

# ANÁLISE DA EFETIVIDADE DA PROGRAMAÇÃO PAREADA NO ENSINO-APRENDIZAGEM DE ORIENTAÇÃO A OBJETOS PARA O DESENVOLVIMENTO MOBILE

*PAIR PROGRAMMING EFFECTIVENESS ANALYSIS IN OBJECT-ORIENTED LEARNING TO THE MOBILE APPLICATION DEVELOPMENT*

*ANÁLISIS DE LA EFICACIA DE LA PROGRAMACIÓN EN PARES EN LA ENSEÑANZA-APRENDIZAJE CON ORIENTACIÓN A OBJETOS PARA EL DESARROLLO DE APLICACIONES MÓVILES*

Ana Paula Mandelli<sup>1</sup>

## Resumo

Os sistemas de software estão cada vez mais complexos. Dentro desta perspectiva, as empresas de desenvolvimento estão mais competitivas, exigindo rapidez na produção, sem diminuir a qualidade do seu produto. A engenharia de software dedica-se ao desenvolvimento de boas práticas por meio da orientação a objetos. Porém, um dos maiores desafios a serem superados está na formação de desenvolvedores. A fim de minimizar essas questões, surge a metodologia ágil de programação pareada (PP), utilizada no *Extreme Programming* (XP), por meio da qual é possível desenvolver códigos mais eficazes com o mínimo de erros. Diante disso, por meio de pesquisa bibliográfica, esse trabalho se propõe a analisar a efetividade da programação pareada como ferramenta no ensino da orientação a objetos no desenvolvimento móvel, a fim de compreender as vantagens, os desafios e a importância da programação em par na formação dos futuros profissionais. Assim, a partir desta pesquisa, foi possível identificar os benefícios da PP no processo de ensino-aprendizagem da programação, a qual pode ser utilizada como ferramenta pedagógica, colaborando em variáveis como desempenho, confiança e motivação dos alunos, além do compartilhamento de informações, da facilidade de aprendizado e da melhora na qualidade do código, que comprovam a sua efetividade.

**Palavras-chave:** mobile; método ágil; programação pareada; orientação a objetos; ensino.

## Abstract

Software systems are increasingly more complex. In this perspective, development companies are more competitive, demanding fast production without reducing product's quality. Software engineering dedicates to developing good practices through object-oriented programming. However, future developers training is one of the biggest challenges one must overcome. To minimize these issues, pair programming arises as an agile methodology used in Extreme Programming (XP), which able to develop more efficient codes with minimal number of errors. In the face of that, bibliographic research was carried out to analyze pair programming effectiveness as an object-oriented teaching tool in mobile application development, to understand pair programming's benefits, challenges, and importance in future professional's training. Thus, from this research it was identified pair programming's benefits in programs' teaching-learning process as pedagogical tool, collaborating in variables as student's performance, confidence, and motivation, in addition to the sharing of information, ease of learning and improvement in code quality, proving its effectiveness.

**Keywords:** mobile; agile method; pair programming; object-oriented; teaching.

## Resumen

Los sistemas de software son cada vez más complejos. Dentro de esta perspectiva, las empresas desarrolladoras son más competitivas, exigiendo rapidez en la producción, sin disminuir la calidad del producto. La ingeniería de software se centra en el desarrollo de buenas prácticas a través de la orientación a objetos. Sin embargo, uno de

---

<sup>1</sup> Tecnóloga em Análise e Desenvolvimento de Sistemas (UTFPR). E-mail: ana.paula.mandelli@gmail.com

los mayores retos a superar es la formación de los desarrolladores. Para minimizar esos problemas, surge la metodología ágil de programación en pares (PP) utilizada en *Extreme Programming* (XP), mediante la cual es posible desarrollar códigos más eficientes con errores mínimos. Por ello, a través de una investigación bibliográfica, este trabajo se propone analizar la eficacia de la programación en pares como herramienta en la enseñanza de la orientación a objetos para el desarrollo móvil, con el objetivo de comprender las ventajas, retos e importancia de la programación en pares en la formación de futuros profesionales. A partir de esta investigación, se identificaron los beneficios de la PP en el proceso de enseñanza-aprendizaje de la programación, que puede ser utilizada como herramienta pedagógica, colaborando en variables como el rendimiento, la confianza y la motivación de los estudiantes, además de compartir información, de la facilidad de aprendizaje y de la mejora en la calidad del código, que demuestran su eficacia.

**Palabras-clave:** móvil; método ágil; programación en pares; orientación a objetos; enseñanza.

## 1 Introdução

Com a popularização dos equipamentos eletrônicos, em especial do computador, houve a necessidade de desenvolvimento de aplicações e softwares para o seu gerenciamento. Com o passar do tempo, os eletrônicos diminuíram de tamanho, sendo possível até mesmo guardá-los no bolso. Estes eletrônicos, denominados dispositivos móveis ou mobile, em inglês, são dispositivos de computação portáteis, geralmente equipados com um método de entrada e uma tela de exibição.

De acordo com Pressman (2006), o crescimento computacional leva à criação de sistemas de softwares complexos. Para que o sistema seja viável há uma preocupação por parte dos engenheiros de software sobre o desenvolvimento destes produtos dentro do prazo e em perfeito funcionamento.

A engenharia de software, dentro da computação, dedica-se à concepção, especificação, desenvolvimento e manutenção de sistemas de software, aplicando tecnologias e práticas de gerência de projeto. Para Pressman (2006), ela enfatiza aspectos práticos da produção de um sistema de software, relacionados às tecnologias e práticas que envolvem as linguagens de programação, banco de dados, ferramentas, plataformas, bibliotecas, padrões e processos, entre outros.

Com ideias inovadoras é possível agilizar processos de trabalho. Com a evolução da tecnologia e sua contínua expansão, um dos caminhos para “materializar” a ideia é desenvolver softwares por meio de algoritmos criados, estudados e melhorados. Uma boa prática na engenharia de software para estes códigos deu-se com a programação orientada a objeto, que é um paradigma de computação que se assemelha à forma de pensar dos seres humanos. Para realizar a computação, ao invés de uma série de comandos, utilizamos objetos que possuem dados associados e se comunicam entre si (AURELIANO; TEDESCO, 2012, p. 2).

No entanto, a dificuldade que enfrentam os alunos durante o processo de ensino-aprendizagem de programação orientada a objetos é um dos desafios a serem superados. Uma

dessas dificuldades é a modularização do programa ou a eliminação de erros e conceitos mais abstratos. Contudo, a maior delas está relacionada à combinação e à utilização adequada dos conceitos básicos de programação para a criação de um programa. Segundo Aureliano e Tedesco (2012), os alunos parecem entender os conceitos e as estruturas de uma linguagem de programação, mas não sabem utilizá-los corretamente.

Ricardo Castelhana (2018) propõe uma metodologia de trabalho pareado, ou seja, desenvolvida em pares, defendida pelo *eXtreme Programming* (XP), onde cada unidade de programação é composta por dois programadores em um único computador. Trata-se de uma prática ágil que tem sido utilizada no processo de ensino-aprendizagem da programação orientada a objetos. De acordo com especialistas, ela pode trazer diversos benefícios, como o compartilhamento de informações, a facilidade de aprendizado, a motivação dos alunos e a melhoria na qualidade do código.

O objetivo de este estudo é analisar a efetividade da programação pareada como ferramenta de ensino-aprendizagem em orientação a objetos no desenvolvimento móvel, para entender as vantagens, os desafios e a relevância da programação em par na formação dos futuros profissionais de desenvolvimento, através de uma pesquisa bibliográfica.

Além disso, este trabalho procura apresentar conceitos fundamentais sobre o tema, para que se possa entender a importância das boas práticas desenvolvidas pela engenharia de software por meio da orientação a objetos, bem como a relação das metodologias ágeis com os métodos convencionais de desenvolvimento.

Com os resultados dos estudos foi possível identificar os benefícios da programação pareada no processo de ensino-aprendizagem da programação, que pode ser utilizada de forma efetiva como ferramenta pedagógica, colaborando em variáveis como o desempenho, a confiança e a motivação dos alunos. Também se pôde observar que a programação em par surge como uma importante ferramenta neste processo, especialmente aquelas orientadas a objetos. Tal procedimento gera benefícios, entre eles a troca de informações, o aumento da qualidade do produto, a facilidade de aprendizado e a melhoria na qualidade do código. Além disso, esta pesquisa também aponta vantagens como o compartilhamento do conhecimento, a correção de falhas, a facilidade de manutenção, a confiança, o amadurecimento, a divisão do trabalho e a agilidade, entre outras.

Algumas dificuldades também são levantadas por meio desta pesquisa, como os atritos devido à personalidade dos desenvolvedores envolvidos, o entendimento do código, a falta de entrosamento, as críticas e as substituições de desenvolvedores — nada que comprometa a sua eficácia.

## 2 Parâmetros acerca da tecnologia móvel e seu desenvolvimento

As tecnologias estão avançando significativamente nesta década nas áreas de comunicação celular; redes sem fio e serviços via satélite permitem que informações e recursos possam ser acessados e usados em qualquer lugar em qualquer momento. Figueiredo (2009) aponta que a tecnologia de comunicação pioneira foi lançada em meados da década de oitenta, mais precisamente em 1983, com redes celulares de telefonia móvel voltadas para a comunicação de voz; caracterizam-se por serem sem fio, móveis e pessoais.

O aumento do uso da Internet e outras tecnologias causou uma alteração em toda a população em relação ao seu modo de pensar, agir e comprar. A partir da interatividade, segurança, velocidade e da redução dos custos de acesso à rede, a Internet deixou de ser instrumento básico de pesquisa para tornar-se uma ferramenta de marketing, além de permitir a realização de vendas, entre outras vantagens (WEBER, 2008).

Produtos tecnológicos geralmente afetam a rotina da população. Para que a nova tecnologia se torne economicamente viável, é necessário conhecimento não apenas dos processos de desenvolvimento técnico, mas também de seus estágios de inclusão e difusão nos mercados consumidores. Um exemplo é a tecnologia móvel que, na sua grande maioria, é de telefonia móvel e vem crescendo rapidamente segundo dados da Agência Nacional de Telecomunicações (ANATEL) (FIGUEIREDO, 2009).

Além da telefonia, outras tecnologias vêm atuando sobre a informação, como os recursos de interação, colaborativos, executivos que, por sua vez, direta ou indiretamente, influenciam o consumo de produtos e serviços. Rosângela Spagnol Fedoce (2011) destaca, entre outras, as mídias móveis — notebooks, celulares, tocadores de MP3/MP4, iPads, iPods, palmtops —, como meios que facilitam o processo de comunicação. Os dispositivos móveis destacam-se entre as mídias interativas, pois contam com recursos de mobilidade e portabilidade, que oferecem ao usuário a comunicação em qualquer lugar e tempo (FEDOCE, 2011).

O novo paradigma permite que usuários tenham acesso a serviços independentemente de onde estão localizados e, o mais importante, de mudanças de localização, ou seja, mobilidade. Esses dispositivos móveis em sua grande maioria possuem aplicativos para o acesso das pessoas à informação e ao conhecimento, sem restrição de tempo e espaço; essas características agregam valor estratégico à nova sociedade da chamada “era da informação” (KAMADA *et al.*, 2012).

Aplicativos móveis fazem parte de um software de serviço, entretenimento e comunicação específico, com navegadores e acesso à Internet e em sites baseados em tecnologias móveis. Para a criação de programas voltados à tecnologia móvel, o desenvolvedor obtém orientações elementares sobre o processo de configuração do ambiente e ferramentas de desenvolvimento (KAMADA *et al.*, 2012).

Kamada *et al.* (2012), em sua pesquisa, fazem uma análise comparativa das plataformas de desenvolvimento que podem ajudar a criar e implantar dispositivos móveis inteligentes, em áreas como serviços, educação e saúde. Os autores evidenciam que a nova geração de dispositivos móveis pode auxiliar na aprendizagem dos seres humanos.

Para a criação de softwares, os técnicos utilizam plataformas aplicadas à tecnologia mobile, adaptam modelos de software com as características de interface e funcionalidade de aplicação. Por meio de emuladores, avaliam uma prévia do produto, antes mesmo de ser aplicado ao dispositivo. Muitos desses aplicativos são desenvolvidos para plataformas Android e Windows Phone (KAMADA *et al.*, 2012).

O Android é uma plataforma de software para dispositivos móveis e seu sistema operacional foi escrito sob um kernel Linux; oferece uma rica interface com o usuário, suporte para funcionalidades de telefonia, aplicativos diversos e sua arquitetura está dividida em camadas, subsidiadas por diversas empresas que regulamentam e implementam os requisitos necessários para atender a cada versão da plataforma.

Um exemplo de uma ferramenta de código aberto que agrega à plataforma de desenvolvimento é o Eclipse (IDE); com códigos Java e extensões permite o desenvolvimento de linguagens Python e C/C++ (KAMADA *et al.*, 2012).

Essa tecnologia aliada à estruturação adequada de códigos e metodologias ágeis permite a elaboração de aplicativos eletrônicos que colaboram com o cotidiano de muitas pessoas.

## 2.1 Programação orientada a objetos

Muitas pesquisas têm sido feitas nos últimos anos no sentido de encontrar caminhos que facilitem o aprendizado das técnicas de orientação a objetos. Problemas e soluções já foram experimentados indicando caminhos para o aprendizado.

A maioria dos desenvolvedores utiliza a chamada programação imperativa, que aprendem em seus primeiros contatos com a área de programação. É nessa etapa que o aluno desenvolve conceitos iniciais extremamente importantes, geralmente pautados no “Portugol”

ou, muitas vezes, na Linguagem C. A programação imperativa, em resumo, é o desenvolvimento de uma sequência lógica para se chegar ao objetivo final (PIRES, 2016).

Com o tempo, os métodos de ensino foram evoluindo, as informações ficaram mais acessíveis com a Internet e recursos das tecnologias penetraram no ambiente acadêmico para auxiliar o docente no processo de ensino/aprendizagem (PIRES, 2016).

Os professores que lecionam disciplinas de programação em cursos da área de informática ou engenharia observam grande dificuldade na compreensão de conceitos básicos para construir programas de orientação a objetos.

Pires (2016) relata que, após o processo de aprendizado por meio da programação imperativa (geralmente adquirido através de aulas de introdução à programação, fundamentos de programação, algoritmos etc.), o desenvolvedor percebe que muitas das ferramentas e linguagens atuais como Java, Ruby, Python, entre outras, utilizam, além do paradigma de programação imperativa, o paradigma orientado a objetos.

Os sistemas informatizados baseiam-se em rotinas; para isso é necessário transcrever e transferir o ambiente organizacional, tal como ele funciona no cotidiano, para sistemas de computadores. Criam-se rotinas automatizadas que repliquem as práticas de forma a alcançar agilidade e precisão em processos organizacionais de qualquer natureza.

Sistemas organizacionais atuais têm sido construídos através do paradigma da orientação a objetos (OO). A organização e a documentação dos dados são feitas através da linguagem UML (Unified Modeling Language), linguagem visual utilizada para modelar softwares no paradigma da OO. Para Oliveira (2011, p. 68):

[...] a orientação a objetos surgiu com a necessidade de se criar um paradigma de programação simples, baseado na percepção humana dos objetos ao seu redor. Este novo paradigma não é apenas um modo de programar, mas uma maneira de pensar e conceber as ideias.

Ainda de acordo com o autor (OLIVEIRA, 2011, p. 70),

[...] a utilização do paradigma orientado a objetos está presente em todas as etapas do processo de desenvolvimento [de sistemas], indo da análise, passando pelo projeto, até as fases de teste e implementação. É um fator decisivo para a construção de sistemas, visando alcançar os objetivos esperados no determinado espaço de tempo com determinado custo.

O autor relata vantagens na utilização da programação orientada a objetos para a programação pareada. As principais são: menor esforço na manutenção/criação de novas

funcionalidades, reuso de código, organização do projeto e facilidade para realizar testes unitários.

No desenvolvimento mobile, grande parte das aplicações são desenvolvidas com linguagens de programação que utilizam o paradigma de orientação a objetos, que possibilita ao desenvolvedor criar seu aplicativo encapsulando lógicas de negócio e escrevendo códigos menores e mais organizados (SANTI; BEGOSSO, 2016).

O padrão da orientação a objetos é, segundo Vianna (2013), um processo conceitual independente da linguagem de programação. O desenvolvimento baseado em objetos é uma forma de pensar e não uma técnica de programação. Oliveira destaca (2011, p. 70):

[...] a orientação a objetos surgiu com a necessidade de se criar um paradigma de programação simples, baseado na percepção humana dos objetos ao seu redor. Este novo paradigma não é apenas um modo de programar, mas uma maneira de pensar e conceber as ideias.

A programação orientada a objeto introduz a noção de "classe" e "objeto". Classes são estruturas de dados que contemplam campos de dados próprios e os seus métodos. Um objeto é uma instância de uma classe, ou seja, os objetos gerados possuem as mesmas características e métodos. O surgimento desta estrutura está associado à facilitação da escrita de programas de computador, que tornou o processo mais intuitivo (VIANNA, 2013).

Vianna (2013, p. 6) relata que “um programa orientado a objetos deve ser visto como um conjunto de objetos capazes de interagir entre si. Cada objeto é capaz de receber, processar e enviar dados”. Pela programação orientada a objeto, o programador é induzido a manter o acesso aos dados somente quando a ação é associada a tal objeto.

Desde a década de sessenta, a ascensão da programação orientada a objetos foi fortemente alimentada pela emergência da Internet como um lugar de negócios. Aliada a esse fator, podemos considerar que Java foi uma das primeiras linguagens orientadas a objetos a se tornar popular; baseou-se em C++. Por ter sido projetada para uso em rede, a linguagem Java mostrou-se perfeita para a Internet, acompanhada da linguagem de marcação HTML (SANTI; BEGOSSO, 2016). A maioria das linguagens de programação até então possuíam como característica um conteúdo estático; o Java se tornou a alternativa ideal, provendo funcionalidade e dinâmica ao ambiente online.

As dificuldades atribuídas ao ensino de algoritmos, ou linguagem de programação, de modo geral, são difíceis de serem superadas; mas, com a programação pareada e metodologia pedagogicamente correta, é possível ser efetivo nessa questão. Sabe-se que a dificuldade está no reconhecimento dos procedimentos necessários para se chegar à solução do problema. Isso

implica ter que trabalhar de forma mais eficiente os processos cognitivos, em especial a abstração e a formalização, necessários à construção de um algoritmo; ou seja, a modelagem da solução do problema (BEZERRA JÚNIOR, 2013).

Para Elias Vidal Bezerra Júnior (2013), o ensino de programação e modelagem orientados a objetos tornou-se comum, mas continua sendo um desafio para muitos estudantes iniciantes. Alguns fatores afetam a aprendizagem de conceitos de programação orientada a objeto, como a falta de experiência em programação; os ambientes e softwares acessíveis para o ensino de programação orientada a objeto usualmente são muito complicados, deixando assim os estudantes confusos e dificultando o aprendizado dos conceitos básicos necessários.

## 2.2 Metodologias ágeis e a programação pareada

A engenharia de software apresenta diferentes paradigmas, como abordagem sistêmica de criação de conteúdo, bases de dados, ferramentas, linguagem de programação, produção de mídias, aspectos estéticos, tecnológicos, e de desenvolvimento de software (MATOS; BRITTO, 2007).

Segundo Breve (2002), muitos desenvolvedores se concentram na implementação e não utilizam as técnicas de engenharia de software, para melhorar e agilizar o planejamento bem como o desenvolvimento do projeto. Encontram problemas de estruturação, que tem um papel importante nesta fase.

Ocupando-se de todos os aspectos do desenvolvimento de sistemas, a engenharia de software busca selecionar o método mais apropriado para solucionar os desafios enfrentados (SOMMERVILLE, 2004).

Segundo Medeiros (2017), pode-se dizer que a engenharia de software obtém a tecnologia necessária para produzir software de alta qualidade a um baixo custo, por meio de métodos ágeis, em especial, a programação pareada. Com os desafios impostos pelo cotidiano empresarial e pelo surgimento constante de novas tecnologias, o desenvolvimento de produtos e serviços de software precisou acompanhar esse ritmo, necessitando metodologias e processos adequados.

Medeiros (2017) define os métodos ágeis como um “conjunto de metodologias que serve para acelerar o ritmo dos processos de desenvolvimento de software”. O uso dos métodos ágeis vem substituindo os modelos tradicionais, que são considerados lentos e confusos. Portanto, o objetivo é reduzir tempo e custo.

Desta forma, partindo do princípio de que os projetos têm início e fim definidos, algumas características podem ser citadas, como a vantagem de serem desenvolvidos em etapas, além de agilidade (MEDEIROS, 2017).

Como exemplo de metodologias ágeis, pode ser citado o *Extreme Programming*, conhecido como XP; trata-se de uma metodologia de desenvolvimento com foco em agilidade de equipes e qualidade de projetos. Tem como característica a simplicidade, comunicação e feedback. Essa metodologia pode ser aplicada em vários portes de projetos (MEDEIROS, 2017).

Segundo Bonfim (2014), a programação pareada é utilizada no *Extreme Programming* (XP) e a intenção dessa prática é que se desenvolvam códigos mais eficazes com o mínimo de erros. O processo de desenvolvimento consome tempo na programação em par, porém garante maior maturidade e qualidade do código que seguirá para produção. O autor também relata que cada programador que participa do desenvolvimento em par possui um papel específico, como o “condutor” que desenvolve o código em si e o “navegador” que possui a função de revisar o código, ou seja, enquanto um profissional da área escreve o código, o outro lê e antecipa situações. Para que tudo ocorra de forma mais eficaz, é importante o entrosamento da equipe (BONFIM, 2014).

Teles (2006) descreve que o XP trabalha com a premissa de que o cliente deve conduzir o desenvolvimento a partir do feedback que recebe do sistema, sendo importante a sua participação no processo, para viabilizar a sua simplicidade em vários aspectos.

Um fator importante nesta metodologia está ligado às novas aplicações e à manutenção das aplicações antigas. Estes são problemas no ciclo de vida do desenvolvimento de software, gerados pela forma como são particionadas as aplicações, geralmente usadas como módulos dependentes entre si. Nesse contexto, a programação em par é uma das práticas utilizadas pelos desenvolvedores que adotam o *Extreme Programming*. Ela sugere que todo e qualquer código produzido no projeto seja sempre implementado por duas pessoas, diante do mesmo computador, revezando-se no teclado (TELES, 2006).

### 2.3 A programação pareada como ferramenta no processo de ensino-aprendizagem: benefícios e desafios

O ensino-aprendizagem é o nome dado ao sistema de interações comportamentais entre professores e alunos. São processos complexos que recebem o nome de “ensinar” e “aprender”, constituídos por comportamentos múltiplos de interação. Sua percepção e entendimento

constitui algo crucial para o desenvolvimento de qualquer trabalho de aprendizagem, de educação ou de ensino (KUBO, 2001).

Para Kubo (2001), para contribuir com a educação em um momento de mudanças, incertezas e franca expansão do uso das tecnologias, surge a necessidade de resgatar valores importantes para a sociedade contemporânea. O professor deve entender que exercerá um novo papel neste processo, como mediador na busca do conhecimento. Precisa de novos métodos e ferramentas para a construção dos saberes, favorecer a postura reflexiva e investigativa dos educandos, colaborando assim com o seu desenvolvimento.

Tais desafios também são enfrentados no ensino superior, principalmente no ensino de linguagens de programação na engenharia de software. Ao aumentar a complexidade dos sistemas de software, cresce a dificuldade em formar profissionais qualificados para tais necessidades. Melhorar o ensino para que futuros programadores possam colaborar no desenvolvimento de software de qualidade é essencial para que esta área do conhecimento alcance a maturidade esperada. Trata-se de uma arte, conforme afirma Bezerra Júnior (2013, p. 36), apoiado em Byrne e Lyons (2001), “a arte da programação inclui o conhecimento de ferramentas de desenvolvimento, resolução de problemas, domínio de linguagens, habilidades e estratégias eficazes para a concepção e implementação de um programa”.

Nesse sentido, a programação pareada surge como uma importante ferramenta para o processo de ensino-aprendizagem da programação, especialmente aquelas orientadas a objetos. Esta prática oferece benefícios, entre eles o compartilhamento de informações, o aumento da qualidade do produto, a facilidade de aprendizado, a motivação dos alunos e a melhoria na qualidade do código.

Além disso, segundo Bonfim (2014), a programação em par traz vantagens que justificam a sua presença entre as metodologias ágeis, pois há compartilhamento do conhecimento, correção de falhas, facilidade de manutenção, confiança, amadurecimento, divisão do trabalho e agilidade. Mas o autor também aponta dificuldades recorrentes na programação em par, como a personalidade dos desenvolvedores envolvidos, o entendimento do código, a falta de entrosamento, críticas e substituições de desenvolvedores.

Como uma das técnicas aplicadas no desenvolvimento mobile, de acordo com Bonfim (2014), programar em par consiste em dois membros da equipe programar juntos, em turnos; ou seja, quem estava a produzir o código passa a ser a pessoa que revisa o código, isso em uma sequência definida pela equipe. Desta forma, os dois desenvolvedores assumem a responsabilidade e o interesse em produzir o código.

Na opinião de Bonfim (2014), a programação em par pode ser denominada como “uma mente – quatro mãos”, pois os programadores instantaneamente entendem o código que está sendo escrito sem necessidade de explicação, ou seja, um desenvolvedor escreve o código enquanto o outro o revisa. Essa técnica exige que ambos os programadores possuam conhecimentos similares.

Para Teles (2006), a programação em par é uma forma eficiente de reduzir a incidência de *bugs* em um sistema. Ambos os programadores estão trabalhando juntos na solução e trocam ideias sobre a solução. Quem trabalha frequentemente com programação em par se habitua a corrigir e ter o seu trabalho corrigido dezenas de vezes, o que também é vantagem, pois a quantidade de erros identificados costuma ser elevada depois de um período de experimentação. Isso denota a confiabilidade da programação em par, que leva muitos programadores a se sentirem inseguros ao programar sozinhos (TELES, 2006).

Teles (2006) destaca que há pesquisas que demonstram este fato, positivo em favor da utilização da programação em par. Nesse sentido, a programação é a incorporação da técnica de inspeção de código a tudo o que é produzido no projeto, o tempo todo.

Em relação ao ensino-aprendizagem no curso superior, essas abordagens se complementam na tentativa de responder aos desafios do processo; a prática pedagógica constitui uma das categorias fundamentais da atividade humana, rica em valores e significados, pois a questão metodológica se torna, muitas vezes, tão essencial quanto o conhecimento (LEAL, 2004 apud GOMES *et al.*, 2006).

A programação pareada é pouco praticada em meio acadêmico; em pesquisas realizadas nos últimos anos, observa-se preocupação crescente com o processo de ensino/aprendizagem de programação. Essa dificuldade, nos cursos da área da computação, acontece por diversos fatores, como a exigência lógico-matemática predominante na disciplina, ou pela dificuldade de apreensão, por parte do professor, do ritmo de aprendizagem de cada aluno, entre outros (BONFIM, 2014).

Independentemente da metodologia e do paradigma utilizado, a lógica de programação é requisito nos cursos de computação, importante na estruturação do raciocínio lógico e na formulação de algoritmos. A condução desse estudo é vista com preocupação pelos alunos, que sentem resistência ante o conteúdo, de natureza lógico-matemática, dado o enraizado estigma relacionado a ele (BONFIM, 2014).

A inserção da programação pareada no ensino aprendizagem pode minimizar a dificuldade encontrada pelos acadêmicos na resolução de seus algoritmos. A troca de conhecimento entre as equipes contribui para uma implementação mais simples e eficaz.

Segundo Teles (2006), a variável mais relatada pelos estudos foi a qualidade do código, pois os alunos que se habituem a programar em par realizam atividades mais complexas, com maior qualidade.

### **3 Metodologia**

Este artigo apresenta uma revisão bibliográfica de trabalhos publicados no âmbito da educação, computação e tecnologia, que abordam a utilização de metodologias ágeis em projetos de desenvolvimento de software. Busca destacar fatores que contribuem para a adoção da metodologia ágil baseada em programação pareada no processo ensino-aprendizagem da programação orientada a objetos, voltada para o desenvolvimento mobile, bem como seus principais benefícios e desafios, a fim de comprovar a sua efetividade como ferramenta.

A estratégia de pesquisa adotada baseou-se em conteúdos educacionais, objetivando facilitar o ensino-aprendizagem. Foram considerados como principais fontes de pesquisa Google Acadêmico, livros, base de dados de teses e trabalhos de conclusão de curso de universidades e outras fontes especializadas na área.

### **4 Considerações finais**

Devido ao crescimento da tecnologia em computação, torna-se necessário o desenvolvimento de sistemas de softwares complexos. Para que o sistema seja viável, há uma preocupação por parte dos engenheiros de software sobre o desenvolvimento destes produtos dentro do prazo e em perfeito funcionamento. A engenharia de software dentro da computação dedica-se à concepção, especificação, desenvolvimento e manutenção de sistemas de software, aplicando tecnologias, práticas de gerência de projeto e desenvolvimento de boas práticas por meio da orientação a objetos.

Dentro desta perspectiva, as empresas de desenvolvimento de software estão mais competitivas, evoluem com a utilização de métodos ágeis, sem perder qualidade e com maior rapidez no desenvolvimento de seus produtos. Para que isso ocorra, há um processo de análise e levantamento de requisitos para agregar qualidade ao software, que permite adquirir amplo conhecimento sobre desenvolvimento de códigos orientados a objeto, além do uso da metodologia ágil *Extreme Programming* (XP), método que tem como peculiaridade e objetivo tornar os processos de desenvolvimento mais ágeis e eficientes.

Também foi possível observar que os métodos ágeis de desenvolvimento têm sido utilizados pelos engenheiros de software com o objetivo de fornecer resultados mais rápidos e

de maior valor para o negócio do cliente, promovendo uma comunicação facilitada, respostas rápidas às mudanças, entre outros benefícios — como a produção de código enxuto, compartilhamento do conhecimento, correção de falhas, facilidade de manutenção, confiança, amadurecimento, divisão do trabalho e agilidade, entre outros.

No que se refere ao processo de ensino-aprendizagem, a pesquisa também apresenta os benefícios da programação pareada, a qual pode ser utilizada de forma efetiva como ferramenta pedagógica, colaborando em variáveis como o desempenho, confiança e motivação dos alunos. Também foi possível observar que a programação em par surge como uma importante ferramenta neste processo, especialmente aquelas orientadas a objetos. Esta prática possui diversos benefícios, entre eles o compartilhamento de informações, o aumento da qualidade do produto, a facilidade de aprendizado e a melhora na qualidade do código, comprovando a sua efetividade.

Por fim, a pesquisa também aponta algumas dificuldades levantadas durante a revisão de literatura, como atritos devido à personalidade dos desenvolvedores envolvidos, entendimento do código, falta de entrosamento, críticas e substituições de desenvolvedores — nada que comprometa sua eficácia.

Portanto, com base nas informações levantadas durante esta pesquisa, sugere-se como possibilidade de futuros estudos, o desenvolvimento de metodologias práticas a serem inseridas no ensino-aprendizagem de orientação a objetos, utilizando como base a programação pareada.

## Referências

AURELIANO, V.C. O.; TEDESCO, P. C. A. R. Ensino-aprendizagem de Programação para Iniciantes: uma revisão sistemática da literatura focada no SBIE e WIE. *In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO*, 23., 2012, Rio de Janeiro. **Anais** [...]. Rio de Janeiro: UFRJ, UNIRIO, UNERJ, 2012. Disponível em: <http://www.br-ie.org/pub/index.php/sbie/article/view/1718/1479>. Acesso em: 08 ago. 2021.

BEZERRA JÚNIOR, E. V. **Análise da prática docente no processo de ensino de programação orientada a objeto mediado por meio de rede social educativa**. 2013. 141 f. Dissertação (Mestrado em Ciência da Computação) - Universidade Federal de Pernambuco, Recife, 2013. Disponível em: <https://repositorio.ufpe.br/bitstream/123456789/11958/1/Dissertacao%20Elias%20Bezerra%20Jr.pdf>. Acesso em: 14 dez. 2021.

BREVE, F. A. Engenharia para a Web. **Fabício Breve**, São Carlos, jun. 2002. Disponível em: [http://www.fabriciobreve.com/trabalhos/engenharia\\_web.php#UZwNGrW39cc](http://www.fabriciobreve.com/trabalhos/engenharia_web.php#UZwNGrW39cc). Acesso em: 07 out. 2021.

BONFIM, M. Programação em par: vantagens e desvantagens. **DevMedia**, São Paulo, 2014. Disponível em: <https://www.devmedia.com.br/pair-programming-vantagens-e-desvantagens-da-programacao-em-par/30537>. Acesso em: 23 out. 2021.

CASTELHANO, Ricardo. **Programação Pareada 2018**. Disponível em: <http://ricardocastelhana.net/programacao-pareada/>. Acesso em: 08 ago. 2021.

FIGUEIREDO, J. C. B. Estudo da difusão da tecnologia móvel celular no Brasil: uma abordagem com o uso de dinâmica de sistemas. **Production**, São Paulo, v. 19, n. 1, p. 230-245, jan./abr. 2009. Disponível em: <http://www.redalyc.org/pdf/3967/396742035014.pdf>. Acesso em: 09 out. 2021.

FEDOCE, R. S. A tecnologia móvel e os potenciais da comunicação na educação. **Logos: Comunicação e Universidade**, Rio de Janeiro, v. 18, n. 2, 2011. Disponível em: <http://www.e-publicacoes.uerj.br/index.php/logos/article/view/2264/2248>. Acesso em: 09 out. 2021.

GOMES, A. M. de A *et al.* Os saberes e o fazer pedagógico: uma integração entre teoria e prática. **Educar em Revista**, Curitiba, v. 28, dez. 2006. Disponível em: <https://www.scielo.br/j/er/a/cGmN7WyL7hP5DzCh6ZWYJCr/?lang=pt>. Acesso em: 09 out. 2021.

KAMADA *et al.* Análise das plataformas de desenvolvimento Mobile aplicados na área educacional, usando Android - Eclipse e Windows Phone. Estudo de caso: Aplicativo Planetas no Windows Phone. **Renote**, Porto Alegre, v. 10, n. 1, jul. 2012. Disponível em: <https://seer.ufrgs.br/renote/article/download/30916/19896>. Acesso em: 31 out. 2021.

KUBO, O. M. Ensino-aprendizagem: uma interação entre dois processos comportamentais. **Interação em Psicologia**, Curitiba, v. 5, 2001. Disponível em: <https://revistas.ufpr.br/psicologia/article/view/3321>. Acesso em: 08 nov. 2021.

MATOS, A. L.; BRITTO, T. C. P. **Engenharia de Web**. 2007. Disponível em: <http://pt.scribd.com/doc/83425694/Engenharia-Web>. Acesso em: 11 out. 2021.

MEDEIROS, M. P. **Implementando Pair Programming em sua equipe**: Conhecendo as dificuldades e as vantagens dessa prática XP. 2017. Disponível em: <https://www.devmedia.com.br/implementando-pair-programming-em-sua-equipe/1694>. Acesso em: 24 ago. 2021.

OLIVEIRA, Jorge Martins; AMARAL, Júlio Rocha. O pensamento abstrato. **Cérebro e Mente**, Campinas, n. 12, c2001. Disponível em: <http://www.cerebromente.org.br/n12/opiniao/pensamento.html>. Acesso em: 14 dez. 2021.

OLIVEIRA, Artur Elias Hayne. Introdução à orientação a objetos. **Revista SQL Magazine**, Rio de Janeiro, n. 89, p. 68-74, 2011.

PIRES, J. O que é Programação Orientada a Objetos e por que você precisa saber. **Becode**, [s. l.], 2016. Disponível em: <https://becode.com.br/programacao-orientada-a-objetos-poo/>. Acesso em: 14 dez. 2021.

PRESSMAN, R. S. **Engenharia de software**. 6. ed. Rio de Janeiro: McGraw-Hill, 2006. 720 p.

SANTI, R.; BEGOSSO, L. R. **Metodologias ágeis para o desenvolvimento de software**. 2016. 14 f. Monografia (Iniciação Científica em Análise e Desenvolvimento de Sistemas) - Fundação Educacional do Município de Assis, Conselho Nacional de Desenvolvimento Científico e Tecnológico, Assis - SP, 2016. Disponível em: <https://cepein.femanet.com.br/BDigital/arqPIBIT/1511320798T677.pdf>. Acesso em: 31 out. 2021.

SOMMERVILLE, I. **Engenharia de software**. 6. ed. São Paulo: Pearson Prentice Hall, 2004.

TELES, V. M. **Extreme Programming**: Aprenda como encantar seus usuários desenvolvendo software com agilidade e alta qualidade. São Paulo: Novatec, 2006

VIANNA, J. T. A. **Introdução à orientação a objetos**. 2013. Disponível em: <http://www.ufjf.br/peteletrica/files/2012/10/Curso-Orienta%C3%A7%C3%A3o-a-objetos.pdf>. Acesso em: 23 out. 2021.

WEBER, F. A importância da sua empresa ter um site. **Administração.com**, [s. l.], 20 fev. 2008. Disponível em: <http://www.administradores.com.br/artigos/marketing/a-importancia-da-sua-empresa-ter-um-site/21267/>. Acesso em: 11 out. 2021.