



**UNIVERSIDADE
FEDERAL RURAL
DE PERNAMBUCO**

**Pró-Reitoria de Ensino de Graduação
Departamento de Estatística e Informática (DEINFO)
Bacharelado em Ciência da Computação**

Thomás André da Silva Leal

Retrospectiva: Apoiando o Aprendizado em Times Ágeis

Trabalho de Conclusão de Curso

**Recife
2018**

Thomás André da Silva Leal

Retrospectiva: Apoiando o Aprendizado em Times Ágeis

Trabalho de Conclusão de Curso apresentado ao Curso de Ciência da Computação da Universidade Federal Rural de Pernambuco, como parte dos requisitos necessários à obtenção do título de Bacharel em Ciência da Computação.

Orientador: Suzana Cândido de Barros Sampaio

**Recife
2018**



MINISTÉRIO DA EDUCAÇÃO E DO DESPORTO
UNIVERSIDADE FEDERAL RURAL DE PERNAMBUCO (UFRPE)
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

<http://www.bcc.ufrpe.br>

FICHA DE APROVAÇÃO DO TRABALHO DE CONCLUSÃO DE CURSO

Trabalho defendido por Thomás André da Silva Leal às 15 horas do dia 06 de fevereiro de 2018, na Sala 31 do CEAGRI-02, como requisito para conclusão do curso de Bacharelado em Ciência da Computação da Universidade Federal Rural de Pernambuco, intitulado **Retrospectiva: Apoiando o Aprendizado em Times Ágeis**, orientado por Suzana Cândido de Barros Sampaio e aprovado pela seguinte banca examinadora:

Suzana Cândido de Barros Sampaio
DEINFO/UFRPE

Marcelo Luiz Monteiro Marinho
DEINFO/UFRPE

Ana Paula Cavalcanti Furtado
DEINFO/UFRPE

Agradecimentos

Primeiramente agradeço aos meus pais Miguel e Ivanilda, por terem me guiado e dado educação durante toda a minha vida. Sem eles eu não estaria no lugar onde me encontro agora.

Agradeço aos meus irmãos Clarissa e Thiago, por sempre estarem ao meu lado e me servirem de exemplos para que eu pudesse atingir todos meus objetivos.

Agradeço a minha orientadora Suzana, que foi fundamental em minha formação e me ajudou com sua sabedoria no desenvolvimento deste trabalho.

Agradeço aos amigos da universidade Daniel, Dennys, Leonardo, Marco, Pedro e Rodrigo que por terem compartilhado momentos felizes e difíceis tornaram minha jornada mais fácil.

Por fim, agradeço aos demais familiares e amigos, que apesar de me ausentar diversas vezes por conta da universidade sempre foram compreensíveis e me deram forças para continuar.

Resumo

As metodologias ágeis seguem uma série de valores e princípios que oferecem diversas vantagens no processo de desenvolvimento de software. A aprendizagem contínua e o compartilhamento de conhecimentos estão entre os princípios ágeis que desempenham uma constante evolução entre os seus praticantes. Apesar dos benefícios, do ponto de vista de gerenciamento de aprendizados existem diversas lacunas nos processos de metodologias ágeis sobre como conseguir atingir estes princípios. Da forma que desenvolvimento de software é uma atividade intensiva em aprendizagem, seu sucesso depende muito do conhecimento e da experiência dos desenvolvedores. Dessa forma, um dos principais objetivos em se dar apoio ao aprendizado é obter uma melhoria na produtividade através do compartilhamento e transferência efetiva do conhecimento. Nesse contexto, este trabalho tem como objetivo reconhecer quais técnicas que dão apoio ao aprendizado são mais eficiente e utilizadas entre os times ágeis e apresenta-las em forma de uma guia, de modo que o aprendizado possa ser: identificado, efetivado, codificado, disseminado e retido. Através de um levantamento bibliográfico e opiniões obtidas mediante questionários foi possível identificar diversas técnicas e realizar a construção do guia. Além disso, este trabalho identificou os principais problemas encontrados pelos times ágeis no gerenciamento de seus aprendizados.

Palavras-chave: Metodologias Ágeis, Aprendizado em Times Ágeis, Gestão de Aprendizados, Compartilhamento de Conhecimento, Gestão de Conhecimentos.

Abstract

Agile methodologies follow a series of values and principles that offer several advantages in the software development process. Continuous learning and knowledge sharing are among the agile principles that constantly improve practitioners skills. Despite the benefits, from the learning management standpoint there are several gaps in the process of agile methodologies on how to achieve these principles. In the way that software development is an intensive learning activity, your success depends heavily on the knowledge and experience of the developers. Thus, one of the main objectives in supporting learning is to obtain an improvement in productivity through the sharing and effective transfer of knowledge. In this context, this work aims to recognize which techniques that gives support learning are more efficient and which techniques are used among agile teams and presents them in the form of a guide, so that learning can be: identified, effective, codified, disseminated and retained. Through a bibliographical survey and opinions obtained through questionnaires, it was possible to identify several techniques and carry out the construction of the guide. In addition, this work identified the main problems encountered by agile teams in managing their learning.

Keywords: Agile Methodologies, Learning in Agile Teams, Learning Management, Knowledge Sharing, Knowledge Management

Lista de ilustrações

Figura 1 – Fluxo da pesquisa	29
Figura 2 – Nível de experiência com desenvolvimento de software	34
Figura 3 – Metodologia ágil usada no processo de desenvolvimento	34
Figura 4 – Nível de experiência com metodologias ágeis	35
Figura 5 – Área de atuação	36
Figura 6 – A importância em usar técnicas e ferramentas de apoio ao aprendizado	37
Figura 7 – O quanto costuma aprender com outros membros do time	37
Figura 8 – O quanto costuma compartilhar aprendizados com outros membros do time	38
Figura 9 – Com que frequência costuma usar técnicas e ferramentas de apoio ao aprendizado	39

Lista de tabelas

Tabela 1 – Valores do Manifesto Ágil	18
Tabela 2 – Princípios do manifesto ágil	18
Tabela 3 – Obras Seleccionadas	23
Tabela 4 – Técnicas e ferramentas de apoio ao aprendizado	40
Tabela 5 – Frequência de uso das técnicas e ferramentas	41
Tabela 6 – Técnicas e ferramentas mais eficientes entre os respondentes	42
Tabela 7 – Pontuação	45
Tabela 8 – Técnicas e ferramentas organizadas por pontuação	45

Sumário

1	Introdução	11
1.1	Problema da Pesquisa	12
1.2	Justificativa	14
1.3	Objetivos	15
1.3.1	Objetivo Geral	15
1.3.2	Objetivos Específico	15
1.4	Estrutura do Trabalho	15
2	Revisão da Literatura	17
2.1	Metodologias Ágeis	17
2.2	Gestão do Conhecimento	21
2.3	Técnicas e ferramentas de Apoio ao Aprendizado em Times Ágeis	23
2.3.1	Apresentações Técnicas, Workshops ou Seminários	23
2.3.2	Coaching, Mentoria ou Treinamentos Formais	24
2.3.3	Comunidade de Prática, Grupos de Interesse ou Fóruns Técnicos	24
2.3.4	Ferramentas (Jira, Trello, Redmine, Google Docs, etc)	24
2.3.5	Forúns de Discussão	24
2.3.6	Pair Programming, Rotação de Pares, Programação em Grupo ou Coding Dojo	25
2.3.7	Quadros Brancos, Quadros de Inovação, Quadros Eletrônicos, Scrum Boards, Kanban Boards	25
2.3.8	Reuniões de Planejamento da Sprint/Projeto, Reuniões Diárias, Reunião de Revisão da Sprint	26
2.3.9	Retrospectivas e Práticas Reflexivas	26
2.3.10	Wiki ou Repositórios de Lições Aprendidas	27
2.4	Considerações Finais	27
3	Metodologia	28
3.1	Método da Pesquisa	28
3.2	Fases da Pesquisa	29
3.2.1	Levantamento Bibliográfico	29
3.2.2	Análise dos Artigos	30
3.2.3	Aplicação do Questionário	30
3.2.4	Análise dos Resultados	30
3.2.5	Construção do Guia	31
3.2.6	Avaliação do Guia	31
3.3	Considerações Finais	31

4	Resultados e Discussão	33
4.1	O Questionário	33
4.2	Perfil dos Respondentes	33
4.3	Importância das Técnicas e Ferramentas de Apoio ao Aprendizado	36
4.4	Aprender e Compartilhar Aprendizados com Outros Membros	37
4.5	Uso das Técnicas e Ferramentas que dão Apoio ao Aprendizado	38
4.6	Técnicas e Ferramentas mais Eficientes	42
4.7	Problemas Encontrados no Gerenciamento de Aprendizados e Recomendações	43
4.8	Análise dos Resultados e Seleção das Técnicas e ferramentas para Montagem do Guia	44
4.9	Considerações Finais	46
5	Desenvolvimento do Guia	48
5.1	Guia com as técnicas e Ferramentas Selecionadas	48
5.1.1	Reuniões de planejamento da sprint/projeto	48
5.1.2	Reuniões diárias	49
5.1.3	Reunião de revisão da sprint	50
5.1.4	Ferramentas (Jira, trello, redmine, google docs, etc)	52
5.1.5	Quadros brancos, quadros de inovação, quadros eletrônicos, etc	53
5.1.6	Retrospectivas e práticas reflexivas	55
5.2	Considerações Finais	56
6	Conclusão	57
6.1	Limitações e Trabalhos Futuros	57
6.2	Dificuldades Encontradas	58
	Referências	59
	APÊNDICES	63
	APÊNDICE A – Protocolo do Questionário	64

1 Introdução

Uma vez que a indústria de software, as tecnologia e as expectativas dos clientes estavam se movendo muito rapidamente e os clientes estavam se tornando cada vez mais capazes de declarar completamente suas necessidades, metodologias e práticas ágeis emergiram como uma tentativa explícita de abraçar mais formalmente taxas maiores de mudanças de requisitos (SHAWKY, 2014). As abordagens tradicionais de desenvolvimento de software passaram a ser consideradas ineficazes em lidar com a complexidade do desenvolvimento (HIGHSMITH, 2013). Esse novo estilo de desenvolvimento ganhou força a partir da união de um grupo de integrantes independentes que ficou conhecido como o Manifesto Ágil (MELO et al., 2013). No início de 2001 foi proposto então o Manifesto Ágil seguindo um conjunto de doze princípios e quatro valores para assim poder consolidar um ambiente ágil (FOWLER; HIGHSMITH, 2001).

Métodos Ágeis surgiram como uma reação às antigas formas tradicionais de desenvolvimento de software e reconheceram a necessidade de uma alternativa aos processos de desenvolvimento de software orientados por documentação (BECK et al., 2001b). A indústria e a tecnologia se movem muito rápido, os requisitos mudam com mais frequência que sobrecarregam os métodos tradicionais e os clientes tornaram-se cada vez mais incapazes de impor suas necessidades do que esperam a mais de seu software (HIGHSMITH; ORR; COCKBURN, 2000). Métodos Ágeis visam atender a necessidade de desenvolver softwares rapidamente e com mais eficiência em um ambiente em que os requisitos estão em constante mudança.

Ágeis é baseada na evolução e adaptação. A aprendizagem é a base desses dois. Portanto ágil e aprendizagem estão altamente relacionados. O Manifesto Ágil em si diz que as equipes ágeis devem ter um ambiente e suporte que eles precisam (BECK et al., 2001a). No mundo de desenvolvimento de software, onde novas técnicas e tecnologias surgem com frequência, um ambiente de trabalho que oferece aos desenvolvedores uma maneira de continuar aprendendo e melhorar suas habilidades de desenvolvimento tem uma vantagem técnica sobre os outros. Construir um ambiente de aprendizagem não é fácil, requer tempo e motivação. No entanto, os benefícios são claros, como software com melhor qualidade, melhorias contínuas, novos membros da equipe alcançam um nível de produtividade mais alto entre outras vantagens (HIGHSMITH; ORR; COCKBURN, 2000).

A aprendizagem ocorre no indivíduo e pode ser acessada pela organização de duas maneiras, através da disseminação de conhecimento de indivíduo para organização ou aquisição de indivíduos que adquiriram esse conhecimento (CAMPBELL; ARMSTRONG, 2013). O aprendizado é resultado de experiências em que só se obtém quando se tenta solucionar um problema e, portanto, ocorre durante a execução de

uma atividade. O aprendizado pela experiência aumenta o desempenho individual, o desempenho do time de desenvolvimento e o da organização (BOH; SLAUGHTER; ESPINOSA, 2007). Desta maneira há uma relação direta entre o aumento de desempenho com o aumento do aprendizado.

O desenvolvimento de software é uma atividade intensiva em aprendizagem e seu sucesso depende muito do conhecimento e da experiência dos desenvolvedores (KELLY, 2008). A aprendizagem ocorre quando indivíduos interagem e recebem novas informações de seu ambiente, dentro ou fora da organização. Ao fazer a aprendizagem colaborativa, a organização pode acelerar o novo processo de criação de conhecimento, que é a fonte de competitividade da organização (PETERS et al., 2010). Um dos objetivos do apoio ao aprendizado é melhorar a produtividade através do compartilhamento e transferência efetiva do conhecimento. O desenvolvimento de software ágil colocou um novo foco na questão de como compartilhar conhecimento adquirido entre os membros das equipes de desenvolvimento de software. Em contraste com os métodos centrados em documentação, as abordagens ágeis dependem da comunicação presencial para a transferência de conhecimento.

O compartilhamento de conhecimento é definido como o processo de disseminação de conhecimento que ocorre voluntariamente, de uma entidade para outra (BOCK; LEE; LEE, 2010). Esse tema cresceu à medida que empresas e organizações perceberam que o sucesso está muitas vezes determinado pela capacidade de criação, divulgação e incorporação de aprendizados em produtos e serviços (KAVITHA; AHMED, 2011). Essa realização levou a um maior interesse em examinar as formas como o aprendizado pode ser efetivamente efetivado, identificado, codificado, disseminado e retido. O campo da gestão de aprendizado surgiu para resolver essas necessidades. O conhecimento organizacional está incorporado nas pessoas, sistemas, procedimentos e produtos (KAVITHA; AHMED, 2011). Os trabalhadores são obrigados a melhorarem seus conhecimentos diariamente em um processo que se acumulam em uma melhoria significativa no desempenho para toda a empresa.

Este trabalho propõe apresentar um guia de técnicas e ferramentas que podem ser usadas com o propósito de apoiar o aprendizado em times ágeis que buscam uma melhoria em seus desempenhos.

1.1 Problema da Pesquisa

Quando o conhecimento não flui entre as equipes de projeto dentro de uma organização, os recursos são desperdiçados. Novas equipes de projetos descobrem novamente o mesmo trabalho já feito ao contrário de aprender de projetos anteriores (NEWELL et al., 2006). Alguns projetos repetem erros durante anos, devido ao fato que

o compartilhamento de aprendizados com projetos anteriores não aconteceu. Além disso, as empresas enxergam estes desperdício sob a forma de um potencial perdido para desenvolver habilidades dos funcionários (AJMAL; KOSKINEN, 2008). Assim, quando as equipes do projeto não compartilham as lições aprendidas, soluções ruins são duplicadas, erros repetidos e conhecimento de bons procedimentos perdidos, levando a retrabalho e oportunidades perdidas (PETTER; RANDOLPH, 2009).

Softwares desenvolvidos usando métodos tradicionais, o conhecimento é explícito e pode ser facilmente capturado, enquanto a metodologia ágil trata de conhecimento tácito. O conhecimento explícito é um conhecimento que pode ser facilmente articulado, codificado, acessado e verbalizado e o conhecimento tácito é armazenado nas mentes das pessoas sob forma de memória, habilidades, experiências educação, imaginação e criatividade (PALMIERI, 2002). Um dos maiores desafios das organizações é extrair o conhecimento tácito possuído por seus funcionários, de modo a aplicar o conhecimento certo no lugar certo como forma de incentivo a dar apoio ao aprendizado dos demais integrantes do time ágil (KAVITHA; AHMED, 2011).

Um outro desafio, se tratando de aprendizado em times ágeis, se dá devido a pressão de iteração. Aquela pressão sofrida pelas equipes para entregar toda a sua meta planejada sem atrasos dentro daquela iteração. Embora a pressão possa ser considerado algo benéfico pelo fato de ajudar equipes ágeis a se manterem motivadas em cumprir suas metas, um ambiente com vários projetos sendo executados paralelamente leva muitas vezes ao sacrifício de princípios básicos de métodos ágeis em relação com a aprendizagem contínua (HODA; BABB; NORBJERG, 2013).

Desse modo, devido a restrição de tempo que conseqüentemente acarreta uma pressão significativa nas equipes ágeis, as práticas relacionadas com a aprendizagem muitas vezes são sacrificadas ou mal executadas ao longo do processo de desenvolvimento (BABB; HODA; NORBJERG, 2014). Adaptações feitas em resposta a tais restrições de tempo geralmente não são planejadas de modo que suas conseqüências são difíceis de lidar. Dessa forma, para se obter uma aprendizagem eficaz sob pressão, estratégias devem ser criadas de modo que as práticas relacionadas com aprendizagem não deixem de ser executadas. Tanto as equipes quanto os clientes devem reconhecer a importância em gerenciar seus aprendizados (HODA; BABB; NORBJERG, 2013).

Dessa maneira, há um grande dilema que deve ser enfrentado pelas equipes ágeis. Aproveitar as oportunidades de aprendizado contínuo ou adiar as práticas focadas na aprendizagem para atender à pressão das iterações? O gerenciamento desse dilema requer um certo equilíbrio entre os dois lados. Equipes ágeis que sofrem uma pressão excessiva muitas vezes levam em conta decisões que favorecem soluções rápidas, sendo assim, decidem cumprir prazos de iterações ao invés de investir em aprendizado contínuo por ser uma opção que só se obtém resultados a longo prazo

(BABB; HODA; NORBJERG, 2014).

1.2 Justificativa

A importância desta pesquisa decorre das experiências que as organizações de Tecnologia da Informação (TI) tiveram na entrega de projetos mal sucedidos por décadas. Os projetos continuam a falhar por muitas das mesmas razões que eles fizeram há 30 anos (CERPA; VERNER, 2009). Essas falhas levam a consequências econômicas. De acordo com um estudo realizado pela *International Association of Software Architects* (2009) o Brasil perde cerca de 160 bilhões de dólares por ano devido a falhas em projetos de TI. Essa valor chega a um total de 6,2 trilhões de dólares em todo o mundo (IASA, 2009). Muito desses gastos podem ser atribuídas a mecanismos pobres de aprendizado organizacionais dentro das organizações de desenvolvimento de software (DESOUZA; DINGSØYR; AWAZU, 2005). As equipes do projeto não estavam aprendendo lições de outras equipes e isso contribuiu para maiores custos do projeto, desse modo, a falta de conhecimento foi considerada uma importante razão pela qual os projetos de TI falharam (HANISCH et al., 2009).

Esta pesquisa foi focada em fatores que possibilitem o fluxo de conhecimento entre as equipes ágeis relacionados com seus desempenhos, que como consequência, viabilizam o sucesso do projeto. O gerenciamento do conhecimento de modo que dê apoio ao aprendizado tem muito valor para as equipes ágeis, pois ela permite que uma organização obtenha, capture, armazene e transfira conhecimento, garantindo a utilização eficaz, a fim de proporcionar às pessoas a compreensão de porquê, como e o que realizar (KHALID; AHMED, 2015).

Portanto, times ágeis devem englobar-se regularmente em práticas que aprimoram vários tipos de aprendizagem. Ambas essas áreas de conhecimento e aprendizagem alimentam a melhoria contínua, um princípio básico de métodos ágeis no qual diz que em intervalos regulares, a equipe reflete sobre como se tornar mais eficaz (BECK et al., 2001a).

Dar suporte a compartilhamento de conhecimento pelos membros da equipe é fundamental para aumentar a agilidade em toda a organização e é considerado o novo horizonte para o desenvolvimento de software ágil (SANTOS; GOLDMAN; SOUZA, 2015). Uma aplicação bem sucedida utilizando técnicas de apoio ao aprendizado aumentam substancialmente a autoconsciência da equipe sobre sua cultura e torna seus hábitos mais visíveis para outras partes interessadas, melhorando também na comunicação (SCHNEIDER, 2015). Portanto técnicas para suprir as dificuldades em compartilhar conhecimentos e aprendizados entre integrantes de times ágeis são importantes práticas que devem ser seguidas.

Nesse contexto, chegamos as seguintes questões de pesquisa:

- 1) Como times de desenvolvimento de softwares ágeis tratam seus aprendizados?
- 2) Como esses times fazem a gestão de seus conhecimentos?

A partir de uma pesquisa, com o propósito de descobrir quais técnicas e ferramentas de apoio ao aprendizado estão sendo mais utilizadas por times ágeis e quais demonstram um melhor resultado para o time, este trabalho propõe um guia com técnicas e ferramentas que dão apoio ao aprendizado, que são consideradas como mais eficientes entre os times ágeis localizados na cidade de Recife e que poderão ser usadas por outros times que buscam uma melhoria em seus desempenhos.

1.3 Objetivos

Esta seção apresenta os objetivos geral e específicos deste trabalho.

1.3.1 Objetivo Geral

O trabalho proposto tem como objetivo geral realizar um estudo das principais técnicas e ferramentas que dão apoio ao aprendizado utilizadas por times ágeis de desenvolvimento de software em organizações situadas na cidade do Recife. Este trabalho visa analisar como esses times fazem a gestão de seus conhecimentos e como eles refletem seus aprendizados entre os integrantes do time.

1.3.2 Objetivos Específico

- 1) Identificar quais técnicas e ferramentas de apoio ao aprendizado estão sendo utilizadas pelos times ágeis na cidade do Recife.
- 2) Identificar quais técnicas e ferramentas de apoio ao aprendizado são consideradas mais eficientes para os times ágeis e suas organizações localizados na cidade de Recife.
- 3) Desenvolver um guia com as práticas que são mais utilizadas no mercado e que dão mais resultados no desempenho dos times ágeis.

1.4 Estrutura do Trabalho

Além deste capítulo introdutório, este trabalho apresenta outros 5 capítulos.

O capítulo 2 apresenta uma revisão da literatura com todo o referencial teórico, que de uma certa maneira contribuem para o entendimento sobre os tópicos abordados por este trabalho.

O capítulo 3 exhibe a metodologia do trabalho no qual descreve de forma clara e precisa como o estudo foi executado durante todas as etapas da pesquisa.

O capítulo 4 descreve os resultados obtidos pela pesquisa. Neste capítulo, além de expor todos os dados coletados referentes ao que se foi respondido nos questionários, são selecionadas as técnicas e ferramentas de apoio ao aprendizado para serem apresentadas no guia.

O capítulo 5 apresenta o desenvolvimento do guia. O guia descreve as técnicas e ferramentas selecionadas, dá dicas de quando usar-las, indicações, contra indicações e por fim informa boas práticas de como usa-las.

O capítulo 6 apresenta as considerações finais e dificuldades encontradas.

2 Revisão da Literatura

Neste capítulo será apresentado todo o referencial teórico que de alguma maneira serviu de base para o desenvolvimento deste trabalho. Aqui o leitor poderá ter uma melhor compreensão e contextualização sobre os tópicos abordados pelo trabalho relacionados com o estado atual da arte.

2.1 Metodologias Ágeis

Em fevereiro de 2001, um grupo de 17 pessoas se reuniram para conversar, esquiar, relaxar e tentar encontrar um objetivo em comum. O que se emergiu desse encontro foi uma aliança de desenvolvimento de software ágil formados por esses 17 integrantes. Esse encontro ficou conhecido como o Manifesto Ágil (MELO et al., 2013)

O surgimento do manifesto ágil foi motivado por uma série de desafios impostos pelas realidades modernas:

- Na economia moderna, muitas vezes é difícil ou impossível prever como um sistema baseado em computador evoluirá com o passar do tempo.
- As condições do mercado mudam rapidamente.
- As necessidades dos usuários finais evoluem e emergem novas ameaças competitivas sem aviso prévio.
- Em muitas situações, você não poderá definir os requisitos completamente antes do início do projeto.

Diante dessas circunstância e motivados por esses desafios, uma mudança na metodologia de como os softwares eram desenvolvidos se tornou primordial (PRES-SMAN; MAXIM, 2014). O manifesto ágil estava descobrindo melhores formas de desenvolver software e ajudandar os outros a desenvolverem.

Para se consolidar um ambiente ágil, o manifesto ágil apresentou um conjunto de quatro valores e doze princípios que deveriam ser seguidos (FOWLER; HIGHSMITH, 2001). A tabela 1 expõe os quatro valores do manifesto ágil.

Tabela 1 – Valores do Manifesto Ágil

Valores	
Indivíduos e Interações	Processos e Ferramentas
Software Funcionando	Documentação Abrangente
Colaboração do Cliente	Negociação de Contratos
Responder a Mudanças	Seguir o Plano

o autor

Isto é, na tabela 1, enquanto houver valor nos itens da direita, o Manifesto Ágil valoriza mais os itens da esquerda.

A tabela 2 expõe os doze princípios referentes ao manifesto ágil.

Tabela 2 – Princípios do manifesto ágil

Princípios	
1	Nossa maior prioridade é satisfazer o cliente através da entrega contínua e adiantada de software com valor agregado.
2	Mudanças nos requisitos são bem-vindas, mesmo tardiamente no desenvolvimento. Processos ágeis tiram vantagem das mudanças visando vantagem competitiva para o cliente.
3	Entregar frequentemente software funcionando, de poucas semanas a poucos meses, com preferência à menor escala de tempo.
4	Pessoas de negócio e desenvolvedores devem trabalhar diariamente em conjunto por todo o projeto.
5	Construa projetos em torno de indivíduos motivados. Dê a eles o ambiente e o suporte necessário e confie neles para fazer o trabalho.
6	O método mais eficiente e eficaz de transmitir informações para e entre uma equipe de desenvolvimento é através de conversa face a face.

Princípios	
7	Software funcionando é a medida primária de progresso.
8	Os processos ágeis promovem desenvolvimento sustentável. Os patrocinadores, desenvolvedores e usuários devem ser capazes de manter um ritmo constante indefinidamente.
9	Contínua atenção à excelência técnica e bom design aumenta a agilidade.
10	Simplicidade - a arte de maximizar a quantidade de trabalho não realizado - é essencial.
11	As melhores arquiteturas, requisitos e designs emergem de equipes auto-organizáveis.
12	Em intervalos regulares, a equipe reflete sobre como se tornar mais eficaz e então refina e ajusta seu comportamento de acordo.

o autor

De acordo com Highsmith e Cockburn (2001) o que é novo sobre métodos ágeis não são as práticas que usam, mas seu reconhecimento das pessoas como os principais impulsionadores do sucesso do projeto, juntamente com um foco intenso em eficácia. Isso produz uma nova combinação de valores e princípios que definem uma visão de mundo ágil (HIGHSMITH; COCKBURN, 2001).

Hawrysh e Ruprecht (2000) afirmam que uma metodologia única não pode funcionar para todo o espectro de projetos diferentes, mas sim a gestão do projeto deve identificar a natureza específica do projeto em questão e selecionar a melhor metodologia de desenvolvimento aplicável (HAWRYSH; RUPRECHT, 2000). Para enfatizar ainda mais, de acordo com McCauley (2001) há uma necessidade de métodos ágeis e orientados a processos, pois não existe um modelo de desenvolvimento de software de tamanho único que se adapte a todos os propósitos imagináveis (MCCAULEY, 2001). Esta opinião é compartilhada por vários especialistas da área.

Miller (2001) fornece as seguintes características aos processos de software ágil do ponto de vista de entrega rápida, que permitem encurtar o ciclo de vida dos projetos (MILLER, 2001):

- 1) Modularidade no nível do processo de desenvolvimento.
- 2) Iterativo com ciclos curtos permitindo verificações rápidas e correções.
- 3) Ciclos de iteração de uma a seis semanas.
- 4) Parcimônia no processo de desenvolvimento remove todas as atividades desnecessárias.

- 5) Adaptável com possíveis novos riscos emergentes.
- 6) Abordagem de processo incremental que permite o funcionamento da construção de aplicativos em pequenos passos.
- 7) A abordagem convergente e incremental minimiza os riscos.
- 8) Os processos ágeis orientados para as pessoas favorecem as pessoas sobre os processos e a tecnologia.
- 9) Estilo de trabalho colaborativo e comunicativo.

Highsmith e Cockburn (2001) relatam que o ambiente em mudança no negócio de software também afeta os próprios processos de desenvolvimento de software. Segundo eles, satisfazer os clientes no momento da entrega tem precedência sobre a satisfação do cliente no momento da iniciação do projeto. Isso exige procedimentos que não tratam tanto de como parar a mudança no início de um projeto, mas de como lidar melhor com mudanças inevitáveis ao longo de seu ciclo de vida. Afirma-se ainda que os métodos ágeis:

- Produz a primeira entrega em semanas, para obter uma vitória antecipada e um feedback rápido.
- Inventa soluções simples, então há menos para mudar e fazer essas mudanças é mais fácil.
- Melhorar a qualidade do projeto continuamente, tornando a próxima história menos dispendiosa para implementar.
- Testa constantemente, para detecção de defeito anterior, menos dispendiosa.

Os princípios básicos dos métodos ágeis compreendem uma honestidade implacável do código de trabalho, a eficácia das pessoas que trabalham em conjunto com a boa vontade e o foco no trabalho em equipe. Um conjunto de abordagens de senso comum emergentes de processos de desenvolvimento de software ágil foram sugeridos (AMBLER, 2002) da seguinte maneira:

- Pessoas importam
- É possível menos documentação
- A comunicação é um problema crítico
- As ferramentas de modelagem não são tão úteis como costumava pensar

- Não é necessário um grande design inicial.

Há pouco mais de uma década, a articulação do manifesto ágil em 2001 trouxe mudanças sem precedentes para o campo da engenharia de software. De fato, a transformação que o manifesto ágil trouxe a sua volta foi bastante notável. Uma série de métodos, aderindo a graus variados aos princípios do manifesto surgiram no cenário de desenvolvimento de software. Extreme programming (BECK, 1999), scrum (SCHWABER; BEEDLE, 2002), lean software development e feature-driven development (PALMER; FELSING, 2002) são alguns desses métodos que surgiram no decorrer do tempo. Em termos gerais, todos esses métodos se esforçaram para abordar os princípios fundamentais do manifesto.

Após mais de uma década de metodologias ágeis, a comunidade de pesquisa tem dedicado atenção sobre as questões relacionadas ao desenvolvimento ágil de software desde que o manifesto ágil foi pronunciado em 2001 (DINGSØYR et al., 2012). Torgeir Dingsoyr et al. (2012) acrescenta ser evidente a partir do número de publicações científicas, o interesse generalizado no tópico em vários fóruns científicos e o número de países envolvidos na pesquisa ágil. O número de questões especiais dedicadas ao desenvolvimento ágil também é uma indicação do grande interesse exibido na engenharia de software e outros campos relacionados, principalmente os sistemas de informação.

2.2 Gestão do Conhecimento

O processo de desenvolvimento de software é uma série de práticas intensivas em conhecimento que englobam atividades como coleta de requisitos, design, análise de problemas, codificação, teste e sua manutenção (LARMAN; BASILI, 2003). Desenvolvimento de software é uma atividade baseada no conhecimento humano, onde o nível de incerteza é alto (LEVY; HAZZAN, 2009). Meira Levy e Orit Hazzan (2009) acrescentam que a alta rotatividade de integrantes, a psicologia e a dinâmica da sociologia entre os membros da equipe e as mudanças de requisitos de software são apenas alguns exemplos dos desafios dos projetos de software. Desse modo, o sucesso de um projeto de software depende fortemente do conhecimento e da experiência trazidos ao projeto pelos seus praticantes (RUS; LINDVALL; SINHA, 2001).

O conhecimento é considerado o principal ativo competitivo da organização, permitindo que a empresa seja produtiva e ofereça produtos e serviços competitivos (LEVY; HAZZAN, 2009).

O conhecimento é uma mistura fluida de experiência emoldurada, valores, informações contextuais e visão especializada que fornece uma estrutura para avaliar e incorporar novas experiências e informações.

Ele origina e é aplicado nas mentes dos conhecedores. Nas organizações, muitas vezes se torna incorporado não apenas em documentos ou repositórios, mas também em rotinas, processos, práticas e normas organizacionais (DAVENPORT; PRUSAK, 1998).

Os membros da equipe frequentemente compartilham conhecimentos complexos e específicos do contexto que são essenciais para oferecer valor comercial aos clientes (MARTIN; BIDDLE; NOBLE, 2009). Nesta perspectiva, o compartilhamento efetivo de conhecimento na equipe é primordial para o sucesso dos projetos ágeis (MELNIK; MAURER, 2004).

As equipes ágeis se esforçam para desenvolver softwares com menos custo e com maior qualidade ao mesmo tempo em que envolvem mudanças ao longo do processo de desenvolvimento de software, dessa forma, precisam elevar seus conhecimentos, técnicas e ferramentas para gerenciar o desenvolvimento de produtos de software (DORAIRAJ; NOBLE; MALIK, 2012). Silva Dorairaj et al. (2012) complementam que aprender e adquirir conhecimento é crítico no desenvolvimento de software ágil que se concentra na satisfação do cliente através da entrega contínua de software.

O valor do conhecimento aumenta dentro da equipe ou organização quando o conhecimento tem um propósito chave e se concentra em valores, no entanto, as equipes de desenvolvimento de software enfrentam dificuldades para obter uma compreensão clara em relação ao conhecimento e ao gerenciamento de conhecimento (RUS; LINDVALL, 2002).

De acordo com Polanyi (2009) o conhecimento é dividido em dois tipos, conhecimento tácito e explícito (POLANYI, 2009). O conhecimento tácito é orientado a ação, ou seja, que orienta o compartimento humano, porém não pode ser expressado abertamente, enquanto o explícito é o conhecimento acadêmico ou que pode ser representado na forma escrita ou verbal (SMITH, 2001).

A gestão do conhecimento é um método que simplifica o processo de compartilhamento, distribuição, criação, captura e compreensão do conhecimento de uma empresa (DAVENPORT; PRUSAK, 1998). Gestão do conhecimento descreve os processos através dos quais as organizações gerenciam o conhecimento tácito e explícito realizado dentro dos indivíduos das organizações para obter vantagem competitiva (ALAVI; LEIDNER, 1999).

Dorairaj et al. (2012) explica que o conhecimento existe dentro das pessoas, portanto, o conhecimento transmitido fora da mente torna-se uma mera informação para o receptor. Nesta perspectiva, as organizações podem oferecer ferramentas, técnicas, processos e procedimentos, contudo, a gestão do conhecimento ainda será responsabilidade do indivíduo.

2.3 Técnicas e ferramentas de Apoio ao Aprendizado em Times Ágeis

A seguir serão citados as principais técnicas e ferramentas que dão apoio ao aprendizado em times ágeis de desenvolvimento de software de acordo com o que foi encontrado na literatura durante a fase de levantamento bibliográfico. Após uma análise com abordagem qualitativa, foram encontradas diversas técnicas e ferramentas mais relevantes para o estudo. Posteriormente foi realizada uma classificação e agrupamento com esses resultados de modo que se totalizaram dez principais técnicas e ferramentas. A Tabela 3 exibe as obras selecionadas que serviram de base para o levantamento dessas técnicas e ferramentas e o ano em que foram publicados.

Tabela 3 – Obras Selecionadas

Obras selecionadas	Ano
Agile Product Management with Scrum	2011
Agile Software Development with Scrum	2002
A systemic and cognitive view on collaborative knowledge building with wikis	2008
Digital scrum board using leap motion	2016
Distributing expertise in agile software development projects	2014
Embedding Reflection and Learning into Agile Software Development	2014
Knowledge Management through Pair Programming	2002
Knowledge sharing in distributed agile projects: Techniques, strategies and challenges	2014
Learning Across Teams: Knowledge Management Review	1999
The evolution of agile software development in Brazil	2013

o autor

2.3.1 Apresentações Técnicas, Workshops ou Seminários

De acordo com Razzak e Ahmed (2014), apresentações técnicas, workshops e seminários são organizados através da colaboração entre equipes de negócios, equipes técnicas e clientes, para compartilhar conhecimentos sobre projetos e as últimas tecnologias. Essas técnicas não só facilitam o entendimento e a comunicação comuns entre os diferentes membros da equipe como também ajudam a facilitar o compartilhamento de conhecimento tácito através da socialização (RAZZAK; AHMED, 2014).

2.3.2 Coaching, Mentoria ou Treinamentos Formais

Coaching, mentoria e treinamentos são práticas para melhorar o compartilhamento de conhecimento. Além de melhorar o compartilhamento de conhecimento essas técnicas visam uma melhoria na estratégia de negócios de longo prazo e na excelência técnica de seus profissionais (MELO et al., 2013). Rejab, Noble e Allan (2014) afirmam que coaching e mentoria são métodos para se obter conhecimento e as habilidades essenciais. Treinamentos fortalecem as habilidades dos membros da equipe, particularmente para aqueles que não conseguiram atender às expectativas. Com orientação e conhecimentos adequados os membros da equipe podem obter benefícios através da abordagem de treinamentos e orientações sendo elas essenciais para a distribuição bem sucedida de competências nas equipes ágeis (REJAB; NOBLE; ALLAN, 2014).

2.3.3 Comunidade de Prática, Grupos de Interesse ou Fóruns Técnicos

Segundo Razzak e Ahmed (2014) para ter sucesso em projetos ágeis, aprender é um ativo importante para as equipes ágeis. As equipes ágeis praticam dois modos de aprendizagem, aprendizagem entre pares e aprendizado comunitário. Na aprendizagem por pares, os membros da equipe começam a aprender através da interação e colaboração com os membros da equipe. O aprendizado comunitário está acessando e concebendo informações que estão disponíveis em arquivos de conhecimento ou em fóruns de discussão.

2.3.4 Ferramentas (Jira, Trello, Redmine, Google Docs, etc)

Hoje em dia, é muito comum equipes ágeis distribuídas ou dispersas de médio e grande porte usarem ferramentas como Jira, Trello ou Redmine para rastrear problemas, erros, tarefas, prazos, códigos e horas. Como ferramentas de colaboração e compartilhamento de conteúdo, os times ágeis usam essas ferramentas para compartilhar documentos, arquivos, idéias, especificações, diagramas e maquetes (RAZZAK; AHMED, 2014).

2.3.5 Fóruns de Discussão

De acordo com Razzak e Ahmed (2014) a idéia por trás de fóruns de discussão é aprender através do compartilhamento de conhecimento. Os fóruns de discussão são como comunidades de prática que criam uma rede entre membros com diversos níveis de conhecimento. Eles são grupos auto-organizados que consistem em indivíduos que compartilham informações, experiência e habilidades técnicas em uma disciplina especializada (MCDERMOTT, 1999). Os fóruns de discussão ajudam as equipes

distribuídas na rápida resolução de problemas e reduzem o tempo de desenvolvimento, uma vez que os membros da equipe não ficam presos em problemas recorrentes. Além de criar confiança entre os desenvolvedores, os fóruns de discussão ajudam a criar e compartilhar aprendizados locais e distribuídos (RAZZAK; AHMED, 2014).

2.3.6 Pair Programming, Rotação de Pares, Programação em Grupo ou Coding Dojo

Razzak e Ahmed (2014) citam que a técnica de pair programming é usada para a criação de tanto conhecimento local quanto global. Através de um estudo, Razzak e Ahmed (2014) constataram membros de uma equipe local e remota trabalhando juntos em uma estação de trabalho para resolver problemas específicos no qual eles se ajudam mutuamente para compartilhar seus pensamentos e criar aprendizados através da discussão. Na programação de pares, dois desenvolvedores trabalham juntos em um computador com um objetivo comum (PALMIERI, 2002).

Na programação grupal o desenvolvimento de software baseia-se em conhecimento tácito e repertório de acordo que o ritmo de trabalho e a velocidade da equipe serão estabelecidos, e quaisquer correções e ajustes, particularmente aqueles feitos “no local”, serão mais beneficiados com a reflexão em ação (BABB; HODA; NORBJERG, 2014). De forma explícita e tácita, os hábitos, as habilidades e o conhecimento passam entre os membros da equipe de forma reveladora em termos de trajetória do projeto e mitigação de riscos.

2.3.7 Quadros Brancos, Quadros de Inovação, Quadros Eletrônicos, Scrum Boards, Kanban Boards

De acordo com (RITTITUM; VATANAWOOD; THONGTAK, 2016) scrum board é um quadro que mostra o progresso do desenvolvimento da equipe. A equipe usa um quadro de scrum para planejar e rastrear o progresso durante cada sprint. É comum os quadros do scrum conterem quatro colunas padrão para indicar o progresso do sprint: coluna 'To Do' para as tarefas ainda não iniciadas, coluna 'In Progress' para as tarefas iniciadas, mas ainda não concluída, coluna 'Testing' para as tarefas concluídas, mas no processo de teste e coluna 'Concluída' para as tarefas que foram concluídas e testadas com sucesso.

As idéias mais inovadoras são mantidas na mente humana como um conhecimento tácito. Devido a cargas de trabalho contínuas, às vezes é impossível ter uma discussão com um membro da equipe ou outra pessoa experiente. Então, ao invés de falar com alguém, as pessoas compartilham suas idéias através de quadros de inovação de forma explícita (RAZZAK; AHMED, 2014).

2.3.8 Reuniões de Planejamento da Sprint/Projeto, Reuniões Diárias, Reunião de Revisão da Sprint

Daily Scrum é uma reunião diária de 15 minutos da equipe Scrum, permitindo que os membros examinem o progresso do trabalho e eliminem os impedimentos (SCHWABER; BEEDLE, 2002). Jeffry Babb et al. (2014) afirma que Daily Scrum permite que o praticante e a equipe lembrem da reflexão recente enquanto ela ainda está recente. Esta rotina diária constitui um componente criticamente importante para se engajar na prática reflexiva para beneficiar a manutenção do processo. Ao longo da prática diária, os profissionais da equipe irão, idealmente, acumular observações e reflexões.

Reunião de planejamento de Sprint é uma reunião em que a equipe do Scrum planeja o trabalho para o próximo sprint (PICHLER, 2011). O objetivo de sprint resultante é um compromisso da equipe para o sprint subsequente, que promove sua auto-organização e deve ser realista em relação às capacidades e capacidades da equipe para manter a motivação.

De acordo com Jeffry Babb et al. (2014) na técnica de apoio ao aprendizado de estimativa coletiva a equipe seleciona e estima coletivamente tarefas para a iteração atual e, ao fazê-lo, os membros da equipe adquirem e confirmam o conhecimento sobre as habilidades e áreas de especialização de cada um. Incluindo toda a equipe - não apenas desenvolvedores ou líderes, mas as partes interessadas também - asseguram uma visão compartilhada e compreensão do projeto com base em insumos de múltiplas perspectivas e acordos coletivos. As equipes podem usar isso como uma excelente oportunidade para discutir as expectativas dos clientes em relação à produtividade da equipe, que é um elemento-chave na gestão da aprendizagem ao longo do projeto (BABB; HODA; NORBJERG, 2014).

2.3.9 Retrospectivas e Práticas Reflexivas

Sprint retrospective é um encontro destinado a permitir que os participantes avaliem a efetividade e a eficiência da colaboração da equipe durante a última sprint. São identificados problemas, suas causas e soluções apropriadas para serem implementadas na próxima sprint. Assim, a reunião contribui para uma melhor colaboração e maior satisfação do trabalho nas próximas sprints (PICHLER, 2011). De acordo com Ken Schwaber e Mike Beedle (2002) em reuniões de retrospectiva todos os membros da equipe do projeto discutem os problemas de sprint anteriores e sugerem melhorias contínuas do processo. Retrospectivas fornecem um meio de examinar os pensamentos, experiências e notas sobre as várias consultas e exames do repertório individual e de equipe (BABB; HODA; NORBJERG, 2014).

2.3.10 Wiki ou Repositórios de Lições Aprendidas

Razzak e Ahmed (2014) afirmam que esses tipos de repositórios fornecem mecanismos eficientes para acessar o conhecimento codificado. Os praticantes são mais dependentes de repositórios para compartilhar conhecimentos entre membros da equipe local e distribuída. Wiki oferece novas oportunidades para aprender e usar o conhecimento colaborativo de construção e compartilhamento, através da interação social e da aprendizagem individual (CRESS; KIMMERLE, 2008). Segundo Jeffry Babb et al. (2014) As equipes geralmente capturam terminologia baseada em projetos e convenções específicas de contexto em um formato wiki compartilhado entre a equipe e o cliente. Os indivíduos também podem se envolver na redação de blogs para capturar e compartilhar o aprendizado adquirido com experiências de projetos.

2.4 Considerações Finais

Este capítulo apresentou todo o referencial teórico que foi utilizado como embasamento para a construção deste trabalho e para que o leitor tenha conhecimento acerca do que será exposto por ele. Foram apresentados tópicos relacionados à metodologia ágil, tais como sua motivação pela qual surgiu, seus valores e princípios e as técnicas e ferramentas de apoio ao aprendizado que foram encontradas na literatura.

3 Metodologia

Este capítulo descreve como o estudo foi executado durante todas as fases deste pesquisa. Este estudo foi desenvolvido através de uma abordagem exploratória com o objetivo de identificar respostas, por meio de aplicação de procedimentos científicos, para o problema da pesquisa. De acordo com a definição de Kothari (2004) pesquisas exploratórias buscam ganhar mais familiaridade com um fenômeno ou alcançar novos conhecimentos (KOTHARI, 2004).

3.1 Método da Pesquisa

No que diz respeito à abordagem de investigação, esta pesquisa seguiu um comportamento qualitativo. Segundo Terence e Escrivão Filho (2006) o propósito de pesquisas qualitativas são aprimorar o conhecimento do tema de acordo com a visão dos praticantes que adquiriram experiências diretas com o problema da pesquisa sem se preocupar com a representação numérica (TERENCE; ESCRIVÃO FILHO, 2006).

A validação da pesquisa foi realizada por meio de questionários durante a etapa de aplicação do questionário. Um questionário é um grupo ou sequência de perguntas destinadas a obter informações sobre um assunto de um respondente (FRANKLIN; SARAH; WALKER, 2003). Questionários desempenham um papel central no processo de coleta de dados, pois têm um impacto na qualidade dos dados e influenciam a imagem projetada do público. A técnica de aplicação de questionários dá permissão a um levantamento de uma grande quantidade de dados dos quais não se teria um mesmo resultado com uma pesquisa bibliográfica.

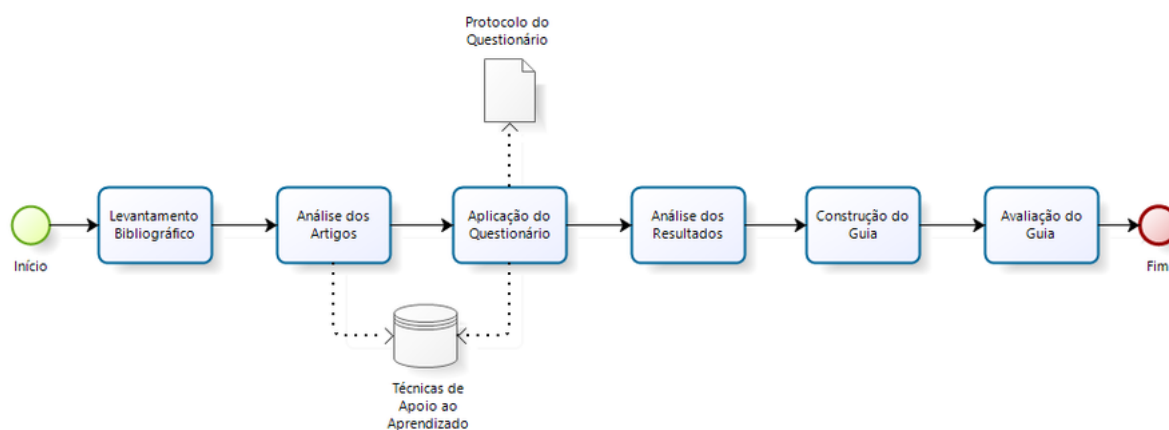
Durante a pesquisa foram utilizadas ferramentas de buscas para encontrar livros, trabalhos, revistas e artigos científicos relacionados ao tema da pesquisa. Durante todo o desenvolvimento da pesquisa as ferramentas de busca permitiram o acesso à trabalhos em diversos sites com conteúdos acadêmicos. O principal alvo das buscas foram trabalhos mais recentes, cujo tinham conteúdos mais semelhantes ao cenário atual quando se diz respeito à apoio ao aprendizado em times ágeis de desenvolvimento de software.

Trabalhos muito antigos, que foram publicados em um ano inferior a 2010, passaram por uma análise antes de serem utilizados. Essa análise foi feita com o intuito de evitar trabalhos com argumentos antigos e ultrapassados. As seguintes palavras-chaves foram utilizadas para a pesquisa: *agile*, *learn*, *learning*, *self-management*, *shared knowledge*, *learning team*, *knowledge management*, *distributed agile* entre outras palavras com idiomas variados. Desse modo foi possível entender, investigar e confrontar os trabalhos encontrados por meio da leitura.

3.2 Fases da Pesquisa

A pesquisa foi dividida em 6 fases. A figura 1 apresenta o fluxo dos processos da pesquisa.

Figura 1 – Fluxo da pesquisa



o autor

3.2.1 Levantamento Bibliográfico

A primeira fase da pesquisa foi destinada a reunir materiais (revistas científicas, livros, trabalhos e artigos acadêmicos) sobre o que se tinha relacionado a apoio ao aprendizado em time ágeis de desenvolvimento de software. Durante a fase de levantamento bibliográfico, os motores de busca que serviram como fornecedor de materiais relacionados foram as seguintes:

- IEEE Xplore Digital Library;
- Google Acadêmico;
- Springer Link;
- Science Direct;
- ACM Digital Library.

Essas ferramentas foram utilizadas durante todo o desenvolvimento deste trabalho e contribuíram para que fossem encontrados trabalhos relevantes ao estudo.

Uma simples análise foi feita durante o levantamento bibliográfico com o propósito de filtrar os resultados. Dessa forma, uma leitura era feita de modo que obras não relacionadas ao tema da pesquisa ou que não fossem contribuir de alguma forma fossem descartadas nessa etapa evitando assim que passassem para a próxima fase da pesquisa.

3.2.2 Análise dos Artigos

Já com posse dos artigos relacionados ao tema da pesquisa, resultantes da primeira fase, foi realizado uma análise aprofundada com uma abordagem qualitativa. Nessa fase foram extraídas da literatura técnicas e ferramentas que poderiam ser utilizadas pelos times ágeis de desenvolvimento de software no que se trata de apoio ao aprendizado. As técnicas e ferramentas foram identificadas e classificadas de modo que pudessem ser agrupadas de acordo como elas deveriam ser executadas.

Além disso, foi possível adquirir conhecimentos sobre os principais desafios encontrados pelos times ágeis em gerenciar seus aprendizado e dos principais problemas resultantes de sua ausência. Como consequência dessas análises, informações referentes às técnicas e ferramentas que dão apoio ao aprendizado foram coletadas para que o questionário conseguisse ser desenvolvido e posteriormente ser aplicado.

3.2.3 Aplicação do Questionário

Antes de sua aplicação, um protocolo foi criado com diretrizes de como o questionário deveria ser conduzido. O protocolo criado pode ser visto no APÊNDICE A deste trabalho. No protocolo foi definido o público alvo da aplicação do questionário, o nível de experiência de seus respondentes, a localidade de onde trabalham, o objetivo do questionário, definição de suas contribuições, análise dos riscos, limitações e ameaças à validade dos resultados e outros pontos relevantes que foram seguidos durante sua aplicação.

O questionário foi desenvolvido com grande parte de suas questões baseadas em conhecimentos extraídos dos trabalhos relacionados após a fase de levantamento bibliográfico e a fase de análise dos artigos. Uma outra parte de suas questões foram adicionadas ao questionário a partir de adaptações, visando obter respostas essenciais para a pesquisa e uma melhor qualidade nos resultados.

3.2.4 Análise dos Resultados

Esta fase da pesquisa foi destinada para analisar os dados obtidos a partir dos resultados coletados nas entrevistas e questionários. Durante esta etapa foi feita uma análise das técnicas e ferramentas de apoio ao aprendizado que são utilizadas por esses times, problemas encontrados por não utilizarem elas e dificuldades encontradas no seu uso.

3.2.5 Construção do Guia

A fase seguinte da pesquisa foi reservada para construção de um guia com as técnicas e ferramentas que obteram um melhor resultado de acordo com os questionários. A seleção das técnicas e ferramentas foi realizada de acordo com os resultados obtidos na etapa anterior da pesquisa, durante a fase de análise dos resultados. As técnicas e ferramentas que foram citadas como mais eficientes e as que foram as mais utilizadas pelos respondentes do questionário, foram selecionadas para serem apresentadas no guia. A tabela 5 apresenta as técnicas mais utilizadas entre os respondentes e a tabela 6 apresenta as que foram citadas como mais eficientes entre os times ágeis. Por fim, apenas um subconjunto reduzido de técnicas e ferramentas foram apresentadas no guia por serem as mais relevantes pelos respondentes.

O guia desenvolvido apresenta as técnicas e ferramentas selecionadas e ensina boas práticas sobre como utiliza-las, no qual seu principal objetivo é aumentar o desempenho dos times ágeis que optarem por usá-las. No guia, cada técnica contém os seguintes itens:

- Título
- Descrição
- Quando usar
- Indicações e contra indicações
- Trabalhos Relacionados
- Modos de usar

3.2.6 Avaliação do Guia

Finalmente, após construção do guia, a última fase da pesquisa ficou destinada para a validação do guia. Nesta fase, o guia construído na etapa da pesquisa anterior foi analisado pela doutora em Ciências da Computação e especialista na área de metodologias ágeis e também orientadora deste trabalho Suzana Cândido de Barros Sampaio. Esta análise contou, basicamente, com uma verificação e validação do que foi construído e apresentado pelo guia. Após análise do especialista, algumas correções surgiram e mudanças no guia foram necessárias.

3.3 Considerações Finais

Neste capítulo foi apresentado toda a metodologia da pesquisa utilizada por este trabalho. O método da pesquisa e todas as fases da pesquisa como levantamento

bibliográfico, análise dos dados, a aplicação do questionário, análise dos resultados, construção do guia e por fim a validação do guia tiveram todos seus passos descritos sobre como foram feitos.

4 Resultados e Discussão

Este capítulo apresenta os resultados obtidos após a realização da aplicação do questionário com pessoas que trabalham na cidade do Recife e utilizam metodologias ágeis em seu processo de desenvolvimento de software. Os resultados demonstram, de maneira generalizada, o perfil dos profissionais de TI, como os times ágeis fazem a gestão e refletem seus aprendizados entre os integrantes do time dentro de suas organizações.

4.1 O Questionario

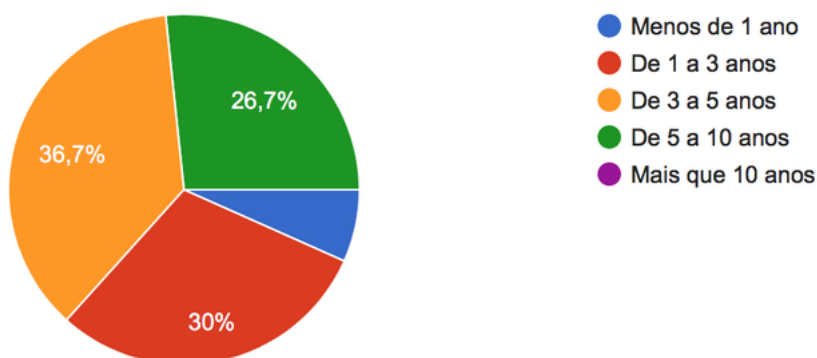
De acordo com o que foi especificado na seção 2.3 do capítulo 3, o questionário foi aplicado durante o dia 7 de novembro de 2017 até o dia 4 de dezembro de 2017. Totalizando assim um período de 27 dias. O questionário foi aplicado com pessoas que trabalhavam com desenvolvimento de software em empresas situadas na cidade do Recife e utilizavam algum tipo de metodologias ágeis. Durante a aplicação questionário foi possível coletar respostas de 30 pessoas que se enquadravam com as especificações da pesquisa. O questionário foi aplicado na forma digital e ele é composto por 15 perguntas sendo elas divididas em 2 seções. O questionário pode ser visto no Apendice A desde trabalho.

4.2 Perfil dos Respondentes

A primeira seção do questionário contém 5 perguntas e tinha o objetivo de conhecer, de modo geral, o profissional de TI que iria responde-lo. A seção envolve perguntas referentes ao nível de experiência do respondente no que se diz respeito à desenvolvimento de software e metodologias ágeis.

Dentre as respostas recebidas, o questionário coletou informações sobre como esses profissionais gerenciam seus aprendizado em um total de 19 diferentes empresas. Dessas 19 empresas, 8 delas são empresas de Tecnologia da Informação e Comunicação (TIC) embarcadas no Porto Digital. O Porto Digital é um dos maiores parques tecnológicos do Brasil, situado na capital pernambucana Recife, que atua nas áreas de tecnologia da informação e comunicação (PORTO. . . , 2018). Os respondentes foram questionados sobre o seu nível de experiência com desenvolvimento de software como mostra a Figura 2.

Figura 2 – Nível de experiência com desenvolvimento de software

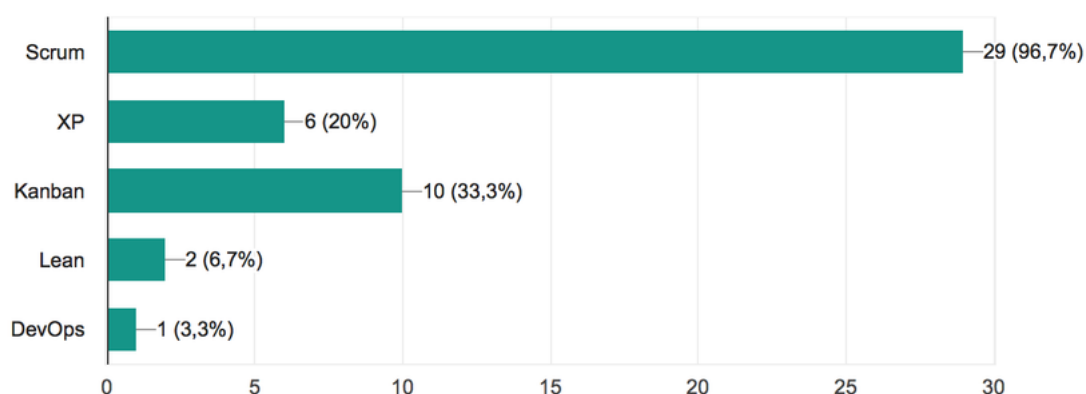


o autor

Observa-se um equilíbrio entre os respondentes com experiência em desenvolvimento de software de 1 a 3 anos, de 3 a 5 anos e de 5 a 10 anos. A grande minoria dos respondentes tinham um nível de experiência inferior a 1 ano e não houveram respondentes com mais que 10 anos de experiência.

No que se diz respeito a metodologias ágeis, os respondentes ficaram livres para assinalar mais de uma alternativa e inserir alguma outra que não estivesse entre as opções de resposta. Contudo, as respostas foram divididas em 5 diferentes tipos de métodos como mostra a Figura 3.

Figura 3 – Metodologia ágil usada no processo de desenvolvimento



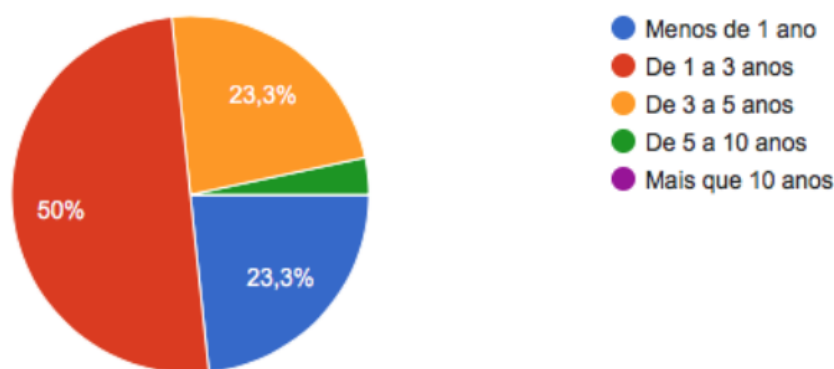
o autor

De acordo com o gráfico é evidente que o método ágil Scrum é a mais usada pelos times ágeis, pois apenas 1 dos 30 respondentes não utiliza essa metodologia em seu processo de desenvolvimento. A metodologia Kanban ficou como a segunda mais

usada entre eles, ficando com um total de um terço das respostas, seguida por XP com 20%. As metodologias ágeis Lean e DevOps também aparecem nas respostas para uma pequena minoria dos respondentes. No total os métodos somaram 48 alternativas assinaladas, o que deixa claro que uma boa parte dos respondentes utiliza mais de uma metodologia ágil em seu processo de desenvolvimento.

Além de quais métodos ágeis, os respondentes também foram questionados sobre a quanto tempo eles a usavam. Figura 4 apresenta o grau de experiência desses usuários com metodologias ágeis.

Figura 4 – Nível de experiência com metodologias ágeis

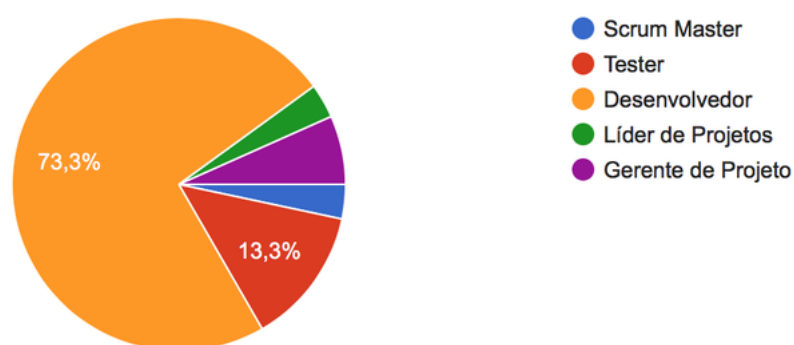


o autor

Exatamente metade dos respondentes afirmam ter de 1 a 3 anos de experiência com metodologias ágeis. Coincidentemente, com a mesma quantidade de respostas, os respondentes com graus de experiência inferior à 1 ano e de 3 a 5 anos somam 7 das respostas cada. Por fim, totalizando 100% das respostas, apenas 1 respondente informa ter de 5 a 10 anos de experiência.

Finalizando a seção que corresponde as perguntas de caráter de seus perfis, os respondentes também informaram sobre a sua área de atuação dentro da organização em que trabalham. Os respondentes eram livres para assinalar alguma das opções ou inserir alguma outra que não estivesse entre elas. Com uma ampla maioria, 22 dos 30 respondentes atuam com o cargo/papel de desenvolvedor. O cargo/papel de testes alcançou 4 dos respondentes. Os restantes dos respondentes declaram atuar na área de Gerente de projetos, Líder de projetos e Scrum master conforme mostra a Figura 5.

Figura 5 – Área de atuação



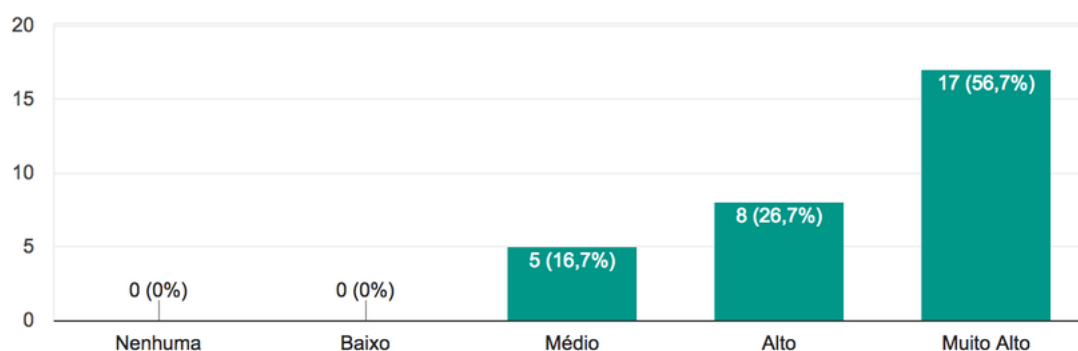
o autor

De acordo com as respostas coletadas pelos 30 respondentes, é possível afirmar que há um equilíbrio entre seus níveis de experiência em desenvolvimento de software e com metodologias ágeis. Dentre os respondentes, existem tanto profissionais com um nível de experiência mais elevado, quanto profissionais com grau de experiência intermediário e também profissionais que estão começando a trabalhar na área de desenvolvimento com métodos ágeis. Scrum é a metodologia ágil mais utilizada entre eles e a maioria atua como desenvolvedor.

4.3 Importância das Técnicas e Ferramentas de Apoio ao Aprendizado

A segunda seção abrangia 10 perguntas e tinha como foco coletar respostas relativas ao objetivo principal desta pesquisa. Descobrir como os times ágeis gerenciam seus aprendizados e os principais problemas enfrentados por eles. Desse modo, a seção englobava perguntas relacionados ao apoio ao aprendizados e seus desafios.

Dentre os 30 respondentes do questionário, 25 afirmam que seu time ágil faz uso de alguma técnica que auxilia no gerenciamento de seus aprendizados dentro de sua organização. Questionados pelo quanto importante eles consideram times ágeis em fazer o uso de técnicas e ferramentas que dão apoio aos aprendizados, em geral, eles classificaram como de média importância para muito importante o uso dessas técnicas e ferramentas durante seu processo de desenvolvimento de software. A grande maioria dos respondentes classificaram com grau de maior importância utilizar essas técnicas e ferramentas dentro de suas organizações, como mostra Figura 6.

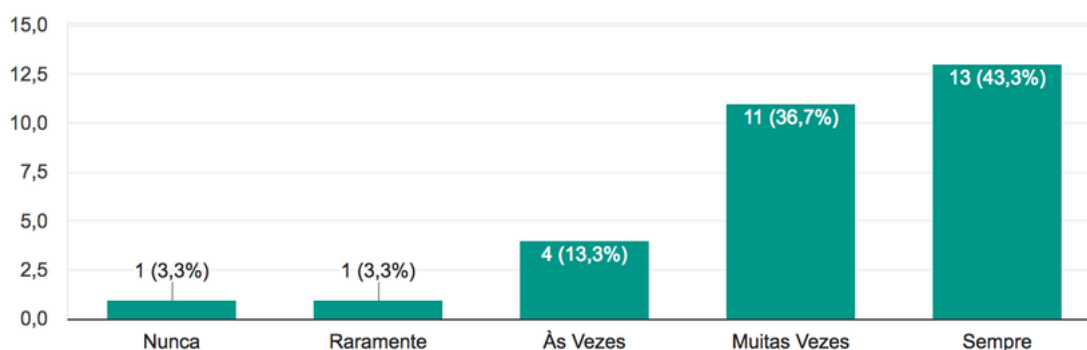
Figura 6 – A importância em usar técnicas e ferramentas de apoio ao aprendizado

o autor

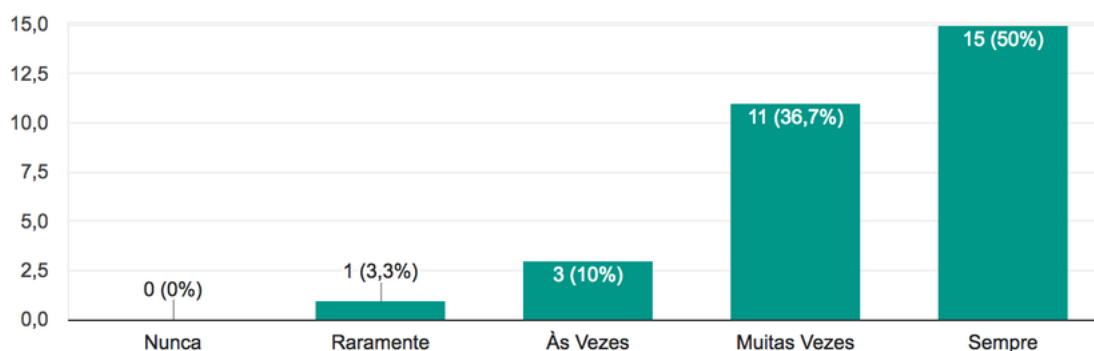
Isso mostra que mesmo os respondentes que admitem não fazer o uso de alguma técnica que dê apoio ao aprendizado, eles não consideram que essas técnicas e ferramentas não sejam importantes.

4.4 Aprender e Compartilhar Aprendizados com Outros Membros

Saber o quanto os respondentes costumam aprender e compartilhar seus aprendizados com os membros dos seus times foram outras questões abordadas no questionário. A Figura 7 e a Figura 8 apresentam o quanto os respondentes demonstram aprender e compartilhar seus aprendizados respectivamente.

Figura 7 – O quanto costuma aprender com outros membros do time

o autor

Figura 8 – O quanto costuma compartilhar aprendizados com outros membros do time

o autor

Na questão sobre aprender com outros membros do time, houveram respostas em ambas as opções. A grande maioria dos respondentes indicam ser acostumados a aprender sempre ou muitas vezes com os integrantes de seus times, pois 24 dos 30 assinalaram uma dessas duas opções. Entre os respondentes, 4 dizem aprender às vezes e 2 deles informam que aprendem raramente ou nunca aprendem com algum membro de seu time ágil.

No que se diz respeito sobre compartilhar seus aprendizados, as respostas recebidas não foram muito diferentes das respostas sobre aprender com outros membros. A maior parte dos respondentes reconhece ter o costume de compartilhar seus aprendizados com seus parceiros de desenvolvimento. Ao total 26 respondentes tem a conduta de compartilhar seus aprendizados sempre ou muitas vezes. Três deles dizem realizar essa conduta às vezes e apenas 1 confessa compartilhar raramente seus aprendizados.

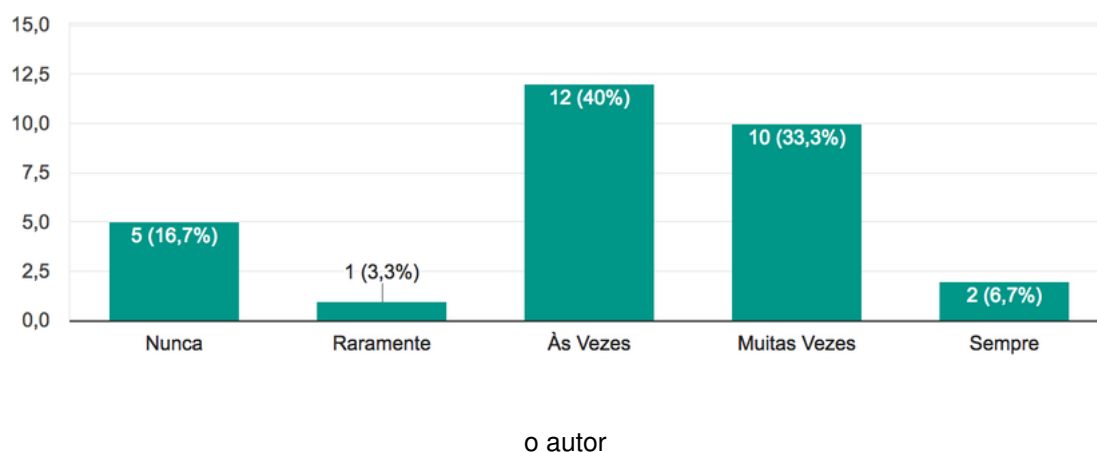
Além de considerar as práticas que dão apoio ao aprendizado importantes, aprender e compartilhar seus aprendizados com outros membros de seus times são hábitos muito frequentes entre os respondentes.

4.5 Uso das Técnicas e Ferramentas que dão Apoio ao Aprendizado

Um outro ponto explorado pelo questionário foi com que frequência os times tinham o costume em utilizar técnicas e ferramentas que dão apoio aos seus aprendizados. De acordo com o gráfico a seguir é possível ver que as respostas dos respondentes ficaram divididas em todas as opções de frequência. A maioria dos respondentes afirmou fazer uso de técnicas e ferramentas de apoio ao aprendizados às vezes ou muitas vezes. Contudo, uma considerável parte deles, 5 no total, assumem nunca ter feito o uso dessas técnicas e ferramentas. Dos restantes, 2 dizem sempre usá-las e 1

dos respondentes confirma que raramente usa técnicas ou ferramentas de apoio ao aprendizado.

Figura 9 – Com que frequência costuma usar técnicas e ferramentas de apoio ao aprendizado



A Figura 9 reflete com que frequência os times ágeis estão gerenciando seus aprendizados dentro de suas organizações. De acordo com o gráfico, fica evidente que, apesar de considerá-las importantes, realizar o uso dessas técnicas e ferramentas que dão apoio ao aprendizado não são tratadas com importância máxima. Contudo, são práticas que são lembradas e que estão frequentemente sendo usadas por eles em suas organizações.

Um catálogo com as principais técnicas e ferramentas que dão apoio ao aprendizado, de acordo com o que mais foi identificado na literatura, foi apresentado aos participantes de modo que eles pudessem avaliá-las. A Tabela 4 seguir mostra as técnicas e ferramentas já classificadas e agrupadas que foram apresentadas aos respondentes.

Tabela 4 – Técnicas e ferramentas de apoio ao aprendizado

Técnicas e ferramentas encontradas na literatura	
1	Apresentações técnicas, workshops ou seminários
2	Coaching, mentoria ou treinamentos formais
3	Comunidade de prática, grupos de interesse ou fóruns técnicos
4	Ferramentas (Jira, Trello, Redmine, Google Docs, etc)
5	Fóruns de discussão
6	Pair programming, rotação de pares, programação em grupo ou coding dojo
7	Quadros brancos, quadros de inovação, quadros eletrônicos, scrum boards, kanban boards
8	Reuniões de planejamento da sprint/projeto, reuniões diárias, reunião de revisão da sprint
9	Retrospectivas e práticas reflexivas
10	Wiki ou repositórios de lições aprendidas

o autor

Com o catálogo em vista, os 30 respondentes analisaram cada uma das 10 técnicas e ferramentas contidas nele e assinalaram com que frequência eles e seus times usam durante seus processos de desenvolvimento ágeis. Eles tiveram 4 opções de respostas para cada técnica: Não conheço, não uso, uso eventualmente e uso frequentemente. Dentre todas as técnicas e ferramentas de apoio ao aprendizado apresentadas aos respondentes, as ferramentas (Jira, trello, redmine, google docs, etc) foram assinaladas como utilizada frequentemente por 27 dos 30 respondentes e usada eventualmente por 1 deles. Reuniões de planejamento da sprint/projeto, reuniões diárias ou estimativa coletiva ficou como a segunda técnica mais usada por eles sendo utilizada frequentemente por 20 e eventualmente por 7 respondentes. A terceira técnica ficou com quadros brancos, quadros de inovação, quadros eletrônicos, scrum boards e kanban boards sendo frequentemente usada por 14 respondentes e usada eventualmente por 10 deles. Retrospectivas e práticas reflexivas foi a próxima técnica de apoio ao aprendizado mais usada com 10 confirmações dos respondentes por ser utilizada com frequência e 14 eventualmente. Com 4 indicações por ser usada com frequência e 16 eventualmente a técnica wiki ou repositórios de lições aprendidas ficou em quinto como a mais utilizada entre os respondentes e curiosamente foi a técnica que teve um maior número de respondentes, 3 no total, que assumiram não

conhece-la. Os resultados com as frequências que todas as técnicas e ferramentas de apoio ao aprendizado utilizadas pelos respondentes e seus times podem ser vistas na Tabela 5. A Tabela 5 apresenta todas as técnicas e ferramentas e a frequência com que são usadas entre os respondentes. A frequência está classificada entre não conhece, não usa, usa eventualmente ou usa com frequência.

Tabela 5 – Frequência de uso das técnicas e ferramentas

Técnica e ferramentas	Não conhece	Não usa	Usa eventualmente	Usa com frequência
Apresentações técnicas, workshops ou seminários	1	9	18	2
Coaching, mentoria ou treinamentos formais	1	15	13	1
Comunidade de prática, grupos de interesse ou fóruns técnicos	1	14	8	7
Ferramentas (Jira, Trello, Redmine, Google Docs, etc)	1	1	1	27
Fóruns de discussão	2	14	8	6
Pair programming, rotação de pares, programação em grupo ou coding dojo	2	9	16	3
Quadros brancos, quadros de inovação, quadros eletrônicos, scrum boards, kanban boards	1	5	10	14
Reuniões de planejamento da sprint/projeto, reuniões diárias, reunião de revisão da sprint	2	1	7	20
Retrospectivas e práticas reflexivas	2	4	14	10
Wiki ou repositórios de lições aprendidas	3	7	16	4

o autor

Técnicas como comunidades de prática, grupos de interesse ou fóruns técnicos, fóruns de discussão e coaching, mentoria ou treinamentos formais foram assinaladas como não utilizadas por 14 ou mais dos respondentes. Deste modo são classificadas como as técnicas de apoio ao aprendizado menos importantes entre os respondentes.

4.6 Técnicas e Ferramentas mais Eficientes

Questionados sobre quais principais técnicas e ferramentas de apoio ao aprendizado, utilizados por eles e seus times dentro de suas organizações são mais eficientes, os respondentes informaram 15 diferentes tipos de técnicas e ferramentas. A Tabela 6 ilustra todas as técnicas e ferramentas, que dão apoio ao aprendizado, citadas como mais eficientes entre os respondentes e seus times.

Tabela 6 – Técnicas e ferramentas mais eficientes entre os respondentes

Votos	Técnicas e ferramentas de apoio ao aprendizado
11	Reuniões de Planejamento da Sprint/Projeto
6	Reuniões Diárias
6	Scrum board
6	Ferramentas (Jira, Trello, Redmine, Google docs, etc.)
5	Estimativa Coletiva
4	Wikis
4	Retrospectivas
2	Pair Programming
2	Quadros Kanban
2	Quadros brancos

o autor

Os resultados mostram que a técnica mais presente entre eles são as reuniões de planejamento da Sprint do projeto. Ao todo essa técnica somou 11 das indicações dos respondentes. Um dos respondentes que informou planejamento de sprint ser a principal técnica de apoio ao aprendizado em seu time recomendou:

Planejar para cada Sprint, seja no início, meio ou fim, uma reunião para transferência de conhecimento de um para todos do time. Visando igualar os conhecimentos técnicos do time. Podendo ser chamada de pílula de conhecimento ao time.

Após as reuniões de planejamento, três outras técnicas e ferramentas foram mencionadas 6 vezes cada uma. A primeira delas foram as ferramentas (Jira, Trello, Redmine, Google docs, etc), a segunda ficou com as Scrum boards e a terceira, também com 6 menções, foram as reuniões diárias. Citada 5 vezes pelos respondentes,

a estimativa coletiva foi a seguinte técnica mais eficiente entre eles. Wiki e práticas de retrospectivas foram técnicas e ferramentas que receberam 4 indicações cada. De acordo com um dos participantes:

Reuniões e práticas reflexivas é a principal técnica utilizada e hoje a que tem o processo melhor definido, mas por exemplo, para wikis e repositórios de lições aprendidas ainda não encontramos o melhor processo para que todos possam tomar conhecimento do que foi postado/aprendido por um membro do time, seja por descentralização das informações, seja por tempo disponível para outros membros ou outro fator.

As técnicas e ferramentas que envolvem quadros kanban, quadros brancos e pair programming foram lembradas por duas vezes cada uma. Um dos representantes citou:

O pair programming é muito eficiente, mas utilizávamos de forma muito informal. Ele só era usado eventualmente, quando alguém estava com um problema que não conseguia resolver. Talvez se ele fosse algo mais formalizado seria bom.

Outras técnicas que foram lembradas apenas 1 vez por algum dos respondentes foram: reunião para desenho técnico da solução (DT), documentação via código, mentoring, apresentações técnicas, workshops ou seminários e testes de demonstrações.

4.7 Problemas Encontrados no Gerenciamento de Aprendizados e Recomendações

Os respondentes deram comentários sobre quais maiores problemas eles e seus times ágeis enfrentam no gerenciamento de seus aprendizados. Foram recebidos importantes comentários, por exemplo, que existem muitas dificuldades em compartilhar aprendizados com novos integrantes em seus times, dificuldades em gerir um alto número de informação que vem de diversas fontes ou até mesmo pelo fato que o uso de técnicas e ferramentas que dão apoio ao aprendizado não estarem presentes em seus processos de desenvolvimento se dá muitas vezes por existir uma falta de interesse por parte de alguns integrantes. Um dos respondentes que afirma não fazer uso de técnicas de apoio ao aprendizado diz:

Os problemas passados e que foram “aprendidos” não ficam registrados em um lugar que possa ser buscado facilmente para que pessoas que não passaram por aquilo possam aprender. Acontece que a forma que esse conhecimento é compartilhado é por membros do time que passaram por aquele momento e, por experiência, podem repassar o aprendizado

Outros respondentes reconhecem que o maior problema pelo seu time não fazer o uso dessas técnicas ou até mesmo deixar de fazê-las se dá pela falta de tempo disponível. Quando existe uma pressão para entregas, devido a ausência de tempo ou sobrecarga de projetos, as práticas de apoio ao aprendizado são facilmente ignoradas.

Recomendações sobre as técnicas e ferramentas que dão apoio ao aprendizado, de acordo com experiências vivenciadas por eles dentro de suas organizações na área de T.I. também foram compartilhadas. Muitos participantes do questionário acreditam que essas técnicas e ferramentas são muito importantes para aumentar o desempenho da equipe. Sugestões como documentar o máximo todo o aprendizado adquirido das práticas de desenvolvimento diárias, planejar uma reunião periódica com práticas reflexivas e de retrospectivas para compartilhamento de aprendizados, fazer uma apresentação inicial de algumas técnicas e começar inserir de forma lenta no time foram compartilhadas pelos respondentes. Um deles acrescenta:

O gerenciamento da informação é muito importante para o crescimento técnico da equipe. Ajudando os integrantes, principalmente os de menor senioridade e conhecimento, a serem menos dependentes de outros membros da equipe.

4.8 Análise dos Resultados e Seleção das Técnicas e ferramentas para Montagem do Guia

Os respondentes validaram as técnicas e ferramentas de apoio ao aprendizado levantadas pela literatura e forneceram valiosas informações sobre a importância do uso dessas técnicas no processo de suas organizações. O uso dessas técnicas acarretam uma significativa melhoria no desempenho de suas equipes ágeis e na qualidade do processo de desenvolvimento. Com base nos resultados expostos neste trabalho foi possível selecionar as melhores práticas e técnicas de apoio ao aprendizado e criar um guia prático de como utilizá-las. De acordo com os resultados, apenas as técnicas e ferramentas citadas como mais eficientes e mais utilizadas foram selecionadas para a construção deste guia.

Para a criação do guia foi necessário realizar uma seleção, de caráter qualitativo, das técnicas e ferramentas de apoio ao aprendizado mais eficientes. A escolha das técnicas e ferramentas foram baseadas de acordo com as respostas dos respondentes nos questionários. Quais técnicas são mais utilizadas e quais técnicas e ferramentas são mais eficientes, conforme afirmações colhidas pelos respondentes e expostas nas seções 5 e 6 do capítulo 4, foram os parâmetros usados para a seleção dessas técnicas e ferramentas.

Foi necessário criar um quadro de pontuação para definir quais técnicas e ferramentas deveriam ser usadas para a criação deste guia. De acordo com os

resultados apresentados na seção 5 do capítulo 4, que retrata o uso das técnicas que dão apoio ao aprendizado, as técnicas em que os respondentes afirmam utilizar frequentemente receberam 2 pontos para cada marcação. As técnicas e ferramentas que foram assinaladas como usadas eventualmente receberam 1 ponto. Contudo, para cada técnica que foi indicada como não conhecida pelos respondentes foram negativadas por 1 ponto. Por fim, as técnicas e ferramentas de apoio ao aprendizado, citadas pelos respondentes por ser as mais eficientes em suas organizações, apresentadas na seção 6 do capítulo 4, também receberam 1 ponto para cada marcação. A Tabela 7 apresenta a pontuação recebida pra cada resposta diferente dos respondentes.

Tabela 7 – Pontuação

Resposta	Pontos
Frequentemente	2
Eventualmente	1
Mais eficiente	1
Não conhecida	-1

o autor

Após soma de todos os pontos decorrentes das respostas dos respondentes, o resultado final desta pontuação ficou definida na Tabela 8. A tabela 8 apresenta todas as técnicas e ferramentas, sua pontuação de acordo com a frequência de uso entre os respondentes, sua pontuação de acordo com quantas vezes foram citadas entre os respondentes como mais eficientes e por fim a soma total das duas pontuações.

Tabela 8 – Técnicas e ferramentas organizadas por pontuação

Posição	Técnica e ferramentas	Frequência de Uso	Mais Eficiente	Total
1	Reuniões de planejamento da sprint/projeto, reuniões diárias, reunião de revisão da sprint	45	17	62

Posição	Técnica e ferramentas	Frequência de Uso	Mais Eficiente	Total
2	Ferramentas (Jira, Trello, Redmine, Google Docs, etc)	54	6	60
3	Quadros brancos, quadros de inovação, quadros eletrônicos, scrum boards, kanban boards	37	10	47
4	Retrospectivas e práticas reflexivas	32	4	36
5	Wiki ou repositórios de lições aprendidas	21	4	25
6	Apresentações técnicas, workshops ou seminários	21	1	25
7	Pair programming, rotação de pares, programação em grupo ou coding dojo	20	2	22
8	Comunidade de prática, grupos de interesse ou fóruns técnicos	21	0	21
9	Fóruns de discussão	18	0	18
10	Coaching, mentoria ou treinamentos formais	15	0	15

o autor

Por fim, foram selecionadas as seguintes técnicas e ferramentas de apoio ao aprendizado para a construção do guia por obterem uma pontuação total superior a 30 pontos:

- 1) Reuniões de planejamento da sprint/projeto, reuniões diárias, reuniões de revisão da sprint
- 2) Ferramentas (Jira, trello, redmine, google docs, etc)
- 3) Quadros brancos, quadros de inovacao, quadros eletronicos, etc.
- 4) Retrospectivas e praticas reflexivas

4.9 Considerações Finais

Neste capítulo foi exposto os resultados obtidos com a realização da fase de aplicação do questionário. O questionário foi aplicado com pessoas que utilizam metodologias ágeis em seus processos de desenvolvimento e que trabalham na cidade

do Recife. Foi apresentado o perfil dos respondentes deste questionário e os resultados relacionados à importância do uso das técnicas e ferramentas de apoio ao aprendizado e os principais problemas ao gerenciar aprendizados. O guia teórico com as técnicas e ferramentas mais eficientes será discutido no próximo capítulo.

5 Desenvolvimento do Guia

Este capítulo corresponde às etapas seguintes decorrentes dos resultados da pesquisa. Após uma análise dos resultados da pesquisa, um guia teórico com as técnicas e ferramentas selecionadas que dão apoio ao aprendizado serão apresentados neste capítulo.

5.1 Guia com as técnicas e Ferramentas Selecionadas

Em seguida será apresentado um guia teórico com as técnicas e ferramentas selecionadas de acordo com os resultados da pesquisa revelados neste trabalho. A forma que as técnicas e ferramentas são apresentadas é através de *guidelines*, cujo o formato segue um padrão que foi baseado na forma que Sommerville (SOMMERVILLE; SAWYER, 1997) descreveu para engenharia de requisitos e melhoria de processo e com algumas adaptações para descrever padrões de software. Cada técnica apresentada neste guia segue o padrão detalhado abaixo:

- Título: Nome da técnica
- Descrição: Breve introdução e explicação conceitual sobre a técnica
- Quando usar: Orientações sobre quando usar a técnica
- Indicações/Contra Indicações: Recomendações para se obter sucesso no uso da técnica
- Trabalhos Relacionados: Quem recomenda o uso da técnica
- Modo de usar: Passos com orientações sobre como usar a técnica

5.1.1 Reuniões de planejamento da sprint/projeto

Descrição

A reunião de planejamento da sprint é uma reunião na qual deverá ser discutido entre os membros dos time quais funcionalidades serão desenvolvidas para a próxima sprint. Sprint é um período de tempo, que deve ser definido de acordo com o escopo do projeto e tempo para seu desenvolvimento, no qual as atividades planejadas para ela devem ser concluídas até seu término.

Quando usar

As reuniões de planejamento da sprint devem ser usadas sempre antes do início de uma nova sprint.

Indicações

- Todos os interessados pelo projeto devem estar presentes.

Trabalhos Relacionados

Ken Schwaber e Mike Beedle (2002), Jeffrey Babb et al. (2014) e Roman Pichler (2011).

Modo de usar

No decorrer da reunião de planejamento de sprint, o representante do cliente dentro do time deverá descrever todas as funcionalidades do produto a ser desenvolvido. Todas essas funcionalidades serão organizadas de acordo com as prioridades do cliente. Essas funcionalidades priorizadas deverão ser descritas e exibidas ao restante da equipe em forma de uma história de usuário. O resultado disso será uma lista de tarefas priorizadas, em modo de histórias, que deverão ser desenvolvidas para se obter um produto final. De acordo com a lista das tarefas priorizadas, cada membro da equipe deverá escolher as atividades que irão se comprometer a desenvolver até o final da sprint. Nesse momento, os membros devem ser precisos em suas estimativas para que não corram o risco de não terem tempo suficiente para finalizarem suas atividades ou até mesmo terminá-las antes do tempo do término da sprint. O resultado final da reunião de planejamento da sprint será uma lista com todas as funcionalidades, em formato de histórias, que deverão ser desenvolvidas durante toda a próxima sprint.

5.1.2 Reuniões diárias

Descrição

Reuniões diárias são reuniões de curta duração que devem ser realizadas todos os dias entre os membros dos times ágeis. Elas são organizadas para acompanhar continuamente o progresso da equipe. As reuniões diárias são de grande importância para o discernimento do conhecimento sobre o que foi feito no dia anterior, o que está sendo feito por cada membro e identificar possíveis impedimentos para que possam ser contornados. Dessa forma, elas servem para manter a equipe informada sobre o andamento do projeto.

Quando usar

As reuniões diárias devem ser feitas todos os dias.

Indicações

- Todos os membros do time devem estar presentes.
- Realizar as reuniões diárias sempre no mesmo horário.
- Realizar as reuniões diárias sempre no mesmo local.
- O local da reunião seja próximo a todos e de preferência no mesmo local de trabalho.
- A reunião deve ser realizada com todos os membros em pé.
- Duração de no máximo 15 minutos

Trabalhos Relacionados

Ken Schwaber e Mike Beedle (2002), Jeffry Babb et al. (2014) e Roman Pichler (2011).

Contra indicações

- Para equipes que tem dificuldades em manter o foco na reunião.

Modo de usar

Em um determinado horário, todos os membros do time devem se reunir em um local previamente planejado. No decorrer da reunião, todos os membros devem falar, um por um, o que foi feito no dia anterior, o que está fazendo e o que vai ser feito até o final do dia e se há algum tipo de impedimento que está atrapalhando em seu trabalho.

5.1.3 Reunião de revisão da sprint

Descrição

A reunião de revisão da sprint tem como objetivo realizar uma análise sobre tudo que se passou e o que foi feito durante toda a sprint. Durante cada sprint é necessário fornecer um incremento do produto. O que significa que no final da sprint uma nova versão do produto estará disponível. Desse modo, no final de cada sprint uma reunião

é realizada na qual a equipe irá mostrar tudo o que foi alcançado durante a sprint finalizada. O projeto também deve ser avaliado em relação aos objetivos da sprint que foram determinados durante a sua reunião de planejamento.

Quando usar

Reuniões de revisão da sprint são reuniões que devem ser realizadas após o final de cada sprint.

Indicações

- É fortemente indicado que todos os envolvidos com o desenvolvimento com o projeto participem desta reunião.
- Seja feita uma demonstração das novas funcionalidades desenvolvidas durante a sprint.

Trabalhos Relacionados

Ken Schwaber e Mike Beedle (2002), Jeffry Babb et al. (2014) e Roman Pichler (2011).

Contra indicações

- Não ultrapassar 2 horas de reunião.
- Evitar o uso de apresentações com slides.
- Não tornar a reunião uma distração ou desvio para a equipe.

Modo de usar

Após término de uma sprint todos os membros devem se reunir para rever tudo que foi desenvolvido durante a sprint. O foco dessa reunião é que a equipe mostre às partes interessadas do projeto todo o trabalho que foi realizado durante a sprint. Um membro da equipe deverá conduzir esta reunião e deverá revisar os seguintes dados:

- O trabalho que a equipe se comprometeu a entregar até o final daquela sprint.
- O trabalho que foi feito pelo time.
- As principais decisões que foram tomadas durante a iteração da sprint (Isto pode incluir requisitos técnicos e as decisões tomadas pela equipe, clientes ou qualquer outra pessoa).

- Métricas do projeto.
- Uma demonstração do trabalho feito.

5.1.4 Ferramentas (Jira, trello, redmine, google docs, etc)

Descrição

Jira, Trello, Redmine são ferramentas essenciais utilizadas por times de desenvolvimento ágil no qual tem o objetivo de gerenciar projetos e atingir resultados de qualidade com mais agilidade. Essas ferramentas servem para armazenar todas as características, requisitos, funcionalidades e bugs do projeto. Além disso, elas exibem todo o status do projeto, permitindo assim efetivar o monitoramento das tarefas/funcionalidades que ainda devem ser ou que já foram desenvolvidas, se já foram testadas, se já estão em produção ou até mesmo se deixaram de ter prioridade no projeto e foram descartadas. São importantes ferramentas que tornam o desenvolvimento do projeto mais ágil e que permitem uma melhor visualização dos projetos e seus deadlines. O Google Docs é um pacote de ferramentas que permite integrantes criar, editar e compartilhar documentos, apresentações e planilhas com todos os membros do time. Os times ágeis fazem uso dela para compartilhar qualquer informação sobre projetos. Ferramentas como skype, slack, telegram e whatsapp também se enquadram nesse conjunto de ferramentas importantes para compartilhamento de conhecimentos e gerenciamento de aprendizados. Elas são ferramentas que permitem a comunicação de toda a equipe e são usadas para trocar qualquer tipo de informação em tempo real.

Quando usar

Sempre que forem precisas, as ferramentas devem ser usadas em qualquer momento.

Indicações

- É essencial que todos os membros usem ferramentas no decorrer de projetos.
- Manter as ferramentas sempre atualizadas de acordo com o status do projeto.
- Compartilhar as informações com todos.
- Fazer uso de ferramentas que auxiliem na comunicação do time.

Contra indicações

- Evitar distrações ao utilizar ferramentas de comunicação instantânea.

Trabalhos Relacionados

Razzak e Ahmed (2014)

Modo de usar

Não há uma maneira correta para uso dessas ferramentas, porém existem boas práticas que visam um melhor desempenho para a equipe. Ferramentas de gestão de projetos englobam diversas funcionalidades, contudo, sua principal função é usada para organizar todas as atividades do projeto que devem ser exercidas pela equipe. Cada atividade deve estar associada às pessoas que estão envolvidas nelas, ou seja, os integrantes que se comprometerem a fazê-las. As atividades devem ser organizadas em um quadro que irá revelar qual o status dessas atividades. Em geral o quadro de status deverá dividir as atividades que devem ser feitas, que estão sendo feitas e atividades que já foram concluídas. Entretanto, esses status podem ser mais complexos e subdivididos em atividades que estão impedidas, que estão em teste ou que estão em produção. Há inúmeras formas de organizar os quadros de atividades e isso vai depender muitas vezes da complexidade do projeto. As atividades devem conter o maior número de informações possíveis, especialmente quando envolve mais de um integrante da equipe. As informações contidas nas atividades devem ser bem claras de modo que evite perdas de tempo para esclarecer dúvidas ou exercer atividades erroneamente. Sempre que as atividades mudarem de um status para outro elas devem ser atualizadas no quadro de atividades. Desse modo todos os integrantes podem ver qual o status de cada atividade e como está o andamento do projeto como um todo.

5.1.5 Quadros brancos, quadros de inovação, quadros eletrônicos, etc

Descrição

O uso de quadros são importantes meios para se obter um ambiente de trabalho colaborativo e são bastante poderosos no que se trata em compartilhar de conhecimentos e aprendizados. Os quadros, de maneira geral, são usados para exibir informações que sejam úteis para os membros da equipe. As informações postadas nos quadros vão desde informações referentes à alguma atividade do projeto até medidas que devem ser realizadas pela equipe. Quadros também são úteis para gerenciar o progresso do time, apontar atividades que já foram feitas, atividades que ainda devem ser concluídas ou até mesmo as atividades que estão em desenvolvimento. Realizar planejamentos das sprints, passar alguma informação para o time ou até mesmo serem usados para tirar dúvidas ou esclarecer idéias também são outras utilidades dos quadros. Dessa forma é muito comum fazer uso desses quadros para externar conhecimentos ou aprendizados

através de esboços ou até mesmo usá-los para dar início em discussões.

Quando usar

Quadros podem ser usados em qualquer momento.

Indicações

- Todos os integrantes do time podem usar.
- Fazer uso de post-it.
- Manter o quadro organizado e atualizado.
- Expor os quadros em locais acessíveis e que todos da equipe possam ver.
- Ter cuidado na caligrafia.
- Usar quadros digitais em times distribuídos.

Contra indicações

- Usar quadro físico em times distribuídos.

Trabalhos Relacionados

Razzak e Ahmed (2014) Rittitum et al. (2016)

Como usar

Quadros são úteis em diversas situações. Sempre que for preciso compartilhar conhecimentos ou aprendizados use-os para expor o que tiver para repassar para o restante do time. Fazer o uso de post-it é sempre bem vindo em quadros brancos, pois ajuda na organização e facilita ao repassar informações. Tenha cuidado na caligrafia, pois após expor suas idéias os membros interessados poderão visualizar e relembrar o que foi exposto. Sempre que a informação já não servir mais para nenhum membro ou houver algum equívoco sobre o que foi escrito, o conteúdo apresentado deve ser apagado.

5.1.6 Retrospectivas e práticas reflexivas

Descrição

Retrospectivas e práticas reflexivas são reuniões estruturadas em torno de reflexões sobre como a equipe está trabalhando e encontrar maneiras de melhorar. De acordo com os aprendizados, a equipe deve refletir sobre o que eles devem continuar fazendo, o que se deve parar de fazer e o que eles devem começar a fazer. A equipe ágil revisa tudo o que aconteceu durante a sprint, procura repassar os aprendizados para todos os membros da equipe de modo que esses ensinamentos possam melhorar em seus desempenhos em futuras sprints. Os aprendizados e opiniões de melhorias compartilhadas pelos membros devem ser revisados por todos da equipe para que assim possam entrar em consenso e decidir se elas podem ser adotadas para a equipe.

Quando usar

Essa técnica deve ser a última coisa a ser feita em uma sprint, desse modo ela deve ser usada no final de cada sprint.

Indicações

- É indicado que essa técnica aconteça com toda a equipe presente.
- Sua duração tenha em média 1 hora, podendo se estender por mais tempo de acordo com sua necessidade.

Trabalhos Relacionados

Jeffrey Babb et al. (2014), Pichler (2011) e Schwaber e Beedle (2002).

Modo de usar

Um dos membros deve agir como um facilitador no decorrer da retrospectiva. O facilitador deve perguntar para todos os membros presentes para se concentrarem e identificar coisas, visando uma melhoria no processo, que eles achem que devem começar, parar ou continuar a fazer durante a próxima sprint. Depois de uma lista inicial de idéias, provocadas por um brainstorming, todos os membros devem votar em cada item específico e tomar uma decisão sobre o que fazer para a próxima sprint. Os novos itens que forem decididos por começar a fazer, deverão ser lembradas novamente na próxima retrospectiva de modo que irão decidir se foi uma decisão eficaz.

5.2 Considerações Finais

Este capítulo do trabalho foi reservado para apresentar o guia prático com as técnicas e ferramentas de apoio ao aprendizado mais eficientes de acordo com os resultados obtidos na aplicação do questionário. As técnicas e ferramentas que foram citadas como mais eficientes e as que foram as mais utilizadas pelos respondentes do questionário, foram selecionadas para serem apresentadas no guia. O guia desenvolvido apresenta essas técnicas e ferramentas selecionadas e ensina boas práticas sobre como utilizá-las, no qual seu principal objetivo é aumentar o desempenho dos times ágeis que optarem por usá-las. No guia, cada técnica contém descrição, quando usar, indicações, contra indicações e como usar. O capítulo a seguir descreve a conclusão deste trabalho.

6 Conclusão

Este trabalho teve o objetivo de explorar como os times de desenvolvimento localizados na cidade do Recife e praticantes de metodologias ágeis realizam a gestão de seus aprendizados. Através de pesquisa e estudo foi possível realizar um levantamento das principais técnicas e ferramentas de apoio ao aprendizados citadas na literatura. Mediante a aplicação de questionários via internet, especificamente com pessoas que usam métodos ágeis em seu processo de desenvolvimento e que trabalhem na cidade do Recife, foi possível identificar quais técnicas e ferramentas de apoio ao aprendizado são mais utilizadas por eles, quais são as mais eficientes e os principais problemas encontrados pelos times. Com base nos resultados alcançados, foi possível perceber que as técnicas e ferramentas encontradas na literatura são aceitas e muito utilizadas no mercado atual. Conforme os resultados, um guia prático foi desenvolvido, apresentando assim, as técnicas e ferramentas mais utilizadas e eficientes de acordo com o cenário tecnológico de Recife. O guia apresenta as técnicas e ferramentas, dá dicas de quando se deve utilizá-las, indicações, contra indicações e por fim, ensina boas práticas de como usá-las.

O guia apresentado neste trabalho, com as técnicas e ferramentas que dão apoio ao aprendizado, tem o propósito aumentar no desempenho dos times que dão importância para a gestão de seus aprendizados. Considerando que hoje em dia o mercado de trabalho é muito competitivo, aumentar o desempenho dos times ágeis significa uma melhor qualidade no produto ou serviço que esses times desenvolvem. Além disso, fazer uso dessas técnicas e ferramentas não exige muito dos times e podem ser consideradas fáceis de implementar em seus processos de desenvolvimento.

6.1 Limitações e Trabalhos Futuros

Os resultados apresentados por este trabalho apresenta limitações, sobretudo, pelo tamanho reduzido da amostra de participantes do questionário e pelo fator regional da qual a pesquisa foi realizado. Além disso, o número total das empresas acobertadas pelos questionários não é o suficiente para uma generalização a respeito do cenário sobre gestão de aprendizados. Uma outra limitação é a eficácia do guia proposto como resultados deste trabalho para times ágeis que estejam dispostos a darem apoio aos seus aprendizados.

Dessa forma, como possíveis trabalhos futuros temos as seguintes possibilidades:

- Expandir o fator regional da pesquisa com praticantes de metodologias ágeis afim de identificar como é feita a gestão dos aprendizados.

- Avaliar o uso do guia com times ágeis que estejam dispostos a darem apoio aos seus aprendizados.
- Definir quais técnicas e ferramentas são melhores de serem usadas de acordo com as experiências dos times.

6.2 Dificuldades Encontradas

No decorrer do trabalho foram encontradas diversas dificuldades. Durante a fase de levantamento bibliográfico houve uma certa dificuldade para se encontrar obras relacionados ao tema da pesquisa. O levantamento das técnicas e ferramentas que dão apoio ao aprendizado também foi uma atividade que precisou demandar tempo e dedicação. Da mesma forma, foram encontradas dificuldades no decorrer da fase de aplicação do questionário. Além de ter dificuldades em encontrar pessoas que fossem fiéis às condições estipuladas para a pesquisa, conseguir um número satisfatório para que a mesma pudesse ter uma amostra aceitável foram desafios que tiveram que ser contornadas.

Referências

- AJMAL, M. M.; KOSKINEN, K. U. Knowledge transfe in project-based organizations: An organization culture perspective. *Project Management Journal*, v. 39, n. 1, p. 7 – 15, 2008.
- ALAVI, M.; LEIDNER, D. Knowledge management systems: Issues, challenges and benefits. *Communications of the Association for Infor- mation Systems*, v. 1, n. 7, 1999.
- AMBLER, S. Lessons in Agility from Internet-Based Development. *IEEE Software*, p. 66 – 73, 2002.
- BABB, J.; HODA, R.; NORBJERG, J. Embedding Reflection and Learning into Agile Software Development. *IEEE Software*, v. 31, n. 4, p. 51 – 57, Julho 2014.
- BECK, K. Extreme programming explained. Reading, Mass. Addison-Wesley, 1999.
- BECK, K. et al. *Manifesto for agile software development*. 2001. Disponível em: <<http://agilemanifesto.org>>. Acesso em: 03/01/2018.
- BECK, K. et al. *Agile Manifesto*. 2001. Disponível em: <www.agilemanifesto.org>. Acesso em: 03/01/2018.
- BOCK, G. W.; LEE, J. Y.; LEE, J. Cross Cultural Study on Behavioral Intention Formation in Knowledge Sharing. *Asia Pasific Journal of Information System*, v. 20, n. 3, p. 1 – 32, 2010.
- BOH, W. F.; SLAUGHTER, S. A.; ESPINOSA, J. A. Learning from Experience in Software Development: A Multilevel Analysis. *Management Science*, v. 53, n. 8, p. 1315 – 1331, 2007.
- CAMPBELL, T. T.; ARMSTRONG, S. J. A longitudinal study of individual and organizational learning. *The Learning Organization*, v. 20, n. 3, p. 240 – 258, 2013.
- CERPA, N.; VERNER, J. M. Why did your projects fail? *Communications of the ACM*, v. 52, n. 12, p. 130 – 134, 2009.
- CRESS, U.; KIMMERLE, J. A systemic and cognitive view on collaborative knowledge building with wikis. *International Journal of Computer-Supported Collaborative Learning*, p. 105 – 122, Junho 2008.
- DAVENPORT, T. H.; PRUSAK, L. *Working Knowledge: How Organizations Manage What They Know*. [S.I.]: Harvard Business School, 1998.
- DESOUZA, K. C.; DINGSØYR, T.; AWAZU, Y. Experiences with Conducting Project Postmortems: Reports vs. Stories and Practitioner Perspective. *Software Process: Improvement and Practice*, v. 10, n. 2, p. 203 – 215, 2005.
- DINGSØYR, T. et al. A decade of agile methodologies: Towards explaining agile software development. *The Journal of Systems and Software*, v. 88, n. 6, p. 1213 – 1221, 2012.
- DORAIRAJ, S.; NOBLE, J.; MALIK, P. Knowledge Management in Distributed Agile Software Development. In: AGILE CONFERENCE. IEEE, 2012. [S.I.], 2012. p. 64 – 73.

- FOWLER, M.; HIGHSMITH, J. The agile manifesto. *Software Development*, p. 28 – 35, 2001.
- FRANKLIN; SARAH; WALKER, C. *Survey Methods and Practices*. Ottawa: Canada, 2003.
- HANISCH, B. et al. Knowledge management in project environments. *Journal of Knowledge Management*, v. 13, n. 4, p. 148 – 160, 2009.
- HAWRYSH, S.; RUPRECHT, J. Light Methodologies: It's Like Déjà Vu All Over Again. *Cutter IT Journal*, v. 13, p. 4 – 12, 2000.
- HIGHSMITH, J. *Adaptive Software Development: A Collaborative Approach To Managing Complex Systems*. [S.l.]: Addison-Wesley, 2013.
- HIGHSMITH, J.; COCKBURN, A. Agile Software Development: The Business of Innovation. *Computer*, v. 34, n. 9, p. 120 – 122, 2001.
- HIGHSMITH, J.; ORR, K.; COCKBURN, A. "Extreme Programming,". *E-Business Application Delivery*, p. 4 – 17, 2000.
- HODA, R.; BABB, J.; NORBJERG, J. Toward Learning Teams. *IEEE Software*, v. 30, n. 4, p. 95 – 98, 2013.
- IASA. 2009. Disponível em: <<https://www.iasaglobal.org>>. Acesso em: 12/01/2018.
- KAVITHA, R. K.; AHMED, M. S. I. A knowledge management framework for agile software development teams. In: PROCESS AUTOMATION, CONTROL AND COMPUTING (PACC), 2011 INTERNATIONAL CONFERENCE ON. IEEE, 2011. [S.l.], 2011. p. 1 – 5.
- KELLY, A. *Changing Software Development: Learning to Become Agile*. [S.l.]: John Wiley & Sons, 2008.
- KHALID, A.; AHMED, M. Impact of knowledge sharing on organizational learning: Moderating effect of organizational leadership. *Asian Journal of Management Research*, v. 5, n. 3, 2015.
- KOTHARI, C. R. *Research methodology: Methods and techniques*. [S.l.]: New Age International, 2004.
- LARMAN, C.; BASILI, V. R. Iterative and incremental developments: A brief history. *Computer*, v. 36, n. 6, p. 47 – 56, 2003.
- LEVY, M.; HAZZAN, O. Knowledge Management in Practice: The Case of Agile Software Development. In: COOPERATIVE AND HUMAN ASPECTS ON SOFTWARE ENGINEERING, 2009, Vancouver. Vancouver, 2009. p. 60 – 65.
- MARTIN, A.; BIDDLE, R.; NOBLE, J. The XP Customer Team: A Grounded Theory. In: AGILE CONFERENCE, 2009, Chicago. Chicago, 2009. p. 57 – 64.
- MCCAULEY, R. Agile Development Methods Poised to Upset Status Quo. *SIGCSE Bulletin*, v. 33, n. 4, p. 14 – 15, 2001.

- MCDERMOTT, R. Learning Across Teams. *Knowledge Management Review*, p. 32 – 36, Junho 1999.
- MELNIK, G.; MAURER, F. Direct verbal communication as a catalyst of Agile knowledge sharing. In: AGILE DEVELOPMENT CONFERENCE, 2004. [S.l.], 2004. p. 21 – 31.
- MELO, C. et al. The evolution of agile software development in Brazil. *Journal of the Brazilian Computer Society*, v. 19, p. 523 – 552, 11 2013.
- MILLER, G. The Characteristics of Agile Software Processes. In: THE 39TH INTERNATIONAL CONFERENCE OF OBJECT-ORIENTED LANGUAGES AND SYSTEMS, 2001, Santa Barbara. Santa Barbara, 2001.
- NEWELL, S. et al. Sharing knowledge across projects: Limits to ICT- led project review practices. *Management Learning*, v. 37, n. 2, p. 167 – 185, 2006.
- PALMER, S.; FELSING, J. M. A Practical Guide to Feature-Driven Development. 2002.
- PALMIERI, D. W. “Knowledge Management through Pair Programming”, Thesis. 2002.
- PETERS, L. D. et al. Collaboration and collective learning: networks as learning organisations. *Journal of Business & Industrial Marketing*, v. 25, n. 6, p. 478 – 484, 2010.
- PETTER, S.; RANDOLPH, A. B. Developing soft skills to manage user expectations in IT projects: Knowledge reuse among IT project managers. *Project Management Journal*, v. 40, n. 4, p. 45 – 59, 2009.
- PICHLER, R. *Agile Product Management with Scrum. Creating Products that Customers Love*. 4. ed. [S.l.]: Upper Saddle River: Addison, 2011.
- POLANYI, M. *The Tacit Dimension*. [S.l.]: The University of Chicago Press, 2009.
- PORTO Digital. 2018. Disponível em: <<http://farol.portodigital.org>>. Acesso em: 07/01/2018.
- PRESSMAN, R. S.; MAXIM, B. *Software Engineering: A Practitioner’s Approach*. 8. ed. [S.l.]: McGraw-Hill Science/Engineering/Math, 2014.
- RAZZAK, M. A.; AHMED, R. Knowledge Sharing in Distributed Agile Projects: Techniques, Strategies and Challenges. *Federated Conference on Computer Science and Information Systems*, v. 2, p. 1431 – 1440, Setembro 2014.
- REJAB, M.; NOBLE, J.; ALLAN, G. Distributing Expertise in Agile Software Development Projects. In: AGILE CONFERENCE (AGILE), 2014. [S.l.]: IEEE, 2014. p. 33 – 36.
- RITTITUM, P.; VATANAWOOD, W.; THONGTAK, A. Digital scrum board using leap motion. *IEEE/ACIS 15th International Conference on Computer and Information Science*, Junho 2016.
- RUS, I.; LINDVALL, M.; SINHA, S. S. Knowledge Management in Software Engineering - A State of the Art report. *Fraunhofer Center for Experimental Software Engineering and the University of Maryland*, p. 1 – 57, 2001.

- RUS Ioana; LINDVALL, M. Knowledge management in software engineering. *IEEE Software*, v. 19, n. 3, p. 26 – 38, 2002.
- SANTOS, V.; GOLDMAN, A.; SOUZA, C. R. B. D. Fostering effective inter-team knowledge sharing in agile software development. *Empirical Software Engineering*, v. 20, n. 4, p. 1006 – 1051, 2015.
- SCHNEIDER, M. The Guide Board, an Artefact to Support the Continuous Improvement of an Agile Team's Culture. In: AGILE PROCESSES IN SOFTWARE ENGINEERING AND EXTREME PROGRAMMING, 2015. *Lecture Notes in Business Information Processing*. [S.I.], 2015. v. 212.
- SCHWABER, K.; BEEDLE, M. *Agile Software Development with Scrum*. [S.I.]: Upper Saddle River: Prentice Hall,, 2002.
- SHAWKY, D. M. Traditional vs Agile Development A Comparison Using Chaos Theory. In: SOFTWARE PARADIGM TRENDS (ICSOFT-PT), 9TH INTERNATIONAL CONFERENCE ON, 2014, Vienna. Vienna, 2014.
- SMITH, E. A. The role of tacit and explicit knowledge in the workplace. *Journal of Knowledge Management*, v. 5, n. 4, p. 311 – 321, 2001.
- SOMMERVILLE, I.; SAWYER, P. *Requeriments engineering a good practice guide*. 1. ed. [S.I.]: Wiley, 1997.
- TERENCE, A. C. F.; ESCRIVÃO FILHO, E. Abordagem quantitativa, qualitativa e a utilização da pesquisa-ação nos estudos organizacionais. *Encontro Nacional de Engenharia de Produção*, v. 26, 2006.

Apêndices



UFRPE

Universidade Federal Rural de Pernambuco

DEINFO – Departamento de Estatística e Informática

**Protocolo de Questionário Estruturado
Relativamente a Pesquisa Exploratória Sobre Apoio
ao Aprendizado em Times de Desenvolvimento Ágil**

Thomás André da Silva Leal

Recife, Novembro/2017

DOCUMENTO DO PROTOCOLO DO QUESTIONÁRIO

1. Identificação

Pesquisador: Thomás André da Silva Leal.

Título: Graduando em Bacharelado em Ciência da Computação do Departamento de Estatística e Informática (DeInfo), Universidade Federal Rural de Pernambuco.

Área de Concentração: Metodologias Ágeis.

Linha de Pesquisa: Apoio ao Aprendizado em Times de Desenvolvimento Ágil

2. A Pesquisa

Um convite para participar de uma pesquisa de estudo, supervisionada pela Universidade Federal Rural de Pernambuco. Este documento foi desenvolvido para lhe fornecer conhecimentos sobre o estudo e informações referentes ao questionário da pesquisa. A motivação para a pesquisa surgiu devido oposição entre os benefícios decorrentes do Apoio ao Aprendizado e os problemas que se tem relacionados a Gestão dos Aprendizados. Entre os benefícios, times ágeis de desenvolvimento de software que utilizam técnicas que dão suporte ao Apoio ao Aprendizado, seus membros estão em aprendizagem contínua, novos integrantes atingem um nível de conhecimento superior com mais facilidade, afetando assim na qualidade final dos produtos desenvolvidos ou serviços prestados. Entre os desafios encarados pelos times ágeis, decorrentes à ausência da gestão de aprendizados, estão a dependência do time sobre membros que dominam um maior nível de conhecimento técnico, pouco suporte sobre novos membros da equipe e a falta de um registro sobre informações relevantes. Os resultados deste questionário serão analisados por alunos da UFRPE e utilizados para fins acadêmicos no qual servirá também como pesquisa de apoio para um trabalho de conclusão de curso.

3. Objetivo

O principal objetivo do questionário é ter conhecimento sobre qual o nível de relevância, quando se trata de apoio ao aprendizado, em times que utilizam metodologias

ágeis em seu processo de desenvolvimento de software, analisar como esses times fazem a gestão de seus aprendizados e como eles refletem seus aprendizados entre os demais integrantes do time. Através deste questionário, procura-se:

1. Realizar um levantamento de quais as técnicas de Apoio ao Aprendizado estão sendo utilizadas pelos times ágeis no mercado atual.
2. Descobrir se essas técnicas de Apoio ao Aprendizado são eficientes e dão resultados satisfatórios para os times ágeis e empresas que dão importância à gestão de aprendizado.
3. Analisar os resultados e desenvolver um guia com todas as técnicas citadas e esclarecer como e quando elas devem ser usadas.

4.Pontos Relevantes

- O participação deste questionário é voluntária.
- O questionário deverá ser respondido individualmente.
- À respeito da população:
 - I. O questionário deverá ser aplicado exclusivamente com pessoas que trabalham com desenvolvimento de software.
 - II. O questionário deverá ser aplicado exclusivamente com pessoas que trabalham com metodologias ágeis.
- O questionário não será remunerado.
- O questionário leva, em torno de, 10 minutos para ser respondido.
- Caso o entrevistado tiver dúvidas sobre alguma questão, o mesmo poderá oferecer uma respostas aproximada do que melhor lhe representa.

5.Confidencialidade

Apenas o pesquisador, indicado na seção 1 deste documento, terá acesso aos dados referentes às respostas deste questionário. Os resultados obtidos serão analisados e

apresentados em forma de um guia sem que quaisquer informações pessoais a respeito de seus fornecedores sejam divulgadas.

6.Contribuições

Este questionário terá como contribuição mostrar um panorama de como apoio ao aprendizado é abordado por desenvolvedores de software e praticantes de métodos ágeis. Uma comparação, como técnicas utilizadas e problemas enfrentados, poderá ser feita entre o que foi citado na literatura e o que se foi visto no mercado.

7.Limitações e Ameaças à Validade

O trabalho pode apresentar limitações e ameaças à validade devido a um tamanho reduzido da população que deverá ser utilizada na pesquisa. Uma pequena amostra pode ser uma possível ameaça à validade permitindo generalizações acerca do cenário real quando se trata de apoio ao aprendizado em times ágeis. Uma outra possível limitação se deve ao fato da pesquisa ser aplicada apenas numa pequena região.

Pesquisa Exploratória Sobre Gerenciamento de Aprendizado em Times Ágeis de Desenvolvimento de Software

Esta pesquisa está sendo realizada por pesquisadores da UFRPE. O principal objetivo do questionário é ter conhecimento sobre qual o nível de relevância, quando se trata de apoio ao aprendizado, em times ágeis de desenvolvimento de software. Buscamos realizar um levantamento de quais técnicas estão sendo utilizadas pelos times ágeis e saber se as mesmas técnicas são eficientes. Os resultados deste questionário serão analisados por alunos da UFRPE e utilizados para fins acadêmicos no qual servirá também como pesquisa de apoio para um projeto de conclusão de curso.

Obrigado por dedicar algum tempo para responder este questionário. Esta pesquisa é composta por 14 perguntas sendo 10 de múltipla escolha e 4 perguntas abertas de respostas rápidas. Todo o questionário leva cerca de 6 minutos para se responder.

***Obrigatório**

Conhecendo o Entrevistado

As perguntas a seguir são referentes ao nível de experiência do entrevistado em relação a desenvolvimento de software e metodologias ágeis. (4 Perguntas)

1. Qual seu nível de experiência com desenvolvimento de software? *

Marcar apenas uma oval.

- Menos de 1 ano
- De 1 a 3 anos
- De 3 a 5 anos
- De 5 a 10 anos
- Mais que 10 anos

2. Qual(s) metodologia ágil seu time utiliza no processo de desenvolvimento de software? *

Marque todas que se aplicam.

- Scrum
- XP
- Kanban
- Lean
- Outro: _____

3. **Há quanto tempo você utiliza métodos ágeis no desenvolvimento de software? ***

Marcar apenas uma oval.

- Menos de 1 ano
- De 1 a 3 anos
- De 3 a 5 anos
- De 5 a 10 anos
- Mais que 10 anos

4. **Qual sua área de atuação no momento? ***

Marcar apenas uma oval.

- Scrum Master
- Tester
- Desenvolvedor
- Líder de Projetos
- Gerente de Projeto
- Outro: _____

Problemas Relacionados ao Apoio ao Aprendizado

Desafios enfrentados pelos times ágeis para gerir seus aprendizados (10 Perguntas).

5. **Seu time utiliza técnicas que auxiliam no gerenciamento de aprendizados? ***

Marcar apenas uma oval.

- Sim
- Não

6. **De 1 a 5, o quão importante você classifica times ágeis de desenvolvimento de software utilizar técnicas de gerenciamento de aprendizados? ***

1- Nenhuma / 2- Baixa / 3- Médio / 4- Alto / 5- Muito Alto

Marcar apenas uma oval.

	1	2	3	4	5	
Nenhuma Importância	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Muito Importante

7. **De 1 a 5, o quanto você costuma aprender com outros membros do seu time? ***

1- Nunca / 2- Raramente / 3- Às Vezes / 4- Muitas Vezes / 5- Sempre

Marcar apenas uma oval.

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Sempre

8. **De 1 a 5, o quanto você costuma compartilhar o seu aprendizado com outros membros do seu time? ***

1- Nunca / 2- Raramente / 3- Às Vezes / 4- Muitas Vezes / 5- Sempre
Marcar apenas uma oval.

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Sempre

9. **De 1 a 5, com que frequência seu time costuma utilizar técnicas de gerenciamento de aprendizado?**

1- Nunca / 2- Raramente / 3- Às Vezes / 4- Muitas Vezes / 5- Sempre
Marcar apenas uma oval.

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Sempre

10. **Quais técnicas que dão apoio ao aprendizado seu time utiliza? ***

Marcar apenas uma oval por linha.

	Não Conheço	Não Uso	Uso Eventualmente	Uso Frequentemente
Apresentações Técnicas, Workshops ou Seminários	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Coaching, Mentoria ou Treinamentos Formais	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Comunidades de Prática, Grupos de Interesse ou Fóruns Técnicos	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Ferramentas (Jira, Trello, Redmine, Google Docs, etc)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Foruns de Discussão	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Pair Programming, Rotação de Pares, Programação em Grupo ou Coding Dojo	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Quadros Brancos, Quadros de Inovação, Quadros Eletrônicos, Scrum Boards, Kanban Boards	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Reuniões de Planejamento da Sprint/Projeto, Reuniões Diárias, Estimativa Coletiva	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Retrospectivas e Práticas Reflexivas	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Testes de demonstração e aceitação	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Wiki ou Repositórios de Lições Aprendidas	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

11. Há alguma técnica que não foi citada e é utilizada pelo seu time? Qual(s)?

12. Qual a principal técnica que dá apoio ao aprendizado é utilizada pelo seu time? *

13. Quais maiores problemas seu time enfrenta no gerenciamento de aprendizados? *

14. O que você recomendaria a um time ágil de desenvolvimento de software que ainda não utiliza técnicas para gestão de aprendizado?
