

A IMPORTÂNCIA DO DESIGN CENTRADO NO USUÁRIO EM METODOLOGIAS ÁGEIS COMO REQUISITO DE USABILIDADE

THE IMPORTANCE OF USER-CENTERED DESIGN IN AGILE METHODOLOGIES AS A USABILITY REQUIREMENT

Pedro Manoel de Azevedo – pedro87azevedo@gmail.com
Faculdade de Tecnologia (Fatec) – Taquaritinga – SP – Brasil

Daniela Gibertoni – daniela.gibertoni@fatectq.edu.br
Faculdade de Tecnologia (Fatec) – Taquaritinga – SP – Brasil

DOI: 10.31510/infa.v17i2.986

Data de publicação: 18/12/2020

RESUMO

Em virtude da competitividade no mercado de desenvolvimento de software, ao longo das últimas décadas, os métodos de desenvolvimento de software vêm a cada nova geração se especializando e tornando os processos cada vez mais eficientes. Entretanto, nenhuma metodologia foca explicitamente no Design Centrado no Usuário, o que implica em insatisfação do cliente pela falta de adequação as suas necessidades, retrabalho, deficiência de requisitos que atendam ao projeto e outros problemas, que acabam afetando os custos e cronograma das equipes de desenvolvimento. Em decorrência dessas vulnerabilidades, o presente artigo tem como finalidade, correlacionar os modelos ágeis com técnicas de usabilidade e Design Centrado no usuário para gerar satisfação aos clientes. A revisão bibliográfica sistemática abordada neste artigo propõe-se a abordar artigos e teses acadêmicas referentes ao Design Centrado no Usuário, Metodologias Ágeis e Usabilidade, fazendo a convergência dos benefícios dessas metodologias, assim como a correlação entre elas. Como resultados foram encontradas vulnerabilidades nas metodologias ágeis em relação a comunicação entre usuários e as partes interessadas do projeto e que podem ser sanadas com a utilização de uma eficiente análise de requisitos aliado ao uso de outras ferramentas direcionadas ao design centrado no usuário.

Palavras-chave: Design Centrado no Usuário. Desenvolvimento Ágil de Processo de Software. Integração Ágil de Design Centrado no Usuário. Usabilidade.

ABSTRACT

Due to the competitiveness in the software development market, over the last few decades, software development methods have been specializing in each new generation and making processes increasingly efficient. However, no methodology explicitly focuses on User-Centered Design, which implies customer dissatisfaction due to the lack of adaptation to their needs, rework, deficiency of requirements that meet the project and other problems, which

end up affecting the costs and schedule of the development teams. . As a result of these vulnerabilities, this article aims to correlate agile models with usability techniques and User-Centered Design to generate customer satisfaction. The systematic bibliographic review addressed in this article proposes to address articles and academic theses referring to User-Centered Design, Agile Methodologies and Usability, converging the benefits of these methodologies, as well as the correlation between them. As a result, vulnerabilities were found in agile methodologies in relation to communication between users and project stakeholders, which can be remedied with the use of an efficient requirements analysis combined with the use of other tools aimed at user-centered design.

Keywords: User-Centered Design, Agile Software Process Development, Agile User-Centered Design Integration, Usability.

1 INTRODUÇÃO

O Design Centrado no Usuário (DCU) realça a importância que o usuário/cliente tem nos processos de desenvolvimento de software. Segundo Norman e Drapper (1986), o princípio do DCU coloca o usuário no centro das soluções de design, através das iterações que ocorrerão em cada fase do projeto.

Essas iterações podem ser vistas nas metodologias ágeis segundo o Agile Manifest (2015), entretanto os modelos ágeis não explicitam o usuário como entidade principal do desenvolvimento. Logo, o presente estudo visa por meio de uma revisão bibliográfica sistemática, elaborar uma pesquisa científica especializada através da utilização de livros, artigos e teses acadêmicas referentes ao Design Centrado no Usuário, Metodologias Ágeis e Usabilidade. Dessa forma, objetiva-se realizar a intersecção dos benefícios das metodologias, assim como conectar a abordagem e os pontos de sinergia entre o DCU e as Metodologias Ágeis, centralizando o desenvolvimento em torno do usuário e aliando as competências das duas metodologias com princípios de usabilidade para sanar as vulnerabilidades do modelo ágil e assim canalizar o projeto a partir das necessidades do usuário para garantir satisfação, eficácia e eficiência aos sistemas que serão desenvolvidos.

Projetos que não são centrados no usuário podem ter deficiências de usabilidade e de experiência do usuário, o que acaba gerando erros no projeto, reclamações dos usuários, deficiência ergonômica, entre inúmeras vulnerabilidades. Logo, esse estudo tem amplo valor bibliográfico para o tema correspondente.

Portanto, este artigo possui cinco sessões, sendo a primeira delas a introdução. A segunda sessão contém toda a fundamentação teórica que embasa este trabalho, tais como Usabilidade, DCU e Metodologias Ágeis (Scrum). A terceira seção compreende os Procedimentos Metodológicos que detalham o processo de pesquisa realizado a partir da delimitação do problema, os objetivos da pesquisa, a metodologia da pesquisa e a justificativa. Na quarta seção são apresentados os resultados da pesquisa através de uma revisão bibliográfica sistêmica que, por meio da comparação de metodologias, foi possível encontrar as falhas dos modelos estudados e apontar uma solução adequada ao problema. Por fim, a quinta seção compreende a conclusão deste estudo e os argumentos pertinentes ao aprofundamento.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Usabilidade

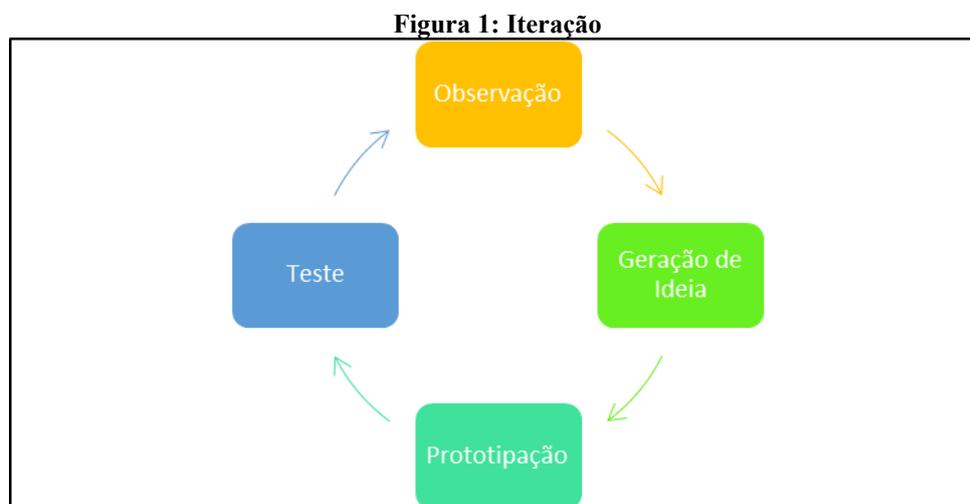
A Interação Humano Computador (IHC) é oriunda das teorias de Ergonomia e conceitua o relacionamento entre homens e computadores a fim de estabelecer um vínculo no desenvolvimento de interfaces computacionais centradas no usuário, alcançando sua plena usabilidade (REIS, 2007). A usabilidade na área da computação refere-se à simplicidade e facilidade com que uma interface de um programa de computador ou um website pode ser utilizado. De acordo com as normas ISO 9241-11 (2018), a usabilidade é definida como a potencialidade de um produto ou software atender seus usuários com eficácia, eficiência e satisfação, sendo usados em sistemas existentes (FILARDI, 2007). Segundo Preece, Rogers e Sharp (2005), os princípios da usabilidade são nomeados como heurística, que são modelos normativos que visam tomar objetivando a racionalidade progressiva de um dado problema. Para Nielsen (1986) *apud* Preece, Rogers e Sharp (2005), as técnicas de avaliação de interface que norteiam a usabilidade devem estar presentes no desenvolvimento de software, pois devem atender o cliente através da visibilidade do sistema, compatibilidade, controle (liberdade do usuário), padronização, diagnóstico de erros, prevenção dos erros, eficiência de uso, estética e documentação.

2.2 Design Centrado no Usuário

Para Ritter, Baxter e Churchill (2014), o Design Centrado no Usuário (DCU) foi originado na década de 1980, através das pesquisas de Donald A. Norman na Universidade da Califórnia, onde foi feita a publicação do livro “Design de Sistema Centrado no Usuário: Novas Perspectivas sobre Interação Homem-Computador”.

O DCU consiste em um conjunto de técnicas, métodos, procedimentos e processos, sendo uma filosofia que coloca o usuário no centro do processo de desenvolvimento de uma forma intensa, conveniente e sistemática. O intuito da utilização do DCU é uma busca da aproximação para satisfazer os usuários por meio da elaboração utilizável de produtos perceptíveis que atendam às suas necessidades e interesses, além de seus objetivos, contexto de uso, habilidades e limitações. A usabilidade de um produto é resultante do trabalho pautado no DCU, que decorre ao longo do processo de desenvolvimento e continua mesmo após o lançamento do produto, a fim de melhorar versões posteriores (SALAH; PAIGE; CAIRNS, 2014).

Segundo Norman (1986), o DCU concentra em torno do usuário as suas necessidades para que resoluções de design sejam realizadas. Ele é realizado por meio de um processo iterativo constantemente definido em 4 fases, tal como é apresentado pela Figura 1 a seguir.



Fonte: Norman (1986), adaptado pelo autor

Na fase da observação, procura-se compreender o contexto de uso do produto por meio de entrevistas, enquetes, observação etc. Para a etapa de geração de ideia, procura-se ampliar o número de soluções, frequentemente realizando um *brainstorming* com o propósito

de chegar a um conceito. Na prototipação, procura-se elaborar um protótipo rápido e barato do conceito construído na fase anterior para ser testado e avaliado durante a etapa de teste. Geralmente, isso é realizado através da observação do uso do protótipo por algum possível usuário (NORMAN, 1986). Ocasionalmente, o DCU pode ser explicitado por outro conjunto de fases. Entretanto, ele ainda expressa as mesmas atividades aglutinadas de outras maneiras (NORMAN, 1986).

Na visão de Abras, Maloney-Krichmar e Preece (2004), a abordagem do design centrado no usuário é enfática na perspectiva de visão do usuário, visando identificar e aproximar o mesmo do processo de design desde o início, sendo uma percepção mais acentuada dos fatores psicológicos, organizacionais, sociais e ergonômicos que influenciam o uso da tecnologia da informática na participação do usuário em todas as fases do design e avaliação do produto. O engajamento dos usuários permite que o produto seja relevante para a finalidade a que é destinado no ambiente em que será usado. Essa perspectiva leva ao desenvolvimento de produtos mais eficazes, eficientes e seguros.

No desenvolvimento de um novo produto, o DCU (Desenvolvimento Centrado no Usuário) ajuda os designers a coordenarem as expectativas do usuário. Portanto, os usuários sabem desde o início o que esperar de um produto quando eles se envolvem na criação do seu design e sentem que suas ideias e sugestões foram levadas em consideração durante esse processo. Isso leva a um senso de participação no produto, o que garante uma maior satisfação do cliente e maior integração do produto no ambiente. Portanto, o designer tem a missão de facilitar a tarefa para o usuário e assegurar que ele possa usar o produto conforme o planejado e tendo o menor esforço possível para aprender a usá-lo (ABRAS; MALONEY-KRICHMAR; PREECE, 2004).

2.3 Design Centrado no Usuário X Usabilidade

A norma ISO 13407 introduz diretrizes sobre como abranger a usabilidade apresentando o design centrado no usuário às atividades no decorrer do ciclo de vida de sistemas interativos baseados em computador. O DCU é descrito como uma atividade multidisciplinar (BEVAN, 2003).

A norma, segundo Bevan (2003), lista quatro atividades de design centradas no usuário que precisam começar nos primórdios do projeto. Elas são:

- Compreender e especificar o contexto de uso;
- Especificar os requisitos do usuário e da organização;
- Produzir soluções de design;
- Avaliar projetos em relação aos requisitos.

O processo requer iterar até que os objetivos sejam satisfeitos. A norma ISO 13407 descreve os preceitos básicos, embora não estipule métodos específicos. A sequência em que as atividades são realizadas e o nível de esforço e os detalhes apropriados variam dependendo do ambiente de design e o estágio do processo de design (BEVAN, 2003).

2.4 Metodologias Ágeis

O modelo de desenvolvimento Ágil teve origem em 2001, quando 17 especialistas em desenvolvimento de software discutiam e buscavam melhorias nas metodologias de produção de software que serviriam como alternativa para os modelos tradicionais de desenvolvimento (AGILE MANIFEST, 2015).

Foi a partir desta ruptura que se difundiu o Manifesto Ágil, que tem como paradigma 4 preceitos básicos (BECK et al., 2001):

- Indivíduos e iterações são mais importantes do que processos e ferramentas;
- Software funcionando é mais importante do que documentação completa;
- Colaboração com o cliente é mais importante do que negociação de contratos;
- Adaptação a mudanças é mais importante do que seguir o plano inicial.

As iterações e as pessoas são essenciais no desenvolvimento ágil, pois há um conjunto sequencial de tarefas nas quais são feitos pequenos incrementos a fim de tornar o processo mais eficiente e que se repetem ao longo do mesmo. Também traz melhorias e novas funcionalidades ao produto desenvolvido. A cada Iteração, a equipe é capaz de reavaliar as prioridades do projeto, o que torna o mesmo simples e altamente customizável. As pessoas se organizam e se auto gerenciam, sendo o modelo ágil concebido para indivíduos proativos, coesos e que utilizam da comunicação em todo momento para gerir as equipes, entregando um software testado e em pleno funcionamento exigindo o mínimo de documentação. Isso

contrapõe todos os modelos de desenvolvimento tradicionais que eram puramente focados em processos (FIGUEIREDO, 2019).

Durante o processo de desenvolvimento recomenda-se que os clientes façam parte das equipes envolvidas no projeto, para que haja uma interação diária e, conseqüentemente, se alcance uma excelência na qualidade ao invés de apenas depender dos termos contratuais. As metodologias ágeis priorizam a flexibilidade na aceitação de alterações nos requisitos do projeto, além de darem autonomia às equipes para gerir planos flexíveis que acomodem essas mudanças, descartando assim seguir estritamente o planejamento prévio (FIGUEIREDO, 2019).

Em decorrência dos valores citados acima, o Agile Manifest (2015) menciona que existem 12 princípios que servem como diretrizes para as equipes de desenvolvimento ágil. Estes princípios devem ser compreendidos, já que há uma importante integração deles com o DCU que tem pontos de sinergia e algum conflito.

Portanto, a metodologia ágil enfatiza o conceito de iterações proposto por Boehm (2002), evidenciando uma perspectiva mais flexível, a relevância das alterações para o negócio, a criação de valor para o cliente e a interação entre os envolvidos no projeto

3 PROCEDIMENTOS METODOLÓGICOS

Neste projeto, foi feita uma revisão bibliográfica sistemática a respeito do tema abordado, utilizando de pesquisa científica especializada através de livros, artigos e teses acadêmicas referentes ao Design Centrado no Usuário, Metodologias Ágeis e Usabilidade, fazendo intersecção dos benefícios dessas metodologias, assim como a complementariedade entre elas.

A revisão bibliográfica tem enorme peso para comunidade científica, sendo considerada um passo inicial para qualquer pesquisa, na qual exige-se certo ineditismo e originalidade. Uma revisão sistêmica e rigorosa agrega para a evolução e difusão do conhecimento, desenvolvendo a produção de teorias em áreas do conhecimento já existentes. Isso permite que outros pesquisadores utilizem os resultados destas pesquisas com maior confiabilidade, possibilitando reutilizar estudos já concretizados. Os resultados de uma revisão sistemática possibilitam identificar “brechas” na teoria que podem ser investigadas por outros pesquisadores, mas que não foram descobertas em estudos de áreas similares

devido abordagem superficial e falta de rigor na revisão bibliográfica (CONFORTO; AMARAL, 2011).

Para obter os resultados da pesquisa foi utilizada a base de dados *Association for Computing Machinery* (ACM), utilizando a parte de *Journals*. Os termos pesquisados foi “*integration of agile methods and user-centered design*”, que retornou 53.108 artigos. Após isso, foi feita uma estratificação por data de publicação, utilizando artigos publicados nos últimos 5 anos. Deste modo, o buscador retornou 17.490 artigos dos quais foram selecionados 16 artigos. Esta seleção foi baseada no resumo e título dos mesmos e buscou convergir com as palavras-chave desta pesquisa. Todavia, apenas 7 foram usados neste estudo, sendo escolhidos devido a integração de metodologias, a base de dados ser renomada em artigos de Engenharia de Software, publicações de autores reconhecidos na temática e apontamento das soluções buscadas nessa pesquisa.

O mesmo processo foi usado para a base de dados Google Scholar para o restante do artigo, que primeiramente retornou 9.910 e depois da estratificação dos dados, mostrou 5.610 artigos, dos quais foram selecionados 24 artigos para leitura baseados no resumo e título dos mesmos. Apenas 13 foram aproveitados neste estudo, seguindo critérios como publicação recente e intersecção entre os temas abordados, ou seja, apresentaram resultados envolvendo a sinergia entre modelos ágeis (Scrum), DCU e usabilidade. Outros estudos foram preteridos por se tratar de apenas um tipo de abordagem e deixarem de lado as outras soluções buscadas neste estudo. O Quadro 1 detalha todo processo de busca dos artigos descrito anteriormente e o aproveitamento dos que foram utilizados nesta pesquisa.

Quadro 1: Estratificação dos artigos

Base de Dados	ACM	Google Scholar
Total de artigos encontrados	53.108	9.910
Total de artigos após estratificação	17.490	5.610
Artigos Lidos	16	24
Artigos usados no estudo	7	13
Aproveitamento	0,013181%	0,131181%

Fonte: Elaborado pelo autor (2020)

4 RESULTADOS E DISCUSSÃO

Para este estudo, foi adotada metodologia Scrum, pois atualmente o cenário de desenvolvimento de software é muito competitivo e as metodologias ágeis como o Scrum prometem benefícios como entrega no prazo e satisfação do cliente, visto que modelos ágeis priorizam as necessidades do cliente, o envolvendo em várias etapas do desenvolvimento, e visam agregar valor ao negócio através de iterações curtas (SALAH; PAIGE; CAIRNS, 2014). Entretanto, existe uma lacuna nos modelos Ágeis como Scrum, Kanban e *Extreme Programming* (EX) que é a carência de definição certa do produto e essa carência deve ser sanada através de uma boa experiência do usuário (UX), aliada às práticas de Design Centradas no Usuário (DCU). A integração com os modelos de Desenvolvimento Ágeis torna o processo de desenvolvimento mais humano. O envolvimento das partes interessadas no projeto e do usuário é fundamental como um fator de sucesso para um sistema obter êxito e, se comparado com as abordagens tradicionais, esse envolvimento não se limita às fases iniciais do desenvolvimento, já que as partes interessadas e o usuário estão envolvidos em todo o processo de desenvolvimento (SCHÖN; THOMASCHEWSKI; ESCALONA, 2017).

O DCU procura sanar aquilo que muitas vezes, os manuais que acompanham os produtos são extensos, de difícil compreensão e não são centrados no usuário. Assim, o ideal seria a utilização de pequenos panfletos que fossem lidos rapidamente com base na visão do usuário (ABRAS; MALONEY-KRICHMAR; PREECE, 2004).

Conforme Salah, Paige e Cairns, (2014), ao longo do processo de desenvolvimento de software, a engenharia apresenta certas vulnerabilidades e desafios como a falta de tempo para atividades iniciais, dificuldade de gerir o grupo de trabalho, comunicação entre a equipe de desenvolvimento e praticantes de DCU, condução de testes de usabilidade, ausência de praticantes de DCU ou seu aumento carga de trabalho em caso, falta de documentação, de falta de suporte de gestão às atividades de DCU. Essa má gestão impacta diretamente no gerenciamento de recursos, pois as empresas não direcionam um engenheiro especialista em usabilidade para conduzir esses processos centrados no usuário. Logo, isso cria toda uma vulnerabilidade ao projeto.

A solução encontra-se nos requisitos, que são a base de todos os produtos de software e, conseqüentemente, a Engenharia de Requisitos (ER) desempenha um papel importante no

desenvolvimento do sistema. Nas metodologias Ágeis, a ER é uma ferramenta de convergência entre modelos ágeis e DCU, pois o documento de requisitos é uma declaração formal dos requisitos para os *stakeholders*, usuários e clientes, no qual é identificado o escopo do projeto e todas as necessidades do cliente são especificadas. Em comparação com as abordagens tradicionais de ER, uma lista de requisitos priorizados é usada e tem como principais atividades de ER: a elicitación (identificação das necessidades dos *stakeholders*); documentação (os requisitos são documentados em alto nível de detalhe para atender a satisfação dos *stakeholders*); validação (valida os requisitos do sistema sem inconsistências ou ambiguidades, sendo todos os erros sanados e corrigidos); negociação (explora e classifica os requisitos conforme as necessidades dos *stakeholders*) e gestão (LOSADA, 2018).

A ER é realizada de forma iterativa durante todo o processo de desenvolvimento ao invés de ser durante uma fase fechada no início. Para esse fim, um modelo *Just-in-time* é frequentemente usado para refinar requisitos de alto nível em tarefas de baixo nível que podem ser implementadas por desenvolvedores, empresários, *stakeholders*, usuários e desenvolvedores trabalham juntos de forma colaborativa. Todo fluxo de trabalho pode ser gerenciado por um Kanban através de *Story boards* para design, desenvolvimento e entregas (SCHÖN; THOMASCHEWSKI; ESCALONA, 2017).

Observam-se problemas relacionados à documentação de requisitos nos quais existiam dificuldades para identificar o tipo certo de artefatos, o que aumenta a colaboração entre as partes interessadas, o desenvolvedor e a equipe ágil. Este é um desafio especial no que diz respeito à configuração do projeto (por exemplo, equipes colocadas ou distribuídas). No entanto, é importante criar diretrizes adequadas para o gerenciamento de requisitos para o desenvolvimento de software ágil (SALAH, PAIGE E CAIRNS, 2014).

Dentre os vários estudos analisados, várias funções adicionais devem ser adotadas a um ambiente ágil, a fim de abordar uma deficiência de comunicação ou assumir atribuições por tarefas específicas. Logo, o papel de um especialista em metodologias ágeis voltadas ao DCU deve sanar as falhas de comunicação entre o desenvolvedor e o designer de UX, usando critérios de usabilidade para realizar as especificações de requisitos. Assim, os benefícios dos ciclos de *feedback* contínuos é que os mal-entendidos são detectados a tempo e as alterações podem ser aplicadas antes que se transformem em problemas maiores. Também foi possível observar que os clientes estavam diretamente envolvidos em cada iteração, sendo que os

requisitos foram identificados, refinados e validados por meio de comunicação face a face com o cliente (BRUUN et al, 2018).

Já em relação ao DCU, propõem-se diferentes métodos seguindo as atividades da ISO 9241-210 como princípios norteadores de usabilidade, realizando iterações que envolvem; compreensão e especificação do contexto de uso; especificar os requisitos do usuário; produzir soluções de design e avaliar os designs em relação aos requisitos, sempre explorando o contexto de uso. Além disso, também é proposto fazer pesquisas, entrevistas, discussões, grupos de foco, análise de concorrentes e jornadas do usuário para especificar os requisitos do usuário, com o intuito de coletar dados para a criação de soluções de design e orientação do usuário para se fazerem os testes de usabilidade.

Foram observadas vulnerabilidades relacionadas ao envolvimento direto dos usuários e partes interessadas. No entanto, identificaram-se quatro metodologias (DCU, Design Thinking, Investigação Contextual e Design Participativo) que foram integradas ao desenvolvimento de software ágil, com o objetivo de aumentar o conhecimento sobre as necessidades do usuário e, conseqüentemente, integrá-las de forma mais eficiente para gerar satisfação ao cliente.

5 CONCLUSÃO

Esta análise de revisão da literatura contempla uma priorização de uso do Design Centrado no Usuário como ferramenta para complementar as vulnerabilidades das metodologias Ágeis, trazendo benefícios para as empresas desenvolvedoras, pesquisadores e usuários. Os resultados apresentados evidenciaram que o tema deve ter maior abrangência e continuidade em futuras pesquisas, uma vez que outros modelos devem ser apresentados. O tema encontra-se em constante evolução devido as novas tecnologias, o que gera novas experiências ao usuário e novos conceitos de usabilidade.

Assim como as iterações da metodologia ágil, esta temática encontra-se em constante renovação, o que torna necessária a investida em novas pesquisas usando, com o intuito de melhorar a usabilidade do usuário e sua experiência, trazendo produtos de melhor qualidade ao mercado de desenvolvimento de software. Para as pesquisas, novas bases de dados serão utilizadas e assim os resultados serão confrontados, por ser uma pesquisa introdutória ainda há um vasto horizonte a ser percorrido em futuras publicações.

REFERÊNCIAS

ABRAS, C.; MALONEY-KRICHMAR, D.; PREECE, J. **User-Centered Design**. In Bainbridge, W. Encyclopedia of Human-Computer Interaction. Thousand Oaks: Sage Publications, 2004.

AGILE ALLIANCE. **Manifesto For Agile. Software Development**. 2001. Disponível em <<https://www.agilealliance.org/agilealliancebrazil/>>. Acesso em: setembro, 2020.

BECK, K. et al. **Os doze princípios do software ágil**. [S.l:s.n], 2011. Disponível em: <<http://www.manifestoagil.com.br/principios.html>>. Acesso em: 01 setembro de 2020.

BEVAN, N. **UsabilityNet Methods for User Centred Design. Human-Computer Interaction: Theory and Practice (Part 1)**, Volume 1 of the Proceedings of HCI International 2003.

BOEHM, B. **Get ready for agile methods, care**. Computer, IEEE, v.35, n. 1, p. 64-69, 2002.

BRUUN, A., LARUSDOTTIR. M. K., NIELSEN, L., NIELSEN, P. A., PERSSON, J.S. **The Role of UX Professionals in Agile Development: A Case Study From Industry**. Setembro, 2018.

CONFORTO, E. C.; AMARAL, D.C.; SILVA, S.L. **Roteiro para Revisão Bibliográfica Sistemática: aplicação no desenvolvimento de produtos e gerenciamento de projetos**. In: 8o. Congresso Brasileiro de Gestão de Desenvolvimento de Produto - CBGDP 2011, 2011, Porto Alegre-RS. 8o. Congresso Brasileiro de Gestão de Desenvolvimento de Produto - CBGDP 2011. Porto Alegre: Instituto de Gestão de Desenvolvimento de Produto, 2011.

FIGUEIREDO, L. B. **UP-FRONT DESIGN NO DESENVOLVIMENTO ÁGIL DE SOFTWARE CENTRADO NO USUÁRIO**. 2019, 166f. Dissertação Escola Politécnica da Universidade de São Paulo em Engenharia de Produção, São Paulo, Brasil.

FILARDI, A. L. **Análise e avaliação de técnicas de interação humano-computador para sistemas de recuperação de imagens por conteúdo baseadas em um estudo de caso**. 2007, 149. Mestrado em Ciências da Computação e Matemática Computacional, , Universidade de São Paulo, USP, Brasil.

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. **ISO 9241-11: Ergonomics of human-system interaction — Part 11: Usability: Definitions and concepts**, 2018. Disponível em: <[https://www.iso.org/standard/63500.html#:~:text=ISO%209241%2D11%3A2018%20provides,services%20\(including%20technical%20and%20personal](https://www.iso.org/standard/63500.html#:~:text=ISO%209241%2D11%3A2018%20provides,services%20(including%20technical%20and%20personal)> Acesso em 28/08/2020.

LOSADA, B. **Flexible Requirement Development through User Objectives in an Agile-UCD Hybrid Approach**. Department of Languages and Computer Systems University of the Basque Country Spain, INTERACCIÓN 2018, setembro 2018.

NORMAN, D. A., DRAPER S. W. **User Centered System Design: New Perspectives on Human-Computer Interaction.** New Jersey: Lawrence Erlbaum Associates, 1986.

PREECE, J. ROGERS, Y. SHARP, H. **Design de Interação: além da interação homem computador.** Porto Alegre: Bookman. 2005.

REIS, G.. **Centrando a Arquitetura de Informação no Usuário.** São Paulo, 2007, 250f. Mestrado em Ciência da Informação, Universidade de São Paulo, USP, Brasil.

SALAH, D.; PAIGE, R. F.; CAIRNS, P. **A systematic literature review for agile development processes and user centred design integration.** In: ACM. Proceedings of the 18th international conference on evaluation and assessment in software engineering. p.5 2014.

SALAH, D.; PAIGE, R.; CAIRNS, P. **A practitioner perspective on integrating agile and user centred design.** In: BCS. Proceedings of the 28th International BCS Human Computer Interaction Conference on HCI 2014-Sand, Sea and Sky-Holiday HCI. p.100–109, 2014.

SALAH, D.; PAIGE, R.; CAIRNS, P. **Integrating Agile Development Processes and User Centred Design- A Place for Usability Maturity Models?.** 5th International Conference on Human-Centred Software Engineering (HCSE), Paderborn, Germany. pp.108-125, 2014.

SCHÖN, E.M., THOMASCHEWSKI, J., ESCALONA, M.J.. **Agile Requirements Engineering: A systematic literature review.** Computer Standards & Interfaces 49 p.79–91, 2017.