

DOI:10.5748/9788599693131-14CONTECSI/RF-4651

THE IMPORTANCE OF APPLICATION OF AGILE TESTS IN THE SOFTWARE INDUSTRY: AN EXPLORATORY APPROACH USING INTERVIEW

Isaac Souza Elgrably (Universidade Federal do Pará, Pará, Brasil) – isaacelgrably@gmail.com

Sandro R. B. Oliveira (Universidade Federal do Pará, Pará, Brasil) – srbo@ufpa.br

This work presents the results of an exploratory analysis using the interview research method with professionals accredited in MPT.Br model, emphasizing the use of agile tests in the implementation process of quality practices in the Brazilian software industry. The research carried out sought to identify the maturity of the agile test practice and how it is usability in the Brazilian software market by qualified professionals in test, as well as to verify the main agile test practices used, the use of TDD in the local scenario, what the main ways of training are and the acceptance of the agile tests practice by the members allocated in software projects.

Keywords: software tests, agile tests, TDD, MPT.Br, interview.

A IMPORTÂNCIA DA APLICAÇÃO DE TESTES AGÉIS NA INDÚSTRIA DE SOFTWARE: UMA ABORDAGEM EXPLORATÓRIA USANDO ENTREVISTA

Este trabalho apresenta os resultados de uma análise exploratória utilizando o método de pesquisa da entrevista com profissionais credenciados no MPT.Br, enfatizando a utilização de testes ágeis no processo de implantação de práticas de qualidade na indústria brasileira de software. A pesquisa realizada buscou identificar a maturidade da prática de testes ágeis e como é a sua usabilidade no mercado de trabalho brasileiro por profissionais habilitados em testes, além de verificar quais as principais práticas de testes ágeis utilizadas, qual a utilização do TDD no cenário local, quais as principais formas de treinamento e a aceitação da prática de testes ágeis pelos membros alocados em projetos de software.

Palavras-chave: testes de software, testes ágeis, TDD, MPT.Br, entrevista.

1. INTRODUÇÃO

O desenvolvimento de software é um trabalho que pode se tornar complexo (SOMMERVILLE, 2007). Segundo o autor, a falta de restrições naturais significa que o software pode facilmente tornar-se extremamente complexo e, portanto, muito difícil de ser compreendido. Existem muitos fatores que podem ser grandes ameaças para a qualidade final do software. No contexto nacional existem alguns modelos de qualidade de software que buscam auxiliar as equipes de desenvolvimento a diminuir estas possíveis ameaças que podem ocorrer durante o processo de criação dos seus projetos, entre eles pode-se citar o programa para a Melhoria do Processo de Software Brasileiro - MPS.BR (SOFTEX, 2016) e o modelo para a Melhoria do Processo de Teste Brasileiro - MPT.Br (SOFTEX-RECIFE, 2011), que buscam apresentar boas práticas para apoiar a implementação de um processo de qualidade na indústria de software nacional.

No processo tradicional de desenvolvimento de software existe uma sequência de etapas a serem seguidas, entre elas pode-se citar: análise de requisitos, projeto do software, implementação do software, testes, implantação, descoberta de defeitos, operação e manutenção. Segundo PRESSMAN (2006), cada fase do desenvolvimento prossegue em uma ordem estrita, sem qualquer sobreposição ou passos iterativos.

Já no processo de desenvolvimento ágil, os processos consideram uma diversidade grande de abordagens. Uma delas é a de trabalhar com pequenas iterações e entregas ao final de cada iteração (BECK, 2001). Conforme ABRAHAMSSON (2002), os processos ágeis buscam descobrir não como conter as mudanças mais cedo em um projeto de software, mas a melhor maneira de tratar mudanças inevitáveis ao longo de seu ciclo de vida. Para alcançar seu objetivo, os métodos ágeis são projetados, em princípio, para: (1) produzir a primeira entrega em semanas e alcançar *feedback* rápido e mais cedo; (2) criar soluções mais simples, de modo que, se houver mudanças, haja mais facilidade e menor volume de alterações a serem feitas; (3) melhorar continuamente a qualidade do projeto, fazendo com que a iteração seguinte tenha menor custo de implementação; (4) testar constantemente, para detectar defeitos mais cedo e removê-los com menor custo. Nesse cenário, o papel dos testes quase sempre é realizado de forma contínua e integrada durante todo o processo e deve estar devidamente planejado para que os testes atendam seus respectivos objetivos em tempo hábil ao longo do processo de desenvolvimento do software.

Tendo em vista esse cenário, este artigo visa apresentar resultados de uma pesquisa realizada com os profissionais credenciados no modelo MPT.Br. Nesta pesquisa foi utilizado o método de pesquisa da entrevista com perguntas relacionadas, principalmente, à experiência dos participantes no campo de testes, principais linguagens utilizadas, tipos de testes utilizados, principais formas de treinamento para as práticas de teste e a aceitação dos participantes dos projetos sobre as práticas adotadas, além da aceitação do próprio participante sobre as práticas de testes ágeis.

Nesta análise, foi escolhido o MPT.Br como modelo de referência, pois ele é um modelo de melhoria do processo de teste de software concebido para apoiar organizações a partir dos elementos essenciais para o desenvolvimento da disciplina de teste, inserida no processo de desenvolvimento de software (SOFTEX-RECIFE, 2011). O MPT.Br trata a melhoria do processo de teste a partir das melhores práticas relativas às atividades desenvolvidas ao longo do ciclo de vida de teste do produto de software. É importante

ênfatizar que este modelo tomou como base outros modelos de referênça de testes internacionais, como por exemplo o *Testing Maturity Model Integration* (TMMi) e o *Testing Maturity Model* (TMM). A pesquisa foi realizada no período de junho a dezembro de 2016.

Além desta seção introdutória, este artigo é composto das seguintes seções: a Seção 2 apresenta a fundamentação teórica; na Seção 3 é apresentada a metodologia da pesquisa; na Seção 4 são apresentados os resultados da aplicação da entrevista; na Seção 5 é apresentada uma síntese dos resultados das entrevistas; na Seção 6 são apresentadas análises sobre a importância dos testes ágeis no cenário nacional de desenvolvimento de software; e, finalmente, na Seção 7 são apresentadas as conclusões.

2. REFERENCIAL TEÓRICO

Sobre os trabalhos aqui considerados com maior relevância para ser utilizado nessa pesquisa, foram usadas obras que tratam de pesquisas qualitativas e quantitativas, a maioria relacionadas a teste de software, testes ágeis e à prática TDD, assim como outras referências sobre a utilização do método de pesquisa entrevista e pesquisas já realizadas sobre o modelo MPT.Br.

2.1 Teste de Software

O teste de software é um importante processo no desenvolvimento de software, no qual são executados seqüências de procedimentos que seguem especificações definidas pela equipe para determinar se o software atingiu seus objetivos. Assim, segundo MYERS (2004), dentre as técnicas utilizadas para garantir a qualidade de um software tem-se a atividade de testes de software. Testar um software é executar um programa a procura de erros. Segundo GRAHAM (2008), existem várias propostas de testes para a abordagem tradicional na literatura, onde a maioria contém as seguintes atividades: planejamento e controle, análise e *design*, implementação e execução, avaliação dos critérios de saída e reporte, e atividades de fechamento de teste.

No planejamento e controle é determinado o escopo dos testes, seus riscos, objetivos, a abordagem que será utilizada, os recursos necessários e os marcos para novas atividades e reuniões. Na etapa de análise e *design* os objetivos viram condições de testes, definindo o que será testado. Em implementação e execução, são desenvolvidos os casos de testes que serão usados, e posteriormente os testes são executados e as falhas/*bugs* são documentadas. Em seguida, são avaliados os critérios de saída e reporte, que analisam se a quantidade de testes foi aceitável e verificam a necessidade de mais testes. Por fim, há o fechamento dos testes, que avalia se o que foi planejado foi realizado, é avaliado o resultado dos testes e é armazenado o *feedback* dos participantes.

2.2 Testes Ágeis

No desenvolvimento ágil, os testes possuem um conceito mais humano e menos documental. Como na maioria dos processos ágeis, é necessária a ideia de colaborações de todos os atores participantes do projeto, como cliente, gerente do projeto e equipe de desenvolvimento, e menos páginas de documentos. Prioriza-se os indivíduos ao invés das ferramentas, a adaptação às mudanças ao invés de seguir o plano.

Para NADALETE e KRYSZCZUN (2013), os mesmos princípios usados para direcionar o desenvolvimento ágil devem ser considerados quando for aplicado teste ágil,

ou seja, testar de forma ágil exige uma forte adaptação na rotina e dinâmica da equipe de teste, em relação ao processo de desenvolvimento adotado. Esta adaptação objetiva propiciar um processo relativamente simples e que possa ser executado com grande facilidade e agilidade, cobrindo o maior número de riscos, com um nível de qualidade que seja apreciada e valorizada pelo cliente ou usuário final. Ainda segundo NADALETE e KRYSZCZUN (2013), é comum ao se aplicar teste ágil confundir o conceito de “agilidade” com “rapidez”, quando na realidade está associado à “qualidade” e “integridade” dos artefatos finais entregues. Ao se aplicar uma abordagem ágil, as iterações tendem a ser menores, assim como as entregas tendem a ser realizadas em um curto prazo de tempo. No entanto, cabe à equipe de teste em conjunto com as demais áreas garantir a consistência e a qualidade final do produto gerado, assim como a satisfação do cliente diante das suas expectativas.

O teste nos cenários ágeis está fortemente associado a prática de desenvolver um bom produto desde o início do projeto, ao contrário do panorama dos testes tradicionais que é evitar a entrega de um produto ruim.

2.3 Desenvolvimento Guiado a Testes (TDD)

O TDD – *Test Driven Development* é uma técnica que consiste no desenvolvimento de testes de unidade antes da criação do código. Estes testes servirão de guia durante o processo de desenvolvimento. Inicialmente, esta técnica foi uma prática proposta pelo método *eXtreme Programming* - XP, no entanto, tem sido utilizada por outras metodologias ágeis. BECK (2010) afirma que o desenvolvimento orientado por testes (TDD) é uma abordagem evolutiva para o desenvolvimento, onde deve-se escrever um teste antes de escrever um código de produção suficiente para realizar esse teste e realizar a sua refatoração.

Segundo MILLER (2004), o TDD reivindica um desenvolvimento incremental do código que inicia com testes, incluindo frequentemente testes de regressão. Uma das regras do TDD é a seguinte: “se você não consegue escrever um teste para aquilo que você está codificando, então você nem deveria pensar sobre codificar”.

Sobre o modo de funcionamento do TDD, segundo BECK (2010), o TDD é uma forma de programação, onde as tarefas de codificação são realizadas em pequenos ciclos de acordo com a abordagem *red/green/refactor*, onde: na fase *red* haverá a falha do teste de unidade; na fase *green*, o código é alterado de maneira simples para que ele passe com sucesso; e na fase *refactor*, o código será melhorado, caso haja necessidade, a fim de melhorar sua qualidade, porém, sempre mantendo seu comportamento.

2.4. Trabalhos Relacionados ao Tipo de Pesquisa

Como já apresentado, esta pesquisa pretende investigar a utilização de práticas de testes ágeis por profissionais credenciados ao modelo de melhoria do processo de teste brasileiro, o MPT.Br. Desta forma, assim como em (BRITO NETO, 2015), optou-se por adotar o MPT.Br como modelo de referência com o intuito de apresentar características das organizações adotantes deste modelo e considerar o grau de interesse na aderência de outros modelos, destacando principalmente o TMMi, por ser referência na disciplina de teste em nível internacional. Adicionalmente, é dado ênfase às características que destacam essas organizações na indústria de software.

A partir desta pesquisa, que tem como objetivo de investigar o perfil dos

profissionais credenciados no MPT.Br, resolveu-se analisar as características desses profissionais e suas preferências em nível técnico para a adoção de práticas ágeis de teste, seja nas empresas em que atuam desenvolvendo software, seja em empresas em que atuam fornecendo consultoria para o uso das melhorias práticas sobre testes de software.

3. METODOLOGIA DA PESQUISA

Pesquisas são ações que são utilizadas com o objetivo de buscar uma solução para um determinado problema, utilizando procedimentos racionais e sistemáticos. Assim essa pesquisa possui as seguintes classificações: quanto ao ponto de vista da natureza, essa é uma pesquisa Aplicada, que tem como um objetivo gerar conhecimento da aplicação de práticas de testes ágeis na indústria de software brasileira; do ponto de vista da abordagem do problema, a pesquisa é Quantitativa, pois traduz em números diversas opiniões e informações para posteriormente analisá-las e classificá-las, sendo utilizado um questionário para obter essas opiniões ainda na abordagem dos problemas, e também a pesquisa tem caráter Qualitativo, havendo uma pergunta descritiva para ser respondida no questionário; quanto ao ponto de vista dos objetivos, essa pesquisa é Exploratória, utilizando a técnica de entrevista com profissionais credenciados do MPT.Br, e é uma pesquisa Descritiva, utilizando os dados obtidos na entrevista para determinar como a população lida com testes ágeis nas empresas nacionais; e quanto ao ponto de vista dos procedimentos técnicos, é utilizado o Levantamento abordando os profissionais de teste do âmbito nacional e conhecendo como são suas abordagens às práticas de testes ágeis, principalmente sobre a prática do TDD.

Utilizando uma pesquisa de campo de cunho exploratório, foi estruturado um questionário baseado na experiência dos envolvidos nesse trabalho e também nos trabalhos relacionados citados na seção anterior, aplicando-o para profissionais credenciados no MPT.Br. O passo a passo para a realização dessa pesquisa pode ser visto na Figura 1, onde: no primeiro passo o problema a ser abordado foi definido, sendo ele o uso de práticas de testes ágeis pelos profissionais de teste de software da indústria brasileira; posteriormente, foi criado um questionário na ferramenta de formulários do Google para esses profissionais responderem; no terceiro passo foi definido como público-alvo da pesquisa os profissionais de testes credenciados no MPT.Br; no quarto passo o questionário foi enviado por e-mail; a seguir os dados foram coletados por meio da ferramenta utilizada; e por último os dados foram analisados e os resultados são mostrados mais nas seções a seguir.

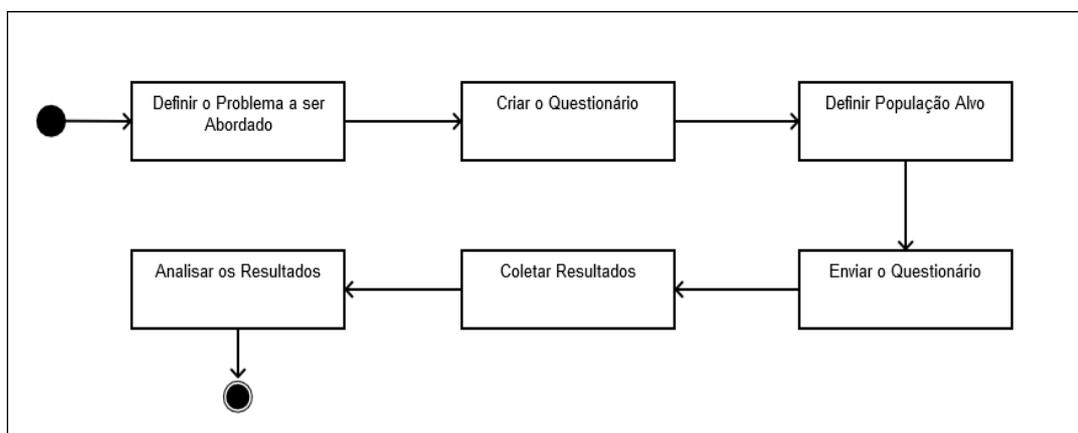


Figura 1 – Passo a passo da realização da pesquisa

Fonte: Elaboração própria (2017).

As questões presentes no questionário foram separadas em 2 grupos: (1) questões relacionadas ao perfil do profissional entrevistado; e (2) questões relacionadas aos tipos de testes ágeis realizados pelo profissional, assim como o *feedback* da equipe e da organização sobre as práticas; como pode ser visto no Quadro 1.

Quadro 1 – Questões contidas no questionário

Índice	Itens do Questionário
1	Questões Relacionadas ao Perfil do Entrevistado
1.1	Qual seu tipo de profissional credenciado MPT.Br?
1.2	Qual seu nível como profissional credenciado MPT.Br?
1.3	Qual o seu nível de conhecimento em Testes de Software?
1.4	Tempo de Experiência em Testes de Software?
1.5	Qual seu nível de conhecimento em Testes Ágeis?
2	Questões Relacionadas as Práticas de Testes Ágeis Realizados pelo Profissional
2.1	Já utilizou Práticas Ágeis de Testes em alguma implantação do MPT.Br?
2.2	Quais as Práticas Ágeis de Testes utilizadas?
2.3	Se aplicam TDD, qual(is) a(s) linguagem(ns) de programação mais utilizada?
2.4	Se aplicam TDD, qual tipo de treinamento é dado para a prática?
2.5	Se aplicam, os profissionais são estimulados a utilizar TDD ou alguma outra Prática Ágil de Teste?
2.6	Se aplicam, em relação à utilização de Práticas Ágeis de Teste, a aceitação dos membros das empresas, pode-se dizer que foi?
2.7	Se aplicam, em relação à utilização de Práticas Ágeis de Teste, a aceitação das empresas, pode-se dizer que foi?
2.8	Se aplicam, em relação à utilização de TDD, a aceitação dos desenvolvedores e testadores das empresas, pode-se dizer que foi?
2.9	Se aplicam, em relação à utilização de TDD, a aceitação do (s) gerentes de projeto e/ou coordenadores das empresas, pode-se dizer que foi?
2.10	Na sua opinião a prática ágil de TDD é satisfatória para melhorar a qualidade do produto e dos códigos das empresas?

Fonte: Elaboração própria (2017).

A população da pesquisa foi composta por profissionais que foram contatados a partir de uma consulta ao site do MPT.Br. Segundo o guia de referência do MPT.Br, este modelo é voltado para a melhoria das áreas de Teste de Software de empresas de qualquer porte. O modelo é leve e possível de ser adotado por áreas de Teste de Software para apurar o seu nível de maturidade, sem com isso onerar os seus processos anteriormente implementados. Além disso, o mesmo guia de referência afirma que o objetivo principal é garantir que áreas de Teste de Software de tamanho reduzido possam ser avaliadas e estimuladas a alcançarem níveis maiores de maturidade, sem que para isso tenham que

incorrer em altos custos operacionais (SOFTEX-RECIFE, 2011).

Assim, foram entrevistados 4 profissionais credenciados, não foi possível aumentar essa amostragem devido ao difícil acesso aos profissionais contatados. Entre as questões que foram abordadas, foi possível identificar o perfil do profissional credenciado, que pode vir a possuir 2 categorias diferentes (implementador ou avaliador), seu nível como profissional credenciado, seu tempo de experiência com teste de software no mercado brasileiro e seu nível de conhecimento sobre teste de software e práticas ágeis em teste.

Nas questões relacionadas às práticas de testes ágeis realizadas pelos profissionais, foi possível identificar: se utilizam práticas ágeis, as principais práticas utilizadas, se usam TDD como prática, principais treinamentos dados para a prática de TDD, também, foi verificado qual o *feedback* dos desenvolvedores, testadores, membros da empresa e gerentes de projeto sobre a aceitação de práticas ágeis e TDD.

Os resultados desta entrevista foram analisados de forma quantitativa e qualitativa como foi dito anteriormente, questionando os entrevistados com questões fechadas e abertas para que eles expusessem as devidas informações a respeito das práticas utilizadas e a pergunta aberta visou dar uma liberdade para que os entrevistados apresentassem mais informações que o solicitado.

4. RESULTADO DA APLICAÇÃO DA ENTREVISTA

Para a realização das entrevistas foi definido um texto de confidencialidade para os profissionais abordados, a fim de garantir que as informações fornecidas e divulgadas com o resultado da pesquisa não fossem associadas a uma pessoa específica. Portanto, não são identificados os nomes dos profissionais entrevistados nesse trabalho.

Para conhecer um pouco mais sobre os profissionais participantes credenciados pelo MPT.Br, foram definidos dois grupos de questões: um grupo tratando do perfil do profissional credenciado com o tipo de profissional, nível de profissional, entre outras; e no outro grupo tem-se questões sobre as técnicas de testes ágeis aplicadas pelo profissional e aceitação das equipes que eles trabalham sobre essas técnicas. Nas subseções a seguir são descritos em detalhes os resultados obtidos com as questões do Quadro 1. Para cada subseção será registrado um gráfico com o resultado das respostas dos entrevistados.

4.1 Questões sobre os Perfis dos Profissionais Credenciados

Este grupo de questões foi definido com o propósito de entender o perfil dos profissionais que estão respondendo ao questionário. As perguntas seguem a ordem do Quadro 1 e foram divididas em subseções, além de inserir o gráfico relativo de como foi o resultado diante da quantidade de respostas.

4.1.1 Questão 1.1

Com a primeira questão buscou-se saber “Qual o tipo de profissional credenciado MPT.Br?”. A Quadro 2 mostra quais são as possibilidades de profissionais credenciados.

Quadro 2 – Tipos de profissionais credenciados

Id.	Tipo de Profissionais Credenciados
1	Implementadores
2	Avaliadores
3	Implementadores e Avaliadores

Fonte: Elaborado pelo autor com dados extraídos de (SOFTEX-RECIFE, 2011).

Segundo a SOFTEX-RECIFE (2011), os implementadores são profissionais credenciados pela Unidade Executora do MPT.Br, com proficiência reconhecida para implantar o modelo de forma a melhor agregar valor aos processos organizacionais. Desta forma, o implementador MPT.Br deve possuir experiência e conhecimento da teoria e prática do teste de software e do modelo MPT.Br. Já os avaliadores, são profissionais credenciados pela Unidade Executora do MPT.Br, com proficiência reconhecida para avaliar o modelo de forma a melhor agregar valor aos processos organizacionais.

Para um profissional se credenciar como implementador do MPT.Br ele deve ter 2 anos como profissional de testes reconhecido e realizar o exame; e para avaliador ele deve previamente ser implementador do mesmo nível. Assim, o Gráfico 1 mostra que metade dos entrevistados 50% são Implementadores do modelo MPT.Br e os outros 50% são Implementadores e Avaliadores.

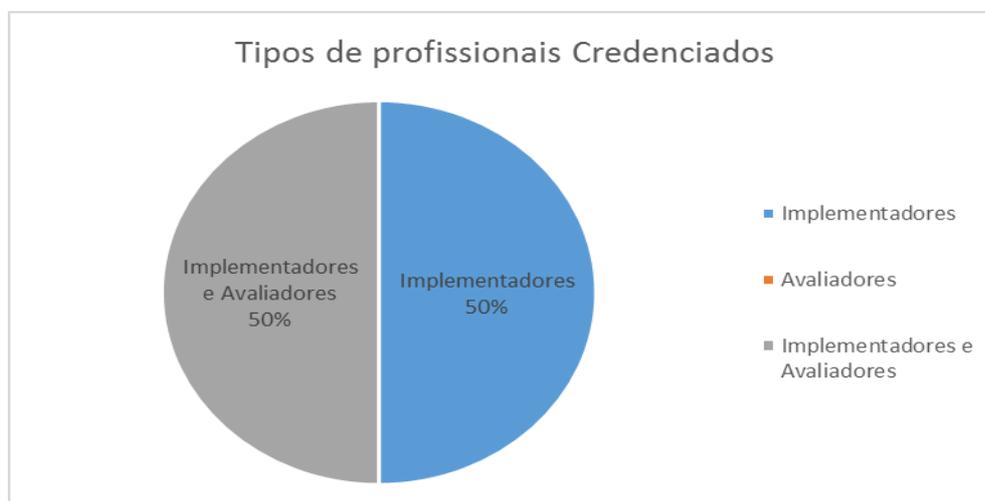


Gráfico 1 – Respostas da Questão 1.1

Fonte: Elaborado pelo autor (2017).

4.1.2 Questão 1.2

Como segunda questão, ainda falando sobre o perfil do profissional, averiguou-se “Qual seu nível como profissional credenciado MPT.Br?”. Esse nível é relacionado a até qual nível de maturidade esse profissional pode implementar ou avaliar uma empresa para ela aderir ao modelo MPT.Br. No Quadro 3 pode-se ver como são esses níveis de maturidade.

Quadro 3 – Níveis de profissionais credenciados

Níveis dos Profissionais Credenciados	
1	Parcialmente Gerenciado
2	Gerenciado
3	Definido
4	Prevenção de Defeitos

5	Automação e Otimização
---	------------------------

Fonte: Elaborado pelo autor com dados extraídos de (SOFTEX-RECIFE, 2011).

O nível *Parcialmente Gerenciado* representa o primeiro patamar de maturidade, onde analisa se a empresa tem o mínimo necessário para realizar testes aplicados aos seus projetos além dessa aplicação ser planejada e controlada. No nível *Gerenciado* a aplicação do processo de teste possui maior visibilidade. O controle já vem desde o escopo do projeto e existe um processo de gestão de mudança que faz o controle, existindo uma padronização e os processos são monitorados e controlados. No nível *Definido*, segundo a SOFTEX-RECIFE (2011), os processos padrões de teste são adotados, a garantia da qualidade é instituída de modo a auxiliar a definição dos processos, sendo definidas responsabilidades para a organização do teste e um programa de medição é implantado na organização. Também neste nível o ciclo de vida do teste é integrado ao ciclo de vida do desenvolvimento, o teste estático e de aceitação são formalizados e os procedimentos sistemáticos são aplicados para o fechamento do teste.

No nível de *Prevenção de Defeitos* é adicionado um processo de gestão de defeitos, onde defeitos são buscados mais cedo no projetos e ações proativas para suas correções são realizadas e armazenadas para futuros defeitos do mesmo tipo. Também é adicionada uma análise para verificar a eficácia do teste.

Por último, o nível de *Automação e Otimização* estabelece que o processo tenha constante melhoria e os testes sejam automatizados. Nesse momento os testes devem ser automatizados, existindo um processo de criação e uso das ferramentas de teste utilizadas e que os resultados dos testes sejam controlados estatisticamente para auxiliar a melhoria contínua.

Assim, o Gráfico 2 mostra que 50% dos entrevistados é do nível 1 e 2, 25% dos entrevistados são só do nível 1, 2 e 3 e 25% dos entrevistados são do maior nível existente, do 1 ao 5.

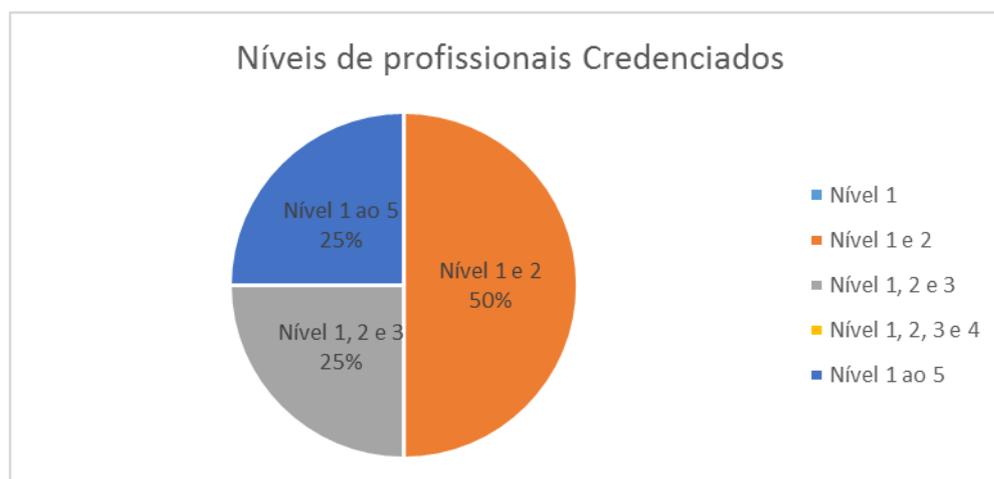


Gráfico 2 – Respostas da Questão 1.2

Fonte: Elaborado pelo autor (2017).

4.1.3 Questão 1.3

Ainda sobre o perfil dos entrevistados, a terceira pergunta foi sobre “Qual o seu nível de

conhecimento em Testes de Software?”. Esta pergunta teve como ênfase descobrir junto aos profissionais de teste, qual seu nível de conhecimento em relação ao tempo em que eles já trabalham com teste de software. Os fatores de escolha eram *Grande* (acima de 4 anos trabalhando como profissional de teste), *Médio* (de 3 a 4 anos), *Baixo* (de 1 a 2 anos) ou *Nenhum*. A partir dos resultados obtidos, todos os entrevistados têm um grande conhecimento em teste de software, como pode ser visto no Gráfico 3.

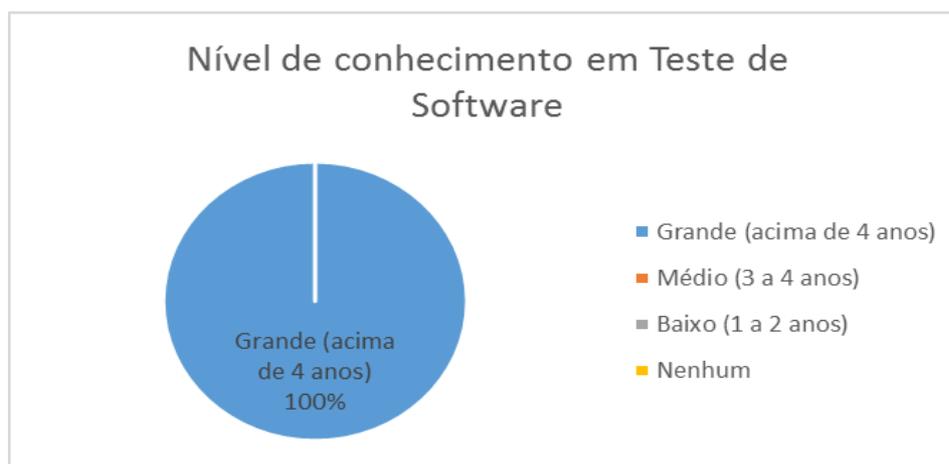


Gráfico 3– Respostas da Questão 1.3

Fonte: Elaborado pelo autor (2017).

4.1.4 Questão 1.4

A quarta questão do questionário, “Tempo de Experiência em Testes de Software?”, buscou saber o tempo (em anos) dos profissionais trabalhando em teste de software. Como resultado tem-se que todos os profissionais entrevistados têm um tempo de experiência acima de 5 anos como profissional de teste de software, como visto no Gráfico 4. Relacionando-se com os resultados obtidos da Questão 1.2, sobre o nível de cada participante como implementar ou avaliador MPT.Br, é visto que apenas 1 está no nível de maturidade mais alto. Segundo a SOFTEX-RECIFE (2011), dos 61 implementadores do MPT.Br apenas 6 deles são do nível 1 ao 5, o que pode-se entender que a pesquisa teve a participação de um experiente membro do MPT.Br.

Analisando isso, para Silva e Sampaio (2013) algumas experiências ouvidas sobre o MPS.BR, mostram que empresas buscaram o selo apenas para serem melhor colocadas em licitações. De início, a empresa aceita o investimento e passa por todo o trabalho de implementação da melhoria do processo, mas, depois que o avaliador “vira as costas” engavetam-se toda a documentação e volta à vida normal. Assim, determina-se na pesquisa que o interesse para as certificações é limitado pela maior parte dos profissionais credenciados, mesmo com alto tempo de experiência em teste de software os profissionais não buscam elevar o seu nível como implementador MPT.Br.

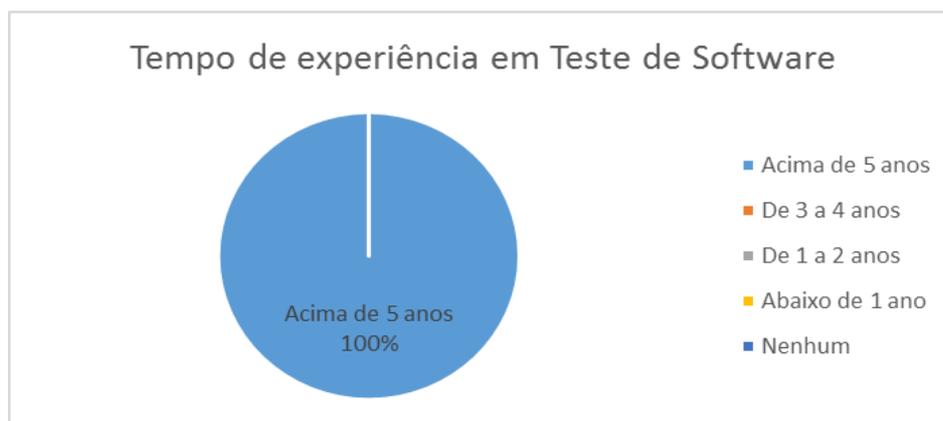


Gráfico 4 – Respostas da Questão 1.4

Fonte: Elaborado pelo autor (2017).

4.1.5 Questão 1.5

A última questão sobre o perfil dos entrevistados refere-se a “Qual seu nível de conhecimento em Testes Ágeis?”. Assim como a Questão 1.3, possui os mesmos fatores *Grande* (acima de 4 anos trabalhando como profissional de teste), *Médio* (de 3 a 4 anos), *Baixo* (de 1 a 2 anos) ou *Nenhum*. Assim como a Questão 1.3, todos os profissionais entrevistados alegam possuir grande nível de conhecimento em testes ágeis, conforme mostra o Gráfico 5.

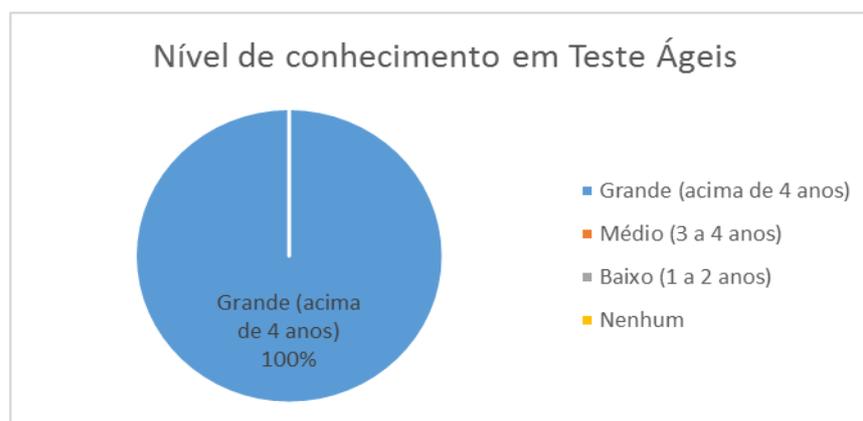


Gráfico 5 – Respostas da Questão 1.5

Fonte: Elaborado pelo autor (2017).

4.2 Questões sobre as Práticas de Testes ágeis dos Profissionais Credenciados

Este grupo de questões foi definido com o propósito de entender as práticas de testes ágeis dos profissionais que estão respondendo ao questionário. As perguntas seguem a ordem do Quadro 1 e foram divididas em subseções, onde além de inserir o gráfico relativo de como foi o resultado diante a quantidade de respostas uma última pergunta em aberto foi realizada para se fazer uma análise qualitativa sobre seu conteúdo.

4.2.1 Questão 2.1

Com o intuito de começar a delimitar o conhecimento sobre a utilização das práticas ágeis

pelos profissionais, a primeira pergunta foi se algum dos entrevistados “Já utilizou Práticas Ágeis de Testes em alguma implantação do MPT.Br?”, com possibilidades de respostas Sim ou Não. Assim, 50% dos profissionais selecionou cada uma das opções, como visto no Gráfico 6. Foi interessante analisar que ao se relacionar com as respostas obtidas na Questão 1.5, no qual todos os entrevistados afirmam ter grande experiência em testes ágeis, apenas 2 (dois) entrevistados com o maior nível profissional, e ambos implementadores e avaliadores, já utilizaram a prática de testes ágeis numa implementação do MPT.Br.

Como um dos fatores que se pode levar em consideração, segundo NADALETE e KRYSZCZUN (2013), no acompanhamento dos projetos o uso de práticas ágeis é elegível apenas quando o número de horas totais do projeto é inferior a 1000 horas. Alguns clientes são mais conservadores e estabelecem um teto de 640 a 800 horas. Projetos acima são executados dentro das práticas do RUP, modelos CMMI/MPS.BR e os testes complementados com base no Syllabus/ISTQB, CBOK/QAI, normas ISO e IEEE. Assim, isso mostra que pode haver uma limitação das empresas ao uso das práticas ágeis.

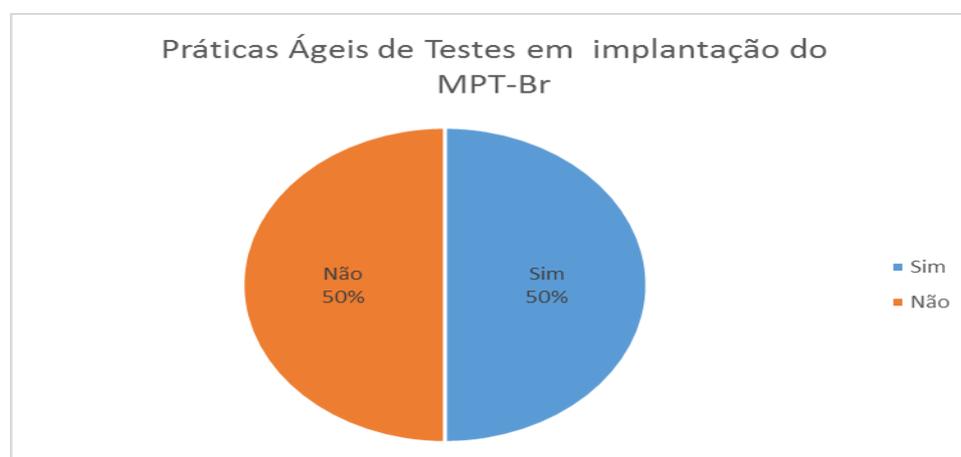


Gráfico 6 – Respostas da Questão 2.1

Fonte: Elaborado pelo autor (2017).

4.2.2 Questão 2.2

Para saber as principais práticas ágeis de teste utilizadas pelos entrevistados foi formulada a questão “Quais Práticas Ágeis de Testes utilizadas?”. Esta questão é de múltipla escolha, com a possibilidade dos entrevistados escolherem entre as seguintes opções: TDD, ATDD, BDD, Testes de Regressão, Testes Exploratórios e Testes de Usabilidade. O resultado pode ser visto no Gráfico 7.

Foi verificado que todos os entrevistados utilizam Testes Exploratórios. Segundo CAETANO (2013), o teste exploratório é, na sua definição mais básica, a criação e a execução ao mesmo tempo de um teste. Quando se realiza um teste exploratório, normalmente o testador não tem informações detalhadas sobre o que vai testar e como vai testar. O testador baseia-se na sua experiência, assim como no conhecimento que ele vai adquirindo sobre o aplicativo durante a execução do teste exploratório. A partir dessa perspectiva, pode-se afirmar que o teste exploratório é uma atividade iterativa e empírica de exploração, que exige idas e vindas num processo de investigação contínuo, onde a intuição, a criatividade e a experiência do testador são indispensáveis para garantir a

eficiência do teste.

Outro destaque foi o TDD, que faz uma cobertura sobre possíveis falhas no código ainda em sua fase de implementação, visando melhorar sua qualidade desde que ele é concebido. Conforme CAETANO (2013), diferente do que muitas pessoas acreditam, as técnicas de testes preconizadas pelas metodologias ágeis, não são limitadas a aferir apenas o código. Sem sombra de dúvida, técnicas como o TDD ajudam o desenvolvedor a eliminar defeitos no código, no entanto, o código não é a única fonte de defeitos. Em função disso, é importante destacar que os métodos ágeis oferecem um conjunto bastante sofisticado de técnicas de testes para prevenir e detectar defeitos no código, requisitos, arquitetura, assim como, defeitos originados por falta de entendimento, assunções erradas, etc.

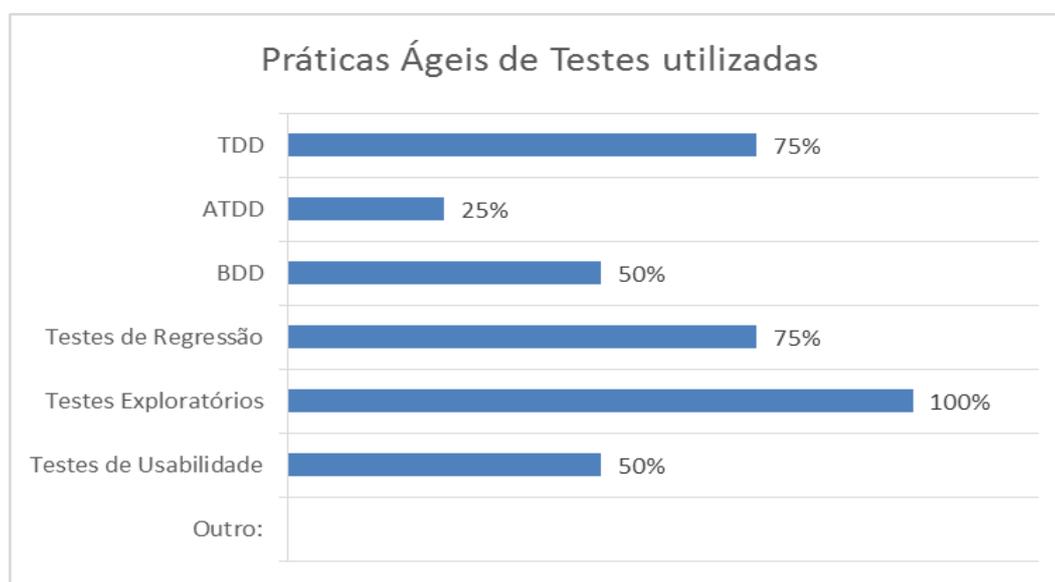


Gráfico 7– Respostas na Questão 2.2

Fonte: Elaborado pelo autor (2017).

4.2.3 Questão 2.3

BECK (2001) elaborou três fases de impacto de como o TDD afeta a qualidade do software: (1) toda vez que alguém toma uma decisão e não a testa, existe uma grande probabilidade de que esta decisão esteja errada; (2) funcionalidades de software que não podem ser demonstradas através de testes automatizados simplesmente não existem; (3) e, testes dão à chance de pensar sobre o que quer, independente da forma como a solução será implementada.

Como a maioria dos entrevistados aplicam a prática de TDD, foi formulada uma questão sobre “Qual(is) a(s) linguagem(ns) de programação mais utilizada(s)?”. Todos os entrevistados trabalham com a linguagem Java. Diante disso, vale enfatizar que, segundo CASS (2016), através do IEEE Spectrum ranking, colocou Java como a segunda linguagem mais utilizada no ano de 2016 atrás apenas do C. O próprio livro “TDD Desenvolvimento Guiado por Testes” de BECK (2010), possui todos os exemplos em Java e é tido como guia de referência da prática de TDD.

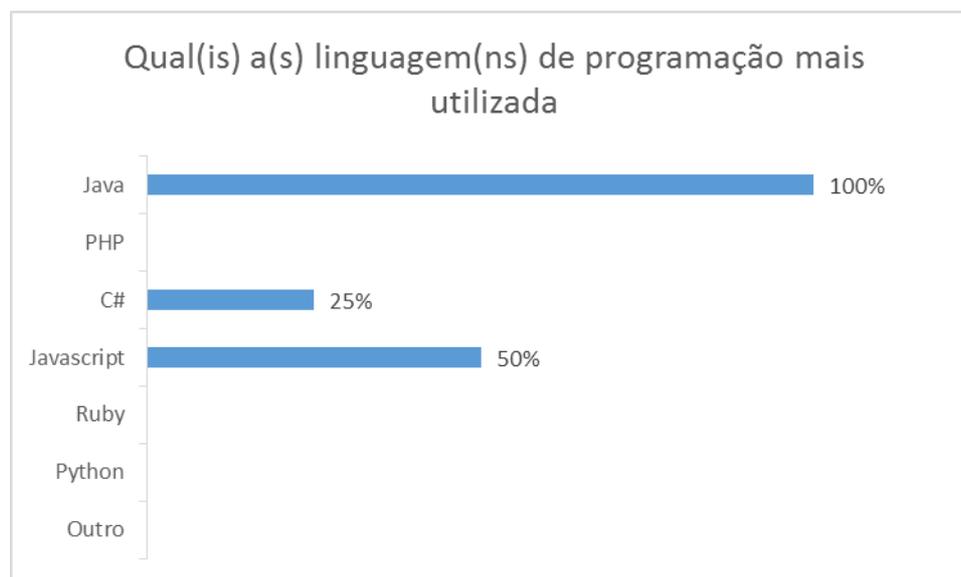


Gráfico 8 – Respostas na Questão 2.3

Fonte: Elaborado pelo autor (2017).

4.2.4 Questão 2.4

Como mencionado anteriormente, o TDD é uma das práticas de teste ágeis mais utilizadas no mercado, assim foi a seguinte questão sobre o TDD: “Se aplicam TDD, qual tipo de treinamento é dado para a prática?”. Os entrevistados tinham como opções as técnicas de treinamento: DOJO, *Mentoring*, *Coaching* e Workshop.

O treinamento de DOJO de programação teve origem em Paris, idealizado por David Thomas, o mais utilizado é o do tipo Randori Kata, que consiste em solucionar um problema utilizando a técnica de TDD, *baby steps* e pareamento, fazendo uma rotação dos pares com um tempo pré-determinado. O DOJO é tido como sucesso quando o problema é solucionado em tempo hábil determinado no início da prática.

O treinamento *Mentoring* prega que um desenvolvedor mais experiente ensine um desenvolvedor menos experiente sobre o uso da técnica ou prática de desenvolvimento, desenvolvendo assim a carreira e a evolução profissional do desenvolvedor mais jovem, além de mostrar boas práticas já utilizadas na técnica.

O treinamento *Coaching* trabalha com conceito de um instrutor contratado, quase sempre externo da empresa, ajudar os seus clientes a adquirirem ou solidificarem o conhecimento sobre determinada área, nesse caso as práticas de TDD.

O Workshop é um seminário, grupo de discussão ou fórum, que solidifica a troca de informações de determinado assunto. Os participantes interessados costumam buscar aprender as práticas através de palestras ou treinamentos de curto prazo.

Nos resultados obtidos no Gráfico 9, os entrevistados priorizam as técnicas de *Mentoring* e Workshop, por serem até mais acessíveis às equipes de desenvolvimento. O *Mentoring* pode ser relacionado com um pareamento, mobilizando poucos profissionais, e o Workshop é realizado em grande parte nos fim de semana com duração de algumas horas, o que não impacta diretamente no tempo de desenvolvimento dos desenvolvedores.

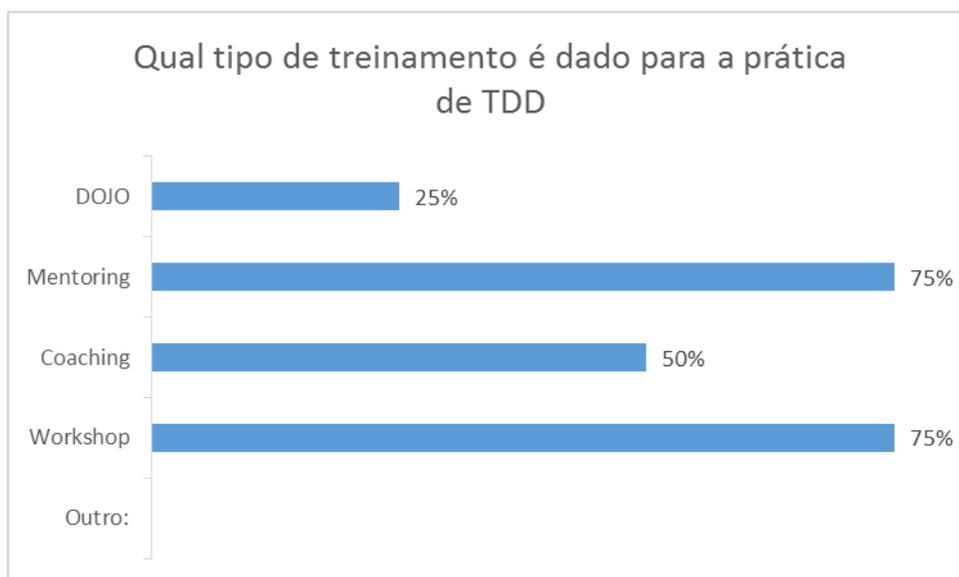


Gráfico 9 – Respostas na Questão 2.4

Fonte: Elaborado pelo autor (2017).

4.2.5 Questão 2.5

Com o questionamento sobre “Se aplicam, os profissionais são estimulados a utilizar TDD ou alguma outra Prática Ágil de Teste?”, todos os entrevistados afirmaram que sim, como visto no Gráfico 10, acontecendo um estímulo do aprendizado e continuidade do uso da técnica.

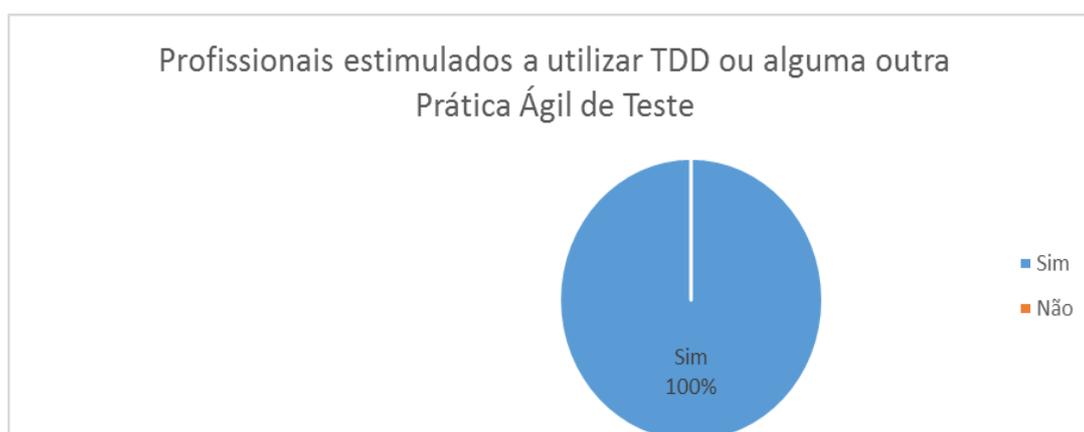


Gráfico 10 – Respostas na Questão 2.5

Fonte: Elaborado pelo autor (2017).

4.2.6 Questões 2.6, 2.7, 2.8 e 2,9

Depois de ter se averiguado as formas de utilização das práticas de testes ágeis, principalmente a de TDD que os entrevistados utilizam, fez-se necessário entender o nível de aceitação dos outros participantes do projeto. Assim, foram formuladas questões referentes à aceitação, com possibilidade escolha das opções: *Muito Boa*, *Boa*, *Média*, *Ruim* e *Péssima*.

Na questão sobre “Se aplicam, em relação à utilização de Práticas Ágeis de Teste, a

aceitação dos membros das empresas, pode-se dizer que foi?”, os entrevistados afirmaram que 75% da aceitação dos membros da empresa foi *Muito Boa* e 25% foi *Boa*, como visto, no Gráfico 11. Aqui pode-se observar uma consideração crescente de aceitação do uso da prática.

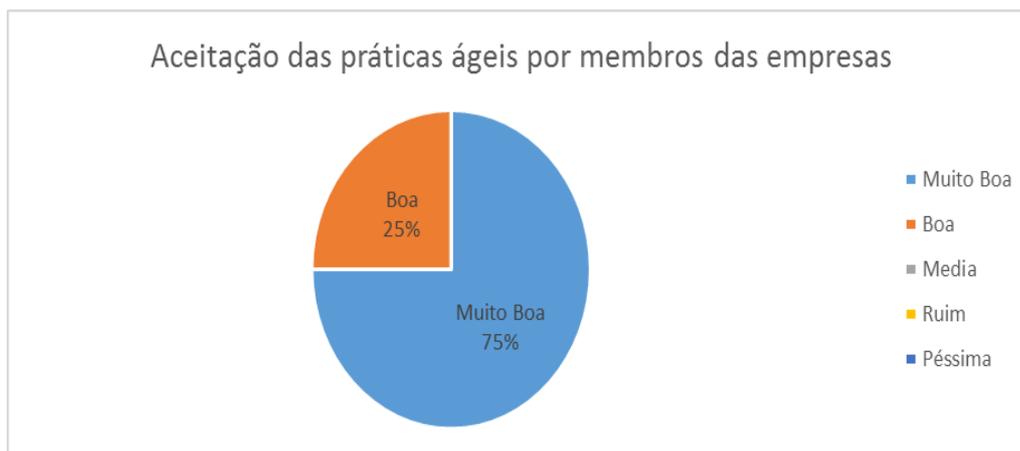


Gráfico 11 – Respostas na Questão 2.6

Fonte: Elaborado pelo autor (2017).

Na questão sobre “Se aplicam, em relação à utilização de Práticas Ágeis de Teste, a aceitação das empresas, pode-se dizer que foi?”, os entrevistados afirmaram que 50% da aceitação da empresa foi *Muito Boa* e 50% foi *Boa*, como visto, no Gráfico 12.

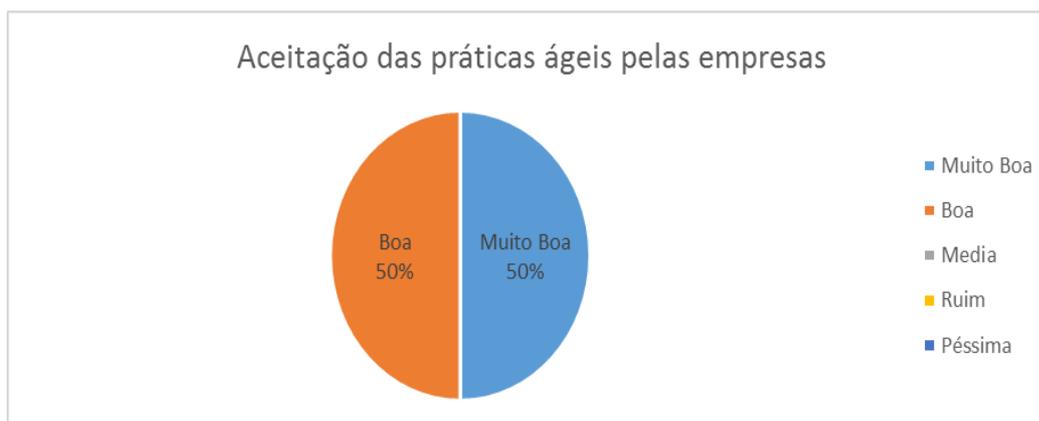


Gráfico 12 – Respostas na Questão 2.7

Fonte: Elaborado pelo autor (2017).

Na questão sobre “Se aplicam, em relação à utilização de TDD, a aceitação dos desenvolvedores e testadores das empresas, pode-se dizer que foi?”, os entrevistados afirmaram que 50% da aceitação dos desenvolvedores e testadores da empresa foi *Muito Boa*, 25% foi *Boa* e 25% foi *Média*, como visto no Gráfico 13.

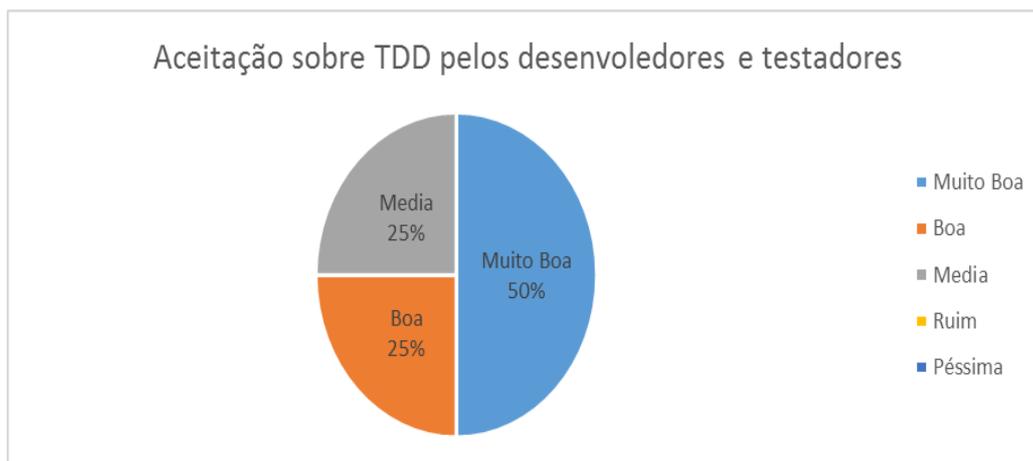


Gráfico 13 – Respostas na Questão 2.8
Fonte: Elaborado pelo autor (2017).

Na questão sobre “Se aplicam, em relação à utilização de TDD, a aceitação do(s) gerentes de projeto e/ou coordenadores das empresas, pode-se dizer que foi?”, os entrevistados afirmaram que 25% da aceitação dos gerentes de projeto e/ou coordenadores das empresas foi *Muito Boa*, 50% foi *Boa* e 25% foi *Média*, como visto no Gráfico 14.

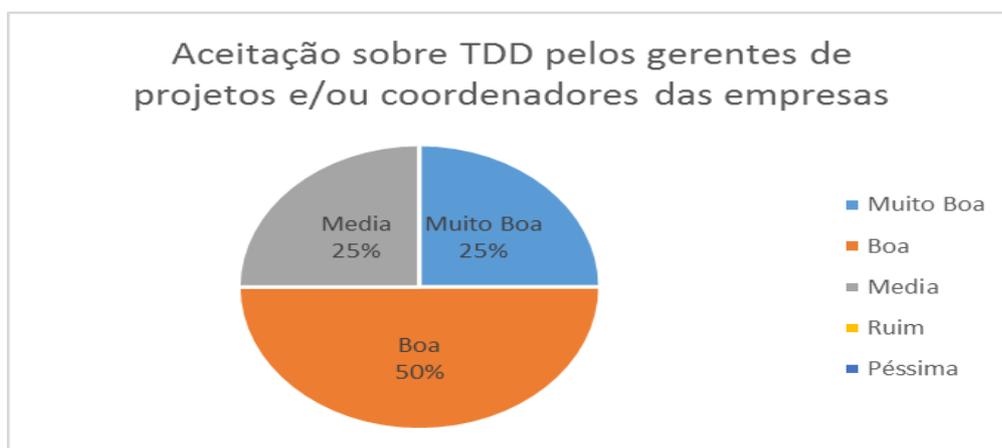


Gráfico 14 – Respostas na Questão 29
Fonte: Elaborado pelo autor (2017).

4.2.7 Questão 2.10

Por fim foi formulada uma questão aberta para ser analisada de forma qualitativa, questionando os entrevistados para que expusessem de maneira espontânea as informações a respeito de “Na sua opinião, a prática ágil de TDD é satisfatória para melhorar a qualidade do produto e dos códigos das empresas?”. Para analisar as respostas os entrevistados serão identificados com letras de A a D. Assim, o entrevistado A afirmou que a prática do TDD é satisfatória à qualidade do código, porém ele vem tendo mais

contato com o BDD nas suas implantações. O entrevistado B falou que em metodologias ágeis é fundamental que o código permaneça estável, pois agilidade significa aceitar mudanças ao longo do desenvolvimento e o TDD auxilia nisso. Já o entrevistado C afirmou que o TDD melhora a qualidade do produto e do código, pois garante que o desenvolvimento esteja direcionado para o aceite da aplicação e foca no que é mais importante nesta aplicação, isso quando o teste também já está alinhado com a equipe certificadora e tem no seu *script* os cenários de maior relevância para a área usuária. Por fim, o entrevistado D respondeu que como avaliador MPT.Br ele realmente tem encontrado muitas empresas que usam com sucesso as práticas ágeis.

5. ANÁLISE DA IMPORTÂNCIA DE TESTES ÁGEIS

Esta seção apresenta algumas análises sobre a importância dos testes ágeis provenientes dos resultados obtidos com as entrevistas e apresentados na Seção 4.

5.1 Sobre a Aceitação de Uso de Testes Ágeis pelos Profissionais de Teste

Quanto à aceitação dos profissionais de teste ao uso de práticas de testes ágeis, 100% dos entrevistados responderam que seu nível de conhecimento sobre as práticas de testes ágeis é grande, acima de 4 anos, mas apenas 50% destes entrevistados utilizaram essas práticas durante uma implantação do modelo MPT.Br.

O porquê dessa discrepância, pode vir do fato de que uma implantação nível 1 segundo RIOS (2017), “...tem o prazo estimado para a sua efetivação em torno de 6 meses, dependendo da situação da organização em relação aos seus testes”. Esses prazos curtos de convívio entre os consultores e as empresas acabam desestimulando a adoção das práticas ágeis de teste, pois estas seguem os princípios dos métodos ágeis como: valorizar os indivíduos e participantes; colaboração com o cliente; e responder às mudanças. Em um prazo curto é mais confortável a adoção de testes de natureza mais manual, como pôde ser visto, já que 100% dos entrevistados realizaram testes exploratórios em alguma implantação.

5.2 Sobre as Abordagens da Prática do TDD

Sobre a abordagem da prática de teste ágil de TDD, feita por profissionais de testes, foi realizada uma análise a nível individual no conteúdo das questões utilizadas nas entrevistas. Nela observou-se que 75% dos entrevistados já a utilizaram em alguma implantação e todos também empregaram a prática na linguagem de programação Java, 25% deles em C# e 50% em Javascript, e todos os profissionais participantes foram estimulados a usar a referida prática. Assim, como todos os entrevistados utilizaram Java como base de alguma implantação e fazendo o relacionamento com que a literatura (BECK, 2010), que utiliza em todos os seus exemplos a linguagem Java, pode-se inferir que esta linguagem ainda é a mais utilizada na prática de TDD.

A partir desses resultados é possível observar que os profissionais entrevistados preferem utilizar a prática de TDD em ambiente de desenvolvimento que utiliza Java. Tal dado também pode ser destacado por Java ser uma das linguagens mais utilizadas no mercado e pela maioria dos treinamentos e *workshops* nacionais sobre a prática de TDD utilizam a citada linguagem como exemplo. Também é possível concluir que os profissionais implementadores e avaliadores, na sua totalidade, já utilizaram a prática de TDD com Java.

5.3 Sobre a Aceitação dos Participantes do Projeto nas Práticas Ágeis de Teste e TDD

No Gráfico 16, é possível fazer uma análise de cada entrevistado sobre o que eles perceberam dos membros da empresa e o que a própria empresa achou sobre as práticas de Teste Ágeis, tendo como opções de escolha se a aceitação é: *Muito Boa*, *Boa*, *Média*, *Ruim* e *Péssima*.

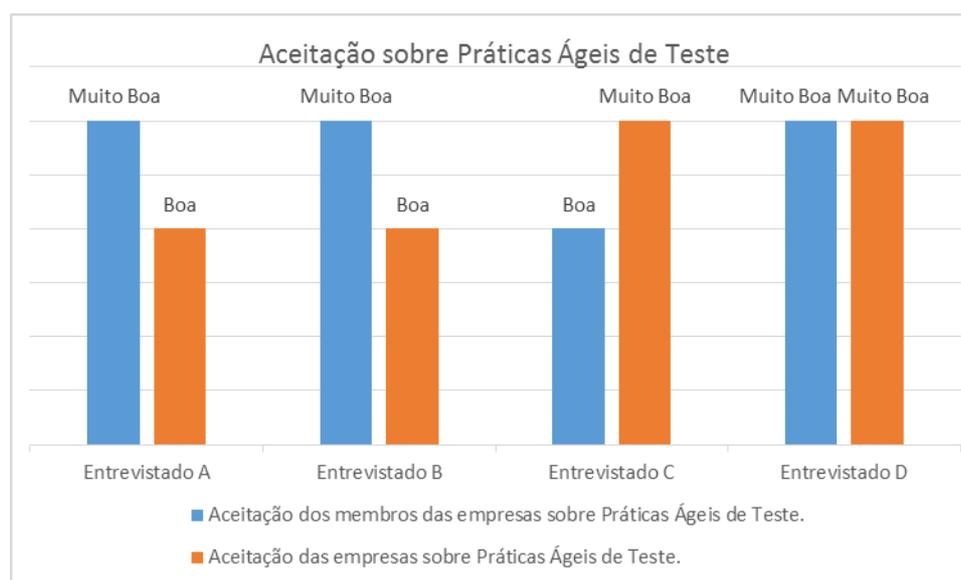


Gráfico 16 – Aceitação das empresas e membros das empresas sobre Práticas Ágeis de Teste

Fonte: Elaborado pelo autor (2017).

Sobre os membros das empresas, todos os entrevistados afirmam que a aceitação de Práticas de Testes Ágeis é *Muito Boa*. Assim, é possível verificar que as práticas ágeis estão a cada dia mais imersas no cotidiano da indústria de software nacional. Portanto, os membros da empresa já não enxergam com desconfiança essas práticas, mesmo quando voltadas para a área de testes e qualidade. No entanto, as empresas ainda possuem um pouco mais de desconfiança sobre as práticas ágeis se comparadas aos seus membros, já que 75% dos entrevistados colocaram que a aceitação foi *Boa* e apenas 25% que foi *Muito Boa*. Porém, as empresas já começam a se atualizar perante a evolução do cenário mundial de desenvolvimento de software, pois, segundo GARTNER (2015), até o ano de 2018 três em cada quatro empresas de Tecnologia da Informação no mundo estará utilizando metodologias ágeis de trabalho, padrão este que prioriza a entrega de valor, eficiência, segurança e assertividade.

O Gráfico 17 retrata uma análise individual dos entrevistados sobre a aceitação dos membros da equipe de desenvolvimento, desenvolvedores e testadores e dos gerentes de projetos ou coordenadores da empresa sobre a prática de teste ágil TDD, tendo como opções de escolha se a aceitação é: *Muito Boa*, *Boa*, *Média*, *Ruim* e *Péssima*.

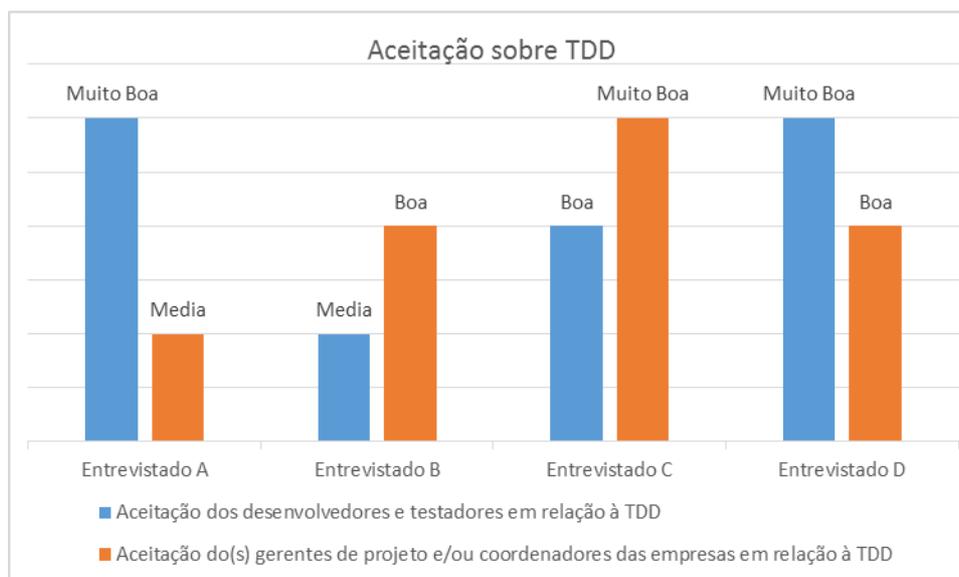


Gráfico 17 – Aceitação da equipe de desenvolvimento e dos gerentes de projetos/coordenadores de empresas

Fonte: Elaborado pelo autor (2017).

Comparado ao Gráfico 16, é possível ver que a aceitação da prática de TDD é menos regular do que a de práticas de testes ágeis. Já sobre a aceitação dos desenvolvedores e testadores, o gráfico mostra certa heterogeneidade, o que é facilmente justificável por diversas diferenças que podem haver entre as equipes de desenvolvimento que participaram das implantações dos entrevistados. Segundo LEMOS (2015), infelizmente muitos desenvolvedores não fazem o uso do TDD quando desenvolvem os seus códigos, fazendo com que percam a garantia e principalmente a qualidade do código desenvolvido. Não testar o código desenvolvido é igual a não documentar aquilo que se desenvolveu. Na grande maioria estes desenvolvedores não praticam os testes de unidade, pois não tem segurança naquilo que desenvolveu, aprendizado, falta de cultura, documentação e trabalho em equipe, a falta também de um suporte gerencial, ou seja, toda a empresa tem que “abraçar” a ideia e ver que perder tempo escrevendo testes é sinônimo de qualidade do produto que está sendo desenvolvido. Um outro fator muito importante diz respeito ao fato de muitos desenvolvedores acharem que aquilo que desenvolveram não tem a necessidade de testar, uma vez que tem uma larga experiência como desenvolvedores. Outro caso que pode ser determinante é que muitos desenvolvedores acham que não devem fazer teste, pois esse trabalho já é dos testadores e eles não deveriam perder seu tempo com isso.

Em relação aos gerentes de projeto e os coordenadores das empresas, a aceitação é ainda um pouco menor se comparada com a da equipe de desenvolvimento, com 50% de aceitação com a escolha *Boa*, 25% *Muito Boa* e 25% *Média*. Esses resultados podem ser justificados, pois em curto prazo o TDD é uma prática que costuma possuir uma curva de aprendizado um pouco mais demorada, até os membros da equipe de desenvolvimento obterem certa experiência na técnica. Então a partir daí, o código já com melhor qualidade possui quase sempre menos falhas. Existem alguns gerentes de projeto que ao invés de agirem como facilitadores para suas equipes acabam focando na entrega mais rápida do

produto, mesmo que ele possua menos qualidade. Isso acaba sendo o oposto da prática do TDD que foca em entregar um código limpo e que funciona para o seu cliente.

6. CONCLUSÃO

Essa pesquisa teve como objetivo identificar como os profissionais de teste do mercado de software brasileiro credenciados pelo MPT.Br tratam a prática de testes ágeis, principalmente o TDD, em suas implantações e como essas práticas são aceitas pelos participantes dos projetos, além de identificar possíveis preferências dos entrevistados em relação à utilização de tipos de teste, tipos de treinamentos, utilização de TDD e linguagens de programação mais utilizadas. Para tal foi utilizado o método de entrevista, utilizando um questionário de 15 perguntas: 14 delas com um caráter objetivo e a última uma pergunta aberta para os entrevistados.

Nesse contexto, foi verificado que os entrevistados consideram as práticas ágeis satisfatórias para serem utilizadas. No entanto, também foi verificado que apenas 50% dos entrevistados já utilizaram práticas ágeis em implantações, apenas pelos profissionais que eram implementadores e avaliadores. Uma justificativa para esse resultado pode vir do fato de que os profissionais que já utilizam as práticas ágeis em suas implantações também são os que possuem maior nível como profissional credenciado no MPT.Br, como pode ser visto no Gráfico 18.

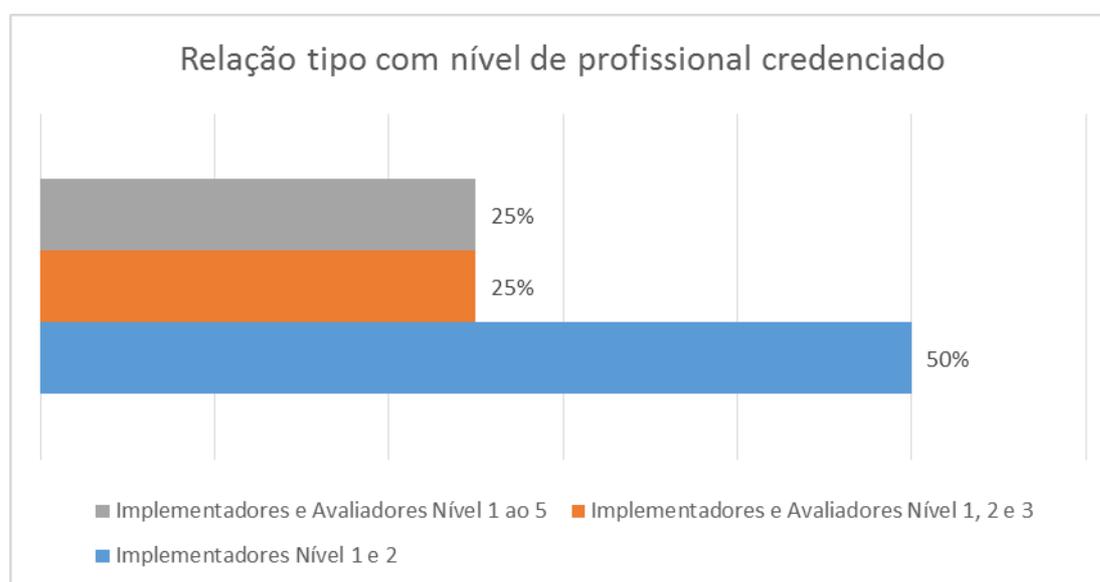


Gráfico 18 – Nível dos profissionais entrevistados

Fonte: Elaborado pelo autor (2017).

Esses profissionais, possivelmente, possuem mais experiência e participaram de mais processos de implantação. Outro ponto significativo foi a utilização da prática de TDD pelos participantes da entrevista e como a prática é bastante utilizada, em torno de 75%. Em relação às principais práticas de treinamento utilizadas pelos profissionais foi identificado que 50% dos profissionais utilizaram *Mentoring*, *Coaching* e *Workshop*, e a prática mais utilizada foi a de *Mentoring*, por 75% dos profissionais, como pode ser visto no Gráfico 19.

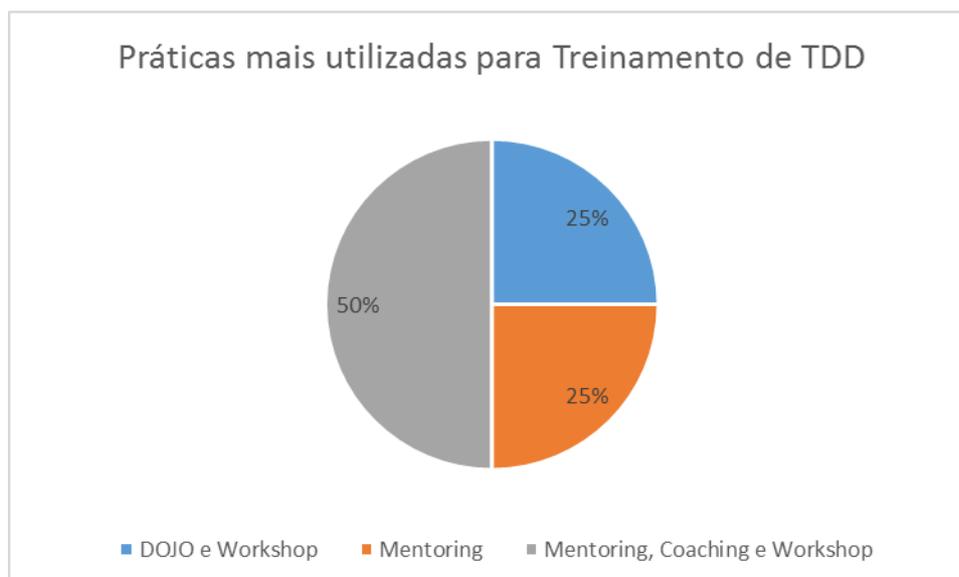


Gráfico 19 – Práticas de treinamento de TDD utilizadas pelos entrevistados.

Fonte: Elaborado pelo autor (2017).

Foi possível analisar através dessa pesquisa que as práticas de testes ágeis são do conhecimento de todos os profissionais de teste entrevistados, mas ainda não são as principais técnicas utilizadas para implantações, principalmente por profissionais com menor nível credenciado ao MPT.Br.

Os resultados que foram obtidos nessa entrevista são para verificar a importância e a utilização das práticas de testes ágeis pela indústria de software brasileira. Pretende-se, assim, desenvolver através dos dados obtidos uma ferramenta de apoio ao aprendizado de práticas de TDD para os alunos de graduação de cursos de computação, buscando prepará-los de melhor forma para as exigências do mercado nacional de qualidade e teste de software.

Finalmente, esses resultados alcançados serão de livre consulta, e almeja-se que seja possível em outro momento refazer essa experiência com ainda mais profissionais entrevistados, assim podendo conseguir novos *feedbacks* e possibilitando o enriquecimento dos resultados já obtidos.

AGRADECIMENTOS

Os autores gostariam de agradecer o apoio financeiro da CAPES pela concessão de bolsa institucional de Mestrado ao PPGCC/UFPA.

REFERÊNCIAS BIBLIOGRÁFICAS

ABRAHAMSSON, P., Salo, O., Ronkainen, J., Warsta, J. Agile Software Development Methods. Review and Analysis. Espoo. VTT Publications 478, 2002.

BECK, K. TDD Desenvolvimento Guiado por Testes. 1. ed. Porto Alegre: Bookman Editora, 2010.

BECK, K., BEEDLE, M., BENNEKUM, A. V. Manifesto para Desenvolvimento Ágil de Software. In: Manifesto para Desenvolvimento Ágil de Software, 2001.

BRITO NETO, O. N., OLIVEIRA, S. R. B. Uma pesquisa qualitativa sobre o perfil de uma amostra das organizações certificadas no modelo MPT.Br. Abakós, Belo Horizonte, v. 2, n. 2, p.03-23, 2014.

CAETANO, C. Testes em métodos ágeis. Florianópolis: Qualister - Fundação: 2007, 2013. 36 slides, color.

CASS, S. The 2016 Top Programming Languages. 2016. Disponível em: <<http://spectrum.ieee.org/computing/software/the-2016-top-programming-languages>>. Acesso em: 11 jan. 2017.

DIAKOPOULOS, N., CASS, S. The Top Programming Languages 2016. 2016. Disponível em: <<http://spectrum.ieee.org/static/interactive-the-top-programming-languages-2016>>. Acesso em: 12 jan. 2017.

GARTNER. G. Highlights Five Key Steps to Delivering an Agile I&O Culture. 2015. Disponível em: <<http://www.gartner.com/newsroom/id/3032517>>. Acesso em: 10 fev. 2017

MILLER, K. W. Test Driven Development on the Cheap: Text Files and Explicit Scaffolding. Disponível em <http://www.csc.org/northwest/docarchive/2004/augustmailing2004final.doc>.

LEMOS, G. TDD - Test Driven Development em .NET Part 1: Princípios Básicos RED, Refactor & Green. 2016. Disponível em: <<https://social.technet.microsoft.com/wiki/pt-br/contents/articles/33672.tdd-test-driven-development-em-net-part-1-principios-basicos-red-refactor-green.aspx>>. Acesso em: 11 fev. 2017.

LEVISON, M., SANTOS, W. R. Tornando TDD Simples: Problemas e Soluções para Implementadores. 2009. Disponível em: <<https://www.infoq.com/br/articles/levison-TDD-adoption-strategy>>. Acesso em: 11 jan. 2017.

MYERS, G. The Art of Software Testing. John Wiley & Sons, 2ª edição, 2004.

NADALETE, L. G., KRYSZCZUN, J. Teste ágil, como implementar? 2013. Disponível em: <<http://eliasnogueira.com/o-mundo-de-teste-de-software/capitulo-7-teste-agil-como-implementar/>>. Acesso em: 01 jan. 2017.

PRESSMAN, R. Software Engineering: A Practitioner's Approach, sétima edição, 7, Mc Graw Hill, 2006.

RIOS, E. Por que implantar o MPT.Br?. Disponível em: <http://www.emersonrios.eti.br/mpt_artigos_arquivos/Por_que_implantar_o_MPT.pdf>.

Acesso em: 01 fev. 2017.

SILVA, U. A., SAMPAIO, C. R. F. Capítulo 10: MPT.BR – Melhoria do Processo de Teste Brasileiro. 2013 Disponível em: <<http://eliasnogueira.com/o-mundo-de-teste-de-software/capitulo-10-mpt-br-melhoria-do-processo-de-teste-brasileiro/>>. Acesso em: 11 jan. 2017.

SOFTEX-RECIFE. Guia de Referência do Modelo – MPT.Br. 2011. Acessado em Janeiro/2017. Disponível em: http://mpt.org.br/mpt/wpcontent/uploads/2013/05/MPT_Guia_de_referencia.pdf.

SOFTEX. Guia Geral MPS de Software: 2016”. 2016. Acessado em Janeiro/2017. Disponível em: http://www.softex.br/wp-content/uploads/2016/04/MPS.BR_Guia_Geral_Software_2016-com-ISBN.pdf?x15632.