

Um Processo para o Monitoramento de Projetos Distribuídos de Software em Instituições de Ensino Superior do Norte do Paraná

A Process for Monitoring Distributed Software Projects in Higher Education Institutions in Northern Paraná

DOI:10.34117/bjdv7n8-343

Recebimento dos originais: 07/07/2021

Aceitação para publicação: 13/08/2021

Vanessa Faria de Souza

Mestre em Informática pela UTFPR

Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul (IFRS),
Campus Ibirubá – R. Nelsi Ribas Fritsch, 1111 - Esperança, Ibirubá – RS – Brasil

Email: vanessa.souza@ibiruba.ifrs.edu.br

Alexandre L'Erario

Doutor em Engenharia Engenharia pela Escola Politécnica da USP

Universidade Tecnológica Federal do Paraná (UTFPR), Campus Cornélio Procópio
Av. Alberto Carazzai, 1640, Vila Seugling, Cornélio Procópio – PR – Brasil

Email: alerario@utfpr.edu.br

José Augusto Fabri

Doutor em Engenharia de Produção pela Escola Politécnica da USP

Universidade Tecnológica Federal do Paraná (UTFPR), Campus Cornélio Procópio
Av. Alberto Carazzai, 1640, Vila Seugling, Cornélio Procópio – PR – Brasil

Email: fabri@utfpr.edu.br

RESUMO

É cada vez maior o número de empresas que estão distribuindo seus processos de produção de software ao redor do mundo, visando ganhos de produtividade, redução de custos e melhorias na qualidade. Engenheiros de software têm reconhecido a grande influência desta forma de trabalho e estão em busca de processos que facilitem projetos distribuídos. Além dos engenheiros também os gerentes têm enfrentado desafios em diferentes níveis no desenvolvimento distribuído de software (DDS), em especial no monitoramento do projeto. Desta forma, este trabalho tem como objetivo comprovar a seguinte hipótese: Um projeto distribuído de software pode ter o cronograma de suas atividades monitoradas, por meio de um processo que aplique a técnica do Kanban e a modelagem BPMN. O método utilizado para validação do modelo proposto foi o experimental, que simulou um ambiente real de DDS, mobilizando 82 desenvolvedores. Para uma aplicação inicial, o processo apresentou resultados positivos, portanto é eficaz para o monitoramento do cronograma das atividades de um projeto distribuído.

Palavras Chave: Desenvolvimento Distribuído de Software, Monitoramento, Processo, Kanban, BPMN.

ABSTRACT

An increasing number of companies are distributing their software production processes around the world, aiming at productivity gains, cost reduction and quality improvements. Software engineers have recognized the great influence of this form of work and are looking for processes that facilitate distributed projects. In addition to engineers, managers have also faced challenges at different levels in distributed software development (DDS), especially in project monitoring. Thus, this work aims to prove the following hypothesis: A distributed software project can have the schedule of its activities monitored, through a process that applies the Kanban technique and the BPMN modeling. The method used to validate the proposed model was the experimental one, which simulated a real DDS environment, mobilizing 82 developers. For an initial application, the process has shown positive results, so it is effective for monitoring the schedule of activities for a distributed project.

Keywords: Distributed Software Development, Monitoring, Process, Kanban, BPMN.

1 INTRODUÇÃO

Um processo de produção de software bem definido, estruturado e alinhado com os intuítos da organização e do cliente é algo imprescindível nas empresas produtoras de software em todo o mundo. Porém, apenas a adoção de um bom processo de software não garante o sucesso do projeto, acima de tudo se faz necessário planejar e monitorar todas as atividades envolvidas durante o ciclo de vida do produto (SCHARFF, 2011). A escolha em se realizar o monitoramento no processo de software possibilita aumento na velocidade do desenvolvimento e da qualidade, facilita o acompanhamento, minimiza os riscos e melhora a relação entre os colaboradores e com o cliente (SCHARFF, 2011). Porém, caso não haja sua execução no processo de software algumas consequências negativas podem ocorrer como: baixa produtividade, retrabalho, insatisfação do cliente e realização de atividades desnecessárias (SCHARFF, 2011). No Desenvolvimento Distribuído de Software (DDS), os cuidados quanto a monitorar o projeto são ainda mais fundamentais, pois o produto é desenvolvido por equipes remotamente distantes.

Esta dispersão das equipes dificulta o trabalho realizado por elas, desta forma métodos aprimorados para monitoramento dos projetos de software neste ambiente tornam-se cada vez mais necessários. São vários motivos que conduzem as organizações a se engajarem ao DDS, dentre eles destacam-se: o custo de produção, a proximidade com clientes e a disponibilidade de mão de obra (SCHARFF, 2011). Segundo Herbsleb (2003), não é incomum para um projeto de software de grande porte ter equipes em mais de um local, muitas vezes em mais de um continente. Com o crescimento do DDS surgiram novos desafios para os projetos de software até então inexistentes ou amenos. O DDS trouxe novos desafios para as equipes de desenvolvimento: separação geográfica, fusos

horários diferentes, culturas diferentes e barreiras linguísticas (LESCHER, 2014). Herbsleb (2003) neste sentido, afirma que tais particularidades passaram a ser consideradas como fatores de risco (diferenças culturais, fuso horário, idioma). Para enfrentar esses obstáculos os gestores da área de produção de software precisam adequar-se à esta modalidade e encarar estes desafios de forma a diminuir os problemas no monitoramento do processo.

No contexto de monitoramento de projetos de software, uma técnica ganhou notoriedade a partir da ascensão das metodologias ágeis, o Kanban (AHMAD; MARKKULA; OVIO, 2013). Este surgiu para atender necessidades da indústria em um momento em que a qualidade passou a ser um fator crítico da produção, o que era alcançado, por meio de manobras que visavam exclusivamente o atendimento a novas demandas de Mercado (AHMAD; MARKKULA; OVIO, 2013). Apesar de sua essência ser baseada em uma circunstância distinta a de projetos, seu objetivo e sua abordagem são compatíveis às principais necessidades em um monitoramento eficiente. Desta forma, os cartões Kanban representam os pacotes de Atividades, que neste trabalho são representados, por meio *Business Process Modeling Notation* (BPMN), e alocados nas suas respectivas fases do ciclo de vida do projeto, por meio do painel de monitoramento (AHMAD; MARKKULA; OVIO, 2013). O BPMN será utilizado para realçar a visão geral do escopo do projeto, esta é compreensível por usuários do negócio, analistas, desenvolvedores técnicos e também por aqueles que gerenciam e monitoram o processo (WHITE, 2015).

Quanto ao emprego do Kanban em projetos de software, as principais vantagens segundo Ahmad et al (AHMAD; MARKKULA; OVIO, 2013) são além da organização do escopo de trabalho, viabilizando um monitoramento mais eficiente, o aprimoramento das comunicações e da integração do projeto perante à equipe, que por ser distribuída, enfrenta problemas desta ordem (AHMAD; MARKKULA; OVIO, 2013). Diante da possibilidade da aplicação do Kanban em projetos, a grande motivação para utilização desta técnica neste trabalho deu-se devido ao seguinte fato, mesmo com sua crescente adoção em projetos de software, esta técnica não é aplicada com frequência no ambiente de desenvolvimento distribuído.

Ahmad, Markkula e Ovio (2013) desenvolveram uma revisão sistemática de literatura a respeito do Kanban e investigaram 492 pesquisas que abordam o tema, a qual constatou, que no que diz respeito ao contexto, 84% dos estudos foram relacionados a

projetos centralizados, e o restante 16%, não identificaram claramente o ambiente de trabalho. O que indica que a maioria dos estudos eram realizados em configurações de equipe co-localizada. Além disso, nenhum estudo, pesquisa e/ou relatório técnico, investigado pelos autores, relatou o uso de Kanban em ambientes distribuídos.

Desta forma, diante dos fatos apresentados, este trabalho pretende comprovar a seguinte hipótese: Um projeto distribuído de software pode ter o cronograma de suas atividades monitoradas, por meio de um processo que aplique a técnica do Kanban e a modelagem BPMN. Para comprovar tal hipótese será utilizado o método experimental, espera-se que com isto a técnica Kanban, pouco explorada no DDS seja vista como vantajosa por gerentes de projetos distribuídos. Para tanto, este artigo está organizado da seguinte forma, esta seção tratou da introdução ao tema, a seção 2 aborda a revisão bibliográfica, na seção 3 está especificado o processo proposto. A seção 4 traz a metodologia utilizada no trabalho, enquanto que a seção 5 sistematiza os resultados alcançados, e por fim a seção 6 aborda conclusões do trabalho.

2 REVISÃO BIBLIOGRÁFICA

2.1 PROCESSO

O conceito de processo, segundo a NBR ISO 9001: 2008 corresponde ao “conjunto de atividades inter-relacionadas ou interativas que transformam insumos (entradas) em produtos (saídas)” (ABNT, 2008). Além dos insumos – entradas e resultados – saídas, um processo possui, ainda, mais dois elementos fundamentais para sua execução: o controle e os recursos. Assim, um processo típico é formado por quatro elementos como salienta Cruz (2005): *Entradas*: são os insumos, materiais ou informações que servirão de subsídio para o processo ser executado; *Saídas*: são os produtos ou serviços produzidos pelo processo; *Controle/Motoramento*: é o responsável pela verificação do cumprimento de diretrizes, métodos, objetivos, procedimentos e padrões na execução do processo; **Recursos**: representam a provisão do que se faz necessário para a execução do processo: recursos administrativos, financeiros, humanos, materiais, informações, entre outros.

2.2 KANBAN

Kanban é técnica de monitoramento por meio de cartões ou registros em painéis, ou seja, um monitoramento de toda a produção, puxando a responsabilidade para si. O Kanban é utilizado em vários seguimentos industriais, também na produção de software, este tem seus conceitos principais adaptados para condizer com o processo de

desenvolvimento de software (AHMAD; MARKKULA; OVIO, 2013). O Kanban, dentro desse contexto, procura aperfeiçoar os processos, as equipes e projetos. É útil para empresas que estão procurando melhorar constantemente seus processos, ao passo que melhoram também sua produtividade e sua relação com os clientes. Esse método começou a crescer após sua adoção na conferência Agil, em agosto de 2007. Segundo Ahmad, Markkula e Ovio (2013) a primeira aplicação para Engenharia de Software foi na empresa Microsoft em 2004.

O Kanban fornece visibilidade ao processo de software, porque mostra o trabalho atribuído de cada desenvolvedor, comunica claramente as prioridades. Além disso, seu objetivo é minimizar WIP, desenvolver apenas os itens que são solicitados. Isto produz um fluxo constante de itens de trabalho divulgados aos clientes, como os desenvolvedores focam apenas naqueles poucos itens em determinado momento. O Método visa adaptar rapidamente o processo usando feedbacks em ciclos mais curtos. Os resultados da aplicação da técnica Kanban em produção de software é altamente positivo, correspondendo às vantagens conseguidas na indústria de transformação (AHMAD; MARKKULA; OVIO, 2013). Kanban é muito adaptativo. Com isso as equipes que o adotam precisam estar atentas ao processo aplicado para visualizar locais de melhoria e adaptações para que o processo possa fluir de forma satisfatória. Além de, facilitar a visualização e gestão, bem como dirigir os membros da equipe, cooperar e comunicar de forma mais eficiente e eficaz (AHMAD; MARKKULA; OVIO, 2013).

2.3 BPMN

A simbologia da BPMN permite criar modelos de processos de negócio – Business Process Diagrams (BPD) para finalidades de documentação e comunicação. Esses modelos seguem uma notação padrão, desenvolvida pelo Instituto de Gestão de Processos de Negócio – The Business Process Management Initiative (BPMI) e foi lançada publicamente em maio de 2004. A especificação da BPMN representa mais de dois anos de esforços do grupo de trabalho do Business Process Management Initiative (BPMI). A BPMN foi selecionada, dentre as notações para modelagem de processos, para ser utilizada neste trabalho, porque esta foi desenvolvida para ser de fácil utilização e entendimento e também fornece a habilidade de modelar processos de negócios complexos.

Assim apesar de sua manipulação ser simples, ela é capaz modelar os mais diversos tipos de processos. A BPMN é uma notação que tem como propósito a geração

de um diagrama de processos de negócio chamado de Business Process Diagram (BPD) (BPMN, 2015). O BPD é construído, por meio de um conjunto básico de elementos gráficos. Estes elementos permitem o desenvolvimento de diagramas que são, normalmente, familiares para a maioria dos analistas de negócio (WHITE, 2015).

2.4 MONITORAMENTO DE PROJETOS EM DDS

Segundo L'Erario (2009), o DDS ocorre quando vários sites cooperam e/ou colaboram para desenvolver um mesmo produto ou parte dele estando distantes geograficamente. Neste cenário, a complexidade do processo se amplia, assim como os desafios inerentes ao desenvolvimento de software. Sengupta (2010) apresenta como dificuldade básica do DDS a inabilidade de monitorar o processo efetivamente devido à distância geográfica, diferenças culturais e fuso horário. Mesmo em projetos desenvolvidos em um contexto intra-organizacional, em que a priori a linguagem, a cultura e a organização seriam as mesmas, existem dificuldades de monitoramento (SENGUPTA, 2010). Neste contexto, ao se analisar as principais abordagens de gerência de projetos, nota-se que boa parte de suas composições tratam do monitoramento, contudo este termo vem atrelado ao controle, o qual não faz parte do escopo deste trabalho. Por exemplo o *Project Management Body of Knowledge (PMBOK)*, apresenta 10 processos de monitoramento e controle de um total de 42 processos propostos. O *Association of Project Management Body of Knowledge (APMBOK)* insere planejamento, medição, monitoramento e ações corretivas em um grande ciclo de controle.

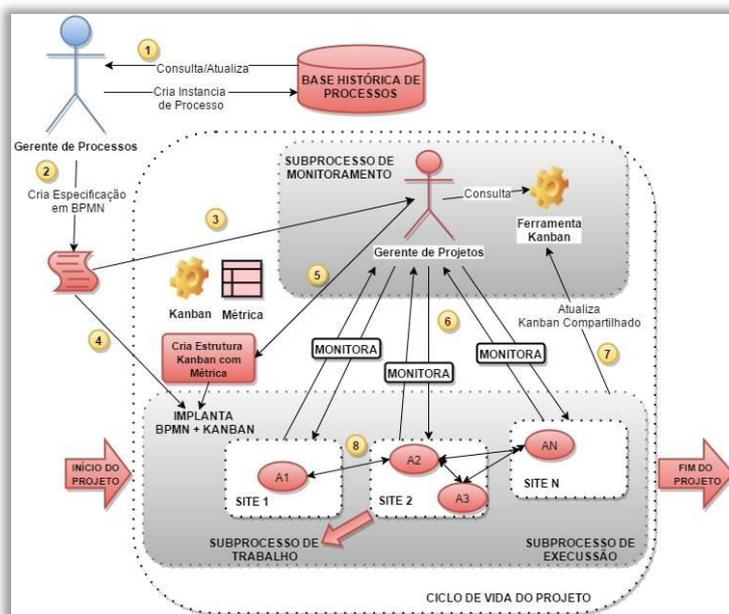
Levando em consideração que muitos dos problemas em projetos de software referem-se a monitoramento, é fácil entender a razão das significantes parcelas dedicadas a monitoramento das abordagens de gerência de projetos. Jones (2006) destaca que entre as principais causas de problemas em projetos de software estão: estimativas inadequadas, relato de status inadequado, ausência de dados históricos de projetos similares, e processos de controle de qualidade inadequados. O monitoramento é função básica na execução de qualquer projeto, pois serve para alertar possíveis problemas antes que eles se tornem irreversíveis. Desta forma, nota-se sua importância em projetos de software, fator que é indispensável no DDS. Wiredu (2005) afirma que um projeto neste contexto possui diversos obstáculos para o gerente realizar suas atividades, tais como: a dificuldade de planejar e criar um cronograma devido à separação temporal das pessoas, a necessidade de considerar custos de viagens e de investimento em infra-estrutura, a dificuldade de agrupar a equipe devido a diferenças culturais e de idiomas. Desta forma,

são necessários processos de monitoramento com uma abordagem específica para o DDS, para que assim os desafios pertinentes a sua realização não levem o projeto ao fracasso.

3 PROCESSO DE MONITORAMENTO DE PROJETOS DISTRIBUÍDOS DE SOFTWARE

O processo de monitoramento proposto como instrumento para comprovar a hipótese levantada, se caracteriza como um conjunto de ações que podem ser usadas para auxiliar no monitoramento de projetos distribuídos de software. Este procura sanar os desafios do DDS. Para isto o processo se utiliza da técnica Kanban e da modelagem BPMN. A Figura 01 apresenta o processo com todos os seus subprocessos e elementos. O processo é composto por três subprocessos: o subprocesso de execução, o subprocesso de trabalho e o subprocesso de monitoramento das atividades. Também por elementos, o gerente de processos, o gerente de projetos, a base histórica, o Kanban, o BPMN, a Métrica e as atividades em si. Além destes itens o processo também necessita da execução de uma série de etapas, denominadas neste trabalho, ações do processo de monitoramento. Tanto os subprocessos, os elementos como as ações são descritos em sequência.

Figura 1 – Modelo de monitoramento e controle



Fonte: Autores

Subprocesso de execução: Este é responsável por gerar um produto final, tem como objetivo efetivamente a produção do software. Caracterizado como as diversas etapas necessárias (subprocessos e atividades) para se chegar ao produto.

Subprocesso de trabalho: Este denota uma unidade do subprocesso de execução e os seus resultados, é uma pequena parte do processo como um todo, pode-se defini-lo então como uma atividade e os artefatos gerado por esta. Este deve oportunizar também um ambiente compartilhado para a visualização dos artefatos. O subprocesso de trabalho pode ser configurado em um ambiente (online e/ou em rede) em que os indivíduos pertencentes ao projeto acessam a especificação das atividades/tarefas que devem realizar, e depositam os artefatos que cada atividade executada gerou.

Subprocesso de monitoramento das atividades: o gerente de projetos é o responsável por este subprocesso, que trata do acompanhamento das atividades executadas no decorrer do processo (que será apoiada por uma ferramenta online que implementa a técnica do Kanban).

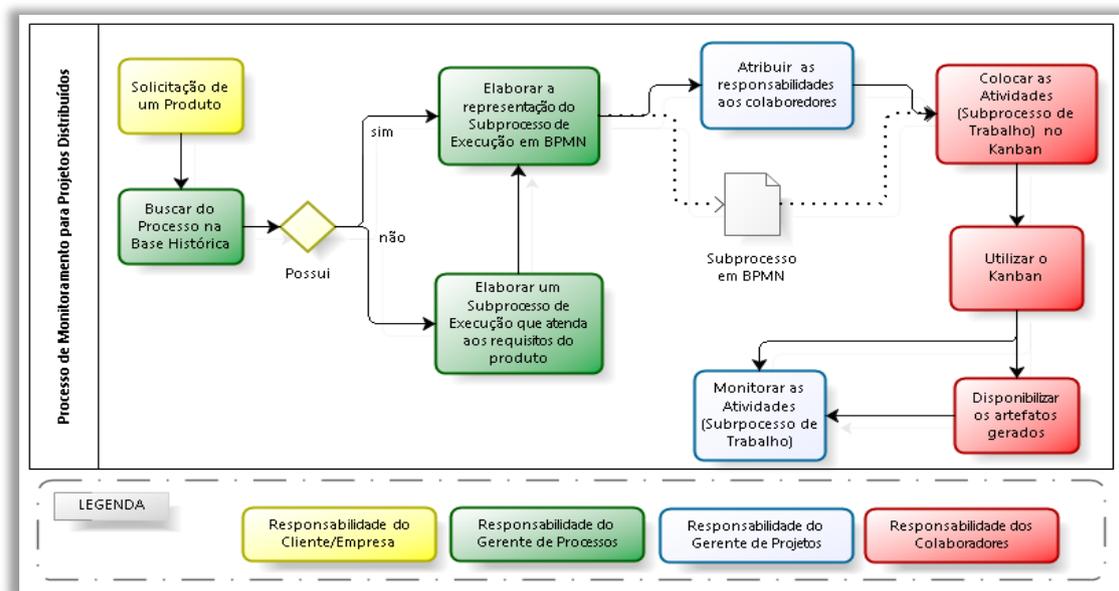
Os subprocessos descritos são à base do processo proposto, e têm como objetivo principal sanar o problema relacionado à dificuldade de monitoramento dos projetos em ambientes de desenvolvimento distribuído, contudo além destes, o processo possui também alguns elementos relacionados. O subprocesso de execução por exemplo, possui os elementos para elaboração do produto, que são as atividades. Cada atividade pertencente ao subprocesso de execução junto aos artefatos que geram é um subprocesso de trabalho. As atividades (A1, A2, A3, ..., An) são interativas, podem ser refeitas ou reelaboradas no decorrer do projeto. A execução destas atividades é de responsabilidade dos colaboradores, ou seja, dos sites de desenvolvimento, que correspondem a outro elemento do processo, além do desenvolvimento das atividades é deles a responsabilidade de atualizar a ferramenta Kanban.

O projeto é iniciado assim que há uma solicitação, esta pode ser externa ou interna a organização, o subprocesso de execução é acessado na base histórica da empresa, que é um elemento essencial neste processo, para que seja iniciado. Caso ela ainda não possua, o processo de execução adequado ao produto que se deseja desenvolver, este deve ser sistematizado pelo gerente de processo. O gerente de processos, é uma pessoa especializada em processos de software baseados nas boas práticas. Este apenas interage com o processo de software, corrigindo e especificando etapas no mesmo, melhorando os que a empresa já faça uso, ou ajudando a criar processos de execução melhorados, ele também especifica o subprocesso de execução em BPMN como pode ser percebido na Figura 1.

Diferentemente do gerente de processo, o gerente de projetos é o responsável pela equipe, também por todas as unidades de trabalho que devem ser realizadas para o

desenvolvimento do produto, analisa em especial os artefatos gerados e sua adequação. Ele é responsável também por monitorar o cronograma de execução das atividades e realizar as interações com as equipes e com os clientes. Ele monitora o projeto em consultas frequentes ao Kanban compartilhado, como descrito na figura 01, e baseia-se em uma métrica, que este aplica como parâmetro. A métrica utilizada neste trabalho é o Ideal Day. Ele também atribui os papéis e responsabilidades a cada site, ou individualmente, esta ocorre geralmente no início do projeto, podendo ser alterada se necessário. Por causa destas atribuições, o gerente de projeto em ambiente distribuído necessita do auxílio ferramentas para monitoramento, para este processo é utilizado a implementação da técnica Kanban, o BPMN e uma métrica para tempo de execução.

Figura 2 – Ações do Processo



Fonte: Autores

Para que o processo de monitoramento proposto seja implementado de maneira adequada, pelos interessados em aplicá-lo, estes devem seguir algumas especificações as ações do processo. Na Figura 02 são apresentadas as ações para implementação do processo. O processo de monitoramento, possui nove ações e um artefato. Das quais, uma delas está sob responsabilidade de um agente externo ao projeto, que é a solicitação do produto. O gerente de processo se responsabiliza por três ações e o artefato, verificar se existe um subprocesso de execução que atenda as especificações do produto na base histórica da empresa. Caso ela não possua, o gerente de processos deve especificar alterações em algum já existente, ou com base em padrões de qualidade e em sua

experiência especificar um processo que seja adequado, posteriormente ele faz a representação deste em BPMN, que trata do subprocesso de execução.

As ações que estão, sob incumbência do gerente de projetos são apenas duas, a atribuição das responsabilidades dos colaboradores, assim como monitorar as atividades e os artefatos resultantes. Enquanto que, os colaboradores devem preencher a ferramenta, em que é implementada a técnica Kanban, assim como atualizá-la frequentemente e disponibilizar os artefatos a cada atividade concluída. Com a realização destas ações o processo de monitoramento é implementado. Neste trabalho ele representa um instrumento para comprovação da hipótese levantada, contudo ele pode ser aplicado em outros contextos, e até mesmo com objetivos de monitoramento mais abrangentes.

4 MÉTODOS E PROCEDIMENTOS

O método adotado por este trabalho com objetivo de validar o a hipótese é o experimental. De acordo com Travassos [13] a experimentação é o centro do processo científico; somente experimentos verificam as teorias e podem explorar os fatores críticos e dar luz ao fenômeno novo, para que as teorias possam ser formuladas e corrigidas. O processo experimental, utilizado é o proposto por Wohlin *et al.* [14] este consta de cinco subprocessos cada um com suas respectivas etapas, estes são: a (1) definição do escopo, (2) o planejamento, a (3) operacionalização do experimento, a (4) análise e interpretação dos dados e a (5) sistematização dos resultados . A descrição dos subprocessos 1, 2 e 3 são realizadas nesta seção, enquanto que o subprocesso 4 é apresentado na seção V e a sistematização dos resultados (considerações finais) do experimento é exposto na seção VI.

4.1 DEFINIÇÃO DO ESCOPO

Na definição do escopo é definido o contexto de aplicação do experimento: o qual é “Um processo de monitoramento, desenvolvido a partir da utilização da técnica do Kanban e da notação BPMN, causa alterações, positivas, no decorrer de um projeto distribuído, do ponto de vista dos investigadores, analistas e desenvolvedores de sistemas (graduandos ou pós-graduandos), durante o desenvolvimento de um projeto de software distribuído”.

4.2 PLANEJAMENTO

O planejamento permite que se tenha uma visão mais clarificada de como irá decorrer o desenvolvimento do experimento e possui 7 etapas descritas na sequência:

Seleção do Contexto: A seleção da dimensão determina o contexto no qual são aplicáveis os resultados obtidos no experimento e também a possibilidade de generalização dos mesmos, este trabalho está compreendido na dimensão *Problema Específico VS. Problema geral*, que passa de um problema específico para um geral.

Formulação das Hipóteses: As hipóteses que serão testadas neste experimento são: **H0:** (Corresponde a hipótese nula a qual pretende-se negar). Não é possível monitorar o cronograma de atividades de um projeto de distribuído de software utilizando um processo que aplica a técnica Kanban e a modelagem BPMN. **H1:** Um projeto distribuído de software pode ter o cronograma de suas atividades monitoradas, por meio de um processo que aplique a técnica do Kanban e a modelagem BPMN.

Seleção das Variáveis e Métricas: No Quadro 1 estão descritas as variáveis independentes e dependentes identificadas neste experimento.

Quadro 1 – Variáveis independentes e dependentes

Variáveis Independentes	Variáveis Dependentes
Experiências dos envolvidos	Utilização do Mecanismo Kanban
Cronograma das Atividades	Desenvolvimento das Atividades
Modelagem do projeto em BPMN	

Fonte: Autores

Ainda nesta etapa foi possível identificar a métrica que é utilizada no decorrer do experimento, esta é essencial no auxílio da validação da hipótese proposta e diz respeito ao cronograma de execução das atividades. Isto posto, a métrica utilizada foi *Ideal Day*, ela é responsável por realizar estimativas de forma ágil, sendo aplicada para planejar o projeto e iterações. Segundo Martins [15] a *Ideal Day* corresponde à quantidade de trabalho que um profissional da área consegue concluir em um dia dedicado a este. De acordo com Martins [15], a velocidade é calculada a partir do número de horas que a equipe gasta para implementar um trabalho equivalente a um *Ideal Day*. O mesmo autor define que, para efetuar o cálculo dos dias estimados é necessário utilizar a seguinte fórmula:

$$DE = \frac{IED}{1 - IED_{REAL} \%}$$

Onde:

DE: quantidade de dias estimado para concluir a tarefa;

IED: prazo necessário para implementar o item, esse prazo é definido pela equipe;

IED_REAL%: percentual que indica a estimativa de quanto tempo do dia o desenvolvedor ficará dedicado a implantação do item.

Seleção dos Sujeitos: Para o experimento desenvolvido neste trabalho, os participantes se caracterizam como alunos de graduação e pós-graduação lato e stricto sensu, que estudam em Universidades da região norte do Paraná - Brasil.

Seleção do Design: O experimento desenvolvido neste trabalho utiliza o design de Completamente Aleatorizado com um único Fator, apropriado quando somente um fator experimental está sendo estudado, neste caso o cronograma das atividades realizadas no projeto (CATEN, RIBEIRO, 2011).

Instrumentalização: Os instrumentos utilizados foram um repositório online de dados, para compartilhar os resultados das atividades. Um aplicativo Kanban online, e uma ferramenta para o desenho do processo em BPMN.

Avaliação da Validade: Existem 4 tipos de validações, interna, externa, construtiva e conclusiva, segundo Wohlin *et al.* (2012). O experimento foi avaliado utilizando a validade externa, por se tratar da generalização dos resultados, a que melhor se enquadra nos objetivos deste experimento.

4.3 OPERACIONALIZAÇÃO

O subprocesso de operacionalização tem três etapas básicas de acordo com Wohlin *et al.* (2012) a preparação e a execução do experimento, e a validação dos dados. Estas, são descritas a seguir.

Preparação: A etapa de preparação do experimento tem como função organizar o decorrer do experimento. Participaram deste um total de 82 pessoas, entre 19 e 45 anos, estas foram divididas em 4 etapas (repetições) diferentes do experimento, sendo que uma foi caracterizada como grupo de controle, na qual não foi utilizado o processo proposto, servindo assim como um parâmetro para comparar os efeitos da aplicação do processo. A cada etapa houve a participação de aproximadamente 20 pessoas. Para cada repetição os participantes foram divididos em 4 equipes distribuídas (sites de desenvolvimento) com

um gerente de projetos. É importante salientar que não houve contato pessoal entre as equipes no decorrer do experimento, simulando realmente o ambiente de distribuído.

Houve um treinamento para a participação, de 4 horas para cada equipe, também a de controle, a diferença é que pra estes não houve treinamento com as ferramentas. O intuito do treinamento, como um todo, foi explicar sobre como seriam atribuídas as responsabilidades e como seria realizado o monitoramento do cronograma das atividades no decorrer do projeto. Quanto a preparação do experimento, esta também consistiu em definir os elementos e as ações do processo de monitoramento. A definição dos elementos constituiu-se em delimitar quem seria o gerente de processos e o de projetos. Quem ficou a encargo, de ser o gerente de processos, foi o autor do experimento. Como este não está acontecendo dentro de uma empresa, não há base histórica, por isso o processo foi sistematizado.

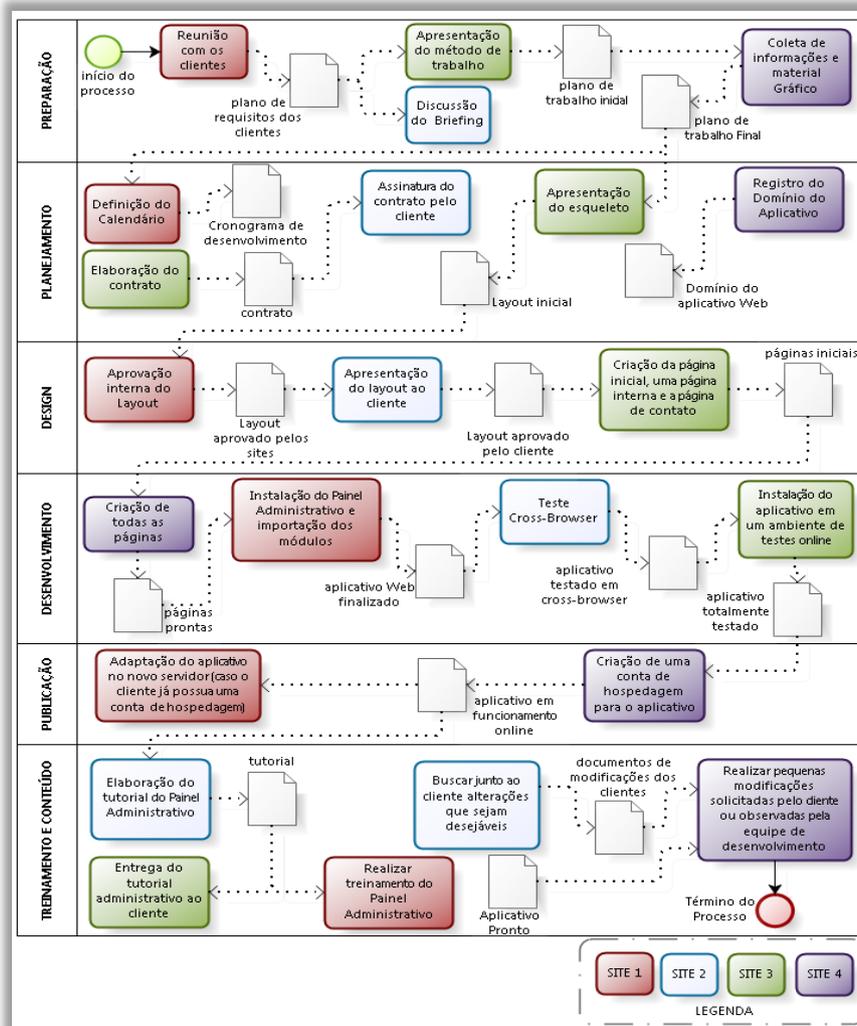
Para realizar o papel de gerente de projetos foi selecionado, com base na aceitação do participante e casualidade, um a cada repetição do experimento realizado. Este definiu os papéis de cada participante no experimento, assim como o agrupamento das equipes. As ações referentes ao processo de monitoramento foram desenvolvidas parcialmente, pois existem algumas que são preliminares a execução do experimento, enquanto outras devem ser executados no decorrer do mesmo. As ações do processo de monitoramento que foram realizadas antes do início da experimentação, correspondem a:

1. A solicitação do produto (agente externo): realizada pelos autores do trabalho em conjunto a um grupo de uma Associação de Caridade (ONG – Organização não Governamental), (agentes externos ao projeto): O produto a ser desenvolvido é um aplicativo Web, para uma Instituição sem fins lucrativos da cidade de Bandeirantes – PR.

2. Especificar um subprocesso de execução que seja adequado ao produto a ser desenvolvido (gerente de processos): O Subprocesso de execução foi elaborado a partir de pesquisas em padrões de projetos para área de desenvolvimento *Web*. O subprocesso de execução utilizado para o experimento possui 6 etapas, 23 atividades que geram 18 artefatos, dentre estes o produto final. As atividades foram divididas entre os sites de acordo com a afinidade, o conhecimento que cada desenvolvedor possui e como o gerente de projetos julgou correto. Foi proposto então aos sites a produção do aplicativo web que contemplasse as atividades e gerasse os artefatos contidos no subprocesso descrito.

3. Representar o subprocesso em BPMN (gerente de processos): A partir da definição do Subprocesso de execução, foi elaborada sua representação em BPMN, esta pode ser observada na Figura 3.

Figura 3 - Processo de execução em BPMN



Fonte: Autores

Foi apresentado nesta seção as ações realizadas antes da inicialização do experimento, as demais foram realizadas durante o andamento deste, e são descritas na seção de Execução. Também durante a preparação do experimento, foi aplicada a métrica *Ideal Day*, para cada atividade do subprocesso de execução. Para aplicação desta métrica foi consultado um analista de sistemas com experiência na área e atuante em empresa de desenvolvimento. O cálculo *Ideal Day*, juntamente com o tempo estimado para realização de cada atividade, assim como para realização total do projeto pode ser observado na Tabela 1. A porcentagem que corresponde a variável IED_REAL(%), é a fração do dia,

em horas que o desenvolvedor dedica a atividade. Exemplo: 33% do dia é aproximadamente 8 horas, empregadas para o desenvolvimento da atividade.

Tabela 1 – Cálculo do tempo com a métrica IDEAL DAY

ID Atividade	Definição das Variáveis	Tempo gasto em dias	Tempo gasto em horas	ID Atividade	Definição das Variáveis	Tempo gasto em dias	Tempo gasto em horas
1	IED: 1 IED_REAL (%): 0,33	1,5	11,9	12	IED: 4 IED_REAL (%): 0,33	5,6	44,8
2	IED: 1 IED_REAL (%): 0,33	1,5	11,9	13	IED: 8 IED_REAL (%): 0,33	9,7	77,6
3	IED: 2 IED_REAL (%): 0,33	3,0	23,9	14	IED: 5 IED_REAL (%): 0,33	6,7	53,7
4	IED: 2 IED_REAL (%): 0,33	3,0	23,9	15	IED: 4 IED_REAL (%): 0,33	6,0	47,8
5	IED: 2 IED_REAL (%): 0,25	2,5	15,2	16	IED: 2 IED_REAL (%): 0,33	3,0	23,9
6	IED: 1 IED_REAL (%): 0,2	1,3	6,0	17	IED: 1 IED_REAL (%): 0,2	1,3	6,0
7	IED: 1 IED_REAL (%): 0,1	1,1	2,7	18	IED: 2 IED_REAL (%): 0,33	2,8	22,7
8	IED: 1 IED_REAL (%): 0,1	1,1	2,7	19	IED: 4 IED_REAL (%): 0,33	5,6	44,8
9	IED: 1 IED_REAL (%): 0,1	1,1	2,7	20	IED: 1 IED_REAL (%): 0,2	1,3	6,0
10	IED: 1 IED_REAL (%): 0,1	1,1	2,7	21	IED: 1 IED_REAL (%): 0,33	1,5	11,9
11	IED: 1 IED_REAL (%): 0,2	1,3	6,0	22	IED: 1 IED_REAL (%): 0,2	1,3	6,0
				23	IED: 4 IED_REAL (%): 0,33	4,5	35,8
				TOTAL	67,6	490,5	

Fonte: Autores

Para este trabalho o tempo será analisado individualmente para cada atividade, então a mensuração em horas é mais oportuna, se o cronograma geral fosse o foco, o período em dias seria um critério importante a se observar, mas este não é o caso. Existem algumas atividades também, que poderiam ser realizadas em paralelo, contudo o que se pretende analisar no experimento, não é o tempo total gasto no projeto, mas sim é o tempo individual de cada atividade, e se o processo proposto, com a aplicação do Kanban e do BPMN, auxiliam para que o cronograma seja cumprido efetivamente.

Execução: Antes do início do experimento todos os participantes foram reunidos, treinados para utilização das ferramentas e também para validar os modelos de artefatos disponibilizados, assim como o escopo do projeto foi apresentado, por meio da notação BPNM. Com exceção dos participantes do grupo de controle, que apenas validaram os modelos de artefatos e para explicar o escopo do projeto foi utilizada uma descrição apenas verbal. Foram realizadas 4 repetições do experimento, cada repetição contou com aproximadamente 20 pessoas, divididas em 4 equipes, cada equipe foi montada aleatoriamente pelo gerente de projetos, sua seleção foi realizada por meio de aspecto de casualidade, cada repetição contou com um gerente de projetos. Este definiu a numeração de cada site, e as atividades que cada equipe seria responsável por implementar. As duas primeiras repetições do experimento foram realizadas no primeiro

semestre de 2015, as outras foram realizadas no segundo semestre. Entre treinamento, preparação do ambiente, desenvolvimento e coleta dos dados, foram gastos aproximadamente 2 meses e 20 dias para cada repetição. No decorrer do experimento as atividades do subprocesso de execução foram desenvolvidas de maneira distribuída, não houve interação durante a execução entre as equipes, esta ocorreu apenas por meio de aplicativos online, como e-mail, hangout e/ou Skype. Como foi explanado na seção anterior de preparação, há ações do processo de monitoramento que fazem parte da execução propriamente dita, portanto foram realizadas no decorrer do experimento. Estas correspondem a:

4. Atribuir as responsabilidades (gerente de projetos): Este realizou esta atividade no início do projeto (experimento).

5. Monitorar as atividades na ferramenta do Kanban (gerente de projetos): O gerente de projetos realizou esta atividade amparado pela métrica *Ideal Day*. Assim que uma equipe puxa sua atividade para o desenvolvimento no quadro Kanban, o gerente fica atento aos prazos, e manda 4 alertas para a equipe por e-mail. O primeiro quando se passa 25% do tempo, depois com 50%, com 75%, e quando o tempo está próximo do fim ele cobra o retorno imediato do artefato. O gerente de projetos do grupo de controle, também teve acesso a métrica, contudo se utilizou de outras técnicas, não as propostas no processo para o monitoramento, os quais ficaram a seu critério.

6. Dispor as atividades na ferramenta do Kanban (equipe); e 7. Utilizar a ferramenta do Kanban (equipe); No início do projeto, assim que cada equipe recebeu o subprocesso de execução em BPMN, para visualização do escopo do projeto, cada equipe organizou suas atividades na ferramenta, e foram instruídas e monitoradas pelo gerente de projetos a seguirem as orientações de uso. No caso do grupo de controle isso não ocorreu, pois eles não utilizaram essas ferramentas.

8. Disponibilizar os artefatos gerados, para dar sequência ao projeto (equipe): As equipes ao terminarem cada artefato os postavam em uma plataforma Moodle (configurando o subprocesso de trabalho), para o compartilhamento, e para que as outras equipes, que fossem desenvolver atividades que necessitavam destes artefatos como entrada, pudessem acessá-los, assim como o gerente teria livre acesso para verificá-los.

É relevante expor que a ferramenta que implementa a técnica do Kanban utilizada tem a funcionalidade de contagem de tempo, que auxiliou o gerente de projetos em suas atividades. Outro fato relevante é que para o grupo de controle, em que não houve

a utilização das ferramentas para o Kanban e o BPMN, o gerente de projeto foi orientado a realizar as atividades de monitoramento por meio de outras vias como por dispositivos e/ou aplicativos móveis, assim como por e-mail ou Skype. Ele teve liberdade para realizar o monitoramento, as ferramentas foram apenas sugeridas. Todavia, a não utilização do processo, o restante das condições foi idêntico as das outras repetições do experimento.

Validação dos dados: Para cada repetição do experimento foram coletados os tempos em horas, que as equipes empregaram no desenvolvimento das atividades, do subprocesso de execução. O tempo de dedicação diária para cada uma delas corresponde ao que foi estabelecido pela métrica *Ideal Day* utilizada. Não foi necessário realizar cortes em dados, todos ficaram relativamente homogêneos, algumas das durações das atividades foram aproximadas, para melhor entendimento. Alguns critérios, de acordo com Piattini *et al.* (2014) foram aplicados para comprovar a validade dos dados, estes são:

Exatidão – refere-se ao grau de concordância entre o resultado de uma medição e um valor verdadeiro do mensurando.

Repetitividade – refere-se ao grau de concordância entre os resultados de medições sucessivas de um mesmo mensurando, efetuadas sob as mesmas condições de medição, chamadas de condições de repetitividade.

Robustez – é a medida da sua capacidade de permanecer inalterado frente a pequenas, mas deliberadas, variações dos parâmetros associados ao método, demonstrando sua confiabilidade durante seu uso na rotina de trabalho.

5 ANÁLISE E INTERPRETAÇÃO DOS DADOS

Depois do experimento e do levantamento dos dados, segundo Piattini *et al.* (2014) devem ser realizadas sua correta análise e interpretação, nas quais é empregado a abordagem quantitativa, que define uma metodologia para mensuração de dados numéricos, classificados e analisados utilizando-se técnicas estatísticas (PIATTINI *et al.*, 2014).

5.1 ESTATÍSTICAS DESCRITIVAS

Os dados coletados durante as repetições e com o grupo de controle podem ser observados na Tabela 2.

Tabela 2 – Dados coletados no decorrer do experimento

ID Atividades	Grupo de Controle	Repetição 1	Repetição 2	Repetição 3
1	13	10	12	8
2	13,5	11,5	11	9
3	13	9	11,5	10
4	26	22	24	23
5	18	16	15	14
6	6,5	5	6	5,5
7	3,5	1,5	2,5	2
8	2,5	2,5	2,5	2,5
9	2,7	2	2	2,5
10	2,8	2,2	2,2	2,6
11	7,5	5	5	5
12	50	45	47	46
13	97	90	95	92
14	61	60	60	59
15	49	45	47	44
16	24,5	23	23	24
17	8	3	6	5,5
18	26,5	21	24	23
19	48	48	48	45
20	6	5	6	5,5
21	12,5	11,5	11	10,5
22	7	6	6	6
23	48	48	47	46
Total	541	492,2	513,7	490,6

Fonte: Autores

A partir destes dados foi elaborada uma média dos tempos gastos nas atividades para cada repetição e depois comparados com o grupo de controle e com a métrica utilizada como parâmetro, estes critérios podem ser analisados na Tabela 3. Com pode ser percebido, na Tabela 3 com a comparação, tanto individualmente de cada atividade, como no total do tempo gasto para realização das mesmas, nas repetições, em que houve o uso do processo proposto houve um atraso pequeno com relação a métrica utilizada, de aproximadamente 8 horas. Enquanto que no grupo de controle houve atraso de aproximadamente 50 horas. O que leva a crer que o processo proposto, é eficaz no monitoramento do cronograma das atividades em projetos distribuídos.

Tabela 3 – Comparação das repetições, com a métrica e o grupo de controle

ID Atividades	Média das Repetições	Grupo de Controle	Métrica Ideal Day
1	10	13	11,9
2	10,5	13,5	11,9
3	10,2	12,5	23,9
4	23	25	23,9
5	15	17	15,2
6	5,5	6,5	6,0
7	2	3	2,7
8	2,5	2,5	2,7
9	2,2	2,7	2,7
10	2,3	2,8	2,7

11	5	7	6,0
12	46	50	44,8
13	92,3	97	77,6
14	59,7	61	53,7
15	45,3	48,5	47,8
16	23,3	24	23,9
17	4,8	7	6,0
18	22,7	26,5	22,7
19	47	48	44,8
20	5,5	6	6,0
21	11	12,5	11,9
22	6	7	6,0
23	47	48	35,8
TOTAL	498,8	541,0	490,5

Fonte: Autores

5.2 PROVA DAS HIPÓTESES

A primeira hipótese formulada foi a **H₀**: Não é possível monitorar o cronograma de atividades de um projeto distribuído de software utilizando um processo que aplica a técnica Kanban e a modelagem BPMN. Esta corresponde a Hipótese nula, a qual o experimento tem o intuito de refutá-la. Ficou claro com a sistematização dos dados coletados, nas três repetições do experimento, que é possível monitorar o cronograma das atividades com o processo proposto. Pode-se notar este fato com as comparações dos dados do tempo gasto nas atividades realizadas no experimento, com o critério utilizado como base de análise. Desta forma pode-se afirmar que a **H₀** não é válida. Isto posto, fica averiguado que a hipótese **H₁**: Um projeto distribuído de software pode ter o cronograma de suas atividades monitoradas, por meio de um processo que aplique a técnica do Kanban e a modelagem BPMN é válida.

Também é possível averiguar a comprovação de **H₁**, com relação aos dados apurados na execução do experimento pelo grupo de controle, que não utilizou o processo e tiveram um atraso considerável comparado com o parâmetro estabelecido. Na avaliação da validade proposta na metodologia foi definido que os resultados seriam comprovados, por meio da Validade Externa (generalização dos resultados), desta forma como o experimento foi realizado 4 vezes com pessoas que são da área de desenvolvimento de software, e a quantidade de participantes estudados é uma parcela de perfil consideravelmente diverso, e um número razoável de indivíduos da população pesquisada, é plausível estender tais afirmações para outros grupos de participantes e profissionais da indústria.

6 CONSIDERAÇÕES FINAIS

Com a crescente adoção do DDS por inúmeras organizações, este tem atraído um grande número de pesquisas na área de Engenharia de Software. Em um projeto de software distribuído ou centralizado o monitoramento das atividades é indispensável para que se obtenha sucesso. Em particular no DDS, Wiredu (2005) constata, que este é um desafio, porque inclui a coordenação de interações entre pessoas, processos, informações e tecnologias distribuídas. Diante dos fatos citados, sua relativa importância e a dificuldade de sua realização em projetos distribuídos, foi escolhido focar este trabalho no monitoramento de projetos de software em DDS, porque este possibilita o aumento na velocidade do desenvolvimento e da qualidade, facilita o acompanhamento, minimiza os riscos e melhora a relação entre os colaboradores e com o cliente.

No contexto do monitoramento de projetos de software uma técnica ganhou notoriedade a partir da ascensão das metodologias ágeis, o Kanban (AHMAD; MARKKULA; OVIO, 2013) assim como o BPMN, vem se destacando em pesquisas sobre engenharia de software. Motivado por essas informações, e também por ter o conhecimento que a técnica Kanban não é aplicada com frequência no DDS (AHMAD; MARKKULA; OVIO, 2013) da mesma maneira que, objetivando complementar as pesquisas sobre melhorias de projetos de software em ambientes distribuídos, este trabalho teve o intuito de comprovar a seguinte hipótese: Um projeto distribuído de software pode ter o cronograma de suas atividades monitoradas, por meio de um processo que aplique a técnica do Kanban e a modelagem BPMN. Para isto, foi desenvolvido um experimento que oportunizou a validação da hipótese formulada. Para que fosse possível cumprir com o objetivo da pesquisa, como citado, foi necessário o desenvolvimento de um experimento. O foco do monitoramento foi o cronograma individual das atividades. As medições realizadas no experimento diziam respeito aos tempos gastos pelos participantes na realização de cada atividade. Como parâmetro de comparação, dos dados coletados no experimento, com o que deveria ser executado, foi utilizada uma métrica para medição de tempo. Essa métrica denomina-se *Ideal Day*.

Na preparação do experimento foi sistematizado um projeto para o desenvolvimento de um aplicativo Web. Participaram 82 pessoas no experimento desenvolvido, que foi realizado em 4 etapas, 3 repetições utilizando as ferramentas, e um grupo de controle. Havia aproximadamente 20 pessoas por repetição e também no grupo de controle. Na primeira repetição utilizando o processo, foi possível perceber um atraso de tempo de aproximadamente 2 horas comparado a métrica. A segunda repetição

apresentou por volta de 23 horas de atraso. Na terceira repetição do experimento houve a maior semelhança com o que foi definido pela métrica, particularmente o tempo previsto. Enquanto que no grupo de controle, que não fez uso do processo, o tempo gasto excedeu aproximadamente 50 horas, do que foi previsto pela *Ideal Day*.

Com as análises estatísticas empregadas na comparação das 3 repetições do experimento, com grupo de controle, e a métrica utilizada como parâmetro. Foi possível comprovar a hipótese levantada, como objetivo do trabalho realizado. Desta forma, pode-se definir que a principal contribuição desta pesquisa foi validar que é possível monitorar o cronograma das atividades de um projeto distribuído de software, utilizando um processo que aplica as técnicas do Kanban e a modelagem BPMN. O processo proposto como instrumento para a validação da hipótese também se caracteriza como uma contribuição para a área da Engenharia de Software. Tal contribuição pode incentivar gerentes de projetos distribuídos a aplicar a técnica do Kanban em DDS, esta que não tem expressividade neste ambiente de desenvolvimento.

6.1 LIMITAÇÕES DA PESQUISA

Mesmo se mostrando eficiente no monitoramento do cronograma das atividades de um projeto distribuído de software, o processo juntamente as ferramentas utilizadas, não foram testados em um contexto no qual se leva em consideração os artefatos gerados no decorrer do projeto. Apenas algumas incoerências mais perceptíveis foram corrigidas nestes artefatos. Desta forma, não se pode levar em contas os resultados das atividades como perfeitos no decorrer do projeto.

Outro aspecto, que caracteriza uma limitação da pesquisa, é o caráter empírico da métrica que foi utilizada como parâmetro para medição do tempo. Se esta métrica fosse calculada por analistas de outras empresas, provavelmente os resultados para os tempos previstos seriam divergentes. Isto dependendo da seriedade de aplicação do cálculo da métrica pode assinalar um problema na validação dos dados coletados. Todavia, nesta pesquisa o cálculo da métrica feito para prever o cronograma das atividades, foi realizado junto a um analista de sistemas experiente, o que evidencia maior confiabilidade nos resultados obtidos. Quanto a generalização dos dados, apesar de ter sido afirmado, que a parcela da população ao qual participou do experimento é significativa, isto causa preocupação, pois este foi realizado em instituições de ensino superior do norte do Paraná-Brasil. Se fosse executado em instituições de outros estados, provavelmente os

resultados seriam diferentes, por isso o cuidado em se generalizar a toda população de profissionais.

6.2 PESQUISAS FUTURAS

Identifica-se um grande potencial de crescimento nesta linha de pesquisa, como dito no início desta seção o Kanban não é uma técnica habitualmente utilizada em DDS, e com sua grande utilização junto a metodologias ágeis, este pode auxiliar na boa condução de projetos distribuídos, em ambientes industriais. Deste Modo, foi possível detectar, com a aplicação do processo proposto no experimento, algumas pesquisas futuras: (1) Utilização do processo junto a outros tipos de métricas, para monitoramento da qualidade do artefato gerado e do produto final. Também a aplicação de métricas que auxiliem em monitorar a produtividade das equips; (2) Aplicação do processo em um contexto empresarial, nas indústrias de software, para validar a hipótese levantada em um contexto real desenvolvimento; (3) Desenvolvimento de uma métrica que verificasse o cronograma das atividades e a qualidade dos artefatos gerados em termos numéricos, para ser aplicado junto ao processo proposto.

REFERÊNCIAS

AHMAD, M. O.; MARKKULA, J.; OVIO, M. Kanban in software development: A systematic literature review. **In 39th Euromicro Conference Series on Software Engineering and Advanced Applications**. Santander, Spain, p. 9-16 (2013).

ABNT - Associação Brasileira De Normas Técnicas. **ABNT NBR ISO 9001:2008 - Sistemas de gestão da qualidade - Requisitos**. Rio de Janeiro, Brasil (2008).

BPMN. **Business Process Modeling Notation (BPMN) Information**. OMG, 2015. Disponível em: <<http://www.bpmn.org>>. Acesso em: 05 jun. (2015).

CATEN, C. S.; RIBEIRO, J. L. D. **Projeto de experimentos**. Porto Alegre: Universidade Federal do Rio Grande do Sul, Escola de Engenharia, Programa de Pós-Graduação em Engenharia de Produção (2011)

CRUZ, T. Sistemas, métodos & processos: **administrando organizações por meio de processos de negócio**. Atlas, São Paulo, Brasil, 304 p. (2005).

HERBSLEB, J. D.; MOCKUS, A. An empirical study of speed and communication in globally distributed software development. **IEEE Transactions on Software Engineering**. EUA, v.29, n.6, p.481-494 (2003).

JONES, C. Social and Technical Reasons Software Project Failures. **CrossTalk: The Journal of Defense Software Engineering**, June (2006).

L'ERARIO, A. M3DS: **Um modelo de dinâmica de desenvolvimento distribuído de software**. 175 p. Tese (doutorado) – Escola Politécnica da Universidade de São Paulo (USP), São Paulo, Brasil (2009).

LESCHER, C.; Li, Y.; BRUEGGE, B. Teaching global software engineering: Interactive exercises for the classroom. **IN: 9th IEEE International Conference on Global Software Engineering, ICGSE**, p. 163-172, Shanghai, China (2014).

MARTINS, J. C. C. **Técnicas para Gerenciamento de Projetos de Software**. 1. ed. Florianópolis: Brasport Editora, 465 p. (2001).

PIATTINI, V. M. G.; GENERO B.; Cruz-Lemus, J. A. **Métodos de Investigación en Ingeniería Del Software**. RA-MA, 1 ed, 314 p. Espanha (2014).

SCHARFF, C. An evolving collaborative model of working in students' global software development projects. **IN: Workshop on Collaborative Teaching of Globally Distributed Software Development, ICSE**, Honolulu, Hawaii, p. 11-15 (2011).

SENGUPTA, B.; Chandra, S.; Sinhá, V. A. Research Agenda for Distributed Software Development. **IN: ICSE. Proceedings**, Shangai, China. p. 731-740 (2006).

TRAVASSOS, G. H.; GUROV, D.; AMARAL, E.. A. G. **Introdução a Engenharia de Software Experimental**. Relatório Técnico, Programa de Engenharia de Sistemas e Computação, COPPE/UFRJ, 66 p. (2002).

WHITE, S. A. **Using BPMN to Model a BPEL Process**. IBM, New York, 2006. Disponível em: <<http://www-106.ibm.com/developerworks/webservices/library/ws-bpel/>>. Acesso em: 05 jun. (2015).

WIREDU, G. O. Coordination as the Challenge of Distributed Software development. **IN: Workshop of Distributed Software Development.** Paris. France (2005).

WOHLIN, C. **Experimentation in Software Engineering: an introduction.** Kluwer Editors Academics, EUA (2012).