetos reais e a importância de tutores diretamente do mercado profissional que agregaram conceitos não teóricos as atividades do projeto: “[...] *Achei demais a ideia, antes não sabia que a disciplina corria dessa forma, com clientes e projetos reais*”, “[...] *consegui aprender novas coisas que podem vir a ser úteis no futuro [...]*”, “[...] *Gostei muito da imersão que tivemos, principalmente da participação de clientes reais no projeto!*”.

### **3.4. Sobre o ensino remoto**

Quando questionados sobre o quanto aprenderam na disciplina, 76,9% dos alunos indicaram que absorveram melhor os conceitos apresentados devido ao projeto prático desenvolvido. Quando questionados se poderiam ter aprendido mais se as aulas fossem presenciais, 50% dos alunos afirmaram que sim e 26,9% não soube opinar.



**Figura 1. Principais dificuldades enfrentadas pelos estudantes**

Neste item foi avaliado também as expectativas iniciais em oposição as expectativas concretizadas, na percepção dos estudantes. Com relação às expectativas que tinham com o ensino remoto, observou-se que os alunos imaginavam que a disciplina seria completamente modificada da sua proposta original, com carga horária reduzida, somente vídeo-aulas e com um projeto simples: *“Minha expectativa era de que seriam cortadas muitas das atividades da disciplina planejadas para serem presenciais”, “As minhas expectativas foram de que seria mais difícil a comunicação e que seria bem complicado tomar decisões”, “Que a qualidade de ensino diminuiria”,*

Quando questionados sobre as expectativas concretizadas, algumas respostas obtidas foram: *“Minha preocupação era se eu iria conseguir testar meus conhecimentos [...], o que foi realizado com sucesso pois me adaptei muito bem com o ensino remoto, mas mesmo assim prefiro o presencial”, “[...]consegui levar muito bem a matéria, principalmente na parte prática e acho que o saldo disso foi bem positivo”, “Tudo ocorreu de forma muito mais organizada que pensava [...]”, “[...] surpreendentemente a proposta continuou e a matéria foi desenvolvida da melhor forma possível”.*

Quando questionados sobre o concretizado em relação às expectativas iniciais, de maneira geral, os estudantes perceberam um crescimento pessoal principalmente em relação à gestão de tempo: *“[...] tivemos dificuldade em reunir todos integrantes do time, mas as ferramentas disponíveis como o Jira colaboraram para que a comunicação, planejamento e desenvolvimento de forma assíncrona”, “[...]colocar em prática dicas importantes dadas pelos mentores [...]”, “[...] o trabalho à distância pareceu ter facilitado o andamento do projeto”, “A distância não foi um problema, apesar de que acredito que a integração do time seria muito melhor se as reuniões fossem feitas cara a cara”.*

#### **4. Lições Aprendidas**

O aproveitamento da disciplina foi obtido por meio de duas avaliações e pela nota do projeto final. Os critérios adotados para avaliar o projeto foram: qualidade dos artefatos gerados, uso adequado da metodologia ágil, o produto desenvolvido e a participação do aluno no projeto. Como resultado final, os estudantes tiveram um excelente aprendizado. Além disso, formulários foram aplicados aos alunos para avaliarem suas equipes.

Os clientes também avaliaram o desenvolvimento dos projetos. Eles perceberam a evolução dos alunos durante o decorrer do projeto em relação à gestão de tempo e atividades, aprendizado de tecnologias e o amadurecimento da compreensão do produto que estavam construindo. Os clientes também relataram que se preocuparam em orientar os alunos sobre organizar o projeto antes de focar na codificação, pois inicialmente os alunos estavam ansiosos para codificar e negligenciavam o processo.

Os resultados dos produtos desenvolvidos foram positivos, tanto na avaliação pelo professor, estagiários e clientes, quanto na análise das percepções dos alunos. Pelos resultados, percebe-se que conceitos principais de engenharia de software foram aprendidos, bem como, o evolução dos alunos em relação à gestão de tempo e comunicação.

As equipes utilizaram ferramentas de gestão de projetos e, com isso, foi possível acompanhar pontualmente a participação de cada membro das equipes, pois tudo era registrado a medida que cada um evoluía no projeto. Além disso, os estudantes preencheram um formulário indicando o nível de participação de seus colegas e fazendo uma autoavaliação. Como o desenvolvimento foi acompanhado semanalmente pelo professor e monitor, foi possível avaliar a colaboração de todos nos projetos.

Com base nos resultados, as seguintes considerações podem ser extraídas:

**a) Em relação ao uso do processo ágil:** acredita-se que a experiência permitiu aos estudantes “viverem” o desenvolvimento de um projeto, seguindo um processo, no caso, o Scrum. Com o acompanhamento sistemático ao longo da disciplina, garantiu-se que os estudantes aplicassem corretamente o processo, o que lhes deu maior segurança para o desenvolvimento do projeto. Apesar do desafio em desenvolver de maneira remota, com os membros das equipes distribuídos, os estudantes observaram que o processo ágil foi útil e ajudou muito nesse processo, colaborando com a comunicação na equipe.

**b) Sobre a aprendizagem percebida:** os alunos demonstraram que a aprendizagem foi potencializada com o desenvolvimento dos projetos prático, com base na qualidade dos projetos e preocupação em aplicar corretamente o processo. Como docente, sempre foi desafiador tentar convencer os estudantes de que é importante seguir um processo de desenvolvimento, se preocupar com os requisitos, especificação e projeto antes de codificar. Porém, sem a prática, o sentimento é que o aluno não consegue dimensionar essa importância. O PBL proporcionou esse aprendizado. As respostas ao questionário também indicaram que os alunos perceberam que aprenderam melhor por meio do PBL.

**c) Sobre o desenvolvimento de projetos de maneira remota:** comparado com os oferecimentos anteriores, observou-se que a postura dos estudantes mudou, estando mais proativos e conscientes de suas responsabilidades para o sucesso dos projetos. Os estudantes buscaram formas de se comunicarem com os clientes e com a equipe, aproveitando as ferramentas de gestão de projetos e de comunicação. As respostas indicaram que havia uma grande preocupação no início sobre conseguir ou não desenvolver os projetos remotamente. Acreditamos que essa preocupação foi importante para o sucesso final;

**d) Sobre as expectativas iniciais X expectativas concretizadas:** analisando as respostas, notou-se que inicialmente os estudantes estavam preocupados sobre como se comunicariam, se conseguiriam horários comuns para trabalharem nos projetos e com o desconhecimento sobre as tecnologias envolvidas. Analisando as respostas sobre o concretizado, observou-se que os estudantes ficaram satisfeitos com seus resultados, pois conseguiram

criar um bom fluxo de trabalho, melhorar a comunicação e sentiram que aprenderam as tecnologias envolvidas. Alguns relataram que a falta das aulas presenciais dificultou um vínculo maior entre os integrantes dos times.

Apesar dos ganhos observados, o ensino remoto trouxe alguns desafios:

- A mudança urgente para o modo remoto sem um tempo de planejamento, preparação de materiais e organização da disciplina foi um desafio. Devido a esse cenário, optou-se pelo ensino remoto síncrono, visando manter os alunos integrados à disciplina;
- Entender e acompanhar os alunos frente a essa mudança brusca de realidade, pois não se sabia se todos teriam as condições ideais para acompanhar a disciplina. Procurou-se usar o tempo das aulas para entender como estavam se sentindo e como estavam lidando com a situação;
- Ensinar sem ter um feedback visual dos estudantes foi bastante desafiador. Os alunos não habilitavam as câmeras apesar de solicitado. Durante os encontros, havia espaço para os alunos fazerem perguntas e tirarem dúvidas, mas poucos interagiam. Alguns utilizaram o *chat* disponível no Google Meet para escrever suas dúvidas ou pedir mais explicações. Isso foi crucial para direcionar o que estava sendo apresentado e reforçar a explicação;
- O maior desafio foram as avaliações, pois o modelo de prova não pode ser aplicado e ajustes foram feitos. Assim, optou-se por preparar avaliações formativas, nas quais os estudantes se organizaram em grupos e responderam questões na forma de problemas de engenharia de software. Com isso, tiveram que entender a teoria vista e aplicá-la para resolver os problemas das avaliações.

## 5. Conclusões

Este artigo apresentou um relato de experiência sobre o ensino de Engenharia de Software remoto, utilizando *Project-Based Learning*. Os projetos práticos foram desenvolvidos em parceria com uma empresa de tecnologia, a qual simulou “clientes” para os projetos desenvolvidos. Os projetos não são monetizados pela empresa; os clientes oferecem ideias de projetos sociais ou que possam ser continuados pelos estudantes em outras oportunidades. Os projetos foram desenvolvidos seguindo os preceitos de código aberto e os estudantes são os proprietários principais e podem dar sequência aos projetos no futuro.

A proposta de ensinar Engenharia de Software com PBL permite exercitar os conceitos teóricos. O desenvolvimento inclui planejamento e gestão do projeto, análise, projeto, testes e implementação. Com o acompanhamento sistemático, evita-se a tendência de partir para o código antes das fases anteriores, pois é enfatizado que para essa disciplina o código é menos relevante e o processo de desenvolvimento deve ser priorizado. Com base nos relatos dos estudantes, foi possível observar que o uso do PBL permitiu a vivência das principais dificuldades enfrentadas por desenvolvedores de software:

- Entender o problema a ser resolvido, de um domínio pouco conhecido;
- Interagir com um cliente e extrair dele informações sobre o projeto;
- Trabalhar com requisitos de software imprecisos e inconsistentes;
- Trabalhar em equipe, tomar decisões e gerenciar o tempo;
- Utilizar tecnologias novas e saber aplicá-las; e
- Conseguir estabelecer cronogramas e prazos realistas.

Em novos oferecimentos da disciplina, sugestões feitas deverão ser incorporadas. Dentre elas, pretende-se disponibilizar exemplos de projetos já desenvolvidos e disponibilizar com antecedência o cronograma de artefatos a serem produzidos.

## Referências

- ACM and IEEE (2013). *Computer Science Curricula 2013: Curriculum Guidelines for Undergraduate Degree Programs in Computer Science*. ACM, New York, NY, USA.
- Bastarrica, M. C., Perovich, D., and Samary, M. M. (2017). What can students get from a software engineering capstone course? In *39th International Conference on Software Engineering: Software Engineering Education and Training Track*, pages 137–145.
- DeFillippi, R. J. (2001). Introduction: Project-based learning, reflective practices and learning. *Sage Publications Sage CA: Thousand Oaks, CA*, 32(1):5–10.
- Fioravanti, M., Sena, B., Paschoal, L., Ribeiro Silva, L., Allian, A., Nakagawa, E., Souza, S., Isotani, S., and Barbosa, E. (2018). Integrating project based learning and project management for software engineering teaching: An experience report. In *49th ACM Technical Symposium on Computer Science Education*, pages 806–811.
- Jianguo Chen, Huijuan Lu, Lixin An, and Yongxia Zhou (2009). Exploring teaching methods in software engineering education. In *4th International Conference on Computer Science Education (ICCSE IEEE)*, pages 1733–1738.
- Moresi, E. A. D., Filho, M. O. B., Hartmann, V. C., and Carvalho, C. M. S. (2019). Gamificação como metodologia ativa: estudo de caso na disciplina de engenharia de software. *Revista Iberoamericana de Sistemas Cibernética e Informática*, 16(2):63–68.
- Paschoal, L. N. and Souza, S. R. S. (2020). Teaching scrum practices with the support of brazilian startups: an initiative in software engineering course. In *CibSE - Ibero-American Conference on Software Engineering*, pages 1–14.
- Ramos, C. S., Kosloski, R. A. D., Venson, E., Costa, R. M. F., and Deon, V. H. A. (2018). TBL as an active learning-teaching methodology for software engineering courses. In *XXXII Brazilian Symposium on Software Engineering, SBES '18*, page 289–297.
- Silva, W. A. F., Steinmacher, I. F., and Conte, T. U. (2017). Is it better to learn from problems or erroneous examples? In *IEEE 30th Conference on Software Engineering Education and Training (CSEET)*, pages 222–231.
- Souza, M., Moreira, R., and Figueiredo, E. (2019). Students perception on the use of project-based learning in software engineering education. In *XXXIII Brazilian Symposium on Software Engineering*, page 537–546. SBC.
- Souza, S. R. S., Oliveira, B., Grillo, F., and Cico, C. (2016). Construção de plataformas digitais durante o ensino de engenharia de software: um relato de experiência. In *Fórum de Educação em Engenharia de Software (FEES 2016) - SBES*, pages 1–10.
- Zorzo, A. F., Nunes, D., Matos, E., Steinmacher, I., Leite, J., Araujo, R. M., Correia, R., and Martins, S. (2017). *Referenciais de Formação para os Cursos de Graduação em Computação*. Sociedade Brasileira de Computação (SBC).